

Universidad Nacional de Ingeniería

Escuela de posgrado



TESIS

**MODELO PREDICTIVO DEL DESEMPEÑO INSTANTÁNEO DE
MOTORES DE COMBUSTIÓN INTERNA EN APLICACIONES
INDUSTRIALES BASADO EN MACHINE LEARNING**

Para obtener el grado de Doctor en Ciencias con mención en Energética

Elaborado por

César Elías Mendoza Suárez

Asesor

Dr. Juan Guillermo Lira Cacho

Co-Asesor

Dr. Ricardo Rodríguez Bustinza

LIMA – PERÚ

2024

Dedicatoria

A Dios, por todo lo bueno que me dio en la vida. A mis padres, de quienes aprendí mis primeras letras y el amor a las ciencias, sin los cuales no hubiera podido finalizar este trabajo.

Agradecimientos

A mi asesor y co-asesor, quienes revisaron e hicieron hincapié en mejorar este trabajo.

Resumen

En el desarrollo de los procesos industriales modernos impulsados por motores de combustión interna, es crucial mantener un monitoreo constante de su desempeño para prever acciones de operación, mantenimiento y producción que impacten de manera relevante en su aplicación. En este sentido esta investigación tiene por objetivo determinar de qué manera un modelo de pronóstico basado en machine learning logra la predicción del desempeño instantáneo de un motor de combustión interna de aplicación industrial.

Para esta investigación se empleó un diseño no experimental y de corte transversal, partiendo del registro del comportamiento de un motor de combustión interna de encendido por compresión (motor diésel) de inyección directa, turboalimentado, equipado con un sistema electrónico de suministro de combustible del tipo "common rail", al cual se le sometió a pruebas de carga en un banco dinamométrico. Una vez formada la base de datos, se procedió al análisis y ejecución de modelos de regresión en machine learning para evaluar la predicción de los parámetros de potencia, torque, consumo de combustible y factor de carga. Posteriormente, desarrollado el modelo adecuado, se procedió a la validación del factor de carga, frente a condiciones de operación reales..

Los resultados de la investigación permitieron determinar los parámetros de entrada más relevantes en el proceso de predicción, así como también la eficacia de los modelos evaluados (Multiple Linear Regresion, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, Artificial Neural Network y XGboost) mediante métricas de desempeño como MAE, MSE, RMSE y R^2 . En el caso del torque, el modelo XGBoost presentó el mejor desempeño, con un RMSE de 0,092 y el R^2 de 0,991, seguido por el modelo Random Forest (RF), que obtuvo un RMSE de 0,095 y un R^2 de 0,99. Para la potencia, el mejor desempeño correspondió al modelo Random Forest (RF), con un RMSE de 0,081 y un R^2 de 0,993, seguido de XGBoost, con un RMSE de 0,098 y un R^2 de 0,990. En cuanto al consumo de combustible, el modelo Random Forest mostró el mejor desempeño, con un RMSE de 0,065 y un R^2 de 0,996, seguido por XGBoost, que alcanzó un RMSE de 0,083 y un R^2 de 0,993. Sin embargo, cabe destacar que la carga computacional del modelo Random Forest es considerablemente elevada. Los demás modelos evaluados no lograron resultados relevantes en comparación.

Finalmente se concluye que el modelo basado en XGBoost es el más adecuado para la predicción del desempeño instantáneo en motores de combustión interna destinados a aplicaciones industriales.

Palabras clave — Inteligencia artificial, machine learning, motor diésel, common rail, potencia, torque, consumo de combustible, XGBoost.

Abstract

In the development of modern industrial processes driven by internal combustion engines, it is crucial to maintain constant monitoring of their performance to foresee operation, maintenance and production actions that have a relevant impact on their application. In this sense, this research aims to determine how a forecasting model based on machine learning achieves the prediction of the instantaneous performance of an internal combustion engine for industrial application.

For this research, a non-experimental cross-sectional was used, based on the recording of the behavior of a direct injection, turbocharged, compression ignition internal combustion engine (diesel engine), equipped with an electronic fuel supply system. "common rail" type fuel, which was subjected to load tests on a dynamometric bench. Once the database was formed, the analysis and execution of regression models in machine learning were carried out to evaluate the prediction of the parameters of power, torque, fuel consumption and load factor. Subsequently, having developed the appropriate model, we proceeded to the validation of the load factor, against real operating conditions..

The results of the research allowed to determine the most relevant input parameters in the prediction process, as well as the effectiveness of the evaluated models (Multiple Linear Regression, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, Artificial Neural Network and XGboost) through performance metrics such as MAE, MSE, RMSE and R². In the case of torque, the XGBoost model presented the best performance, with an RMSE of 0.092 and R² of 0.991, followed by the Random Forest (RF) model, which obtained an RMSE of 0.095 and an R² of 0.99. For power, the best performance corresponded to the Random Forest (RF) model, with an RMSE of 0.081 and an R² of 0.993, followed by XGBoost, with an RMSE of 0.098 and an R² of 0.990. In terms of fuel consumption, the Random Forest model showed the best performance, with an RMSE of 0.065 and an R² of 0.996, followed by XGBoost, which achieved an RMSE of 0.083 and an R² of 0.993. However, it should be noted that the computational load of the Random Forest model is considerably high. The other models evaluated did not achieve relevant results in comparison. Keywords — Artificial intelligence, machine learning, diesel engine, common rail, power, torque, fuel consumption, XGBoost.

Finally, it is concluded that the model based on XGBoost is the most suitable for predicting instantaneous performance in internal combustion engines intended for industrial applications.

Keywords — Artificial intelligence, machine learning, diesel engine, common rail, power, torque, fuel consumption, XGBoost.

Tabla de Contenido

Resumen	iv
Abstract	vi
Tabla de Contenido	viii
Lista de Tablas	xiii
Lista de Figuras	xv
Capítulo I. Introducción	1
1.1. Generalidades.....	1
1.2. Descripción del problema de investigación.....	1
1.3. Objetivos del estudio.....	5
1.3.1. Objetivo general	5
1.3.2. Objetivos específicos.....	5
1.4. Antecedentes investigativos.....	6
1.5. Justificación e importancia de la investigación	15
1.6. Hipótesis	17
1.6.1. Hipótesis general.....	17
1.6.2. Hipótesis específicas.....	17
1.7. Variables e indicadores	17
1.7.1. Variable 1	17
1.7.2. Variable 2	18
1.8. Tipo y nivel de la investigación.....	18
1.9. Muestra.....	19
Capítulo II. Marcos teórico y conceptual	20
2.1. Marco teórico	20
2.1.1. La inteligencia artificial.....	20
2.1.2. El machine learning o aprendizaje automático.....	21
2.1.3. Tipos de machine learning.....	23
2.1.4. Regresión y clasificación	24

2.1.5. Algoritmos fundamentales de aprendizaje supervisado	24
2.1.6. Regresión lineal.....	25
2.1.7. K-vecinos más cercanos (KNN).....	26
2.1.8. Máquina de vectores soporte (SVM).....	28
2.1.9. Árboles de decisión	30
2.1.10. Bosques aleatorios	31
2.1.11. Redes neuronales	32
2.1.12. XGBoost.....	34
2.1.13. Reducción de dimensionalidad	35
2.1.14. Normalización y estandarización de datos.....	36
2.1.15. Evaluación de la calidad de la predicción	37
2.1.16. Python y el machine learning.....	38
2.1.17. Motor de combustión interna	40
2.1.18. Parámetros indicados de un motor de combustión interna.....	40
2.1.19. Parámetros efectivos de un motor de combustión interna.....	42
2.1.20. Torque al freno y Potencia al freno	43
2.1.21. Eficiencia mecánica.....	45
2.1.22. Potencia efectiva del motor.....	45
2.1.23. Factor de carga	46
2.2. Marco conceptual.....	48
Capítulo III. Desarrollo del trabajo de investigación	49
3.1. Panorama del desarrollo de la investigación	49
3.2. Banco de ensayos.....	51
3.2.1. Motor por evaluar	51
3.2.2. Freno dinamométrico.....	52
3.2.3. Instrumentación	52
3.3. Proceso de registros de los monitoreos	55
3.4. Parámetros disponibles en el controlador de motor.....	55
3.5. Conformación de la base de datos.....	57

3.6.	Procesamiento de datos.....	58
3.6.1.	Evaluación de los datos registrados	59
3.6.2.	Normalización.....	59
3.6.3.	Análisis correlacional	60
3.6.4.	Establecimiento del parámetro de entrada al modelo	64
3.6.5.	Establecimiento de los parámetros de salida del modelo.....	64
3.7.	Desarrollo de los modelos de predicción	65
3.7.1.	Entrenamiento de la Regresión Lineal Múltiple (MLR)	66
3.7.1.	Entrenamiento de K-Vecinos más cercanos (KNN) en regresión	67
3.7.2.	Entrenamiento de Maquina de vector soporte (SVM) en Regresión.....	67
3.7.3.	Entrenamiento de árboles de decisión (DT) en regresión	68
3.7.4.	Entrenamiento de bosques aleatorios (RF) en regresión	69
3.7.5.	Entrenamiento de redes neuronales (ANN)	70
3.7.6.	Entrenamiento del modelo XGBoost.....	71
3.8.	Determinación del factor de carga.....	71
Capítulo IV. Análisis y discusión de resultados		72
4.1.	Análisis descriptivo.....	72
4.1.1.	Variable dependiente: Desempeño del motor	72
4.1.2.	Dimensión 1: Torque instantáneo	72
4.1.3.	Dimensión 2: Potencia instantánea.....	73
4.1.4.	Dimensión 3: Consumo instantáneo de combustible.....	73
4.1.5.	Dimensión 4: Factor de carga instantáneo.....	74
4.1.6.	Variable dependiente: Desempeño pronosticado del motor	75
4.1.7.	Dimensión 1: Torque pronosticado	75
4.1.8.	Dimensión 2: Potencia pronosticada.....	75
4.1.9.	Dimensión 3: Consumo de combustible pronosticado.....	76
4.1.10.	Dimensión 4: Factor de carga pronosticado.....	77
4.2.	Evaluación de modelos	77
4.2.1.	Pronóstico del modelo de regresión lineal múltiple (MLR)	78

4.2.2. Análisis de significancia de los coeficientes de regresión	81
4.2.3. Verificación de los Supuestos del modelo MLR	84
4.2.4. Análisis del pronóstico del modelo MLR	91
4.2.5. Pronóstico del modelo KNN en regresión	92
4.2.6. Análisis del pronóstico del modelo KNN	95
4.2.7. Pronóstico del modelo máquina de vectores soporte (SVM)	96
4.2.8. Análisis del pronóstico del modelo SVM	97
4.2.9. Pronóstico del modelo de regresión árboles de decisión (DT)	98
4.2.10. Análisis del pronóstico del modelo DT	105
4.2.11. Pronóstico del modelo bosques aleatorios (RF) en regresión	105
4.2.12. Análisis del pronóstico del modelo RF	112
4.2.13. Pronóstico del modelo de redes neuronales artificiales	113
4.2.14. Análisis del pronóstico del modelo ANR	117
4.2.15. Pronóstico del modelo XGBoost	118
4.2.16. Análisis del pronóstico del modelo XGboost	120
4.3. Análisis de los modelos de Pronóstico	120
4.4. Determinación del factor de carga.....	125
4.5. Validación del factor de carga en aplicación del motor	126
4.6. Contrastación de hipótesis	128
4.7. Discusión de resultados	129
Conclusiones	133
Recomendaciones	135
Referencias bibliográficas	136
Anexos	144
Anexo I: Matriz de operacionalización de variables.....	144
Anexo II: Matriz de Consistencia.....	145
Anexo III: Representación del modelo árbol de decisión desarrollado.	146
Anexo IV. Características del motor diésel ensayado	147
Anexo V. Curvas de desempeño de motor ensayado	148

Anexo VI. Codificación - Análisis_de_datos	149
Anexo VII. Codificación – Correlación	154
Anexo VIII. Codificación – Regresión Lineal multiple	155
Anexo IX. Codificación – K vecinos más cercanos.....	162
Anexo X. Codificación - Máquinas de vectores soporte	168
Anexo XI. Codificación - Árboles de decisión.....	174
Anexo XII. Codificación - Bosques aleatorios.....	181
Anexo XIII. Codificación - Redes neuronales	187
Anexo XIV. Codificación - XG Boost	193
Anexo XIV. Base de datos depurado de prueba dinamométrica	199
Anexo XV. Acceso digital a la documentación de la tesis.	231

Lista de Tablas

Tabla 1 Factores típicos de carga de camiones, según Kennedy.	47
Tabla 2 Factores típicos de carga para varios equipos, según Day.	47
Tabla 3 Factores de carga en diferentes condiciones de funcionamiento, según Chitkar.	47
Tabla 4 Características del motor ensayado.	52
Tabla 5 Parámetros del controlador principal del motor SAA12V140E-3.	55
Tabla 6 Parámetros que conforman la base de datos.	58
Tabla 7 Coeficiente de Spearman.	60
Tabla 8 Correlación de Spearman para la dimensión “torque instantáneo”.	62
Tabla 9 Correlación de Spearman para la dimensión “potencia instantánea”.	63
Tabla 10 Correlación para la dimensión “consumo instantáneo de combustible”.	63
Tabla 11 Características de los parámetros de entrada.	64
Tabla 12 Características de los parámetros de salida.	64
Tabla 13 Parámetros de salida mediante MLR con validación cruzada k-folds	78
Tabla 14 Promedio de métricas de evaluación dados por el modelo MLR.	79
Tabla 15 Coeficientes para las ecuaciones de torque, potencia, y consumo.	80
Tabla 16 Evaluación de independencia de los residuos.	85
Tabla 17 Prueba de normalidad. Evolución del p_valor.	85
Tabla 18 Evaluación de linealidad. Con evolución de media de residuos.	87
Tabla 19 Resumen de Verificación de los Supuestos del modelo MLR.	91
Tabla 20 Evaluación de los parámetros pronosticados por el modelo KNN.	92
Tabla 21 Parámetros de salida mediante KNN con validación cruzada k-folds.	93
Tabla 22 Promedio de métricas de evaluación dados por el modelo KNN.	94
Tabla 23 Parámetros de salida mediante SVM con validación cruzada k-folds.	96
Tabla 24 Promedio de métricas de evaluación dados por el modelo SVM.	97
Tabla 25 Parámetros pronosticados por el modelo DT para diferentes niveles de profundidad.	102
Tabla 26 Parámetros de salida mediante DT con validación cruzada k-folds.	103
Tabla 27 Promedio de métricas de evaluación dados por el modelo DT.	104
Tabla 28 Parámetros pronosticados por el modelo RF para diferentes niveles de profundidad.	110
Tabla 29 Parámetros de salida mediante RF con validación cruzada k-folds.	111
Tabla 30 Evaluación de los parámetros pronosticados por el modelo RF.	112
Tabla 31 Evaluación de los parámetros pronosticados por el modelo ANR (8,8).	113

Tabla 32	<i>Evaluación de los parámetros pronosticados por el modelo ANR (10,8,4).</i>	114
Tabla 33	<i>Evaluación de los parámetros pronosticados por el modelo ANR (10,10,10).</i>	.114
Tabla 34	<i>Evaluación de los parámetros pronosticados por el modelo ANR (12,12,10).</i>	.115
Tabla 35	<i>Parámetros de salida mediante ANR con validación cruzada k-folds</i>	116
Tabla 36	<i>Evaluación de los parámetros pronosticados por el modelo ANR.</i>	117
Tabla 37	<i>Parámetros de salida mediante XGboost con validación cruzada k-folds</i>	118
Tabla 38	<i>Evaluación de los parámetros pronosticados por el modelo XGBoost.</i>	119
Tabla 39	<i>Evaluación de los modelos para el pronóstico del torque del motor.</i>	121
Tabla 40	<i>Evaluación de los modelos para el pronóstico de la potencia del motor.</i>	121
Tabla 41	<i>Evaluación de los modelos para el pronóstico del consumo de combustible.</i>	...122
Tabla 42	<i>Tiempo de ejecución de los modelos.</i>	122
Tabla 43	<i>Evaluación del factor de carga para la prueba dinamométrica.</i>	125
Tabla 44	<i>Pronóstico de factor de carga en operación de camiones mineros.</i>	127

Lista de Figuras

Figura 1 <i>Inteligencia artificial, aprendizaje automático y aprendizaje profundo.</i>	21
Figura 2 <i>Predicciones de un modelo lineal.</i>	26
Figura 3 <i>Predicciones hechas por el modelo KNN.</i>	27
Figura 4 <i>Predicciones realizadas por 3 vecinos más cercanos.</i>	27
Figura 5 <i>Clasificación de margen amplio.</i>	29
Figura 6 <i>Estructura de un árbol de decisión.</i>	31
Figura 7 <i>Estructura de una neurona artificial.</i>	33
Figura 8 <i>Interpretación geométrica de la presión media indicada</i>	42
Figura 9 <i>Esquema del principio de operación de un dinamómetro</i>	44
Figura 10 <i>Diagrama de flujo de las actividades de la investigación.</i>	50
Figura 11 <i>Motor diésel Komatsu modelo SAA12V140-3, instalado en el dinamómetro.</i>	51
Figura 12 <i>Detalle del panel de monitoreo del dinamómetro.</i>	53
Figura 13 <i>Hardware de comunicación para el monitoreo del motor diésel.</i>	53
Figura 14 <i>Interfaz de comunicación Inline 6.</i>	54
Figura 15 <i>Software de monitoreo Insite.</i>	54
Figura 16 <i>Importación de las librerías para el procesamiento de datos.</i>	58
Figura 17 <i>Distribución de los datos obtenidos.</i>	59
Figura 18 <i>Código para el desarrollo del mapa de calor de correlaciones.</i>	60
Figura 19 <i>Mapa de calor de la matriz de correlación de los ítem de la Tabla 6</i>	61
Figura 20 <i>Código fuente para el análisis de las correlaciones.</i>	62
Figura 21 <i>Establecimiento de los parámetros y división de conjuntos.</i>	65
Figura 22 <i>Entrenamiento de la regresión lineal múltiple.</i>	66
Figura 23 <i>Entrenamiento de la regresión KNN.</i>	67
Figura 24 <i>Entrenamiento de la regresión SVM.</i>	68
Figura 25 <i>Entrenamiento de la regresión árboles de decisión.</i>	68
Figura 26 <i>Entrenamiento de la regresión bosques aleatorios.</i>	69
Figura 27 <i>Declaración del modelo de redes neuronales.</i>	70
Figura 28 <i>Entrenamiento del modelo de redes neuronales.</i>	70
Figura 29 <i>Entrenamiento del modelo XGBoost.</i>	71
Figura 30 <i>Distribución porcentual de la dimensión torque instantáneo.</i>	72
Figura 31 <i>Distribución porcentual de la dimensión potencia instantánea.</i>	73
Figura 32 <i>Distribución porcentual de consumo instantáneo de combustible.</i>	74
Figura 33 <i>Distribución porcentual de la dimensión factor de carga instantáneo.</i>	74
Figura 34 <i>Distribución porcentual de la dimensión torque pronosticado.</i>	75

Figura 35 Distribución porcentual de la dimensión potencia pronosticada.	76
Figura 36 Distribución porcentual de la dimensión combustible pronosticado.	76
Figura 37 Distribución porcentual del factor de carga (FC) pronosticado.	77
Figura 38 Salidas instantáneas vs salidas pronosticadas mediante modelo MLR	79
Figura 39 Codificación para el análisis de cuadrado mínimos ordinarios	81
Figura 40 Significancia de los coeficientes del torque	82
Figura 41 Significancia de los coeficientes de la potencia.....	83
Figura 42 Significancia de los coeficientes del consumo.....	84
Figura 43 Prueba de normalidad de los residuos del torque	86
Figura 44 Prueba de normalidad de los residuos de la potencia	86
Figura 45 Prueba de normalidad de los residuos del consumo,	87
Figura 46 Linealidad de los residuos del torque	88
Figura 47 Linealidad de los residuos de la potencia.....	88
Figura 48 Linealidad de los residuos del consumo.....	89
Figura 49 Prueba de Homocedasticidad de los residuos del torque.....	89
Figura 50 Prueba de Homocedasticidad de los residuos de la potencia.....	90
Figura 51 Prueba de Homocedasticidad de los residuos del consumo	90
Figura 52 Evaluación de los residuos del modelo MLR.....	91
Figura 53 Registro de R^2 para cada valor de K.	93
Figura 54 Salidas instantáneas vs Salidas pronosticadas para K=4.	95
Figura 55 Evaluación de los residuos del modelo KNN.....	95
Figura 56 Salidas instantáneas vs salidas pronosticadas.	97
Figura 57 Evaluación de los residuos del modelo SVM.....	98
Figura 58 Salidas instantáneas vs salidas pronosticadas, profundidad =2.....	98
Figura 59 Salidas instantáneas vs salidas pronosticadas, profundidad =3.	99
Figura 60 Salidas instantáneas vs salidas pronosticadas, profundidad =4.	99
Figura 61 Salidas instantáneas vs salidas pronosticadas, profundidad =5.	100
Figura 62 Salidas instantáneas vs salidas pronosticadas, profundidad =6.	100
Figura 63 Salidas instantáneas vs salidas pronosticadas, profundidad =7.	101
Figura 64 Salidas instantáneas vs salidas pronosticadas, profundidad =8.....	101
Figura 65 Registro de R^2 para cada nivel de profundidad en modelo DT	102
Figura 66 Representación del modelo árbol de decisión desarrollado.	104
Figura 67 Evaluación de los residuos del modelo DT.....	105
Figura 68 Salidas instantáneas vs salidas pronosticadas, profundidad =2.	106
Figura 69 Salidas instantáneas vs salidas pronosticadas, profundidad =3.....	106
Figura 70 Salidas instantáneas vs salidas pronosticadas, profundidad =4.....	107
Figura 71 Salidas instantáneas vs salidas pronosticadas, profundidad =5.	107

Figura 72 Salidas instantáneas vs salidas pronosticadas, profundidad =6.	108
Figura 73 Salidas instantáneas vs salidas pronosticadas, profundidad =7.	108
Figura 74 Salidas instantáneas vs salidas pronosticadas, profundidad =8.	109
Figura 75 Salidas instantáneas vs salidas pronosticadas, profundidad =9.	109
Figura 76 Evaluación de los residuos del modelo RF.....	112
Figura 77 Salidas instantáneas vs salidas pronosticadas, ANR (8,8).	113
Figura 78 Salidas instantáneas vs salidas pronosticadas, ANR (10,8,4).	114
Figura 79 Salidas instantáneas vs salidas pronosticadas, ANR (10,10,10).	115
Figura 80 Salidas instantáneas vs salidas pronosticadas, ANR (12,12,10).	115
Figura 81 Evaluación de los residuos del modelo ANR.....	117
Figura 82 Salidas instantáneas vs salidas pronosticadas, para el modelo XGBoost.	119
Figura 83 Evaluación de los residuos del modelo XGBoost	120
Figura 84 Características del computador personal empleado.....	123
Figura 85 Importancia de cada característica para el torque pronosticado.....	123
Figura 86 Importancia de cada característica para la potencia pronosticada	124
Figura 87 Importancia de cada característica para el consumo pronosticado	124
Figura 88 Determinación del factor de carga calculado y pronosticado.	125
Figura 89 Resultados factor de carga pronosticado.	126

Capítulo I. Introducción

1.1. Generalidades

Los motores de combustión interna alternativos (MCIA) son ampliamente utilizados en la industria moderna, en sectores como la agricultura, el transporte, la generación de energía, la construcción y la minería, entre otros. Sin embargo, a pesar de la constante evolución tecnológica que han experimentado desde su invención en el siglo XIX, que ha mejorado su rendimiento, fiabilidad, longevidad y reducción de emisiones de manera sostenible en las últimas décadas, éstos siguen siendo agentes que impactan negativamente en el ambiente. Su responsabilidad en la contaminación del aire de las ciudades, el incremento de los gases de efecto invernadero, así como las consecuencias directas de estas en la salud humana, el ecosistema mundial y el calentamiento global, han sido demostradas por diversos estudios (Abed et al., 2019; Fayyazbakhsh et al., 2022; Feng et al., 2022; Fiebig et al., 2014; Gao et al., 2019 ; Greim, 2019; Lapuerta et al., 2008; Ma et al., 2021; Najafi et al., 2019; Reşitoğlu et al., 2015; Yadav et al., 2023; Zhang et al., 2022).

Por otro lado, la industria del transporte, manufactura, extracción, construcción y generación, entre otras, que dependen de motores de combustión interna, exigen una mayor eficiencia energética, reflejada en una mayor potencia disponible con el menor consumo específico de combustible, un requerimiento que también impulsa el desarrollo de nuevos diseños. Por este motivo, existe un interés creciente en estudiar el comportamiento, desarrollo y modelado de los motores de combustión interna, ya sean de encendido por compresión empleando combustible diésel (Billerot et al., 2023; Feng et al., 2019; Gogoi & Baruah, 2010; Harris & Pearce, 1990; Nezloui et al., 2023; Reşitoğlu et al., 2015); Sun et al., 2017); o de los motores de encendido por chispa (Carreño, 2016; Gutiérrez et al., 2020; Krebs & Biet, 2021; Chen et al., 2014; Urroz et al., 2023; Wang et al., 2019). Las investigaciones se mantienen con el propósito de mejorar el diseño y proponer desarrollos en los motores de combustión interna (Carreño, 2016), lo que beneficia a la industria moderna, que depende principalmente de motores de encendido por compresión, comúnmente llamados motores diésel, nombre que llevan en honor a su inventor.

1.2. Descripción del problema de investigación

En la industria minera, donde los equipos de movimiento de mineral están equipados con motores de combustión interna, especialmente con motores diésel, el

impacto de su operación afecta la productividad del proceso, los costos operativos y el ambiente. Estos factores impulsan el interés por estudiar su comportamiento y desempeño.

Los estudios realizados en unidades mineras de tajo abierto en Estados Unidos, donde se ha monitoreado el desempeño de motores de combustión en diversos equipos de aplicaciones mineras, han demostrado que estos tienen impactos importantes en los costos y el ambiente. Así, los tractores de carrilera consumen una gran cantidad de combustible diésel y, en consecuencia, producen una cantidad significativa de CO₂ (Kecojevic & Komljenovic, 2011b). De igual manera, los camiones de acarreo, que son una parte integral del proceso productivo de minería a cielo abierto, consumen una cantidad sustancial de combustible, lo que también genera una cantidad considerable de CO₂. Como consecuencia, los costos ambientales y económicos relacionados con el consumo de combustible y las emisiones de CO₂ representan un desafío sustancial para la industria minera (Kecojevic & Komljenovic, 2011a). Esta característica se extiende a la operación de otras maquinarias de minería, que por su volumen de producción suelen ser equipos gigantes, propulsados por motores diésel de gran potencia.

Durante la operación de los motores diésel, se destacan las características de su desempeño debido a que trabajan a diversos regímenes y grados de exigencia, según la aplicación de la máquina que propulsan. Su operación es demandante, y cuanto más exigidos están, sus parámetros de desempeño, consumo de combustible y emisiones de CO₂ aumentan, al igual que su envejecimiento. Las características de desempeño habitualmente se consideran en la evaluación de la potencia, el torque, el consumo de combustible y el factor de carga, que reflejan el grado de exigencia al que está sometido el motor de combustión interna.

El factor de carga, definido como la relación entre la potencia utilizada por el equipo y la potencia nominal disponible por el motor (Tan et al., 2021), se utiliza para estimar el consumo de combustible de las maquinarias de minería. Cada tipo de maquinaria interviene en la operación con un trabajo específico, por lo que los factores de carga son específicos para cada tipo de equipo y sus condiciones de aplicación u operación ((Klanfar et al., 2016). Sin embargo, para la determinación del factor de carga es necesario conocer la potencia suministrada por el motor de combustión durante la propulsión del vehículo, dato que se puede estimar en función del flujo de combustible consumido durante una operación específica, aunque esto requiere el uso de caudalímetros invasivos en el equipo.

Existe, por lo tanto, una relación entre los parámetros energéticos de salida del motor, como la potencia al freno y el torque suministrado, y los parámetros energéticos de entrada, como el flujo másico de combustible consumido, que ayudan al cálculo del factor

de carga. Estos parámetros, obtenidos en pruebas dinamométricas, exigen el uso de dicho equipo. Sin embargo, ha sido posible estudiar el comportamiento de los motores utilizando diversos modelos matemáticos que relacionan los parámetros de salida con otros parámetros de entrada y funcionales del motor de combustión.

El estudio del comportamiento de los motores de combustión interna para mejorar su rendimiento, reducir sus emisiones y disminuir su consumo energético depende en gran medida de las características de combustión, tanto cualitativas como cuantitativas. Por esta razón, se han desarrollado numerosos modelos de combustión de dichos motores que permiten determinar las condiciones de comportamiento (potencia y torque) y proyectar la composición de los gases emitidos según el tipo de motor y el combustible utilizado. En este sentido, Aliramezani et al., (2022) indican que los modelos de motores de combustión interna están diseñados para diferentes propósitos, como diagnóstico, análisis de datos, optimización y control para el desarrollo de motores de alto rendimiento y bajas emisiones. Según ello los modelos de motores de combustión interna se pueden dividir en tres grupos principales: modelos físicos, modelos de caja negra, y modelos de caja gris.

Los modelos basados en la física capturan las características de funcionamiento del motor mediante la aplicación de leyes físicas, incluidas las leyes de la termodinámica, la conservación de la masa y las relaciones termocinéticas, para simular los fenómenos que tienen lugar durante el ciclo de operación del motor. Una ventaja principal de un modelo basado en la física es que si se incluye un fenómeno físico en el modelo, puede adaptarse a los cambios en esas características físicas del sistema debido a su comprensión subyacente basada en la física de los procesos de los motores e combustión interna, sin embargo, muchos de los fenómenos de combustión de motores de combustión interna, como la formación de emisiones dentro de los cilindros, la autoignición y la variabilidad cíclica, son extremadamente complejos y no se han capturado con precisión con modelos basados en la física de última generación. Estos modelos suelen estar limitados a la velocidad del motor y las condiciones de carga para las que han sido calibrados. Además, muchos de los modelos basados en la física no son adecuados para el control en tiempo real porque se requieren modelos computacionalmente eficientes para diseñar sistemas de control (Aliramezani et al., 2022).

Estos modelos también toman en cuenta los parámetros geométricos del diseño del motor, ya que son relevantes para el modelado de su comportamiento en regímenes estacionarios (estables). Sin embargo, en los regímenes transitorios, con el cambio de velocidad o de carga, se producen picos en los registros de torque y potencia desarrollada, y especialmente el incremento de las emisiones contaminantes, cuyas estimaciones y

proyecciones son cada vez más difíciles de modelar. Este comportamiento se agrava en condiciones de operación específicas, de mantenimiento y ambientales en las que operan los motores de combustión analizados, que no fueron consideradas durante el modelado, lo que dificulta la evaluación de sus parámetros de desempeño.

En este contexto, los modelos de predicción del desempeño y de emisiones contaminantes presentan desventajas frente a las exigencias actuales. Sin embargo, las modernas técnicas computacionales basadas en inteligencia artificial han tenido éxito en la predicción del comportamiento de sistemas no lineales (Contreras et al., 2019). Tal es el caso de las técnicas de Redes Neuronales Artificiales (RNA), que han demostrado ser una alternativa eficiente para predecir el comportamiento de los motores de combustión interna, aunque su aplicación en la minería aún es incipiente. Las redes neuronales artificiales (RNA) ya han sido desarrolladas y aplicadas en la industria, incluyendo el modelado de motores de combustión. Sin embargo, el alto costo computacional para su desarrollo y aplicación demanda la exploración de otras técnicas de inteligencia artificial (Zhang et al., 2022; Najafi et al., 2019; Liu et al., 2022; Yang et al., 2022; Wong et al., 2013). Por otro lado, cuando se trata del control y diagnóstico de motores, se puede emplear otra técnica de inteligencia artificial: la lógica difusa, que, aunque robusta, y combinada con algoritmos de redes neuronales, presenta una eficiencia mayor pero limitada debido a la naturaleza compleja de los procesos y la presencia de big data, como ocurre en el control de motores de vehículos híbridos (Ineza et al., 2023; Narendranathan et al., 2022).

Ante estas circunstancias, una técnica de inteligencia artificial que cobra relevancia es el "Machine Learning" o aprendizaje automático, que incluye diversos algoritmos de entrenamiento para aprender el comportamiento del fenómeno analizado y, posteriormente, predecir resultados completamente nuevos. Así, el machine learning puede ser útil para el desarrollo de nuevos modelos de estimación del desempeño y predicción de las principales emisiones de los motores de combustión (Khurana et al., 2020; Zeng et al., 2020) abriendo la puerta a soluciones novedosas para la problemática del pronóstico del desempeño de los motores de combustión interna.

Con el objetivo de determinar si un modelo de machine learning puede predecir el desempeño de un motor de combustión interna, evidenciado a través de la potencia al freno, el torque desarrollado, el consumo instantáneo de combustible, y el factor de carga, basándose en sus parámetros operacionales como el régimen de giro, la presión del riel de combustible, la presión del múltiple de admisión, la temperatura del aire de admisión, la temperatura del combustible, entre otros, surge la siguiente pregunta de investigación:

¿En qué grado un modelo de pronóstico basado en machine learning logra la predicción del desempeño instantáneo de motores de combustión interna de aplicación industrial?

De este análisis se desprenden los siguientes problemas específicos:

- ¿Qué modelos de aprendizaje supervisado son adecuados en la de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales?
- ¿Cuáles son los principales parámetros del motor que son relevantes en el modelo de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales?
- ¿De qué manera se logra la predicción del factor de carga de motores de combustión interna en aplicaciones industriales?

1.3. Objetivos del estudio

Los objetivos de la siguiente investigación se dividen en generales y específicos:

1.3.1. Objetivo general

Desarrollar un modelo de pronóstico basado en machine learning que pueda predecir el desempeño instantáneo de motores de combustión interna en aplicaciones industriales.

1.3.2. Objetivos específicos

- Desarrollar y evaluar diversos modelos de aprendizaje supervisado aplicables a la de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales
- Evidenciar los principales parámetros del motor que sean relevantes en el modelo de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales
- Establecer la predicción del factor de carga de motores de combustión interna en aplicaciones industriales

1.4. Antecedentes investigativos

Shateri et al. (2024) publicaron el artículo titulado “Utilizing Artificial Intelligence to Identify an Optimal Machine Learning Model for Predicting Fuel Consumption in Diesel Engines”. Este artículo describe la utilización de técnicas de inteligencia artificial (IA) para identificar un modelo óptimo de “Machine Learning” (ML) para predecir el consumo de combustible de dodecano ($C_{12}H_{16}$) en la combustión de motores diésel. El estudio tiene como objetivo llenar el vacío de investigación en la evaluación de diferentes modelos ML para predecir el consumo de combustible en motores diésel. Respecto a su metrología, el estudio incorpora un análisis de sensibilidad para evaluar los niveles de impacto de varios parámetros sobre el consumo de combustible, destacando así los factores más influyentes. Además, este estudio aborda el impacto del ruido e implementa técnicas de limpieza de datos para asegurar la confiabilidad de los resultados obtenidos. Para validar la precisión de las predicciones, el estudio emplea varias métricas y procesos de validación, incluidas comparaciones con resultados de dinámica de fluidos computacional (CFD) y datos experimentales. Como unidad de análisis el estudio, se emplea un motor diésel de encendido por compresión con una carrera del motor de 158,54 mm, mientras que la longitud de la biela es de 270,0 mm, relación de compresión de 18:1. Operando a un régimen de 1100 rpm. El conjunto de datos utilizado para el análisis se obtuvo en simulaciones CFD, que pueden introducir varias fuentes de ruido. Como datos de entrada se han seleccionado varios parámetros como la relación de combustible (FR), masa del combustible (FM), fracción de masa de combustible promediada (FMFA), tasa liberación de calor aparente (AHRR), Relación de equivalencia aire/combustible (AFER), Penetración de pulverización (LSP-VSP), Velocidad de inyección (IV) y Volumen del cilindro (CV). Temperatura del cilindro (CT), Remolino del cilindro (CS), Presión del cilindro (CP), relación de equivalencia aire/combustible (AFER), masa de aire Fracción promedio (AMFA), relación de aire (AR) y relación de equivalencia de aire (ARE). Estas entradas capturan varios aspectos del proceso de combustión, tales como características de pulverización, geometría de la cámara, turbulencia, temperatura, presión y propiedades de la mezcla de aire y combustible. Los resultados indican que regresión de procesos gaussianos (GPR) supera a los demás en términos de precisión, como lo demuestran las métricas: MAE, MSE, PC y R2. Por ejemplo, en el caso de AHRR, GPR alcanza un MAE de 0,05124, MSE de 0,06514, coeficiente de Pearson (PC) de 0,99992 y R2 de 0,99982, mientras que RFR y redes neuronales (NN) exhiben valores ligeramente más altos para estas métricas. Tendencias similares son observado para FMFA, FM y FR. Además, se demuestra que el rango de error para GPR está entre 0 y 0,02, mientras que el rango de error para regresión de bosque aleatorio (RFR) se encuentra entre 0 y 0,08, y para redes neuronales (NN) es

mucho mayor y supera 0,1. Estos hallazgos enfatizan la precisión superior del modelo regresión de procesos gaussianos (GPR) en comparación con RFR y NN. Como Conclusión, los hallazgos de este estudio demuestran que el modelo GPR supera a otros modelos en términos de precisión. Métricas como el error absoluto medio (MAE), error cuadrático medio (MSE), coeficiente de Pearson (PC) y R cuadrado (R^2) indican consistentemente el rendimiento superior del modelo GPR.

Kocak et al. (2023) en su trabajo titulado “Condition Monitoring and Fault Diagnosis of a Marine Diesel Engine with Machine Learning Techniques”, El estudio tuvo el objetivo desarrollar algoritmos basados en aprendizaje automático (ML) para la detección de fallos en motores diésel marinos. Esto permitiría implementar un mantenimiento predictivo, reduciendo costos y fallos catastróficos al anticipar problemas antes de que sean detectables manualmente. Respecto a la metodología, los autores abordan el problema de la detección de fallos en un motor diésel utilizado como motor principal de un buque cisterna, los métodos basados en Machine learning requieren suficientes datos que representen todos los posibles casos de falla y no falla de manera equilibrada. Para obtener dichos datos se utilizó un simulador realista de sala de máquinas, el simulador utilizado es el simulador de sala de máquinas del Kongsberg Maritime (Noruega) y el modelo es el petrolero K-Sim equipado con motor principal MAN B&W 5L90MC VLCC, el que fue simulado en tres condiciones de fallo específicas. Los parámetros fueron ajustados según escenarios controlados para registrar comportamientos antes, durante y después de los fallos. Respecto a los parámetros de entrada, Los datos fueron recolectados en un simulador realista del motor principal de un buque tipo VLCC. Se utilizaron lecturas de sensores relacionadas con temperaturas de agua, aceite y gases de escape, Presión de sistemas de agua y combustible, Consumo de combustible y potencia del eje, entre otros. Se recopilaron más de 270,000 muestras para tres tipos de fallos: obstrucción de válvula de inyección, fugas en la válvula de escape, grieta en el cilindro. Siendo los parámetros de salida: la Identificación de la condición de falla en forma de porcentaje (0%-100%). La clasificación entre estado normal y fallido para cada fallo simulado. Como Resultados, Los métodos basados en árboles de decisión, particularmente "Fine Tree", mostraron la mayor precisión ($R^2 > 0.999$) tanto en la fase de entrenamiento como en la de prueba. Otros algoritmos, como SVM ($R^2 > 0.98$) y regresión lineal ($R^2 > 0.98$), también obtuvieron resultados altos pero con menor eficiencia en tiempo de ejecución. La reducción de dimensionalidad con PCA identificó las variables más significativas para cada fallo, optimizando la modelación. Como conclusiones: Los algoritmos de ML demostraron alta eficacia para la detección temprana de fallos en motores diésel marinos, especialmente al usar modelos de árboles. Esto refuerza el potencial de aplicar ML en sistemas de

mantenimiento predictivo a bordo. Además, los simuladores permiten generar grandes conjuntos de datos para entrenar modelos, representando una alternativa escalable para capturar conocimientos expertos y reducir el tiempo de análisis humano.

Ricci et al. (2023), en su artículo científico titulado “NARX Technique to Predict Torque in Internal Combustion Engines”, El objetivo del estudio fue implementar la técnica NARX (redes neuronales autorregresivas no lineales con entradas exógenas) para predecir el torque en motores de combustión interna (ICE). Se busca optimizar la predicción, reducir la complejidad de los datos de entrada y lograr una precisión en torno al 1%, lo que contribuiría a una mejor calibración y control en tiempo real de los motores. Respecto a la metodología, esta investigación se centra en un motor SMART W451T con un diseño de tres cilindros y tecnología de inyección de puerto como unidad de análisis. Se emplearon configuraciones de hardware como sensores piezoeléctricos, un banco dinamométrico de corrientes de Eddy y sistemas de adquisición de datos para registrar las señales experimentales. Respecto a los parámetros de entrada: El modelo utiliza 12 variables recopiladas de sensores físicos y la ECU (Unidad de Control del Motor) tales como: Temperaturas y presiones en el sistema de admisión y escape, tiempo de inyección y avance del encendido, posición del acelerador y velocidad del motor, los datos se registraron a una frecuencia de 10 Hz y se analizaron durante cinco ciclos transitorios con variaciones de velocidad de 1000 a 3500 RPM. De igual manera los parámetros de salida, son torque predicho en función del tiempo, errores porcentuales promedio entre la predicción y las mediciones reales. Los resultados alcanzados, indican que el rendimiento del entrenamiento de 42 diferentes combinaciones de neuronas y capas ocultas se evaluaron en términos de RMSE cada combinación muestra un valor RMSE por debajo del nivel aceptable de umbral del 5%. En particular, la estructura compuesta por 2 capas ocultas con 21 y 23 neuronas respectivamente presentó el mejor rendimiento con un valor RMSE igual a 3,37%. Se seleccionó dicha estructura, es decir, {21 23}, para la sesión de prueba, logrando predecir el 20% del resto de la señal de Torque restante. Respecto al Análisis de sensibilidad: La técnica SHAP permitió reducir las entradas de 12 a 9, eliminando variables con menor impacto en la predicción (e.g., posición del acelerador). La predicción con un conjunto de datos reducido mejoró tanto la precisión como la estabilidad de los resultados. Como conclusiones, se indicó que: la técnica NARX es efectiva para modelar sistemas dinámicos no lineales como el torque en ICE. El análisis de sensibilidad y la reducción de dimensiones optimizan el rendimiento del modelo, logrando predicciones consistentes incluso con conjuntos de datos limitados. Este enfoque presenta un potencial significativo para aplicaciones en tiempo real y reducción de costos computacionales en la industria

automotriz. Se sugiere explorar modelos adicionales y configuraciones más complejas para abordar variaciones en diferentes ciclos de motor.

Ineza et al. (2023), en su artículo científico titulado “Review of Artificial Intelligent Algorithms for Engine Management”, Tuvieron por objetivo analizar los avances y aplicaciones de algoritmos de inteligencia artificial (IA) en la gestión de motores de combustión interna. Su objetivo principal es revisar técnicas aplicadas al rendimiento, control y diagnóstico de motores en los últimos 15 años, identificando mejoras y direcciones futuras para optimizar eficiencia, emisiones y diagnósticos en estos sistemas. Respecto a la Metodología, el artículo utiliza la metodología PRISMA para seleccionar referencias relevantes, asegurando un análisis sistemático y riguroso de los estudios relacionados. Las áreas principales estudiadas incluyen: Rendimiento del motor (economía de combustible y control de emisiones). Control del motor (gestión de operaciones transitorias y vehículos híbridos). diagnóstico del motor (detección de fallas como fallos de encendido, golpeteo y fugas en el sistema de admisión). Los algoritmos analizados emplean múltiples parámetros relacionados con el rendimiento del motor, tales como: Velocidad del motor. torque indicado, consumo de combustible, temperatura y presión del cilindro, señales de sensores como flujo de aire y emisión de gases. Además, los sistemas basados en IA generan como resultados: predicciones de emisiones (CO, NOx, CO₂), diagnósticos de fallas específicas (e.g., misfire). optimización de consumo de combustible. Respecto a la Unidad de análisis, el estudio se centra en motores de combustión interna, incluyendo su integración en vehículos híbridos y eléctricos, examinando su comportamiento bajo diferentes escenarios operativos, como regímenes estacionarios y transitorios. De la revisión ejecutada, este trabajo muestra los siguientes resultados, los algoritmos de redes neuronales (ANN) demostraron alta precisión en el control de emisiones y economía de combustible: técnicas como sistemas neuro-difusos y aprendizaje por refuerzo se usaron para predecir y reducir emisiones; Respecto al control del motor: En vehículos híbridos, estrategias como lógica difusa y aprendizaje por refuerzo mejoraron la gestión de energía, maximizando la eficiencia del tren motriz. En operaciones transitorias, métodos como control feedforward dinámico y aprendizaje profundo optimizaron el rendimiento sin aumentar emisiones. Respecto al Diagnóstico del motor: algoritmos de aprendizaje automático fueron efectivos para detectar fallas como misfires, superando desafíos de ruido en los datos. Como conclusiones, los algoritmos de IA ofrecen soluciones robustas para problemas complejos en motores, mejorando la precisión en predicciones y diagnósticos en tiempo real. La integración de enfoques como aprendizaje por refuerzo y redes neuronales profundas es clave para abordar las crecientes demandas de eficiencia y sostenibilidad en los motores modernos.

Pająk et al. (2023), en su trabajo titulado “Ship Diesel Engine Fault Diagnosis Using Data Science and Machine Learning”, El objetivo principal fue diseñar un método basado en aprendizaje automático (ML) para diagnosticar fallos en motores diésel marinos, específicamente en el motor Sulzer 6AL20/24. Se busca mejorar la identificación del estado de confiabilidad y los fallos en escenarios con datos limitados, incompletos o de baja calidad. Para la unidad de análisis: el motor estudiado es el Sulzer 6AL20/24, equipado con seis cilindros y utilizado comúnmente en barcos como generador auxiliar o propulsión principal. Las pruebas se realizaron en un banco de laboratorio simulado, registrando las señales durante operaciones normales y fallos inducidos. Respecto a la metodología, el análisis comprende la selección de parámetros de entrada, los cuales consisten en señales de vibración y ruido grabadas en cada cilindro del motor bajo diferentes estados operativos: como vibraciones verticales, horizontales y ruido. Estados evaluados: estado operativo normal (N) y cinco tipos de fallos simulados (VC02, VC08, CLE, INJ, y PUMP) con una frecuencia de muestreo: 32,768 Hz. Los Datos divididos en vectores de medida con 0.5 s de duración. Como parámetros de salida: se usaron clasificadores de los estados de confiabilidad (habilidad o incapacidad). Identificación específica del tipo de fallo. Métricas de evaluación: precisión, recuperación y puntuación. Entre los Resultados alcanzados, se determina el empleo un clasificador SVM no lineal con kernel polinómico (P-SVM) cuyo rendimiento alcanzó un 100% de precisión en la identificación de estados de habilidad/incapacidad y tipos de fallos. La reducción de características utilizando análisis de correlación redujo las dimensiones a 157 variables relevantes. Respecto a otros modelos, el método superó en precisión a variantes anteriores y modelos alternativos, como redes neuronales y SGD, que alcanzaron un máximo de 96-97%. Los autores llegan a la siguiente conclusión, el método propuesto es efectivo para diagnosticar fallos en motores diésel marinos, incluso con datos limitados o incompletos. Su enfoque basado en SVM permite la aplicación a sistemas técnicos complejos y giratorios, ofreciendo una solución escalable y generalizable. Los autores sugieren expandir las pruebas a entornos operativos reales y explorar la combinación de aprendizaje supervisado y no supervisado para mejorar aún más la robustez del sistema.

Sebayang et al. (2022), ejecutaron el trabajo titulado “Modelling and Prediction Approach for Engine Performance and Exhaust Emission Based on Artificial Intelligence of Sterculia Foetida Biodiesel”, El objetivo del estudio fue investigar el desempeño del motor y las emisiones de escape de mezclas de gasolina con bioetanol derivado de Manihot

glaziovii a diferentes proporciones (E5, E10, E15, E20). Además, se desarrolló un modelo de Extreme Learning Machine (ELM) para predecir parámetros de rendimiento del motor y emisiones, optimizando recursos y reduciendo experimentación. Respecto a la metodología, el estudio es de diseño experimental: Como unidad de análisis se empleó un motor monocilíndrico Subaru EX17 probado con un dinamómetro de corrientes de Eddy. Los parámetros a monitorear fueron el torque, potencia de frenado, consumo específico de combustible (BSFC), y la concentración de emisiones de CO, HC y NOx; para el ensayo se utilizaron mezclas homogéneas de bioetanol-gasolina de 5%, 10%, 15% y 20%. Para el desarrollo del modelo ELM, se consideraron Entrenamiento con datos de experimentos físicos, como parámetros de entradas: velocidad del motor y proporción de mezcla, y como salidas: parámetros de desempeño (torque, potencia, BSFC) y emisiones (CO, HC, NOx). Como resultados el estudio indicó, El torque y la potencia de frenado incrementaron al aumentar la proporción de bioetanol (máximos en E20). El BSFC disminuyó, alcanzando 270.7 g/kWh a 3200 rpm para E20. Y respecto a las emisiones la concentración de CO y HC, disminuyeron significativamente con mayor bioetanol debido al oxígeno adicional, mejorando la combustión. Sin embargo, la concentración de NOx se incrementó con bioetanol más concentrado por mayor temperatura de combustión. Adicionalmente el modelo ELM propuesto muestra un coeficiente de determinación (R^2): 0.980–1.000 para todas las variables; un error porcentual absoluto medio (MAPE): 0.411%–2.782%, mostrando alta precisión. Como conclusiones, el estudio establece que, la mezcla E20 optimiza el rendimiento y minimiza emisiones de CO y HC, aunque aumenta ligeramente los NOx. El modelo ELM es eficiente, preciso y práctico para predecir desempeño y emisiones, ahorrando costos y tiempo en experimentación. Se valida el bioetanol como un combustible alternativo sostenible y eficaz para motores de combustión interna, compatible con regulaciones ambientales estrictas.

Şahin (2023), ejecutaron un estudio titulado “Comparison of Machine Learning Algorithms for Predicting Diesel/Biodiesel/Iso-Pentanol Blend Engine Performance and Emissions”, El objetivo del estudio fue buscar comparar la eficacia de tres algoritmos de aprendizaje automático (ANN, SVM y XGBoost) para predecir el rendimiento y las emisiones de motores diésel utilizando mezclas de combustible de biodiésel, diésel e isopentanol, con el propósito de optimizar estas mezclas para equilibrar el rendimiento del motor y reducir las emisiones nocivas. Para este estudio la unidad de análisis fue un motor diésel de 4 cilindros evaluados con mezclas de combustibles que incluyen isopentanol, biodiésel de cártamo y diésel en diferentes proporciones. La metodología desarrollada, parte del empleo de aceite de cártamo mediante transesterificación; preparación de mezclas de combustible: Se mantuvo el biodiésel al 10% y se variaron las

proporciones de iso-pentanol (5%, 10%, 20%) con diésel. Para luego ejecutar las pruebas de motor: donde se midieron parámetros como consumo específico de combustible (BSFC), eficiencia térmica del freno (BTE) y emisiones (NOx, HC, CO₂). Posteriormente se ejecutó el modelado con Machine Learning: Los datos se analizaron usando ANN, SVM y XGBoost, ajustados con un enfoque de búsqueda en cuadrícula. Para la evaluación de modelos: Se utilizaron métricas como R², RMSE y MAE para evaluar la precisión de las predicciones. En este estudio los parámetros de entrada fueron; velocidad del motor (1100-2400 rpm). Tipo de combustible (diésel, biodiésel, iso-pentanol) Los parámetros de salida fueron Eficiencia térmica del freno (BTE), consumo específico de combustible (BSFC). emisiones de gases (NOx, HC, CO₂), temperatura de los gases de escape (EGT). Como resultados del estudio, se establecieron que el modelo ANN sobresalió en predicción de BTE (R² = 0.984, RMSE = 0.411%) y BSFC (R² = 0.958, RMSE = 6.90 g/kWh). XGBoost fue intermedio, mientras que SVM tuvo el peor desempeño. Respecto a las predicciones de emisiones: NOx: ANN fue el mejor (R² = 0.959), CO₂ y HC: XGBoost mostró la mejor precisión con R² de 0.956 y 0.973, respectivamente Sin embrago EGT: SVM superó a los otros modelos con R² = 0.981. Respecto al combustible empleado, el mayor contenido de iso-pentanol incrementó el BSFC y las emisiones de HC pero redujo las emisiones de NOx y CO₂, y la temperatura de escape. Como conclusiones, el modelo basado en ANN es ideal para predecir parámetros de rendimiento del motor, mientras que XGBoost es más eficaz para estimar emisiones.

Yang et al. (2022), desarrollaron una investigación titulada “The Application of Machine Learning Methods to Predict the Power Output of Internal Combustion Engines”, El objetivo principal del estudio fue comparar el rendimiento predictivo de tres modelos de aprendizaje automático (ANN, SVR y RF) en la predicción de la presión media efectiva indicada (IMEP), un indicador clave de la potencia de los motores de combustión interna, utilizando parámetros como el tiempo de chispa (ST), velocidad del motor y carga como entradas. Como unidad de análisis, el estudio se centra en un motor de encendido por chispa (SI) de un solo cilindro, con inyección de combustible en el puerto (PFI) y un modelo validado de dinámica de fluidos computacional (CFD) unidimensional. Como Metodología, el estudio comprende el modelado y simulación, puesto que se empleó un modelo CFD validado para generar 756 conjuntos de datos divididos en entrenamiento, validación y pruebas. Posteriormente se desarrolló tres modelos de Machine Learning: red neuronal artificial (ANN), regresión por soporte vectorial (SVR) y bosque aleatorio (RF). Para el entrenamiento y evaluación, los datos se dividieron en 80% para entrenamiento y validación, y 20% para pruebas. Se evaluaron los modelos con métricas como el coeficiente de determinación (R²) y el error cuadrático medio (RMSE). En el modelamiento se

consideraron como parámetros de entrada: tiempo de chispa (spark timing, ST), velocidad del motor (1000-6000 rpm), carga del motor, representada por la presión de admisión (0.5-1 bar). Mientas que los parámetros de salida: presión media efectiva indicada (IMEP), un indicador del rendimiento del motor. El estudio logró los siguientes resultados: la evaluación respecto al rendimiento predictivo: SVR tuvo el mejor desempeño predictivo con R^2 cercano a 1 y RMSE bajo. ANN mostró resultados comparables, aunque con fluctuaciones menores debido a la sensibilidad a la inicialización de parámetros. RF tuvo el peor desempeño, especialmente en condiciones de baja carga, mostrando errores más altos y dificultad para aprender patrones intrínsecos del motor. Respecto a la robustez, los modelos SVR y ANN demostraron robustez al no sobre ajustar los datos de entrenamiento. El modelo RF mostró aleatoriedad en sus resultados y menor capacidad de generalización. Respecto al comportamiento en condiciones específicas: Tanto SVR como ANN capturaron con precisión las relaciones no lineales entre los parámetros de entrada y el IMEP. RF falló en describir correctamente los patrones inherentes al rendimiento del motor. Como conclusiones, SVR y ANN son preferibles para la predicción del IMEP debido a su alta precisión y capacidad para modelar relaciones no lineales complejas. El modelo RF, aunque aceptable en ciertos escenarios, no es adecuado para aplicaciones que requieran precisión en condiciones variables.

Datta y Mandal (2018), publicaron el artículo titulado “Numerical Prediction of the Performance, Combustion and Emission Characteristics of a CI Engine Using Different Biodiesels” El objetivo de dicho artículo fue investigar numéricamente el rendimiento, las características de combustión y las emisiones de un motor diésel de inyección directa utilizando diferentes biodiésels como combustibles alternativos, validando los resultados numéricos con datos experimentales. Respecto a su metodología, los investigadores utilizaron el software Diesel-RK para modelar el comportamiento del motor en términos de rendimiento, combustión y emisiones, empleando seis combustibles: diésel, biodiésel de jatrofa, soya, rapeseed, palma estearina y karanja. Luego se ejecutaron pruebas experimentales con diésel y biodiésel de palma estearina realizadas en un motor monocilíndrico con las mismas condiciones operativas para validar los resultados simulados. Respecto a la unidad de análisis, este está conformado por un motor diésel de inyección directa monocilíndrico, refrigerado por agua, alimentado con diferentes biodiésels como combustibles alternativos, con relación de compresión 17.5: El modelamiento emplea parámetros de entrada: como propiedades termoquímicas de cada combustible (calor de combustión, viscosidad, densidad, número de cetano). Y como salida, parámetros de rendimiento: eficiencia térmica, consumo específico de combustible, características de combustión: presión pico en el cilindro, tasa de liberación de calor. Y el

nivel de emisiones: tales como óxidos de nitrógeno (NO_x), partículas (PM), dióxido de carbono (CO₂). Los resultados mostraron que el rendimiento del motor se deteriora al usar diferentes tipos de biodiésel como combustible. La eficiencia térmica al freno del motor disminuyó hasta un 3% con el uso de biodiésel de estearina de palma, mientras que el consumo específico de combustible aumentó, alcanzando valores máximos del 25,8%. Entre los diferentes tipos de biodiésel, el de jatrofa mostró el mejor rendimiento, mientras que el de estearina de palma, el peor. Se observó que tanto el período de retardo de la ignición como la tasa de liberación de calor disminuyeron ligeramente con los diferentes biodiésels en comparación con el diésel. En conclusión, el uso de biodiésel proporciona un escape más limpio en comparación con el diésel, siendo el biodiésel de jatrofa el más eficiente en términos de partículas y emisiones de humo, aunque el uso de biodiésel también incrementa la formación de óxidos de nitrógeno, siendo el biodiésel de colza el que presenta el mayor aumento.

Liu et al. (2018), en su artículo científico titulado “An artificial neural network developed for predicting the performance and emissions of a spark ignition engine fueled with butanol–gasoline blends”, presentaron un estudio cuyo objetivo es desarrollar un modelo de red neuronal artificial (ANN) para predecir con éxito el rendimiento del motor y las emisiones de escape cuando un motor de encendido por chispa es alimentado con mezclas de n-butanol-gasolina (0-60 vol.% n-butanol mezclado con gasolina, referido como G100-B60) bajo varias relaciones de equivalencia. La investigación, en su metodología, tiene un diseño experimental, explicativo, de enfoque cualitativo. En el modelo de red neuronal artificial, se utilizaron como parámetros de entrada la relación de compresión, la relación de equivalencia, el porcentaje de mezcla y la carga del motor; mientras que el rendimiento del motor y las emisiones, como la eficiencia térmica, el consumo de combustible específico, la potencia al freno, el monóxido de carbono, los hidrocarburos no quemados y el NO_x, se emplearon como parámetros de salida. En los resultados de dicha investigación, al comparar los datos experimentales con las predicciones, se obtuvo un coeficiente de correlación (R^2) que varió de 0,9929 a 0,9996 y un error relativo medio que varió de 0,1943 a 9,9528%. Se concluye que el modelo de red neuronal artificial desarrollado fue capaz de predecir la combustión de mezclas de n-butanol-gasolina con una precisión encomiable. En comparación con diferentes mezclas de butanol-gasolina, B30 proporcionó mejores resultados en base a su BTE similar y menores emisiones de CO (8,7%), HC (15,3%) y NO_x (6,4%) que las del G100. Con base en el R^2 cercano a 1 y el bajo valor de la raíz del error cuadrático medio (RMSE) y el error relativo medio (MRE), se indicó que ANN resultó ser una herramienta útil para la correlación y simulación de parámetros del motor. ANN proporcionó un enfoque preciso y simple en el análisis del

rendimiento del motor y las emisiones de escape del motor de encendido por chispa. Como conclusión, se obtiene que el rendimiento del motor y las emisiones de escape del motor alimentado con mezclas de butanol y gasolina se predijeron utilizando el método ANN.

Jassim et al. (2017), en su artículo científico titulado “Predicting Energy Consumption and CO₂ Emissions of Excavators in Earthwork Operations: An Artificial Neural Network Model”, presentaron un estudio cuyo objetivo es proponer un modelo que utiliza una red neuronal artificial (ANN) para predecir el consumo de energía por hora de una excavadora y las emisiones de CO₂ en diferentes condiciones del sitio de operación. En su metodología, la investigación se ejecutó con un enfoque cuantitativo, de alcance explicativo, y empleó una técnica analítica. El modelo ANN propuesto incluye cinco parámetros de entrada: profundidad de excavación, tiempo del ciclo, carga útil del cucharón, potencia del motor y factor de carga. Los datos del manual de Caterpillar, que incluían características operativas de veinticinco modelos de excavadoras, se utilizaron para desarrollar los conjuntos de entrenamiento y prueba para el modelo ANN. Los modelos ANN propuestos también se diseñaron para identificar qué factores de todos los parámetros de entrada tienen el mayor impacto en la energía y las emisiones, según el análisis del método de pesos divididos. Los resultados mostraron que los modelos ANN propuestos proporcionan una herramienta de estimación precisa para la etapa de planificación inicial, permitiendo predecir el consumo de energía y las emisiones de CO₂ de las excavadoras. Los análisis revelaron que, entre todos los parámetros de entrada, el tiempo de ciclo tiene el mayor impacto en el consumo de energía y las emisiones de CO₂. Respecto a las conclusiones, el estudio indica que los hallazgos de la investigación permiten el control de factores cruciales que impactan significativamente en el consumo de energía y las emisiones de CO₂.

1.5. Justificación e importancia de la investigación

El vertiginoso avance de la tecnología abre una enorme posibilidad de nuevos enfoques para su aprovechamiento. Por lo tanto, emplear tecnologías novedosas basadas en inteligencia artificial para abordar problemas tecnológicos se ha convertido en una herramienta prometedora en este campo de aplicación. En este sentido, este trabajo de investigación impulsa el desarrollo de técnicas de “*machine learning*” para resolver problemas del ámbito industrial moderno, lo cual también ha sido cubierto en la literatura revisada con abundantes algoritmos de redes neuronales y algunos pocos de regresión lineal. Sin embargo, no se han considerado en aquellas investigaciones algoritmos novedosos como XGBoost, ni algoritmos específicos como SVM, KNN y Random Forest, los cuales son evaluados en este trabajo. Asimismo, la revisión de la literatura muestra el

empleo de potentes herramientas informáticas en el desarrollo de algoritmos de inteligencia artificial, como los “toolboxes” de Matlab. No obstante, este trabajo se desarrolla con software libre Python y las librerías de Scikit-Learn.

Adicionalmente, este trabajo aborda una perspectiva diferente en la elección de los parámetros de lectura en pruebas dinamométricas. Muchos de los análisis previos consideran al torque suministrado por el motor como un parámetro de entrada al modelo de predicción. Sin embargo, en este trabajo se evidencia la relación del torque con otros parámetros de operación de fácil lectura, como la presión del riel de combustible y la presión del múltiple de admisión, los cuales pueden ser tomados directamente en la operación del motor en el equipo que propulsan, convirtiendo al torque en un parámetro de salida para el modelo propuesto. En esta línea, se justifican los aportes teóricos, prácticos, ambientales y académicos del presente estudio.

El presente estudio posee justificación teórica, ya que muestra los conceptos fundamentales para el desarrollo de modelos de machine learning, evidencia la influencia de estos modelos en la predicción del comportamiento del motor de combustión interna, define, caracteriza y calcula el factor de carga, y valida los procesos mediante la evaluación de errores estadísticos. Finalmente, este trabajo valida sus resultados con estudios previos sobre el comportamiento del factor de carga y mediante procesos estadísticos de validación de hipótesis, aplicando el método científico.

Este trabajo también posee justificación práctica, ya que su metodología y sus resultados pueden ser rápidamente aplicados para determinar el factor de carga instantáneo en la operación de cualquier equipo impulsado por un motor de combustión interna previamente evaluado con las técnicas de *machine learning*. Las predicciones pueden emplearse en actividades de operación, mantenimiento predictivo, o evaluación del valor del activo.

Este trabajo también posee justificación ambiental, puesto la determinación del consumo instantáneo de combustible, así como del como el factor de carga, permite abrir cálculos de determinación de la producción de CO₂ durante la operación del motor de combustión interna, con ello es posible pronosticar el aporte de CO₂ en un proceso productivo equipado con motores de combustión interna, permitiendo mejorar la huella de carbono de dicha industria.

Desde una perspectiva académica, este trabajo valida modelos y técnicas de inteligencia artificial en procesos de aplicación industrial, abriendo la posibilidad de evaluar futuros modelos de *machine learning* aún por desarrollarse, y estudiar su impacto en aplicaciones industriales.

Este trabajo tiene un alto grado de importancia, ya que presenta un modelo predictivo del comportamiento del motor de combustión interna, con el fin de calcular el factor de carga instantáneo al que está sometido en la operación regular de la máquina que propulsa. Así, este trabajo impacta en la industria al contribuir a la reducción de costos operativos y de mantenimiento de los motores de combustión interna, al detectar disfunciones operacionales y deficiencias en el comportamiento del equipo mediante la predicción de su rendimiento.

1.6. Hipótesis

1.6.1. Hipótesis general

Un modelo de pronóstico basado en machine learning logrará la predicción del desempeño instantáneo de un motor de combustión interna de aplicación industrial

1.6.2. Hipótesis específicas

- Modelos no lineales de aprendizaje supervisado logran la predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales.
- Los parámetros asociados la masa de aire y a la masa de combustible son relevantes en el modelo de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales
- Un modelo no lineal que logra determinar la potencia pronosticada logra la predicción del factor de carga de motores de combustión interna en aplicaciones industriales

1.7. Variables e indicadores

1.7.1. Variable 1

Variable 1: Modelo de pronóstico basado en machine learning.

Sus dimensiones son:

- Torque pronosticado
- Potencia pronosticada
- Consumo de combustible pronosticado
- Factor de carga pronosticado

1.7.2. Variable 2

Variable 2: Desempeño instantáneo.

Sus dimensiones son:

- Torque instantáneo
- Potencia instantánea
- Consumo instantáneo de combustible
- Factor de carga instantáneo

1.8. Tipo y nivel de la investigación

El presente trabajo de investigación posee un enfoque cuantitativo. De acuerdo con lo indicado por Hernández y Mendoza (2018) el enfoque cuantitativo debe tener, en primer lugar, un proceso secuencial que abarca desde el planteamiento del problema, la determinación del objetivo, el establecimiento de las hipótesis y variables, hasta el bosquejo de un plan para probar dicha hipótesis en un contexto determinado, con la capacidad de extraer conclusiones. En segundo lugar, se destaca la necesidad de medir y estimar magnitudes relacionadas con el problema de investigación, utilizando la recolección de datos, el análisis estadístico y la cuantificación. Finalmente, este enfoque se desarrolla en la "realidad externa" del individuo, es decir, en una realidad objetiva donde el punto de vista o pensamiento de los individuos no interfieren en los resultados.

El presente trabajo de investigación posee un nivel correlacional. Según Arias (2012), los estudios correlacionales buscan identificar si dos o más variables están relacionadas, sin establecer una relación de causa y efecto directa. Mediante análisis estadísticos, se mide la fuerza y dirección de esta relación, lo que puede sugerir posibles vínculos causales.

El diseño desarrollado en el presente trabajo ha sido no experimental de corte transversal correlacional. En cuanto al análisis no experimental, Hernández y Mendoza (2018) señalan que este diseño se lleva a cabo sin manipular intencionalmente las variables, sin provocar una nueva cualidad, sino observando las cualidades existentes, las cuales no han sido ocasionadas de manera premeditada por el investigador. En otras palabras, en la investigación no experimental, las variables se comportan espontáneamente sin que el investigador ejerza control o manipulación.

El trabajo desarrollado incluye una revisión bibliográfica de los aportes investigativos sobre la aplicación de modelos de *machine learning* en el ámbito del desempeño de motores de combustión interna.

1.9. Muestra

La muestra tomada para este trabajo ha sido el resultado de un muestreo no probabilístico por conveniencia, que incluye un motor de combustión interna de encendido por compresión, comúnmente denominado motor diésel, con un sistema de inyección del tipo “common rail” de la marca Komatsu. Las características detalladas del motor se pueden encontrar en la sección Anexo IV.

Capítulo II. Marcos teórico y conceptual

2.1. Marco teórico

2.1.1. *La inteligencia artificial*

La inteligencia artificial (IA) se refiere al intento de automatizar tareas intelectuales que habitualmente realizan los seres humanos. Es un campo amplio que incluye tanto el aprendizaje automático como el aprendizaje profundo (ver Figura 1), pero también abarca otros enfoques que no requieren aprendizaje. Un ejemplo de esto son los primeros programas de ajedrez, los cuales solo utilizaban reglas preestablecidas por los programadores y no se consideraban aprendizaje automático. Durante mucho tiempo, muchos especialistas pensaron que la inteligencia artificial a nivel humano podría alcanzarse mediante la creación de un conjunto lo suficientemente grande de reglas explícitas que permitieran manipular el conocimiento. Este enfoque, conocido como IA simbólica, fue el paradigma principal en la IA desde la década de 1950 hasta finales de la década de 1980, alcanzando su mayor popularidad durante el auge de los sistemas expertos en los años 80.(Chollet, 2018).

Aunque la IA simbólica fue eficaz para resolver problemas lógicos bien definidos, como jugar al ajedrez, resultó ser inadecuada para encontrar reglas explícitas que resolvieran problemas más complejos y ambiguos, como la clasificación de imágenes, el reconocimiento de voz y la traducción de idiomas. De esta manera, surgió un nuevo enfoque, el aprendizaje automático, para reemplazar a la IA simbólica. (Chollet, 2018).

La IA siempre ha tenido como referencia las capacidades de la inteligencia humana, enfocándose en distintos aspectos. Así, la primera motivación fue intentar construir máquinas capaces de razonar como el ser humano, o por lo menos poder emularle en alguna capacidad de tal forma que denotara cierto grado de inteligencia (Ponce, 2010).

Russel y Norvig (2012) en su libro “Artificial Intelligence: A Modern Approach” recalcan la definición de inteligencia artificial en cuatro aspectos en el pensar humanamente, pensar racionalmente, actuar humanamente y el actuar racionalmente. Asimismo, el libro resalta la capacidad de los ordenadores para utilizar el razonamiento y las acciones lógicas para alcanzar objetivos. Estos dispositivos combinan grandes volúmenes de datos con procedimientos rápidos y repetitivos, junto con algoritmos diseñados para aprender automáticamente a partir de patrones o características. La

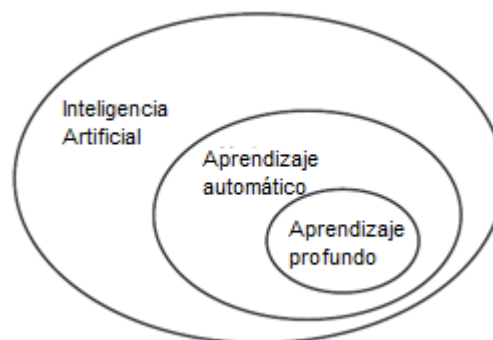
inteligencia artificial (IA) genera modelos predictivos mediante la utilización de algoritmos y datos históricos, lo que permite realizar predicciones sobre resultados futuros.

2.1.2. El machine learning o aprendizaje automático

El aprendizaje automático, o machine learning, es un área de la informática que facilita la programación de máquinas para que puedan aprender a partir de un conjunto de datos previamente dados. (Géron, 2019). El aprendizaje automático es un subconjunto de la inteligencia artificial (Figura 1); para lo cual autores como Arthur Samuel lo definen como el área de la ciencia que permite a las computadoras aprender sin necesidad de ser programadas de manera explícita para hacerlo (Wiederhold y McCarthy, 2010). En resumen, el aprendizaje automático es capaz de generalizar un comportamiento a partir de un conjunto de experiencias.

Figura 1

Inteligencia artificial, aprendizaje automático y aprendizaje profundo.



Fuente: (Chollet, 2018).

El aprendizaje automático puede considerarse como una disciplina de la informática en la que los datos existentes se emplean para predecir o dar respuestas a datos futuros (Paluszek y Thomas, 2017). Este campo se centra en extraer conocimiento de datos previamente recolectados. Es un área de investigación que se encuentra en la intersección de la estadística, la inteligencia artificial y la informática, y también es conocida como análisis predictivo o aprendizaje estadístico (Müller y Guido, 2017).

De forma similar, el aprendizaje automático consiste en programar computadoras para que mejoren su desempeño en función de un criterio específico, utilizando datos de

ejemplo o experiencias anteriores. Este proceso implica aprender a optimizar o ajustar los parámetros de un modelo, el cual puede ser conocido previamente o desarrollado desde cero, basándose en datos de entrenamiento o experiencias pasadas. Dicho modelo puede ser de carácter predictivo, para realizar estimaciones futuras; descriptivo, para extraer información relevante de los datos; o incluso cumplir ambas funciones simultáneamente (Alpaydin, 2014).

El machine learning se aplica tanto en el desarrollo de tecnologías emergentes como en actividades diarias. En el ámbito de las tecnologías emergentes, está estrechamente vinculado con áreas como el reconocimiento de patrones, las estadísticas computacionales y la inteligencia artificial. El aprendizaje automático es fundamental en campos como el reconocimiento facial, el filtrado de spam y otros casos donde resulta inviable, o incluso imposible, crear algoritmos específicos para realizar una tarea (Paluszek y Thomas, 2017). En el día a día, se utiliza en recomendaciones automáticas sobre películas, comida o productos a comprar, así como en servicios como radios en línea personalizadas o el reconocimiento de amigos en fotos. Muchos sitios web y dispositivos actuales incorporan algoritmos de machine learning en su funcionamiento central. Al visitar plataformas complejas como Facebook, Amazon o Netflix, es muy probable que cada sección de estas contenga diversos modelos de aprendizaje automático.

El machine learning se aplica tanto en el desarrollo de tecnologías emergentes como en actividades diarias. En el ámbito de las tecnologías emergentes, está estrechamente vinculado con áreas como el reconocimiento de patrones, las estadísticas computacionales y la inteligencia artificial. El aprendizaje automático es fundamental en campos como el reconocimiento facial, el filtrado de spam y otros casos donde resulta inviable, o incluso imposible, crear algoritmos específicos para realizar una tarea (Paluszek y Thomas, 2017). En el día a día, se utiliza en recomendaciones automáticas sobre películas, comida o productos a comprar, así como en servicios como radios en línea personalizadas o el reconocimiento de amigos en fotos. Muchos sitios web y dispositivos actuales incorporan algoritmos de machine learning en su funcionamiento central. Al visitar plataformas complejas como Facebook, Amazon o Netflix, es muy probable que cada sección de estas contenga diversos modelos de aprendizaje automático (Müller y Guido, 2017).

Según lo mencionado, el aprendizaje automático se emplea principalmente para resolver los siguientes tipos de problema (Géron, 2019):

- Aquellos que requieren una extensa cantidad de reglas y programación. En estos casos, el aprendizaje automático puede simplificar el código y optimizar el rendimiento del programa.
- Problemas complejos para los cuales no existe una solución eficaz mediante métodos tradicionales.
- Entornos cambiantes o dinámicos.
- Problemas que tienen una gran complejidad y generan grandes volúmenes de datos.

2.1.3. Tipos de machine learning

Para Raschka y Mirjalili (2019), la manera en que una máquina puede aprender se puede clasificar en tres categorías principales: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje reforzado.

Aprendizaje supervisado: En este enfoque, se enseña al algoritmo a realizar su tarea mediante un conjunto de datos previamente clasificados, basados en una determinada apreciación o concepto, con el fin de identificar patrones que puedan aplicarse en un análisis y generar una salida conocida.(Mueller y Massaron, 2021). Es decir, el aprendizaje supervisado implica que el algoritmo aprenda una función que asigna una entrada a una salida, basándose en ejemplos de pares entrada-salida. Este proceso utiliza datos de entrenamiento etiquetados, los cuales consisten en ejemplos específicos para inferir la función (Mahesh, 2020).

Aprendizaje no supervisado: Este tipo de aprendizaje, aunque similar al aprendizaje supervisado, se aplica a datos no clasificados ni etiquetados. El objetivo es identificar patrones y similitudes dentro de los datos, sin intervención externa sobre las respuestas correctas. Los algoritmos de aprendizaje no supervisado descubren la estructura subyacente en los datos por sí mismos. Se utiliza principalmente para tareas como agrupación de datos (clustering) y reducción de dimensionalidad (Mahesh, 2020; Bishop, 2007). En resumen, no existen respuestas predeterminadas, y el algoritmo aprende a reconocer patrones y clasificar los datos sin la supervisión de un "maestro".

Aprendizaje reforzado: En este tipo de aprendizaje automático, el sistema no es entrenado con datos clasificados o etiquetados. En lugar de eso, el modelo aprende en un entorno donde no se conoce la salida esperada. A través de la interacción con el entorno,

el sistema realiza acciones y recibe retroalimentación en forma de recompensas o penalizaciones. Este proceso refuerza el modelo a medida que resuelve el problema de la manera más eficiente posible, según las recompensas obtenidas (Mueller y Massaron, 2021; Manrique, 2020).

2.1.4. Regresión y clasificación

La regresión y la clasificación son las tareas habituales de los modelos de aprendizaje supervisado.

La clasificación es un problema que consiste en asignar automáticamente una etiqueta a un ejemplo no etiquetado. Un ejemplo conocido de clasificación es la detección de spam. En el aprendizaje automático, el problema de clasificación se resuelve mediante un algoritmo que toma un conjunto de ejemplos etiquetados como entradas y genera un modelo capaz de asignar una etiqueta a un ejemplo no etiquetado o producir un número que el analista de datos pueda usar para inferir fácilmente la etiqueta. Un ejemplo de este número podría ser una probabilidad. En un problema de clasificación, una etiqueta pertenece a un conjunto limitado de clases. Si el conjunto tiene dos clases (“enfermo” o “saludable”, “spam” o “no spam”), se trata de clasificación binaria (también conocida como binomial en algunos textos). La clasificación multiclase (o multinomial) se refiere a problemas con tres o más clases. Mientras que algunos algoritmos de aprendizaje están diseñados para manejar más de dos clases, otros son inherentemente algoritmos de clasificación binaria. Sin embargo, existen métodos para convertir un algoritmo de clasificación binaria en uno que maneje múltiples clases (Burkov, 2019).

La regresión es un problema de predicción en el que se estima una etiqueta de valor real (comúnmente conocida como objetivo) a partir de un ejemplo sin etiqueta. Un ejemplo clásico de regresión es la estimación del precio de una vivienda según características como el tamaño, el número de habitaciones, la ubicación, entre otras. Este problema se resuelve mediante un algoritmo de aprendizaje de regresión, el cual toma un conjunto de ejemplos etiquetados como entrada y genera un modelo capaz de predecir el objetivo a partir de un ejemplo sin etiqueta. (Burkov, 2019).

2.1.5. Algoritmos fundamentales de aprendizaje supervisado

El Machine Learning utiliza diversos algoritmos para abordar problemas relacionados con los datos. No hay un único algoritmo que sea el más adecuado para todos los problemas. La elección del algoritmo depende del tipo de problema a resolver, la

cantidad de variables involucradas, el modelo que mejor se ajuste a la situación, entre otros factores. (Mahesh, 2020).

Los algoritmos de aprendizaje automático supervisado son aquellos que requieren intervención externa. El conjunto de datos de entrada se divide en dos partes: un conjunto de entrenamiento y uno de prueba. El conjunto de entrenamiento incluye una variable de salida que debe ser predicha o clasificada. Los algoritmos aprenden patrones a partir de los datos de entrenamiento y luego los aplican al conjunto de prueba para realizar la predicción o clasificación. (Mahesh, 2020). Estos se describen a continuación.

2.1.6. Regresión lineal

La regresión lineal pertenece al grupo de los modelos lineales, los cuales son ampliamente utilizados en la práctica y han sido objeto de estudio durante las últimas décadas, con una trayectoria de más de cien años. Estos modelos realizan predicciones utilizando una función lineal basada en las características de entrada. La fórmula general de predicción para un modelo lineal es la siguiente:

$$\hat{y} = w[0] * x[0] + w[1] * x[1] + \dots + w[p] * x[p] + b \quad (1)$$

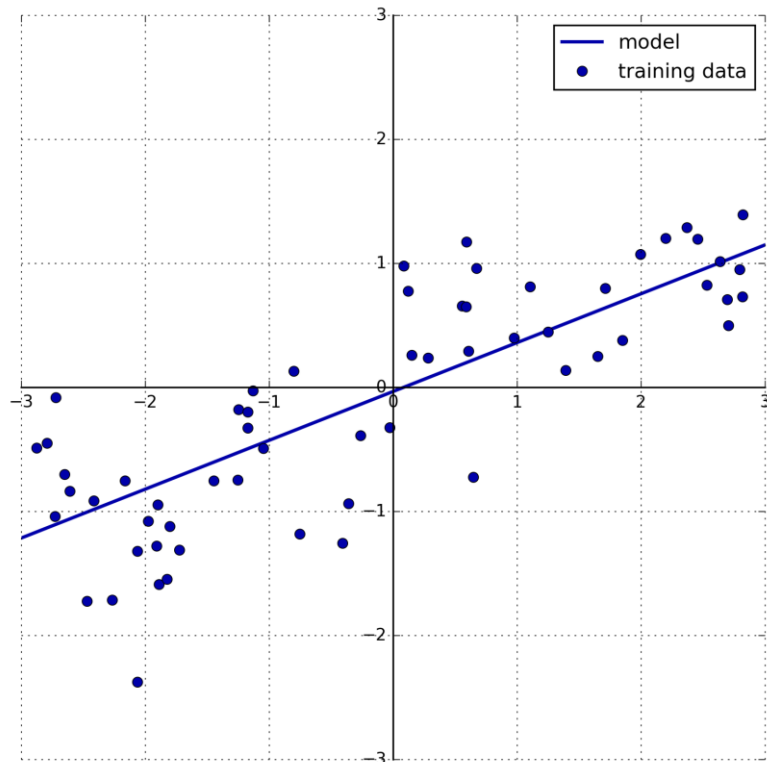
En este caso, $x[0]$ a $x[p]$ representan las características (siendo p el número total de características) de un único punto de datos, mientras que w y b son los parámetros del modelo que se ajustan durante el proceso de aprendizaje, y $\hat{y}$ es la predicción realizada por el modelo. Para un conjunto de datos con una sola característica, esto se expresa de la siguiente manera:

$$\hat{y} = w[0] * x[0] + b \quad (2)$$

Aquí, $w[0]$ es la pendiente y b es el desplazamiento del eje y . Para más funciones, w contiene las pendientes a lo largo de cada eje de función. Alternativamente, puede pensar en la respuesta pronosticada como una suma ponderada de las características de entrada, con pesos (que pueden ser negativos) dados por las entradas de w . Tratar de aprender los parámetros $w[0]$ y b en el conjunto de datos es el propósito de este algoritmo, un caso particular se puede observar en la Figura 2 (Müller y Guido, 2017).

Figura 2

Predicciones de un modelo lineal.



Fuente: (Müller y Guido, 2017).

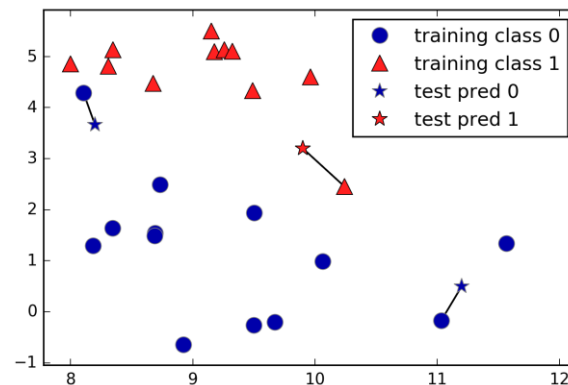
2.1.7. K-vecinos más cercanos (KNN)

El algoritmo K-vecinos más cercanos (K-Nearest Neighbors – KNN en inglés) es probablemente uno de los algoritmos más sencillos de aprendizaje automático. La creación del modelo se limita a almacenar el conjunto de datos de entrenamiento. Para realizar una predicción sobre un nuevo punto de datos, el algoritmo identifica los puntos más cercanos en el conjunto de entrenamiento, es decir, sus "vecinos más cercanos". (Müller y Guido, 2017). Se pueden aplicar en problemas de clasificación y regresión.

K-vecinos en clasificación. En su forma más básica, el algoritmo KNN considera únicamente un vecino más cercano, que es el punto de datos de entrenamiento más próximo al punto para el cual se quiere hacer una predicción. La predicción se obtiene simplemente tomando la salida conocida de este punto de entrenamiento (Müller y Guido, 2017). La Figura 3 ilustra esto para el caso de la clasificación en un conjunto de datos de entrenamiento.

Figura 3

Predicciones hechas por el modelo KNN.

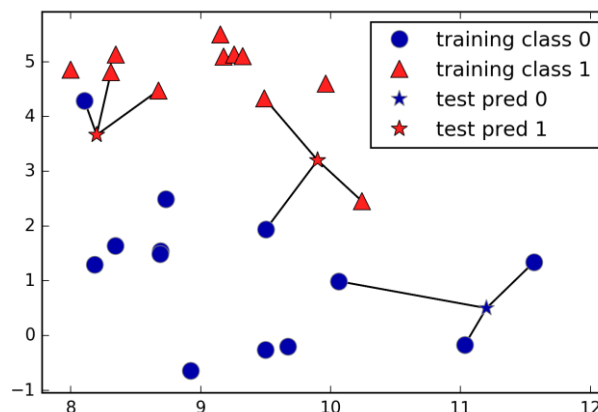


Fuente: (Müller y Guido, 2017).

En este caso específico, se añaden tres nuevos puntos de datos, representados por estrellas. Para cada uno de ellos, se identifica el punto más cercano en el conjunto de entrenamiento. La predicción del algoritmo de K-Vecinos más Cercanos es la etiqueta de ese punto, que se indica con el color de la cruz. En lugar de considerar solo al vecino más cercano, también es posible tomar en cuenta un número arbitrario, k , de vecinos. Este es el origen del nombre del algoritmo de k -vecinos más cercanos. Al incluir más de un vecino, se utiliza un sistema de votación para asignar una etiqueta. Esto implica que, para cada punto de prueba, se cuenta cuántos vecinos pertenecen a la clase 0 y cuántos a la clase 1. La clase que más se repite entre los k -vecinos más cercanos se asigna como la predicción, como se muestra en la Figura 4. (Müller y Guido, 2017).

Figura 4

Predicciones realizadas por 3 vecinos más cercanos.



Fuente: (Müller y Guido, 2017).

Una vez más, la predicción se indica mediante el color de la cruz. Se observa que la predicción para el nuevo punto de datos en la parte superior izquierda cambia cuando se utiliza más de un vecino, en comparación con el caso en el que solo se considera un vecino. Aunque este ejemplo corresponde a un problema de clasificación binaria, este enfoque también puede aplicarse a conjuntos de datos con cualquier número de clases. Para casos con más clases, se cuenta cuántos vecinos pertenecen a cada clase y, luego, se predice la clase más frecuente. (Müller y Guido, 2017).

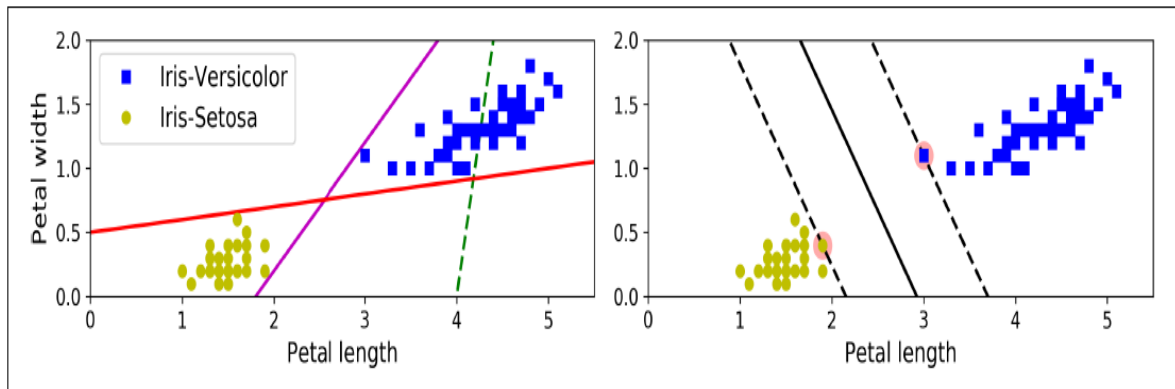
2.1.8. Máquina de vectores soporte (SVM)

Otra técnica avanzada y de amplio uso en aprendizaje automático es la Máquina de Vectores de Soporte (Support Vector Machine, SVM). En el ámbito del aprendizaje automático, las SVM son modelos supervisados que emplean algoritmos específicos para analizar los datos, con el fin de realizar tareas de clasificación y análisis de regresión. (Mahesh, 2020).

Las Máquinas de Vectores de Soporte (SVM) se utilizan en problemas de clasificación. Este modelo es especialmente útil para manejar datos complejos, donde las relaciones entre los datos no son lineales, es decir, cuando las observaciones deben clasificarse en diferentes grupos que no pueden separarse por un hiperplano (una línea que divide un espacio de N dimensiones en dos partes) en el espacio dimensional definido por los datos. En tales situaciones, es necesario proyectar los datos a un espacio de mayor dimensión, donde una función permita ordenar los datos en ese nuevo espacio utilizando un hiperplano. Por lo tanto, se busca un hiperplano que esté equidistante de los puntos más cercanos de cada grupo. La idea básica detrás de las SVM se ilustra en la Figura 5. En el gráfico de la izquierda, las dos clases se pueden separar fácilmente con una línea recta (son linealmente separables). Muestra los límites de decisión de tres posibles clasificadores lineales. El modelo con el límite de decisión representado por la línea discontinua es ineficaz, ya que no separa correctamente las clases. Los otros dos modelos funcionan bien en este conjunto de entrenamiento, pero sus límites de decisión están demasiado cerca de las instancias, lo que sugiere que no funcionarán tan bien con nuevos datos. En contraste, la línea continua en el gráfico de la derecha muestra el límite de decisión de un clasificador SVM, que no solo separa las clases, sino que también se aleja lo máximo posible de las instancias de entrenamiento más cercanas. Se puede visualizar un clasificador SVM como si ajustara la "calle" más ancha posible (representada por las líneas discontinuas paralelas) entre las dos clases. Esto se conoce como clasificación de margen amplio. (Géron, 2019)

Figura 5

Clasificación de margen amplio.



Fuente: (Géron, 2019).

Las máquinas de vectores de soporte (SVM) también se aplican en problemas de regresión, siendo conocidas como Support Vector Regression (SVR). A diferencia de la clasificación, donde el objetivo es encontrar un hiperplano que separe las clases, en la regresión se trata de encontrar una función que se ajuste a los datos y prediga un valor continuo. El enfoque fundamental de SVR consiste en identificar una función que minimice la diferencia entre los valores predichos y los reales, asegurando al mismo tiempo que los errores permanezcan dentro de un margen determinado. (Géron, 2019).

La metodología general para usar SVR en un problema de regresión es:

1. **Preparación de los datos:** Es necesario tener un conjunto de datos etiquetados con pares de características y valores objetivo.
2. **Normalización de los datos:** Es crucial normalizar las características para garantizar que todas estén en la misma escala.. Es posible utilizar técnicas como la estandarización o la normalización min-max.
3. **Selección del kernel:** Se debe elegir un kernel adecuado que determine cómo se mapearán los datos a un espacio de mayor dimensión. Los kernels comunes incluyen el lineal, polinomial y radial (RBF).
4. **Definición del modelo:** Configurar el SVR especificando el tipo de kernel y los parámetros asociados, como el margen y la penalización de errores. Estos parámetros pueden ajustarse mediante técnicas de validación cruzada o búsqueda de hiper parámetros.

5. **Entrenamiento del modelo:** Ajustar el SVR con los datos de entrenamiento. Durante este proceso, se busca encontrar la función que minimice los errores y se adapte a los datos de manera óptima.
6. **Predicción:** Emplear el modelo entrenado para hacer predicciones en nuevos datos de prueba. La función de regresión obtenida proporcionará una estimación del valor continuo correspondiente a las características de entrada.

Es importante tener en cuenta que, al igual que en la clasificación, las SVM pueden enfrentar problemas de eficiencia computacional en conjuntos de datos muy grandes debido a su naturaleza basada en vectores soporte. Sin embargo, existen enfoques como la formulación dual y el uso de métodos de aproximación para abordar este problema. Es necesario seleccionar los parámetros y el kernel según las características y la naturaleza de los datos, y evaluar el rendimiento del modelo mediante métricas de evaluación de regresión, como el error cuadrático medio (MSE) o el coeficiente de determinación (R^2) (Cervantes et al., 2020).

2.1.9. Árboles de decisión

Los árboles de decisión son modelos de aprendizaje automático y técnicas de minería de datos que son utilizados para tomar decisiones o hacer predicciones basadas en un conjunto de condiciones o características. Se les llama árboles de decisión debido a que la estructura del modelo se parece a un árbol invertido, donde las ramas indican las decisiones posibles, los nodos internos representan las características o atributos, y las hojas reflejan las conclusiones o resultados. Básicamente, aprenden una serie jerárquica de preguntas "Si-Entonces" que llevan a una decisión final (Müller y Guido, 2017).

En un árbol de decisión, se comienza con un nodo raíz que contiene un atributo o característica que se utiliza para realizar una partición en el conjunto de datos. Cada rama del nodo raíz representa un valor posible para ese atributo. Luego, se crean nodos internos adicionales que representan características adicionales y se van dividiendo los datos en función de esas características. El proceso de particionar los datos se repite recursivamente hasta que se alcanzan los nodos finales, que son las hojas del árbol. En las hojas, se toma una decisión o se hace una predicción basada en los datos de entrenamiento. Los árboles de decisión son populares porque son fáciles de interpretar y visualizar. Además, pueden manejar datos numéricos y categóricos, y son capaces de manejar problemas de clasificación y regresión. También se utilizan en conjunto con otras

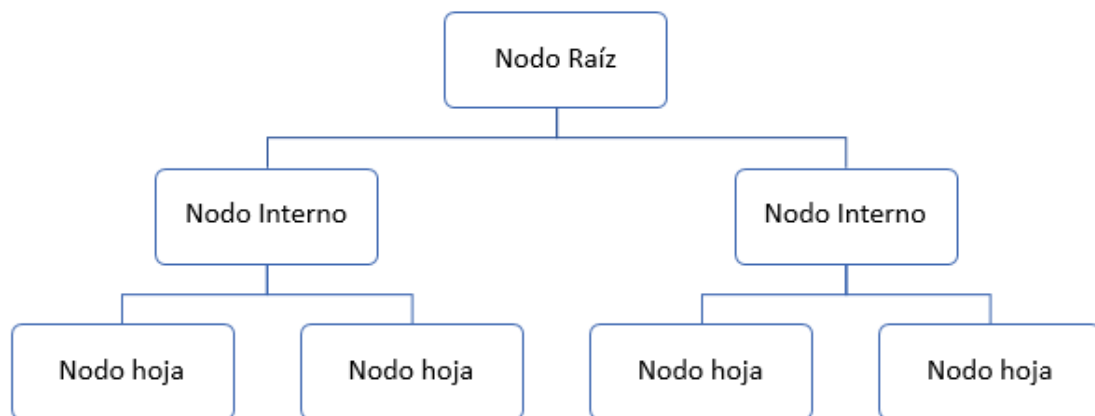
técnicas, como el ensamblaje de árboles, donde varios árboles de decisión se combinan para obtener predicciones más precisas (Müller y Guido, 2017).

Cuando un árbol de decisión "aprende", significa que aprende la secuencia de preguntas "Si-Entonces" que conduce a la respuesta correcta de la manera más rápida posible. En el entorno de aprendizaje automático, estas preguntas se denominan pruebas (que no deben confundirse con el conjunto de prueba, que son los datos que usamos para probar y ver qué tan generalizable es nuestro modelo). Normalmente, los datos no se presentan como características binarias (sí/no), sino que se expresan como características continuas. Las pruebas aplicadas a datos continuos suelen ser del tipo "¿Es la característica i mayor que el valor a ?". (Müller y Guido, 2017).

Como se muestra en la Figura 6, un árbol de decisión comienza con un nodo raíz, que no recibe ramas entrantes. Las ramas que salen de este nodo conducen a los nodos internos, también llamados nodos de decisión. Según las características disponibles, ambos tipos de nodos realizan evaluaciones para dividir los datos en subconjuntos homogéneos, los cuales se representan mediante nodos hoja o terminales. Los nodos hoja reflejan todos los resultados posibles dentro del conjunto de datos. (IBM, 2019).

Figura 6

Estructura de un árbol de decisión.



Fuente: (IBM, 2019).

2.1.10. Bosques aleatorios

Los algoritmos de Bosques Aleatorios (también conocidos como Random Forest en inglés) son métodos de aprendizaje automático muy utilizados que combinan las predicciones de varios árboles de decisión para producir un resultado final. Su popularidad

radica en su simplicidad y versatilidad, lo que les permite resolver tanto problemas de clasificación como de regresión.

En algunos estudios de predicción que comparan la regresión lineal con los bosques aleatorios, se ha observado que los bosques aleatorios ofrecen un mejor rendimiento predictivo. ¿Por qué funciona mejor el bosque aleatorio que la regresión lineal en las tareas de predicción? La regresión lineal parte de la premisa de linealidad, lo cual hace que el modelo sea fácil de interpretar, pero a menudo carece de la flexibilidad necesaria para realizar predicciones precisas. En cambio, los bosques aleatorios pueden adaptarse mejor a las no linealidades presentes en los datos, lo que les permite hacer predicciones más precisas. Además, los algoritmos de aprendizaje en conjunto, como los bosques aleatorios, son ideales para conjuntos de datos de tamaño mediano y grande. Cuando el número de variables independientes supera al de observaciones, los métodos de regresión lineal y regresión logística no pueden ejecutarse, ya que el número de parámetros a estimar excede el número de datos disponibles. Los bosques aleatorios, por su parte, funcionan porque no requieren el uso simultáneo de todas las variables predictoras. (Schonlau y Zou, 2020).

2.1.11. Redes neuronales

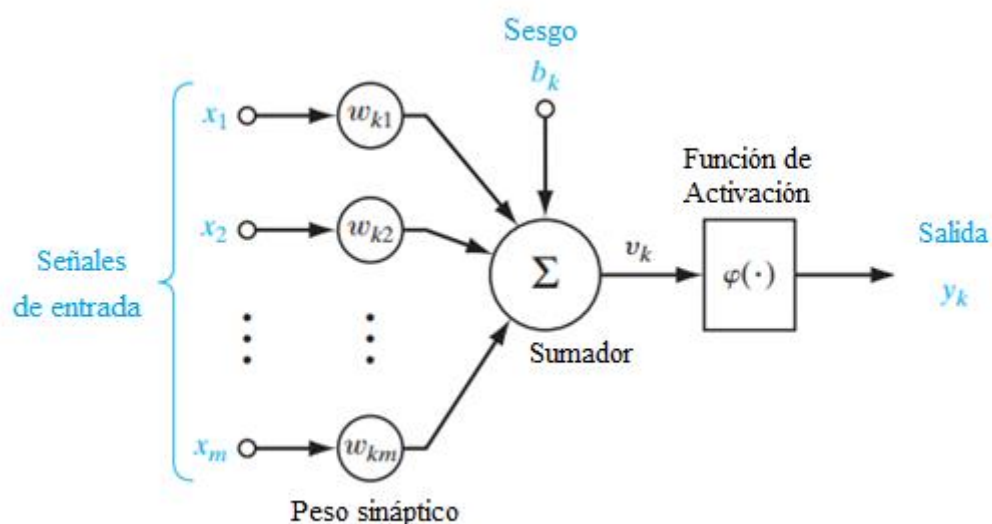
La investigación sobre las redes neuronales artificiales, también conocidas como redes neuronales o neuro redes, tiene como objetivo replicar la forma en que los sistemas nerviosos biológicos procesan la información. Su enfoque principal es entender cómo funciona el cerebro humano, que es considerablemente distinto al de una computadora digital convencional. El cerebro humano es un sistema extremadamente complejo, no lineal y paralelo, lo que le permite realizar varias tareas al mismo tiempo, a diferencia de los ordenadores tradicionales, que son secuenciales y ejecutan una sola tarea a la vez. En resumen, una neuro red es un procesador de información con una estructura altamente paralela, formada por muchas unidades de procesamiento simples denominadas neuronas (Izaurieta y Saavedra, 1999).

Una neurona es una unidad clave en el procesamiento de información dentro de una red neuronal. El diagrama de bloques mostrado en la Figura 7 ilustra el modelo de una neurona, que sirve como base para el diseño de diversas redes neuronales. Este modelo consta de tres elementos esenciales: (1) Un conjunto de sinapsis o conexiones, cada una con un peso o fuerza propia. Específicamente, una señal x_j que entra por la sinapsis j hacia la neurona k se multiplica por el peso sináptico w_{kj} . Es importante notar cómo se escriben

los subíndices del peso sináptico w_{kj} : el primer subíndice indica la neurona, mientras que el segundo subíndice se refiere a la sinapsis de entrada. A diferencia de los pesos en las sinapsis del cerebro, el peso sináptico en una neurona artificial puede ser tanto negativo como positivo. (2) Un sumador que se encarga de sumar las señales de entrada, ponderadas según las fuerzas sinápticas correspondientes a cada neurona. Este proceso constituye un combinador lineal. (3) Una función de activación que limita la amplitud de la salida de la neurona. También conocida como función de aplastamiento, su tarea es restringir el rango de la señal de salida a un valor finito. Por lo general, el rango de salida de una neurona se normaliza dentro del intervalo unitario cerrado $[0,1]$ o, alternativamente, en el intervalo $[-1,1]$.

Figura 7

Estructura de una neurona artificial.



Fuente: (Haykin, 2008).

El modelo neuronal de la Figura 7 muestra, además, la presencia de un sesgo externo, indicado por b_k . El sesgo b_k logra aumentar o disminuir la entrada neta de la función de activación, en función de si su valor es positiva o negativa. Matemáticamente, la neurona k mostrada en la Figura 7 queda expresada por las siguientes ecuaciones:

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (3)$$

$$y_k = \varphi(u_k + b_k) \quad (4)$$

Donde $x_1, x_2, \dots, x_m$ son las señales de entrada; $w_{k1}, w_{k2}, \dots, w_{km}$ son los respectivos pesos sinápticos de la neurona k ; u_k es la salida del sumador lineal de las señales de entrada; b_k es el sesgo; $\varphi(\cdot)$ es la función de activación; y y_k es la señal de salida de la neurona. La aplicación del sesgo b_k tiene el efecto de aplicar cálculos afines a la salida u_k del sumador lineal en el modelo de la Figura 7, cuya representación matemática es:

$$y_k = \varphi \left(\sum_{j=1}^m w_{kj} x_j + b_k \right) \quad (5)$$

2.1.12. XGBoost

El algoritmo de aumento de gradiente extremo XGBoost, es un algoritmo de aprendizaje conjunto que presenta múltiples ventajas: alta flexibilidad, gran capacidad de predicción, generalización robusta, alta escalabilidad, eficiencia en el entrenamiento de modelos y gran robustez (Zhang et al., 2022).

Desde su introducción en 2014, XGBoost se ha convertido en el algoritmo de aprendizaje automático preferido por científicos de datos e ingenieros de aprendizaje automático. Es una biblioteca de código abierto que permite entrenar y probar modelos con grandes volúmenes de datos. Ha sido ampliamente utilizado en diversos campos, desde la predicción de tasas de clics en anuncios hasta la clasificación de eventos físicos de alta energía. XGBoost es especialmente popular por su rapidez, sin que esta velocidad afecte la precisión. XGBoost es una implementación de árboles de decisión que aumentan el gradiente y ha sido utilizada por científicos e investigadores de datos de todo el mundo para optimizar sus modelos de aprendizaje automático. Está diseñado para ofrecer velocidad, facilidad de uso y rendimiento en grandes conjuntos de datos, sin necesidad de optimización de parámetros o ajustes, lo que permite su uso inmediato tras su instalación (Simplilearn, 2023).

XGBoost incluye regularización, lo que le permite controlar el sobreajuste mediante la incorporación de penalizaciones L1/L2 en los pesos y sesgos de cada árbol, una

característica que no está disponible en muchas otras implementaciones de aumento de gradiente. Además, XGBoost puede gestionar conjuntos de datos dispersos mediante el algoritmo de boceto de cuantiles ponderados, lo que le permite trabajar con entradas no nulas en la matriz de características manteniendo la misma complejidad computacional que otros métodos como el descenso de gradiente estocástico. XGBoost también está diseñado con una estructura de bloques para aprendizaje paralelo, lo que facilita su escalabilidad en máquinas o clústeres multinúcleo. Utiliza técnicas de reconocimiento de caché, lo que optimiza el uso de la memoria durante el entrenamiento de modelos con grandes volúmenes de datos. Por último, XGBoost ofrece capacidades de cómputo fuera del núcleo utilizando estructuras de datos basadas en disco en lugar de en memoria cuando se ejecuta cálculo. (Simplilearn, 2023).

XGBoost se utiliza principalmente por dos razones: la velocidad de ejecución y el rendimiento del modelo. La velocidad de ejecución es crucial, especialmente al trabajar con grandes conjuntos de datos, ya que no hay restricciones en el tamaño del conjunto de datos al utilizar XGBoost, lo que permite trabajar con volúmenes de datos que superarían las capacidades de otros algoritmos. El rendimiento del modelo es igualmente esencial, ya que permite crear modelos que pueden superar a otros en términos de precisión. Comparaciones con otros algoritmos, como el bosque aleatorio (RF), las máquinas de aumento de gradiente (GBM) y los árboles de decisión de aumento de gradiente (GBDT), han demostrado que XGBoost supera a estos otros algoritmos tanto en velocidad de ejecución como en rendimiento del modelo (Simplilearn, 2023).

2.1.13. Reducción de dimensionalidad

En el ámbito del machine learning, resolver problemas a menudo requiere manejar una gran cantidad de datos o características, lo que presenta dos desafíos significativos. En primer lugar, el entrenamiento del modelo se vuelve muy lento, y en segundo lugar, encontrar la solución adecuada se torna complicado. Este desafío, conocido como la "maldición de la dimensionalidad", surge debido al manejo de grandes volúmenes de datos. Afortunadamente, por lo general es posible reducir considerablemente la cantidad de características, simplificando así la resolución del problema (Géron, 2019).

Una de las técnicas utilizadas es el Análisis de Componentes Principales (PCA, por sus siglas en inglés). El objetivo principal del algoritmo PCA es reducir la dimensionalidad de un conjunto de datos con múltiples variables interrelacionadas, mientras se conserva la

mayor parte de la variabilidad presente en el conjunto original. Esto se consigue transformando los datos originales en un nuevo conjunto de variables, conocidas como componentes principales (CP), que son independientes entre sí y están ordenadas de tal manera que los primeros componentes capturan la mayor parte de la variación de las variables originales (Jolliffe, 2002).

2.1.14. Normalización y estandarización de datos

La eficacia de varios algoritmos de aprendizaje automático se optimiza cuando todas las características y el valor objetivo numérico se encuentran dentro de un rango similar. Reducir la escala de los valores de entrada también facilita a los algoritmos la búsqueda de soluciones, agilizando así el proceso. Por lo tanto, es común realizar una normalización de las muestras. Un método simple de normalización implica dividir cada muestra por su valor máximo (Bobadilla, 2021). La ecuación de normalización que permite llevar los datos a una escala de 0 a 1 es:

$$x_i = \frac{x_i - \min(X)}{\max(X) - \min(X)} \quad (6)$$

La normalización podría generar problemas si los datos están sesgados o incluyen valores atípicos. La presencia de estos valores atípicos puede comprometer la efectividad de la normalización. Por ejemplo, en una secuencia como -128, 0, 128, 2000, 240, 255, la normalización produce resultados que no reflejan adecuadamente la distribución de los datos (-0.064, 0.0, 0.064, 1, 0.12, 0.127). Por esta razón, la estandarización, también conocida como normalización z-score o "standardization", se considera una mejor alternativa. La estandarización ajusta los datos para que estén centrados en su media y distribuidos según la desviación estándar. Cada valor estandarizado puede interpretarse como el número de desviaciones estándar que lo separan de la media (Bobadilla, 2021).

$$x_i = \frac{x_i - \mu(X)}{\sigma(X)} \quad (7)$$

Donde x_i representa el valor de la variable a estandarizar, $\mu(x)$ es la media aritmética y $\sigma(x)$ la desviación estándar de la variable, con esto se garantiza que se tengan únicamente valores dentro de un determinado rango.

2.1.15. Evaluación de la calidad de la predicción

Los especialistas en el campo de la ingeniería de datos dedican una cantidad significativa de tiempo a mejorar la precisión de los modelos obtenidos. A menudo, no es factible evaluar visualmente la superioridad de un resultado sobre otro, por lo que se requiere el análisis de medidas de calidad para determinar la exactitud de los resultados. Según el enfoque de aprendizaje automático utilizado, se emplean diferentes métricas de calidad, algunas aplicables a la clasificación, otras a la regresión, y así sucesivamente. En este contexto, es esencial comprender y aplicar adecuadamente diversas medidas de calidad.

Sin embargo, antes de profundizar en estas métricas, es crucial establecer la metodología adecuada. Una regla fundamental en la metodología de validación es evitar evaluar la calidad del modelo utilizando el mismo conjunto de datos que se empleó para su entrenamiento. Esta práctica busca prevenir el sobreajuste del modelo, donde este memoriza los datos de entrenamiento, pero no generaliza bien con nuevos datos. Por lo tanto, para llevar a cabo una validación rigurosa, es imperativo dividir el conjunto de datos y los valores objetivos en varios conjuntos, tales como los conjuntos de entrenamiento, prueba y validación (Bobadilla, 2021).

El conjunto de datos de entrenamiento suele representar la mayor parte de los datos disponibles, generalmente alrededor del 80%, y se utiliza para entrenar al modelo. El conjunto de validación, por su parte, contiene ejemplos destinados a optimizar el modelo ajustando sus hiperparámetros, que son valores clave para controlar el comportamiento de los algoritmos. Por último, el conjunto de prueba sirve para medir el rendimiento del modelo utilizando datos completamente nuevos que no participaron en el entrenamiento ni en la validación. Es crucial que los tres conjuntos tengan una distribución de datos similar, sean representativos y no se solapen entre sí. (Bobadilla, 2021).

Después de entrenar el modelo con el conjunto de entrenamiento y optimizarlo mediante el conjunto de validación, se utiliza el conjunto de prueba para generar predicciones sobre datos nuevos no procesados anteriormente. La calidad del modelo se mide comparando estas predicciones con los valores reales. Dos métricas comunes para evaluar el desempeño son el Error Medio Absoluto (MAE) y el Error Cuadrático Medio (MSD), ambas diseñadas para penalizar las diferencias entre las predicciones y los valores esperados (Bobadilla, 2021). Sus ecuaciones son:

$$\text{MSD}(X, y) = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - p(X_i))^2 \quad (8)$$

$$\text{MAE}(X, y) = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - p(X_i)|^2 \quad (9)$$

A diferencia de los parámetros anteriores que ofrecen valores absolutos, la fórmula del coeficiente R^2 (R^2 score) proporciona valores relativos. Este indicador de calidad varía entre 0 y 1, donde 1 indica una predicción perfecta y 0 se corresponde con el modelo de regresión más básico, que consiste en pronosticar la media del conjunto de muestras de prueba. Se espera que el rendimiento del modelo evaluado no sea inferior al del regresor más básico. Al coeficiente R^2 también se le conoce como coeficiente de determinación. (Bobadilla, 2021) La fórmula de R^2 es:

$$R^2 = 1 - \frac{\sum_{i=0}^{n-1} (y_i - f(x_i))^2}{\sum_{i=0}^{n-1} (y_i - \bar{y})^2} \quad (10)$$

Donde $f(x)$ es la predicción del regresor para la muestra de x_i .

2.1.16. Python y el machine learning

Python se ha consolidado como el lenguaje universal para diversas aplicaciones en ciencia de datos. Combina la versatilidad de los lenguajes de programación de propósito general con la simplicidad de lenguajes especializados como MATLAB o R. Cuenta con bibliotecas que facilitan tareas como la carga de datos, visualización, análisis estadístico, procesamiento de lenguaje natural, manipulación de imágenes y más. Esta extensa colección de herramientas ofrece a los científicos de datos funcionalidades tanto generales como específicas.

Una de las grandes ventajas de Python es su capacidad para interactuar directamente con el código mediante herramientas como la terminal o Jupyter Notebook. Tanto el aprendizaje automático como el análisis de datos son procesos iterativos que dependen de los datos para guiar el análisis, por lo que resulta crucial contar con herramientas que faciliten la iteración rápida y la interacción sencilla. Además, como lenguaje de propósito general, Python permite desarrollar interfaces gráficas de usuario

(GUI), servicios web complejos e integrarse fácilmente en sistemas existentes. (Müller y Guido, 2017) .

Python es un lenguaje de programación dirigido a científicos e ingenieros que pueden no tener una formación formal en desarrollo de software. Se utiliza para crear prototipos, diseñar, simular y probar porque proporciona un ciclo de desarrollo incremental e inherentemente fácil, interoperabilidad con códigos existentes, acceso a una gran base de códigos fuente abiertos y confiables, y una filosofía de diseño jerárquica compartimentada. Python es un lenguaje interpretado, lo que significa que los códigos Python se ejecutan en una máquina virtual Python que proporciona una capa de abstracción entre el código y la plataforma en la que se ejecuta, haciendo que los códigos sean portátiles entre diferentes plataformas. Por ejemplo, el mismo script que se ejecuta en una computadora portátil con Windows también se puede ejecutar en una supercomputadora basada en Linux o en un teléfono móvil, facilitando así la programación porque la máquina virtual maneja los detalles de bajo nivel de implementación de la lógica empresarial del script en la plataforma subyacente.

Python es un lenguaje de tipo dinámico, lo que significa que el propio intérprete determina los tipos (por ejemplo, flotantes, enteros) de forma interactiva o en tiempo de ejecución. Esto contrasta con lenguajes como Fortran, que emplean compiladores que analizan el código de principio a fin, realizan optimizaciones a nivel de compilador, se vinculan estrechamente con bibliotecas específicas de una plataforma y luego generan un ejecutable independiente del compilador.

Además, Python ofrece una solución multiplataforma para códigos científicos. Al ser un proyecto de código abierto, está disponible en cualquier lugar donde se pueda compilar, y hoy en día es estándar en muchos sistemas operativos. Esto permite transferir scripts de una plataforma a otra siempre que las bibliotecas necesarias estén disponibles. Si faltan bibliotecas, configurarlas en diferentes sistemas solía ser laborioso, pero con la madurez de Python, ahora existe una amplia gama de bibliotecas pre-empaquetadas para las principales plataformas (Windows, MacOS, Linux, Unix).

Python facilita el mantenimiento de códigos científicos gracias a su sintaxis limpia, libre de distracciones visuales, y a sus herramientas integradas de prueba, documentación y desarrollo. Estas características son especialmente útiles para científicos sin formación en desarrollo de software.

Python también proporciona enlaces a bibliotecas de machine learning, como Scikit-learn, que es popular por su API coherente, su amplia gama de algoritmos y su

documentación clara. Desde su creación en 2007, Scikit-learn se ha consolidado como el módulo de aprendizaje automático de código abierto más popular, ampliamente adoptado tanto en la industria como en el ámbito académico (Unpingco, 2022).

2.1.17. Motor de combustión interna

Los motores de combustión interna alternativos (MCIA) son máquinas térmicas de desplazamiento positivo o volumétrico, donde el trabajo se produce a partir del movimiento lineal del émbolo conectado a un mecanismo de biela-manivela. Se denominan motores de combustión interna porque el calor necesario para el proceso se genera directamente en el fluido que se transforma dentro del motor (Rovira & Muñoz, 2015) .

Los motores, según su proceso de combustión, se clasifican en motores de encendido por chispa (MECH) y motores de encendido por compresión (MEC).

Los motores de encendido por chispa (MECH) también conocidos como motores de ciclo Otto o de encendido provocado (MEP) Son motores que funcionan con una mezcla homogénea de aire y combustible al final de la fase de compresión. La combustión se lleva a cabo por deflagración, donde la mezcla se quema a través de un frente de llama, y el encendido es iniciado por un agente externo, generalmente una chispa eléctrica.(Rovira & Muñoz, 2015)

Los motores de encendido por compresión (MEC), también conocidos como motores diésel o de ciclo diésel, funcionan comprimiendo aire en la cámara de combustión. La combustión se inicia al inyectar el combustible al final del proceso de compresión, donde parte de este se autoinflama debido a las altas temperaturas alcanzadas. El resto del combustible se quema mediante una combustión por difusión que ocurre de manera simultánea con la inyección. Las condiciones de compresión deben ser lo suficientemente intensas para garantizar la autoinflamación de la mezcla (Rovira & Muñoz, 2015).

2.1.18. Parámetros indicados de un motor de combustión interna

Son los parámetros resultantes que se obtienen del ciclo termodinámico del motor, sin incluir el proceso de admisión y escape, Estos resultados se pueden calcular a partir de la variación de la presión en el cilindro, la cual se mide mediante un sensor de presión ubicado en la cámara de combustión.

Trabajo indicado. Es el trabajo neto generado por el ciclo termodinámico básico (que incluye los procesos de compresión, combustión y expansión) en el diagrama del indicador, sin considerar la renovación de la carga; El trabajo indicado por ciclo $W_{c,i}$ (Trabajo en cada cilindro) se calcula integrando la curva, que es prácticamente el área encerrada en el diagrama (Rovira & Muñoz, 2015):.

$$W_{c,i} = \oint p dV \quad (11)$$

Potencia indicada. La potencia indicada por cilindro está relacionado al trabajo realizado en un ciclo, por:

$$P_i = \frac{W_{c,i} N}{60 n_R} \quad (12)$$

Donde

P_i : Potencia indicada, en W

W_{ci} es el trabajo indicado por ciclo de cada cilindro, en J.

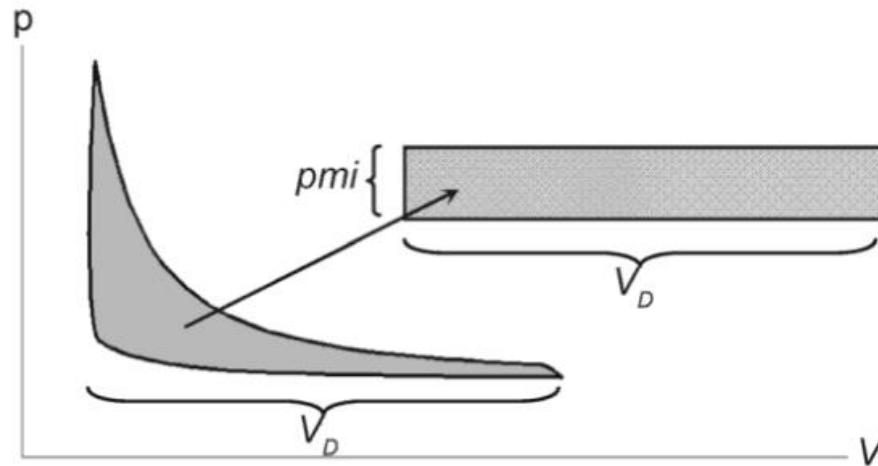
N : es la velocidad rotacional del cigüeñal, en rpm.

n_r : es el número de revoluciones que se tiene que alcanzar para generar una carrera de trabajo, así $n_r = 2$, para motores de cuatro tiempos, $n_r = 1$ para motores de 2 tiempos.

Presión media indicada (pmi). Determinar matemáticamente la potencia generada, implica determinar el trabajo indicado por ciclo según la ecuación (11), conlleva ejecutar complicados cálculos. Para facilitar dichos cálculos se define en el mismo diagrama p-v una presión equivalente, llamada presión media (P_m) de tal forma que entregue una región de la misma área del ciclo indicado, de tal forma que el trabajo se mantenga constante. Así pues, el trabajo indicado por ciclo será igual al área del rectángulo mostrado en la Figura 8, el cual es el producto de la cilindrada por la presión media (Heywood, 2018). Existen dos tipos de presión media, basados si el trabajo es hecho por el gas sobre el pistón (presión media indicada), o por el trabajo disponible como salida en el eje del motor (presión media efectiva).

Figura 8

Interpretación geométrica de la presión media indicada



Fuente: (Rovira & Muñoz, 2015)

$$P_{im} = \frac{\text{Area del diagrama indicado}}{\text{Cilindrada}} \quad (13)$$

2.1.19. Parámetros efectivos de un motor de combustión interna

Son los resultados que se pueden obtener en las pruebas del motor en el banco de pruebas. Son similares a los parámetros indicados, pero a diferencia de estos, consideran el motor en su totalidad; por lo tanto, tienen en cuenta las diversas pérdidas mecánicas del motor, incluyendo el trabajo utilizado en el lazo de bombeo (ejecutar la carrera de admisión y la carrera de escape), y las magnitudes se aplican al motor completo y no a un solo cilindro (Rovira & Muñoz, 2015).

Par efectivo. Determinado en prueba dinamométrica

Potencia efectiva. Determinado en prueba dinamométrica

Presión media efectiva (pme) es el valor equivalente a la pmi pero contabilizando el trabajo efectivo en lugar del indicado. (Rovira & Muñoz, 2015) .La presión media efectiva es la presión teórica y constante que se encuentra en el interior de los cilindros de un motor que teóricamente produce potencia semejante a la potencia determinada en un banco dinamométrico (Obert, 1997).

Se define:

$$Pme = \frac{W_{ciclo}}{V_d} \quad (14)$$

Donde

W_{ciclo} : es el trabajo indicado por ciclo de todos los cilindros, en J.

V_d : volumen de todos los cilindros que posee el motor, cilindrada. En m^3 .

Sin embargo, de la ecuación (12), se sabe que:

$$W_c = 60 \frac{n_R P}{N} \quad (15)$$

$$Pme = 60 \frac{n_R P}{N V_d} \quad (16)$$

Donde

Pme : es la presión media efectiva. En Pa.

P : es la potencia del motor. En W.

V_d : Volumen de todos los cilindros que posee el motor, cilindrada. En m^3 .

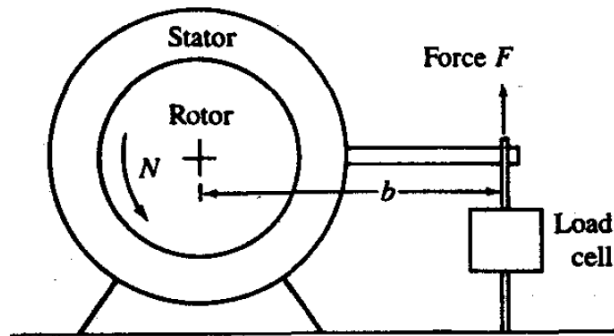
N : es la velocidad rotacional del cigüeñal, en rpm.

2.1.20. Torque al freno y Potencia al freno

El torque del motor es normalmente medido con un instrumento llamado dinamómetro, el cual le da una resistencia a la operación del motor mediante la adición de carga mecánica, hidráulica, eléctrica por generación, o eléctrica por corrientes parasitas. EL método comúnmente empleado para medir el torque es el mostrado en la Figura 9. Donde la volante del motor está conectado al rotor del dinamómetro, dicho rotor que soportado en rodamientos de baja fricción es acoplado electromagnéticamente, hidráulicamente, o mediante fricción mecánica a su estator, el cual esta balanceado con el estator estacionario del dinamómetro. El torque ejercido sobre el estator debido a la rotación de la volante de motor es medido por la célula de carga, que puede ser determinado por balanceo del estator con pesos, resortes, o medios neumáticos (Heywood, 2018) .

Figura 9

Esquema del principio de operación de un dinamómetro



Fuente (Heywood, 2018).

Usando la notación de la Figura 19, si el torque ejercido por el motor es T:

$$T = Fb \quad (17)$$

La potencia P generada por el motor y absorbida por el dinamómetro es producto del torque y la velocidad angular ω , la cual puede expresarse en función de su frecuencia.

$$P = 2\pi fT \quad (18)$$

O en función del régimen de giro del motor N, medido regularmente en rpm.

$$P = \frac{\pi N}{30} T \quad (19)$$

Dónde:

P: potencia en W.

T torque en Nm.

N: es la velocidad rotacional del cigüeñal, en rpm.

El torque es una medida de habilidad del motor para hacer trabajo, la potencia es la rapidez con la cual el trabajo es hecho.

El valor de la potencia medida de la forma anteriormente medida, se llama potencia al freno P_b . Esta potencia es la potencia útil entregada por el motor frente a una carga; en este caso el freno.

2.1.21. Eficiencia mecánica

Una parte de la potencia indicada bruta se utiliza para expulsar los gases de escape e introducir aire fresco; otra parte se destina a superar la fricción entre el movimiento relativo de los componentes internos del motor, además de impulsar los accesorios externos del motor. Toda esta potencia requerida se agrupa bajo el término "potencia de fricción" (P_f), dejando como potencia disponible en el eje la denominada potencia al freno. Las potencias indicadas brutas, las potencias de fricción y la potencia al freno, tienen la siguiente relación:

$$P_{ig} = P_b + P_f \quad (20)$$

La potencia de fricción es complicada de determinar con precisión. Una forma común de aproximarla en motores de alta velocidad es hacer funcionar el motor con un dinamómetro (por ejemplo, operando el motor sin encenderlo) y medir la potencia que debe ser proporcionada por el dinamómetro para superar todas las pérdidas por fricción. Durante las pruebas, se mantienen constantes la velocidad del motor, las temperaturas del aceite y del refrigerante, así como las condiciones ambientales, tanto en las pruebas sin encender como en las de funcionamiento normal del motor. La mayor fuente de inexactitud con este método proviene de la presión de los gases que fuerzan el pistón y sus anillos son bajos en el motor testeado en comparación cuando el motor está en operación, además que la temperatura de aceite en las paredes del cilindro es más reducida bajo las condiciones de la prueba.

La relación entre la potencia al freno o potencia útil, entregada por el motor a la potencia indicada es llamada la eficiencia mecánica:

$$\eta_m = \frac{P_b}{P_{ig}} = 1 - \frac{P_f}{P_{ig}} \quad (21)$$

Ya que la potencia de fricción incluye la potencia requerida para ejecutar los procesos de admisión y de escape del ciclo de trabajo; la eficiencia mecánica se encuentra dependiente de restricciones del flujo de aire, así como del diseño del motor, así como la velocidad del motor.

2.1.22. Potencia efectiva del motor

La potencia efectiva del motor se puede estimar con la siguiente ecuación:

$$P_{ef} = PC \cdot \dot{m} \cdot \eta \quad (22)$$

Donde:

P_{ef} es la potencia efectiva del motor (en kW)

PC es el poder calorífico del combustible (en kJ/kg)

$\dot{m}$ es el flujo másico de combustible (en kg/s)

η es la eficiencia de conversión

Se considera que el poder calorífico del diésel es generalmente de 43000 kJ/kg, y que la eficiencia de conversión del calor de combustión del motor convertido en potencia efectiva, generalmente es del 40% (Tan et al., 2021). Encontrándose el rendimiento térmico para motores diésel de aplicación automotriz en el rango de 35-40%, y para motores diésel de aplicación marítima y de generación esta eficiencia térmica puede alcanzar hasta el 50%.

2.1.23. Factor de carga

Los factores de carga (llamados también porcentajes de carga), que representan la parte de la potencia del motor utilizada, se emplean para estimar el consumo de combustible en equipos impulsados por motores de combustión. Estos factores son específicos para cada tipo de equipo y sus condiciones de operación (Klanfar et al., 2016).

El factor de carga se define por la ecuación (Tan et al., 2021):

$$FC = \frac{P_{ef}}{P_{nominal}} \quad (23)$$

donde

FC es el factor de carga del motor (en escala porcentual),

P_{ef} es la potencia efectiva durante la operación del motor (en kW)

$P_{nominal}$ es la potencia nominal del motor obtenida en dinamómetro (en kW),

La potencia nominal es la potencia máxima al freno de un motor según lo especificado por el fabricante del mismo.

Los factores de carga pueden variar según la aplicación de la máquina que es propulsada por el motor de combustión interna.

Tabla 1

Factores típicos de carga de camiones, según Kennedy.

Tipo de camión	Factor de carga*		
	Ligero	Promedio	Pesado
Volquete trasero convencional	0,25	0,35	0,50
Tractor camión	0,35	0,50	0,65
Camión de descarga inferior	0,25	0,35	0,50

*Ligero: Considerable ralentí, acarreo cargados en pendientes favorables y buenos caminos de acarreo

Promedio: Ralentí normal, acarreo cargados en pendientes adversas y buenos caminos de acarreo

Pesado: Ralentí mínimo, acarreo cargados en pendientes adversas pronunciadas

Fuente: (Klanfar et al., 2016).

Tabla 2

Factores típicos de carga para varios equipos, según Day.

Tipo de equipo	Condiciones de operación		
	Excelente	Promedio	Severo
De ruedas, carretera pavimentada	0,25	0,30	0,40
De ruedas, todoterreno	0,50	0,55	0,60
De oruga	0,50	0,63	0,75
Excavadoras motorizadas	0,50	0,55	0,60

Fuente: (Klanfar et al., 2016).

Tabla 3

Factores de carga en diferentes condiciones de funcionamiento, según Chitkar.

Tipo de equipo	Condiciones de operación		
	Ligero	Promedio	Severo
Tractor de oruga	0,60	0,70	0,80
Cargador Frontal	0,35	0,45	0,55
Camión	0,25	0,35	0,45

Fuente: (Klanfar et al., 2016).

2.2. Marco conceptual

Machine learning. Es un campo de la inteligencia artificial que desarrolla sistemas capaces de aprender de manera autónoma. En este caso, "aprender" implica reconocer patrones complejos en grandes cantidades de datos. Un algoritmo de aprendizaje automático examina estos datos y puede predecir comportamientos futuros sin necesidad de intervención humana. (IBM, 2019).

Algoritmo de machine learning. Un algoritmo de machine s un conjunto de procedimientos o directrices que emplea un sistema de inteligencia artificial para llevar a cabo tareas, como identificar patrones o predecir resultados a partir de variables de entrada. Gracias a estos algoritmos, el machine learning es capaz de "aprender".(IBM, 2023)

Dinamómetro. Un dinamómetro es un equipo utilizado en bancos de pruebas para motores de combustión interna que mide la rotación del motor, el torque y la potencia al freno, entre otros parámetros (Martyr y Plint, 2012)

Capítulo III. Desarrollo del trabajo de investigación

3.1. Panorama del desarrollo de la investigación

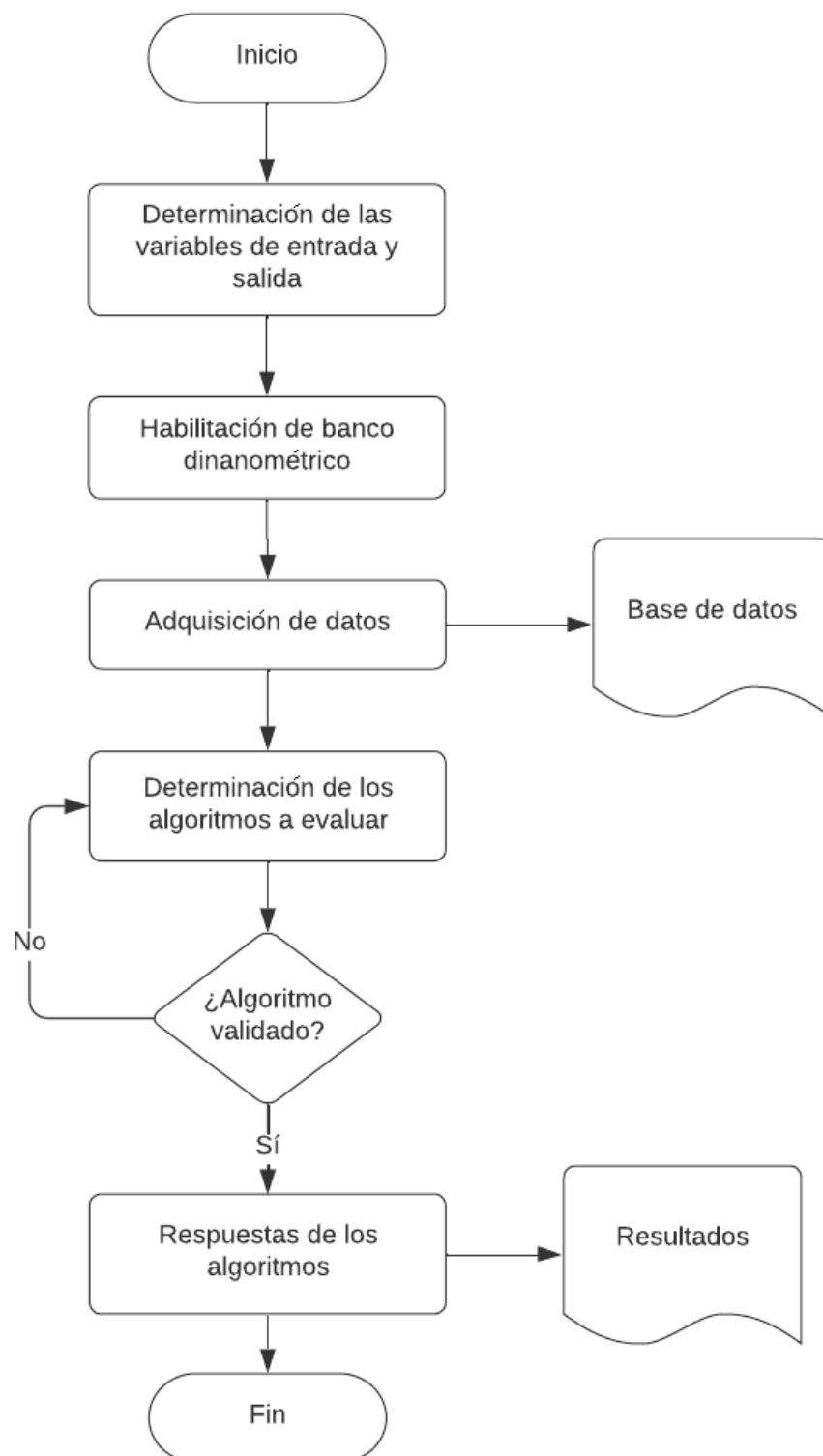
La investigación busca desarrollar un modelo de predicción de las características de desempeño de un motor de combustión interna a través de parámetros de entrada. El modelo se desarrolla mediante la evaluación de diferentes algoritmos de machine learning en la estrategia de aprendizaje supervisado. Los algoritmos considerados son la regresión lineal múltiple (MLR), k vecinos más cercanos (KNN), máquinas de vectores soporte (SVM), árboles de decisión (DT), bosques aleatorios, redes neuronales artificiales (ANN) y XGBoost. La selección de estos algoritmos permite comprobar su efectividad, comparando modelos sencillos (MLR) y complejos (ANN), así como modelos ampliamente utilizados (ANN) con los más novedosos (XGBoost).

El proceso mostrado en la Figura 10 consta de las siguientes etapas:

1. Determinación de los parámetros de entrada y salida: Esto está referido a la identificación de los parámetros de mayor relevancia tanto en la entrada como en la salida, seleccionados según los objetivos de la investigación y la experiencia del investigador.
2. Habilitación del banco dinamométrico para desarrollar el modelo, es necesario obtener datos sincrónicos entre la entrada y la salida durante diversas condiciones de operación del motor.
3. Adquisición de datos: Los parámetros requeridos para esta investigación se tomarán de la memoria del controlador del motor, que permite volcar sus parámetros a un archivo CSV, obteniéndose así la base de datos.
4. Determinación de los algoritmos a evaluar: Dado el interés en obtener precisión basada en datos previamente registrados, se seleccionan algoritmos de aprendizaje supervisado y se verifica su comportamiento.
5. Respuesta de los algoritmos: Una vez entrenados y validados en su comportamiento, se registran los resultados de la evaluación comparativa entre ellos.

Figura 10

Diagrama de flujo de las actividades de la investigación.



Fuente: Elaboración propia.

3.2. Banco de ensayos

El banco de ensayos se compone el freno dinamométrico y el sistema de registro de datos, los que se describen a continuación:

3.2.1. Motor por evaluar

Para el desarrollo de esta investigación se recopilaron datos del motor diésel marca Komatsu modelo SAA12V140E-3, controlado electrónicamente, utilizado en el camión minero HD785-7 con número de serie 500878 nuevo con sólo 3.8 horas de servicio acumulado (Figura 11). Las especificaciones del motor se pueden observar en la Tabla 4.

Figura 11

Motor diésel Komatsu modelo SAA12V140-3, instalado en el dinamómetro.



Fuente: Elaboración propia.

Tabla 4*Características del motor ensayado.*

Marca del motor		Komatsu
Denominación del motor		SAA12V140E-3
Modelo de la máquina		HD785-7
Número de cilindros		12
Diámetro x carrera	mm	140 x 165
Cilindrada	l (cm ³)	30,5 {30480}
Orden de inyección		R1-L1-R5-L5-R3-L3-R6-l6-R2-L2-R4-l4
Potencia nominal de salida	kW (HP)@rpm	895 (1200) @ 1900 rpm
Torque, máx.	N.m (kg.m)@rpm	5,076 (518) @ 1350 rpm
Velocidad máxima, sin carga	rpm	2250 ± 25
Velocidad mínima, sin carga	rpm	650 ± 20
Consumo específico a la potencia nominal	g/kW·h (g/HP·h)	207 (154)

Nota: Adaptado del manual de servicio de Komatsu.

3.2.2. Freno dinamométrico

El freno dinamométrico es responsable de crear las condiciones de carga durante la operación del motor. De esta forma, se toman las lecturas del torque o par, el régimen de giro y la potencia al freno. El freno dinamométrico está acoplado al volante del motor a ensayar mediante un árbol de transmisión con juntas universales.

El freno dinamométrico empleado en esta investigación es del fabricante Power Test, modelo 50X de 1000 HP, del tipo hidráulico. Este equipo cuenta con un sistema de monitoreo de potencia, torque, rpm, y permite la lectura de diversos puntos de temperatura y presión de los fluidos del motor de combustión, los cuales pueden observarse a través del monitor del banco (Figura 12).

3.2.3. Instrumentación

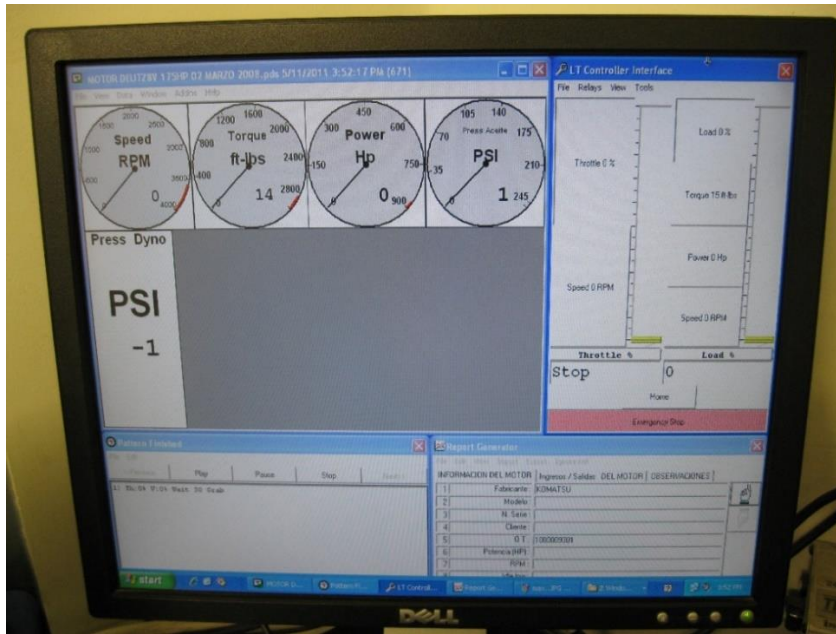
Para la lectura y registro de los parámetros de operación del motor diésel se ha empleado, hardware de comunicación INLINE 6 y software de monitoreo INSITE ambos desarrolladas por el fabricante de motores diésel Cummins Inc.

El hardware requerido para la comunicación con el ECM es mostrado en la Figura 13, el que consta de (1) Cable DB25, (2) Botón de reseteo, (3) Computadora personal, (4) Interfaz

de comunicación Inline 6, (5) Cable mini backbone, (6) Cable cambiador de género, (7) Cable de conexión serial o conexión USB, (8) Terminal de comunicación del arnés de motor diésel.

Figura 12

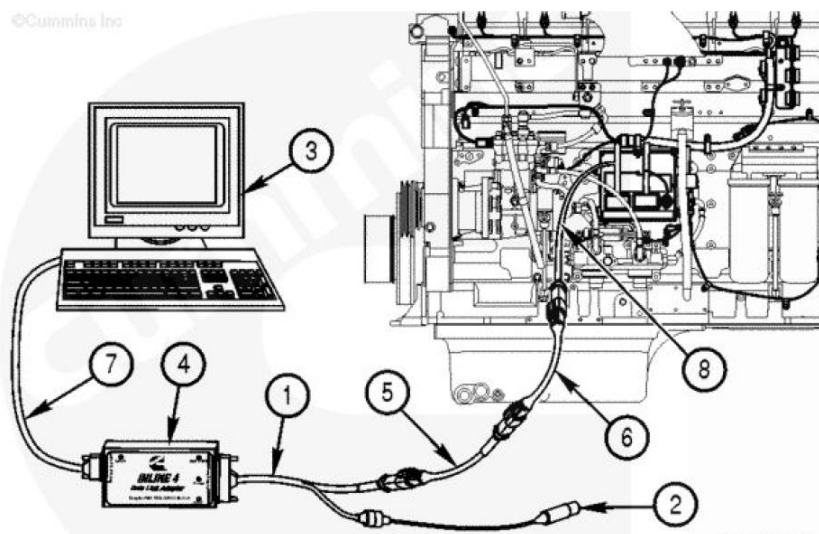
Detalle del panel de monitoreo del dinamómetro.



Fuente: Elaboración propia.

Figura 13

Hardware de comunicación para el monitoreo del motor diésel.



Fuente: (Cummins inc, 2012).

El uso de la interfaz Inline 6, mostrada en la Figura 14, es imprescindible para establecer la comunicación entre la computadora personal y el ECM a través del protocolo de comunicación CAN Bus j1939 del equipo, ya que este último no está implementado en las computadoras personales. Para iniciar la comunicación con el ECM y acceder a la visualización de los datos transmitidos en tiempo real, es necesario utilizar el software Insite, desarrollado por la marca de motores diésel Cummins (Figura 15).

Figura 14

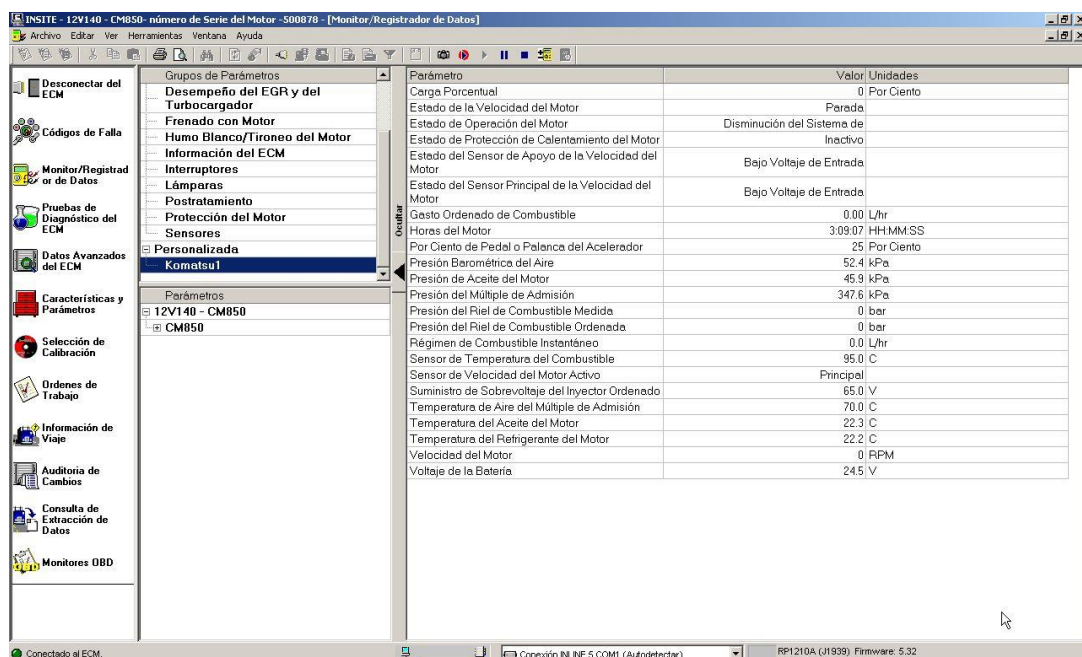
Interfaz de comunicación Inline 6.



Fuente: Elaboración propia.

Figura 15

Software de monitoreo Insite.



Fuente: Elaboración propia.

3.3. Proceso de registros de los monitoreos

Con el banco dinamométrico equipado, se ejecutan los procesos de monitoreo y registro del comportamiento del motor diésel durante las pruebas dinamométricas. Esto se realiza mediante la toma de los parámetros de funcionamiento del motor, los cuales son leídos directamente por los sensores, así como las señales de salida hacia los actuadores. Todos estos datos son exportables a un archivo de datos en formato *.csv, que posteriormente se convierte a un archivo *.xls para su análisis.

Los registros fueron obtenidos con el software INSITE, que se configuró en idioma español, y los datos se configuraron para mostrarse en unidades del Sistema Internacional (S.I.). Sin embargo, en lo que respecta a las magnitudes de temperatura de los fluidos (ya sea aire, refrigerante o aceite), solo existía configuración para temperaturas relativas, en °C o °F. Por lo tanto, los registros se almacenaron en °C.

3.4. Parámetros disponibles en el controlador de motor

Se hace una revisión de los parámetros disponibles para lectura en el controlador principal del motor SAA12V140E-3, siendo los mostrados en la Tabla 5.

Tabla 5

Parámetros del controlador principal del motor SAA12V140E-3.

#	Parámetro
1	Fecha
2	Tiempo
3	Calentador 1 del Aire de Admisión - Identificador 0
4	Calentador 1 del Aire de Admisión - Identificador 1
5	Calentador 2 del Aire de Admisión - Identificador 0
6	Calentador 2 del Aire de Admisión - Identificador 1
7	Carga Porcentual (%) - Identificador 0
8	Conteos de Conexión del Interruptor de Llave - Identificador 0
9	Conteos de Desconexión del Interruptor de Llave - Identificador 0
10	Curva de Torque Seleccionada - Identificador 0
11	Curva de Torque Seleccionada - Identificador 1
12	Disminución Activa de Velocidad del Turbocargador - Identificador 0
13	Disminución Activa de Velocidad del Turbocargador - Identificador 1
14	Estado de Operación del Motor - Identificador 0
15	Estado de Protección de Calentamiento del Motor - Identificador 0
16	Estado de la Lámpara Espere Para Arrancar - Identificador 0
17	Estado de la Lámpara Espere Para Arrancar - Identificador 1
18	Estado de la Lámpara Roja de Paro - Identificador 0

19	Estado de la Lámpara Ámbar de Advertencia - Identificador 0
20	Estado de la Señal del Sensor Principal de la Velocidad del Motor - Identificador 0
21	Estado de la Señal del Sensor de Apoyo de la Velocidad del Motor - Identificador 0
22	Estado de la Velocidad del Motor - Identificador 0
23	Estado del Sensor Principal de la Velocidad del Motor - Identificador 0
24	Estado del Sensor de Apoyo de la Velocidad del Motor - Identificador 0
25	Fase del Software de Calibración - Identificador 1
26	Fuel Rail Pressure Commanded (bar) - Identificador 0
27	Gasto Ordenado de Combustible (L/h) - Identificador 0
28	Horas del Motor (HH:MM:SS) - Identificador 0
29	Horas del Motor (HH:MM:SS) - Identificador 1
30	Interruptor de Anulación del Paro de Protección del Motor - Identificador 0
31	Interruptor de Anulación del Paro de Protección del Motor - Identificador 1
32	Interruptor de Ralentí Bajo Alternó Conmutado - Identificador 0
33	Interruptor de Validación de Ralentí - Identificador 0
34	Interruptor de Llave - Identificador 0
35	Interruptor del Modo de Prueba de Diagnóstico - Identificador 0
36	LSI Increment Switch State - Identificador 1
37	Nivel de Refrigerante del Motor - Identificador 0
38	Nivel de Refrigerante del Motor - Identificador 1
39	Por Ciento de Pedal o Palanca del Acelerador (Por Ciento) - Identificador 0
40	Presión de Aceite del Motor (kPa) - Identificador 0
41	Presión de Aire Barométrica del OEM (kPa) - Identificador 0
42	Presión del Múltiple de Admisión (kPa) - Identificador 0
43	Presión del Riel de Combustible Medida (bar) - Identificador 0
44	Presión del Riel de Combustible Medida (bar) - Identificador 1
45	Régimen de Combustible Instantáneo (L/h) - Identificador 0
46	Régimen de Combustible Instantáneo (L/h) - Identificador 1
47	Selección de Caída Conmutada - Identificador 0
48	Selección de Caída Conmutada - Identificador 1
49	Sensor de Temperatura del Combustible (°C) - Identificador 0
50	Sensor de Temperatura del Combustible (°C) - Identificador 1
51	Sensor de Velocidad del Motor Activo - Identificador 0
52	Suministro de Sobrevoltaje del Inyector Medido (V) - Identificador 0
53	Suministro de Sobrevoltaje del Inyector Medido (V) - Identificador 1
54	Suministro de Sobrevoltaje del Inyector Ordenado (V) - Identificador 0
55	Suministro de Sobrevoltaje del Inyector Ordenado (V) - Identificador 1
56	Suprimir Disminución de Potencia para Protección del Motor - Identificador 0
57	Suprimir Disminución de Potencia para Protección del Motor - Identificador 1
58	Temperatura de Aire Ambiente del OEM (°C) - Identificador 0
59	Temperatura de Aire Ambiente del OEM (°C) - Identificador 1
60	Temperatura de Aire del Múltiple de Admisión (°C) - Identificador 0
61	Temperatura del Aceite del Motor (°C) - Identificador 0
62	Temperatura del Refrigerante del Motor (°C) - Identificador 0
63	Tiempo del ECM (Tiempo de Conexión de la Llave) (HH:MM:SS) - Identificador 0
64	Tiempo del ECM (Tiempo de Conexión de la Llave) (HH:MM:SS) - Identificador 1
65	Tipo de Gobernador - Identificador 0

66	Tipo de Gobernador - Identificador 1
67	Tipo de Sensor de Presión de Aceite del Motor - Identificador 0
68	Velocidad del Motor (rpm) - Identificador 0
69	Voltaje de Alimentación del Sensor de Posición del Pedal o Palanca del Acelerador (V) - Identificador 0
70	Voltaje de Señal del Sensor de Posición del Pedal o Palanca del Acelerador (V) - Identificador 0
71	Voltaje de Señal del Sensor de Presión de Aceite del Motor (V) - Identificador 0
72	Voltaje de Señal del Sensor de Presión de Aire Barométrica del OEM (V) - Identificador 0
73	Voltaje de Señal del Sensor de Presión del Múltiple de Admisión (V) - Identificador 0
74	Voltaje de Señal del Sensor de Presión del Riel de Combustible (V) - Identificador 0
75	Voltaje de Señal del Sensor de Presión del Riel de Combustible (V) - Identificador 1
76	Voltaje de Señal del Sensor de Temperatura de Aceite del Motor (V) - Identificador 0
77	Voltaje de Señal del Sensor de Temperatura de Aire Ambiente del OEM (V) - Identificador 0
78	Voltaje de Señal del Sensor de Temperatura de Aire Ambiente del OEM (V) - Identificador 1
79	Voltaje de Señal del Sensor de Temperatura del Aire del Múltiple de Admisión (V) - Identificador 0
80	Voltaje de Señal del Sensor de Temperatura del Refrigerante del Motor (V) - Identificador 0
81	Voltaje de la Batería (V) - Identificador 0
82	Voltaje de la Batería (V) - Identificador 1
83	Voltaje del Sensor de Temperatura del Combustible (V) - Identificador 0
84	Voltaje del Sensor de Temperatura del Combustible (V) - Identificador 1

Fuente: Elaboración propia.

3.5. Conformación de la base de datos

Dado que existen 84 campos disponibles para monitorear el comportamiento del motor, se realiza una revisión para seleccionar aquellos que contienen información numérica y relevante sobre el desempeño del motor diésel. Basado en la experiencia del investigador. A estos registros se les añadieron las condiciones de lectura de la instrumentación del panel del dinamómetro, acoplando así los datos de las pruebas de carga al 20%, 50%, 80%, 90% y 100%. Se obtuvo un total de 1915 registros, estableciéndose con ellos los parámetros de salida en potencia, torque y consumo de combustible. Mostrados en la Tabla 6.

Tabla 6*Parámetros que conforman la base de datos.*

#	Parámetro	Unidad
1	Aplicación del Pedal del Acelerador	%
2	Presión Barométrica del Aire	kPa
3	Presión de Aceite del Motor	KPa
4	Presión del Múltiple de Admisión	Bar
5	Presión del Riel de Combustible Ordenada	Bar
6	Presión del Riel de Combustible Medida Izquierda	Bar
7	Presión del Riel de Combustible Medida Derecha	°C
8	Temperatura del Combustible Izquierda	°C
9	Temperatura del Combustible Derecha	°C
10	Temperatura de Aire del Múltiple de Admisión	°C
11	Temperatura del Aceite del Motor	°C
12	Temperatura del Refrigerante del Motor	°C
13	Velocidad del Motor	rpm
14	Torque instantáneo	N.m
15	Potencia instantánea	kW
16	Combustible Instantáneo	L/hr

Fuente: Elaboración propia.

3.6. Procesamiento de datos

La base de datos conformada (Tabla 6) contiene información sobre el comportamiento del motor durante las pruebas dinamométricas, abarcando 16 parámetros. Se revisa que los datos sean numéricos y estén completos. Posteriormente, se realiza la estandarización de estos y se calcula la correlación para cada uno de los parámetros de salida. Para ello, se utiliza el software de programación Python en su versión 3.8, junto con las librerías adecuadas como Pandas, Numpy, Matplotlib y, para el entrenamiento de los algoritmos de machine learning, la librería Scikit-Learn, como se muestra en la Figura 16

Figura 16*Importación de las librerías para el procesamiento de datos.*

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

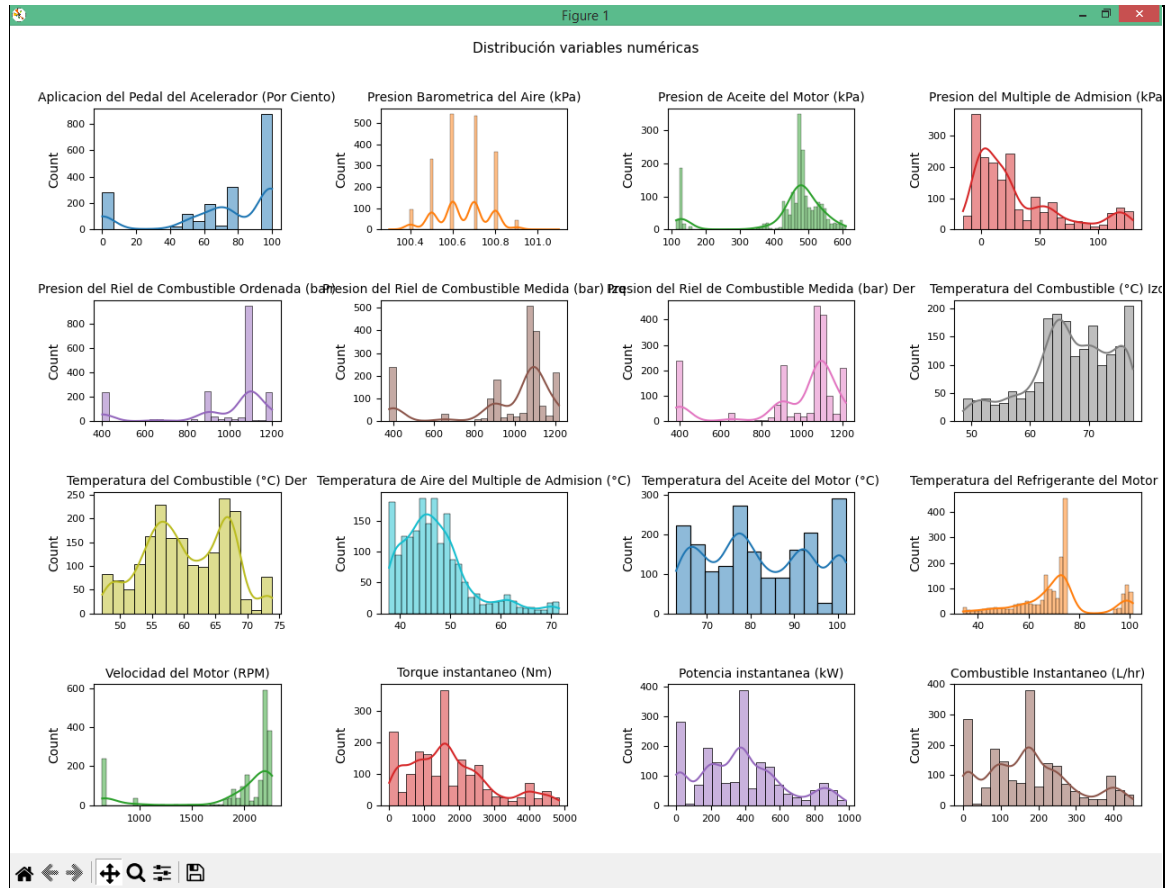
Fuente: Elaboración propia.

3.6.1. Evaluación de los datos registrados

La distribución de los datos tomados durante la prueba dinamométrica (Tabla 6) se observa en los histogramas de la Figura 17.

Figura 17

Distribución de los datos obtenidos.



Fuente: Elaboración propia.

3.6.2. Normalización

Para optimizar la eficacia de varios algoritmos de aprendizaje automático, es fundamental que todas las características y el valor objetivo numérico se encuentren dentro de un rango similar. Se redujo la escala de los valores de entrada mediante estandarización, también conocida como normalización z-score, la cual se ajusta a la siguiente formulación:

$$x_i = \frac{x_i - \mu(X)}{\sigma(X)} \quad (13)$$

3.6.3. Análisis correlacional

Con el fin de determinar la relevancia de los parámetros de entrada leídos por el controlador, se realizó un análisis correlacional para calcular el coeficiente de Spearman entre las entradas y cada una de las tres salidas. La evaluación se presenta en la Tabla 7.

Tabla 7

Coeficiente de Spearman.

Rango	Criterio
0,7 a 1,00	Correlación positiva fuerte
0,5 a 0,7	Correlación positiva moderada
0,2 a 0,5	Correlación positiva baja
-0,2 a 0,2	Correlación (positiva o negativa) débil o nula
-0,2 a -0,5	Correlación negativa moderada
-0,5 a -0,7	Correlación negativa moderada

Fuente: Elaboración propia.

En este sentido, se llevó a cabo un análisis correlacional de los datos según las instrucciones mostradas en la Figura 18. En dicha figura se observa la importación de las librerías adecuadas para el análisis, como Pandas, NumPy y Matplotlib, así como las instrucciones para el desarrollo del mapa de calor (Figura 19).

Figura 18

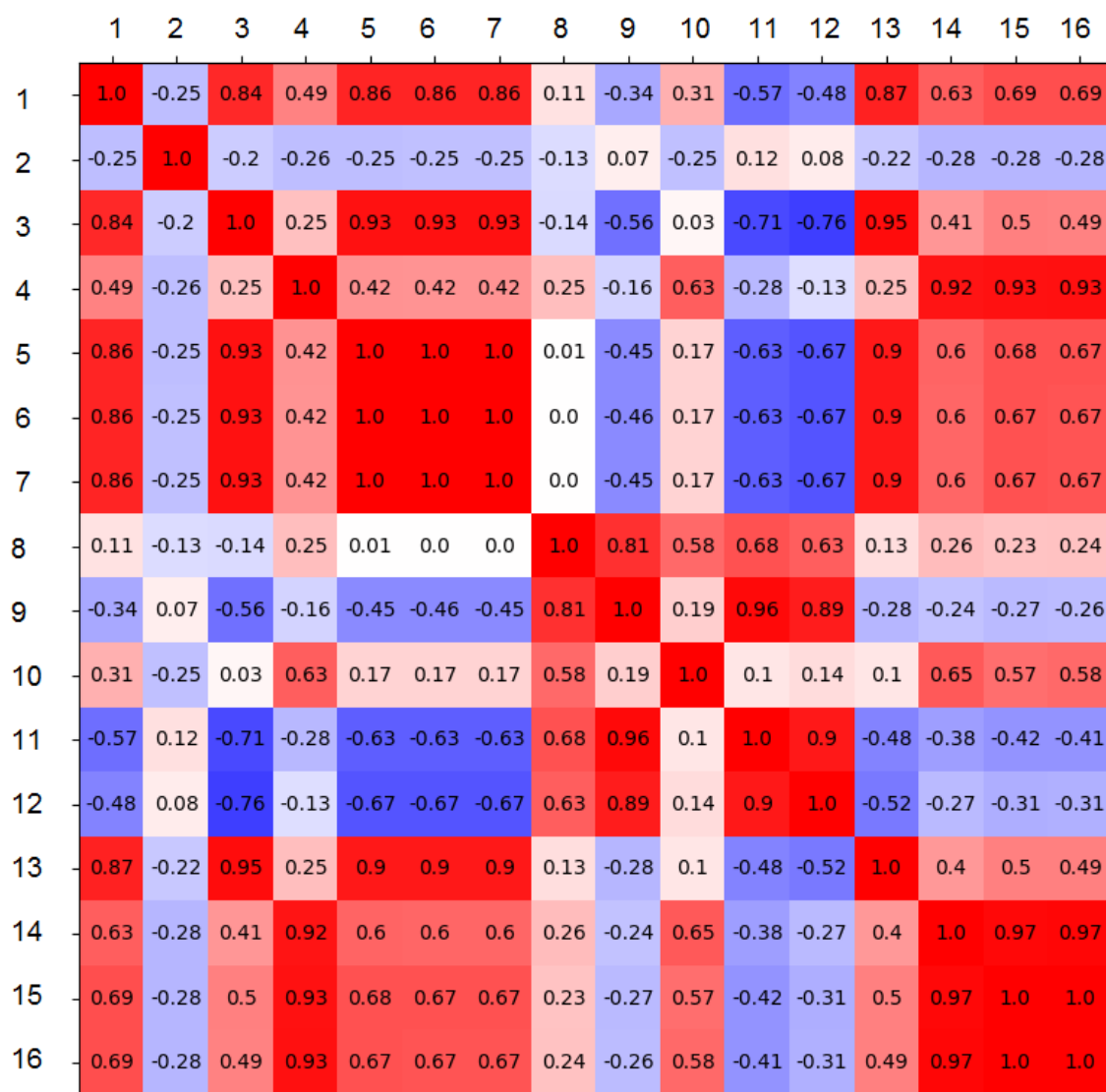
Código para el desarrollo del mapa de calor de correlaciones.

```
#Análisis de correlación
print(" ")
print("Matriz de correlacion")
corr_matriz = datos.corr()
print(corr_matriz)
plt.matshow(corr_matriz,cmap="bwr",vmin=-1, vmax=1 )
plt.xticks(range(18),datos.columns,rotation=90)
plt.yticks(range(18),datos.columns)
for i in range(len(corr_matriz.columns)):
    for j in range(len(corr_matriz.columns)):
        plt.text(i,j,round(corr_matriz.iloc[i,j],2),ha="center",
                va="center")
plt.colorbar()
plt.show()
```

Fuente: Elaboración propia.

Figura 19

Mapa de calor de la matriz de correlación de los ítem de la Tabla 6



Fuente: Elaboración propia.

Figura 20

Código fuente para el análisis de las correlaciones.

```
#Análisis de correlación
print(" ")
print("listado de correlacion para el Torque")
listado=datos.corr("spearman") ["Torque instantaneo (Nm)"]
listado=listado.sort_values(ascending=False)
print(listado)

print(" ")
print("listado de correlacion para la Potencia")
listado=datos.corr("spearman") ["Potencia instantanea (kW)"]
listado=listado.sort_values(ascending=False)
print(listado)

print(" ")
print("listado de correlacion para elConsumo de combustinle")
listado=datos.corr("spearman") ["Combustible Instantaneo (L/hr)"]
listado=listado.sort_values(ascending=False)
print(listado)
```

Fuente: Elaboración propia.

Tabla 8

Correlación de Spearman para la dimensión “torque instantáneo”.

Correlación de parámetros de entrada para el cálculo del torque	
Presión del Múltiple de Admisión (kPa)	0,86052241
Presión del Riel de Combustible Ordenada (bar)	0,64978312
Aplicación del Pedal del Acelerador (%)	0,60764519
Presión del Riel de Combustible Medida (bar) Der	0,57356629
Presión del Riel de Combustible Medida (bar) Izq	0,57344536
Temperatura de Aire del Múltiple de Admisión (°C)	0,49295114
Sensor de Temperatura del Combustible (°C) Izq	0,24061019
Presión de Aceite del Motor (kPa)	0,20552868
Velocidad del Motor (rpm)	-0,06675378
Temperatura del Refrigerante del Motor (°C)	-0,26405147
Presión Barométrica del Aire (kPa)	-0,27041841
Sensor de Temperatura del Combustible (°C) Der	-0,27602853
Temperatura del Aceite del Motor (°C)	-0,38169858

Fuente: Elaboración propia.

Asimismo, en la Figura 20 se presentan las instrucciones para calcular las correlaciones de los parámetros de la base de datos en relación con las variables de potencia, torque y consumo instantáneo de combustible. Con los resultados obtenidos mediante este código, se construyeron la Tabla 8, Tabla 9 y Tabla 10, que muestran la correlación de los parámetros

de entrada para el cálculo del torque, la potencia, y el consumo instantáneo de combustible, respectivamente.

Tabla 9

Correlación de Spearman para la dimensión “potencia instantánea”.

Correlación de parámetros de entrada para el cálculo de la potencia	
Presión del Múltiple de Admisión (kPa)	0,91204206
Presión del Riel de Combustible Ordenada (bar)	0,6761268
Aplicación del Pedal del Acelerador (%)	0,63280018
Presión del Riel de Combustible Medida (bar) Der	0,60078135
Presión del Riel de Combustible Medida (bar) Izq	0,60043003
Temperatura de Aire del Múltiple de Admisión (°C)	0,45569834
Presión de Aceite del Motor (kPa)	0,25854461
Sensor de Temperatura del Combustible (°C) Izq	0,20395494
Velocidad del Motor (rpm)	-0,02867599
Presión Barométrica del Aire (kPa)	-0,26408779
Temperatura del Refrigerante del Motor (°C)	-0,28895135
Sensor de Temperatura del Combustible (°C) Der	-0,28962989
Temperatura del Aceite del Motor (°C)	-0,40173263

Fuente: Elaboración propia.

Tabla 10

Correlación para la dimensión “consumo instantáneo de combustible”.

Correlación de parámetros de entrada para el cálculo del consumo	
Presión del Múltiple de Admisión (kPa)	0,90764256
Presión del Riel de Combustible Ordenada (bar)	0,67824214
Aplicación del Pedal del Acelerador (%)	0,63510687
Presión del Riel de Combustible Medida (bar) Der	0,60177419
Presión del Riel de Combustible Medida (bar) Izq	0,60134151
Temperatura de Aire del Múltiple de Admisión (°C)	0,46372836
Presión de Aceite del Motor (kPa)	0,25616438
Sensor de Temperatura del Combustible (°C) Izq	0,21092699
Velocidad del Motor (rpm)	-0,02913477
Presión Barométrica del Aire (kPa)	-0,26675232
Temperatura del Refrigerante del Motor (°C)	-0,28383410
Sensor de Temperatura del Combustible (°C) Der	-0,28578525
Temperatura del Aceite del Motor (°C)	-0,39821487

Fuente: Elaboración propia.

3.6.4. Establecimiento del parámetro de entrada al modelo

Los parámetros de entrada que se considera en desarrollo del modelo son los evaluados con una correlación fuerte para cada uno de los parámetros considerados como salidas del modelo. Las unidades correspondientes a dichos parámetros se encuentran estipulados en la Tabla 11.

Tabla 11

Características de los parámetros de entrada.

Parámetro	Unidad	Instrumento
Aplicación del Pedal del Acelerador	%	Software Insite
Presión Barométrica del Aire	kPa	Software Insite
Presión de Aceite del Motor	kPa	Software Insite
Presión del Múltiple de Admisión	Bar	Software Insite
Presión del Riel de Combustible Ordenada	Bar	Software Insite
Presión del Riel de Combustible Medida Izquierda	Bar	Software Insite
Presión del Riel de Combustible Medida Derecha	°C	Software Insite
Temperatura del Combustible Izquierda	°C	Software Insite
Temperatura del Combustible Derecha	°C	Software Insite
Temperatura de Aire del Múltiple de Admisión	°C	Software Insite
Temperatura del Aceite del Motor	°C	Software Insite
Temperatura del Refrigerante del Motor	°C	Software Insite
Revoluciones del motor	rpm	Software Insite

Fuente: Elaboración propia.

3.6.5. Establecimiento de los parámetros de salida del modelo

Los parámetros de salida considerados en el desarrollo del modelo son aquellos que se pueden obtener en un banco dinamométrico, como se muestra en la Tabla 12.

Tabla 12

Características de los parámetros de salida.

Parámetro de salida	Unidad	Instrumento
Torque	N.m	Banco de pruebas
Potencia al freno	W	Banco de pruebas
Consumo instantáneo de combustible	L/h	Software Insite

Fuente: Elaboración propia.

3.7. Desarrollo de los modelos de predicción

Una vez determinada la base de datos adecuada para el entrenamiento de los modelos de *Machine learning*, se procede a ejecutar el procedimiento establecido para la creación de cualquier modelo predictivo de *Machine learning*, el cual se divide en tres etapas esenciales: entrenamiento, aprendizaje y predicción. Para esta investigación, el análisis mediante el lenguaje de programación Python requiere el uso de la librería *Sklearn*. Posteriormente, se carga la base de datos ya estandarizada para formar las matrices correspondientes a los parámetros de entrada y salida, los que son mostrados en la Figura 21.

Figura 21

Establecimiento de los parámetros y división de conjuntos.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Gc1=datos["Regimen de Combustible Instantaneo (L/hr) Izq"].values
Gc2=datos["Regimen de Combustible Instantaneo (L/hr) Der"].values
Tcomb1=datos["Sensor de Temperatura del Combustible (C) Izq"].values
Tcomb2=datos["Sensor de Temperatura del Combustible (C) Der"].values
T_aceite=datos["Temperatura del Aceite del Motor (C)"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values

Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values

X=np.array([ Acel,P_adm, P_bar, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, rpm, Tcomb1, Tcomb2, T_refrig, P_aceite ]).T

y=np.array (Par)
z=np.array (Pot)
w=np.array (Consumo)

#Dividiendo los conjuntos de entrenemaiiento y prueba
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test =train_test_split(X,y,z,w, test_size=0.2)
```

Fuente: Elaboración propia.

Para la ejecución de esta investigación, se crearon modelos de predicción basados en los siguientes algoritmos: Regresión Lineal Múltiple (MLR), K-vecinos más cercanos (KNN) regresión, Máquina de Vectores Soporte (SVM) regresión, Árbol de Decisión (DT) regresión, Bosques Aleatorios (RF), Redes Neuronales Artificiales (RNA) y XGBoost.

3.7.1. Entrenamiento de la Regresión Lineal Múltiple (MLR)

Para este algoritmo también llamado MLR por sus siglas del inglés del “Multiple Linear Regression”, se empleó el método “fit” después de realizar la división de los datos y establecer el correspondiente “random_state”. De este modo, en el conjunto de datos de este método se introdujeron las variables independientes, que en este caso fueron “X_train” (Parámetros de entrada) y la variable dependiente que fue “y_train”, “z_train”, “w_train”, como se observa a continuación, en la Figura 22.

Figura 22

Entrenamiento de la regresión lineal múltiple.

```
#Definiendo el algoritmo a utilizar
from sklearn import linear_model
regresion_y=linear_model.LinearRegression()
regresion_z=linear_model.LinearRegression()
regresion_w=linear_model.LinearRegression()

#Entrenado el modelo
regresion_y.fit(X_train,y_train)
regresion_z.fit(X_train,z_train)
regresion_w.fit(X_train,w_train)

#prediccion valores
y_pred=regresion_y.predict(X_test)
z_pred=regresion_z.predict(X_test)
w_pred=regresion_w.predict(X_test)

# Calcular el error cuadrático medio
from sklearn.metrics import mean_squared_error
mse = mean_squared_error(y_test, y_pred)
print("Error cuadrático medio y:", mse)
mse = mean_squared_error(z_test, z_pred)
print("Error cuadrático medio z:", mse)
mse = mean_squared_error(w_test, w_pred)
print("Error cuadrático medio w:", mse)
print(" ")
R=regresion_y.score(X_train, y_train)
print("Coeficiente de determinación o precision y:", R)
R=regresion_z.score(X_train, z_train)
print("Coeficiente de determinación o precision z:", R)
R=regresion_w.score(X_train, w_train)
print("Coeficiente de determinación o precision w:", R)
```

Fuente: Elaboración propia.

Además, en la Figura 22 ilustra el proceso de entrenamiento de un modelo de regresión lineal, utilizando la biblioteca “sklearn.linear_model” de Python. Se integró la librería “LinearRegression”, y mediante el método “fit”, se llevaron a cabo las correlaciones necesarias. Las métricas de Evaluación, MAE, MSE, RMSE, R2 indican que el entrenamiento se realizó correctamente

3.7.1. Entrenamiento de K-Vecinos más cercanos (KNN) en regresión

Para este algoritmo también llamado KNN por sus siglas del inglés del “K-near neighbor”, se utilizó el método “fit” posterior a la división de los datos de entrenamiento y de prueba. Posteriormente se ejecutó la predicción del modelo con el método predict, mostrados en la Figura 23. Para el entrenamiento se evalúa el número de vecinos más cercanos K=2 hasta K=7.

Figura 23

Entrenamiento de la regresión KNN.

```
#Dividiendo los conjuntos de entrenamiento y prueba
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test = train_test_split(X, y, z, w, test_size=0.2, random_state=42)

#Definiendo el algoritmo a utilizar
from sklearn.neighbors import KNeighborsRegressor
knn_y= KNeighborsRegressor(n_neighbors=2)
knn_z= KNeighborsRegressor(n_neighbors=2)
knn_w= KNeighborsRegressor(n_neighbors=2)

#Entrenado el modelo
knn_y.fit(X_train,y_train)
knn_z.fit(X_train,z_train)
knn_w.fit(X_train,w_train)

#prediccion valores
y_pred=knn_y.predict(X_test)
z_pred=knn_z.predict(X_test)
w_pred=knn_w.predict(X_test)
```

Fuente: Elaboración propia.

3.7.2. Entrenamiento de Máquina de vector soporte (SVM) en Regresión

Este modelo agrupa algoritmos de aprendizaje supervisado, específicamente aquellos relacionados con la regresión lineal. También se conoce como SVM, que proviene de sus siglas en inglés "Support Vector Machine". Los datos, distribuidos en conjuntos de entrenamiento y prueba, se encuentran en un hiperplano, donde las etiquetas de clasificación permiten que la máquina se entrene y desarrolle un modelo capaz de predecir nuevas clases de datos. Al igual que en los modelos anteriores, los datos se dividen en un 80% para entrenamiento y un 20% para prueba. Parte del código puede verse en la Figura 24.

Figura 24

Entrenamiento de la regresión SVM.

```
#Definiendo el algoritmo a utilizar
from sklearn import linear_model
svmr_y=linear_model.LinearRegression()
svmr_z=linear_model.LinearRegression()
svmr_w=linear_model.LinearRegression()

#Entrenado el modelo
svmr_y.fit(X_train,y_train)
svmr_z.fit(X_train,z_train)
svmr_w.fit(X_train,w_train)

#prediccion valores
y_pred=svmr_y.predict(X_test)
z_pred=svmr_z.predict(X_test)
w_pred=svmr_w.predict(X_test)
```

Fuente: Elaboración propia.

3.7.3. Entrenamiento de árboles de decisión (DT) en regresión

También conocido como DT, que proviene de las siglas en inglés "Decision Tree". Este algoritmo se seleccionó por su capacidad de aprendizaje supervisado y su naturaleza no paramétrica. Utilizando los datos registrados en la base de datos, se pudieron crear nodos raíz, ramas, hojas y nodos internos. Se emplearon códigos similares a los de otros modelos para el árbol de decisión, utilizando el 80% de los datos para el entrenamiento y el 20% para la validación. De manera similar, se usó el método "fit" y la librería "DecisionTreeRegressor" para realizar el entrenamiento, como se ilustra en la Figura 25..

Figura 25

Entrenamiento de la regresión árboles de decisión.

```
#Definiendo el algoritmo a utilizar
from sklearn.tree import DecisionTreeRegressor
adr_y=DecisionTreeRegressor (max_depth=5)
adr_z=DecisionTreeRegressor (max_depth=5)
adr_w=DecisionTreeRegressor (max_depth=5)

#Entrenado el modelo
adr_y.fit(X_train,y_train)
adr_z.fit(X_train,z_train)
adr_w.fit(X_train,w_train)

#prediccion valores
y_pred=adr_y.predict(X_test)
z_pred=adr_z.predict(X_test)
w_pred=adr_w.predict(X_test)
```

Fuente: Elaboración propia.

3.7.4. Entrenamiento de bosques aleatorios (RF) en regresión

En los modelos mencionados anteriormente, se definieron las variables independientes y dependientes, lo que permitió aplicar los códigos a diferentes modelos, como Random Forest, también conocido como RF por sus siglas en inglés "Random Forest". Al igual que en los casos anteriores, los datos se dividieron en un 80% para el entrenamiento y un 20% para la validación. Durante el proceso de entrenamiento de este modelo, la clase "RandomForestRegressor" permitió incorporar parámetros específicos para el bosque aleatorio. En este caso, se utilizaron 300 árboles, lo que dio lugar a predicciones favorables para este estudio, como se muestra a continuación.

En la Figura 26 se presenta la clase y el número de árboles empleados para desarrollar el bosque aleatorio. Este número de árboles ayudó a evitar el sobreentrenamiento y la sobreajuste de las predicciones. Se probaron entre 100 y 300 árboles, y se observó que usar menos del número óptimo causaba sobreajuste, lo que incrementaba el tiempo de ejecución del código. Con 100 a 150 árboles, el bosque ejecutó el código de manera más rápida. Al añadir el parámetro "criterion" con el valor "squared_error", el entrenamiento fue adecuado, ya que este criterio permitió modelar correctamente la relación entre las variables independientes y la variable dependiente para realizar predicciones futuras.

Figura 26

Entrenamiento de la regresión bosques aleatorios.

```
#Definiendo el algoritmo a utilizar
from sklearn.ensemble import RandomForestRegressor
bar_y = RandomForestRegressor(n_estimators = 300, max_depth = 8)
bar_z = RandomForestRegressor(n_estimators = 300, max_depth = 8)
bar_w = RandomForestRegressor(n_estimators = 300, max_depth = 8)

#Entrenado el modelo
bar_y.fit(X_train,y_train)
bar_z.fit(X_train,z_train)
bar_w.fit(X_train,w_train)

#prediccion valores
y_pred=bar_y.predict(X_test)
z_pred=bar_z.predict(X_test)
w_pred=bar_w.predict(X_test)
```

Fuente: Elaboración propia.

3.7.5. Entrenamiento de redes neuronales (ANN)

Este modelo también se denomina ANN, que corresponde a las siglas en inglés de "Artificial Neural Network". Para entrenar el modelo basado en redes neuronales artificiales, se dividen los datos en dos conjuntos, asignando el 80% para el entrenamiento y el 20% para la validación. Luego, se define el número de variables independientes y la variable dependiente del modelo, y se procede a estructurar las capas ocultas, especificando la cantidad de neuronas en cada capa, como se muestra en la Figura 27

La red neuronal empleó 10 neuronas en cada capa oculta, lo que resultó en modelos estables y con menor probabilidad de error. El segundo parámetro importante fue el activador ("activation"), configurado como "relu", y para el parámetro Solver, se utilizó la función "Adam" (ver Figura 27).

Figura 27

Declaración del modelo de redes neuronales.

```
#Definiendo el algoritmo a utilizar
from sklearn.neural_network import MLPRegressor
rna_y=MLPRegressor(solver='adam',activation='relu', alpha=1e-5, hidden_layer_sizes=(10,10,10), random_state=1,max_iter=200)
rna_z=MLPRegressor(solver='adam',activation='relu', alpha=1e-5, hidden_layer_sizes=(10,10,10), random_state=1,max_iter=200)
rna_w=MLPRegressor(solver='adam',activation='relu', alpha=1e-5, hidden_layer_sizes=(10,10,10), random_state=1,max_iter=200)
```

Fuente: Elaboración propia.

En la Figura 28 se presentan las instrucciones para la compilación de la red, utilizando los parámetros previamente mencionados para establecer una correlación entre los datos y la red misma. Una vez finalizada la compilación, se procedió con el entrenamiento. Este se realizó mediante el método "fit", cuyos parámetros incluyen los datos de entrenamiento de las variables independientes y dependientes "(X_train, y_train)".

Figura 28

Entrenamiento del modelo de redes neuronales.

```
#Entrenado el modelo
rna_y.fit(X_train,y_train)
rna_z.fit(X_train,z_train)
rna_w.fit(X_train,w_train)

#prediccion valores
y_pred=rna_y.predict(X_test)
z_pred=rna_z.predict(X_test)
w_pred=rna_w.predict(X_test)
```

Fuente: Elaboración propia.

3.7.6. Entrenamiento del modelo XGBoost

Este modelo incluye algoritmos de aprendizaje supervisado, específicamente aquellos asociados con la regresión lineal. Los datos, separados en conjuntos de entrenamiento y prueba, se ubican en un hiperplano donde las etiquetas de clasificación permiten que la máquina se entrene para crear un modelo capaz de predecir nuevas clases de datos. De manera similar a los modelos previos, los datos se dividen en un 80% para el entrenamiento y un 20% para la prueba..

Figura 29

Entrenamiento del modelo XGBoost.

```
#Definiendo el algoritmo a utilizar
from xgboost import XGBRegressor
import xgboost as xgb

model_y = XGBRegressor(booster="gbtree", max_depth= 2, eta= 0.3, tree_method="auto", nthread=2)
model_z = XGBRegressor(booster="gbtree", max_depth= 2, eta= 0.3, tree_method="auto", nthread=2)
model_w = XGBRegressor(booster="gbtree", max_depth= 2, eta= 0.3, tree_method="auto", nthread=2)

#Entrenado el modelo
model_y.fit(X_train,y_train)
model_z.fit(X_train,z_train)
model_w.fit(X_train,w_train)

#prediccion valores
y_pred=model_y.predict(X_test)
z_pred=model_z.predict(X_test)
w_pred=model_w.predict(X_test)
```

Fuente: Elaboración propia.

3.8. Determinación del factor de carga

Determinados los parámetros de salida por los métodos de predicción, se procede al cálculo del factor de carga para un ciclo de operación, según la fórmula:

$$FC = \frac{\sum_i^n P_i}{nP_n} \quad (25)$$

Donde:

- P_i es la potencia instancia registrada en cada registro
- n es la cantidad de registros obtenidos en el monitoreo
- P_n es la potencia nominal del motor

Capítulo IV. Análisis y discusión de resultados

4.1. Análisis descriptivo

4.1.1. Variable dependiente: Desempeño del motor

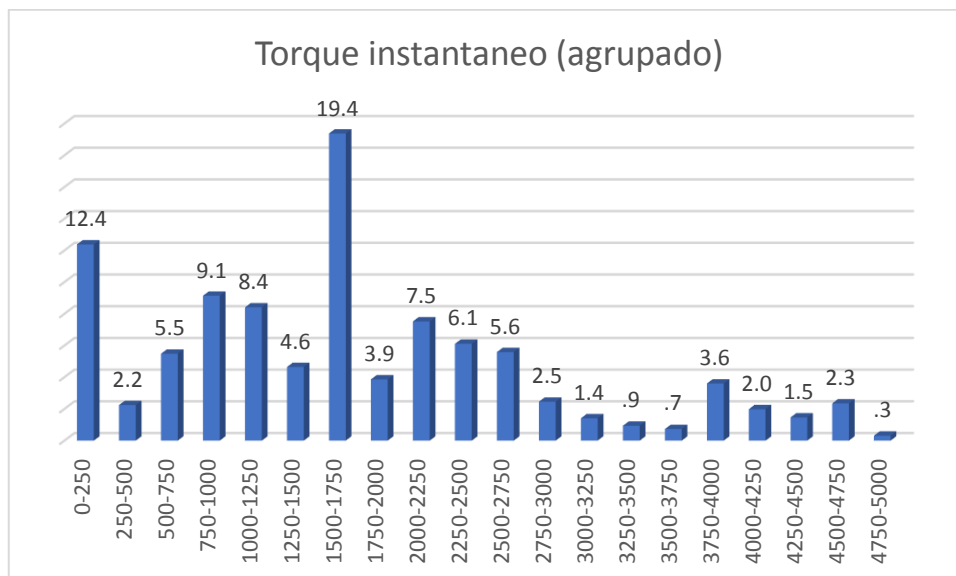
Mediante el software SPSS, se ejecuta análisis descriptivo de las Dimensiones de la Variable 2: Desempeño del motor de combustión interna, las cuales son: torque instantáneo, potencia instantánea, consumo instantáneo de combustible y factor de carga instantáneo. Los 1915 registros obtenidos se agruparon en intervalos para facilitar la interpretación de sus características. De igual manera, se hace el análisis descriptivo de las Dimensiones de la Variable Desempeño pronosticado del motor.

4.1.2. Dimensión 1: Torque instantáneo

Se procede con el análisis descriptivo de la dimensión torque instantáneo. En la Figura 30 se observa la distribución de los datos del torque instantáneo en intervalos de 250 N.m. La mayor cantidad de registros de torque instantáneo obtenidos durante la prueba dinamométrica se encuentra en el rango de 1500 N.m a 1750 N.m, lo que representa 371 eventos, equivalentes al 19,4% de los datos recolectados.

Figura 30

Distribución porcentual de la dimensión torque instantáneo.



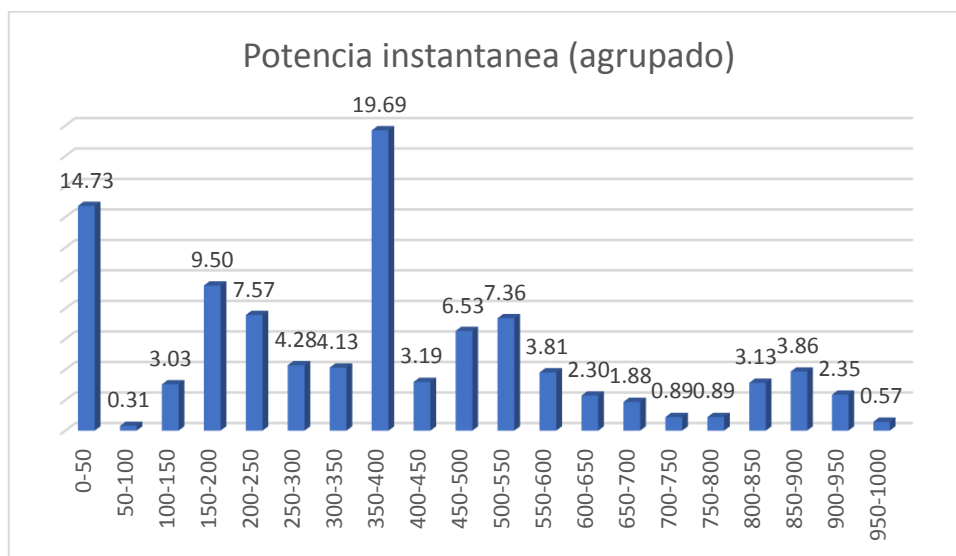
Fuente: Elaboración propia.

4.1.3. Dimensión 2: Potencia instantánea

Figura 31. se observa la distribución de los datos de la potencia instantánea en intervalos de 50 kW. La mayor cantidad de registros de potencia instantánea obtenidos durante la prueba dinamométrica se encuentra en el rango de 350 kW a 400 kW, lo que representa 377 eventos, equivalentes al 19.7% de los datos recolectados.

Figura 31

Distribución porcentual de la dimensión potencia instantánea.



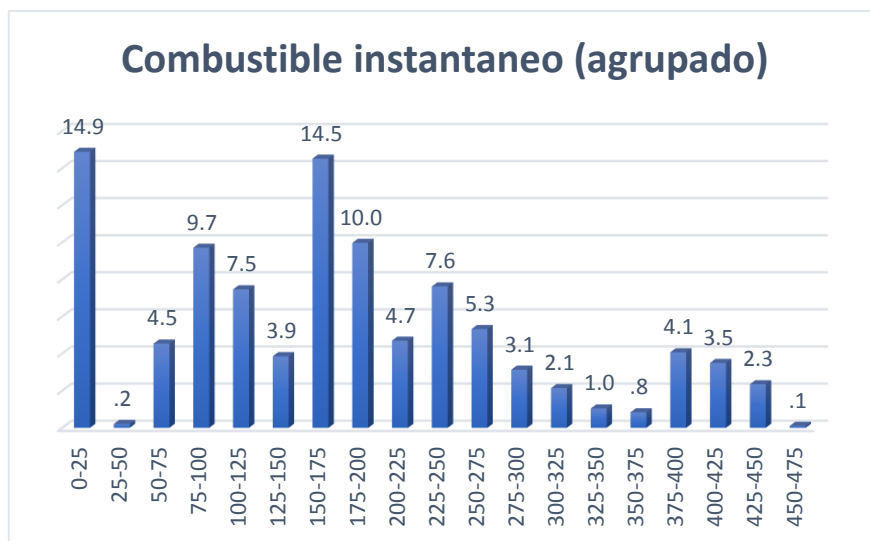
Fuente: Elaboración propia.

4.1.4. Dimensión 3: Consumo instantáneo de combustible

Figura 32. se observa la distribución de los datos del consumo instantáneo de combustible en rangos de intervalos de 25 L/h. La mayor cantidad de registros de consumo instantáneo de combustible obtenidos durante la prueba dinamométrica se encuentra en el rango de 150 L/h a 175 L/h, lo que representa 275 eventos, equivalentes al 14,5% de los datos recolectados.

Figura 32

Distribución porcentual de consumo instantáneo de combustible.



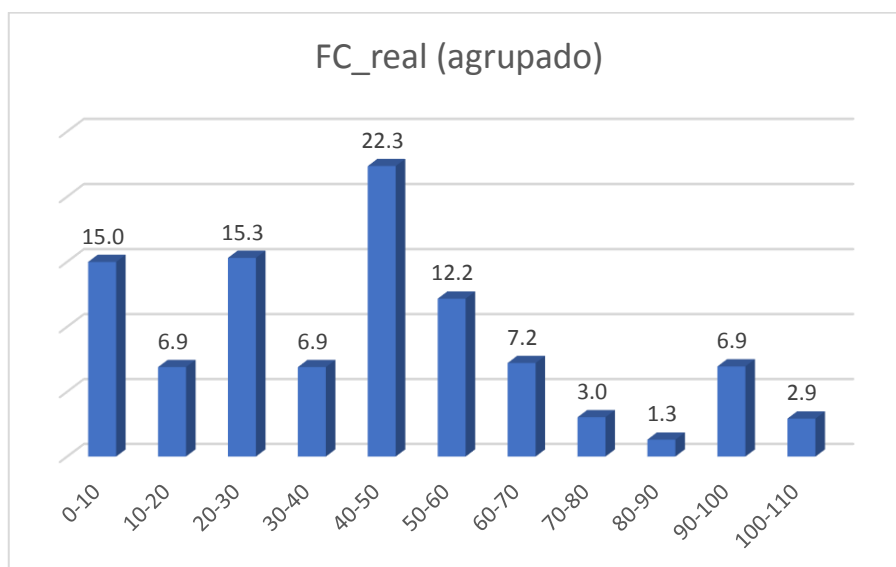
Fuente: Elaboración propia.

4.1.5. Dimensión 4: Factor de carga instantáneo

En la Figura 33 se observa la distribución de los datos del factor de carga instantáneo en intervalos de 10%. La mayor cantidad de registros de factor de carga instantáneo se encuentra en el rango de 40% a 50%, lo que representa 428 eventos, equivalentes al 22,3% de los datos recolectados.

Figura 33

Distribución porcentual de la dimensión factor de carga instantáneo.



Fuente: Elaboración propia.

4.1.6. Variable dependiente: Desempeño pronosticado del motor

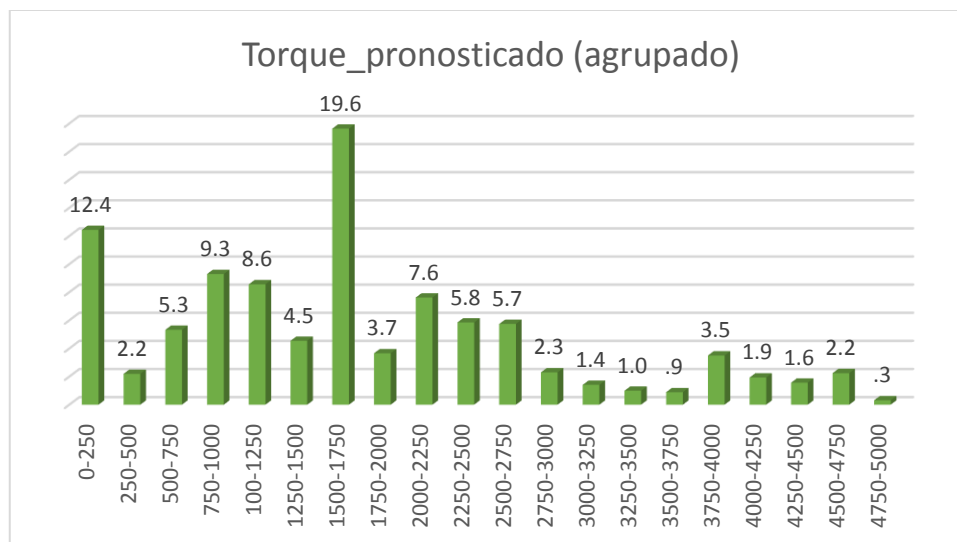
Se presenta un análisis descriptivo de las dimensiones de la variable de desempeño pronosticado del motor de combustión interna, obtenidas mediante la aplicación del modelo de *machine learning* determinado. Las dimensiones de dicha variable son: torque pronosticado, potencia pronosticada, consumo de combustible pronosticado y factor de carga pronosticado. Los 1915 registros obtenidos se agruparon en intervalos para facilitar la lectura de sus características.

4.1.7. Dimensión 1: Torque pronosticado

En la Figura 34 se observa la distribución de los datos del torque pronosticado en rangos de intervalos de 250 Nm. La mayor cantidad de registros de torque pronosticado obtenidos durante la aplicación del modelo de *machine learning* se encuentra en el rango de 1500 N.m a 1750 N.m, lo que representa 376 eventos, equivalentes al 19,6% de los datos tomados.

Figura 34

Distribución porcentual de la dimensión torque pronosticado.



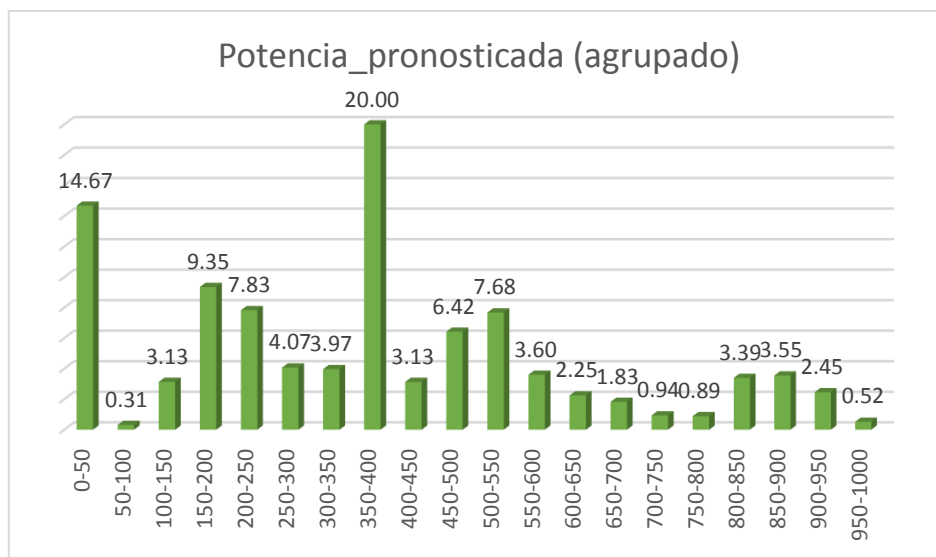
Fuente: Elaboración propia.

4.1.8. Dimensión 2: Potencia pronosticada

En la Figura 35 se observa la distribución de los datos de la potencia pronosticada en rangos de intervalos de 50 kW, donde la mayor cantidad de registros de potencia pronosticada obtenidos durante la aplicación del modelo de *machine learning* se encuentra en el rango de 300 kW a 350 kW, lo que representa 383 eventos, equivalentes al 20% de los datos tomados.

Figura 35

Distribución porcentual de la dimensión potencia pronosticada.



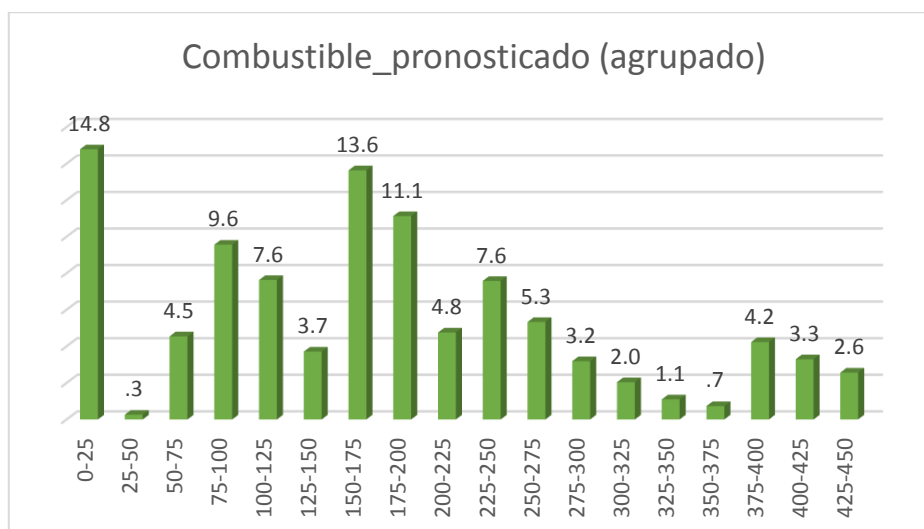
Fuente: Elaboración propia.

4.1.9. Dimensión 3: Consumo de combustible pronosticado

En la Figura 36 se observa la distribución de los datos del consumo de combustible pronosticado en rangos de intervalos de 25 L/h. La mayor cantidad de registros de consumo pronosticado obtenidos durante la aplicación del modelo de *machine learning* se encuentra en el rango de 150 L/h a 175 L/h, lo que representa 213 eventos, equivalentes al 11,1% de los datos recolectados.

Figura 36

Distribución porcentual de la dimensión combustible pronosticado.

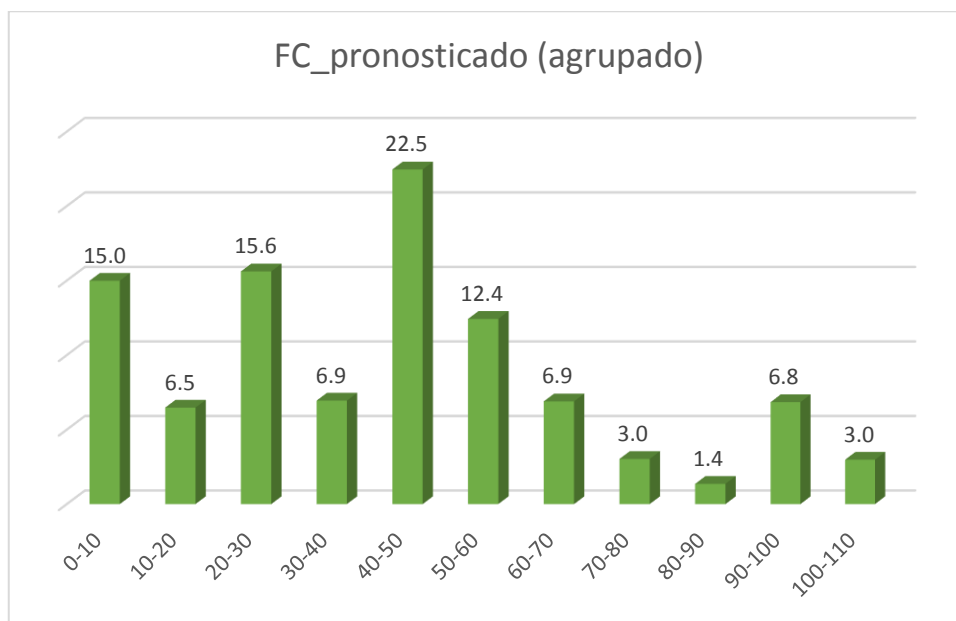


4.1.10. Dimensión 4: Factor de carga pronosticado

En la Figura 37, se observa la distribución de los datos del factor de carga pronosticado en rangos de intervalos de 10%. La mayor cantidad de registros del factor de carga pronosticado obtenidos durante la aplicación del modelo de *machine learning* se encuentra en el rango de 40% a 50%, lo que representa 430 eventos, equivalentes al 22,5% de los datos recolectados.

Figura 37

Distribución porcentual del factor de carga (FC) pronosticado.



Fuente: Elaboración propia.

4.2. Evaluación de modelos

En esta sección se presentan los resultados de las predicciones realizadas por los modelos analizados sobre los parámetros de salida, junto con las métricas de evaluación de los modelos de regresión. Estas métricas incluyen el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2) para cada modelo. Además, se evalúa el Score tanto para los conjuntos de entrenamiento como para los de prueba, que se ejecutaron en cada uno de los 10 conjuntos generados mediante la técnica de validación cruzada k-Folds.

4.2.1. Pronóstico del modelo de regresión lineal múltiple (MLR)

Una vez completado el entrenamiento del modelo de regresión lineal múltiple, se verifica la concordancia entre los parámetros de salida pronosticados y los parámetros de salida instantáneos. Esto se lleva a cabo mediante la evaluación del error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2), como se muestra en la Tabla 13, para cada uno de los dobleces generados por el método de validación cruzada K-folds..

Tabla 13

Parámetros de salida mediante MLR con validación cruzada k-folds

K-Folds	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,12843	0,03114	0,17646	0,96664	0,95869	0,96664
	Potencia	0,10418	0,02395	0,15477	0,97459	0,97179	0,97459
	Consumo	0,10811	0,02357	0,15353	0,97528	0,97224	0,97528
2	Torque	0,16248	0,04517	0,21254	0,96062	0,95919	0,96062
	Potencia	0,12459	0,03848	0,19616	0,96496	0,97287	0,96496
	Consumo	0,12173	0,03192	0,17865	0,97103	0,97262	0,97103
3	Torque	0,13633	0,04044	0,20109	0,96226	0,95904	0,96226
	Potencia	0,11304	0,02694	0,16412	0,97320	0,97192	0,97320
	Consumo	0,11450	0,02654	0,16291	0,97360	0,97240	0,97360
4	Torque	0,15712	0,05786	0,24055	0,93942	0,96138	0,93942
	Potencia	0,12308	0,03852	0,19626	0,96188	0,97318	0,96188
	Consumo	0,12165	0,03732	0,19319	0,96307	0,97353	0,96307
5	Torque	0,14293	0,04511	0,21238	0,95055	0,96032	0,95055
	Potencia	0,11039	0,02588	0,16086	0,97207	0,97204	0,97207
	Consumo	0,11136	0,02858	0,16905	0,96892	0,97288	0,96892
6	Torque	0,14516	0,04315	0,20771	0,95034	0,96030	0,95034
	Potencia	0,11684	0,03219	0,17943	0,96501	0,97270	0,96501
	Consumo	0,12168	0,03538	0,18810	0,96075	0,97364	0,96075
7	Torque	0,13856	0,03694	0,19221	0,96267	0,95910	0,96267
	Potencia	0,11077	0,02312	0,15205	0,97651	0,97156	0,97651
	Consumo	0,11050	0,02316	0,15217	0,97641	0,97210	0,97641
8	Torque	0,14888	0,04667	0,21604	0,95121	0,96031	0,95121
	Potencia	0,11430	0,02515	0,15859	0,97349	0,97189	0,97349
	Consumo	0,11110	0,02440	0,15621	0,97424	0,97233	0,97424
9	Torque	0,13378	0,03364	0,18340	0,96995	0,95802	0,96995
	Potencia	0,11477	0,02785	0,16687	0,97487	0,97165	0,97487
	Consumo	0,11497	0,02577	0,16053	0,97690	0,97192	0,97690
10	Torque	0,13964	0,03979	0,19947	0,96150	0,95911	0,96150
	Potencia	0,11061	0,02319	0,15229	0,97774	0,97135	0,97774
	Consumo	0,11740	0,02554	0,15980	0,97568	0,97209	0,97568

Con los datos generados con los 10 dobleces evaluados, los valores promedios obtenidos se observan en la tabla 14

Tabla 14

Promedio de métricas de evaluación dados por el modelo MLR.

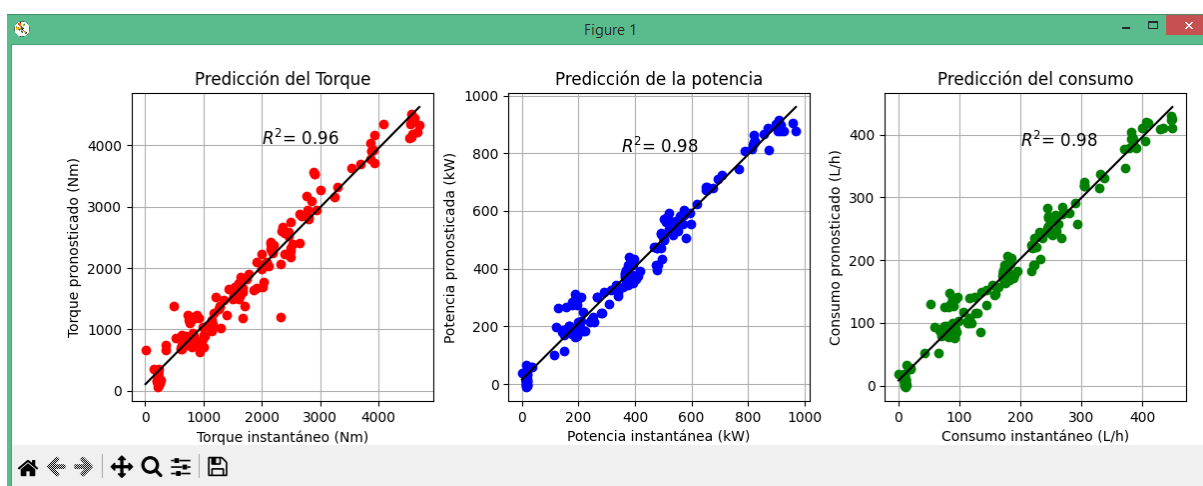
Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
Torque	0,1433305	0,0419907	0,204186	0,957515	0,9595464	0,957515
Potencia	0,1142572	0,028526	0,1681388	0,9714302	0,972095	0,9714302
Consumo	0,1152973	0,0282173	0,1674142	0,9715885	0,9725753	0,9715885

Fuente: Elaboración propia.

La representación gráfica de los resultados de pronóstico se observa en la figura 38, donde el torque instantáneo vs el torque pronosticado en graficado con un $R^2 = 0,96$, Asimismo, el comportamiento de la potencia instantáneo vs potencia pronosticada, con un $R^2 = 0,98$. Finalmente, se observar el consumo de combustible instantáneo vs el consumo de combustible pronosticado, con un $R^2 = 0,98$.

Figura 38

Salidas instantáneas vs salidas pronosticadas mediante modelo MLR



Fuente: Elaboración propia.

Con el modelo ya entrenado, éste entrega los coeficientes para cada uno de los parámetros de salida en función de los parámetros de entrada (ver la Tabla 15).

Tabla 15*Coeficientes para las ecuaciones de torque, potencia, y consumo*

Salida	Var	Torque	Potencia	Consumo
Presión del múltiple de admisión	X1	17,97	4,56	2,09
Presión de riel comandada	X2	2,4	0,63	0,32
Posición el pedal de acelerador	X3	9,32	1,44	0,69
Presión de riel sensada (Der)	X4	-0,48	-0,21	-0,09
Presión de riel sensada (Izq)	X5	0,37	-0,01	-0,03
Temperatura del múltiple de admisión	X6	-13,18	-3,68	-1,55
Presión de aceite de motor	X7	6,09	-0,62	-0,31
Velocidad de motor	X8	-2,56	-0,03	-0,02
Presión Barométrica	X9	124,29	11,51	8,73
Temperatura del refrigerante	X10	5,85	-2,14	-1
Temperatura de combustible (Der)	X11	-50,3	-7,35	-3,69
Temperatura de combustible (Izq)	X12	88,89	10,24	4,98
Bias	-	-14706,77	-1009,43	-808,62

Fuente: Elaboración propia.

De tal forma que, para las pruebas dinamométricas realizadas, las ecuaciones que describen cada uno de parámetros de salida - torque, potencia y consumo de combustible- en función de los parámetros de entrada, serán:

$$\begin{aligned} \text{Torque} = & 17.97x_1 + 2.40x_2 + 9.32x_3 - 0.48x_4 + 0.37x_5 - 13.18x_6 \\ & + 6.09x_7 - 2.56x_8 + 124.29x_9 + 5.85x_{10} - 50.30x_{11} \\ & + 88.89x_{12} - 14706.77 \end{aligned} \quad (26)$$

$$\begin{aligned} \text{Potencia} = & 4.56 + 0.63x_2 + 1.44x_3 - 0.21x_4 + 0.37x_5 - 0.01x_6 - 0.62x_7 \\ & - 0.003x_8 + 11.51x_9 - 2.14x_{10} - 7.35x_{11} - 10.24x_{12} \\ & - 1009.43 \end{aligned} \quad (27)$$

$$\begin{aligned} \text{Consumo de combustible} \\ = & 2.09x_1 + 0.32x_2 + 0.69x_3 - 0.09x_4 - 0.03x_5 - 1.55x_6 \\ & - 0.31x_7 - 0.02x_8 + 18.73x_9 - x_{10} - 3.69x_{11} + 4.98x_{12} \\ & - 808.62 \end{aligned} \quad (28)$$

4.2.2. Análisis de significancia de los coeficientes de regresión

Para el análisis de significancia de los coeficientes de regresión, se ejecuta análisis de cuadrados mínimos ordinarios, mostrados en la Figura 39, Sabiendo que los parámetros de entrada son los correspondientes a la Tabla 11.

Figura 39

Codificación para el análisis de cuadrado mínimos ordinarios

```
import statsmodels.api as sm
X_constant=sm.add_constant(X_train)
lin_reg_y=sm.OLS(y_train,X_constant).fit()
lin_reg_z=sm.OLS(w_train,X_constant).fit()
lin_reg_w=sm.OLS(z_train,X_constant).fit()

print("***** ")
print("\nResumen del modelo de predicion del torque")
print(lin_reg_w.summary())

print("\nResumen del modelo de predicion de la potencia")
print(lin_reg_z.summary())

print("\nResumen del modelo de predicion del consumo")
print(lin_reg_w.summary())
print(" ")
```

Fuente: Elaboración propia

Respecto al análisis del modelo de predicción del torque los resultados se observa en la Figura 40, donde se observa Prob (F-stactistic)=0,00, como $p_valor < 0.05$, entonces el modelo es significativo. Los coeficientes que no son significativas ($p > 0.05$) en el modelo son x_6 , X_{11} , y X_{13} , que corresponden a la presión de riel sensada 1, la temperatura del aceite, y los rpm del motor.

Respecto al análisis del modelo de predicción de la potencia los resultados se observa en la Figura 41, donde se observa Prob (F-stactistic)=0,00, como $p_valor < 0.05$, entonces el modelo es significativo. Los coeficientes que no son significativas ($p > 0.05$) en el modelo son x_6 , X_{11} , y X_{13} , que corresponden a la presión de riel sensada 1, la temperatura del aceite, y los rpm del motor.

Respecto al análisis del modelo de predicción del consumo los resultados se observa en la Figura 42, donde se observa Prob (F-stactistic)=0,00, como $p_valor < 0.05$, entonces el modelo es significativo. Los coeficientes que no son significativas ($p > 0.05$) en el modelo son x_6 , X_{11} , y X_{13} , que corresponden a la presión de riel sensada 1, la temperatura del aceite, y el rpm del motor.

Figura 40

Significancia de los coeficientes del torque

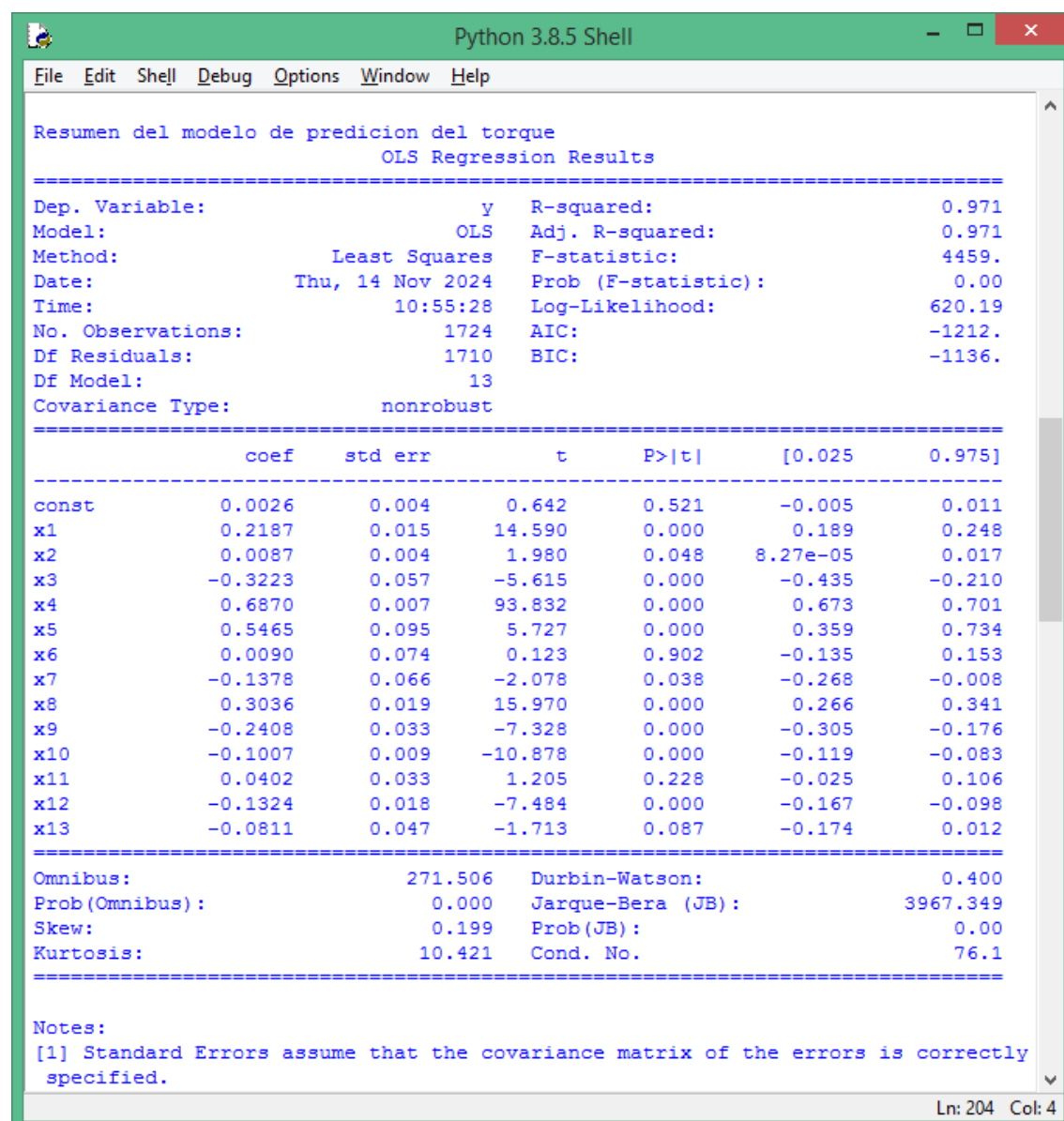


Figura 41

Significancia de los coeficientes de la potencia

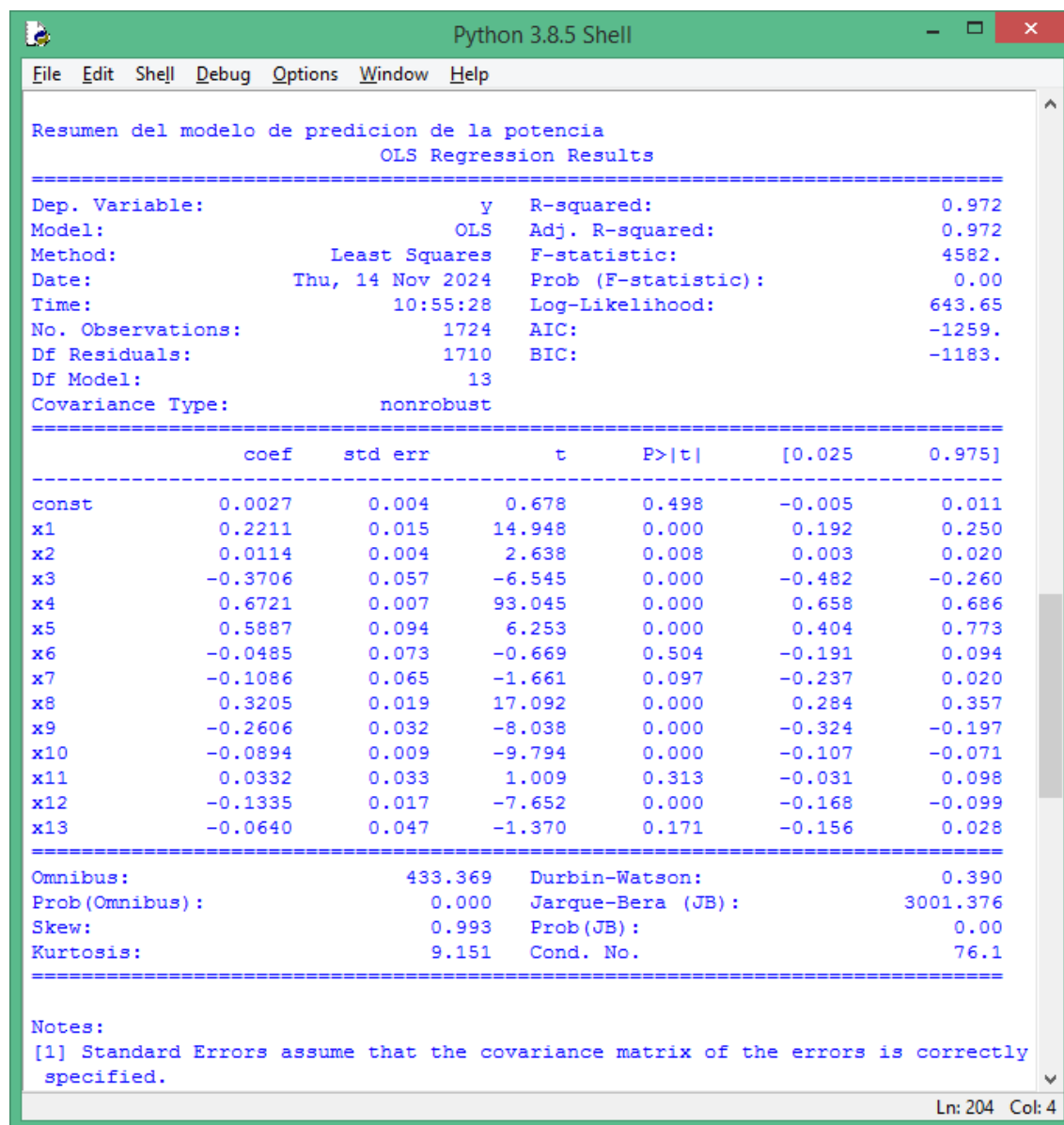


Figura 42

Significancia de los coeficientes del consumo

```
Python 3.8.5 Shell
File Edit Shell Debug Options Window Help

Resumen del modelo de predicion del consumo
      OLS Regression Results
=====
Dep. Variable:          y      R-squared:          0.971
Model:                OLS      Adj. R-squared:      0.971
Method:             Least Squares      F-statistic:      4459.
Date:               Thu, 14 Nov 2024      Prob (F-statistic):      0.00
Time:                10:55:28      Log-Likelihood:      620.19
No. Observations:      1724      AIC:              -1212.
Df Residuals:          1710      BIC:              -1136.
Df Model:              13
Covariance Type:       nonrobust
=====
               coef      std err          t      P>|t|      [0.025      0.975]
-----
const          0.0026      0.004      0.642      0.521      -0.005      0.011
x1             0.2187      0.015     14.590      0.000      0.189      0.248
x2             0.0087      0.004      1.980      0.048      8.27e-05      0.017
x3            -0.3223      0.057     -5.615      0.000     -0.435     -0.210
x4             0.6870      0.007     93.832      0.000      0.673      0.701
x5             0.5465      0.095      5.727      0.000      0.359      0.734
x6             0.0090      0.074      0.123      0.902     -0.135      0.153
x7            -0.1378      0.066     -2.078      0.038     -0.268     -0.008
x8             0.3036      0.019     15.970      0.000      0.266      0.341
x9            -0.2408      0.033     -7.328      0.000     -0.305     -0.176
x10            -0.1007      0.009    -10.878      0.000     -0.119     -0.083
x11             0.0402      0.033      1.205      0.228     -0.025      0.106
x12            -0.1324      0.018     -7.484      0.000     -0.167     -0.098
x13            -0.0811      0.047     -1.713      0.087     -0.174      0.012
=====
Omnibus:              271.506      Durbin-Watson:          0.400
Prob(Omnibus):         0.000      Jarque-Bera (JB):      3967.349
Skew:                  0.199      Prob(JB):              0.00
Kurtosis:              10.421      Cond. No.              76.1
=====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.

Ln: 204 Col: 4
```

4.2.3. Verificación de los Supuestos del modelo MLR

Los resultados alcanzados por la regresión lineal múltiple son analizados bajos la Verificación de los Supuestos del modelo, como La Independencia, Normalidad, linealidad, Homocedasticidad.

Para la Independencia: Las observaciones deben ser independientes, es decir no debe existir autocorrelación entre los residuos, esto se ejecuta por medio de coeficiente de Durbin Watson. Cuyo valor debe ser considerado entre 1,5 y 2,5 para ser considerado independientes

Tabla 16*Evaluación de independencia de los residuos*

Salida	Coef de Durbin-Watson
Torque	0,204791
Potencia	0,389722
Consumo	0,399728

Fuente: Elaboración propia.

Para la Normalidad: Los datos deben tener una distribución normal.

Se analizan las siguientes hipótesis

Ho: Los datos siguen una distribución normal.

Ha: Los datos no siguen una distribución normal

Considerando que:

Si $p_valor < 0.05$ entonces se rechaza la Ho,

Si $p_valor \geq 0.05$ entonces se acepta la Ho

Tabla 17*Prueba de normalidad. Evolución del p_valor.*

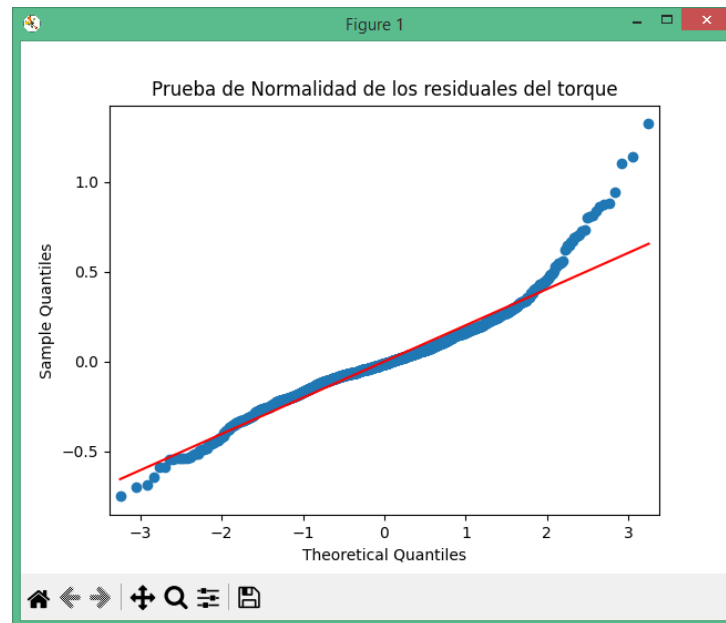
Salida	P_valor
Torque	1,78E-24
Potencia	9,05E-30
Consumo	5,43E-31

Fuente: Elaboración propia

Como el p_valor es menor a 0.05 se determina que los datos siguen una distribución no normal. Gráficamente se puede observar en la Figura 43, Figura 44 y Figura 45, donde los extremos de los valores residuales perdieron correspondencia.

Figura 43

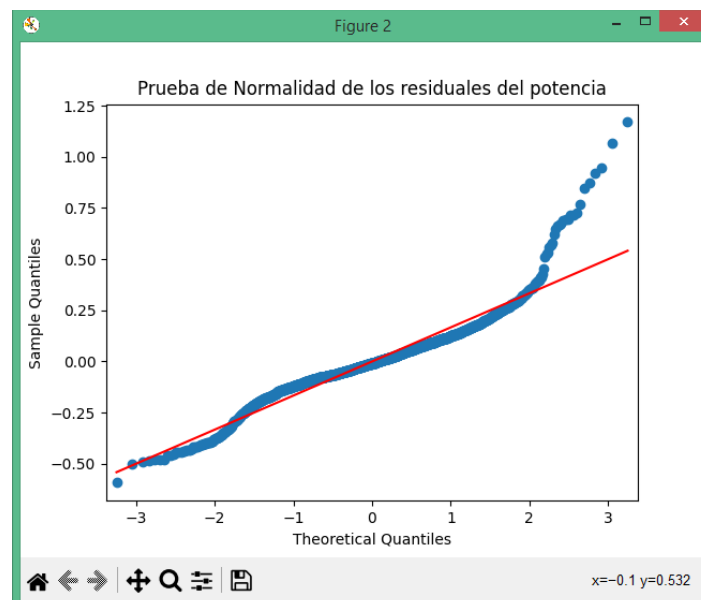
Prueba de normalidad de los residuos del torque



Fuente: Elaboración propia

Figura 44

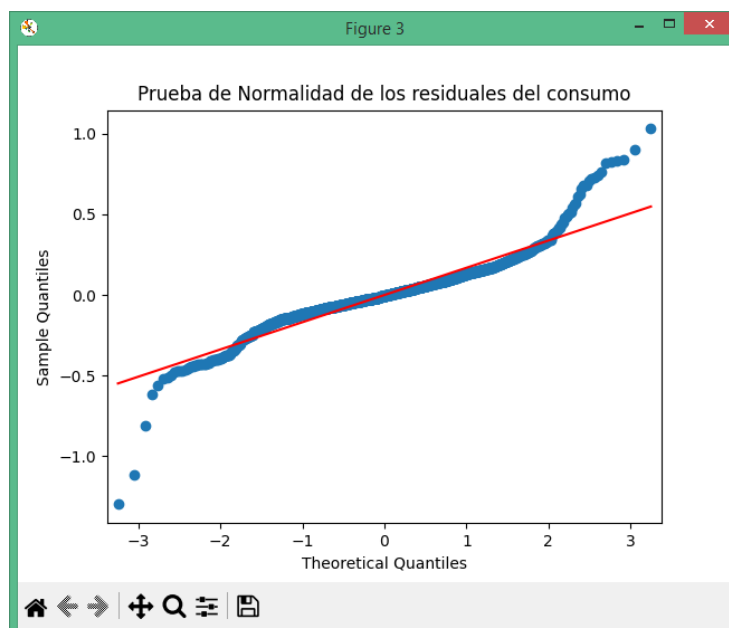
Prueba de normalidad de los residuos de la potencia



Fuente: Elaboración propia

Figura 45

Prueba de normalidad de los residuos del consumo,



Fuente: Elaboración propia

Para la Linealidad: La relación entre las variables debe ser lineal. Si no es así, los resultados de la regresión pueden ser poco confiables. Los datos mostrados en la Tabla 18, indican una media de los residuos muy cercana a cero. Por lo tanto, la linealidad se cumple. Este comportamiento se observa en las gráficas de la Figura 46, figura 47 y Figura 48.

Tabla 18

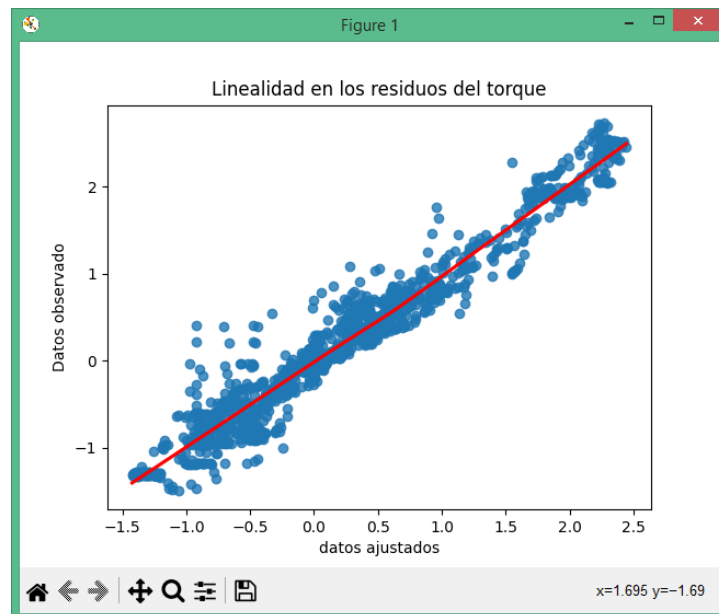
Evaluación de linealidad. Con evolución de media de residuos.

Salida	Media de residuos
Torque	8,88E-16
Potencia	2,26E-15
Consumo	1,76E-15

Fuente: Elaboración propia

Figura 46

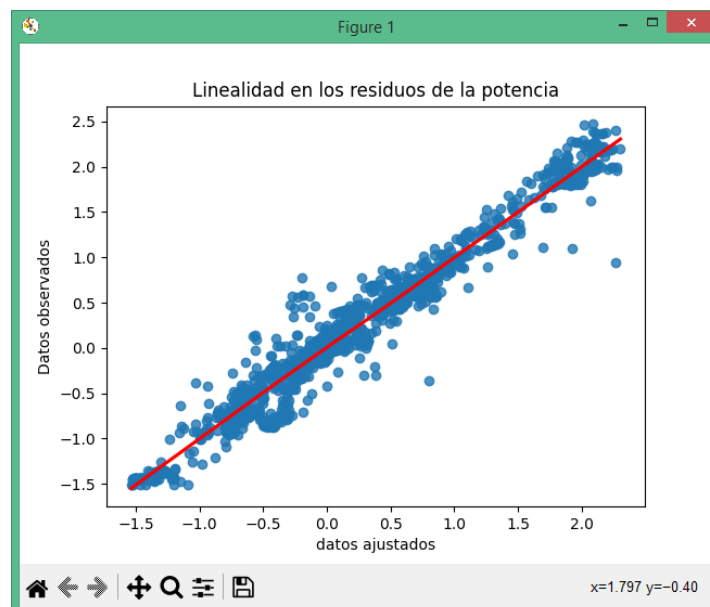
Linealidad de los residuos del torque



Fuente: Elaboración propia

Figura 47

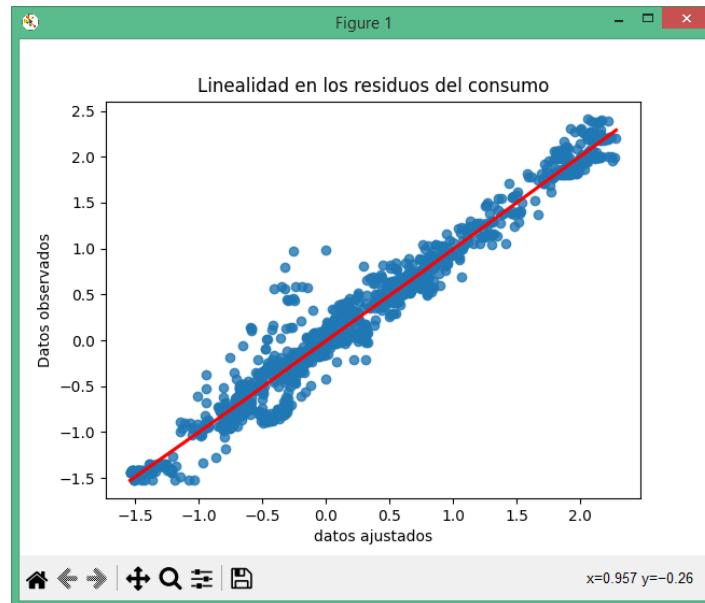
Linealidad de los residuos de la potencia



Fuente: Elaboración propia

Figura 48

Linealidad de los residuos del consumo

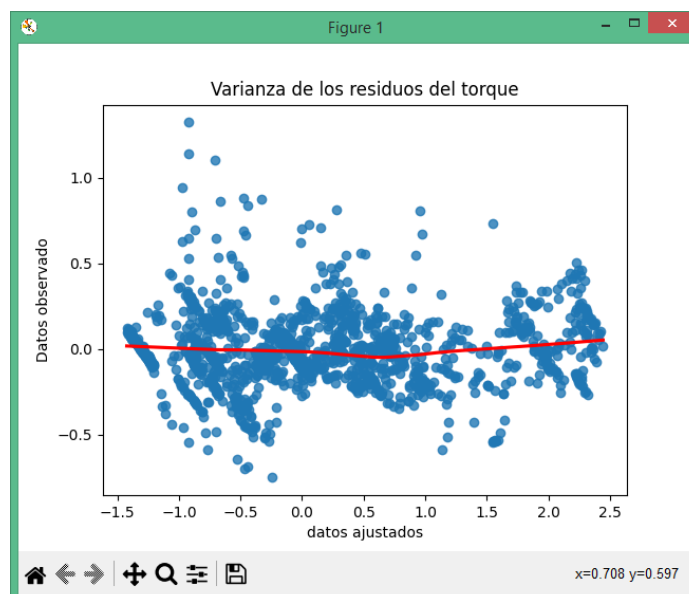


Fuente: Elaboración propia

Para la Homocedasticidad: Los errores deben tener una varianza constante. En la figura 49, Figura 50, Figura. 51, los datos muestran Heterocedasticidad.

Figura 49

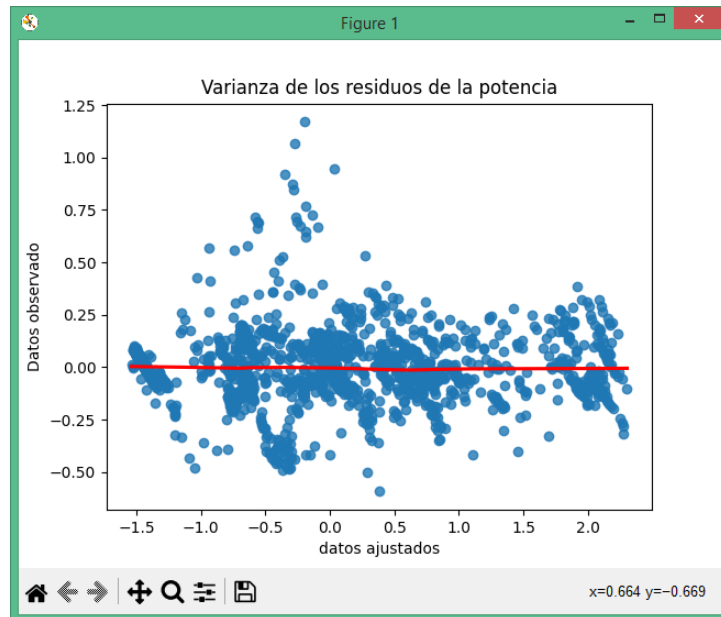
Prueba de Homocedasticidad de los residuos del torque



Fuente: Elaboración propia

Figura 50

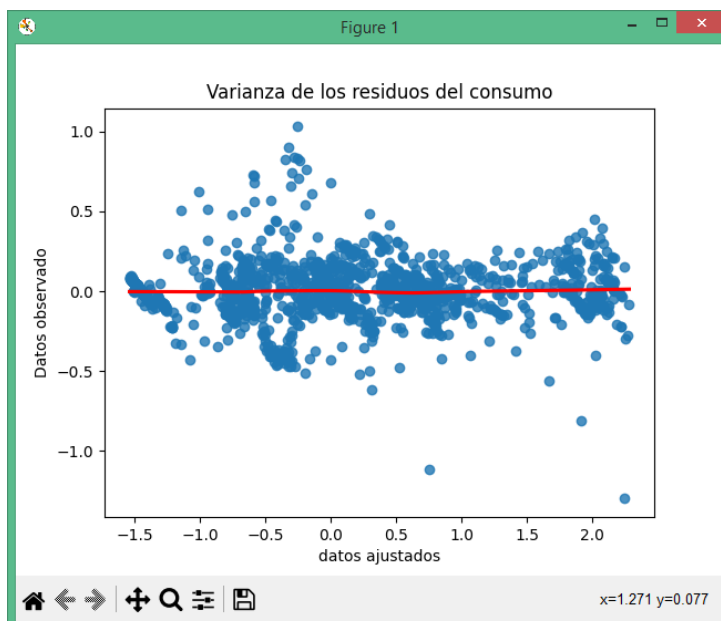
Prueba de Homocedasticidad de los residuos de la potencia



Fuente: Elaboración propia

Figura 51

Prueba de Homocedasticidad de los residuos del consumo



Fuente: Elaboración propia

Del análisis ejecutado, se resume el comportamiento en la Tabla 19, mostrándose que el modelo de regresión lineal múltiple no es el más adecuado para el pronóstico.

Tabla 19

Resumen de Verificación de los Supuestos del modelo MLR.

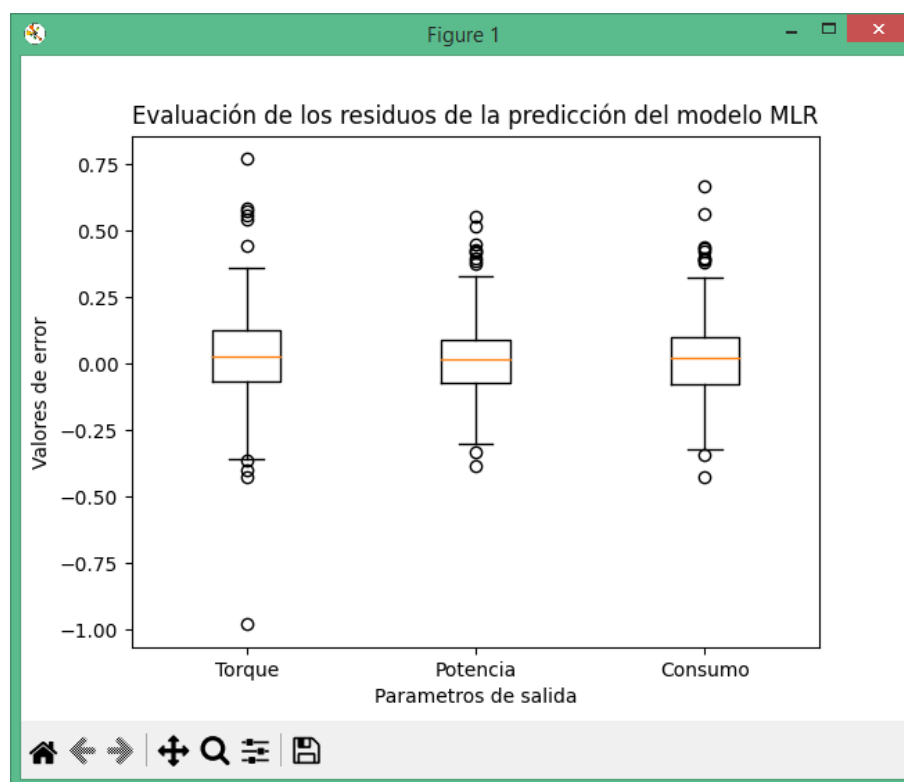
Supuesto	Criterio	Torque	Potencia	Consumo	Condicion
Independencia	Coefficiente de Durbin Watson (entre 1,5 y 2,5)	0,204791	0,389722	0,399728	No cumple este criterio
Normalidad	Prueba de normalidad con $p \geq 0.05$	1,78E-24	9,05E-30	5,43E-31	No cumple este criterio
Linealidad	La media de residuos cerca a cero	8,88E-16	2,26E-15	1,76E-15	Si cumple este criterio
Homocedasticidad	Los errores deben tener una varianza constante	Gráfico disperso	Gráfico disperso	Gráfico disperso	No cumple este criterio

4.2.4. Análisis del pronóstico del modelo MLR

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro de salida estandarizado, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas y bigotes, mostrados en la Figura 52.

Figura 52

Evaluación de los residuos del modelo MLR



Fuente: Elaboración propia

4.2.5. Pronóstico del modelo KNN en regresión

Desarrollado y entrenado el modelo K-vecinos más cercanos (KNN), se verifica la correspondencia entre los parámetros de salida pronosticados y los parámetros de salida instantáneos para cada valor de K: K=3, K=4, K=5, K=6, K=7 y K=8. Se realiza una evaluación para cada uno de estos valores, calculando el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación, cuyos resultados se muestran en la Tabla 20.

Tabla 20

Evaluación de los parámetros pronosticados por el modelo KNN.

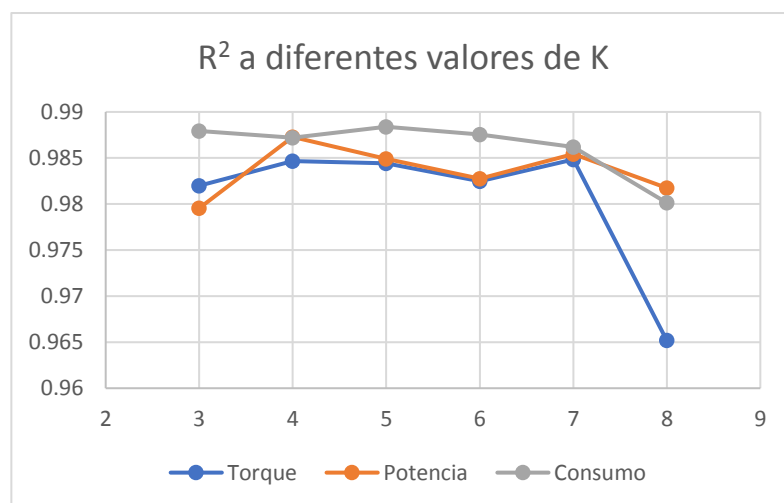
K	Salida	MAE	MSE	RMSE	R ²
K=3	Torque	0,05820363	0,01637956	0,12798265	0,98199396
	Potencia	0,06560619	0,01874686	0,13691918	0,97955036
	Consumo	0,06258157	0,01110742	0,10539175	0,98792328
K=4	Torque	0,06150931	0,0141625	0,11900628	0,98468176
	Potencia	0,06768169	0,01223785	0,11062483	0,98728299
	Consumo	0,06697379	0,01228517	0,11083846	0,98718939
K=5	Torque	0,06375463	0,01552837	0,12461287	0,98440885
	Potencia	0,07026334	0,01492829	0,12218136	0,98489465
	Consumo	0,06873447	0,01147528	0,10712274	0,9883877
K=6	Torque	0,07387285	0,01901023	0,13787759	0,98246098
	Potencia	0,07469359	0,01835126	0,13546683	0,98274544
	Consumo	0,07210437	0,01342035	0,11584622	0,98753313
K=7	Torque	0,06821043	0,01522553	0,12339179	0,98483892
	Potencia	0,07378796	0,0147823	0,12158247	0,98543019
	Consumo	0,07564061	0,0140339	0,11846474	0,98619131
K=8	Torque	0,0916511	0,03268155	0,18078039	0,96518537
	Potencia	0,0819486	0,01728453	0,13147063	0,98174498
	Consumo	0,08587	0,01902696	0,13793825	0,98013056

Fuente: Elaboración propia.

A partir de los datos de la Tabla 20, se observan las tendencias en la Figura 53. El modelo queda entrenado para K=4, debido a que para el cálculo del factor de carga es conveniente emplear el mejor modelo de pronóstico para la potencia.

Figura 53

Registro de R^2 para cada valor de K.



Una vez ejecutado el entrenamiento del modelo de KNN en regresión, se verifica la correspondencia entre los parámetros de salida pronosticados y los parámetros de salida instantáneos. Esto se realiza mediante la evaluación del error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2), como se muestra en la Tabla 21, para cada uno de los dobleces mostrados por el método de validación cruzada k-folds.

Tabla 21

Parámetros de salida mediante KNN con validación cruzada k-folds.

K Fold	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,05493	0,00971	0,09855	0,98959	0,99238	0,98959
	Potencia	0,05738	0,00869	0,09324	0,99078	0,99207	0,99078
	Consumo	0,0546	0,00701	0,0837	0,99265	0,99375	0,99265
2	Torque	0,07073	0,01943	0,13938	0,98307	0,99252	0,98307
	Potencia	0,07693	0,02066	0,14373	0,98119	0,99252	0,98119
	Consumo	0,07157	0,01499	0,12242	0,9864	0,99383	0,9864
3	Torque	0,05621	0,01463	0,12095	0,98635	0,99203	0,98635
	Potencia	0,05745	0,01122	0,10594	0,98883	0,99205	0,98883
	Consumo	0,05602	0,00951	0,0975	0,99054	0,99396	0,99054
4	Torque	0,08109	0,02796	0,16722	0,97073	0,99221	0,97073
	Potencia	0,08209	0,02777	0,16664	0,97252	0,9931	0,97252
	Consumo	0,07143	0,01434	0,11973	0,98582	0,99376	0,98582
5	Torque	0,06665	0,02306	0,15185	0,97472	0,99295	0,97472
	Potencia	0,06207	0,01302	0,11412	0,98594	0,99244	0,98594
	Consumo	0,06367	0,01466	0,12106	0,98406	0,9945	0,98406
6	Torque	0,05315	0,01234	0,11108	0,9858	0,99287	0,9858

	Potencia	0,06724	0,01265	0,11247	0,98625	0,99238	0,98625
	Consumo	0,06273	0,01082	0,10401	0,988	0,99395	0,988
	Torque	0,06427	0,01533	0,12381	0,98451	0,99229	0,98451
7	Potencia	0,06911	0,01471	0,12127	0,98506	0,99276	0,98506
	Consumo	0,06415	0,01113	0,10548	0,98867	0,99429	0,98867
	Torque	0,05489	0,01127	0,10618	0,98822	0,99246	0,98822
8	Potencia	0,05881	0,00998	0,09987	0,98949	0,99223	0,98949
	Consumo	0,05544	0,00807	0,08985	0,99148	0,99407	0,99148
	Torque	0,04436	0,00701	0,08375	0,99373	0,99241	0,99373
9	Potencia	0,05705	0,01105	0,10514	0,99002	0,99257	0,99002
	Consumo	0,06099	0,01067	0,1033	0,99044	0,99419	0,99044
	Torque	0,05907	0,01563	0,12503	0,98487	0,99315	0,98487
10	Potencia	0,06621	0,01202	0,10965	0,98846	0,9921	0,98846
	Consumo	0,07282	0,01476	0,12147	0,98595	0,99418	0,98595

Después de ejecutar, 10 dobleces de entrenamiento, se construye la Tabla 22 que muestra el promedio de los parámetros pronosticados por el modelo KNN para K=4

Tabla 22

Promedio de métricas de evaluación dados por el modelo KNN.

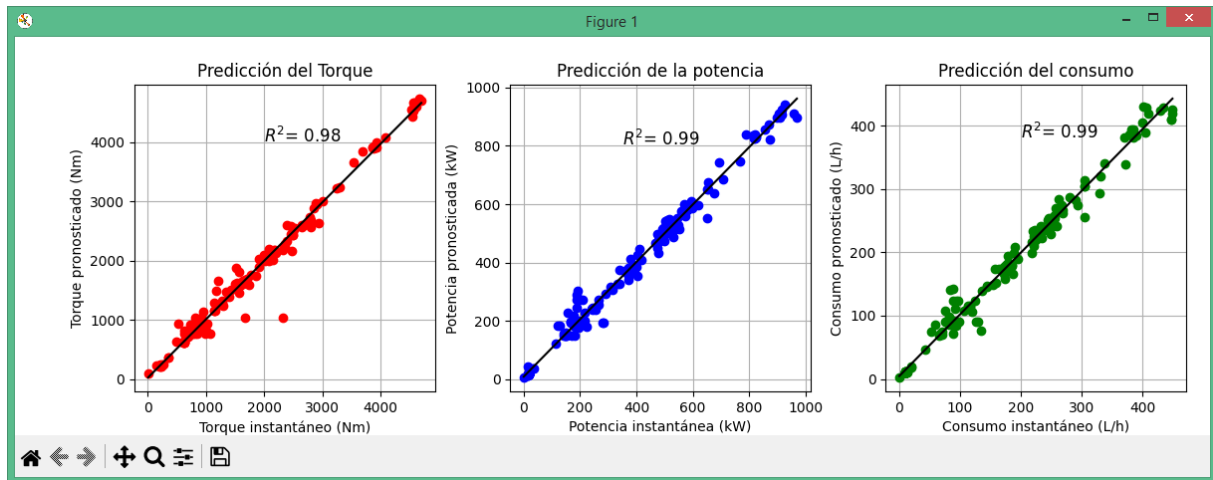
Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
Torque	0,06053	0,01564	0,12278	0,98416	0,99253	0,98416
Potencia	0,06543	0,01418	0,11721	0,98585	0,99242	0,98585
Consumo	0,06334	0,01159	0,10685	0,98840	0,99405	0,98840

Fuente: Elaboración propia.

En la Figura 54, se observa que los resultados de un pronóstico, el comportamiento del torque instantáneo vs el torque pronosticado, con un $R^2 = 0,96$. De igual manera el comportamiento de la potencia instantánea vs potencia pronosticada, con un $R^2 = 0,97$, Finalmente, se observar el consumo de combustible instantáneo vs el consumo de combustible pronosticado, con un $R^2 = 0,98$.

Figura 54

Salidas instantáneas vs Salidas pronosticadas para K=4.



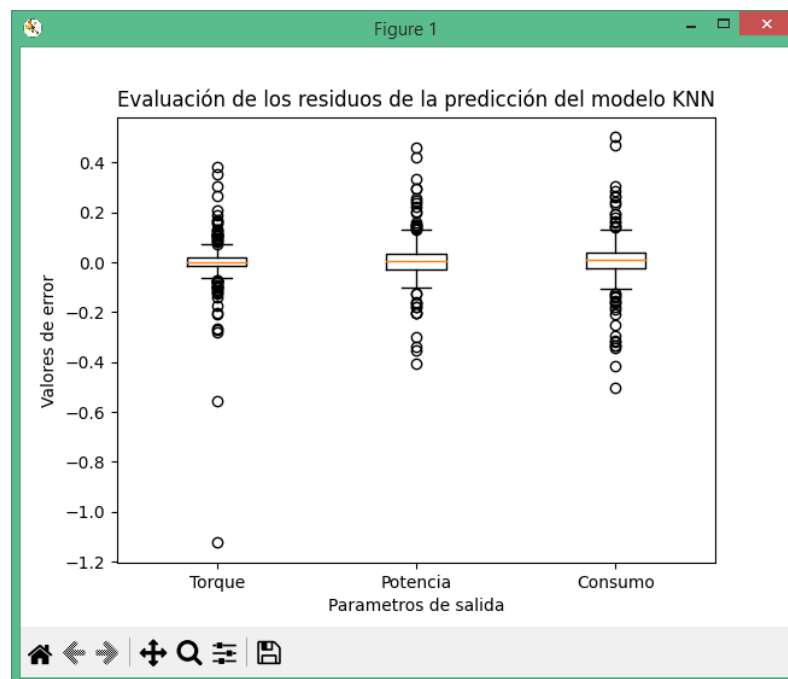
Fuente: Elaboración propia.

4.2.6. Análisis del pronóstico del modelo KNN

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro de salida estandarizada, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas y bigotes, mostrados en la Figura 55.

Figura 55

Evaluación de los residuos del modelo KNN



Fuente: Elaboración propia

4.2.7. Pronóstico del modelo máquina de vectores soporte (SVM)

Desarrollado y entrenado el modelo de máquina de vectores soporte (SVM) en regresión, se verifica la correspondencia entre los parámetros de salida pronosticados y los parámetros de salida instantáneos. La evaluación se realiza mediante el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2), como se muestra en la Tabla 23, para cada uno de los dobleces mostrados por el método de validación cruzada K-folds.

Tabla 23

Parámetros de salida mediante SVM con validación cruzada k-folds

K-Folds	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,1284	0,0311	0,1765	0,9666	0,9587	0,9666
	Potencia	0,1042	0,024	0,1548	0,9746	0,9718	0,9746
	Consumo	0,1081	0,0236	0,1535	0,9753	0,9722	0,9753
2	Torque	0,1625	0,0452	0,2125	0,9606	0,9592	0,9606
	Potencia	0,1246	0,0385	0,1962	0,965	0,9729	0,965
	Consumo	0,1217	0,0319	0,1787	0,971	0,9726	0,971
3	Torque	0,1363	0,0404	0,2011	0,9623	0,959	0,9623
	Potencia	0,113	0,0269	0,1641	0,9732	0,9719	0,9732
	Consumo	0,1145	0,0265	0,1629	0,9736	0,9724	0,9736
4	Torque	0,1571	0,0579	0,2406	0,9394	0,9614	0,9394
	Potencia	0,1231	0,0385	0,1963	0,9619	0,9732	0,9619
	Consumo	0,1216	0,0373	0,1932	0,9631	0,9735	0,9631
5	Torque	0,1429	0,0451	0,2124	0,9505	0,9603	0,9505
	Potencia	0,1104	0,0259	0,1609	0,9721	0,972	0,9721
	Consumo	0,1114	0,0286	0,169	0,9689	0,9729	0,9689
6	Torque	0,1452	0,0431	0,2077	0,9503	0,9603	0,9503
	Potencia	0,1168	0,0322	0,1794	0,965	0,9727	0,965
	Consumo	0,1217	0,0354	0,1881	0,9607	0,9736	0,9607
7	Torque	0,1386	0,0369	0,1922	0,9627	0,9591	0,9627
	Potencia	0,1108	0,0231	0,152	0,9765	0,9716	0,9765
	Consumo	0,1105	0,0232	0,1522	0,9764	0,9721	0,9764
8	Torque	0,1489	0,0467	0,216	0,9512	0,9603	0,9512
	Potencia	0,1143	0,0252	0,1586	0,9735	0,9719	0,9735
	Consumo	0,1111	0,0244	0,1562	0,9742	0,9723	0,9742
9	Torque	0,1338	0,0336	0,1834	0,9699	0,958	0,9699
	Potencia	0,1148	0,0278	0,1669	0,9749	0,9717	0,9749
	Consumo	0,115	0,0258	0,1605	0,9769	0,9719	0,9769
10	Torque	0,1396	0,0398	0,1995	0,9615	0,9591	0,9615
	Potencia	0,1106	0,0232	0,1523	0,9777	0,9713	0,9777
	Consumo	0,1174	0,0255	0,1598	0,9757	0,9721	0,9757

Fuente: Elaboración propia

Después de ejecutar, 10 dobleces de entrenamiento, se construye la Tabla 24 que muestra el promedio de los parámetros pronosticados por el modelo SVM

Tabla 24

Promedio de métricas de evaluación dados por el modelo SVM.

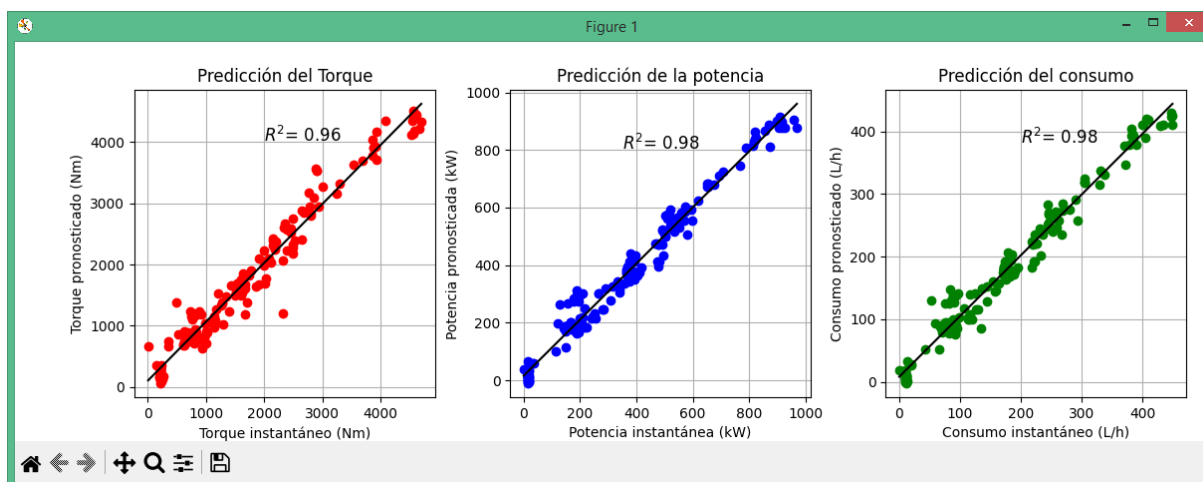
Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
Torque	0,14333	0,04199	0,20419	0,95752	0,95955	0,95752
Potencia	0,11426	0,02853	0,16814	0,97143	0,97210	0,97143
Consumo	0,11530	0,02822	0,16741	0,97159	0,97258	0,97159

Fuente: Elaboración propia.

Así mismo en la Figura 56, se observa que el comportamiento del torque instantáneo vs el torque pronosticado, con un $R^2 = 0,96$. De igual manera el comportamiento de la potencia instantánea vs potencia pronosticada, con un $R^2 = 0,98$, Finalmente, en la misma figura se observa el consumo de combustible instantáneo vs el consumo de combustible pronosticado, con un $R^2 = 0,98$.

Figura 56

Salidas instantáneas vs salidas pronosticadas.



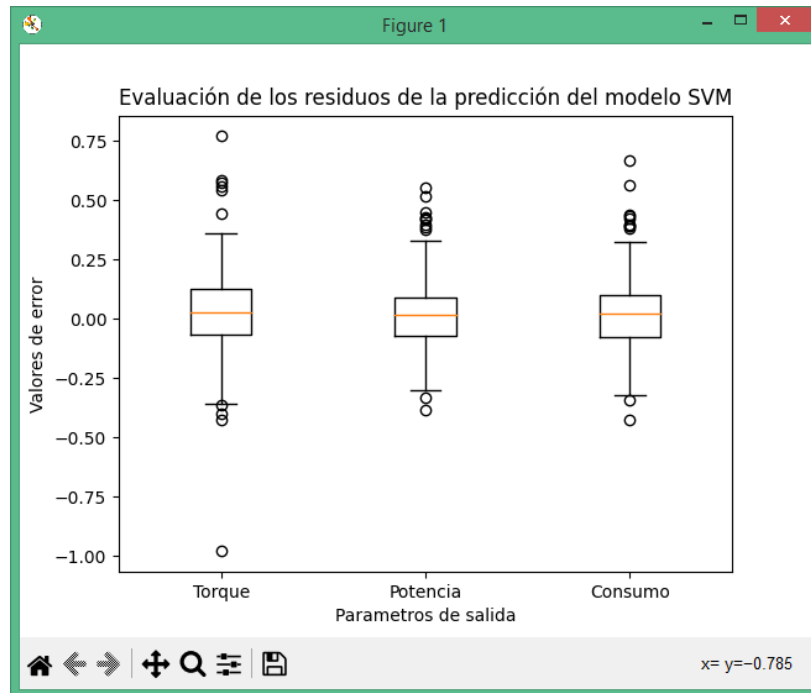
Fuente: Elaboración propia.

4.2.8. Análisis del pronóstico del modelo SVM

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro de salida estandarizado, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas (Figura 57).

Figura 57

Evaluación de los residuos del modelo SVM



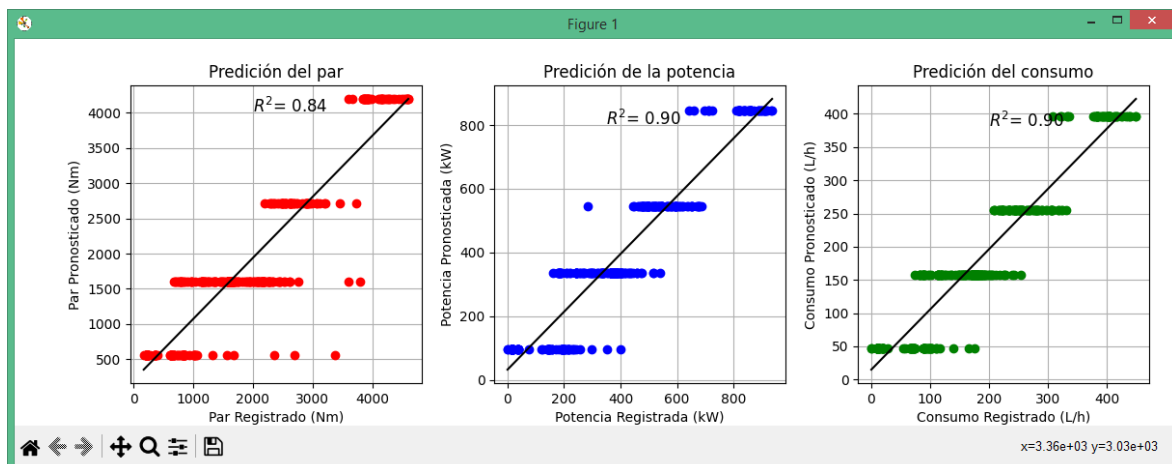
Fuente: Elaboración propia.

4.2.9. Pronóstico del modelo de regresión árboles de decisión (DT)

Desarrollado el modelo de árboles de decisión (DT), se verifica la precisión de las salidas pronosticadas frente a las salidas instantáneas cuando el parámetro "máxima profundidad" cambia de 2 a 8, estos definidos en la sección de entrenamiento de árboles de decisión. Dicho comportamiento se puede observar desde la Figura 58 hasta la Figura 64.

Figura 58

Salidas instantáneas vs salidas pronosticadas, profundidad =2

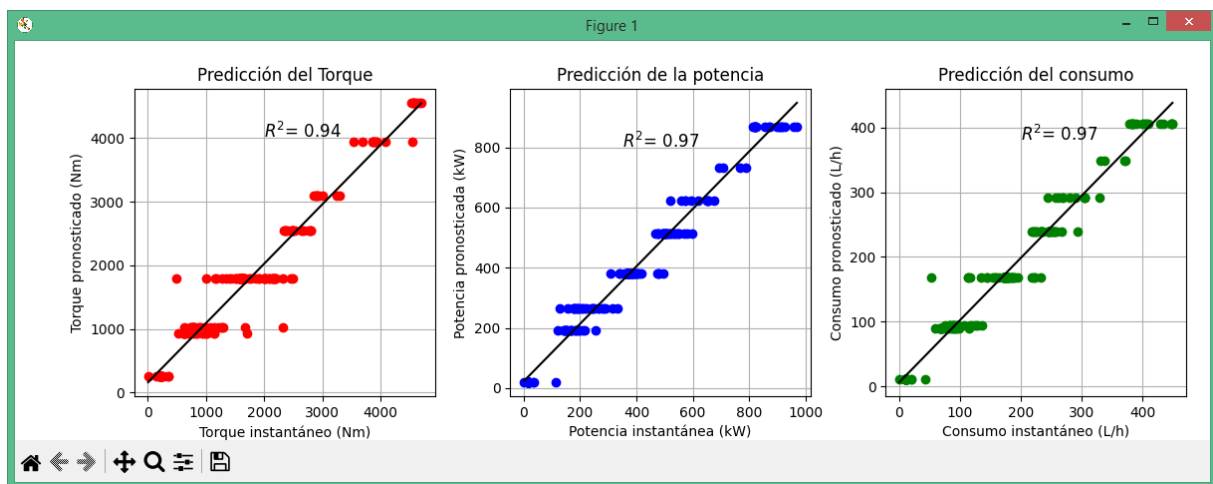


Fuente: Elaboración propia.

La evaluación de la precisión de las dimensiones de salida pronosticadas se presenta en la Tabla 24, donde se analizan el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2).

Figura 59

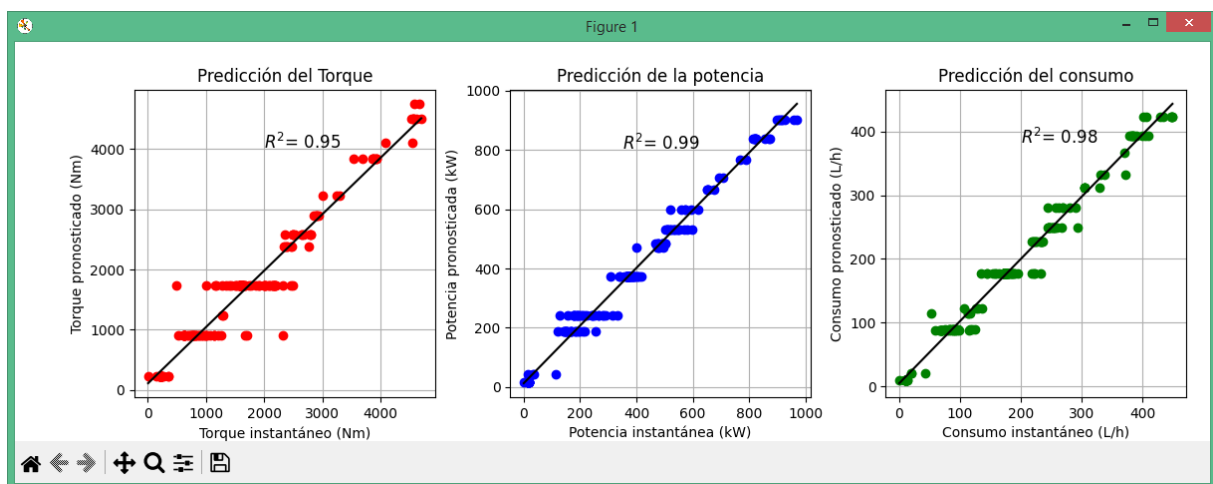
Salidas instantáneas vs salidas pronosticadas, profundidad =3.



Fuente: Elaboración propia.

Figura 60

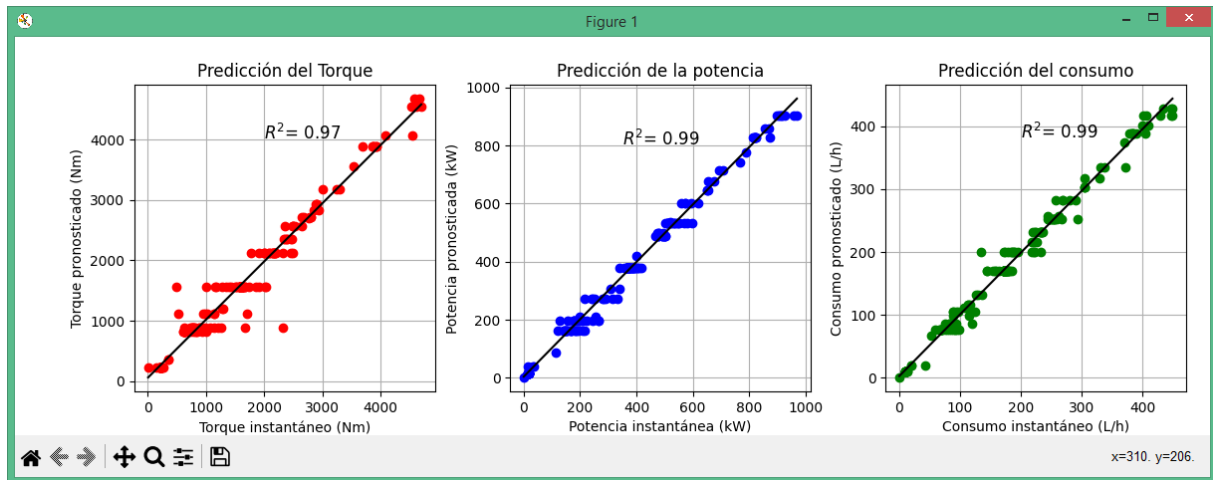
Salidas instantáneas vs salidas pronosticadas, profundidad =4.



Fuente: Elaboración propia.

Figura 61

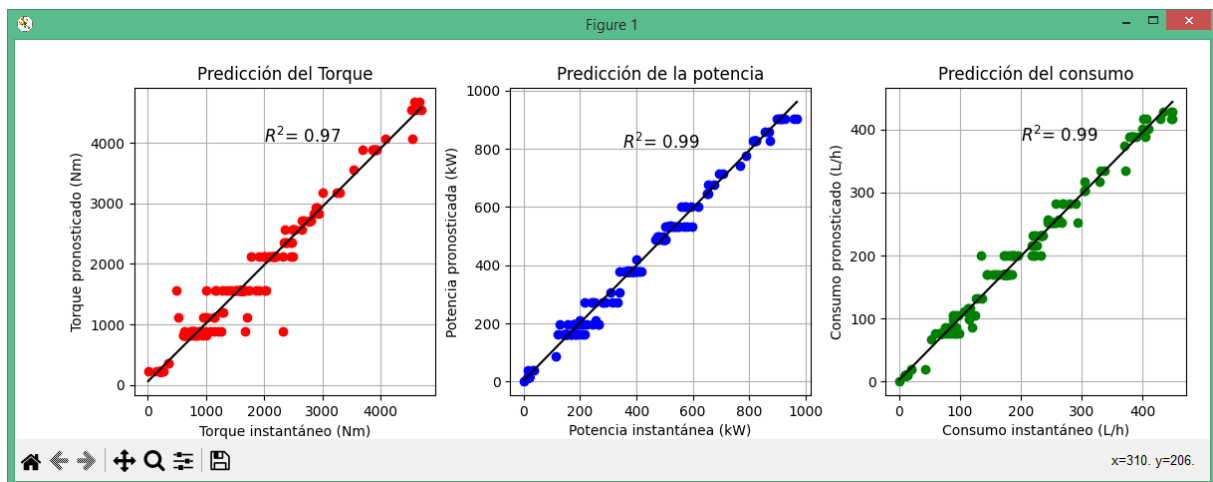
Salidas instantáneas vs salidas pronosticadas, profundidad =5.



Fuente: Elaboración propia.

Figura 62

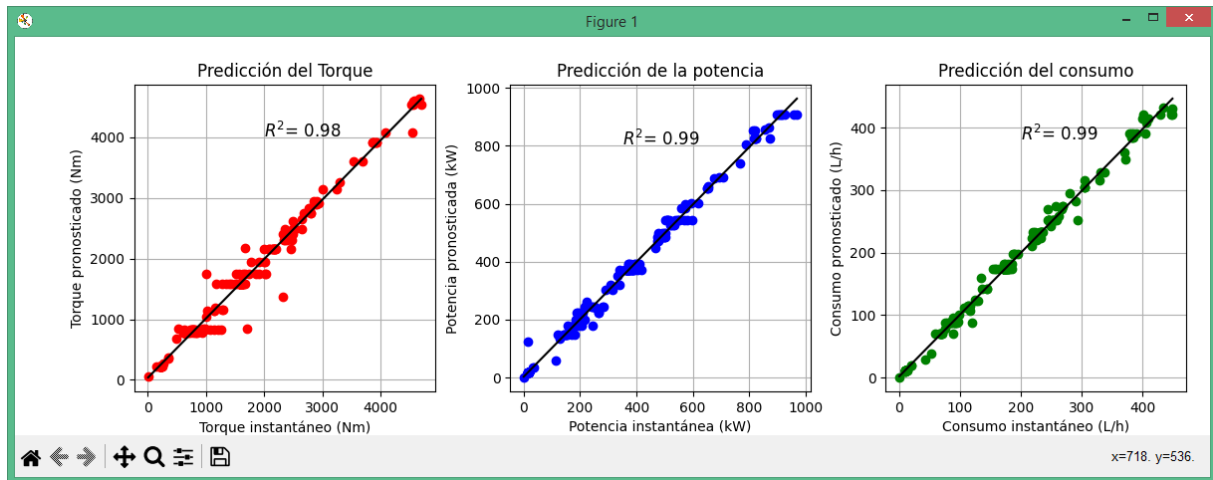
Salidas instantáneas vs salidas pronosticadas, profundidad =6.



Fuente: Elaboración propia.

Figura 63

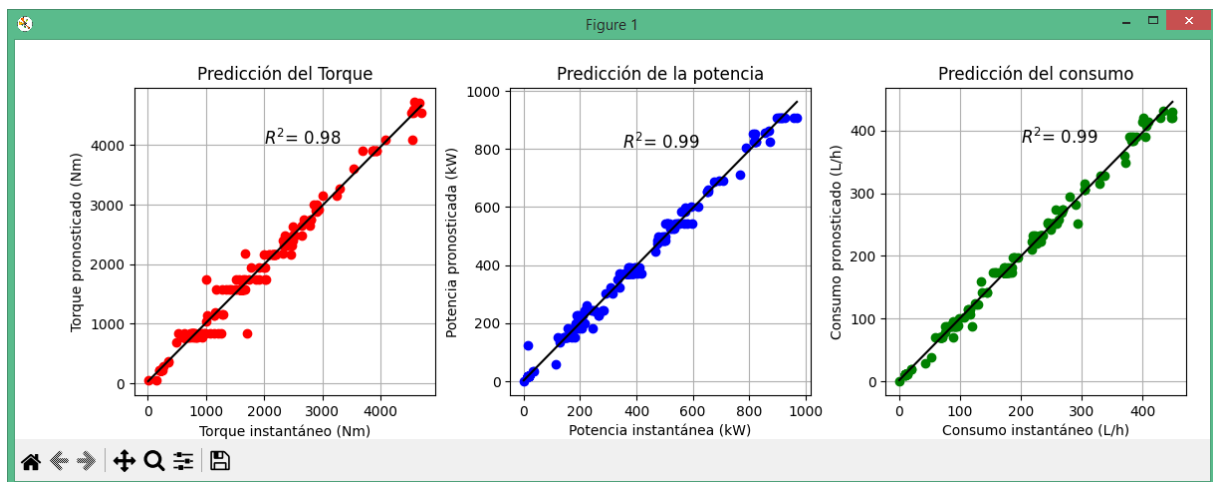
Salidas instantáneas vs salidas pronosticadas, profundidad =7.



Fuente: Elaboración propia.

Figura 64

Salidas instantáneas vs salidas pronosticadas, profundidad =8



Fuente: Elaboración propia.

A partir de los datos de la Tabla 25, se desarrolla el diagrama de la Figura 64. El modelo queda entrenado para un nivel de profundidad de 8, dado que para el cálculo del factor de carga es conveniente utilizar la potencia

Tabla 25

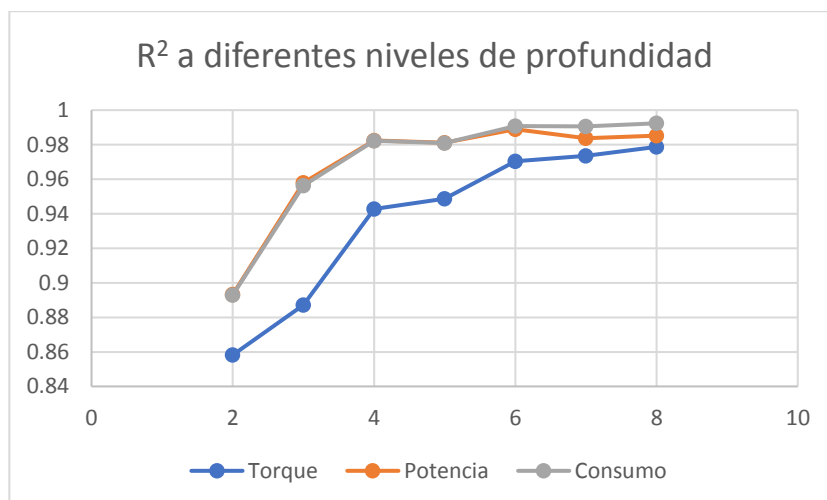
Parámetros pronosticados por el modelo DT para diferentes niveles de profundidad.

Profundidad	Salida	MAE	MSE	RMSE	R ²
2	Torque	0,28135408	0,13905711	0,37290362	0,85821518
	Potencia	0,27076528	0,10392809	0,3223788	0,89328339
	Consumo	0,27255234	0,10454427	0,32333306	0,8929134
3	Torque	0,20089598	0,11191246	0,33453319	0,8870736
	Potencia	0,13662164	0,04133678	0,20331447	0,9579478
	Consumo	0,13558956	0,04279743	0,20687539	0,95637951
4	Torque	0,16584784	0,06350083	0,2519937	0,94277243
	Potencia	0,09512379	0,01917418	0,13847087	0,9824625
	Consumo	0,0974271	0,01941733	0,13934609	0,98216988
5	Torque	0,12882622	0,04475811	0,21156113	0,94867693
	Potencia	0,08313289	0,0168642	0,12986224	0,98112505
	Consumo	0,08126132	0,016995	0,13036489	0,98104285
6	Torque	0,08878801	0,02669977	0,16340064	0,97032407
	Potencia	0,06271421	0,0103736	0,10185087	0,9889382
	Consumo	0,0562074	0,00860277	0,09275111	0,990814
7	Torque	0,06958132	0,02330448	0,15265807	0,97355958
	Potencia	0,05882628	0,01463043	0,12095633	0,98378684
	Consumo	0,05046779	0,0084847	0,09211241	0,99067061
8	Torque	0,06505967	0,02138724	0,14624378	0,97871413
	Potencia	0,05748398	0,01494263	0,12224006	0,98513932
	Consumo	0,0512248	0,00770664	0,08778747	0,99237067

Fuente: Elaboración propia.

Figura 65

Registro de R² para cada nivel de profundidad en modelo DT



Fuente: Elaboración propia.

Una vez ejecutado el entrenamiento del modelo de árboles de decisión con una profundidad de 8, se verifica la correspondencia entre los parámetros de salida pronosticados y los parámetros de salida instantáneos. Esto se realiza mediante la evaluación del error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2), como se muestra en la Tabla 26, para cada uno de los dobles mostrados por el método de validación cruzada k-folds.

Tabla 26

Parámetros de salida mediante DT con validación cruzada k-folds.

K-folds	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,16062	0,05522	0,23499	0,94084	0,95126	0,94084
	Potencia	0,10319	0,02506	0,15831	0,97341	0,97993	0,97341
	Consumo	0,09299	0,02251	0,15002	0,9764	0,9805	0,9764
2	Torque	0,16675	0,09547	0,30898	0,91677	0,94459	0,91677
	Potencia	0,11383	0,03129	0,17688	0,97151	0,98027	0,97151
	Consumo	0,10458	0,02612	0,16163	0,97629	0,98062	0,97629
3	Torque	0,15405	0,0486	0,22046	0,95464	0,95274	0,95464
	Potencia	0,09257	0,01836	0,13549	0,98174	0,97898	0,98174
	Consumo	0,08987	0,0164	0,12807	0,98368	0,97978	0,98368
4	Torque	0,14769	0,04692	0,2166	0,95088	0,94507	0,95088
	Potencia	0,10081	0,03058	0,17488	0,96973	0,97991	0,96973
	Consumo	0,09209	0,02074	0,14401	0,97948	0,9812	0,97948
5	Torque	0,14056	0,08094	0,2845	0,91126	0,94658	0,91126
	Potencia	0,08847	0,02163	0,14707	0,97665	0,97942	0,97665
	Consumo	0,09446	0,02577	0,16052	0,97198	0,9826	0,97198
6	Torque	0,15775	0,04759	0,21815	0,94522	0,95123	0,94522
	Potencia	0,102	0,02455	0,15668	0,97332	0,97953	0,97332
	Consumo	0,10966	0,02686	0,16388	0,97021	0,98196	0,97021
7	Torque	0,13408	0,06278	0,25055	0,93657	0,94351	0,93657
	Potencia	0,09914	0,02321	0,15233	0,97642	0,97931	0,97642
	Consumo	0,09218	0,01916	0,1384	0,98049	0,9813	0,98049
8	Torque	0,16698	0,06405	0,25308	0,93305	0,94698	0,93305
	Potencia	0,0977	0,01983	0,14082	0,9791	0,97915	0,9791
	Consumo	0,08845	0,01566	0,12512	0,98348	0,98143	0,98348
9	Torque	0,15553	0,05004	0,22369	0,9553	0,93953	0,9553
	Potencia	0,08459	0,01852	0,13607	0,98329	0,97863	0,98329
	Consumo	0,08934	0,02389	0,15455	0,97859	0,97972	0,97859
10	Torque	0,15332	0,05464	0,23376	0,94712	0,9455	0,94712
	Potencia	0,08046	0,01407	0,1186	0,9865	0,97828	0,9865
	Consumo	0,08389	0,01668	0,12916	0,98411	0,97965	0,98411

Fuente: Elaboración propia.

Después de ejecutar, 10 dobleces de entrenamiento, se construye la Tabla 27 que muestra el promedio de los parámetros pronosticados por el modelo DT para un nivel de profundidad de 8

Tabla 27

Promedio de métricas de evaluación dados por el modelo DT.

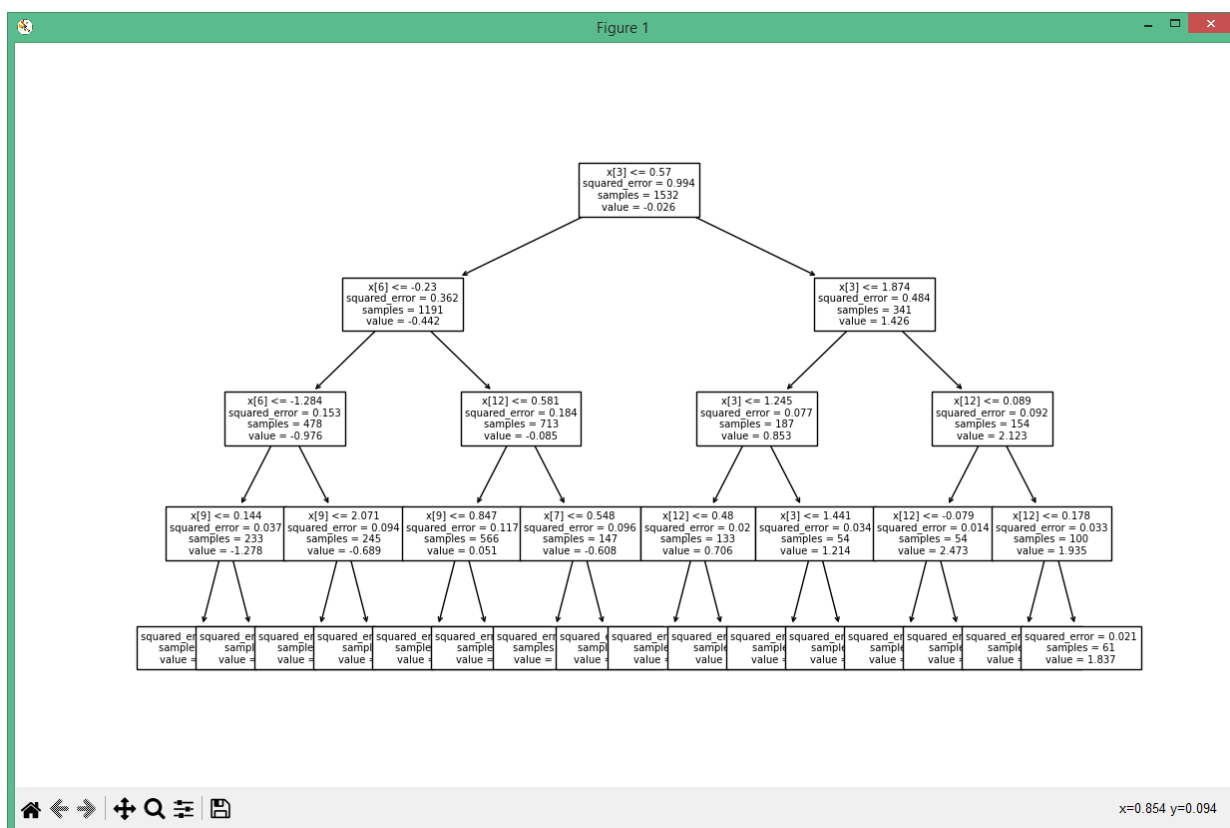
Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
Torque	0,15373	0,06062	0,24448	0,93917	0,94670	0,93917
Potencia	0,09628	0,02271	0,14971	0,97717	0,97934	0,97717
Consumo	0,09375	0,02138	0,14554	0,97847	0,98088	0,97847

Fuente: Elaboración propia.

Se construye el diagrama de árbol de decisiones desarrollado por el modelo, mostrados en la Figura 66.

Figura 66

Representación del modelo árbol de decisión desarrollado.



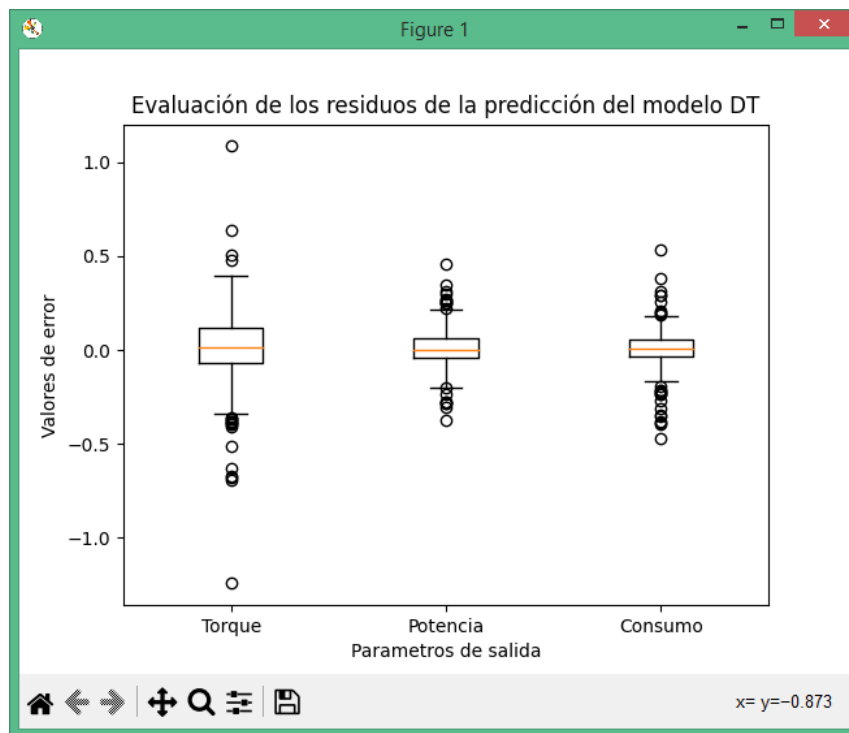
Fuente: Elaboración propia.

4.2.10. Análisis del pronóstico del modelo DT

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro de salida estandarizados, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas y bigotes mostrados en la Figura 67.

Figura 67

Evaluación de los residuos del modelo DT



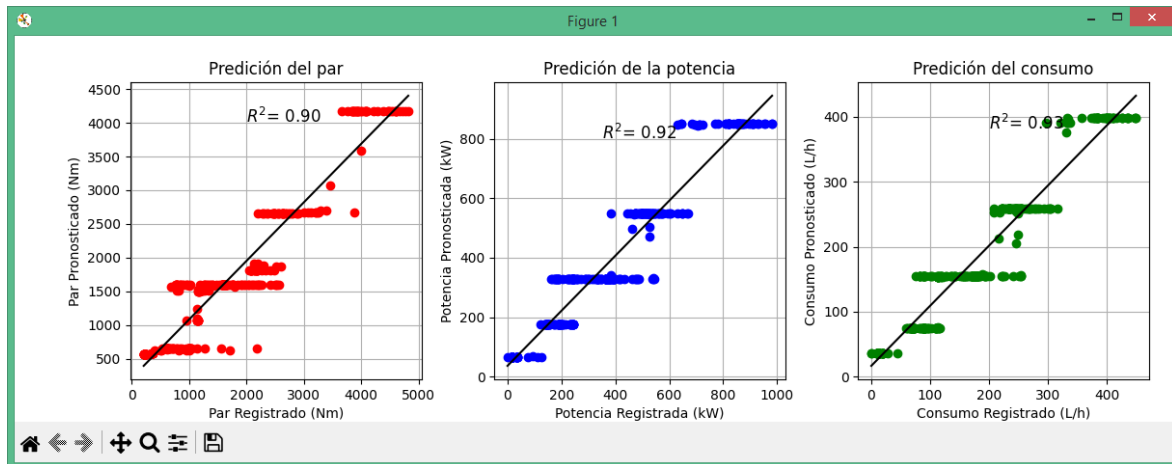
Fuente: Elaboración propia.

4.2.11. Pronóstico del modelo bosques aleatorios (RF) en regresión

Desarrollado el modelo de bosques aleatorios (RF), se verifica la precisión de las salidas pronosticadas frente a las salidas instantáneas, para un nivel de profundidad escalable desde 2 hasta 8, cuyo comportamiento se observar desde la Figura 68 hasta la Figura 75.

Figura 68

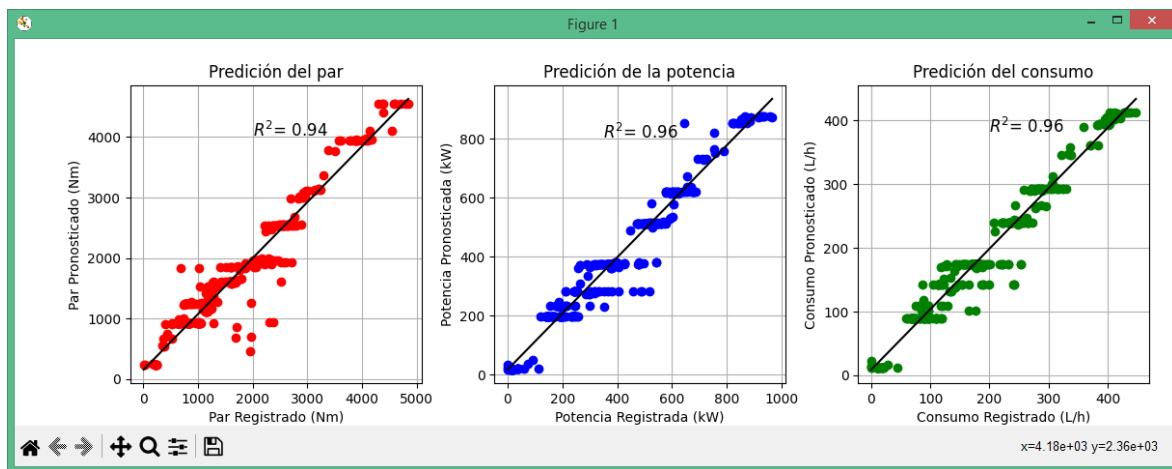
Salidas instantáneas vs salidas pronosticadas, profundidad =2.



Fuente: Elaboración propia.

Figura 69

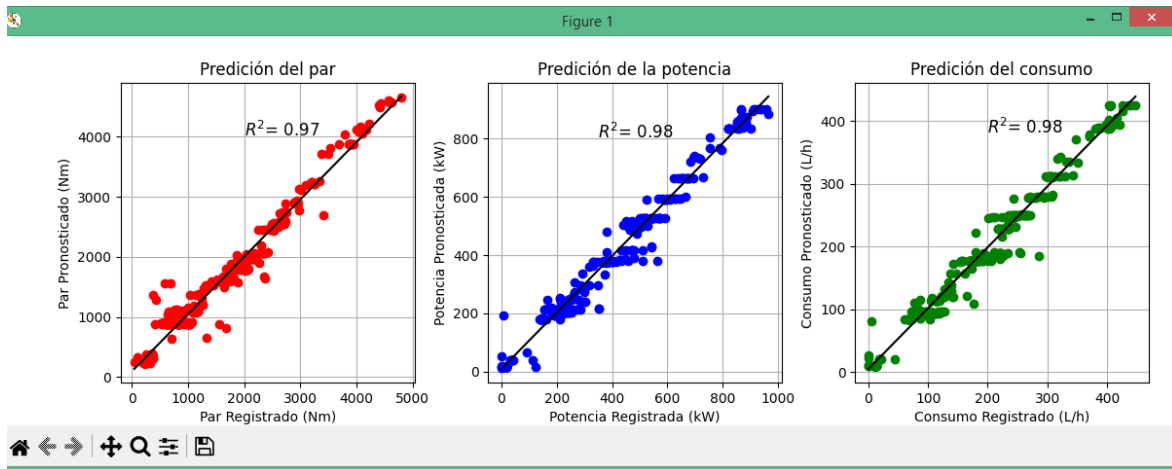
Salidas instantáneas vs salidas pronosticadas, profundidad =3



Fuente: Elaboración propia.

Figura 70

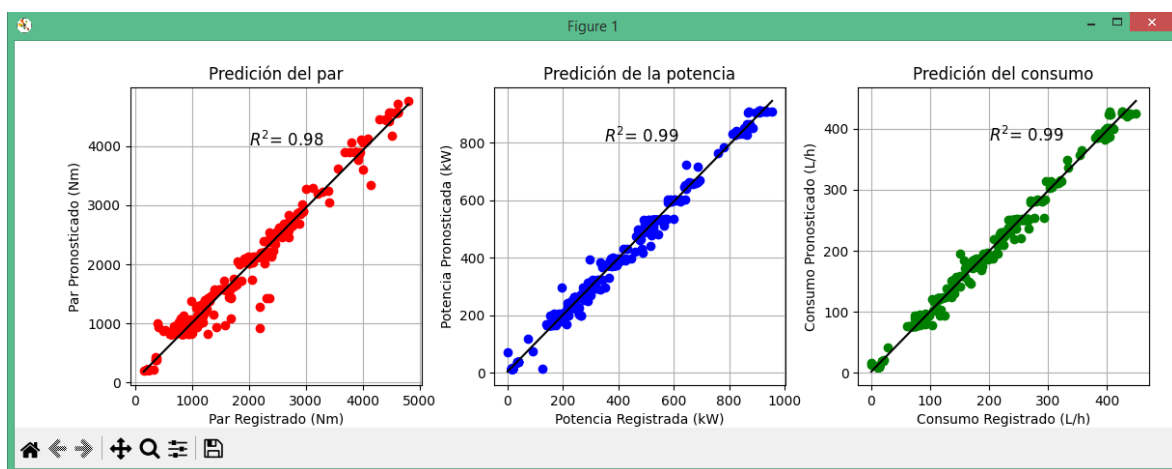
Salidas instantáneas vs salidas pronosticadas, profundidad =4



Fuente: Elaboración propia.

Figura 71

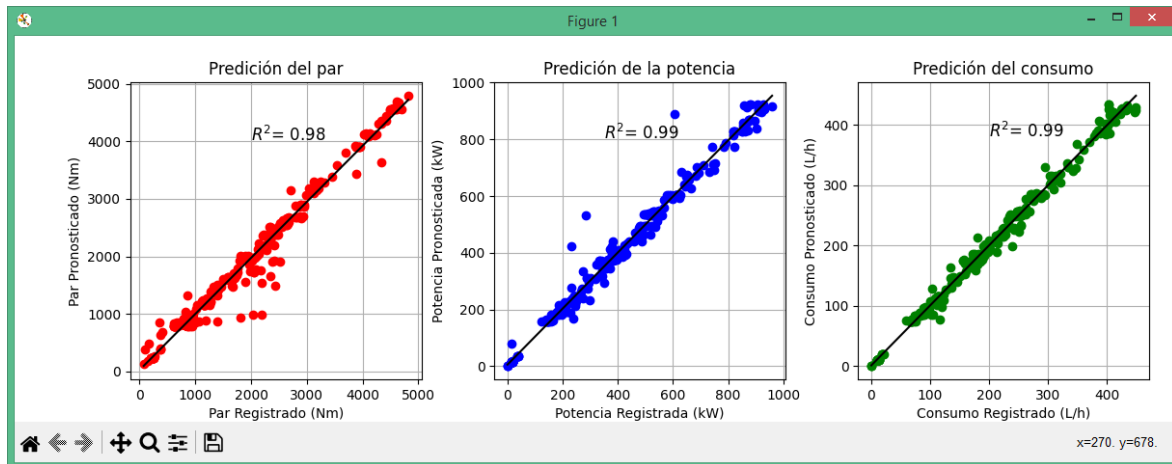
Salidas instantáneas vs salidas pronosticadas, profundidad =5.



Fuente: Elaboración propia.

Figura 72

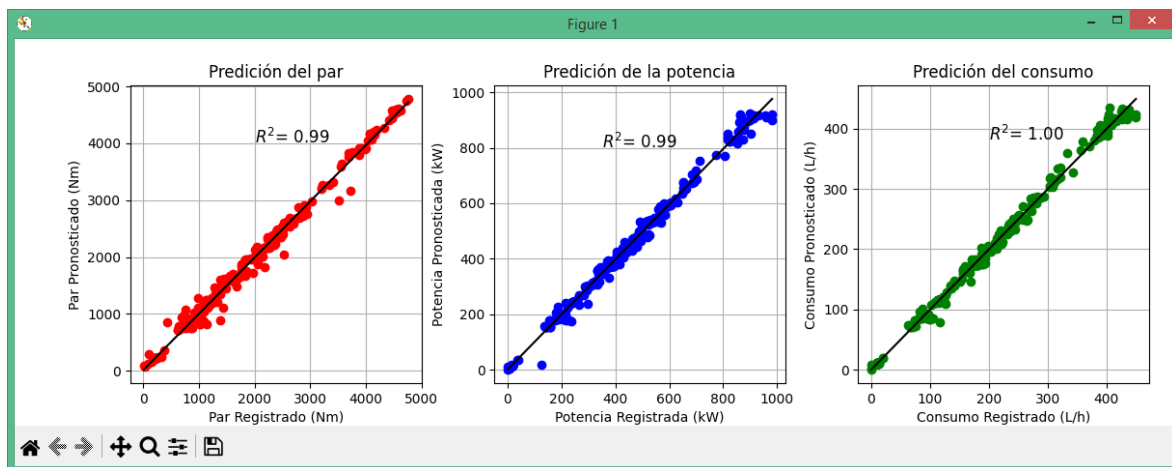
Salidas instantáneas vs salidas pronosticadas, profundidad =6.



Fuente: Elaboración propia.

Figura 73

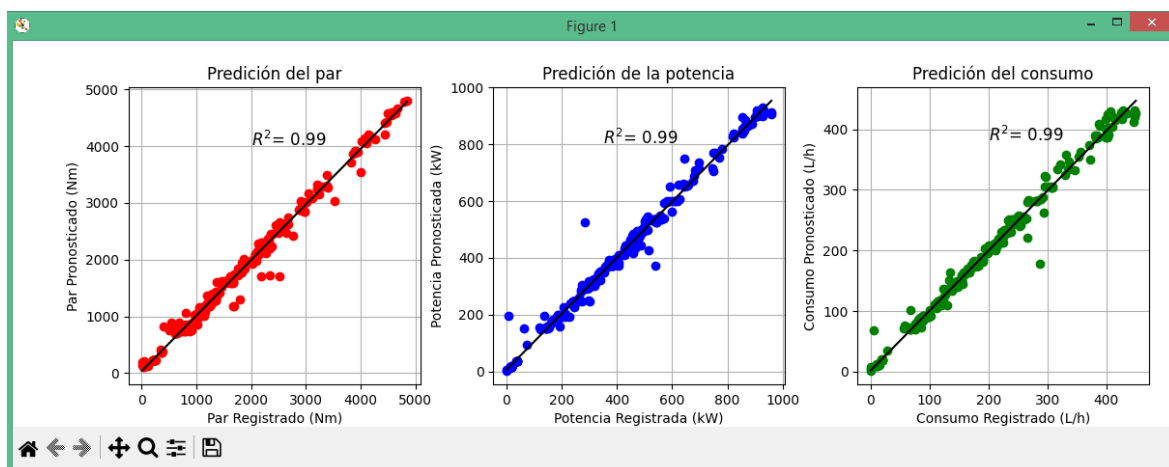
Salidas instantáneas vs salidas pronosticadas, profundidad =7.



Fuente: Elaboración propia.

Figura 74

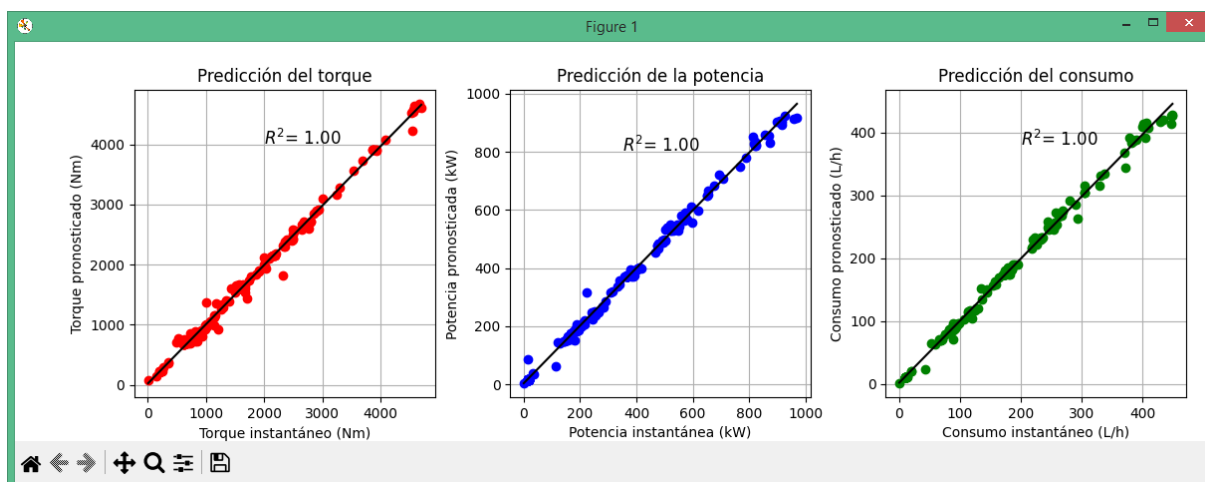
Salidas instantáneas vs salidas pronosticadas, profundidad =8.



Fuente: Elaboración propia.

Figura 75

Salidas instantáneas vs salidas pronosticadas, profundidad =9.



Fuente: Elaboración propia.

La evaluación de la precisión de los parámetros de salidas pronosticadas se presenta en la Tabla 27, donde se evalúan el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2) para cada nivel de profundidad. A partir de los datos de la Tabla 28 se entrena el modelo para un nivel de profundidad de 7, debido a a que para el cálculo del factor de carga, es conveniente emplear la potencia.

Tabla 28

Parámetros pronosticados por el modelo RF para diferentes niveles de profundidad.

Profundidad	Salida	MAE	MSE	RMSE	R ²
2	Torque	0,23743483	0,10315845	0,32118289	0,89015479
	Potencia	0,22281955	0,07715628	0,27777019	0,91975973
	Consumo	0,23244383	0,08543672	0,2922956	0,91046541
3	Torque	0,14497568	0,05505493	0,23463786	0,93621851
	Potencia	0,12427158	0,03427541	0,1851362	0,96255847
	Consumo	0,12461328	0,03360521	0,18331725	0,96363221
4	Torque	0,09700139	0,02717002	0,16483331	0,96924828
	Potencia	0,09062366	0,01923207	0,13867973	0,97963429
	Consumo	0,08961817	0,01894849	0,13765352	0,98003098
5	Torque	0,08344173	0,02214655	0,14881717	0,97764577
	Potencia	0,05661008	0,00760754	0,08722123	0,99239975
	Consumo	0,05532798	0,00633745	0,07960812	0,99367465
6	Torque	0,07298878	0,02098687	0,14486845	0,98023391
	Potencia	0,05858845	0,01354138	0,11636742	0,98712237
	Consumo	0,04999287	0,00594462	0,07710139	0,9943246
7	Torque	0,05007813	0,00706313	0,08404241	0,99270741
	Potencia	0,0444165	0,00493507	0,07025006	0,99489445
	Consumo	0,04194682	0,00403056	0,06348668	0,99578838
8	Torque	0,0506903	0,00919215	0,09587571	0,99050983
	Potencia	0,0461572	0,00949	0,09741665	0,99010591
	Consumo	0,04506462	0,0077263	0,08789936	0,99194139
9	Torque	0,04592098	0,00977286	0,09885777	0,98969463
	Potencia	0,04545302	0,00783641	0,08852351	0,99147355
	Consumo	0,04475039	0,00555938	0,07456128	0,9938949

Fuente: Elaboración propia.

Desarrollado y entrenado el modelo de Bosques aleatorios (RF) en regresión, se verifica la correspondencia entre los parámetros de salida pronosticados y los parámetros de salida instantáneos. La evaluación se realiza mediante el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R²), como se muestra en la Tabla 29, para cada uno de los dobleces mostrados por el método de validación cruzada K-folds.

Tabla 29*Parámetros de salida mediante RF con validación cruzada k-folds*

K-folds	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,04215	0,00613	0,07829	0,99343	0,99768	0,99343
	Potencia	0,03901	0,00497	0,07049	0,99473	0,99819	0,99473
	Consumo	0,03494	0,00296	0,05441	0,99690	0,99870	0,99690
2	Torque	0,05299	0,01061	0,10302	0,99075	0,99746	0,99075
	Potencia	0,04830	0,01111	0,10539	0,98989	0,99832	0,98989
	Consumo	0,04245	0,00472	0,06870	0,99572	0,99868	0,99572
3	Torque	0,04378	0,00919	0,09586	0,99143	0,99735	0,99143
	Potencia	0,03740	0,00391	0,06254	0,99611	0,99818	0,99611
	Consumo	0,03763	0,00370	0,06081	0,99632	0,99874	0,99632
4	Torque	0,05685	0,01577	0,12557	0,98349	0,99785	0,98349
	Potencia	0,05296	0,01506	0,12270	0,98510	0,99840	0,98510
	Consumo	0,04283	0,00538	0,07338	0,99467	0,99876	0,99467
5	Torque	0,05717	0,01662	0,12892	0,98178	0,99767	0,98178
	Potencia	0,04191	0,00663	0,08145	0,99284	0,99824	0,99284
	Consumo	0,04196	0,00706	0,08401	0,99233	0,99880	0,99233
6	Torque	0,03962	0,00580	0,07614	0,99333	0,99775	0,99333
	Potencia	0,04095	0,00469	0,06849	0,99490	0,99813	0,99490
	Consumo	0,04317	0,00563	0,07500	0,99376	0,99874	0,99376
7	Torque	0,04354	0,00865	0,09303	0,99126	0,99775	0,99126
	Potencia	0,04476	0,00786	0,08867	0,99201	0,99826	0,99201
	Consumo	0,04018	0,00362	0,06016	0,99631	0,99873	0,99631
8	Torque	0,04323	0,01268	0,11261	0,98675	0,99760	0,98675
	Potencia	0,04137	0,00416	0,06449	0,99562	0,99810	0,99562
	Consumo	0,03858	0,00325	0,05700	0,99657	0,99872	0,99657
9	Torque	0,03723	0,00481	0,06938	0,99570	0,99770	0,99570
	Potencia	0,04575	0,00730	0,08542	0,99341	0,99832	0,99341
	Consumo	0,03970	0,00397	0,06301	0,99644	0,99874	0,99644
10	Torque	0,03894	0,00502	0,07082	0,99515	0,99753	0,99515
	Potencia	0,03646	0,00383	0,06192	0,99632	0,99814	0,99632
	Consumo	0,03686	0,00371	0,06087	0,99647	0,99864	0,99647

Fuente: Elaboración propia.

Después de ejecutar 10 dobleces de entrenamiento, se construye la Tabla 30 que muestra el promedio de los parámetros pronosticados por el modelo RF para una profundidad de 9.

Tabla 30

Evaluación de los parámetros pronosticados por el modelo RF.

Salida	MAE	MSE	RSME	R2	Score_train	Score_test
Torque	0,04555	0,00953	0,09536	0,99031	0,99763	0,99031
Potencia	0,04289	0,00695	0,08116	0,99309	0,99823	0,99309
Consumo	0,03983	0,00440	0,06573	0,99555	0,99872	0,99555

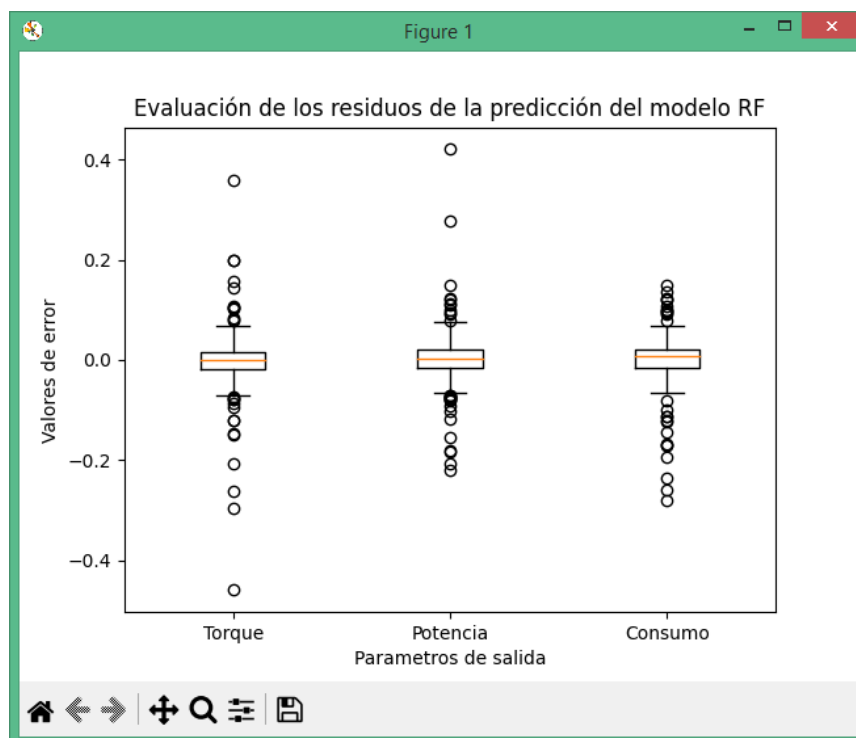
Fuente: Elaboración propia.

4.2.12. Análisis del pronóstico del modelo RF

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro estandarizados de salida, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas y bigotes, mostrados en la Figura 76

Figura 76

Evaluación de los residuos del modelo RF



Fuente: Elaboración propia.

4.2.13. Pronóstico del modelo de redes neuronales artificiales

Desarrollado el modelo, se procede a afinar la configuración de la estructura de la red neuronal. Creando en primera instancia una estructura de redes con capa de entrada en igual número que las entradas, 8 neuronas en la primera capa oculta y 8 neuronas en la segunda capa oculta, y una red en la capa de salida (8,8). El resultado del entrenamiento se puede ver en la Tabla 31.

Tabla 31

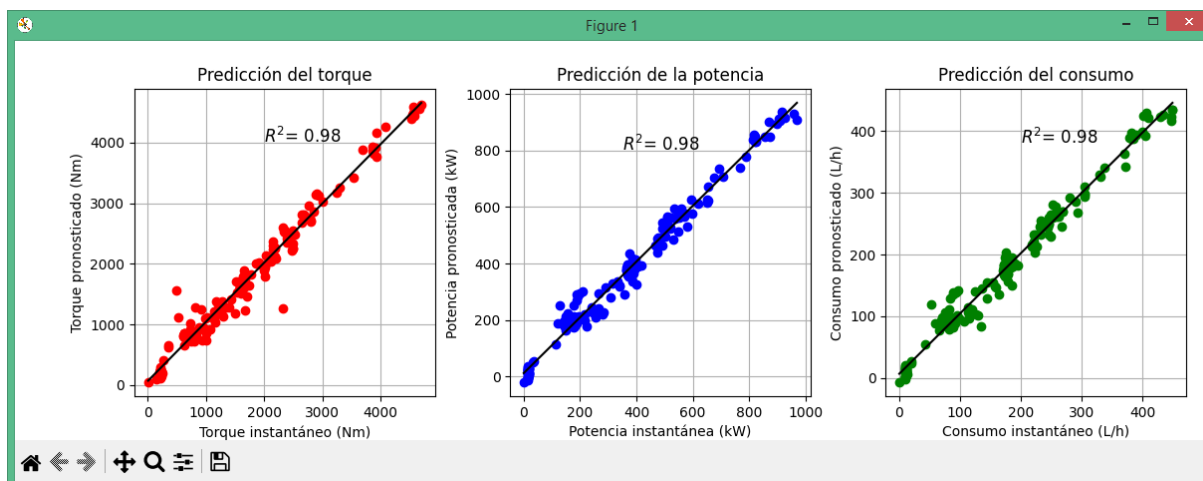
Evaluación de los parámetros pronosticados por el modelo ANR (8,8).

Salida	MAE	MSE	RMSE	R ²
Torque	0,09320483	0,0196404	0,14014423	0,97971554
Potencia	0,11388	0,02880809	0,16972946	0,96983958
Consumo	0,10775189	0,02548513	0,15964062	0,97326753

Fuente: Elaboración propia.

Figura 77

Salidas instantáneas vs salidas pronosticadas, ANR (8,8).



Fuente: Elaboración propia.

Para mejorar las respuestas en los cálculos de los errores y el coeficiente R^2 previamente mostrados, se decide reestructurar la red con capas ocultas de 3 niveles, considerando 10 neuronas para la primera capa oculta, 8 para la segunda, y 4 para la tercera (10, 8, 4). Los resultados se pueden observar en la Tabla 32 y Figura 78.

Tabla 32

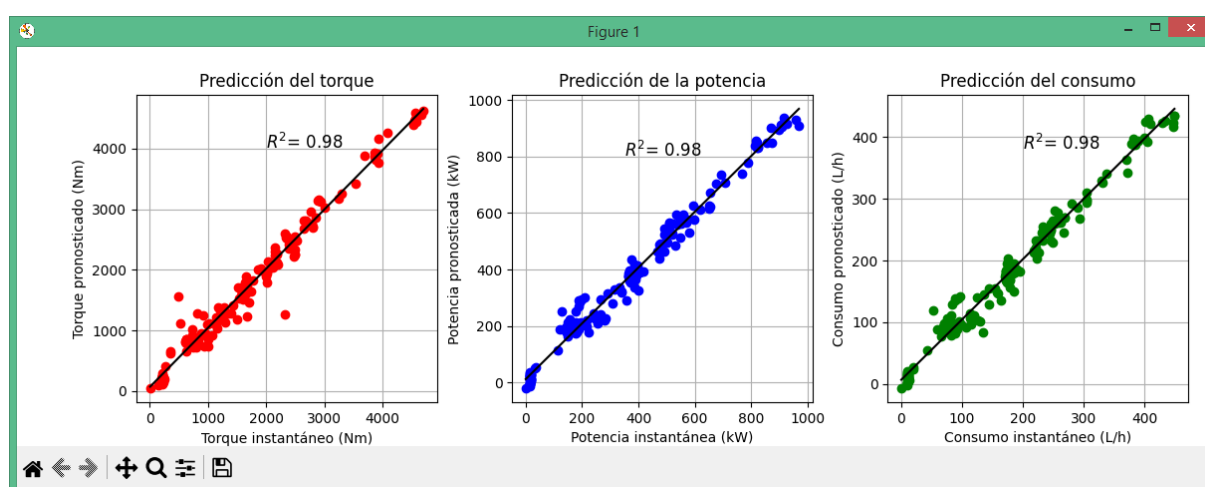
Evaluación de los parámetros pronosticados por el modelo ANR (10,8,4).

Salida	MAE	MSE	RMSE	R2
Torque	0,10411195	0,02678886	0,16367304	0,97370783
Potencia	0,10284796	0,0234444	0,15311565	0,97694992
Consumo	0,09179543	0,01637498	0,12796475	0,98395366

Fuente: Elaboración propia.

Figura 78

Salidas instantáneas vs salidas pronosticadas, ANR (10,8,4).



Para mejorar las respuestas en los cálculos de los errores y el coeficiente R^2 , se decide reestructurar la red con capas ocultas de 3 niveles, considerando 10 neuronas para la primera capa oculta, 10 para la segunda, y 10 para la tercera (10, 10, 10). Los resultados se pueden observar en la Tabla 33 y la Figura 79.

Tabla 33

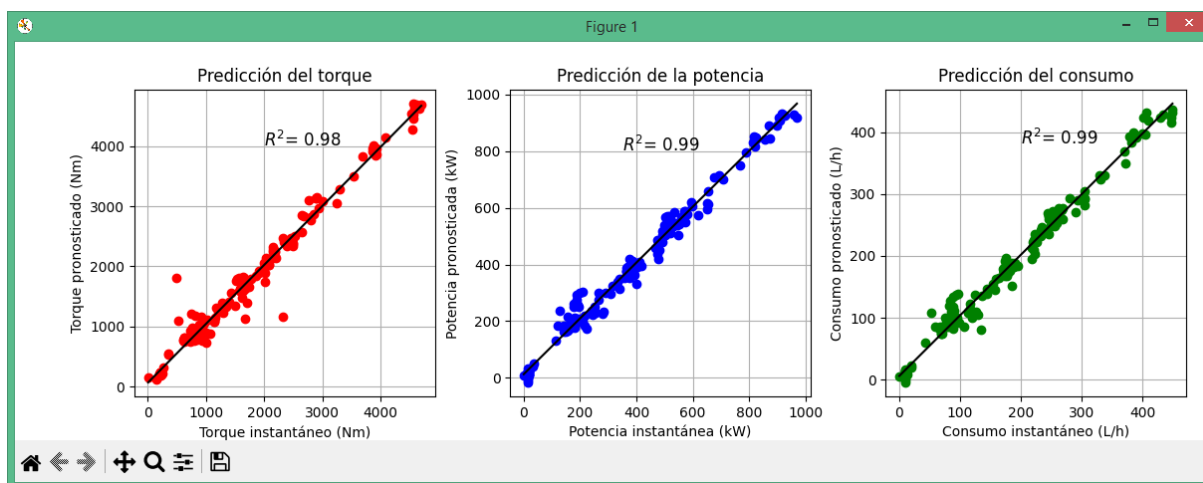
Evaluación de los parámetros pronosticados por el modelo ANR (10,10,10).

Salida	MAE	MSE	RMSE	R2
Torque	0,07858325	0,01388081	0,11781684	0,98588047
Potencia	0,08861262	0,02139528	0,14627125	0,97866486
Consumo	0,08232781	0,01464781	0,12102814	0,98557711

Fuente: Elaboración propia.

Figura 79

Salidas instantáneas vs salidas pronosticadas, ANR (10,10,10).



Fuente: Elaboración propia.

De igual manera que en los casos anteriores, se estructura la red neuronal con una configuración de 12, 12, 10, como se muestra en la Tabla 34 y la Figura 80.

Tabla 34

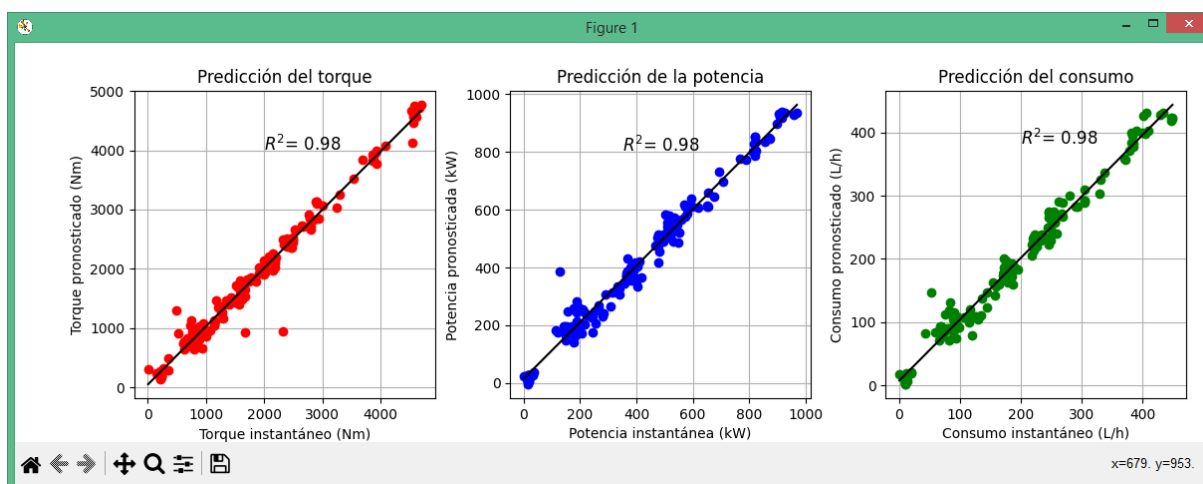
Evaluación de los parámetros pronosticados por el modelo ANR (12,12,10).

Salida	MAE	MSE	RMSE	R2
Torque	0,0808815	0,01382833	0,11759391	0,98634175
Potencia	0,09854305	0,023622	0,1536945	0,9771194
Consumo	0,09558286	0,01974216	0,1405068	0,98096946

Fuente: Elaboración propia.

Figura 80

Salidas instantáneas vs salidas pronosticadas, ANR (12,12,10).



Fuente: Elaboración propia.

Se observa que, al incrementar el número de capas y neuronas, no se obtienen mejores resultados; por el contrario, se somete al modelo a una elevada carga computacional. Por lo tanto, el modelo adecuado queda definido por modelos ANR (10,10,10) errores mostrados en Tabla 32 y comportamiento mostrado en la Figura 80.

Tabla 35

Parámetros de salida mediante ANR con validación cruzada k-folds

K-folds	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,07195	0,01393	0,11802	0,98508	0,98308	0,98508
	Potencia	0,08590	0,01787	0,13368	0,98104	0,98344	0,98104
	Consumo	0,07924	0,01325	0,11512	0,98610	0,98691	0,98610
2	Torque	0,09304	0,02241	0,14969	0,98047	0,98167	0,98047
	Potencia	0,10110	0,02677	0,16361	0,97562	0,98434	0,97562
	Consumo	0,08765	0,01876	0,13698	0,98297	0,98756	0,98297
3	Torque	0,08961	0,02194	0,14814	0,97952	0,98183	0,97952
	Potencia	0,08225	0,01720	0,13115	0,98289	0,98402	0,98289
	Consumo	0,08240	0,01558	0,12481	0,98450	0,98685	0,98450
4	Torque	0,09707	0,03472	0,18633	0,96365	0,98362	0,96365
	Potencia	0,09431	0,02436	0,15607	0,97589	0,98484	0,97589
	Consumo	0,08757	0,01617	0,12714	0,98401	0,98615	0,98401
5	Torque	0,09341	0,02078	0,14414	0,97722	0,98238	0,97722
	Potencia	0,08011	0,01490	0,12206	0,98392	0,98477	0,98392
	Consumo	0,07964	0,01629	0,12763	0,98229	0,98730	0,98229
6	Torque	0,08382	0,01553	0,12461	0,98213	0,98256	0,98213
	Potencia	0,08883	0,01672	0,12929	0,98183	0,98503	0,98183
	Consumo	0,08786	0,01698	0,13030	0,98117	0,98715	0,98117
7	Torque	0,08753	0,01729	0,13148	0,98253	0,98154	0,98253
	Potencia	0,08588	0,01724	0,13129	0,98248	0,98443	0,98248
	Consumo	0,08060	0,01207	0,10985	0,98771	0,98698	0,98771
8	Torque	0,07775	0,01661	0,12888	0,98264	0,98372	0,98264
	Potencia	0,07456	0,01127	0,10618	0,98812	0,98465	0,98812
	Consumo	0,07961	0,01316	0,11472	0,98611	0,98565	0,98611
9	Torque	0,07126	0,01055	0,10271	0,99057	0,98164	0,99057
	Potencia	0,08030	0,01312	0,11452	0,98816	0,98551	0,98816
	Consumo	0,07413	0,01095	0,10464	0,99019	0,98597	0,99019
10	Torque	0,08461	0,02464	0,15697	0,97616	0,98245	0,97616
	Potencia	0,08350	0,01445	0,12021	0,98613	0,98366	0,98613
	Consumo	0,08434	0,01494	0,12222	0,98577	0,98665	0,98577

Fuente: Elaboración propia.

Después de ejecutar 10 dobleces de entrenamiento, se construye la Tabla 36 que muestra el promedio de los parámetros pronosticados por el modelo ANN para una

configuración 10 neuromas en la primera capa oculta, 10 para la segunda, y 10 para la tercera (10, 10, 10).

Tabla 36

Evaluación de los parámetros pronosticados por el modelo ANR.

Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
Torque	0,0850	0,0198	0,1391	0,9800	0,9824	0,9800
Potencia	0,0857	0,0174	0,1308	0,9826	0,9845	0,9826
Consumo	0,0823	0,0148	0,1213	0,9851	0,9867	0,9851

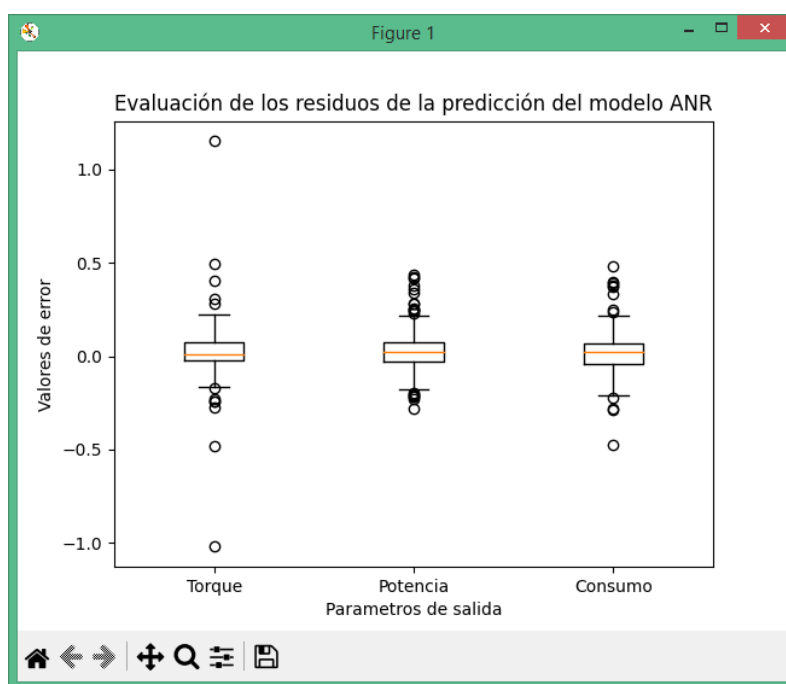
Fuente: Elaboración propia.

4.2.14. Análisis del pronóstico del modelo ANR

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro de salida estandarizada, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas y bigotes mostrados en la Figura 81.

Figura 81

Evaluación de los residuos del modelo ANR



Fuente: Elaboración propia

4.2.15. Pronóstico del modelo XGBoost

Desarrollado el modelo basado en el algoritmo XGBoost, se analiza sus predicciones evaluando del error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2), como se muestra en la Tabla 37, para cada uno de los dobles mostrados por el método de validación cruzada K-folds.

Tabla 37

Parámetros de salida mediante XGboost con validación cruzada k-folds

k_fold	Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
1	Torque	0,05444	0,00778	0,08818	0,99167	0,99544	0,99167
	Potencia	0,05347	0,00727	0,08525	0,99229	0,99482	0,99229
	Consumo	0,04800	0,00483	0,06949	0,99494	0,99559	0,99494
2	Torque	0,06440	0,01085	0,10414	0,99055	0,99509	0,99055
	Potencia	0,06838	0,01695	0,13018	0,98457	0,99514	0,98457
	Consumo	0,06300	0,01021	0,10106	0,99073	0,99571	0,99073
3	Torque	0,04716	0,00589	0,07675	0,99450	0,99542	0,99450
	Potencia	0,05574	0,00822	0,09066	0,99182	0,99452	0,99182
	Consumo	0,05537	0,00723	0,08500	0,99281	0,99574	0,99281
4	Torque	0,06153	0,00980	0,09901	0,98974	0,99547	0,98974
	Potencia	0,06251	0,01331	0,11535	0,98683	0,99526	0,98683
	Consumo	0,05591	0,00696	0,08345	0,99311	0,99587	0,99311
5	Torque	0,06391	0,01210	0,11002	0,98673	0,99551	0,98673
	Potencia	0,05807	0,01142	0,10686	0,98767	0,99504	0,98767
	Consumo	0,05480	0,01040	0,10197	0,98869	0,99619	0,98869
6	Torque	0,06137	0,01156	0,10751	0,98670	0,99517	0,98670
	Potencia	0,05572	0,00622	0,07885	0,99324	0,99477	0,99324
	Consumo	0,05414	0,00669	0,08178	0,99258	0,99604	0,99258
7	Torque	0,05091	0,00824	0,09075	0,99168	0,99515	0,99168
	Potencia	0,06099	0,01112	0,10544	0,98870	0,99543	0,98870
	Consumo	0,05174	0,00639	0,07992	0,99349	0,99609	0,99349
8	Torque	0,05271	0,00871	0,09335	0,99089	0,99500	0,99089
	Potencia	0,05959	0,00820	0,09053	0,99136	0,99471	0,99136
	Consumo	0,05397	0,00640	0,07998	0,99325	0,99602	0,99325
9	Torque	0,05058	0,00576	0,07587	0,99486	0,99499	0,99486
	Potencia	0,05685	0,01075	0,10366	0,99030	0,99478	0,99030
	Consumo	0,04384	0,00449	0,06699	0,99598	0,99627	0,99598
10	Torque	0,04658	0,00576	0,07588	0,99443	0,99516	0,99443
	Potencia	0,04895	0,00508	0,07128	0,99512	0,99460	0,99512
	Consumo	0,04922	0,00570	0,07552	0,99457	0,99606	0,99457

Fuente: Elaboración propia.

Después de ejecutar 10 dobleces de entrenamiento, se construye la Tabla 38 que muestra el promedio de los parámetros pronosticados por el modelo XGboost

Tabla 38

Evaluación de los parámetros pronosticados por el modelo XGBoost.

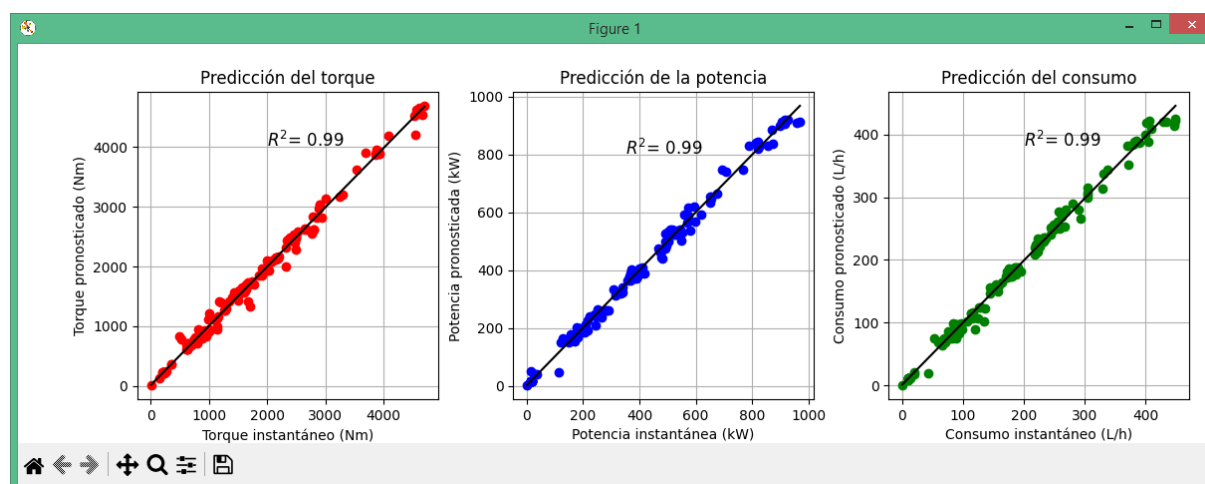
Salida	MAE	MSE	RMSE	R2	Score_train	Score_test
Torque	0,0554	0,0086	0,0921	0,9912	0,9952	0,9912
Potencia	0,0580	0,0099	0,0978	0,9902	0,9949	0,9902
Consumo	0,0530	0,0069	0,0825	0,9930	0,9960	0,9930

Fuente: Elaboración propia.

Así mismo en la Figura 82, se observa que el comportamiento de un proceso de pronóstico del modelo, mostrándose el torque instantáneo vs el torque pronosticado, con un $R^2=0,99$. De igual manera el comportamiento de la potencia instantánea vs potencia pronosticada, con un $R^2=0,99$, Finalmente, en la misma figura se observa el consumo de combustible instantáneo vs el consumo de combustible pronosticado, con un $R^2=0,99$.

Figura 82

Salidas instantáneas vs salidas pronosticadas, para el modelo XGBoost.



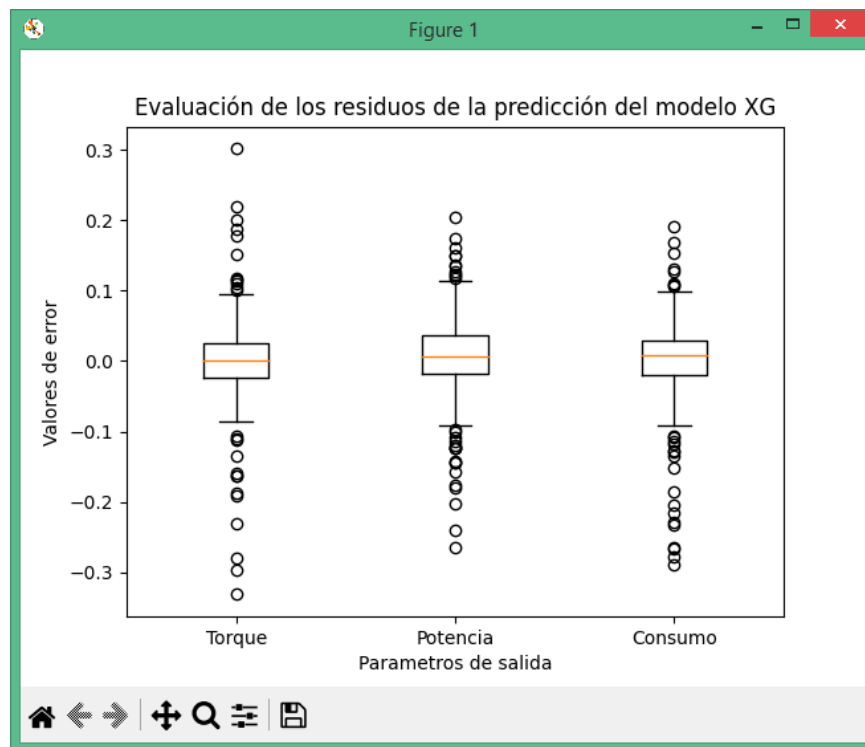
Fuente: Elaboración propia.

4.2.16. Análisis del pronóstico del modelo XGboost

Se analiza el comportamiento de los residuos del valor pronosticado respecto al valor instantáneo por cada parámetro de salida estandarizada, el torque, la potencia y el consumo de combustible, para lo cual se ejecutan un diagrama de cajas y bigotes, mostrada en la Figura 83.

Figura 83

Evaluación de los residuos del modelo XGBoost



Fuente: Elaboración propia.

4.3. Análisis de los modelos de Pronóstico

La evaluación de modelos lineales, como la Regresión Lineal Múltiple (MLR) mediante la evaluación del cumplimiento de los supuestos, ejecutada en la sección 4.2.3 de este documento y consolidada en la Tabla 19, el modelo MLR ha demostrado no ser el más adecuado para el pronóstico del torque, la potencia y el consumo de combustible del motor ensayado.

De los modelos analizados, se puede reorganizar los datos para evaluar los modelos más adecuados, tomando en cuenta el mayor coeficiente de determinación (R^2) y la menor raíz del error cuadrático medio (RMSE), Así como el valor del score de los conjuntos entrenados y el score de los conjuntos testeados, mostrados en las Tabla 14, Tabla 22, Tabla

24, Tabla 27, Tabla 30, Tabla 36, Tabla 38, para los modelos MLR, KNN, SVM,DT, RF, ANR y XGBoost respectivamente: con esta información se evalúan los modelos para el pronóstico del torque (Tabla 39), pronóstico de la potencia (Tabla 40), y pronóstico del consumo (Tabla 41).

Para el caso del torque, el menor valor de los errores MAE, MSE y RMSE se encuentra en el modelo XGBoost, donde el RMSE es de 0,0921 y el R^2 alcanza 0,9911, muy seguido por los valores de Random Forest (RF) tal como se observa en la Tabla 39.

Tabla 39

Evaluación de los modelos para el pronóstico del torque del motor.

Torque						
Modelo	MAE	MSE	RMSE	R2	Score_train	Score_test
MLR	0,1433305	0,0419907	0,204186	0,957515	0,9595464	0,957515
KNN	0,0605344	0,0156373	0,1227783	0,9841584	0,9925276	0,9841584
SVM	0,1433305	0,0419907	0,204186	0,957515	0,9595464	0,957515
DT	0,1537322	0,0606239	0,2444753	0,9391653	0,9466993	0,9391653
RF	0,0455491	0,0095279	0,0953621	0,9903051	0,9976342	0,9903051
ANN	0,0850045	0,019839	0,139097	0,9799964	0,9824467	0,9799964
XG	0,0553581	0,008644	0,0921454	0,9911735	0,99524	0,9911735

Fuente: Elaboración propia.

Para el caso de la potencia, el menor valor de los errores MAE, MSE y RMSE se encuentran entre el modelo XGBoost, y el Random Forest (RF), tal como se observa en la Tabla 40.

Tabla 40

Evaluación de los modelos para el pronóstico de la potencia del motor.

Potencia						
Modelo	MAE	MSE	RMSE	R2	Score_train	Score_test
MLR	0,1142572	0,028526	0,1681388	0,9714302	0,972095	0,9714302
KNN	0,0654337	0,0141772	0,1172063	0,9858532	0,9924224	0,9858532
SVM	0,1142572	0,028526	0,1681388	0,9714302	0,972095	0,9714302
DT	0,0962768	0,0227083	0,1497131	0,977166	0,9793409	0,977166
RF	0,0428869	0,0069518	0,081155	0,9930924	0,9982275	0,9930924
ANN	0,0856725	0,017389	0,1308072	0,982608	0,9844675	0,982608
XG	0,0580289	0,0098517	0,0978077	0,9901911	0,9949056	0,9901911

Fuente: Elaboración propia.

Para el caso del consumo de combustible, el menor valor de los errores MAE, MSE y RMSE se encuentra en el modelo Random Forest, seguido del modelo XGBoost. tal como se observa en la Tabla 41.

Tabla 41

Evaluación de los modelos para el pronóstico del consumo de combustible.

Consumo de combustible						
Modelo	MAE	MSE	RMSE	R2	Score_train	Score_test
MLR	0,1152973	0,0282173	0,1674142	0,9715885	0,9725753	0,9715885
KNN	0,0633405	0,0115934	0,1068525	0,9883996	0,9940477	0,9883996
SVM	0,1152973	0,0282173	0,1674142	0,9715885	0,9725753	0,9715885
DT	0,0937515	0,0213771	0,1455357	0,97847	0,9808755	0,97847
RF	0,0398286	0,0043988	0,0657339	0,9955486	0,9987239	0,9955486
ANN	0,082302	0,0148137	0,1213397	0,9850807	0,9867168	0,9850807
XG	0,0529999	0,0069292	0,0825165	0,9930148	0,9959572	0,9930148

Fuente: Elaboración propia.

De la Evaluación de los residuos del modelo de Pronóstico para datos estandarizados, que los residuos tienen mayor variación en la Pronóstico del torque, y en función de ellos los más convenientes para su empleo son los modelos Random forest (RF), Redes neuronales artificiales (ANN) y XGBoost.

El desarrollo de estos modelos, implica una carga computacional que se muestra en los tiempos de aplicación en cada modelo mostrados en la Tabla 42, para un computador personal cuyas características se observan en la Figura 84. Por lo tanto, entre decidir desarrollar le modelo entre XGBoost y Ramdom Forest, se decide por el modelo XGBoost, el cual es más rápido en su proceso.

Tabla 42

Tiempo de ejecución de los modelos.

Tiempo de ejecución del modelo							
Modelo	MLR	KNN	SVM	DT	RF	ANN	XG
Tiempo (s)	42	32	30	55	300	38	35

Fuente: Elaboración propia.

Figura 84

Características del computador personal empleado

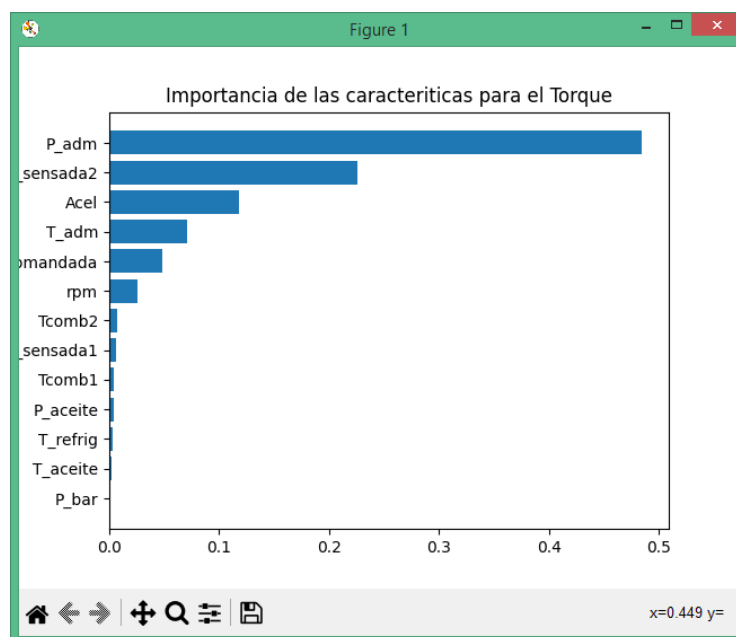
Sistema		
Procesador:	AMD A8-5600K APU with Radeon(tm) HD Graphics	3.60 GHz
Memoria instalada (RAM):	8.00 GB (7.21 GB utilizable)	
Tipo de sistema:	Sistema operativo de 64 bits, procesador x64	
Lápiz y entrada táctil:	La entrada táctil o manuscrita no está disponible para esta pantalla	

Fuente: Elaboración propia.

En función del modelo XGboost se analiza la preponderancia de los parámetros de entrada estandarizados (un análisis de sensibilidad de los parámetros de entrada) por lo cual la importancia de las características se puede ver en la Figura 85 para el caso de la potencia, Figura 86 para el caso del Torque, Figura 87 para el caso del consumo de combustible. Donde se muestra a la Presión del aire de admisión, como uno de los factores más preponderantes. Seguido de la presión sensada de riel de combustible (para el caso del torque), y presión de riel comandada. de riel de combustible (para el caso de la potencia y el consumo de combustible).

Figura 85

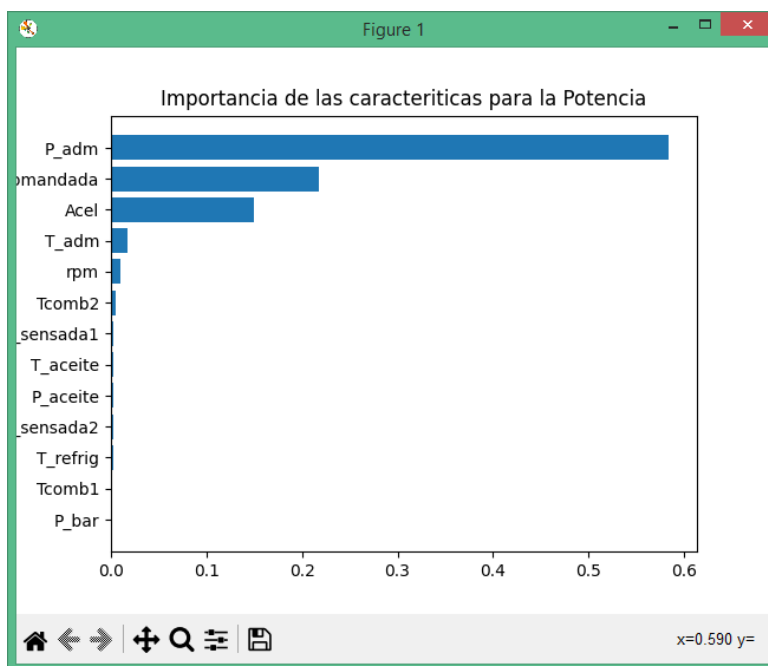
Importancia de cada característica para el torque pronosticado



Fuente: Elaboración propia

Figura 86

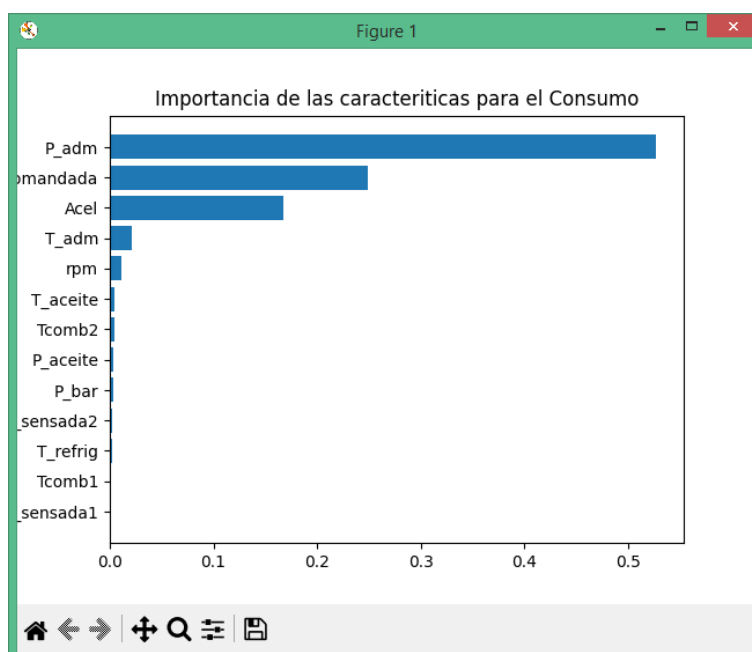
Importancia de cada característica para la potencia pronosticada



Fuente: Elaboración propia

Figura 87

Importancia de cada característica para el consumo pronosticado



Fuente: Elaboración propia

4.4. Determinación del factor de carga

Para la determinación del factor de carga, se relaciona la potencia promedio instantánea en un período de monitoreo (generalmente en múltiplos de ciclos de trabajo) con la potencia nominal del motor de combustión. Esto se codifica según lo mostrado en la Figura 88.

Figura 88

Determinación del factor de carga calculado y pronosticado.

```
print("El factor de carga calculado es:")
pot_moninal=900
avg = sum(z)/len(z)
print (avg/pot_moninal)

print("El factor de carga estimado es:")
avg = sum(z_pred)/len(z_pred)
print (avg/pot_moninal)
```

Fuente: Elaboración propia

Con los registros almacenados en la base de datos, se procedió al reentrenamiento del modelo y al cálculo del factor de carga en cada iteración, registrándose los resultados en la Tabla 43, y Figura 89, De donde se obtuvieron errores que varían desde 0,01% hasta 2,36%.

Tabla 43

Evaluación del factor de carga para la prueba dinamométrica.

Entrenamiento	Factor de carga calculado	Factor de carga pronosticado	Error	Error relativo
1°	41,43%	43,79%	2,36%	5,70%
2°	41,43%	43,18%	1,75%	4,23%
3°	41,43%	41,44%	0,01%	0,03%
4°	41,43%	42,61%	1,18%	2,85%
5°	41,43%	38,96%	-2,47%	-5,97%
6°	41,43%	42,92%	1,49%	3,61%
7°	41,43%	42,22%	0,79%	1,91%
8°	41,43%	41,36%	-0,07%	-0,17%
9°	41,43%	41,68%	0,25%	0,60%
10°	41,43%	42,08%	0,65%	1,56%

Fuente: Elaboración propia.

Figura 89

Resultados factor de carga pronosticado.

```
Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:57:54) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\Programacion\Python\ML\Tesis5\XGBoost_Regresion5.py =====

      Salida      MAE      MSE      RSME      R2
0  Torque  46.341393  8231.962445  90.730163  0.994380
1  Potencia  10.153460  278.850799  16.698826  0.995690
2  Consumo  4.925614  64.315400  8.019688  0.995399

El factor de carga calculado es:
0.4143054830287204
El factor de carga estimado es:
0.4379286091383693
>>>
===== RESTART: D:\Programacion\Python\ML\Tesis5\XGBoost_Regresion5.py =====

      Salida      MAE      MSE      RSME      R2
0  Torque  46.127785  7586.773744  87.102088  0.994475
1  Potencia  10.499371  335.633904  18.320314  0.994696
2  Consumo  4.974722  59.354098  7.704161  0.995655

El factor de carga calculado es:
0.4143054830287204
El factor de carga estimado es:
0.4318481338242015
>>>
===== RESTART: D:\Programacion\Python\ML\Tesis5\XGBoost_Regresion5.py =====

      Salida      MAE      MSE      RSME      R2
0  Torque  47.531668  11622.828179  107.809221  0.991159
1  Potencia  10.864036  455.726640  21.347755  0.992468
2  Consumo  4.655458  52.535135  7.248113  0.995974

El factor de carga calculado es:
0.4143054830287204
El factor de carga estimado es:
0.41440906157925556
>>>
===== RESTART: D:\Programacion\Python\ML\Tesis5\XGBoost_Regresion5.py =====

      Salida      MAE      MSE      RSME      R2
0  Torque  42.629183  7141.004575  84.504465  0.994882
1  Potencia  11.173118  453.562060  21.296997  0.992846
2  Consumo  5.073342  63.923048  7.995189  0.995375

El factor de carga calculado es:
0.4143054830287204
El factor de carga estimado es:
0.42612052641642556
Ln: 4 Col: 80
```

Fuente: Elaboración propia.

4.5. Validación del factor de carga en aplicación del motor

Una vez determinado el modelo de Pronóstico para el cálculo del factor de carga, se procede a analizar otras bases de datos de la operación de dos camiones mineros de volteo propulsado por el motor diésel analizado en este trabajo con series 7832 y series 7868 (Bases

de datos denominadas: DML-20090708-171744-HD785-7-sn7832, DML-20090708-173000-HD785-7-sn7832 y DML-20090911-133436-HD785-7-sn7868). Las bases de datos registran parámetros de operación del motor durante el proceso de traslado de material con el camión, como parte de su ciclo de trabajo; sin embargo, no almacena información relevante sobre los períodos de espera para el carguío. El pronóstico del torque a partir de estas bases de datos se ven en la Tabla 44. De dicha tabla el factor de carga para el camión serie 7832 es pronosticado con 60%, y para el camión serie 7868 el factor de carga es pronosticado con 66%.

La Tabla 43 muestra que las bases de datos de operaciones de camión serie 7832 y camión serie 7868 muestran un factor de carga elevado para la operación del motor en aplicaciones de traslado de mineral, lo cual refleja el nivel de exigencia al que están sometidos estos equipos en operación minera, concordantes con los estudios previos reportados por Klanfar et al.(2016) (ver Tabla 1, Tabla 2 y Tabla 3).

Tabla 44

Pronóstico de factor de carga en operación de camiones mineros.

Base de datos	Descripción	Potencia pronosticada (kW)	Potencia nominal (kW)	Factor de carga pronosticado (desde la base de datos)	Factor de carga pronosticado en el ciclo de trabajo
DML-20090708-171744-HD785-7-sn7832	Registro de medio ciclo de trabajo Camión sin carga descendiendo	259,43	895	0,29 (1er medio ciclo)	0,60
DML-20090708-173000-HD785-7-sn7832	Registro de medio ciclo de trabajo Camión cargado ascendiendo	816,50	895	0,91 (2do medio ciclo)	
DML-20090911-133436-HD785-7-sn7868	Registro de un ciclo de operación de traslado a nivel	593,28	895	0,66 (ciclo completo)	0,66

Fuente: Elaboración propia.

4.6. Contrastación de hipótesis

Para la contrastación de las hipótesis, se recalcan los resultados alcanzados en este trabajo,

Para la hipótesis específica 1, se tiene:

H.E.1: Modelos no lineales de aprendizaje supervisado logran la Pronóstico del desempeño instantáneo de motores de combustión interna en aplicaciones industriales.

Para la contratación de esta hipótesis los resultados alcanzados en la sección 4.3, demuestran que el modelo lineal basado en Regresión lineal múltiple (MLR), no logra cubrir las exigencias de validación del modelo, sin embargo, en esta misma sección se puede observar que los modelos basados en Random Forest, y Redes neuronales y XGBoost, muestran mejores resultados. Por lo tanto, se acepta esta hipótesis.

Para la hipótesis específica 2, se tiene:

H.E.2: Los parámetros asociados la masa de aire y a la masa de combustible son relevantes en el modelo de Pronóstico del desempeño instantáneo de motores de combustión interna en aplicaciones industriales

Para la contratación de esta hipótesis, se ejecuta un análisis Importancia de cada característica, para el modelo no lineal XGBoost, mostrados en la sección 4.3; determinando que la presión de aire de admisión y la presión de riel comandada y la presión de riel sensada (presiones de combustible) son los parámetros más relevantes en el modelamiento del Torque, de la Potencia y del consumo de combustible. Por lo tanto, se acepta esta hipótesis.

Para la hipótesis específica 3, se tiene:

H.E.3: Un modelo no lineal que logra determinar la potencia pronosticada logra la Pronóstico del factor de carga de motores de combustión interna en aplicaciones industriales.

Para la contratación de esta hipótesis, se ejecuta un análisis mostrándose los resultados en la sección 4.5, donde se logra determinar el factor de carga de una aplicación real, basada en el pronóstico del modelo XGBoost. Por lo tanto, se acepta esta hipótesis.

Finalmente, para la hipótesis general, se tiene:

H.G: Un modelo de pronóstico basado en machine learning logrará la Pronóstico del desempeño instantáneo de un motor de combustión interna de aplicación industrial

Para la contratación de esta hipótesis, se analiza los resultados en la sección 4.3 de este trabajo, donde se muestra que el modelo basado en XGboost logra la perdición del desempeño instantáneo del motor, en sus parámetros Torque pronosticado, Potencia

pronosticada, Consumo de combustible pronosticado, y finalmente con el análisis de la sección 4.5 se determina el pronóstico del Factor de carga. Por lo tanto, se acepta esta hipótesis

4.7. Discusión de resultados

Entre los principales hallazgos que se alcanzan al cerrar esta investigación, son referente a la determinación del modelo de Pronóstico basado en XGboost, los parámetros de entrada relevantes para el modelo, la determinación del torque como parámetro de salida y no de entrada, la aplicación práctica del modelo en la determinación del factor de carga durante la operación del motor. Pasándose a detallar cada una de estas aseveraciones.

Las predicciones de los modelos desarrollados fueron evaluadas mediante parámetros estadísticos de error: MAE, MSE, RMSE, el coeficiente de determinación R^2 , el score de los conjuntos de entrenamiento, el score de los conjuntos de prueba, así como la distribución de los residuales generados por cada modelo. Con base en estos, se determinó que el modelo basado en XG-Boost es el más adecuado para este problema de Pronóstico, como se explica en la sección 4.3 de este trabajo. Este modelo ha sido poco explorado en la literatura especializada sobre esta temática, donde principalmente se han desarrollado modelos basados en algoritmos de redes neuronales artificiales.

Durante el proceso de investigación, fue necesario identificar qué parámetros de entrada son los más convenientes para la Pronóstico de los parámetros de salida relacionados con el comportamiento del motor de combustión interna. Principalmente, la presión del múltiple de admisión y, posteriormente, la presión del riel de combustible fueron los más relevantes; esto se puede observar en la sección 4.3 de este documento. Esta información constituye un aporte significativo de este trabajo, ya que no ha sido detallada en la literatura técnica existente. Su explicación está ligada a que el flujo masico y el flujo de combustible son necesarios para la ejecución de la combustión, en el proceso de transformación de la energía química del combustible en trabajo mecánico sobre la cabeza del pistón, y finalmente relacionados con los parámetros de salida como el torque y potencia al freno, medidos en el dinamómetro.

Respecto a la determinación del torque como parámetro de salida, En la literatura revisada, muchos estudios sobre la Pronóstico del comportamiento del motor y la concentración de emisiones parten del nivel de carga o torque como parámetro de entrada. Sin embargo, en este trabajo se ha demostrado que este parámetro puede considerarse como un parámetro de salida, ya que es una respuesta del motor y se evidencia a través de otros parámetros de entrada, como se puede observar en la sección 4.3.

Respecto al factor de carga, el pronóstico del comportamiento del motor, registrado en los parámetros de salida de potencia, torque y consumo de combustible, fue relevante para el cálculo del factor de carga durante las pruebas dinamométricas. De tal forma que abre una metodología para su cálculo, sin el uso de instrumentos invasivos en el equipo a monitorear (como es el caso actual del uso de caudalímetros de combustible por un periodo determinado). La validación de este modelo fue sometida a prueba utilizando bases de datos reales en condiciones de operación severa de un equipo propulsado con el mismo modelo de motor tomado como muestra, confirmando resultados similares a los reportados en análisis previos, como los de Klanfar et al. (2016).

En sustentación y contraste con los antecedentes investigativos, los resultados encontrados en este trabajo de investigación confirman la utilidad de los modelos de inteligencia artificial mediante el empleo de machine learning en la Pronóstico del consumo de combustible, tal como previamente han ejecutado los trabajos previos de Shateri et al. (2024), sin embargo se contrapone con los resultados alcanzados, mientras que este trabajo apunta a que los modelo por Redes neuronales y XGBoost, son los más adecuados (resultados en la Tabla 37), para Shateri et al. (2024) los modelos basados regresión de procesos gaussianos (GPR) supera a los demás en términos de precisión, corroborados mediante simulación en la dinámica de fluidos computacional (CFD), GPR alcanza un MAE de 0,05124, MSE de 0,06514, coeficiente de Pearson (PC) de 0,99992 y R^2 de 0,99982, mientras que RFR y redes neuronales (NN) exhiben valores ligeramente más altos para estas métricas

Adicionalmente, en sustentación y contraste con los antecedentes investigativos, el desarrollo de este trabajo tiene concordancia con los objetivos propuestos por Kocak et al. (2023) en el empleo de datos recopilados por sensores para diagnosticar el desempeño del motor de equipo minero como es el caso de esta investigación o en la aplicación de embarcaciones marítimas como lo propone Kocak et al. (2023), el resultado de ambos estudios arrojan importancia y grado de precisión en las respuestas de los modelos de inteligencia artificial empleados. En contraste, para Kocak et al. (2023) los Árboles de decisión ($R^2 > 0.999$) fueron los más adecuados, superando a SVM ($R^2 > 0.98$) y regresión lineal ($R^2 > 0.98$), los que también obtuvieron resultados altos pero con menor eficiencia en tiempo de ejecución.

Respecto a los trabajos de Ricci et al. (2023) que busca predecir el torque desarrollado por un motor de combustión interna de aplicación automotriz, en donde se declara que el enfoque de "aprendizaje automático" se está convirtiendo en uno de los enfoques más utilizados para realizar actividades de pronóstico con un esfuerzo computacional reducido, debido tanto a su rentabilidad como a su estructura simple y compacta; declaración que

concuera con los resultados alcanzados en este trabajo empleando el modelo XGBoost, cuyos resultados se muestran en la Tabla 27. Sin embargo, en constaste el modelo determinado por Ricci et al. (2023) son los modelos de redes neuronales artificiales.

Este trabajo muestra el pronóstico de los parámetros del comportamiento del motor de combustión interna, que según la clasificación de Ineza et al. (2023) corresponde al grupo de trabajos que abordan el diagnóstico del motor, puesto que trabaja con parámetros de entrada y pronostica los parámetros de salida, los cuales son presentados como valores esperados durante el funcionamiento del motor, cualquiera de los datos fuera de las lecturas iniciales de los parámetros de entrada o de los pronósticos ejecutadas en los parámetros de salida, evidencian un estado de anormalidad en el diagnóstico del motor diésel empelado en la propulsión de las máquina de operación minera.

La metodología empleada en este trabajo refuerza la postura de Pajak et al. (2023) La de generar abundantes datos de operación de un equipo para luego ejecutar análisis mediante aprendizaje automático y establecer relación en las salida objetivo del estudio, la contraparte entre estos dos trabajos es que en esta investigación se usan técnicas de regresión para la pronóstico de las respuesta en los parámetros de salida del motor de combustión interna, mientras que Pajak et al. (2023) emplea la data para afrontar problemas de clasificación.

Este trabajo de investigación guarda correspondencia con los resultados por Şahin (2023) quien demuestra que los modelos basados en inteligencia artificial tiene importante aplicación en el comportamiento del motor de combustión interna, llegando a indicar que los algoritmos basados en Redes neuronales son los más precisos para pronosticar el consumo específico de combustible, y la temperatura de gases de escape, dejando a modelos basados en XGBoos los más precisitos para el pronósticos de las emisiones NO, CO₂ y HC. Estos resultados concuerdan con los alcanzados en este trabajo que muestra la preponderancia d ellos modelos de XGBoos y de Redes neuronales artificiales.

Este trabajo guarda concordancia con los objetivos de Yang et al. (2022), quién establece la potencia de salida de un motor de combustión interna a través del pronóstico de la presión media efectiva por medio de algoritmos de machine learning, resultandos ventajosos en esta tarea los modelos basados en redes neuronales artificiales (ANR) y máquinas de vectores soporte en regresión (SVM). Mientras que en este trabajo el pronóstico por modelos en XGBoost resulta la más adecuada en función de sus parámetros de entrada, frente al SVM.

Al igual que el presente trabajo que emplea técnicas de machine learning para la producción de sus parámetros de salida, los trabajo de Liu et al. (2018) presentan un estudio cuyo objetivo es desarrollar un modelo de red neuronal artificial (ANN) para predecir con éxito

el rendimiento del motor y las emisiones de escape cuando un motor de encendido por chispa. Estos trabajos demuestran la versatilidad de aplicaciones del machine learning en el comportamiento de motores de combusto interna.

El impacto de la aplicación de este trabajo guarda concordancia con los objetivos de Jassim et al. (2017), quien presentan un estudio cuyo objetivo es proponer un modelo que utiliza una red neuronal artificial (ANN) para predecir el consumo de energía por hora de una excavadora y las emisiones de CO₂ en diferentes condiciones del sitio de operación, mientras que en este trabajo de investigación se ejecuta corroboración de los modelos desarrollados en base de algoritmos de XGBoost en equipo mineros en operación de carguío, frente a los resultados d eso modelos de redes neuronales artificiales (ANN).

Conclusiones

Al finalizar la ejecución de este trabajo de investigación, se ha llegado a las siguientes conclusiones.

Respecto al primer objetivo específico: Desarrollar y evaluar diversos modelos de aprendizaje supervisado aplicables a la de pronóstico del desempeño instantáneo de motores de combustión interna en aplicaciones industriales, se ha demostrado que los modelos no lineales son los más adecuados en el proceso de pronóstico del desempeño del motor de combustión, por medio de la evaluación de métricas de desempeño, tomando en cuenta el mayor coeficiente de determinación (R^2) y la menor raíz del error cuadrático medio (RMSE), así como el valor del score de los conjuntos entrenados y el score de los conjuntos testeados, mostrados en las Tabla 14, Tabla 22, Tabla 24, Tabla 27, Tabla 30, Tabla 36, y Tabla 38, para los modelos MLR, KNN, SVM,DT, RF, ANR y XGBoost respectivamente. Cuyo análisis alcanza la evaluación de los residuos del modelo de pronóstico para datos estandarizados, donde los residuos tienen mayor variación en el pronóstico del torque, y en función de ellos los más convenientes para su empleo son los modelos Random Forest, Redes neuronales artificiales y XGBoost.

Respecto al segundo objetivo específico: Evidenciar los principales parámetros del motor que sean relevantes en el modelo de pronóstico del desempeño instantáneo de motores de combustión interna en aplicaciones industriales. el análisis ejecutado en este trabajo cumple su propósito al identificar los parámetros de lectura que son relevantes para el comportamiento de un motor diésel equipado con un sistema de inyección electrónica de riel común (*common rail*). Por lo que, en función del modelo XGboost se analiza la preponderancia de los parámetros de entrada estandarizados por lo cual la importancia de las características se puede ver en la Figura 86 para el caso del torque, Figura 87 para el caso de la potencia, Figura 88 para el caso del consumo de combustible. Donde se muestra a la Presión del aire de admisión, como uno de los factores más preponderantes. Seguido de la presión de la presión sensada (para el caso del torque), y presión de riel comandada. (para el caso de la potencia y el consumo de combustible).

Respecto al tercer objetivo específico: establecer el pronóstico del factor de carga de motores de combustión interna en aplicaciones industriales, esta investigación concluye que el modelo basado en XGBoost logra el mejor rendimiento en el pronóstico del factor de carga del motor de combustión interna entre los modelos desarrollados, lo que lo convierte en la opción más adecuada para el cálculo del factor de carga. Esto queda evidenciado con los resultados de la Tabla 39, que frente a un factor de carga de 41,43% de las pruebas dinamométricas, el valor pronosticado oscila entre 41,44% hasta 43,79%. Además, el

algoritmo fue puesto a pruebas en datos reales de operación alcanzan 60% hasta 66% vistos en la Tabla 40. Valores habituales para la aplicación industrial de propulsión de un camión minero, concordantes con las Tabla 1, Tabla 2 y Tabla 3.

En relación con el objetivo general, este trabajo demuestra que el modelo basado en XGBoost es el más adecuado para la pronóstico desempeño instantáneo en motores de combustión interna de aplicación industrial, Esto queda evidenciado por la evaluación de las métricas de desempeño, valor del score de los conjuntos entrenados y el score de los conjuntos testeados, y la evaluación de los residuos del modelo de pronóstico para datos estandarizados, y la evaluación de los tiempo de ejecución del modelo (Ellos detallados en la sección 4.3 de este trabajo. Así el análisis de los Errores MAE MSE RMSE y R^2 reportados en las Tablas 39, Tabla 40, y Tabla 41, Donde, para el caso del torque (Tabla 38), el menor valor de los errores MAE, MSE y RMSE se encuentra en el modelo XGBoost, donde el RMSE es de 0,0921y el R^2 alcanza 0,9912, Para el caso de la potencia (Tabla 39), el valor de los errores MAE, MSE y RMSE mostrados en el modelo XGBoost, son RMSE es de 0,0978 y el R^2 alcanza 0,9902.. Para el caso del consumo de combustible (Tabla 40) el valor de los errores MAE, MSE y RMSE del modelo XGBoost, es RMSE es de 0,0825 y el R^2 alcanza 0,9930. De igual manera es favorable la la evaluación de los residuos del modelo de pronóstico mostrados en la Figura 84, el tiempo de ejecución del pronóstico, mostrados en la Tabla 41.

Finalmente, el trabajo responde a la pregunta de investigación, corroborando la hipótesis al establecer que un modelo de pronóstico basado en machine learning es capaz de predecir el desempeño instantáneo de motores de combustión interna en aplicaciones industriales, cuya contratación se observa en a la sección 4.6

Recomendaciones

Establecidas las conclusiones de la presente investigación, se proponen las siguientes recomendaciones:

A los investigadores en inteligencia artificial, se les sugiere evaluar otros modelos de pronóstico para analizar su desempeño en la aplicación del comportamiento del motor y la determinación del factor de carga. Para ello, pueden utilizar la base de datos suministrada en este trabajo o desarrollar su propia base de datos mediante el uso del software Insite y la interfaz Inline, como se realizó en esta investigación.

Asimismo, se recomienda a los investigadores en inteligencia artificial que evalúen el desempeño de algoritmos de redes neuronales y de XGBoost en otras características del motor diésel, tales como la cuantificación de la emisión de gases contaminantes, la temperatura de los gases y la opacidad, en el caso de motores diésel.

También se aconseja a los investigadores en inteligencia artificial evaluar el desempeño de los algoritmos analizados en esta tesis en el comportamiento de motores de encendido por chispa en aplicaciones automotrices, siendo recomendable el uso de bases de datos con registros abundantes.

A los ingenieros de diseño, se les recomienda habilitar un monitor de registro del factor de carga en la memoria del controlador del motor, con el fin de pronosticar el nivel de esfuerzo al que está sometido. Esto permitirá ejecutar pronósticos de operación, mantenimiento y determinación de la vida útil del motor de combustión.

A los responsables de la operación de equipos, se les sugiere aplicar el modelo de pronóstico en otros equipos que utilicen el mismo motor ensayado, cuantificando los registros del factor de carga mensual, semanal, diario, por turnos, por ciclo de operación o de manera instantánea. Además, se recomienda incorporar estos parámetros en las estrategias de mantenimiento predictivo.

Referencias bibliográficas

- Abed, K. A., Gad, M. S., El Morsi, A. K., Sayed, M. M., & Elyazeed, S. A. (2019). Effect of biodiesel fuels on diesel engine emissions. *Egyptian Journal of Petroleum*, 28(2). <https://doi.org/10.1016/j.ejpe.2019.03.001>
- Aliramezani, M., Koch, C. R., & Shahbakhti, M. (2022). Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions. In *Progress in Energy and Combustion Science* (Vol. 88). Elsevier Ltd. <https://doi.org/10.1016/j.pecs.2021.100967>
- Alpaydin, E. (2014). Introduction to Machine Learning Ethem Alpaydin. *Introduction to Machine Learning, Third Edition*.
- Billerot, P. Lou, Tétrault, P., Lemaire, R., & Seers, P. (2023). Parametric study of the impact of EGR and fuel properties on diesel engine performance using a predictive thermodynamic model. *International Journal of Engine Research*, 24(4). <https://doi.org/10.1177/14680874221087958>
- Bishop, C. (2007). Pattern Recognition and Machine Learning. *Journal of Electronic Imaging*, 16(4). <https://doi.org/10.1117/1.2819119>
- Bobadilla, J. (2021). Machine Learning y deep learning usando python, scikit y keras. In *Editorial Ra-ma (España)*. Ra-Ma.
- Burkov, A. (2019). The Hundred Page Machine Learning book. In *Book* (Issue 2).
- Carreño, R. (2016). *A comprehensive methodology to analyse the Global Energy Balance in Reciprocating Internal Combustion Engines* [Tesis Doctoral]. Universitat Politècnica de Valencia.
- Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408. <https://doi.org/10.1016/j.neucom.2019.10.118>
- Chen, L., Li, T., Yin, T., & Zheng, B. (2014). A predictive model for knock onset in spark-ignition engines with cooled EGR. *Energy Conversion and Management*, 87, 946–955. <https://doi.org/10.1016/J.ENCONMAN.2014.08.002>
- Chollet, F. (2018). Deep learning with Python. In *Manning* (Issue 9). Manning Shelter island.

- Contreras, W. R., León, R. S., & Maldonado, J. L. (2019). Predicción de emisiones de CO y HC en motores Otto mediante redes neuronales. *Ingenius*, 23. <https://doi.org/10.17163/ings.n23.2020.03>
- Cummins inc. (2012). *Adaptadores de enlace de datos Inline para motores*. <https://www.cummins.com/es/support/digital-products-and-services-support/inline-support>
- Datta, A., & Mandal, B. K. (2018). Numerical prediction of the performance, combustion and emission characteristics of a CI engine using different biodiesels. *Clean Technologies and Environmental Policy*, 20(8). <https://doi.org/10.1007/s10098-018-1567-6>
- Fayyazbakhsh, A., Bell, M. L., Zhu, X., Mei, X., Koutný, M., Hajinajaf, N., & Zhang, Y. (2022). Engine emissions with air pollutants and greenhouse gases and their control technologies. In *Journal of Cleaner Production* (Vol. 376). <https://doi.org/10.1016/j.jclepro.2022.134260>
- Feng, S., Xu, S., Yuan, P., Xing, Y., Shen, B., Li, Z., Zhang, C., Wang, X., Wang, Z., Ma, J., & Kong, W. (2022). The Impact of Alternative Fuels on Ship Engine Emissions and Aftertreatment Systems: A Review. In *Catalysts* (Vol. 12, Issue 2). <https://doi.org/10.3390/catal12020138>
- Feng, Y., Wang, H., Gao, R., & Zhu, Y. (2019). A zero-dimensional mixing controlled combustion model for real time performance simulation of marine two-stroke diesel engines. *Energies*, 12(10). <https://doi.org/10.3390/en12102000>
- Fiebig, M., Wiartalla, A., Holderbaum, B., & Kiesow, S. (2014). Particulate emissions from diesel engines: Correlation between engine technology and emissions. In *Journal of Occupational Medicine and Toxicology* (Vol. 9, Issue 1). <https://doi.org/10.1186/1745-6673-9-6>
- Gao, J., Tian, G., Sornioti, A., Karci, A. E., & Di Palo, R. (2019). Review of thermal management of catalytic converters to decrease engine emissions during cold start and warm up. In *Applied Thermal Engineering* (Vol. 147). <https://doi.org/10.1016/j.applthermaleng.2018.10.037>
- Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems. In *O'Reilly Media*.
- Gogoi, T. K., & Baruah, D. C. (2010). A cycle simulation model for predicting the performance of a diesel engine fuelled by diesel and biodiesel blends. *Energy*, 35(3). <https://doi.org/10.1016/j.energy.2009.11.014>

- Greim, H. (2019). Diesel engine emissions: are they no longer tolerable? In *Archives of Toxicology* (Vol. 93, Issue 9). <https://doi.org/10.1007/s00204-019-02531-5>
- Gutiérrez, R. H. R., Monteiro, U. A., & Vaz, L. A. (2020). Predictive thermodynamic model of the performance of a stationary spark-ignition engine running on natural gas. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 42(8). <https://doi.org/10.1007/s40430-020-02496-y>
- Harris, H. D., & Pearce, F. (1990). A universal mathematical model of diesel engine performance. *Journal of Agricultural Engineering Research*, 47(C). [https://doi.org/10.1016/0021-8634\(90\)80038-V](https://doi.org/10.1016/0021-8634(90)80038-V)
- Haykin, S. (2008). Neural Networks and Learning Machines. In *Pearson Prentice Hall New Jersey USA 936 pLinks* (Vol. 3). <https://doi.org/978-0131471399>
- Hernández, R., & Mendoza, C. P. (2018). *Metodología de la investigación: las rutas cuantitativa, cualitativa y mixta*. McGraw-Hill Education.
- Heywood, J. B. (2018). Internal Combustion Engine Fundamentals. In *European University Institute* (2nd ed., Issue 2). McGraw-Hill Education.
- IBM. (2019). ¿Qué es Machine Learning? ¿Que Es Machine Learning? <https://www.ibm.com/mx-es/topics/machine-learning?>
- IBM. (2023). ¿Qué es un algoritmo de machine learning? <https://www.ibm.com/es-es/topics/machine-learning-algorithms>
- Ineza, L., Liu, B., Liu, F., Zhang, J., Li, B., & Wan, P. (2023). Review of Artificial Intelligent Algorithms for Engine Performance, Control, and Diagnosis. In *Energies* (Vol. 16, Issue 3). MDPI. <https://doi.org/10.3390/en16031206>
- Izaurieta, F., & Saavedra, C. (1999). Redes Neuronales Artificiales. *Charlas de Física*, 1–15. [https://doi.org/10.1016/S0210-5691\(05\)74198-X](https://doi.org/10.1016/S0210-5691(05)74198-X)
- Jassim, H. S. H., Lu, W., & Olofsson, T. (2017). Predicting energy consumption and CO2 emissions of excavators in earthwork operations: An artificial neural network model. *Sustainability (Switzerland)*, 9(7). <https://doi.org/10.3390/su9071257>
- Jolliffe, I. T. (2002). Principal Component Analysis, Second Edition. *Encyclopedia of Statistics in Behavioral Science*, 30(3). <https://doi.org/10.2307/1270093>
- Kecojevic, V., & Komljenovic, D. (2011a). Haul truck fuel consumption and CO2 emission under various engine load conditions. *SME Annual Meeting and Exhibit and CMA 113th National Western Mining Conference 2011*, 7–10.

- Kecojevic, V., & Komljenovic, D. (2011b). Impact of bulldozer's engine load factor on fuel consumption, CO₂ emission and cost. *American Journal of Environmental Sciences*, 7(2), 125–131. <https://doi.org/10.3844/ajessp.2011.125.131>
- Khurana, S., Saxena, S., Jain, S., & Dixit, A. (2020). Predictive modeling of engine emissions using machine learning: A review. *Materials Today: Proceedings*, 38, 280–284. <https://doi.org/10.1016/j.matpr.2020.07.204>
- Klanfar, M., Korman, T., & Kujundžić, T. (2016). Fuel consumption and engine load factors of equipment in quarrying of crushed stone. *Tehnicki Vjesnik*, 23(1). <https://doi.org/10.17559/TV-20141027115647>
- Kocak, G., Gokcek, V., & Genc, Y. (2023). Condition monitoring and fault diagnosis of a marine diesel engine with machine learning techniques. *Pomorstvo*, 37(1), 32–46. <https://doi.org/10.31217/p.37.1.4>
- Krebs, S., & Biet, C. (2021). Predictive model of a premixed, lean hydrogen combustion for internal combustion engines. *Transportation Engineering*, 5. <https://doi.org/10.1016/j.treng.2021.100086>
- Lapuerta, M., Armas, O., & Rodríguez-Fernández, J. (2008). Effect of biodiesel fuels on diesel engine emissions. In *Progress in Energy and Combustion Science* (Vol. 34, Issue 2). <https://doi.org/10.1016/j.pecs.2007.07.001>
- Liu, J., Huang, Q., Ulishney, C., & Dumitrescu, C. E. (2022). Comparison of Random Forest and Neural Network in Modeling the Performance and Emissions of a Natural Gas Spark Ignition Engine. *Journal of Energy Resources Technology, Transactions of the ASME*, 144(3). <https://doi.org/10.1115/1.4053301>
- Liu, Z., Zuo, Q., Wu, G., & Li, Y. (2018). An artificial neural network developed for predicting of performance and emissions of a spark ignition engine fueled with butanol–gasoline blends. *Advances in Mechanical Engineering*, 10(1). <https://doi.org/10.1177/1687814017748438>
- Mahesh, B. (2020). Machine Learning Algorithms - A Review. *International Journal of Science and Research*, 9(1).
- Manrique, E. (2020). Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo. *Revista Ibérica de Sistemas e Tecnologías de Informação*.
- Martyr, A. J., & Plint, M. A. (2012). Engine Testing: The Design, Building, Modification and Use of Powertrain Test Facilities, Fourth Edition. In *Engine Testing: The Design, Building,*

Modification and Use of Powertrain Test Facilities, Fourth Edition.
<https://doi.org/10.1016/C2010-0-66322-X>

- Ma, Z., Yang, Y., Sun, P., Xing, H., Duan, S., Qu, H., & Zou, Y. (2021). Analysis of marine diesel engine emission characteristics of different power ranges in china. *Atmosphere*, 12(9). <https://doi.org/10.3390/atmos12091108>
- Mueller, J. P., & Massaron, L. (2021). Machine Learning for dummies, 2nd Edition. In *John Wiley & Sons, Inc.*
- Müller, A. C., & Guido, S. (2017). Introduction to Machine Learning with Python: a guide for data scientist. *O'Reilly Media, Inc.*
- Najafi, B., Akbarian, E., Lashkarpour, S. M., Aghbashlo, M., Ghaziaskar, H. S., & Tabatabaei, M. (2019). Modeling of a dual fueled diesel engine operated by a novel fuel containing glycerol triacetate additive and biodiesel using artificial neural network tuned by genetic algorithm to reduce engine emissions. *Energy*, 168. <https://doi.org/10.1016/j.energy.2018.11.142>
- Narendranathan, S. K., Thiyagu, M., Shankar., A., & Sureshkumar, T. N. (2022). Implementation of ANN for Predication of Performance and Emission Characteristics in CI Engine. *Proceedings of the 2022 International Conference on Electronic Systems and Intelligent Computing, ICESIC 2022*, 264–269. <https://doi.org/10.1109/ICESIC53714.2022.9783517>
- Nezlioui, F., Benslimane, A., Hamtache, B., Sahi, A., Lounici, M. S., & Sadaoui, D. (2023). Development of a Thermodynamic Model for Supercharged Diesel Engine Performance and Combustion Characteristics Study. *Combustion Science and Technology*. <https://doi.org/10.1080/00102202.2023.2285407>
- Obert, E. F. (1997). *Motores de combustión interna: análisis y aplicaciones*. C.E.C.S.A.
- Pająk, M., Kluczyk, M., Muślewski, Ł., Lisjak, D., & Kolar, D. (2023). Ship Diesel Engine Fault Diagnosis Using Data Science and Machine Learning. *Electronics (Switzerland)*, 12(18). <https://doi.org/10.3390/electronics12183860>
- Paluszek, M., & Thomas, S. (2017). MATLAB Machine Learning. In *MATLAB Machine Learning*. <https://doi.org/10.1007/978-1-4842-2250-8>
- Ponce, P. (2010). *Inteligencia Artificial con aplicaciones a la Ingeniería*. Alfa y Omega.
- Raschka, S., & Mirjalili, V. (2019). *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow (snd)*.

- Reşitoğlu, I. A., Altinişik, K., & Keskin, A. (2015). The pollutant emissions from diesel-engine vehicles and exhaust aftertreatment systems. In *Clean Technologies and Environmental Policy* (Vol. 17, Issue 1). <https://doi.org/10.1007/s10098-014-0793-9>
- Ricci, F., Petrucci, L., Mariani, F., & Grimaldi, C. N. (2023). NARX Technique to Predict Torque in Internal Combustion Engines. *Information (Switzerland)*, 14(7). <https://doi.org/10.3390/info14070417>
- Rovira, A., & Muñoz, M. (2015). *Motores de Combustión Interna*. Universidad Nacional de Educación a Distancia.
- Russel, S., & Norvig, P. (2012). Artificial intelligence, a modern approach. In *The Knowledge Engineering Review* (3ed ed.). <https://doi.org/10.1017/S0269888900007724>
- Şahin, S. (2023). Comparison of machine learning algorithms for predicting diesel/biodiesel/iso-pentanol blend engine performance and emissions. *Heliyon*, 9(11). <https://doi.org/10.1016/j.heliyon.2023.e21365>
- Schonlau, M., & Zou, R. Y. (2020). The random forest algorithm for statistical learning. *Stata Journal*, 20(1). <https://doi.org/10.1177/1536867X20909688>
- Sebayang, A. H., Milano, J., Shamsuddin, A. H., Alfansuri, M., Silitonga, A. S., Kusumo, F., Prahmana, R. A., Fayaz, H., & Zamri, M. F. M. A. (2022). Modelling and prediction approach for engine performance and exhaust emission based on artificial intelligence of sterculia foetida biodiesel. *Energy Reports*, 8. <https://doi.org/10.1016/j.egy.2022.06.052>
- Shateri, A., Yang, Z., & Xie, J. (2024). Utilizing Artificial intelligence to identify an Optimal Machine learning model for predicting fuel consumption in Diesel engines. *Energy and AI*, 16. <https://doi.org/10.1016/j.egyai.2024.100360>
- Simplilearn. (2023). *What is XGBoost? An Introduction to XGBoost Algorithm in Machine Learning*. <https://www.simplilearn.com/what-is-xgboost-algorithm-in-machine-learning-article>
- Sun, X., Liang, X., Shu, G., Wang, Y., Wang, Y., & Yu, H. (2017). Effect of different combustion models and alternative fuels on two-stroke marine diesel engine performance. *Applied Thermal Engineering*, 115. <https://doi.org/10.1016/j.applthermaleng.2016.12.093>
- Tan, D., Tan, J., Peng, D., Fu, M., Zhang, H., Yin, H., & Ding, Y. (2021). Study on real-world power-based emission factors from typical construction machinery. *Science of the Total Environment*, 799. <https://doi.org/10.1016/j.scitotenv.2021.149436>

- Unpingco, J. (2022). Python for Probability, Statistics, and Machine Learning, Third Edition. In *Python for Probability, Statistics, and Machine Learning, Third Edition*. <https://doi.org/10.1007/978-3-031-04648-3>
- Urroz, J. C., Diéguez, P. M., Arzamendi, G., Arana, M., & Gandía, L. M. (2023). Gaseous fueling of an adapted commercial automotive spark-ignition engine: Simplified thermodynamic modeling and experimental study running on hydrogen, methane, carbon monoxide and their mixtures. *Fuel*, 337. <https://doi.org/10.1016/j.fuel.2022.127178>
- Wang, P., Zhu, C., & Gao, J. (2019). Feedforward Model Predictive Control of Fuel-Air Ratio for Lean-Burn Spark-Ignition Gasoline Engines of Passenger Cars. *IEEE Access*, 7. <https://doi.org/10.1109/ACCESS.2019.2919148>
- Wiederhold, G., & McCarthy, J. (2010). Arthur Samuel: Pioneer in Machine Learning. *IBM Journal of Research and Development*, 36(3). <https://doi.org/10.1147/rd.363.0329>
- Wong, K. I., Wong, P. K., Cheung, C. S., & Vong, C. M. (2013). Modeling and optimization of biodiesel engine performance using advanced machine learning methods. *Energy*, 55, 519–528. <https://doi.org/10.1016/j.energy.2013.03.057>
- Yadav, J., Deppenkemper, K., & Pischinger, S. (2023). Impact of renewable fuels on heavy-duty engine performance and emissions. *Energy Reports*, 9. <https://doi.org/10.1016/j.egyr.2023.01.016>
- Yang, R., Xie, T., & Liu, Z. (2022). The Application of Machine Learning Methods to Predict the Power Output of Internal Combustion Engines. *Energies*, 15(9). <https://doi.org/10.3390/en15093242>
- Zeng, W., Khalid, M. A. S., Han, X., & Tjong, J. (2020). A Study on Extreme Learning Machine for Gasoline Engine Torque Prediction. *IEEE Access*, 8, 104762–104774. <https://doi.org/10.1109/ACCESS.2020.3000152>
- Zhang, P., Jia, Y., & Shang, Y. (2022). Research and application of XGBoost in imbalanced data. *International Journal of Distributed Sensor Networks*, 18(6). <https://doi.org/10.1177/15501329221106935>
- Zhang, Y., Wang, Q., Chen, X., Yan, Y., Yang, R., Liu, Z., & Fu, J. (2022). The Prediction of Spark-Ignition Engine Performance and Emissions Based on the SVR Algorithm. *Processes*, 10(2). <https://doi.org/10.3390/pr10020312>

Anexos

Anexo I: Matriz de operacionalización de variables

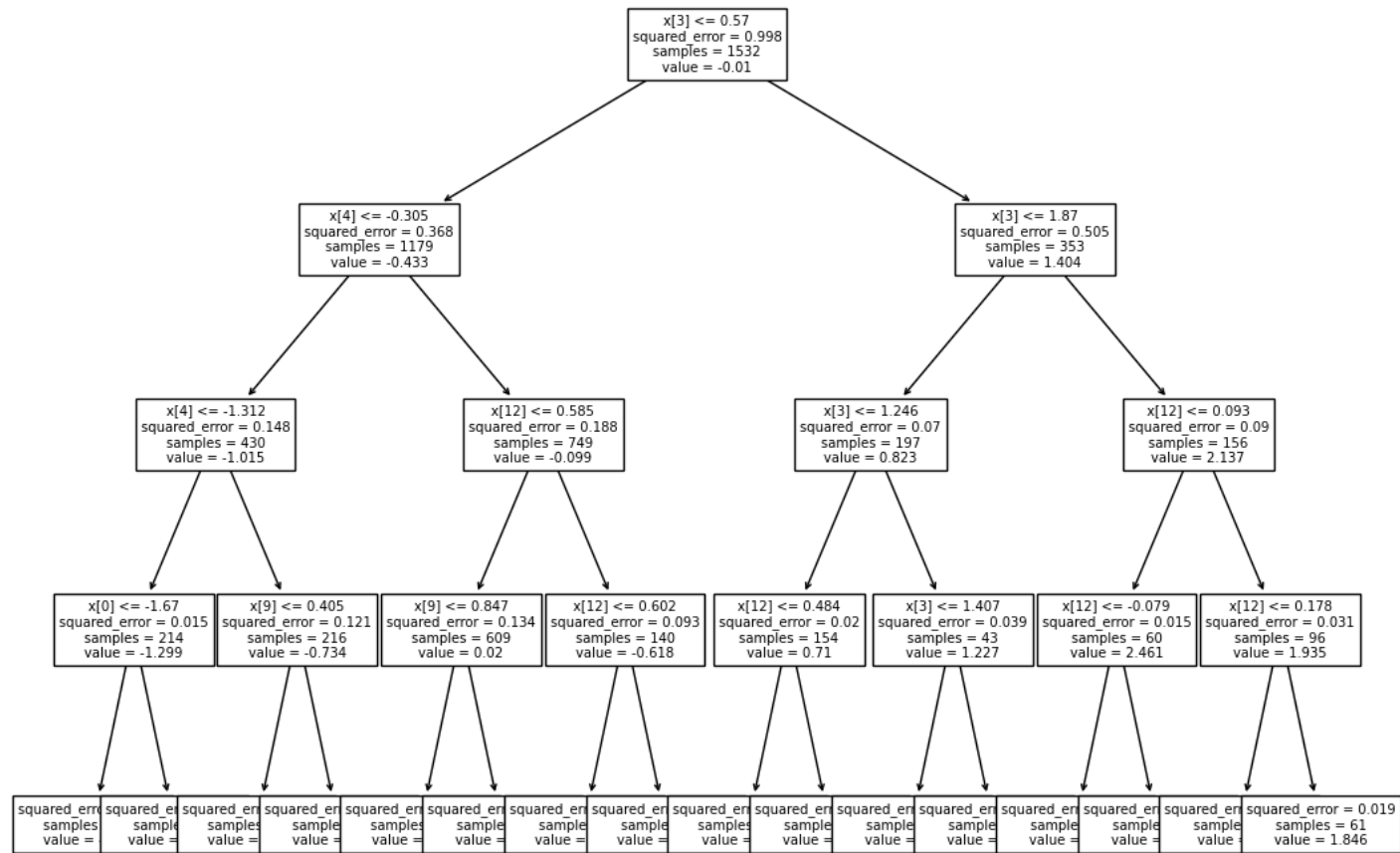
Variable	Definición Conceptual	Definición Operacional	Operacionalización			Escala de medición
			Dimensiones	Indicadores	Índices y/o ítems	
Variable 1: Desempeño pronosticado del motor de combustión interna	Comportamiento cuantificable de manera pronosticada de la operación del motor de combustión interna	El desempeño pronosticado queda definido por el pronóstico del torque, potencia, consumo de combustible y el cálculo del factor de carga	Torque pronosticado	Nivel de Torque	Torque	Nm
			Potencia pronosticada	Nivel de Potencia	Potencia	kW
			Consumo pronosticado	Nivel de Consumo	Consumo	L/hr
			Factor de carga pronosticado	Nivel de Factor de carga	Factor de carga	Adimensional
Variable 2:: Desempeño instantáneo del motor de combustión interna	Comportamiento medible de manera instantánea de la operación del motor de combustión interna	El desempeño instantáneo queda definido por la lectura del torque, potencia, consumo de combustible y el cálculo del factor de carga	Torque instantáneo	Nivel de Torque	Torque	Nm
			Potencia instantánea	Nivel de Potencia	Potencia	kW
			Consumo instantáneo	Nivel de Consumo	Consumo	L/hr
			Factor de carga instantáneo	Nivel de Factor de carga	Factor de carga	Adimensional

Fuente: Elaboración propia.

Anexo II: Matriz de Consistencia

Título: Modelo basado en Machine learning para la predicción del desempeño de motores de combustión interna en aplicaciones industriales				
Problemas	Objetivos	Hipótesis	Variables y dimensiones	Metodología
Problema general ¿De qué manera un modelo de pronóstico basado en machine learning logra la predicción del desempeño instantáneo de motores de combustión interna de aplicación industrial? Problemas específicos: • ¿Qué modelos de aprendizaje supervisado son adecuados en la de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales? • ¿Cuáles son los principales parámetros del motor que son relevantes en el modelo de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales? • ¿De qué manera se logra la predicción del factor de carga de motores de combustión interna en aplicaciones industriales?	Objetivo general Desarrollar un modelo de pronóstico basado en machine learning que pueda predecir el desempeño instantáneo de motores de combustión interna en aplicaciones industriales. Objetivos específicos: • Desarrollar y evaluar diversos modelos de aprendizaje supervisado aplicables a la predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales • Evidenciar los principales parámetros del motor que sean relevantes en el modelo de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales • Establecer la predicción del factor de carga de motores de combustión interna en aplicaciones industriales	Hipótesis general Un modelo de pronóstico basado en machine learning logrará la predicción del desempeño instantáneo de un motor de combustión interna de aplicación industrial Hipótesis específicas: • Modelos no lineales de aprendizaje supervisado logran la predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales • Los parámetros asociados la masa de aire y a la masa de combustible son relevantes en el modelo de predicción del desempeño instantáneo de motores de combustión interna en aplicaciones industriales • El modelo que logra determinar la potencia pronosticada logra la predicción del factor de carga de motores de combustión interna en aplicaciones industriales	Variable 1: Desempeño pronosticado • Torque pronosticada • Potencia pronosticada • Consumo pronosticado • Factor de carga instantáneo Variable 2: Desempeño instantáneo • Torque instantáneo • Potencia instantánea • Consumo instantáneo • Factor de carga instantáneo	• Tipo de investigación: Aplicada • Nivel de investigación: Correlacional • Métodos: Hipotético-Deductivo • Diseño: No experimental • Población y muestra: Motor de combustión interna diésel turboalimentado de inyección directa common rail • Técnicas de recopilación de datos: observación. • Instrumento de recolección de datos: Registro csv por descarga de scanner • Técnicas de análisis y procesamiento de datos: predicción y análisis correlacional • Instrumentos de análisis y procesamiento de datos: hoja de datos, software de predicción, software de análisis estadístico.

Anexo III: Representación del modelo árbol de decisión desarrollado.



Anexo IV. Características del motor diésel ensayado

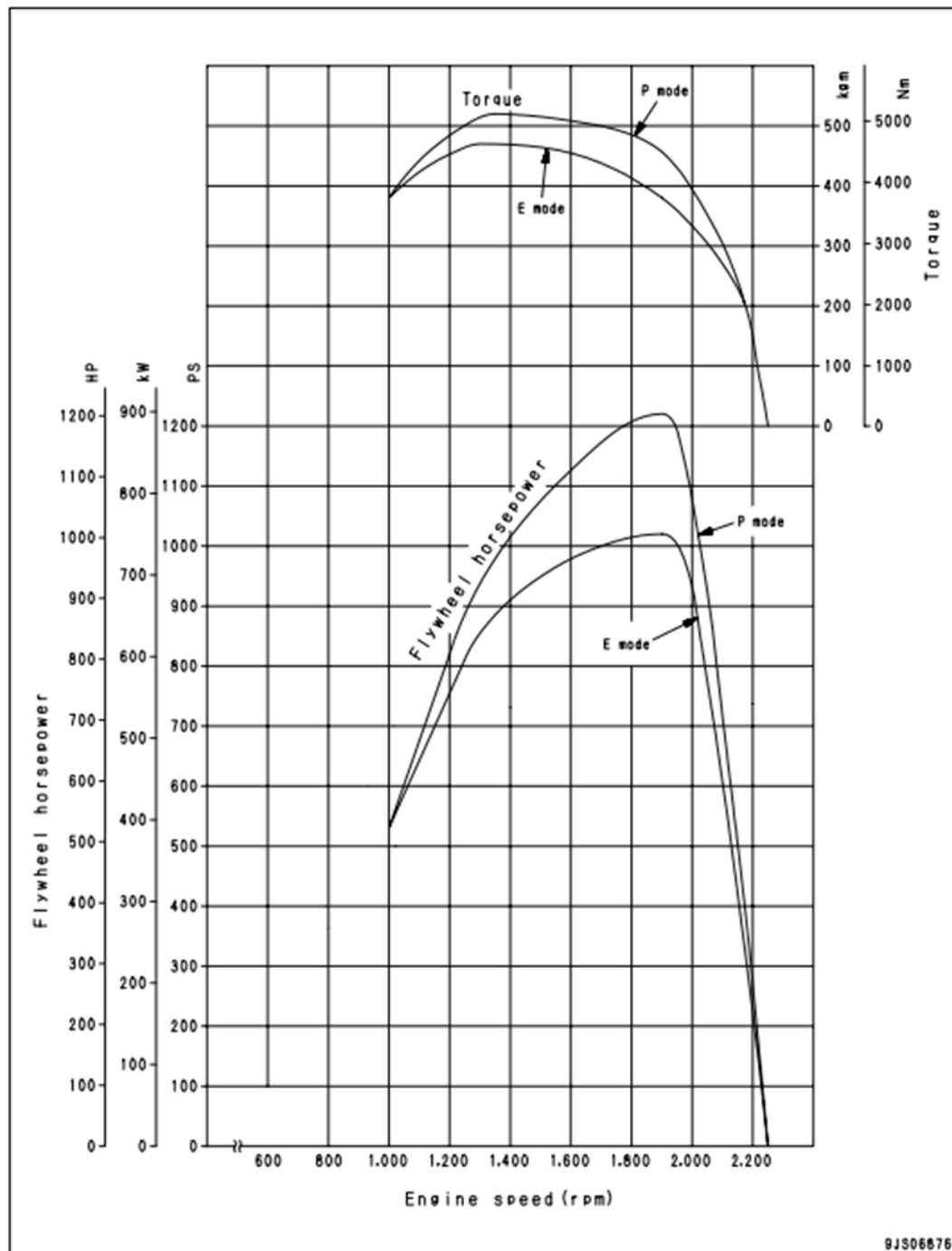
Engine name		SAA12V140E-3
Machine model		HD785-7
Number of cylinders – Bore x Stroke		mm
Total displacement		ℓ {cc}
Firing order		—
		12 – 140 x 165 30.5 {30,480} R1-L1-R5-L5-R3-L3-R6-L6-R2-L2-R4-L4
Dimensions	Overall length	mm
	Overall width	mm
	Overall height (excluding exhaust pipe)	mm
	Overall height (including exhaust pipe)	mm
Performance	Rated output	kW{HP}/rpm
	Max. torque	Nm(kgm)/rpm
	No-load max. speed (High idle speed)	rpm
	No-load min. speed (Low idle speed)	rpm
	Fuel consumption ratio at rated output	g/kW·h {g/HP·h}
Dry weight		kg
Fuel supply pump		—
Governor		—
Quantity of lubricating oil (Refill capacity)		ℓ
Quantity of coolant		ℓ
Alternator		—
Starting motor		—
Battery		—
Turbocharger		—
Air compressor		—
Others		—

Anexo V. Curvas de desempeño de motor ensayado

SAA12V140E-3 (Machine model: HD785-7)

Rated output: 895 kW {1,200 HP}/1,900 rpm (Gross)

Max. torque: 5,076 Nm {518 kgm}/1,350 rpm (Gross)



Anexo VI. Codificación - Análisis_de_datos

```
Analizando y Normalizacion de datos

import pandas as pd

import numpy as np

datos=pd.read_csv("Registro_SAA12V140E-3_dino1.csv")

#Datos del dataframe

print(" ")

print("cantidad de filas y columnas:",datos.shape)

#Eligiendo columnas

base_datos=datos[["Por Ciento de Pedal o Palanca del Acelerador (Por Ciento) - Identificador 0-Primaria",

    "Presion Barometrica del Aire (kPa) - Identificador 0-Primaria",

    "Presion de Aceite del Motor (kPa) - Identificador 0-Primaria",

    "Presion del Multiple de Admision (kPa) - Identificador 0-Primaria",

    "Presion del Riel de Combustible Ordenada (bar) - Identificador 0-Primaria",

    "Presion del Riel de Combustible Medida (bar) - Identificador 0-Primaria",

    "Presion del Riel de Combustible Medida (bar) - Identificador 1-Primaria",

    "Sensor de Temperatura del Combustible (C) - Identificador 0-Primaria",

    "Sensor de Temperatura del Combustible (C) - Identificador 1-Primaria",

    "Temperatura de Aire del Multiple de Admision (C) - Identificador 0-Primaria",

    "Temperatura del Aceite del Motor (C) - Identificador 0-Primaria",

    "Temperatura del Refrigerante del Motor (C) - Identificador 0-Primaria",

    "Velocidad del Motor (RPM) - Identificador 0-Primaria",

    "Torque instantaneo (Nm)",

    "Potencia instantanea (kW)",

    "Combustible Instantaneo (L/hr)"]]

#cambiando las etiquetas a las columnas

base_datos.columns=["Aplicacion del Pedal del Acelerador (Por Ciento)",

    "Presion Barometrica del Aire (kPa)",

    "Presion de Aceite del Motor (kPa)",

    "Presion del Multiple de Admision (kPa)",

    "Presion del Riel de Combustible Ordenada (bar)",

    "Presion del Riel de Combustible Medida (bar) Izq",

    "Presion del Riel de Combustible Medida (bar) Der",

    "Temperatura del Combustible (°C) Izq",

    "Temperatura del Combustible (°C) Der",
```

"Temperatura de Aire del Multiple de Admision (°C)",
 "Temperatura del Aceite del Motor (°C)",
 "Temperatura del Refrigerante del Motor (°C)",
 "Velocidad del Motor (RPM)",
 "Torque instantaneo (Nm)",
 "Potencia instantanea (kW)",
 "Combustible Instantaneo (L/hr)"]

```
base_datos.to_csv('Datos_No_estandarizados.csv', index=False)
base_escalado=base_datos.copy()
#Normalizacion de datos
from sklearn.preprocessing import MinMaxScaler, StandardScaler
#scaler=MinMaxScaler()
scaler=StandardScaler()
base_escalado[["Aplicacion del Pedal del Acelerador (Por Ciento)",
  "Presion Barometrica del Aire (kPa)",
  "Presion de Aceite del Motor (kPa)",
  "Presion del Multiple de Admision (kPa)",
  "Presion del Riel de Combustible Ordenada (bar)",
  "Presion del Riel de Combustible Medida (bar) Izq",
  "Presion del Riel de Combustible Medida (bar) Der",
  # "Regimen de Combustible Instantaneo (L/hr) Izq",
  # "Regimen de Combustible Instantaneo (L/hr) Der",
  "Temperatura del Combustible (°C) Izq",
  "Temperatura del Combustible (°C) Der",
  "Temperatura de Aire del Multiple de Admision (°C)",
  "Temperatura del Aceite del Motor (°C)",
  "Temperatura del Refrigerante del Motor (°C)",
  "Velocidad del Motor (RPM)",
  "Torque instantaneo (Nm)",
  "Potencia instantanea (kW)",
  "Combustible Instantaneo (L/hr)"]]=scaler.fit_transform(base_escalado[
  "Aplicacion del Pedal del Acelerador (Por Ciento)",
  "Presion Barometrica del Aire (kPa)",
  "Presion de Aceite del Motor (kPa)",
  "Presion del Multiple de Admision (kPa)",
  "Presion del Riel de Combustible Ordenada (bar)",
```

```

"Presion del Riel de Combustible Medida (bar) Izq",
"Presion del Riel de Combustible Medida (bar) Der",
#"Regimen de Combustible Instantaneo (L/hr) Izq",
#"Regimen de Combustible Instantaneo (L/hr) Der",
"Temperatura del Combustible (°C) Izq",
"Temperatura del Combustible (°C) Der",
"Temperatura de Aire del Multiple de Admision (°C)",
"Temperatura del Aceite del Motor (°C)",
"Temperatura del Refrigerante del Motor (°C)",
"Velocidad del Motor (RPM)",
"Torque instantaneo (Nm)",
"Potencia instantanea (kW)",
"Combustible Instantaneo (L/hr)"]]]

datos=base_escalado.copy()

# Guardamos el DataFrame en un archivo CSV
datos.to_csv('Datos_estandarizados.csv', index=False)

print ( "Nombre de columnas",datos.columns)

print(" ")

print("informacion de los datos")

print(datos.info())

# Guardar escalador
import pickle

pickle_file = "scaler_scalador.pkl"

with open(pickle_file, 'wb') as file: pickle.dump(scaler, file)

print(f"Escalador guardado en {pickle_file}")

```

Codificacion-

#Análisis de correlación

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

datos=pd.read_csv("Datos_estandarizados.csv")

#Análisis de correlación

print(" ")

print("Matriz de correlacion")

corr_matriz = datos.corr()

print(corr_matriz)

plt.matshow(corr_matriz,cmap="bwr",vmin=-1, vmax=1)

plt.xticks(range(datos.shape[1]),datos.columns,rotation=90)

plt.yticks(range(datos.shape[1]),datos.columns)

for i in range(len(corr_matriz.columns)):

 for j in range(len (corr_matriz.columns)):

 plt.text(i,j,round(corr_matriz.iloc[i,j],2),ha="center",

 va="center")

plt.colorbar()

plt.show()

print(" ")

print("listado de correlacion para el Torque")

listado=datos.corr("Spearman")["Torque instantaneo (Nm)"]

listado=listado.sort_values(ascending=False)

listado.to_csv('Correlaciones.csv', index=True)

print(listado)

print(" ")

print("listado de correlacionde para la Potencia")

listado=datos.corr("Spearman")["Potencia instantanea (kW)"]

listado=listado.sort_values(ascending=False)

listado.to_csv('Correlaciones.csv', index=True,mode='a')

print(listado)

print(" ")

```
print("listado de correlacione para el Consumo de combustinle")
listado=datos.corr("Spearman")["Combustible Instantaneo (L/hr)"]
listado=listado.sort_values(ascending=False)
listado.to_csv('Correlaciones.csv', index=True,mode='a')
print(listado)
```

Anexo VII. Codificación – Correlación

```
# Gráfico de distribución para cada variable numérica

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
#import statsmodels.formula.api as smf
#from statsmodels.stats.diagnostic import het_breuschpagan
#from statsmodels.stats.outliers_influence import variance_inflation_factor
#from statsmodels.tools.eval_measures import rmse
#from scipy import stats

from sklearn.model_selection import train_test_split
#from typing import Union

datos=pd.read_csv("Datos_No_estandarizados.csv")
fig, axes = plt.subplots(nrows=4, ncols=4, figsize=(9, 6))
axes = axes.flat

columnas_numeric = datos.select_dtypes(include=np.number).columns
for i, column in enumerate(columnas_numeric):
    sns.histplot(
        data = datos,
        x = column,
        stat = "count",
        kde = True,
        color = (list(plt.rcParams['axes.prop_cycle'])*2)[i]["color"],
        ax = axes[i]
    )
    axes[i].set_title(column, fontsize = 10)
    axes[i].tick_params(labelsize = 8)
    axes[i].set_xlabel("")

fig.tight_layout()
plt.subplots_adjust(top = 0.9)
fig.suptitle('Distribución variables numéricas', fontsize = 11);

plt.show()
```

Anexo VIII. Codificación – Regresión Lineal multiple

```
Regresion Lineal Multiple_06

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values
Tcomb2=datos["Temperatura del Combustible (°C) Der"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values
T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values


Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values


X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ]).T
y=np.array (Par)
z=np.array(Pot)
w=np.array(Consumo)

#Dividiendo los conjuntos de entrenamiento y prueba
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test =train_test_split(X,y,z,w, test_size=0.2)


#Definiendo el algoritmo a utilizar
from sklearn import linear_model
regresion_y=linear_model.LinearRegression()
regresion_z=linear_model.LinearRegression()
regresion_w=linear_model.LinearRegression()
```



```

#Entrenado el modelo
regresion_y.fit(X_train,y_train)
regresion_z.fit(X_train,z_train)
regresion_w.fit(X_train,w_train)

#prediccion valores
y_pred=regresion_y.predict(X_test)
z_pred=regresion_z.predict(X_test)
w_pred=regresion_w.predict(X_test)

### Calcular el error absoluto medio
##from sklearn.metrics import mean_absolute_error
##mae = mean_absolute_error(y_test, y_pred)
##print("Error absoluto medio y:", mae)
##mse = mean_absolute_error(z_test, z_pred)
##print("Error absoluto medio z:", mae)
##mse = mean_absolute_error(w_test, w_pred)
##print("Error absoluto medio w:", mae)
##print(" ")

### Calcular el error cuadrático medio
##from sklearn.metrics import mean_squared_error
##mse = mean_squared_error(y_test, y_pred)
##print("Error cuadrático medio y:", mse)
##mse = mean_squared_error(z_test, z_pred)
##print("Error cuadrático medio z:", mse)
##mse = mean_squared_error(w_test, w_pred)
##print("Error cuadrático medio w:", mse)
##print(" ")
##

### Calcular raiz error cuadrático medio
##from sklearn.metrics import mean_squared_error
##from math import sqrt
##mse = mean_squared_error(y_test, y_pred)
##print("Raiz del Error cuadrático medio y:", sqrt(mse))
##mse = mean_squared_error(z_test, z_pred)

```

```

##print("Raiz del Error cuadrático medio z:", sqrt(mse))
##mse = mean_squared_error(w_test, w_pred)
##print("Raiz del Error cuadrático medio w:", sqrt(mse))
##print(" ")
##
### Calcular el coeficiente de determinacion
##from sklearn.metrics import r2_score
##R2=r2_score(y_test, y_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(z_test, z_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(w_test, w_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],
    "MAE": [mean_absolute_error(y_test, y_pred),mean_absolute_error(z_test, z_pred),
mean_absolute_error(w_test, w_pred)],
    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
mean_squared_error(w_test, w_pred)],
    "RMSE":[mean_squared_error(y_test, y_pred,squared=False), mean_squared_error(z_test,
z_pred,squared=False), mean_squared_error(w_test, w_pred,squared=False)],
    "R2": [r2_score(y_test, y_pred),r2_score(z_test, z_pred), r2_score(w_test, w_pred)]
}

metricas=pd.DataFrame(metricas_diccionario)
metricas.to_csv('Metricas_RLM.csv', index=True)
print("")
print(metricas)
print("")

# Imprimir los coeficientes y el sesgo - Valores de los pesos del modelo de regresión
# print(" ")
# print("coeficientes para el torque")
# print("Coeficientes:", regresion_y.coef_)
# print("Sesgo (Coeficiente ", regresion_y.intercept_)

```

```

# print("Coeficiente Acel:", regresion_y.coef_[0])
# print("Coeficiente P_bar:", regresion_y.coef_[1])
# print("Coeficiente P_aceite:", regresion_y.coef_[2])
# print("Coeficiente P_adm:", regresion_y.coef_[3])
# print("Coeficiente P_riel_comandada:", regresion_y.coef_[4])
# print("Coeficiente P_riel_sensada1:", regresion_y.coef_[5])
# print("Coeficiente P_riel_sensada2", regresion_y.coef_[6])
# print("Coeficiente Tcombustible1:", regresion_y.coef_[7])
# print("Coeficiente Tcombustible2:", regresion_y.coef_[8])
# print("Coeficiente T_admision", regresion_y.coef_[9])
# print("Coeficiente T_aceite", regresion_y.coef_[10])
# print("Coeficiente TRefrigerante:", regresion_y.coef_[11])
# print("Coeficiente rpm:", regresion_y.coef_[12])

```

```

# print(" ")
# print("coeficientes para la potencia")
# print("Coeficientes:", regresion_z.coef_)
# print("Sesgo (Coeficiente ", regresion_z.intercept_)
# print("Coeficiente Acel:", regresion_z.coef_[0])
# print("Coeficiente P_bar:", regresion_z.coef_[1])
# print("Coeficiente P_aceite:", regresion_z.coef_[2])
# print("Coeficiente P_adm:", regresion_z.coef_[3])
# print("Coeficiente P_riel_comandada:", regresion_z.coef_[4])
# print("Coeficiente P_riel_sensada1:", regresion_z.coef_[5])
# print("Coeficiente P_riel_sensada2", regresion_z.coef_[6])
# print("Coeficiente Tcombustible1:", regresion_z.coef_[7])
# print("Coeficiente Tcombustible2:", regresion_z.coef_[8])
# print("Coeficiente T_admision", regresion_z.coef_[9])
# print("Coeficiente T_aceite", regresion_z.coef_[10])
# print("Coeficiente TRefrigerante:", regresion_z.coef_[11])
# print("Coeficiente rpm:", regresion_z.coef_[12])

```

```

# print(" ")
# print("coeficientes para el consumo")
# print("Coeficientes:", regresion_w.coef_)
# print("Sesgo (Coeficiente ", regresion_w.intercept_)

```

```

# print("Coeficiente Acel:", regresion_w.coef_[0])
# print("Coeficiente P_bar:", regresion_w.coef_[1])
# print("Coeficiente P_aceite:", regresion_w.coef_[2])
# print("Coeficiente P_adm:", regresion_w.coef_[3])
# print("Coeficiente P_riel_comandada:", regresion_w.coef_[4])
# print("Coeficiente P_riel_sensada1:", regresion_w.coef_[5])
# print("Coeficiente P_riel_sensada2:", regresion_w.coef_[6])
# print("Coeficiente Tcombustible1:", regresion_w.coef_[7])
# print("Coeficiente Tcombustible2:", regresion_w.coef_[8])
# print("Coeficiente T_admision", regresion_w.coef_[9])
# print("Coeficiente T_aceite", regresion_w.coef_[10])
# print("Coeficiente TRefrigerante:", regresion_w.coef_[11])
# print("Coeficiente rpm:", regresion_w.coef_[12])

# Cargar escalador
import pickle
pickle_file = "scaler_scalador.pkl"
with open(pickle_file, 'rb') as file:
    loaded_scaler = pickle.load(file)
print(f"Escalador cargado de {pickle_file}")

##-----
""" Transformación Inversa Data Test Normalizada """
### Crear arreglo de datos testeados y salida testeadas
scaled_test = np.hstack((X_test,
                        y_test.reshape((-1, 1)),
                        z_test.reshape((-1, 1)),
                        w_test.reshape((-1, 1))))
# Transformación inversa de arreglo combinado
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)
# Extraer salidas inversamente normalizadas
y_test_no_scale = inverse_scaled_test[:, -3]
z_test_no_scale = inverse_scaled_test[:, -2]
w_test_no_scale = inverse_scaled_test[:, -1]

##-----
""" Transformación Inversa Predicción Normalizada """
### Crear arreglo de datos testeados y salida pronosticada

```

```

scaled_pred = np.hstack((X_test,
                          y_pred.reshape((-1, 1)),
                          z_pred.reshape((-1, 1)),
                          w_pred.reshape((-1, 1))))
# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)
# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]
w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----
""" Renombrar y_test y y_pred """
Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()
ax2.grid()
ax3.grid()

ax1.set_title ("Predicción del par")
ax2.set_title("Predicción de la potencia")

```

```

ax3.set_title("Predicción del consumo")

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]
b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} ", fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} ", fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} ", fontsize = 12)

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')

plt.show(block=True)

```

Anexo IX. Codificación – K vecinos más cercanos

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values
Tcomb2=datos["Temperatura del Combustible (°C) Der"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values
T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values

Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values

X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ])
y=np.array(Par)
z=np.array(Pot)
w=np.array(Consumo)

#Dividiendo los conjuntos de entrenamiento y prueba
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test =train_test_split(X,y,z,w, test_size=0.2)

#Definiendo el algoritmo a utilizar
from sklearn.neighbors import KNeighborsRegressor
```

```

knn_y= KNeighborsRegressor(n_neighbors=4)
knn_z= KNeighborsRegressor(n_neighbors=4)
knn_w= KNeighborsRegressor(n_neighbors=4)


#Entrenado el modelo
knn_y.fit(X_train,y_train)
knn_z.fit(X_train,z_train)
knn_w.fit(X_train,w_train)


#prediccion valores
y_pred=knn_y.predict(X_test)
z_pred=knn_z.predict(X_test)
w_pred=knn_w.predict(X_test)


# Calcular el error absoluto medio
from sklearn.metrics import mean_absolute_error
mae = mean_absolute_error(y_test, y_pred)
print("Error absoluto medio y:", mae)
mse = mean_absolute_error(z_test, z_pred)
print("Error absoluto medio z:", mae)
mse = mean_absolute_error(w_test, w_pred)
print("Error absoluto medio w:", mae)
print(" ")


# Calcular el error cuadrático medio
from sklearn.metrics import mean_squared_error
mse = mean_squared_error(y_test, y_pred)
print("Error cuadrático medio y:", mse)
mse = mean_squared_error(z_test, z_pred)
print("Error cuadrático medio z:", mse)
mse = mean_squared_error(w_test, w_pred)
print("Error cuadrático medio w:", mse)
print(" ")


# Calcular raiz error cuadrático medio
from sklearn.metrics import mean_squared_error

```



```

from math import sqrt

mse = mean_squared_error(y_test, y_pred)
print("Raiz del Error cuadrático medio y:", sqrt(mse))

mse = mean_squared_error(z_test, z_pred)
print("Raiz del Error cuadrático medio z:", sqrt(mse))

mse = mean_squared_error(w_test, w_pred)
print("Raiz del Error cuadrático medio w:", sqrt(mse))

print(" ")

# Calcular el coeficiente de determinacion
from sklearn.metrics import r2_score
R2=r2_score(y_test, y_pred)*100
print("El coeficiente de determinacion (R^2) es:",R2, "%")
R2=r2_score(z_test, z_pred)*100
print("El coeficiente de determinacion (R^2) es:",R2, "%")
R2=r2_score(w_test, w_pred)*100
print("El coeficiente de determinacion (R^2) es:",R2, "%")
print("")

#Calculando la precision media dle modelo
Score=knn_y.score(X_train, y_train)
print("Precision media del modelo:", Score)
Score=knn_z.score(X_train, z_train)
print("Precision media del modelo::", Score)
Score=knn_w.score(X_train, w_train)
print("Precision media del modelo::", Score)

metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],
    "MAE": [mean_absolute_error(y_test, y_pred),mean_absolute_error(z_test, z_pred),
mean_absolute_error(w_test, w_pred)],
    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
mean_squared_error(w_test, w_pred)],
    "RMSE":[mean_squared_error(y_test, y_pred,squared=False), mean_squared_error(z_test,
z_pred,squared=False), mean_squared_error(w_test, w_pred,squared=False)],
    "R2": [r2_score(y_test, y_pred),r2_score(z_test, z_pred), r2_score(w_test, w_pred)]
}

metricas=pd.DataFrame(metricas_diccionario)

```

```

metricas.to_csv('Metricas_KNN.csv', index=True)

print("")
print(metricas)
print("")

# Cargar escalador
import pickle
pickle_file = "scaler_scalador.pkl"
with open(pickle_file, 'rb') as file:
    loaded_scaler = pickle.load(file)
print(f"Escalador cargado de {pickle_file}")

##-----
""" Transformación Inversa Data Test Normalizada """
### Crear arreglo de datos testeados y salida testeadas
scaled_test = np.hstack((X_test,
                        y_test.reshape((-1, 1)),
                        z_test.reshape((-1, 1)),
                        w_test.reshape((-1, 1))))

# Transformación inversa de arreglo combinado
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)

# Extraer salidas inversamente normalizadas
y_test_no_scale = inverse_scaled_test[:, -3]
z_test_no_scale = inverse_scaled_test[:, -2]
w_test_no_scale = inverse_scaled_test[:, -1]

##-----
""" Transformación Inversa Predicción Normalizada """
### Crear arreglo de datos testeados y salida pronosticada
scaled_pred = np.hstack((X_test,
                        y_pred.reshape((-1, 1)),
                        z_pred.reshape((-1, 1)),
                        w_pred.reshape((-1, 1))))

# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)

# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]

```

```

w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----

""" Renombrar y_test y y_pred """

Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()
ax2.grid()
ax3.grid()

ax1.set_title ("Predicción del par")
ax2.set_title("Predicción de la potencia")
ax3.set_title("Predicción del consumo")

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

```

```

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]
b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} ", fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} ", fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} ", fontsize = 12)

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')
plt.show(block=True)

```

Anexo X. Codificación - Máquinas de vectores soporte

```
#VSM

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values
Tcomb2=datos["Temperatura del Combustible (°C) Der"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values
T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values

Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values

X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ]).T

y=np.array (Par)
z=np.array(Pot)
w=np.array(Consumo)

#Dividiendo los conjuntos de entrenamiento y prueba
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test =train_test_split(X,y,z,w, test_size=0.2)
```

```

#Definiendo el algoritmo a utilizar
from sklearn import linear_model
svmr_y=linear_model.LinearRegression()
svmr_z=linear_model.LinearRegression()
svmr_w=linear_model.LinearRegression()

#Entrenado el modelo
svmr_y.fit(X_train,y_train)
svmr_z.fit(X_train,z_train)
svmr_w.fit(X_train,w_train)

#prediccion valores
y_pred=svmr_y.predict(X_test)
z_pred=svmr_z.predict(X_test)
w_pred=svmr_w.predict(X_test)

#### Calcular el error absoluto medio
##from sklearn.metrics import mean_absolute_error
##mae = mean_absolute_error(y_test, y_pred)
##print("Error absoluto medio y:", mae)
##mse = mean_absolute_error(z_test, z_pred)
##print("Error absoluto medio z:", mae)
##mse = mean_absolute_error(w_test, w_pred)
##print("Error absoluto medio w:", mae)
##print(" ")
##

#### Calcular el error cuadrático medio
##from sklearn.metrics import mean_squared_error
##mse = mean_squared_error(y_test, y_pred)
##print("Error cuadrático medio y:", mse)
##mse = mean_squared_error(z_test, z_pred)
##print("Error cuadrático medio z:", mse)
##mse = mean_squared_error(w_test, w_pred)
##print("Error cuadrático medio w:", mse)
##print(" ")
##

```

```

#### Calcular raiz error cuadrático medio
##from sklearn.metrics import mean_squared_error
##from math import sqrt
##mse = mean_squared_error(y_test, y_pred)
##print("Raiz del Error cuadrático medio y:", sqrt(mse))
##mse = mean_squared_error(z_test, z_pred)
##print("Raiz del Error cuadrático medio z:", sqrt(mse))
##mse = mean_squared_error(w_test, w_pred)
##print("Raiz del Error cuadrático medio w:", sqrt(mse))
##print(" ")
##

#### Calcular el coeficiente de determinacion
##from sklearn.metrics import r2_score
##R2=r2_score(y_test, y_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(z_test, z_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(w_test, w_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##print("")

####Calculando la precision media del modelo
##Score=svmr_y.score(X_train, y_train)
##print("Precision media del modelo:", Score)
##Score=svmr_z.score(X_train, z_train)
##print("Precision media del modelo:", Score)
##Score=svmr_w.score(X_train, w_train)
##print("Precision media del modelo:", Score)

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score

metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],
    "MAE": [mean_absolute_error(y_test, y_pred),mean_absolute_error(z_test, z_pred),
mean_absolute_error(w_test, w_pred)],
    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
mean_squared_error(w_test, w_pred)],

```

```

    "RMSE": [mean_squared_error(y_test, y_pred, squared=False), mean_squared_error(z_test,
z_pred, squared=False), mean_squared_error(w_test, w_pred, squared=False)],
    "R2": [r2_score(y_test, y_pred), r2_score(z_test, z_pred), r2_score(w_test, w_pred)]
}

```

```

metricas=pd.DataFrame(metricas_diccionario)
metricas.to_csv('Metricas_VSM.csv', index=True)
print("")
print(metricas)
print("")

```

Cargar escalador

```

import pickle
pickle_file = "scaler_scalador.pkl"
with open(pickle_file, 'rb') as file:
    loaded_scaler = pickle.load(file)
print(f"Escalador cargado de {pickle_file}")

```

##-----

""" Transformación Inversa Data Test Normalizada """

Crear arreglo de datos testeados y salida testada

```

scaled_test = np.hstack((X_test,
                        y_test.reshape((-1, 1)),
                        z_test.reshape((-1, 1)),
                        w_test.reshape((-1, 1))))

```

Transformación inversa de arreglo combinado

```
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)
```

Extraer salidas inversamente normalizadas

```

y_test_no_scale = inverse_scaled_test[:, -3]
z_test_no_scale = inverse_scaled_test[:, -2]
w_test_no_scale = inverse_scaled_test[:, -1]

```

##-----

""" Transformación Inversa Predicción Normalizada """

Crear arreglo de datos testeados y salida pronosticada

```

scaled_pred = np.hstack((X_test,
                        y_pred.reshape((-1, 1)),
                        z_pred.reshape((-1, 1)),
                        w_pred.reshape((-1, 1))))

```



```

# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)

# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]
w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----
""" Renombrar y_test y y_pred """
Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()
ax2.grid()
ax3.grid()

ax1.set_title ("Predicción del par")
ax2.set_title("Predicción de la potencia")
ax3.set_title("Predicción del consumo")

```

```

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]
b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} ", fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} ", fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} ", fontsize = 12)

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')
plt.show(block=True)

```

Anexo XI. Codificación - Árboles de decisión

```
#Arboles de decision

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values
Tcomb2=datos["Temperatura del Combustible (°C) Der"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values
T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values

Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values

X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ]).T

y=np.array (Par)
z=np.array(Pot)
w=np.array(Consumo)

#Dividiendo los conjuntos de entrenamiento y prueba
```

```

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test = train_test_split(X, y, z, w, test_size=0.2)


#Definiendo el algoritmo a utilizar

from sklearn.tree import DecisionTreeRegressor

adr_y = DecisionTreeRegressor ( criterion='squared_error', max_depth=4)
adr_z = DecisionTreeRegressor ( criterion='squared_error', max_depth=4)
adr_w = DecisionTreeRegressor ( criterion='squared_error', max_depth=4)


#DecisionTreeRegressor(max_depth=10, max_features=10, max_leaf_nodes=4, min_samples_leaf=
5, min_samples_split=5, random_state=1)


#Entrenado el modelo

adr_y.fit(X_train, y_train)
adr_z.fit(X_train, z_train)
adr_w.fit(X_train, w_train)


#prediccion valores

y_pred = adr_y.predict(X_test)
z_pred = adr_z.predict(X_test)
w_pred = adr_w.predict(X_test)


# Calcular el error absoluto medio

from sklearn.metrics import mean_absolute_error

mae = mean_absolute_error(y_test, y_pred)
print("Error absoluto medio y:", mae)

mse = mean_absolute_error(z_test, z_pred)
print("Error absoluto medio z:", mae)

mse = mean_absolute_error(w_test, w_pred)
print("Error absoluto medio w:", mae)
print(" ")


# Calcular el error cuadrático medio

from sklearn.metrics import mean_squared_error

mse = mean_squared_error(y_test, y_pred)
print("Error cuadrático medio y:", mse)

mse = mean_squared_error(z_test, z_pred)

```

```

print("Error cuadrático medio z:", mse)
mse = mean_squared_error(w_test, w_pred)
print("Error cuadrático medio w:", mse)
print(" ")

### Calcular raiz error cuadrático medio
##from sklearn.metrics import mean_squared_error
##from math import sqrt
##mse = mean_squared_error(y_test, y_pred)
##print("Raiz del Error cuadrático medio y:", sqrt(mse))
##mse = mean_squared_error(z_test, z_pred)
##print("Raiz del Error cuadrático medio z:", sqrt(mse))
##mse = mean_squared_error(w_test, w_pred)
##print("Raiz del Error cuadrático medio w:", sqrt(mse))
##print(" ")
##

### Calcular el coeficiente de determinacion
##from sklearn.metrics import r2_score
##R2=r2_score(y_test, y_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(z_test, z_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(w_test, w_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##print(" ")
##

###Calculando la precision media del modelo
##Score=adr_y.score(X_train, y_train)
##print("Precision media del modelo:", Score)
##Score=adr_z.score(X_train, z_train)
##print("Precision media del modelo:", Score)
##Score=adr_w.score(X_train, w_train)
##print("Precision media del modelo:", Score)
##

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score

```

```

metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],
    "MAE": [mean_absolute_error(y_test, y_pred),mean_absolute_error(z_test, z_pred),
mean_absolute_error(w_test, w_pred)],
    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
mean_squared_error(w_test, w_pred)],
    "RMSE":[mean_squared_error(y_test, y_pred,squared=False), mean_squared_error(z_test,
z_pred,squared=False), mean_squared_error(w_test, w_pred,squared=False)],
    "R2": [r2_score(y_test, y_pred),r2_score(z_test, z_pred), r2_score(w_test, w_pred)]
}

```

```

metricas=pd.DataFrame(metricas_diccionario)
metricas.to_csv('Metricas_Arbol.csv', index=True)
print("")
print(metricas)
print("")

```

```

# Cargar escalador
import pickle
pickle_file = "scaler_scalador.pkl"
with open(pickle_file, 'rb') as file:
    loaded_scaler = pickle.load(file)
print(f"Escalador cargado de {pickle_file}")

```

```

##-----

```

```

""" Transformación Inversa Data Test Normalizada """
### Crear arreglo de datos testeados y salida testada
scaled_test = np.hstack((X_test,
                        y_test.reshape((-1, 1)),
                        z_test.reshape((-1, 1)),
                        w_test.reshape((-1, 1))))
# Transformación inversa de arreglo combinado
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)
# Extraer salidas inversamente normalizadas
y_test_no_scale = inverse_scaled_test[:, -3]
z_test_no_scale = inverse_scaled_test[:, -2]
w_test_no_scale = inverse_scaled_test[:, -1]

```

```

##-----
""" Transformación Inversa Predicción Normalizada """
### Crear arreglo de datos testeados y salida pronosticada
scaled_pred = np.hstack((X_test,
                          y_pred.reshape((-1, 1)),
                          z_pred.reshape((-1, 1)),
                          w_pred.reshape((-1, 1))))
# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)
# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]
w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----
""" Renombrar y_test y y_pred """
Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()

```

```

ax2.grid()
ax3.grid()

ax1.set_title("Predicción del par")
ax2.set_title("Predicción de la potencia")
ax3.set_title("Predicción del consumo")

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]
b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} " , fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} " , fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} " , fontsize = 12)

```



```

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')


plt.show(block=True)


from sklearn import tree
fig=plt.figure(figsize=(25,30))
tree.plot_tree(adr_y)
plt.show()
#-----

import statsmodels.api as sm
#add constant to predictor variables
x = sm.add_constant(X_test)
y=y_pred


#fit regression model
model = sm.OLS(x,y).fit()


#view AIC of model
print("\nEl coeficiente AIC del algoritmo DT es:")
print(model.aic)

```

Anexo XII. Codificación - Bosques aleatorios

```
#Random Forest

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values
Tcomb2=datos["Temperatura del Combustible (°C) Der"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values
T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values

Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values

X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ]).T

y=np.array (Par)
z=np.array(Pot)
w=np.array(Consumo)

#Dividiendo los conjuntos de entrenamamiento y prueba
from sklearn.model_selection import train_test_split
```

```

X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test = train_test_split(X,y,z,w, test_size=0.2)

#Definiendo el algoritmo a utilizar
from sklearn.ensemble import RandomForestRegressor

bar_y = RandomForestRegressor(n_estimators = 300, max_depth = 9)
bar_z = RandomForestRegressor(n_estimators = 300, max_depth = 9)
bar_w = RandomForestRegressor(n_estimators = 300, max_depth = 9)

#Entrenado el modelo
bar_y.fit(X_train,y_train)
bar_z.fit(X_train,z_train)
bar_w.fit(X_train,w_train)

#prediccion valores
y_pred=bar_y.predict(X_test)
z_pred=bar_z.predict(X_test)
w_pred=bar_w.predict(X_test)

##

### Calcular el error absoluto medio
##from sklearn.metrics import mean_absolute_error
##mae = mean_absolute_error(y_test, y_pred)
##print("Error absoluto medio y:", mae)
##mse = mean_absolute_error(z_test, z_pred)
##print("Error absoluto medio z:", mae)
##mse = mean_absolute_error(w_test, w_pred)
##print("Error absoluto medio w:", mae)
##print(" ")
##

### Calcular el error cuadrático medio
##from sklearn.metrics import mean_squared_error
##mse = mean_squared_error(y_test, y_pred)
##print("Error cuadrático medio y:", mse)
##mse = mean_squared_error(z_test, z_pred)
##print("Error cuadrático medio z:", mse)
##mse = mean_squared_error(w_test, w_pred)
##print("Error cuadrático medio w:", mse)

```

```

##print(" ")
##
### Calcular raiz error cuadrático medio
##from sklearn.metrics import mean_squared_error
##from math import sqrt
##mse = mean_squared_error(y_test, y_pred)
##print("Raiz del Error cuadrático medio y:", sqrt(mse))
##mse = mean_squared_error(z_test, z_pred)
##print("Raiz del Error cuadrático medio z:", sqrt(mse))
##mse = mean_squared_error(w_test, w_pred)
##print("Raiz del Error cuadrático medio w:", sqrt(mse))
##print(" ")
##
### Calcular el coeficiente de determinacion
##from sklearn.metrics import r2_score
##R2=r2_score(y_test, y_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(z_test, z_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(w_test, w_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##print("")
##
###Calculando la precision media del modelo
##Score=bar_y.score(X_train, y_train)
##print("Precision media del modelo:", Score)
##Score=bar_z.score(X_train, z_train)
##print("Precision media del modelo:", Score)
##Score=bar_w.score(X_train, w_train)
##print("Precision media del modelo:", Score)

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score

metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],

```

```

    "MAE": [mean_absolute_error(y_test, y_pred), mean_absolute_error(z_test, z_pred),
            mean_absolute_error(w_test, w_pred)],
    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
            mean_squared_error(w_test, w_pred)],
    "RMSE": [mean_squared_error(y_test, y_pred, squared=False), mean_squared_error(z_test,
            z_pred, squared=False), mean_squared_error(w_test, w_pred, squared=False)],
    "R2": [r2_score(y_test, y_pred), r2_score(z_test, z_pred), r2_score(w_test, w_pred)]
}

```

```

metricas=pd.DataFrame(metricas_diccionario)
metricas.to_csv('Metricas_RF.csv', index=True)
print("")
print(metricas)
print("")

```

Cargar escalador

```
import pickle
```

```
pickle_file = "scaler_scalador.pkl"
```

```
with open(pickle_file, 'rb') as file:
```

```
    loaded_scaler = pickle.load(file)
```

```
print(f"Escalador cargado de {pickle_file}")
```

```
##-----
```

```
""" Transformación Inversa Data Test Normalizada """
```

```
### Crear arreglo de datos testeados y salida testeadas
```

```

scaled_test = np.hstack((X_test,
                          y_test.reshape((-1, 1)),
                          z_test.reshape((-1, 1)),
                          w_test.reshape((-1, 1))))

```

```
# Transformación inversa de arreglo combinado
```

```
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)
```

```
# Extraer salidas inversamente normalizadas
```

```
y_test_no_scale = inverse_scaled_test[:, -3]
```

```
z_test_no_scale = inverse_scaled_test[:, -2]
```

```
w_test_no_scale = inverse_scaled_test[:, -1]
```

```
##-----
```

```
""" Transformación Inversa Predicción Normalizada """
```

```
### Crear arreglo de datos testeados y salida pronosticada
```

```
scaled_pred = np.hstack((X_test,
```

```

        y_pred.reshape((-1, 1)),
        z_pred.reshape((-1, 1)),
        w_pred.reshape((-1, 1))))

# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)

# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]
w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----
""" Renombrar y_test y y_pred """
Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()
ax2.grid()
ax3.grid()

ax1.set_title ("Predicción del par")

```

```

ax2.set_title("Predicción de la potencia")
ax3.set_title("Predicción del consumo")

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]
b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} ", fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} ", fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} ", fontsize = 12)

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')
plt.show(block=True)

```

Anexo XIII. Codificación - Redes neuronales

```
#Regresion Lineal Multiple_06

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values
P_bar=datos["Presion Barometrica del Aire (kPa)"].values
P_aceite=datos["Presion de Aceite del Motor (kPa)"].values
P_adm=datos["Presion del Multiple de Admision (kPa)"].values
P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values
P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values
P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values
Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values
Tcomb2=datos["Temperatura del Combustible (°C) Der"].values
T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values
T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values
T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values
rpm=datos["Velocidad del Motor (RPM)"].values

Par=datos["Torque instantaneo (Nm)"].values
Pot=datos["Potencia instantanea (kW)"].values
Consumo=datos["Combustible Instantaneo (L/hr)"].values

X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ]).T

y=np.array (Par)
z=np.array(Pot)
```



```

w=np.array(Consumo)

#Dividiendo los conjuntos de entrenamiento y prueba
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test =train_test_split(X,y,z,w, test_size=0.2)

#Definiendo el algoritmo a utilizar
from sklearn.neural_network import MLPRegressor

rna_y=MLPRegressor(solver='adam',activation='relu', alpha=1e-5, hidden_layer_sizes=(10,10,10),
random_state=1,max_iter=200)

rna_z=MLPRegressor(solver='adam',activation='relu', alpha=1e-5, hidden_layer_sizes=(10,10,10),
random_state=1,max_iter=200)

rna_w=MLPRegressor(solver='adam',activation='relu', alpha=1e-5, hidden_layer_sizes=(10,10,10),
random_state=1,max_iter=200)

#Entrenado el modelo
rna_y.fit(X_train,y_train)
rna_z.fit(X_train,z_train)
rna_w.fit(X_train,w_train)

#prediccion valores
y_pred=rna_y.predict(X_test)
z_pred=rna_z.predict(X_test)
w_pred=rna_w.predict(X_test)

print("Coeficientes de Torque")
print(rna_y.coefs_)
print("Bias de Torque:")
print(rna_y.intercepts_)

##coeficientes=pd.DataFrame(rna_y.coefs_)
##coeficientes.to_csv('ANR_Coeficientes.csv', index=True)
##coeficientes=pd.DataFrame(rna_y.intercepts_)
##coeficientes.to_csv('ANR_Coeficientes.csv', index=True)
##print("Coeficientes y bias guardados")

```

```

##

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score

metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],
    "MAE": [mean_absolute_error(y_test, y_pred),mean_absolute_error(z_test, z_pred),
mean_absolute_error(w_test, w_pred)],
    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
mean_squared_error(w_test, w_pred)],
    "RMSE":[mean_squared_error(y_test, y_pred,squared=False), mean_squared_error(z_test,
z_pred,squared=False), mean_squared_error(w_test, w_pred,squared=False)],
    "R2": [r2_score(y_test, y_pred),r2_score(z_test, z_pred), r2_score(w_test, w_pred)]
}

metricas=pd.DataFrame(metricas_diccionario)
metricas.to_csv('Metricas_ANR.csv', index=True)
print("")
print(metricas)
print("")

# Cargar escalador
import pickle
pickle_file = "scaler_scalador.pkl"
with open(pickle_file, 'rb') as file:
    loaded_scaler = pickle.load(file)
print(f"Escalador cargado de {pickle_file}")

##-----
""" Transformación Inversa Data Test Normalizada """
### Crear arreglo de datos testeados y salida testeadas
scaled_test = np.hstack((X_test,
                        y_test.reshape((-1, 1)),

```

```

        z_test.reshape((-1, 1)),
        w_test.reshape((-1, 1))))

# Transformación inversa de arreglo combinado
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)

# Extraer salidas inversamente normalizadas
y_test_no_scale = inverse_scaled_test[:, -3]
z_test_no_scale = inverse_scaled_test[:, -2]
w_test_no_scale = inverse_scaled_test[:, -1]

##-----

""" Transformación Inversa Predicción Normalizada """

### Crear arreglo de datos testeados y salida pronosticada
scaled_pred = np.hstack((X_test,
                        y_pred.reshape((-1, 1)),
                        z_pred.reshape((-1, 1)),
                        w_pred.reshape((-1, 1))))

# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)

# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]
w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----

""" Renombrar y_test y y_pred """

Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)

```

```

fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()
ax2.grid()
ax3.grid()

ax1.set_title ("Predicción del par")
ax2.set_title("Predicción de la potencia")
ax3.set_title("Predicción del consumo")

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]

```

```

b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} ", fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} ", fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} ", fontsize = 12)

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')
plt.show(block=True)

```

Anexo XIV. Codificación - XG Boost

```
#Redes neuronales

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

datos=pd.read_csv("datos_estandarizados.csv")

Acel=datos["Aplicacion del Pedal del Acelerador (Por Ciento)"].values

P_bar=datos["Presion Barometrica del Aire (kPa)"].values

P_aceite=datos["Presion de Aceite del Motor (kPa)"].values

P_adm=datos["Presion del Multiple de Admision (kPa)"].values

P_riel_comandada=datos["Presion del Riel de Combustible Ordenada (bar)"].values

P_riel_sensada1=datos["Presion del Riel de Combustible Medida (bar) Izq"].values

P_riel_sensada2=datos["Presion del Riel de Combustible Medida (bar) Der"].values

Tcomb1=datos["Temperatura del Combustible (°C) Izq"].values

Tcomb2=datos["Temperatura del Combustible (°C) Der"].values

T_adm=datos["Temperatura de Aire del Multiple de Admision (°C)"].values

T_aceite=datos["Temperatura del Aceite del Motor (°C)"].values

T_refrig=datos["Temperatura del Refrigerante del Motor (°C)"].values

rpm=datos["Velocidad del Motor (RPM)"].values


Par=datos["Torque instantaneo (Nm)"].values

Pot=datos["Potencia instantanea (kW)"].values

Consumo=datos["Combustible Instantaneo (L/hr)"].values


X=np.array([ Acel, P_bar, P_aceite, P_adm, P_riel_comandada, P_riel_sensada1,
             P_riel_sensada2, Tcomb1, Tcomb2,T_adm, T_aceite, T_refrig, rpm, ]).T


y=np.array (Par)

z=np.array(Pot)

w=np.array(Consumo)


#Dividiendo los conjuntos de entrenamiento y prueba

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test, z_train, z_test, w_train, w_test =train_test_split(X,y,z,w, test_size=0.2)
```

```

#Definiendo el algoritmo a utilizar
from xgboost import XGBRegressor
import xgboost as xgb

model_y = XGBRegressor()
model_z = XGBRegressor()
model_w = XGBRegressor()

#Entrenado el modelo
model_y.fit(X_train,y_train)
model_z.fit(X_train,z_train)
model_w.fit(X_train,w_train)

#prediccion valores
y_pred=model_y.predict(X_test)
z_pred=model_z.predict(X_test)
w_pred=model_w.predict(X_test)

#### Calcular el error absoluto medio
##from sklearn.metrics import mean_absolute_error
##mae = mean_absolute_error(y_test, y_pred)
##print("Error absoluto medio y:", mae)
##mse = mean_absolute_error(z_test, z_pred)
##print("Error absoluto medio z:", mae)
##mse = mean_absolute_error(w_test, w_pred)
##print("Error absoluto medio w:", mae)
##print(" ")
##

#### Calcular el error cuadrático medio
##from sklearn.metrics import mean_squared_error
##mse = mean_squared_error(y_test, y_pred)
##print("Error cuadrático medio y:", mse)
##mse = mean_squared_error(z_test, z_pred)
##print("Error cuadrático medio z:", mse)
##mse = mean_squared_error(w_test, w_pred)
##print("Error cuadrático medio w:", mse)
##print(" ")
##

```

```

#### Calcular raiz error cuadrático medio
##from sklearn.metrics import mean_squared_error
##from math import sqrt
##mse = mean_squared_error(y_test, y_pred)
##print("Raiz del Error cuadrático medio y:", sqrt(mse))
##mse = mean_squared_error(z_test, z_pred)
##print("Raiz del Error cuadrático medio z:", sqrt(mse))
##mse = mean_squared_error(w_test, w_pred)
##print("Raiz del Error cuadrático medio w:", sqrt(mse))
##print(" ")
##
#### Calcular el coeficiente de determinacion
##from sklearn.metrics import r2_score
##R2=r2_score(y_test, y_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(z_test, z_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##R2=r2_score(w_test, w_pred)*100
##print("El coeficiente de determinacion (R^2) es:",R2, "%")
##print("")
##
##
####Calculando la precision media del modelo
##Score=model_y.score(X_train, y_train)
##print("Precision media del modelo:", Score)
##Score=model_z.score(X_train, z_train)
##print("Precision media del modelo:", Score)
##Score=model_w.score(X_train, w_train)
##print("Precision media del modelo:", Score)

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score

metricas_diccionario={
    "Salida":["Torque", "Potencia", "Consumo"],
    "MAE": [mean_absolute_error(y_test, y_pred),mean_absolute_error(z_test, z_pred),
mean_absolute_error(w_test, w_pred)],

```



```

    "MSE": [mean_squared_error(y_test, y_pred), mean_squared_error(z_test, z_pred),
mean_squared_error(w_test, w_pred)],

    "RMSE": [mean_squared_error(y_test, y_pred, squared=False), mean_squared_error(z_test,
z_pred, squared=False), mean_squared_error(w_test, w_pred, squared=False)],

    "R2": [r2_score(y_test, y_pred), r2_score(z_test, z_pred), r2_score(w_test, w_pred)]

}

```

```

metricas=pd.DataFrame(metricas_diccionario)
metricas.to_csv('Metricas_XB.csv', index=True)
print("")
print(metricas)
print("")

```

```

# Cargar escalador
import pickle
pickle_file = "scaler_scalador.pkl"
with open(pickle_file, 'rb') as file:
    loaded_scaler = pickle.load(file)
print(f"Escalador cargado de {pickle_file}")

```

```

##-----
""" Transformación Inversa Data Test Normalizada """
### Crear arreglo de datos testeados y salida testeadas
scaled_test = np.hstack((X_test,
                        y_test.reshape((-1, 1)),
                        z_test.reshape((-1, 1)),
                        w_test.reshape((-1, 1))))
# Transformación inversa de arreglo combinado
inverse_scaled_test = loaded_scaler.inverse_transform(scaled_test)
# Extraer salidas inversamente normalizadas
y_test_no_scale = inverse_scaled_test[:, -3]
z_test_no_scale = inverse_scaled_test[:, -2]
w_test_no_scale = inverse_scaled_test[:, -1]

```

```

##-----
""" Transformación Inversa Predicción Normalizada """
### Crear arreglo de datos testeados y salida pronosticada
scaled_pred = np.hstack((X_test,
                        y_pred.reshape((-1, 1)),

```

```

        z_pred.reshape((-1, 1)),
        w_pred.reshape((-1, 1))))

# Transformación inversa de arreglo combinado
inverse_scaled_pred = loaded_scaler.inverse_transform(scaled_pred)

# Extraer salidas inversamente normalizadas
y_pred_no_scale = inverse_scaled_pred[:, -3]
z_pred_no_scale = inverse_scaled_pred[:, -2]
w_pred_no_scale = inverse_scaled_pred[:, -1]

##-----
""" Renombrar y_test y y_pred """
Y_test = y_test_no_scale
Z_test = z_test_no_scale
W_test = w_test_no_scale

Y_pred = y_pred_no_scale
Z_pred = z_pred_no_scale
W_pred = w_pred_no_scale

##-----

plt.ion()

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(12,4)
fig.subplots_adjust(left=0.1, right=0.98, wspace=0.25)

ax1.plot(Y_test,Y_pred, 'ro')
ax2.plot(Z_test,Z_pred, 'bo')
ax3.plot(W_test,W_pred, 'go')

ax1.grid()
ax2.grid()
ax3.grid()

ax1.set_title ("Predicción del par")
ax2.set_title("Predicción de la potencia")
ax3.set_title("Predicción del consumo")

```

```

ax1.set_ylabel("Par Pronosticado (Nm)")
ax2.set_ylabel("Potencia Pronosticada (kW)")
ax3.set_ylabel("Consumo Pronosticado (L/h)")

ax1.set_xlabel("Par Registrado (Nm)")
ax2.set_xlabel("Potencia Registrada (kW)")
ax3.set_xlabel("Consumo Registrado (L/h)")

coeff_y = np.polyfit(Y_test,Y_pred, 1)
m_y = coeff_y[0]
b_y = coeff_y[1]
Y_test_tend = np.linspace(Y_test.min(), Y_test.max(), 100)
Y_pred_tend = m_y * Y_test_tend + b_y

coeff_z = np.polyfit(Z_test,Z_pred, 1)
m_z = coeff_z[0]
b_z = coeff_z[1]
Z_test_tend = np.linspace(Z_test.min(), Z_test.max(), 100)
Z_pred_tend = m_z * Z_test_tend + b_z

coeff_w = np.polyfit(W_test,W_pred, 1)
m_w = coeff_w[0]
b_w = coeff_w[1]
W_test_tend = np.linspace(W_test.min(), W_test.max(), 100)
W_pred_tend = m_w * W_test_tend + b_w

ax1.text (2000, 4000, f"$R^2$= {r2_score(Y_test, Y_pred):.2f} ", fontsize = 12)
ax2.text (350, 800, f"$R^2$= {r2_score(Z_test, Z_pred):.2f} ", fontsize = 12)
ax3.text (200, 380, f"$R^2$= {r2_score(W_test, W_pred):.2f} ", fontsize = 12)

ax1.plot(Y_test_tend,Y_pred_tend, 'k')
ax2.plot(Z_test_tend,Z_pred_tend, 'k')
ax3.plot(W_test_tend,W_pred_tend, 'k')
plt.show(block=True)

```

Anexo XIV. Base de datos depurado de prueba dinamométrica

Aplicacion del Pedal del Acelerador (Por Ciento)	Presion Barometrica del Aire (kPa)	Presion de Aceite del Motor (kPa)	Presion del Multiple de Admision (kPa)	Presion del Riel de Combustible Ordenada (bar)	Presion del Riel de Combustible Medida (bar) Izq	Presion del Riel de Combustible Medida (bar) Der	Temperatura del Combustible (Â°C) Izq	Temperatura del Combustible (Â°C) Der	Temperatura de Aire del Multiple de Admision (Â°C)	Temperatura del Aceite del Motor (Â°C)	Temperatura del Refrigerante del Motor (Â°C)	Velocidad del Motor (RPM)	Torque instantaneo (Nm)	Potencia instantanea (kW)	Combustible Instantaneo (L/hr)
100	100.7	434.1	6.9	900	896	899	77	72.7	45.7	97.2	98.8	2251	826	190.9	90.1
100	100.6	434.6	6.1	900	891	900	77	72.7	45.8	97.4	98.9	2253	826	197.9	89
100	100.5	435.4	6.7	900	894	903	77	72.7	45.8	97.4	99.1	2252	826	198.6	93
100	100.7	434.1	5.9	900	903	899	77	72.7	45.8	97.5	99.1	2253	826	191.6	89.9
100	100.6	434.7	6.5	900	899	903	77	72.7	45.9	97.5	99.1	2251	827	201.2	89.3
100	100.6	433.2	6.4	900	903	906	77.1	72.7	46	97.5	98.6	2251	830	201.6	94
100	100.4	433.3	6.3	900	904	893	77	72.7	46	97.8	98.2	2251	832	192.6	90.3
100	100.5	434	6.5	900	893	894	77	72.9	46	97.8	97.8	2252	833	191.5	90.1
100	100.4	432.4	6.1	900	890	899	77	72.7	46.1	97.8	97.5	2251	833	190.5	89
100	100.5	432.3	6.5	900	904	900	77	72.7	46.1	97.8	97.2	2254	831	191.8	88.8
100	100.6	434.6	5.9	900	901	897	77	72.7	46.2	97.8	96.7	2251	826	194.1	94.7
100	100.5	433.3	5.8	900	898	899	77	72.9	46.2	98.1	96.6	2250	822	191.7	90.8
100	100.5	431.9	6.5	900	892	902	77	72.9	46.2	98.1	96.5	2250	820	197.7	88.8
100	100.5	431.9	5.9	900	893	891	77	72.9	46.3	98.1	96.3	2251	820	206.9	96.8
100	100.8	432.6	6.3	900	905	904	77	72.9	46.3	98.1	96.3	2254	824	199.9	96.7
100	100.6	432.8	6.5	900	900	894	77	72.9	46.3	98.2	96.3	2251	830	201.4	89.6
100	100.7	432.4	6.2	900	891	895	76.8	72.9	46.4	98.4	96.2	2251	840	209.1	93.4
100	100.8	431.4	5.9	900	900	901	76.8	72.9	46.4	98.4	96	2250	848	198.8	98
100	100.9	432	5.6	900	901	901	76.8	73	46.4	98.4	96	2254	851	190.7	88.9
100	100.9	432.8	6.2	900	902	896	76.8	73	46.4	98.4	96.3	2251	852	191.5	89.4
100	100.5	432.2	6.1	900	897	889	76.8	73	46.5	98.5	96.5	2253	849	190.7	89
100	100.6	432.3	6.3	900	894	895	76.8	73	46.5	98.6	96.9	2251	840	191.3	93.4
100	100.4	432.8	6.7	900	904	894	77	73	46.5	98.7	97.4	2251	831	202	89.3
100	100.8	431.6	6.4	900	894	898	76.8	73	46.5	98.8	97.8	2251	827	200.3	98.9
100	100.5	431.4	6.7	900	895	894	76.8	73	46.5	98.8	98.4	2254	824	190.1	93.2
100	100.5	430.8	6.4	900	893	900	76.8	73	46.6	98.8	98.8	2251	825	198.8	89.3
100	100.6	430.7	6.3	900	899	899	77	73	46.6	99	98.9	2254	831	207.1	96.5
100	100.8	431.6	5.6	900	907	895	77	73	46.6	99.1	99.1	2254	836	200.5	97.2
100	100.8	431.4	5.8	900	899	900	77	73	46.7	99.1	99.4	2251	838	193	89.9
100	100.7	431	5.8	900	893	894	77	73.2	46.7	99.1	99.4	2251	841	190.9	89.7
100	100.8	430.3	5.6	900	897	905	77	73	46.8	99.2	99.7	2254	841	190.3	89.3
100	100.6	432.3	5.9	900	902	901	77	73.2	46.8	99.2	99.7	2255	835	190.5	89.2
100	100.5	430.8	6.6	900	894	903	77	73.2	46.8	99.4	99.8	2250	829	198.4	89.1
100	100.4	430.5	6.4	900	900	907	77	73.2	46.8	99.4	99.9	2252	826	198.6	97.2
100	100.6	429.7	5.8	900	896	898	77	73.2	46.9	99.4	100	2251	822	189.4	93.4
100	100.4	430.4	5.9	900	897	904	77	73.2	46.9	99.6	100	2253	819	189.2	93.2
100	100.4	430.9	7	900	895	899	77	73.2	46.9	99.7	100	2250	821	197.7	88.9
100	100.4	431.3	6.3	900	898	890	77	73.4	46.9	99.7	100.2	2254	823	200.3	96.3
100	100.6	430.5	6.4	900	892	896	77.1	73.4	46.9	99.7	100.3	2252	823	194.5	94
100	100.5	429.3	6.1	900	893	901	77.1	73.4	47	99.7	100.4	2250	826	200.7	90.6
100	100.5	428.1	6.3	900	895	905	77	73.4	47	99.7	100.4	2251	832	206.9	93
100	100.4	428.1	6	900	894	897	77	73.4	47	99.8	100.4	2251	837	199.4	96.8
100	100.7	428.7	6.1	900	896	894	77	73.4	47	100	100.4	2251	841	199.4	89.6
100	100.7	428.2	6.1	900	895	901	77	73.4	47	100	100.4	2254	846	197.1	97
100	100.7	427.4	5.7	900	893	900	77	73.4	47	100	100.4	2251	848	188.8	92.4
100	100.8	428.7	6.3	900	894	900	77	73.4	47	100	100.4	2253	844	191.1	89.4

100	100.8	427.9	6.8	900	895	903	77	73.4	47	100	100.4	2251	838	190.3	89.5
100	100.6	426.9	6.1	900	902	904	77	73.6	47	100.1	100.5	2253	830	189.2	88.2
100	100.5	429.3	6.3	900	902	895	77	73.6	47.1	100.4	100.7	2254	821	190.9	94
100	100.4	428.2	7	900	894	897	77	73.6	47.1	100.4	100.8	2250	816	200.3	89.6
100	100.4	427	6.4	900	902	891	77	73.6	47.1	100.3	100.8	2254	815	201.2	93.2
100	100.6	426.4	6.6	900	898	901	77	73.6	47.1	100.4	100.8	2251	817	193.7	90.3
100	100.6	426.4	6.6	900	889	907	77	73.6	47.1	100.4	100.8	2251	820	192	90.4
100	100.6	427	6.6	900	897	903	77	73.6	47.1	100.4	100.8	2254	826	199	89.3
100	100.4	427.8	6.3	900	891	896	77	73.6	47.2	100.8	100.8	2252	830	199.5	97.1
100	100.8	426.5	6.8	900	900	904	77	73.6	47.3	100.8	100.8	2251	831	190.9	93.1
100	100.6	427.7	6.3	900	891	894	77	73.6	47.3	100.8	100.8	2252	830	190.7	88.9
100	100.6	427.2	6.2	900	902	893	77	73.8	47.3	100.8	100.8	2254	831	200.5	88.8
100	100.6	426.3	6.7	900	895	893	77	73.8	47.3	100.8	100.9	2254	831	209.1	98.5
100	100.8	425.1	5.9	900	896	900	77	73.8	47.3	100.9	100.9	2251	832	198.2	97.5
100	100.5	425.4	6.6	900	899	900	77	73.8	47.3	101.2	101	2251	836	198	88.3
100	100.4	425.9	6.7	900	890	900	77	73.9	47.3	101.2	101.2	2253	839	184.9	95.7
100	100.7	426.7	5.7	900	891	897	77	73.9	47.2	101.2	100.9	2262	833	155	75.1
100	100.6	429.2	4.5	900	894	896	77.1	73.9	47.2	101.2	101	2262	813	136.4	68
100	100.6	425.8	3.8	900	896	904	77	73.9	47.2	101.2	101.2	2261	780	121	58.5
84	100.8	427	3.6	900	900	897	77	73.8	47.2	101.2	101	2261	732	119.7	54.8
63	100.4	424.3	2.5	900	899	904	77	73.9	47.2	101.6	101.2	2245	675	124	56.5
58	100.7	425.9	2.7	900	898	908	77	73.9	47.2	101.6	101.2	2242	620	123.6	58.3
60	100.5	424.5	2.8	900	901	897	77	73.9	47.2	101.6	101.2	2249	576	123.8	55.3
49	100.7	423.5	2.5	900	892	915	77	73.9	47.2	101.6	101.2	2239	536	62.9	57.4
30	100.5	409.6	4.2	900	891	899	77	73.9	47.2	101.6	101.2	2033	489	0	0
11	100.8	373.1	4	823	844	798	76.8	73.9	47.2	101.6	101.2	1819	430	0	0
0	100.5	360.6	3.2	700	685	792	76.1	73.8	47.2	101.7	101.2	1474	356	0	0
0	100.6	335	2.7	627	641	610	75.5	73.6	47.2	101.9	101.2	1327	263	0	0
0	100.7	293.7	2.2	586	587	576	75.2	73.2	47.2	101.9	101.2	1186	169	0	0
0	100.5	253.2	1.8	500	483	479	74.6	73	47.2	101.7	101.2	1064	95	0	0
0	100.6	214.2	1	500	490	495	74.1	72.7	47.2	101.6	101.2	944	43	0	0
0	100.5	182.3	0.9	488	493	487	73.6	72.3	47.2	101.6	101.2	844	11	0.2	0
0	100.5	126.4	0.6	400	383	382	72.7	72.1	47.2	101.6	101.2	660	5	8.1	0.1
0	100.4	113.1	0.3	400	381	382	72.3	71.6	47.2	101.3	100.9	641	19	16.2	11
0	100.6	115.8	-0.4	400	396	398	72	71.4	47.2	101.2	100.8	661	43	15.6	10.8
0	100.8	118.8	-0.9	400	403	398	71.6	71.1	47.1	101.2	100.8	663	74	14.3	9.6
0	100.6	119.5	-0.7	400	403	396	71.3	70.9	47.1	101.1	100.8	662	114	13.9	9.4
0	100.7	120.3	-1.4	400	399	401	70.5	70.9	47.1	101.2	100.8	662	152	13.7	9.4
0	100.6	120.3	-0.8	400	400	396	70.2	70.5	47.1	101.2	100.4	664	180	16.3	11.2
0	100.6	120.4	-1.4	400	401	402	70	70.4	47.1	101	100.4	663	200	16.2	12.3
0	100.6	120.3	-1.2	400	404	400	69.7	70.2	47.1	100.9	100.4	661	212	16	9.2
0	100.7	120.7	-1.7	400	402	399	69.6	70.2	47	100.9	100.4	661	217	16.2	12.6
0	100.7	120.8	-1.2	400	405	402	69.3	70	47	101.1	100.3	662	221	16.5	9.3
0	100.6	120.9	-1.5	400	401	401	69.2	69.9	47	100.9	100.3	666	225	16.3	12.5
0	100.6	121.2	-1.2	400	401	398	69	69.9	47	101.2	100.3	661	228	13.7	9.2
0	100.5	120.9	-1.3	400	403	397	68.9	69.7	46.9	101.2	100	665	230	16.2	9.5
0	100.6	121.5	-1.5	400	401	403	68.7	69.7	46.9	101.2	100	662	230	16.5	10.4
0	100.8	121.7	-1.3	400	403	401	68.6	69.7	46.9	101.1	100	661	227	14.1	9.4
0	100.7	121.8	-1.2	400	400	397	68.5	69.6	46.9	101.2	100	662	225	16.5	9.4
0	100.7	121.4	-1.5	400	401	397	68.3	69.4	46.8	101.2	100	660	226	16.7	10.8
0	100.6	121.8	-1.3	400	400	401	68.2	69.4	46.8	101.2	100	661	225	14.1	9.7
0	100.6	122.2	-1.6	400	402	398	68.2	69.4	46.8	101.2	100	662	225	16.2	11.3
0	100.6	121.9	-1.2	400	403	401	68	69.3	46.8	101.2	100	661	228	18.8	12.5
0	100.6	121.7	-1.6	400	402	402	67.9	69.3	46.8	101.2	99.9	665	231	16.5	10.4
0	100.8	122.5	-1.6	400	398	397	67.8	69.3	46.7	101.2	99.7	666	231	14.1	9.6
0	100.5	122.6	-1.5	400	400	401	67.8	69.2	46.7	101.2	99.7	666	232	14.1	9.4
0	100.6	122.3	-1.5	400	402	397	67.6	69.2	46.7	101.2	99.7	663	232	14.1	9.3
0	100.5	122.3	-1.5	400	400	402	67.6	69.2	46.7	101.2	99.7	662	227	16.5	11.4
0	100.6	122.4	-1.3	400	402	400	67.5	69.2	46.6	101.2	99.7	661	223	16.7	10.8
0	100.6	122.9	-1	400	402	398	67.3	69.2	46.6	101.2	99.7	666	222	16.7	9.8
0	100.8	122.7	-1.5	400	402	400	67.2	69	46.6	101.2	99.7	666	222	16.3	12.5
0	100.7	122.3	-1.1	400	401	399	67	69	46.6	101.3	99.7	662	224	13.9	9.3

0	100.6	122.3	-1.4	400	404	399	67	69	46.5	101.2	99.7	660	225	14.3	9.6
0	100.7	122.4	-1.1	400	400	401	66.9	68.9	46.5	101.3	99.7	665	227	16.7	9.6
0	100.8	122.9	-1.5	400	399	401	66.8	68.9	46.5	101.5	99.7	666	226	16.5	10.5
0	100.7	122.9	-1.2	400	399	402	66.6	68.9	46.4	101.4	99.7	661	225	16.5	9.3
0	100.7	123.6	-1.6	400	405	403	66.6	68.9	46.4	101.4	99.6	665	226	16.9	12.8
0	100.8	123.3	-1.7	400	399	399	66.5	68.7	46.3	101.4	99.6	665	227	14.5	9.9
0	100.6	123.8	-1.2	400	402	403	66.5	68.7	46.3	101.4	99.7	666	229	16.5	9.5
0	100.7	123.6	-1.5	400	401	399	66.5	68.7	46.3	101.4	99.5	664	231	16.5	12.6
0	100.6	123.1	-2.1	400	405	402	66.3	68.7	46.2	101.5	99.4	664	233	16.7	9.4
0	100.8	123.8	-1.3	400	398	404	66.2	68.6	46.2	101.6	99.5	666	234	19	10.7
0	100.9	124.1	-1.4	400	399	401	66.2	68.6	46.2	101.5	99.4	665	236	16.7	10.5
0	100.6	123.8	-1.3	400	401	399	66.2	68.6	46.2	101.6	99.4	663	238	14.3	9.6
0	100.6	123.8	-1.5	400	398	405	66.1	68.6	46.1	101.6	99.4	666	238	14.1	11.4
0	100.8	123.4	-1.4	400	405	401	66.1	68.5	46.1	101.6	99.4	665	236	14.3	11.7
0	100.7	123.6	-1.6	400	402	405	65.9	68.5	46	101.6	99.4	666	231	14.5	9.8
0	100.8	124.2	-1.8	400	403	402	65.8	68.5	46	101.6	99.4	663	224	16.7	9.6
0	100.8	123.8	-1.5	400	399	405	65.8	68.5	46	101.6	99.4	666	220	16.7	10.7
0	100.6	123.7	-1.5	400	402	399	65.8	68.5	46	101.6	99.4	662	218	14.5	9.7
0	100.8	124.2	-1.6	400	401	403	65.8	68.3	45.9	101.6	99.4	665	217	14.3	11.9
0	100.9	124.3	-1.5	400	402	404	65.6	68.3	45.9	101.6	99.3	665	219	14.1	9.4
0	100.7	124.2	-1.2	400	401	404	65.6	68.3	45.9	101.6	99.4	662	220	16.5	9.3
0	100.8	124.1	-1.7	400	403	398	65.5	68.3	45.8	101.6	99.4	662	220	16.7	10.7
0	100.8	124.2	-1.4	400	400	403	65.5	68.3	45.8	101.6	99.4	665	219	14.5	9.9
0	100.7	124.6	-1.2	400	400	401	65.4	68.2	45.8	101.6	99.3	666	220	16.7	11.7
0	100.7	124.5	-1.4	400	404	402	65.4	68.2	45.7	101.6	99.2	661	225	18.8	12.6
0	100.7	124.4	-1.2	400	399	406	65.4	68.2	45.7	101.6	99.2	661	229	16.7	10.7
0	100.7	125	-1.5	400	400	398	65.2	68.2	45.6	101.6	99.1	665	232	14.3	9.7
0	100.8	124.8	-1.3	400	402	403	65.2	68.2	45.6	101.6	99.1	662	234	14.1	11.5
0	100.8	124.9	-1.7	400	402	402	65.1	68.2	45.5	101.6	99.1	664	234	14.1	9.3
0	100.8	124.8	-1.8	400	401	403	65.1	68.2	45.5	101.6	99.1	661	228	14.1	9.4
0	101	125	-1.6	400	401	399	65.1	68.2	45.5	101.6	99.1	663	221	14.1	11.7
0	100.6	125	-1.8	400	397	404	65.1	68.2	45.5	101.6	99.1	666	215	14.1	11.7
0	100.6	125.5	-1.6	400	402	401	65.1	68	45.4	101.6	99.1	661	209	13.9	9.4
0	100.8	125.3	-1.3	400	400	405	65.1	68	45.4	101.6	99.1	661	205	13.9	9.4
0	100.9	124.8	-1.5	400	404	403	65.1	68	45.3	101.6	99.1	665	203	14.5	11.9
0	100.7	125.4	-1.5	400	400	402	65.1	68	45.3	101.6	99.1	665	203	14.7	9.9
0	100.8	125.6	-1.3	400	399	399	65.2	68	45.3	101.6	99.1	661	205	16.9	9.7
0	100.8	125.5	-1.8	400	403	401	65.1	68	45.3	101.6	99.1	662	209	16.7	12.7
0	100.8	124.8	-1.4	400	405	402	65.1	67.9	45.2	101.6	99.1	665	213	14.3	9.4
0	100.7	125.5	-1.5	400	400	404	65.1	67.9	45.2	101.6	99.1	661	217	14.3	11.9
0	100.9	125.7	-1.5	400	400	403	65.1	67.9	45.2	101.6	99.1	662	221	16.7	11.6
0	100.7	125.6	-1.8	400	402	405	65.1	67.9	45.1	101.6	98.9	666	225	19.2	12.7
0	100.8	125.8	-1.8	400	399	403	64.9	67.9	45.1	101.6	99.1	662	228	16.7	10.8
0	100.7	125.5	-1.7	400	405	400	64.9	67.9	45.1	101.6	99	661	232	16.7	9.6
0	100.7	126	-1.3	400	402	402	64.9	67.9	45	101.6	98.9	665	238	19.2	10.8
0	100.8	126	-1.4	400	403	402	64.9	67.8	45	101.6	98.9	662	245	19	12.7
0	100.8	125.9	-1.3	400	400	405	64.9	67.9	45	101.6	98.8	661	249	16.5	10.7
0	100.8	126	-1.5	400	401	399	64.9	67.8	44.9	101.6	98.8	662	252	14.3	9.6
0	100.8	126.1	-1.4	400	400	402	64.9	67.8	44.9	101.6	98.8	661	252	14.3	11.9
0	100.7	127	-1.2	400	399	400	64.9	67.8	44.9	101.6	98.8	665	246	14.1	11.6
0	100.7	126.1	-1.4	400	399	402	64.9	67.8	44.8	101.6	98.8	661	237	16.5	9.4
0	100.9	126	-1.7	400	403	403	64.8	67.8	44.8	101.4	98.8	665	231	19.2	13
0	100.8	126.7	-1.7	400	398	404	64.8	67.6	44.7	101.6	98.8	665	229	17.1	10.9
0	100.7	126.8	-1.3	400	399	397	64.9	67.6	44.7	101.6	98.8	666	228	14.5	9.7
0	100.7	126.4	-1.5	400	402	402	64.9	67.6	44.7	101.5	98.8	666	229	14.5	9.7
0	100.8	126.6	-1.6	400	402	404	64.8	67.6	44.6	101.6	98.8	662	232	14.5	9.7
0	100.7	126.8	-1.2	400	400	401	64.9	67.6	44.6	101.5	98.8	662	230	14.5	11.9
0	100.8	126.9	-1.5	400	403	401	64.9	67.6	44.6	101.4	98.8	666	225	14.5	9.7
0	100.9	126.4	-1.5	400	402	405	64.9	67.6	44.5	101.5	98.8	661	220	14.5	9.6
0	100.8	126.2	-1.4	400	402	402	64.9	67.6	44.5	101.4	98.8	662	215	14.5	9.7
0	100.8	126.6	-1.9	400	403	403	64.9	67.5	44.5	101.3	98.8	665	211	14.7	9.9
0	100.7	127.4	-1.5	400	400	403	64.9	67.5	44.5	101.5	98.8	665	210	14.7	11.9

0	100.8	127	-1.7	400	403	404	64.9	67.5	44.5	101.4	98.8	661	210	14.5	11.6
0	100.8	127.1	-1.7	400	400	404	64.9	67.5	44.4	101.5	98.8	662	210	14.5	11.7
0	100.6	127.4	-1.2	400	403	398	64.9	67.5	44.4	101.4	98.8	665	210	14.5	9.9
0	100.7	127.4	-1.2	400	400	401	64.9	67.5	44.4	101.3	98.7	662	210	14.3	11.8
0	100.7	126.7	-1.3	400	400	400	64.8	67.5	44.3	101.2	98.6	665	210	14.1	11.6
0	100.8	126.9	-1.5	400	402	403	64.8	67.5	44.3	101.2	98.7	661	209	14.3	11.6
0	100.8	126.6	-1.5	400	404	403	64.8	67.3	44.3	101.2	98.5	665	208	14.3	9.7
0	100.8	127.3	-1.4	400	402	400	64.8	67.3	44.3	101.2	98.6	662	208	16.7	11.7
0	100.9	127.6	-1.7	400	401	399	64.6	67.3	44.2	101.2	98.5	665	210	16.7	10.6
0	100.7	127	-1.6	400	402	402	64.8	67.3	44.2	101.2	98.6	666	213	14.1	9.4
0	100.7	126.7	-1.2	400	402	402	64.8	67.3	44.2	101.2	98.6	661	216	16.7	9.5
0	100.8	127.5	-1.4	400	398	402	64.9	67.3	44.1	101.3	98.5	665	222	16.9	13
0	100.8	127.8	-1.7	400	403	402	65.1	67.3	44.1	101.2	98.5	666	225	14.5	9.6
0	100.8	127.6	-1.5	400	402	402	65.1	67.2	44.1	101.2	98.5	663	226	16.7	9.7
0	100.6	127.4	-1.4	400	400	401	65.1	67.3	44	101.2	98.5	661	229	16.5	10.8
0	100.7	127.1	-1.9	400	402	403	65.2	67.2	44	101.2	98.5	665	230	14.3	9.8
0	100.7	128	-1.5	400	401	403	65.2	67.2	44	101.2	98.5	666	228	17.1	9.8
0	100.7	127.4	-1.6	400	401	399	65.2	67.2	44	101.2	98.5	663	230	16.9	12.9
0	100.7	127.5	-1.4	400	404	401	65.2	67.2	43.9	101.2	98.5	660	230	14.1	9.5
0	100.7	127.9	-1.6	400	403	398	65.2	67.2	43.9	101.2	98.5	664	227	14.3	9.7
0	100.8	128.2	-1.4	400	399	403	65.2	67.2	43.9	101.2	98.5	666	226	14.5	11.7
0	100.9	127.4	-1.6	400	402	401	65.2	67.2	43.9	101.2	98.5	666	223	14.5	9.5
0	100.8	127.3	-1.7	400	399	403	65.1	67.2	43.8	101.2	98.4	666	219	16.7	9.6
0	100.8	127.6	-1.6	400	403	400	65.1	67.2	43.8	101.2	98.4	666	219	19	12.9
0	100.8	128.3	-1.4	400	402	402	65.1	67.2	43.8	101.2	98.4	662	222	16.9	10.8
0	100.6	127.6	-1.7	400	403	402	65.1	67.2	43.8	101.2	98.4	666	226	16.9	9.7
0	100.6	127.6	-1.4	400	401	402	65.1	67	43.8	101.1	98.4	663	233	16.7	10.8
0	100.7	127.7	-1.5	400	401	399	65.1	67.2	43.7	101.2	98.4	665	240	16.7	9.5
0	100.8	128.3	-1.6	400	400	402	65.1	67	43.6	101	98.4	666	242	16.9	13
0	100.8	128	-1.8	400	405	401	65.1	67	43.6	101.2	98.4	666	242	14.5	9.7
0	100.8	127.4	-1.4	400	400	405	64.9	67	43.6	101.2	98.4	662	240	14.5	9.7
0	100.6	127.3	-1.4	400	400	398	64.9	67	43.6	101.2	98.4	661	234	14.3	11.8
0	100.8	127.9	-1.6	400	402	398	64.9	67	43.6	101.2	98.4	660	228	14.3	9.9
0	100.8	128.9	-1.4	400	396	402	64.9	67	43.6	101.2	98.4	666	221	14.5	11.7
0	100.6	128.5	-1.4	400	402	400	64.9	67	43.5	101.2	98.4	666	216	14.3	11.7
0	100.8	128.3	-1.2	400	402	401	64.8	66.9	43.5	101.2	98.4	664	213	16.7	9.6
0	100.9	127.8	-1.8	400	403	402	64.8	66.9	43.5	101.1	98.4	665	214	19.2	10.9
0	100.9	128	-1.5	400	400	398	64.8	66.9	43.5	100.9	98.3	662	218	16.7	10.7
0	100.6	128.5	-1.2	400	402	398	64.8	66.9	43.4	101	98.3	661	223	14.3	9.6
0	100.8	128.2	-1.5	400	399	402	64.8	66.9	43.4	101.1	98.1	661	229	14.5	11.7
0	100.9	127.7	-1.8	400	401	398	64.9	66.9	43.4	100.9	98.3	660	232	14.3	9.8
0	100.8	128.5	-1.4	400	402	399	64.9	66.9	43.4	100.9	98.3	666	230	16.7	11.6
0	100.7	128.6	-1.7	400	403	397	64.9	66.9	43.4	101	98.3	661	228	16.9	10.6
0	100.7	128	-1.6	400	400	402	64.9	66.9	43.4	100.8	98.2	663	225	14.5	9.5
0	100.6	127.9	-1.7	400	400	397	64.9	66.9	43.3	100.8	98.1	661	224	16.9	11.8
0	100.8	129	-1.5	400	400	402	64.9	66.9	43.3	100.8	98.1	665	226	16.9	10.8
0	100.8	128.7	-1.6	400	402	400	64.9	66.8	43.3	100.8	98.1	664	228	14.5	9.6
0	100.9	128.5	-1.7	400	402	403	64.9	66.9	43.3	100.8	98.1	666	228	14.3	11.6
0	100.8	127.4	-1.7	400	403	400	64.9	66.9	43.3	100.8	98.1	661	228	14.1	9.5
0	100.8	128.2	-1.7	400	400	398	64.9	66.8	43.2	100.8	98.1	664	224	14.1	9.6
0	100.7	128.9	-1.2	400	400	402	65.1	66.9	43.2	100.8	98.1	664	218	14.3	9.5
0	100.8	128.7	-1.8	400	398	402	65.2	66.8	43.2	100.8	98.1	666	215	16.9	9.7
0	100.8	127.9	-1.8	400	399	398	65.2	66.8	43.1	100.8	98.1	665	214	16.7	10.9
0	100.8	128.3	-1.7	400	399	397	65.1	66.8	43.1	100.8	98.1	664	214	14.1	9.8
0	100.8	129	-1.4	400	399	399	65.1	66.8	43.1	100.8	98.1	662	215	14.1	11.7
0	100.7	128.4	-1.9	400	402	398	65.1	66.8	43.1	100.8	98.1	664	219	16.7	11.6
0	100.6	128.4	-1.5	400	401	403	65.1	66.8	43.1	100.8	98.1	665	222	16.7	10.8
0	100.7	128.7	-1.5	400	399	402	65.1	66.8	43	100.8	98.1	665	221	14.1	11.9
0	100.8	128.8	-1.4	400	400	401	65.2	66.8	43	100.8	98.1	666	223	16.7	11.6
0	100.8	128.8	-1.9	400	400	403	65.1	66.8	43	100.8	98.1	666	225	16.9	10.6
0	100.8	128.7	-1.7	400	398	402	65.1	66.6	43	100.8	98.1	662	225	14.3	9.5
0	100.7	128.8	-1.8	400	400	398	65.1	66.6	43	100.8	98.1	662	226	16.7	11.7

0	100.5	129.3	-1.3	400	400	402	65.1	66.8	43	100.7	98.1	666	229	16.9	10.8
0	100.6	128.8	-1.5	400	404	400	65.1	66.6	42.9	100.6	98.1	662	229	14.5	9.6
0	100.6	128.8	-1.2	400	399	402	65.1	66.6	42.9	100.7	98.1	662	227	14.5	11.6
0	100.8	128.2	-1.5	400	404	401	65.2	66.6	42.9	100.5	98.1	661	226	14.5	9.8
0	100.8	128.4	-1.9	400	399	400	65.2	66.6	42.9	100.4	98.1	665	224	14.5	9.9
0	100.8	129.4	-1.8	400	400	401	65.2	66.6	42.9	100.5	98.1	666	221	16.7	9.7
0	100.9	128.8	-1.7	400	397	398	65.2	66.6	42.9	100.6	98.1	662	219	16.7	12.7
0	100.7	128.8	-1.7	400	403	403	65.2	66.6	42.9	100.6	98.1	665	219	14.5	9.7
0	100.7	128.6	-1.7	400	400	398	65.2	66.6	42.8	100.4	98.1	665	219	14.3	12
0	100.7	129.2	-1.3	400	401	399	65.4	66.6	42.8	100.4	98.1	662	220	14.1	11.6
0	100.7	129	-1.8	400	400	398	65.4	66.5	42.8	100.4	98.1	663	219	14.1	9.4
0	100.9	128.7	-1.7	400	398	401	65.4	66.5	42.8	100.4	98.1	662	217	16.9	11.5
0	100.8	129.1	-1.4	400	396	396	65.5	66.5	42.8	100.4	98.1	661	217	17.1	13.2
0	100.8	129.7	-1.5	400	399	400	65.5	66.5	42.7	100.4	98.1	662	217	14.5	9.7
0	100.7	129	-1.4	400	403	401	65.5	66.5	42.7	100.4	98.1	662	218	14.5	9.7
0	100.5	129.2	-1.5	400	397	400	65.5	66.5	42.7	100.4	98.1	661	220	14.5	9.8
0	100.5	129	-1.1	400	399	398	65.6	66.5	42.7	100.4	98.1	662	221	14.8	9.8
0	100.7	129.6	-1.5	400	397	401	65.6	66.5	42.7	100.4	98.1	666	221	17.1	12
0	100.8	129.4	-1.7	400	400	396	65.6	66.5	42.7	100.3	98.1	664	221	16.7	12.8
0	100.7	129	-1.3	400	399	403	65.6	66.5	42.7	100.3	98.1	661	221	14.3	9.5
0	100.7	129.3	-1.6	400	402	397	65.6	66.5	42.6	100.2	98.1	665	222	17.1	11.9
0	100.7	129.3	-1.3	400	397	400	65.6	66.5	42.6	100.3	98.1	666	226	17.1	10.9
0	100.7	129.5	-1.4	400	400	397	65.6	66.5	42.6	100.1	98.1	666	229	14.8	9.8
0	100.7	129.4	-1.6	400	399	402	65.6	66.5	42.6	100	98	663	230	17.1	9.9
0	100.6	129.4	-1.4	400	403	397	65.6	66.5	42.5	100	98	665	233	16.9	12.9
0	100.8	129.7	-1.6	400	395	400	65.6	66.3	42.5	100	98.1	666	233	14.5	9.8
0	100.6	129.7	-1.4	400	398	397	65.6	66.5	42.5	100	98	665	232	16.9	11.7
0	100.7	129.3	-1.5	400	398	401	65.8	66.5	42.5	100	98.1	666	235	19.2	12.7
0	100.8	129.1	-1.3	400	399	396	65.6	66.5	42.5	100	98.1	665	237	16.7	10.7
0	100.7	129.2	-1.7	400	402	402	65.8	66.3	42.5	100	97.9	665	237	16.7	9.6
0	100.8	129.3	-1.7	400	401	401	65.6	66.5	42.4	100	98	662	241	16.9	12.9
0	100.8	129.1	-1.5	400	401	403	65.8	66.3	42.4	100	98.1	666	243	14.5	9.5
0	100.7	129.3	-1.7	400	400	400	65.6	66.3	42.4	100	98	661	240	14.5	11.8
0	100.8	128.9	-1.6	400	401	399	65.8	66.3	42.4	100	98	663	236	14.5	9.9
0	100.8	129.6	-1.3	400	399	401	65.8	66.3	42.4	100	98	662	232	16.9	9.7
0	100.7	129.1	-1.3	400	399	400	65.8	66.5	42.3	100	97.9	666	227	16.7	12.7
0	100.7	129.5	-1.7	400	401	398	65.8	66.3	42.3	99.8	98	662	223	14.1	9.5
0	100.8	129.4	-1.7	400	399	398	65.6	66.3	42.3	99.8	97.8	663	222	14.1	9.7
0	100.8	129.9	-2	400	399	399	65.6	66.5	42.3	99.9	97.9	662	222	16.7	11.6
0	100.8	130	-1.4	400	402	400	65.6	66.3	42.3	99.9	97.9	662	223	16.9	10.6
0	100.6	129.6	-1.2	400	400	403	65.6	66.3	42.3	99.7	97.8	662	222	14.3	9.5
0	100.7	129.6	-1.6	400	400	398	65.6	66.3	42.3	99.7	97.8	662	222	14.1	9.5
0	100.7	129.8	-1.7	400	395	400	65.6	66.3	42.3	99.7	97.9	666	223	16.7	9.6
0	100.6	130	-1.6	400	401	399	65.6	66.3	42.3	99.7	97.8	666	224	16.9	10.5
0	100.8	129.4	-1.3	400	399	400	65.6	66.3	42.2	99.7	97.8	663	224	16.7	9.7
0	100.8	129.7	-1.6	400	398	395	65.6	66.3	42.3	99.7	97.8	662	226	16.7	10.7
0	100.8	129.5	-1.7	400	400	397	65.6	66.3	42.2	99.7	97.8	663	229	14.3	9.9
0	100.8	129.6	-1.7	400	400	396	65.6	66.3	42.2	99.7	97.8	661	229	14.3	11.7
0	100.8	129.2	-1.7	400	399	399	65.6	66.3	42.2	99.7	97.6	662	230	16.9	9.6
0	100.8	129.1	-1.3	400	402	400	65.6	66.3	42.2	99.7	97	660	230	16.7	11
0	100.7	129.2	-1.7	400	397	396	65.6	66.3	42.1	99.7	96.5	664	227	14.1	11.9
0	100.7	129.4	-1.3	400	399	398	65.6	66.3	42.1	99.7	95.8	662	225	14.3	11.6
0	100.7	129.4	-1.4	400	401	395	65.6	66.2	42.1	99.7	95.3	661	223	14.3	9.5
0	100.7	129.1	-1.4	400	398	402	65.6	66.2	42.1	99.7	95.2	662	222	16.7	9.4
0	100.8	129.6	-1.5	400	399	400	65.6	66.2	42.1	99.7	95.3	661	221	19.2	10.8
0	100.7	130	-1.6	400	397	396	65.6	66.2	42.1	99.7	95.3	665	225	17.1	10.8
0	100.7	129.3	-1.3	400	400	401	65.6	66.2	42.1	99.6	95.3	661	229	17.1	9.9
0	100.5	129.5	-1.2	400	397	402	65.6	66.2	42.1	99.6	95.3	661	236	17.1	11
0	100.7	130.3	-1.6	400	401	401	65.6	66.2	42.1	99.6	95.3	665	243	17.5	10.2
0	100.7	130.8	-1.6	400	398	395	65.6	66.2	42.1	99.5	94.9	665	247	17.5	11.1
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	248	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	247	15	9.9

0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	242	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	236	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	229	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	223	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	219	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	216	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	215	15	9.9
0	100.6	132.4	-1.4	400	400	395	65.6	66.2	42	99.4	93.9	666	217	18	9.9
0	100.7	152.3	-1.8	400	403	403	65.6	66.2	42	99.4	69.7	662	220	18.2	14
0	100.8	151.8	-1.9	400	403	402	65.6	66.2	42	99.4	69.6	665	226	18.2	10.3
0	100.8	151.5	-2.1	400	405	405	65.6	66.2	42	99.4	69.6	664	235	21.2	14.2
0	100.8	152.4	-1.7	400	399	403	65.6	66.2	42	99.4	69.6	665	246	18.6	12
0	100.8	152.3	-1.6	400	405	405	65.6	66.2	42	97.9	69.6	666	255	18.4	10.6
0	100.8	151.5	-2.2	400	405	405	65.6	66.2	41.5	97.8	69.6	661	263	18.6	11.7
0	100.8	151.4	-2	400	400	404	65.2	66.2	41.5	97.8	69.6	665	268	15.8	13.2
0	100.6	152.4	-2	400	400	400	65.2	66.2	41.5	97.8	69.7	665	266	15.4	12.7
0	100.6	152.4	-2	400	400	400	65.2	66.2	41.5	97.8	69.7	665	323	124.2	12.7
74	100.4	480.9	9.5	900	887	901	68.7	64.2	43.6	90	74.1	2245	410	244.1	114.8
74	100.7	480.9	8.8	900	894	891	68.9	64.2	43.6	90	74.1	2242	528	243.9	114.4
74	100.7	481.4	9	900	887	891	68.9	64.2	43.6	90.1	74.1	2244	678	233.4	109.5
74	100.8	482.7	9	900	897	895	68.9	64.2	43.6	90	74.1	2244	862	233.9	114.8
74	100.5	482.5	9.6	900	892	898	68.9	64.2	43.6	90.2	74.1	2244	953	233.2	109.2
74	100.5	482.6	9	900	904	902	68.9	64.2	43.7	90	74.1	2245	1012	233.5	108.9
74	100.5	482.7	8.7	900	889	903	68.9	64.4	43.7	90.1	74.1	2243	1039	235.4	109.4
74	100.6	483.5	8.3	900	893	894	68.9	64.2	43.7	90.2	74.1	2242	1036	236.7	110.4
74	100.6	483.5	9	900	896	894	68.9	64.2	43.7	90.2	74.1	2242	1001	234.3	109.8
74	100.8	483	8.3	900	899	904	68.9	64.2	43.7	90.2	74.1	2245	1001	244.6	108.6
74	100.7	482.9	9.6	900	890	894	68.9	64.2	43.8	90.3	74.1	2242	1006	256.7	114.4
74	100.5	482	9.5	900	898	895	68.9	64.2	43.8	90.3	74.1	2246	1014	245	119
74	100.5	481.9	8.9	900	899	908	68.9	64.2	43.8	90.3	74.1	2245	1021	233	108.9
74	100.6	482.1	9.2	900	896	896	68.9	64.2	43.8	90.3	74.1	2245	1027	232.6	109
74	100.6	482.7	8.9	900	888	898	68.9	64.2	43.8	90.3	74.1	2243	1030	234.1	109
74	100.8	483.3	8.2	900	889	892	68.9	64.2	43.8	90.3	74.1	2242	1025	236.2	109.7
74	100.6	481.5	8.8	900	889	902	68.9	64.2	43.8	90.3	74.1	2246	1020	247.1	109.7
74	100.6	482.8	8.4	900	886	895	68.9	64.2	43.8	90.3	74.1	2245	1018	255.7	115.5
74	100.4	481.2	8.9	900	895	899	68.9	64.2	43.8	90.3	74.1	2242	1019	244.3	114.2
74	100.7	483.3	8.8	900	895	888	68.9	64.4	43.8	90.3	74.1	2242	1022	233.2	109.5
74	100.6	481	8.6	900	898	895	68.9	64.4	43.8	90.3	74.1	2245	1028	233	108.6
74	100.6	482.6	8.3	900	904	902	68.9	64.2	43.8	90.3	74.1	2243	1030	232.8	109.7
74	100.6	482.7	9.5	900	891	900	68.9	64.4	43.8	90.3	74.1	2245	1025	231.3	108.6
74	100.5	482.6	8.8	900	902	891	68.9	64.4	43.8	90.6	74.1	2245	1016	233.7	108.5
74	100.7	482.5	9.4	900	900	899	69	64.4	43.8	90.4	74.1	2242	1008	236.2	109.4
74	100.6	481.4	9.6	900	898	895	69	64.4	43.8	90.5	74.1	2243	1002	244.1	110.7
74	100.6	481.2	8.3	900	893	893	69	64.4	43.8	90.6	74.1	2242	1001	247.8	118.5
74	100.6	481.4	9.4	900	900	904	69	64.4	43.8	90.6	74.1	2240	1008	253.3	120.5
74	100.8	480.7	9.6	912	918	917	69	64.4	43.8	90.6	74.1	2236	1023	271.3	123.9
74	100.6	481.5	11.3	936	947	938	68.9	64.2	43.9	90.6	74.1	2235	1046	284.5	135.3
74	100.5	480.2	12	962	952	947	69	64.4	43.9	90.6	74.1	2230	1076	290.3	135.2
74	100.8	478.9	12.6	959	956	970	69	64.4	43.9	90.6	74.1	2232	1111	292.7	136.5
74	100.5	478.4	13	963	969	963	69	64.5	43.9	90.6	74.1	2232	1149	297.1	138
74	100.7	481.8	13.2	966	968	971	69.2	64.5	44	90.6	74.1	2232	1186	299.3	147.3
74	100.7	481.6	13.8	964	981	992	69.2	64.5	44	90.6	74.1	2232	1218	300.3	139.7
74	100.6	480.4	13.2	983	969	970	69.3	64.5	44	90.6	74.1	2229	1245	317.7	141.7
74	100.6	479.9	13.6	990	975	978	69.2	64.5	44	90.6	74.1	2227	1270	320.9	155.7
74	100.5	481.8	14.7	976	978	978	69.3	64.5	44	90.6	74.1	2231	1290	308.5	144.4

74	100.6	478.1	14.2	993	980	987	69.3	64.5	44	90.6	74.1	2227	1306	307.6	144.1
74	100.5	479.7	14.3	989	988	986	69.3	64.5	44.1	90.6	74.1	2228	1320	307.8	143.2
74	100.7	480.2	14.4	991	993	999	69.3	64.6	44.1	90.6	74.1	2228	1332	328.2	144.6
74	100.7	478.1	15.1	1022	999	1005	69.4	64.6	44.1	90.6	74.1	2224	1340	330.7	161.1
74	100.7	479.8	15.5	996	1008	1017	69.4	64.6	44.1	90.8	74.2	2229	1348	314.9	146.7
74	100.5	478	15.7	1000	1004	995	69.4	64.6	44.1	90.8	74.3	2228	1356	319	148
74	100.7	480.1	16	1015	995	1001	69.6	64.6	44.1	90.7	74.2	2224	1365	324.7	150.4
74	100.7	478.8	16	1031	1009	1023	69.6	64.6	44.1	90.9	74.3	2222	1374	330.9	153.6
74	100.6	478.1	16.4	1042	1038	1042	69.6	64.6	44.1	90.9	74.3	2220	1382	337.8	156.6
74	100.7	477.1	17.3	1052	1042	1034	69.7	64.8	44.1	90.9	74.3	2220	1393	340.6	158.8
74	100.5	478.6	18.4	1037	1039	1044	69.9	64.8	44.1	90.9	74.3	2224	1407	340.3	158.9
74	100.7	476.3	18.2	1041	1049	1055	69.9	64.8	44.1	90.9	74.3	2223	1420	339.9	159.7
74	100.6	476.2	18.5	1042	1043	1041	69.9	64.8	44.1	90.9	74.2	2223	1434	340.6	158.8
74	100.7	476.3	18.4	1042	1044	1049	70	64.8	44.2	90.9	74.3	2224	1451	366.8	159.7
74	100.7	477.3	19.1	1072	1059	1046	70	64.8	44.2	90.9	74.3	2219	1471	377.5	172.4
74	100.6	476.6	20.2	1076	1088	1077	70.2	64.9	44.2	90.9	74.3	2219	1490	364.3	169.8
74	100.7	478.8	20.3	1097	1076	1088	70.2	64.9	44.3	90.9	74.3	2215	1512	366.2	171.2
74	100.7	477.8	20.7	1087	1076	1076	70.4	64.9	44.3	90.9	74.3	2218	1540	384.4	172.7
74	100.8	475.4	21.2	1086	1075	1081	70.4	65.1	44.3	90.9	74.3	2218	1566	382.9	179.9
74	100.7	475.6	21.1	1097	1088	1089	70.5	65.1	44.4	90.9	74.3	2216	1584	364.7	170.8
74	100.8	477.8	20.9	1093	1086	1073	70.5	65.1	44.4	91	74.3	2216	1597	364.7	170.4
74	100.8	477.1	21.1	1077	1091	1088	70.7	65.1	44.4	91	74.3	2219	1606	367.7	170.5
74	100.7	475.8	21.4	1100	1091	1080	70.7	65.1	44.5	91	74.3	2214	1605	370.3	182.2
74	100.6	476	21.3	1086	1098	1084	70.9	65.2	44.5	91.2	74.4	2219	1598	367.5	173.1
74	100.7	475.8	21.4	1082	1075	1076	71.1	65.2	44.5	91.1	74.3	2218	1594	369	171
74	100.7	475.8	21.1	1100	1101	1085	71.1	65.1	44.6	91.2	74.3	2215	1591	370.5	172.6
74	100.6	474.6	21.5	1091	1087	1099	71.1	65.2	44.6	91.3	74.4	2217	1588	368.3	172.1
74	100.6	475.2	21.5	1085	1096	1095	71.3	65.2	44.6	91.3	74.3	2219	1586	367.7	172.3
74	100.7	476.3	21.4	1074	1103	1081	71.3	65.2	44.6	91.3	74.3	2219	1591	385.6	171.4
74	100.7	475.9	20.9	1100	1095	1092	71.3	65.2	44.6	91.3	74.3	2214	1598	388.6	180.9
74	100.8	475.8	21.4	1100	1096	1071	71.3	65.4	44.7	91.2	74.3	2215	1604	369	174.5
74	100.7	473.9	21.2	1085	1094	1097	71.4	65.4	44.7	91.3	74.3	2218	1609	365.8	171.1
74	100.7	475.5	20.9	1100	1083	1081	71.4	65.4	44.7	91.3	74.3	2215	1615	366.6	172.4
74	100.7	476.3	21.1	1085	1082	1107	71.4	65.4	44.7	91.3	74.3	2219	1614	366.2	171.3
74	100.6	476.7	20.8	1091	1081	1079	71.6	65.4	44.8	91.3	74.3	2217	1607	367.1	171.6
74	100.7	476.4	21.1	1094	1081	1081	71.4	65.4	44.8	91.3	74.1	2216	1599	369.6	172.7
74	100.7	475.9	21.4	1095	1087	1079	71.4	65.5	44.8	91.3	74.1	2214	1592	366.8	173.4
74	100.6	474.3	21.9	1099	1084	1076	71.4	65.5	44.9	91.3	74.1	2215	1585	366	178.4
74	100.6	477	21.3	1100	1095	1094	71.4	65.5	44.9	91.4	74.1	2214	1582	369	173.1
74	100.6	475.1	21.3	1086	1091	1096	71.4	65.5	44.9	91.3	74	2219	1583	367.1	172.3
74	100.7	476	21.7	1091	1079	1073	71.4	65.5	45	91.4	73.9	2217	1587	385.2	170.9
74	100.7	476.9	21.8	1091	1076	1079	71.4	65.5	45	91.4	73.9	2217	1593	388	181.3
74	100.7	475.9	21.4	1100	1093	1111	71.6	65.5	45.1	91.5	73.9	2214	1601	372.6	173
74	100.7	476.5	22.5	1088	1081	1087	71.6	65.5	45.1	91.6	73.8	2218	1609	369.8	182.8
74	100.6	475.1	21.2	1088	1088	1078	71.6	65.6	45.2	91.6	73.9	2217	1616	367.1	171.8
74	100.6	475.3	22.4	1100	1081	1082	71.6	65.6	45.2	91.6	73.8	2214	1618	370.1	172.5
74	100.5	477	21.7	1100	1098	1081	71.6	65.8	45.3	91.6	73.8	2215	1615	372.8	173.8
74	100.4	473.8	21.6	1081	1104	1097	71.6	65.8	45.3	91.6	73.8	2220	1609	371.1	172.7
74	100.5	474.8	21.8	1089	1084	1076	71.8	65.8	45.4	91.6	73.8	2218	1603	366.8	172.5
74	100.6	474.5	21.9	1081	1087	1092	71.8	65.8	45.4	91.6	73.7	2220	1597	366.4	178.7
74	100.4	474.2	22.1	1100	1087	1096	72	65.8	45.5	91.6	73.7	2214	1593	367.3	172.2
74	100.6	475	22	1089	1080	1073	72	65.8	45.5	91.6	73.6	2218	1590	366.4	172.1
74	100.7	475.1	21.7	1083	1085	1099	72	65.8	45.5	91.6	73.6	2219	1590	382.9	171.1
74	100.7	476.2	21.4	1098	1082	1087	72	65.8	45.5	91.6	73.6	2215	1593	384.6	179.6
74	100.6	477.7	22.1	1100	1087	1094	72.1	65.8	45.5	91.6	73.6	2213	1597	367.9	172.6
74	100.6	477.1	21.6	1096	1084	1082	72.1	65.8	45.6	91.6	73.6	2216	1601	365.4	171.7
74	100.6	478	21.3	1095	1087	1084	72.1	65.8	45.6	91.6	73.5	2213	1606	365.3	171.3
74	100.6	477.2	21.6	1100	1090	1090	72.1	65.8	45.6	91.6	73.5	2215	1607	369	172.2
74	100.7	474.6	21.8	1098	1074	1074	72.1	65.9	45.7	91.6	73.4	2217	1602	368.6	173.9
74	100.6	474.3	21.4	1080	1097	1079	72.3	65.9	45.7	91.6	73.4	2219	1596	366.6	171.4
74	100.5	476.5	21.7	1100	1092	1090	72.3	65.9	45.7	91.7	73.4	2215	1591	366.8	172.6
74	100.6	477.1	21.1	1077	1091	1078	72.5	65.9	45.7	91.7	73.4	2216	1588	382.9	171.3

74	100.4	476.1	21.2	1100	1078	1084	72.3	65.9	45.8	91.9	73.3	2215	1591	383.1	179.8
74	100.7	476.7	22	1100	1087	1078	72.5	65.9	45.8	91.9	73.4	2215	1596	367	171.8
74	100.5	474.5	21.3	1087	1079	1079	72.5	65.9	45.9	91.9	73.4	2217	1602	370.9	171.5
74	100.5	475.3	21.6	1095	1091	1085	72.5	66.1	45.9	91.9	73.4	2217	1609	370.5	174.8
74	100.6	473.5	21.8	1089	1081	1083	72.5	66.1	45.9	91.8	73.4	2217	1610	368.6	171.8
74	100.6	476.2	21.8	1100	1073	1095	72.5	66.1	46	91.9	73.3	2218	1606	367.5	174.3
74	100.6	474.5	21.9	1100	1076	1077	72.5	66.1	46.1	91.9	73.4	2215	1602	366.4	171.4
74	100.6	476.9	21.2	1100	1099	1102	72.5	66.1	46.1	91.9	73.3	2214	1597	368.3	173.1
74	100.8	474.5	21.1	1082	1103	1082	72.5	66.2	46.2	91.9	73.3	2219	1593	384	172
74	100.6	475.8	21.6	1099	1085	1089	72.7	66.2	46.2	91.9	73.4	2215	1595	385.2	179.2
74	100.6	476	21.7	1100	1090	1083	72.7	66.2	46.3	91.9	73.4	2214	1600	368.1	172.8
74	100.6	475.4	22.1	1080	1090	1103	72.7	66.2	46.3	91.9	73.3	2219	1605	367.7	172.1
74	100.6	475.8	21.5	1100	1097	1074	72.7	66.3	46.4	91.9	73.4	2214	1611	368.8	173.4
74	100.5	476.2	22.1	1082	1086	1098	72.9	66.3	46.4	91.9	73.4	2220	1611	367.1	171.7
74	100.5	475.6	21.9	1099	1082	1081	72.9	66.3	46.4	91.9	73.4	2215	1606	368.6	171.3
74	100.5	475.5	22	1100	1093	1084	72.9	66.3	46.4	91.9	73.4	2214	1600	369.8	173.5
74	100.7	476.9	21.5	1087	1094	1086	73	66.3	46.5	91.9	73.4	2219	1595	369.4	172.7
74	100.5	476.6	22.3	1093	1074	1082	73	66.3	46.5	92	73.4	2218	1590	368.3	172.2
74	100.5	475.2	21.6	1098	1085	1099	73	66.3	46.5	92.1	73.4	2215	1588	367.7	171.7
74	100.6	474.9	22.3	1092	1094	1087	73.2	66.5	46.6	92.2	73.4	2218	1588	367.9	172
74	100.6	475.3	22.1	1082	1099	1108	73	66.5	46.6	92.2	73.4	2220	1588	367.1	171
74	100.8	476	22.2	1099	1080	1087	73.2	66.3	46.6	92.2	73.4	2215	1587	368.3	172
74	100.7	473.7	22	1082	1088	1084	73.2	66.5	46.6	92.2	73.4	2219	1586	369	172.4
74	100.7	474.3	20.7	1092	1082	1083	73.2	66.5	46.7	92.2	73.4	2217	1586	369	172.7
74	100.4	475.9	22.1	1082	1080	1088	73.4	66.3	46.7	92.2	73.5	2219	1586	370.3	174
74	100.6	474.6	22	1084	1098	1075	73.2	66.3	46.8	92.2	73.6	2218	1587	369.8	173
74	100.6	476.4	22.3	1100	1082	1080	73.2	66.5	46.8	92.2	73.6	2213	1588	370.3	173
74	100.6	474.5	22.7	1086	1076	1084	73.2	66.5	46.8	92.2	73.6	2218	1590	370.3	181.3
74	100.5	474.6	22.4	1082	1083	1100	73.2	66.5	46.9	92.2	73.6	2219	1591	369.6	172
74	100.6	475.8	21.8	1093	1081	1103	73.4	66.5	46.9	92.2	73.6	2216	1592	369	172.8
74	100.7	474.6	22.1	1084	1099	1082	73.4	66.5	47	92.2	73.6	2218	1592	366.4	171.6
74	100.5	473	21.9	1096	1080	1093	73.4	66.5	47	92.2	73.6	2216	1591	367.5	170.8
74	100.5	473.9	21.9	1100	1098	1098	73.4	66.5	47.1	92.2	73.6	2215	1591	371.3	173.3
74	100.5	475.9	22.5	1099	1077	1076	73.4	66.5	47.1	92.2	73.7	2216	1593	389.3	174.4
74	100.9	473.5	22.3	1093	1084	1080	73.6	66.5	47.2	92.3	73.8	2218	1599	387.1	181.2
74	100.7	472.9	21.9	1080	1087	1111	73.6	66.5	47.3	92.3	73.8	2219	1605	369.2	172.1
74	100.7	473.5	22.1	1096	1085	1084	73.6	66.5	47.3	92.2	73.8	2216	1613	367	173.3
74	100.7	473.9	21.9	1090	1092	1082	73.6	66.5	47.3	92.4	73.8	2217	1619	366.4	180.2
74	100.6	473.6	22.5	1100	1097	1084	73.6	66.5	47.3	92.5	73.8	2216	1617	367	172.5
74	100.6	474.6	21.5	1095	1095	1080	73.6	66.6	47.4	92.5	73.8	2216	1612	381.6	170.9
74	100.7	472.4	22.2	1100	1081	1083	73.6	66.6	47.4	92.5	73.9	2215	1609	383.5	178.8
74	100.6	474.3	22.2	1089	1087	1097	73.6	66.5	47.5	92.5	73.9	2218	1606	367.7	172.6
74	100.6	470.8	22	1099	1083	1084	73.8	66.6	47.5	92.5	73.9	2215	1604	366.4	171.4
74	100.6	471.3	21.8	1087	1107	1076	73.9	66.6	47.5	92.5	73.9	2218	1606	368.8	172.2
74	100.7	474.4	21.1	1100	1081	1078	73.9	66.8	47.5	92.5	73.9	2214	1607	369.8	173.4
74	100.8	471.8	21.3	1097	1081	1085	74.1	66.8	47.5	92.5	73.9	2216	1606	387.4	173.6
74	100.6	473.2	22	1091	1087	1086	74.1	66.8	47.5	92.5	73.9	2219	1608	387.8	181.7
74	100.6	473.5	21.9	1084	1096	1109	74.1	66.8	47.5	92.5	74	2219	1611	369.4	172.5
74	100.7	473.4	21.4	1096	1078	1076	74.1	66.8	47.6	92.5	74.1	2216	1613	367.7	173.5
74	100.6	472.6	22.1	1100	1075	1086	74.1	66.8	47.7	92.5	74.1	2215	1617	368.6	180.6
74	100.6	473.8	21.5	1096	1084	1088	74.1	66.8	47.7	92.5	74.1	2216	1618	371.1	173.6
74	100.7	473.5	22	1090	1075	1079	74.1	66.8	47.7	92.6	74.1	2217	1612	370.5	181.8
74	100.7	472.6	21.8	1090	1083	1082	74.1	66.8	47.8	92.5	74.1	2219	1606	369.2	172.7
74	100.7	472.5	22.1	1087	1077	1102	74.1	66.8	47.8	92.5	74.1	2217	1600	369.8	172.4
74	100.7	472.2	21.8	1096	1080	1083	74.1	66.8	47.8	92.8	74.1	2216	1595	369.4	173.6
74	100.6	473.3	22	1088	1096	1077	74.1	66.9	47.9	92.8	74.2	2218	1592	369.2	173
74	100.8	473.3	22	1100	1091	1089	74.1	66.9	48	92.8	74.2	2214	1592	370.7	173.7
74	100.8	474	21.9	1090	1087	1091	74.3	66.9	48	92.8	74.1	2217	1596	385.2	174.1
74	100.8	474.7	22	1078	1104	1079	74.3	66.9	48	92.8	74.3	2219	1601	383.5	179.2
74	100.8	472.8	22.4	1095	1078	1079	74.3	66.9	48.1	92.8	74.3	2216	1608	385.7	171.9
74	100.7	473.3	22.6	1093	1080	1094	74.3	66.9	48.2	92.8	74.3	2217	1619	385.2	181.6
74	100.6	473.2	22	1096	1081	1076	74.3	66.9	48.2	92.8	74.3	2216	1629	369	172.7

74	100.6	473	22.1	1099	1075	1081	74.3	67	48.2	92.8	74.2	2217	1634	368.6	174.1
74	100.7	471.8	21.3	1076	1096	1084	74.3	67	48.2	92.8	74.2	2219	1633	366.2	171.4
74	100.7	472.8	21.2	1100	1084	1087	74.3	67	48.3	92.8	74.3	2214	1627	367.5	171.8
74	100.7	470.6	21.5	1100	1080	1098	74.3	67	48.3	92.8	74.2	2215	1617	382.7	173
74	100.7	473.3	21.6	1078	1083	1089	74.3	67	48.3	92.8	74.3	2216	1610	382.2	178.3
74	100.6	471.5	21.9	1098	1084	1088	74.5	67	48.3	92.8	74.3	2214	1606	368.8	171.5
74	100.6	472.7	22.3	1096	1089	1079	74.5	67	48.4	92.8	74.3	2214	1606	370.7	173.4
74	100.8	471.1	21.5	1100	1095	1086	74.5	67.2	48.4	92.8	74.1	2215	1613	388.2	174
74	100.8	473.1	22.3	1096	1085	1086	74.5	67.2	48.4	92.9	74.2	2219	1620	387.6	180.9
74	100.7	473.7	21.7	1079	1099	1109	74.5	67.2	48.4	92.8	74.1	2219	1623	370.1	172.8
74	100.7	472.3	21.6	1099	1083	1092	74.5	67.2	48.4	93	74.1	2215	1626	367.7	173.3
74	100.6	471.5	21.8	1097	1084	1092	74.6	67.2	48.5	93.1	74.2	2214	1628	368.1	180.5
74	100.6	472.1	22.2	1099	1095	1079	74.6	67.2	48.5	93	74.1	2216	1622	370.9	173.8
74	100.8	471.2	21.6	1100	1095	1087	74.6	67.2	48.5	92.9	74.1	2215	1613	368.1	174
74	100.6	471.9	22.6	1100	1085	1081	74.8	67.2	48.6	93.1	74.1	2215	1606	366	171.7
74	100.7	471.4	22.3	1093	1086	1080	74.8	67.3	48.6	93.1	74.1	2217	1598	366.2	171.5
74	100.8	471.8	22.2	1084	1107	1092	74.8	67.3	48.6	93.1	74.1	2220	1591	367.5	170.6
74	100.7	473.1	22.2	1095	1082	1079	75	67.3	48.7	93.1	74.1	2216	1586	369.2	172.8
75	100.6	470.7	21.8	1098	1085	1080	74.8	67.3	48.7	93.1	74.1	2214	1585	370.1	173.5
74	100.6	471.7	21.8	1094	1082	1083	74.8	67.3	48.8	93.1	74.1	2219	1585	368.8	174
74	100.8	472.4	21.9	1086	1090	1102	74.8	67.3	48.8	93.1	74.1	2219	1585	365.8	172
74	100.8	472.6	22.1	1088	1096	1082	74.8	67.3	48.9	93.1	74.1	2218	1585	367.3	171.6
74	100.5	473.8	21.9	1100	1079	1085	75	67.3	48.9	93.1	74.1	2215	1585	366.8	173.6
74	100.7	472	22	1090	1080	1084	75	67.5	48.9	93.1	74.1	2217	1584	366.2	170.4
74	100.5	472.3	22.2	1083	1084	1105	75	67.5	49	93.1	74.1	2219	1583	368.3	172.7
74	100.6	470.7	22	1092	1078	1082	75	67.5	49	93.1	74	2216	1583	370.9	172.2
74	100.5	471.3	22.4	1095	1093	1085	75	67.5	49.1	93.1	74	2219	1583	370.1	173.8
74	100.7	471.5	21.6	1083	1104	1083	75	67.5	49.1	93.1	74.1	2218	1584	366.8	171.9
74	100.7	470.8	22.1	1087	1079	1080	75	67.5	49.1	93.1	73.9	2218	1585	366.4	172.6
74	100.8	470.5	21.8	1080	1091	1111	75	67.5	49.1	93.1	74	2220	1586	367.5	171.3
74	100.7	471.9	22.1	1086	1095	1082	74.8	67.5	49.2	93.1	73.9	2218	1586	367.9	172.1
74	100.7	471.2	22.3	1092	1097	1083	75	67.5	49.2	93.1	73.9	2217	1585	368.6	171.5
74	100.5	471.8	22.5	1094	1095	1083	75	67.6	49.2	93.1	73.9	2217	1584	368.8	173.3
74	100.5	472.4	22.7	1091	1075	1086	75.2	67.6	49.2	93.1	73.9	2217	1584	366.4	172.1
74	100.6	471.1	22.1	1084	1082	1103	75.2	67.5	49.2	93.1	73.9	2219	1584	369.8	171.1
74	100.6	473.3	22.1	1100	1089	1102	75.2	67.5	49.2	93.2	73.9	2216	1585	372.2	174.5
74	100.7	471.5	22.2	1095	1086	1095	75.4	67.5	49.3	93.3	73.9	2215	1587	371.5	172.9
74	100.7	471.7	21.4	1100	1096	1088	75.4	67.5	49.3	93.3	73.9	2215	1593	389.7	174.6
74	100.6	471.7	21.6	1100	1078	1098	75.4	67.5	49.3	93.4	73.9	2216	1602	389.3	181.4
74	100.6	471.9	21.7	1089	1088	1088	75.4	67.5	49.3	93.3	73.9	2219	1611	369.8	173.4
74	100.7	472.2	22.1	1086	1103	1099	75.4	67.5	49.4	93.4	73.9	2219	1618	366.4	172.1
74	100.8	471.9	21.7	1086	1088	1092	75.4	67.5	49.4	93.2	73.9	2219	1627	382.9	170.8
74	100.7	471.7	21.5	1081	1082	1096	75.4	67.6	49.4	93.4	73.9	2215	1631	384.4	179.5
74	100.5	471.6	22.6	1100	1086	1078	75.4	67.6	49.5	93.4	73.9	2215	1627	367.3	171.9
74	100.8	471.7	22.2	1080	1081	1081	75.5	67.6	49.5	93.4	73.9	2216	1622	366	171.2
74	100.6	472	21.8	1100	1083	1084	75.5	67.6	49.6	93.4	73.9	2214	1619	367.7	171.7
74	100.6	471.5	22.1	1100	1080	1084	75.5	67.6	49.6	93.4	73.9	2217	1611	367.7	172.2
74	100.7	470.4	22.8	1083	1082	1105	75.5	67.6	49.6	93.4	73.9	2219	1604	382.4	170.6
74	100.8	471	22.2	1098	1099	1101	75.5	67.6	49.7	93.4	73.9	2216	1604	383.5	179.4
74	100.7	472	22.6	1100	1091	1097	75.5	67.6	49.7	93.4	73.9	2214	1604	365.1	172.4
74	100.7	472.4	22.4	1085	1107	1085	75.5	67.6	49.8	93.4	73.9	2219	1603	363.2	170.9
74	100.6	472.4	23.1	1097	1082	1090	75.5	67.8	49.8	93.4	73.9	2215	1604	366.6	171.3
75	100.6	471.6	22.9	1092	1096	1082	75.5	67.8	49.8	93.5	73.9	2219	1603	368.3	172.3
74	100.6	470.8	21.5	1074	1100	1073	75.5	67.8	49.8	93.5	73.8	2219	1597	366.6	171.1
74	100.6	471.6	22.4	1099	1084	1090	75.5	67.8	49.9	93.5	73.9	2215	1591	367.9	171.8
74	100.5	470.8	22	1080	1102	1097	75.5	67.8	49.9	93.5	73.9	2217	1587	371.1	172.7
74	100.7	472.3	22.8	1100	1092	1088	75.7	67.8	49.9	93.5	73.8	2215	1584	371.8	174.2
74	100.5	471.1	22.4	1086	1082	1093	75.7	67.8	49.9	93.5	73.9	2218	1584	368.3	182.1
74	100.6	472.7	22.3	1081	1083	1091	75.7	67.8	49.9	93.5	73.9	2219	1587	366	171
74	100.7	471.3	21.3	1089	1103	1096	75.7	67.9	50	93.5	73.8	2218	1589	366.2	171.5
74	100.6	471.9	22.3	1094	1082	1081	75.9	67.9	50	93.5	73.8	2217	1588	366.8	179
74	100.4	471.4	22.8	1100	1084	1092	75.9	67.9	50	93.5	73.8	2215	1586	367.5	172

75	100.7	473.1	22.4	1086	1094	1100	75.9	67.9	50	93.6	73.9	2218	1584	367.1	171.1
74	100.7	471.3	22.6	1093	1081	1079	75.9	67.9	50	93.6	73.9	2216	1582	367.5	171.4
74	100.6	471.7	22	1100	1084	1076	75.9	67.9	50	93.5	73.9	2215	1582	369.6	172.6
74	100.6	472.2	21.6	1100	1078	1086	75.9	67.9	50	93.6	73.9	2215	1583	372.4	173.7
74	100.8	471.9	22.8	1089	1092	1090	75.7	67.9	50.1	93.8	73.9	2219	1585	370.9	182.9
74	100.6	470.5	22.8	1097	1079	1081	75.9	67.9	50.1	93.6	73.9	2216	1588	368.1	172.2
74	100.6	473.9	22.8	1100	1091	1089	75.7	67.9	50.1	93.7	73.9	2215	1590	367.7	172.3
74	100.8	469.3	21.7	1099	1080	1076	75.9	67.8	50.2	93.7	73.8	2215	1592	367.5	173.1
74	100.6	472.2	21.6	1091	1106	1089	75.9	67.9	50.2	93.8	73.9	2218	1591	368.8	172.6
74	100.6	471.7	22.5	1100	1086	1077	75.9	67.8	50.3	93.8	73.8	2217	1591	370.5	173.8
74	100.7	471.3	22.6	1082	1080	1086	75.9	67.9	50.4	93.8	73.8	2219	1592	385.4	173.8
74	100.7	471.8	22.4	1093	1106	1084	75.9	67.9	50.4	93.8	73.9	2217	1597	384.4	180
74	100.7	472	22.4	1095	1075	1086	75.9	67.8	50.4	93.8	73.9	2217	1602	368.6	172.3
74	100.7	470.8	21.8	1087	1092	1110	75.9	67.8	50.5	93.8	73.9	2218	1608	367.3	171.7
74	100.8	470.4	21.7	1100	1080	1084	75.9	67.9	50.5	93.8	73.9	2215	1613	367.7	171.5
74	100.7	471.4	22.7	1100	1081	1084	75.9	67.9	50.6	93.8	73.9	2215	1613	370.3	172.8
74	100.7	471.6	22.5	1096	1079	1088	75.9	67.9	50.6	93.8	73.9	2217	1610	384.2	179.9
74	100.7	470.7	22	1071	1097	1081	75.9	67.9	50.6	93.8	73.9	2220	1609	383.1	178.2
74	100.7	469.7	21.7	1097	1086	1085	76.1	67.9	50.6	93.8	73.9	2216	1609	369.8	172.7
74	100.8	472.1	21.5	1100	1093	1093	76.1	67.9	50.6	93.8	73.9	2214	1610	373.5	173.4
74	100.7	471.8	22.1	1088	1092	1091	76.1	67.9	50.7	93.8	73.9	2219	1615	374.5	183.5
74	100.7	470.9	22	1082	1088	1096	76.1	67.9	50.7	93.8	73.9	2220	1618	373.2	174
74	100.6	470.1	22.9	1100	1082	1084	76.1	68	50.7	93.8	73.9	2214	1617	374.8	174
74	100.6	470.8	22.1	1100	1093	1085	76.3	68	50.7	93.8	74	2216	1615	371.3	174.3
74	100.6	470.2	21.8	1100	1096	1097	76.3	68	50.7	93.8	73.9	2215	1612	368.1	172.3
74	100.7	470.9	22.4	1081	1092	1097	76.3	68	50.7	93.8	73.9	2220	1607	368.5	172.3
74	100.8	468.9	21.8	1096	1076	1087	76.3	68	50.7	93.9	74	2216	1606	383.5	171.9
74	100.8	470.8	22.4	1096	1088	1084	76.4	68	50.7	93.9	74.1	2215	1607	384.4	178.8
74	100.7	470.9	21.9	1100	1085	1083	76.4	68	50.7	93.8	74.1	2215	1608	367.7	172.8
74	100.7	470.7	22.2	1086	1105	1081	76.4	68	50.8	93.9	74	2218	1610	367.1	170.8
74	100.7	471.3	22.2	1100	1089	1093	76.4	68	50.8	94	74.1	2214	1613	369.2	179.5
74	100.7	470.2	22.7	1100	1080	1079	76.4	68	50.8	94	74.1	2218	1611	369.2	172.1
74	100.8	468.6	22.7	1089	1097	1084	76.6	68.2	50.8	94.1	74.1	2218	1605	367.9	171.8
74	100.7	469.1	21.5	1100	1080	1078	76.6	68	50.9	94.1	74	2213	1600	369.6	172.2
74	100.6	470.2	18.6	1098	1090	1090	76.6	68.2	50.9	94.1	74.1	2217	1596	370.9	174.1
74	100.8	472.7	17.7	1094	1096	1104	76.6	68.2	50.9	94.1	74.1	2218	1592	372.2	173.1
74	100.6	471.1	17.2	1100	1080	1086	76.6	68.2	51	94.1	74.1	2216	1592	373.7	174.7
74	100.7	470.4	17.6	1100	1091	1091	76.6	68.2	51.1	94.1	74.1	2217	1595	371.1	174.3
74	100.7	471.4	16.1	1096	1093	1082	76.6	68.2	51.1	94.1	74.1	2213	1597	370.1	173.5
75	100.7	470.3	17	1100	1085	1089	76.6	68.2	51.1	94.1	74.1	2213	1600	373.2	173.8
74	100.7	470.1	17.5	1100	1103	1087	76.6	68.2	51.2	94.1	74.1	2215	1602	373.5	176.2
74	100.7	470.6	16.7	1096	1086	1086	76.6	68.2	51.2	94.1	74.1	2212	1603	372.2	174
74	100.7	470.6	15.8	1097	1096	1086	76.6	68.2	51.2	94.1	74.1	2217	1604	372.6	174.4
74	100.6	471.9	15.5	1100	1080	1103	76.6	68.2	51.2	94.1	74.1	2216	1609	392.7	174.3
74	100.6	469.2	15.8	1096	1105	1110	76.6	68.2	51.3	94.1	74.1	2218	1616	393.1	184.1
74	100.6	473.2	15.9	1100	1088	1101	76.6	68.2	51.3	94.1	74.1	2214	1624	373	174.7
74	100.8	468.9	14.9	1099	1077	1097	76.6	68.2	51.3	94.1	74.1	2217	1631	372.6	182.5
74	100.8	472	14.9	1100	1094	1092	76.8	68.2	51.3	94.1	74.2	2213	1638	372.6	174.6
74	100.6	468.2	15.6	1100	1087	1091	76.8	68.2	51.3	94.1	74.1	2216	1638	374.8	174.9
74	100.6	469.7	14.8	1096	1090	1099	76.6	68.3	51.3	94.1	74.1	2218	1632	376.7	175.1
74	100.8	469.4	14.6	1100	1104	1109	76.6	68.3	51.3	94.1	74.1	2213	1627	377.3	176.4
74	100.8	471.2	14.8	1100	1084	1084	76.6	68.3	51.3	94.1	74.1	2215	1623	376.9	176.5
74	100.6	471.8	15.4	1100	1085	1087	76.8	68.2	51.3	94.1	74.2	2215	1619	376.3	175.7
74	100.6	471.3	14.9	1096	1103	1108	76.8	68.3	51.3	94.2	74.2	2217	1619	376	175.6
74	100.6	468.7	14.3	1100	1093	1086	76.8	68.3	51.3	94.1	74.3	2214	1621	377.1	176.4
74	100.7	469.1	14.6	1100	1099	1092	76.8	68.3	51.3	94.1	74.2	2213	1623	376.7	177.4
74	100.7	468.4	13.5	1095	1102	1105	76.8	68.3	51.3	94.3	74.3	2218	1623	375	175
74	100.7	469.4	14.2	1100	1103	1084	76.8	68.3	51.3	94.3	74.3	2214	1623	375.2	175.3
74	100.7	469.3	13.5	1098	1099	1108	77	68.3	51.4	94.4	74.3	2217	1625	393.1	175.3
74	100.8	470.2	13.5	1100	1103	1112	77	68.3	51.4	94.3	74.3	2212	1631	393.3	184.8
74	100.7	469.3	13.9	1098	1091	1093	77	68.3	51.5	94.4	74.2	2218	1638	392.9	175.8
74	100.6	467.9	13.4	1100	1088	1101	77	68.3	51.5	94.4	74.3	2214	1650	392.5	183.6

74	100.6	470.8	13.2	1100	1097	1080	76.8	68.3	51.5	94.4	74.3	2213	1661	374.5	175.6
74	100.7	471.2	13.3	1100	1086	1083	77	68.5	51.6	94.4	74.3	2217	1667	375.2	175.1
74	100.8	470	13.2	1100	1103	1105	76.8	68.5	51.6	94.4	74.3	2217	1666	377.1	175.9
74	100.7	469.4	12.8	1100	1089	1090	77	68.5	51.6	94.4	74.3	2213	1661	378.2	177.2
74	100.7	470.4	13.1	1100	1100	1090	77	68.5	51.7	94.4	74.3	2213	1650	380.1	177.1
74	100.7	470.8	12.5	1100	1098	1088	77	68.5	51.7	94.4	74.3	2214	1640	379	177.6
74	100.7	471.2	12.6	1100	1106	1087	77	68.5	51.8	94.4	74.3	2216	1634	378.8	176.5
74	100.7	470	12.8	1100	1095	1092	77.1	68.5	51.8	94.4	74.3	2213	1632	380.1	178.2
74	100.6	471.6	12.6	1100	1092	1101	77.1	68.5	51.8	94.4	74.3	2217	1632	377.3	177.2
74	100.8	469.3	11.7	1100	1100	1102	77.1	68.5	51.8	94.4	74.3	2216	1633	377.7	175.9
74	100.7	470.4	11.9	1100	1108	1089	77.1	68.5	51.8	94.4	74.3	2215	1633	378.4	177.5
74	100.7	470.5	11.6	1100	1086	1080	77	68.5	51.8	94.4	74.3	2214	1633	378.6	176.8
74	100.6	469.8	11.5	1100	1098	1104	77.1	68.5	51.9	94.4	74.3	2213	1632	380.9	178
74	100.6	469.7	11.2	1100	1088	1095	77.1	68.5	51.9	94.4	74.3	2216	1634	392.1	178.5
74	100.7	470.5	-14.1	1100	1100	1081	77.1	68.6	51.5	94.4	74.4	2210	1642	404.5	190.6
74	100.7	470.2	-13.7	1100	1113	1080	77.1	68.6	51.1	94.4	74.3	2208	1654	406.6	189.6
74	100.8	471.1	-14.2	1100	1077	1091	77.1	68.6	50.5	94.4	74.3	2211	1670	396.3	189.8
74	100.8	470.6	-14.4	1099	1100	1102	77.1	68.5	50	94.4	74.3	2219	1683	366.8	186.7
74	100.8	472.3	-14.9	1056	1050	1034	77.1	68.6	49.5	94.4	74.3	2223	1683	333.5	168
74	100.8	473.9	-14.6	995	986	980	77.1	68.6	49.1	94.4	74.3	2230	1661	303.6	153.9
74	100.7	478.1	-14.6	900	898	903	77.1	68.6	48.5	94.4	74.3	2241	1608	267.9	132.5
74	100.8	481.3	-14.7	900	898	910	77.1	68.5	48.1	94.4	74.3	2248	1522	220.4	115
74	100.7	480.8	-14.7	900	897	898	76.8	68.5	47.3	94.4	74.3	2254	1406	184.1	97.5
74	100.8	481.2	-15.4	900	902	905	76.6	68.3	46.9	94.4	74.3	2259	1273	173.2	81.5
74	100.8	480.5	-14.7	900	898	893	76.6	68.3	46.6	94.4	74.3	2261	1132	158.2	75.9
74	100.6	482	-14.7	900	893	895	76.4	68.3	46.3	94.4	74.3	2260	995	146.2	71.4
74	100.6	481.2	-15.1	900	896	898	76.3	68.2	45.9	94.4	74.3	2262	877	153.5	68.4
74	100.8	480.4	-15.1	900	903	899	76.1	68.2	45.6	94.4	74.3	2257	785	149	73.9
74	100.9	480.9	-15.3	900	891	898	75.9	68.2	45.3	94.4	74.3	2261	715	137.7	66
74	100.7	479.6	-14.7	900	893	902	75.5	68.2	44.7	94.4	74.1	2261	668	140	65.5
74	100.8	479.3	-14.9	900	902	897	75.5	68	44.5	94.4	74.1	2259	640	142.8	66.8
74	100.7	480.1	-15.2	900	902	895	75.4	68	44.2	94.5	74.1	2257	623	141.5	69.2
57	100.8	480.7	-15	900	895	894	75.2	68	44	94.4	74.1	2260	611	146.2	65.5
54	100.7	480.6	-14.9	900	893	901	75.2	68	43.8	94.4	74	2262	596	85.3	71.2
39	100.8	480	-14.3	900	910	901	75	68	43.5	94.6	73.9	2202	562	9.2	4.3
30	100.7	445.7	-12	900	895	896	74.8	67.9	43.3	94.4	73.9	1938	522	92.1	0
45	100.7	461.1	3.5	849	850	849	61.7	56.2	37.9	74.8	65.4	1840	501	193.7	90.5
45	100.6	461.6	3.5	851	857	846	61.7	56.2	38	74.9	65.4	1841	496	193.5	90.8
45	100.8	460.5	3.6	846	843	847	61.6	56.2	38	75	65.4	1837	524	184.5	85.6
45	100.6	461	3.5	847	840	853	61.6	56.2	38	75	65.4	1838	610	184.5	85.9
46	100.6	460	3.9	841	835	837	61.6	56.2	38	75	65.4	1833	723	184.1	85.8
45	100.8	459.9	3.7	838	838	836	61.6	56.2	38	75	65.5	1831	819	195	85.9
46	100.8	458.9	3.6	841	831	834	61.4	56.2	38	75.1	65.5	1832	901	196	91.8
45	100.5	457.2	3.8	839	825	848	61.4	56.2	38	75.2	65.5	1833	960	186.8	91.8
46	100.8	459.4	3.7	836	840	826	61.3	56.2	38	75.1	65.6	1829	985	196.2	86.4
46	100.8	458.9	3.4	836	845	832	61.3	56.2	38	75.2	65.6	1829	992	196.9	91.7
46	100.7	458.7	4	836	823	829	61.3	56.2	38	75.2	65.6	1829	997	186.2	87.2
46	100.8	458.5	3.8	838	842	828	61.1	56.2	38	75.2	65.8	1830	1001	195	87
46	100.7	458.6	4	836	830	837	61.1	56.2	38	75.2	65.8	1829	1006	194.8	91.1
46	100.6	459.7	3.6	838	830	835	61.1	56.2	38	75.2	65.8	1830	1006	183.4	91.5
45	100.7	459.1	3.8	839	832	836	61.3	56.1	38	75.4	65.9	1832	1003	195	86
45	100.8	459.8	3.7	836	829	833	61.1	56.1	38	75.4	65.9	1827	1003	196.2	91.4
46	100.7	458.9	3.9	834	830	836	61.3	56.1	38	75.4	65.9	1827	1003	188.3	86.8
46	100.7	459	4.1	836	835	834	61.1	56.1	38	75.4	65.9	1829	1002	203.1	89
47	100.8	461.3	4.2	856	850	851	61.1	56.1	38	75.4	66	1845	1010	214.4	96.8
47	100.8	460.4	4.4	864	862	868	61.3	56.1	38	75.5	66.1	1852	1023	215.5	95.6
50	100.9	463.6	4.9	927	903	937	61.3	56.1	38	75.5	66.1	1880	1035	210.8	103.9
50	100.6	474.3	5.9	900	899	893	61.4	56.1	38	75.5	66.1	1915	1050	205.4	100.8
50	100.7	470.7	5.8	900	893	906	61.4	56.1	38	75.5	66.2	1919	1061	205.4	101.7
50	100.8	473	5.9	900	898	892	61.6	56.1	38	75.5	66.2	1921	1066	216.3	101.1
50	100.5	473.2	5.6	900	898	890	61.6	56.1	38	75.6	66.2	1926	1072	254.4	102.4
52	100.7	478.4	6.9	983	985	975	61.7	56.1	38	75.7	66.3	1966	1086	265.9	130.1

52	100.8	486.7	8.4	923	938	935	61.7	56.2	38	75.7	66.3	1998	1105	249.5	117.1
52	100.8	487.1	8.3	917	920	913	61.8	56.2	38	75.7	66.3	2002	1131	246.3	115
52	100.7	491.6	7.7	911	904	901	62	56.2	38	75.7	66.5	2010	1162	244.8	114.2
52	100.7	491.2	8.3	907	911	905	62	56.2	38	75.7	66.5	2014	1183	243.7	114
52	100.5	489.9	8.3	924	906	920	62.1	56.2	38	75.9	66.5	2013	1190	243.9	119.4
52	100.7	486.2	8.5	906	918	905	62.1	56.3	38	75.9	66.5	2012	1191	255.7	114
52	100.7	490.6	7.9	912	921	905	62.3	56.3	38	75.9	66.6	2017	1187	254.6	119.7
52	100.7	490.6	8	913	903	926	62.3	56.4	38	75.9	66.6	2016	1179	243.7	113.7
52	100.7	488.3	7.5	920	911	901	62.4	56.4	38	76	66.6	2013	1175	244.1	114.4
52	100.7	486.9	8.3	916	902	921	62.4	56.4	38	76.1	66.8	2017	1177	243.9	119.3
52	100.5	488.5	7.5	911	907	902	62.5	56.4	38	76.1	66.8	2015	1175	243.5	114.3
52	100.6	488.9	7.9	909	899	905	62.5	56.4	38	76.1	66.8	2019	1170	242.6	113.4
52	100.7	489.6	7.7	918	924	910	62.5	56.4	38	76.3	66.9	2017	1165	242.4	113.4
52	100.7	488.4	7.7	908	897	919	62.5	56.4	38	76.3	66.9	2019	1160	242.4	119
52	100.5	488.1	8.3	907	904	898	62.7	56.4	38	76.3	66.9	2019	1154	242.6	113.7
52	100.8	488.1	7.4	915	920	901	62.7	56.4	38	76.3	66.9	2022	1150	242.2	113.8
52	100.8	486.5	8.2	906	906	900	62.7	56.4	38	76.3	66.9	2019	1149	242.6	113.7
52	100.8	486.9	7.8	924	905	902	62.7	56.4	38	76.4	67	2021	1148	242	114.3
52	100.8	487.3	8.1	908	915	903	62.8	56.4	38	76.4	67	2018	1147	241.8	113
52	100.6	487.4	8.2	906	920	904	62.8	56.4	38	76.4	67	2021	1146	242.9	113.6
52	100.6	489.5	8.3	908	903	920	62.8	56.4	38	76.4	67	2020	1146	243.3	119.2
52	100.7	487.4	7.6	912	907	900	62.8	56.4	38	76.5	67.2	2022	1146	243.5	114.2
52	100.8	486.1	8.4	915	911	900	62.8	56.4	38.1	76.5	67.2	2018	1146	243.1	113.8
52	100.7	486.2	7.6	911	909	901	62.8	56.4	38.1	76.5	67.2	2022	1149	254	113.3
52	100.5	487.6	8.2	906	919	909	62.8	56.4	38.1	76.6	67.3	2019	1154	253.8	118.3
52	100.6	486.3	8.3	909	922	902	62.8	56.4	38.1	76.6	67.3	2022	1159	242.6	113.6
52	100.7	486.7	8.6	903	897	915	62.8	56.4	38.1	76.7	67.3	2021	1163	242.9	119.1
52	100.7	485.7	8	903	899	916	62.8	56.4	38.1	76.7	67.4	2021	1167	242	119.1
52	100.7	484.7	8.3	902	905	927	62.8	56.4	38.1	76.8	67.5	2021	1166	242.9	113.4
52	100.6	485.4	8.2	914	904	902	62.8	56.4	38.1	76.8	67.5	2023	1162	244.3	114.1
52	100.6	485.8	8.4	902	911	901	63	56.4	38.1	76.8	67.5	2021	1158	243.9	113.4
52	100.8	484.8	7.9	924	901	897	63	56.4	38.1	76.8	67.6	2023	1154	242.8	113.5
52	100.8	484.4	8.5	907	905	905	63	56.4	38.1	76.9	67.6	2020	1150	242.4	113.8
52	100.7	485.3	8.2	911	908	902	63	56.3	38.1	77	67.6	2023	1151	254.8	113.7
52	100.6	484.8	7.9	911	922	902	63	56.4	38.1	77	67.8	2022	1158	266.6	118.9
52	100.7	484.2	8.3	904	923	904	63	56.3	38.1	77.1	67.8	2020	1166	254.6	119.5
52	100.6	485.3	8.1	908	902	917	63	56.3	38.2	77.1	67.8	2022	1175	242.8	113.7
52	100.6	485.2	8.6	920	901	924	63.1	56.3	38.2	77.1	67.8	2021	1183	242.9	113.7
52	100.6	483.7	8.2	916	901	928	63.1	56.3	38.2	77.1	67.9	2020	1188	242.6	113.4
52	100.7	484.1	8.4	908	905	898	63.1	56.4	38.2	77.1	67.8	2022	1183	242.6	113
52	100.6	483.1	8.7	903	897	899	63.1	56.4	38.2	77.2	67.9	2025	1174	243.1	113.4
52	100.7	482.4	8.6	905	902	898	63.1	56.3	38.2	77.3	67.9	2022	1165	242.4	113.4
52	100.5	485	8.3	901	911	905	63.1	56.3	38.2	77.3	67.9	2023	1155	242.2	113.5
52	100.7	482.3	8.3	917	909	906	63.1	56.3	38.2	77.3	68	2019	1148	242.4	113.5
52	100.7	485	8.4	900	903	906	63.1	56.3	38.2	77.3	68	2023	1146	242.4	113.5
52	100.6	482.7	8.7	907	905	906	63.1	56.3	38.2	77.3	68	2019	1145	242.8	113.3
52	100.7	483.5	8.8	912	906	898	63.1	56.3	38.3	77.3	68.2	2024	1145	243.3	113.9
52	100.7	481.5	8.7	915	905	898	63.1	56.3	38.3	77.5	68.2	2021	1146	242.4	113.6
52	100.5	482.6	8	904	901	896	63.1	56.3	38.3	77.5	68.2	2022	1146	241.1	113.3
52	100.5	483.1	8.5	911	908	909	63.1	56.3	38.3	77.6	68.3	2023	1146	240.9	112.8
52	100.7	482.4	8.8	907	906	904	63.2	56.3	38.3	77.6	68.3	2020	1145	241.8	113.2
52	100.6	481.4	8.1	909	911	907	63.2	56.3	38.3	77.7	68.5	2023	1146	253.5	113.2
52	100.8	481.5	8.8	917	920	908	63.2	56.4	38.3	77.7	68.5	2020	1149	252.9	118.4
52	100.6	484.6	8.7	912	924	898	63.2	56.4	38.4	77.7	68.5	2022	1153	242.2	118.5
52	100.4	484.6	8.3	921	919	903	63.2	56.4	38.4	77.9	68.5	2023	1157	242	113.5
52	100.8	482.3	7.7	923	916	906	63.2	56.3	38.4	77.9	68.5	2022	1161	241.8	118.5
52	100.6	481.5	8	911	926	899	63.2	56.3	38.4	77.9	68.5	2024	1161	241.8	119.3
52	100.6	484.2	8.6	911	897	922	63.2	56.3	38.4	77.9	68.6	2020	1157	241.6	118.6
52	100.7	478.5	8.5	917	900	927	63.4	56.4	38.4	77.9	68.6	2021	1153	241.6	113.4
52	100.6	482.3	8.9	900	903	924	63.4	56.4	38.5	78	68.6	2022	1148	241.1	112.6
52	100.7	479.9	8.3	914	902	900	63.4	56.4	38.5	78	68.7	2024	1144	241.8	112.7
52	100.7	480.3	8.3	917	906	898	63.4	56.4	38.5	78	68.7	2023	1141	242.4	113.8

52	100.6	479.3	7.9	914	905	899	63.5	56.6	38.5	78	68.7	2017	1142	243.7	113.4
52	100.7	477.8	8.9	921	902	927	63.5	56.6	38.5	78.2	68.9	2004	1143	244.6	114.4
52	100.8	477.2	8.6	925	911	910	63.5	56.6	38.5	78.2	68.9	1995	1146	246.7	114.6
52	100.5	472.3	9	920	918	923	63.7	56.6	38.5	78.2	69	1977	1153	251.8	116.1
52	100.5	471.1	9	936	926	928	63.7	56.6	38.5	78.2	69	1974	1163	256.9	118.3
52	100.7	469.6	9.4	944	943	935	63.7	56.6	38.6	78.3	69	1957	1177	261	120.9
52	100.8	467.8	10.3	967	940	944	63.7	56.6	38.6	78.4	69	1941	1196	264.2	122.3
52	100.7	465.4	10	976	973	955	63.7	56.6	38.6	78.4	69	1932	1219	269.6	125.1
52	100.7	460.5	10.8	998	970	976	63.8	56.7	38.6	78.4	69	1906	1243	262.3	128.3
52	100.5	456.5	11.9	976	982	991	63.8	56.7	38.6	78.4	69.1	1892	1266	257.8	117.9
52	100.6	452.8	12.9	991	1002	992	63.8	56.7	38.7	78.4	69.2	1876	1286	264.9	127.8
52	100.6	453.2	13	976	995	975	63.8	56.7	38.7	78.5	69.2	1872	1303	263.2	122.5
52	100.6	453.2	12	1018	1003	982	63.9	56.8	38.7	78.6	69.2	1865	1314	259.9	122.4
52	100.5	452.8	12.4	1023	983	979	63.9	56.8	38.7	78.6	69.3	1869	1322	261	121.6
52	100.7	454.3	12.8	977	975	986	63.9	56.9	38.7	78.6	69.3	1873	1329	261.5	128
52	100.7	452.8	13.9	979	978	980	63.9	56.9	38.7	78.6	69.3	1872	1334	273.6	127.8
52	100.6	451.9	13.1	969	1000	1002	64.1	56.9	38.8	78.8	69.3	1870	1343	288.4	127.3
52	100.6	452.2	12.6	1001	981	976	64.1	56.9	38.8	78.8	69.4	1866	1364	317.5	135.6
54	100.4	451.8	13.5	1100	1088	1127	64.1	57	38.8	78.8	69.4	1884	1403	353.2	168.8
55	100.6	467.5	19.2	1100	1115	1117	64.2	57	38.8	78.8	69.4	1968	1455	355.1	166.9
55	100.6	476	20.8	1095	1088	1084	64.2	57	38.8	78.8	69.4	2002	1516	345	161.5
55	100.7	476	20.4	1083	1079	1073	64.4	57.1	38.9	78.9	69.4	2009	1577	338.2	159.3
55	100.8	474.2	20.2	1100	1074	1098	64.5	57.1	38.9	78.9	69.6	2005	1625	336.7	157.8
55	100.6	475.1	19.2	1087	1076	1104	64.5	57.1	38.9	78.9	69.6	2010	1648	334.6	157.5
55	100.8	476.7	19.3	1086	1065	1074	64.6	57.2	38.9	78.9	69.6	2011	1649	333.1	155.4
55	100.8	475	19.5	1087	1081	1074	64.6	57.2	38.9	78.9	69.7	2005	1637	332.9	156.3
55	100.6	475.3	20.1	1081	1073	1080	64.6	57.2	38.9	79.1	69.7	2007	1619	332.4	155.7
55	100.6	474.8	20.1	1086	1074	1078	64.8	57.4	38.9	79.1	69.8	2007	1602	334.8	157.2
55	100.7	471.7	20.3	1100	1109	1095	64.8	57.4	39	79.1	69.9	2006	1592	333.5	156.9
55	100.7	471.5	19.8	1081	1069	1076	64.9	57.4	39	79.1	69.9	2002	1588	332.2	155.3
55	100.5	473.8	19.7	1080	1063	1075	65.1	57.5	39	79.3	70	2002	1587	333.3	155.9
55	100.5	473.1	20.2	1071	1068	1061	65.1	57.5	39	79.3	70	2001	1587	333.1	155.5
55	100.6	472.3	19.1	1100	1076	1072	65.1	57.6	39	79.3	70	1998	1588	335.4	155.7
55	100.5	472.3	19.2	1076	1071	1080	65.1	57.6	39	79.3	70	2004	1590	338.2	158.1
55	100.6	470.1	20.9	1100	1078	1076	65.2	57.6	39	79.4	70.2	1981	1595	343.1	157.9
56	100.5	467.6	21.6	1100	1084	1078	65.2	57.6	39.1	79.5	70.2	1982	1604	357.9	166
56	100.6	469.2	23.1	1100	1093	1093	65.2	57.7	39.1	79.5	70.2	1998	1621	367.7	172.3
57	100.7	471.5	23.7	1100	1101	1080	65.4	57.7	39.1	79.5	70.2	2009	1643	373	172.4
57	100.6	472.8	24.8	1100	1088	1124	65.4	57.8	39.2	79.5	70.3	2017	1672	380.5	176.2
57	100.7	472.6	25.6	1100	1086	1126	65.5	57.8	39.2	79.6	70.4	2028	1704	377.5	179.3
57	100.6	476.8	26.5	1100	1089	1093	65.5	57.8	39.3	79.6	70.5	2035	1731	370.9	175.5
57	100.5	476.2	25.4	1100	1119	1083	65.6	57.8	39.3	79.6	70.5	2038	1750	371.1	174.3
57	100.7	475.4	26	1100	1092	1122	65.6	57.8	39.3	79.6	70.5	2039	1762	374.5	174.5
57	100.6	474.5	25.8	1100	1090	1090	65.6	57.9	39.3	79.7	70.5	2035	1764	375.8	176.7
57	100.6	473.6	27.2	1100	1088	1083	65.8	57.9	39.4	79.8	70.6	2020	1762	377.5	176.8
57	100.8	470.7	26.8	1100	1110	1102	65.8	57.9	39.4	79.8	70.7	2005	1759	362.8	180
57	100.8	466.4	27.7	1100	1111	1119	65.8	58	39.5	79.8	70.9	1996	1757	368.5	171.1
57	100.8	466.4	28.5	1100	1087	1085	65.9	58.2	39.5	79.9	70.9	2002	1761	392.7	184
57	100.6	466.4	28.7	1100	1090	1088	65.9	58.2	39.5	79.9	70.9	2005	1771	394.8	183.5
57	100.6	468.3	29	1100	1079	1083	66.1	58.2	39.6	80	71.1	1992	1785	393.3	184.8
57	100.7	463	29.8	1100	1109	1085	66.2	58.2	39.6	80	71.1	1975	1808	399.5	181.9
57	100.6	456.6	30	1100	1116	1095	66.2	58.3	39.7	80	71.1	1959	1837	400.2	187
57	100.5	459.6	29.6	1100	1087	1117	66.3	58.3	39.7	80.1	71.2	1953	1863	393.1	185.7
57	100.7	458.9	30.2	1100	1083	1096	66.3	58.4	39.8	80.2	71.3	1955	1887	393.1	185.1
57	100.7	457	31	1100	1087	1116	66.5	58.4	39.8	80.2	71.3	1946	1909	398.3	185.9
57	100.6	454.4	31.4	1100	1092	1078	66.5	58.4	39.9	80.3	71.3	1929	1925	404.3	180.4
57	100.8	451.3	31.4	1100	1116	1082	66.6	58.5	39.9	80.3	71.3	1912	1935	391.6	191.3
57	100.7	449.8	31.7	1100	1086	1080	66.8	58.4	40	80.4	71.4	1913	1940	373	176.3
57	100.8	449.4	31.1	1100	1088	1085	66.8	58.5	40	80.5	71.4	1894	1945	392.5	181.4
57	100.7	447.3	32.1	1100	1110	1087	66.9	58.4	40.1	80.5	71.6	1894	1949	392.5	184.2
57	100.7	446.8	32.5	1100	1087	1092	66.9	58.5	40.1	80.5	71.6	1900	1949	391.4	172.5
57	100.5	446.6	32	1100	1094	1087	66.9	58.5	40.2	80.7	71.7	1886	1954	415.4	195.2

57	100.5	445.3	32.3	1100	1080	1084	66.9	58.5	40.2	80.7	71.8	1882	1969	399.1	195.4
57	100.7	440.4	32.2	1091	1099	1079	66.9	58.6	40.3	80.7	71.8	1866	1980	382.2	188.6
57	100.6	440.2	32.7	1078	1066	1088	66.9	58.6	40.3	80.7	71.8	1846	1990	381.2	179
57	100.5	435.6	32.4	1067	1054	1078	67	58.6	40.4	80.8	72	1827	2002	379.2	177.9
57	100.6	434.8	31	1063	1052	1053	67	58.6	40.5	80.9	72	1820	2003	377.1	177.4
57	100.7	433.8	30.8	1058	1057	1031	67.2	58.6	40.5	80.9	72.1	1814	2000	398.5	175.9
57	100.8	433.1	32.2	1056	1049	1055	67.2	58.7	40.5	80.9	72.1	1810	2004	401.1	185.4
57	100.8	432.7	31.7	1058	1043	1055	67.2	58.7	40.6	81	72.1	1812	2015	400.8	177.5
57	100.7	433.1	31.4	1058	1046	1056	67.3	58.7	40.6	81.2	72.3	1812	2030	401	186.9
57	100.6	431.3	31.6	1050	1055	1064	67.3	58.7	40.7	81.2	72.3	1789	2053	385.6	178.7
57	100.5	427.3	31.4	1050	1039	1039	67.3	58.7	40.7	81.2	72.3	1771	2074	389.7	181.7
57	100.6	422.8	32.9	1050	1035	1032	67.3	58.7	40.8	81.2	72.5	1749	2089	390.6	183
57	100.4	421.9	32.9	1050	1043	1031	67.5	58.9	40.8	81.3	72.5	1744	2102	401.7	171.9
60	100.7	421.6	34.7	1050	1038	1044	67.5	58.9	40.9	81.4	72.7	1759	2116	419.4	194.4
61	100.7	425.7	35.9	1050	1058	1060	67.5	58.9	40.9	81.4	72.7	1769	2139	439.9	200.7
62	100.7	429.6	39.3	1050	1063	1059	67.5	58.9	41	81.4	72.8	1800	2174	450.2	212.5
62	100.7	433.4	43	1065	1082	1045	67.5	58.9	41	81.4	72.9	1825	2217	448.7	208.8
62	100.7	434	43.6	1072	1048	1046	67.5	59	41.2	81.6	73	1836	2261	444.9	208.3
62	100.5	435.2	43.5	1068	1059	1083	67.5	59	41.2	81.6	73	1828	2299	439.7	204.6
62	100.5	435.6	43.1	1073	1080	1054	67.3	59.1	41.3	81.6	73.1	1837	2327	471.8	207.5
62	100.7	435.1	44	1074	1063	1066	67.2	59.1	41.3	81.7	73.2	1839	2352	495.5	223.2
63	100.5	435.2	44.8	1088	1089	1109	67.3	59.1	41.4	81.8	73.2	1861	2375	494.9	230.8
65	100.4	444.3	50.6	1100	1086	1086	67.2	59.1	41.4	81.9	73.3	1898	2408	527.4	234.5
65	100.5	452.4	54.6	1100	1071	1083	67	59.1	41.6	81.9	73.4	1951	2455	547.5	246.2
65	100.7	454.1	55.2	1100	1122	1095	67	59.2	41.7	81.9	73.5	1964	2507	544.5	254.8
65	100.5	453.4	57.3	1100	1082	1088	67.2	59.2	41.8	82	73.6	1966	2553	541.5	253.8
65	100.6	455.3	57.5	1100	1087	1128	67.2	59.3	41.9	82.1	73.6	1971	2596	544.9	252.8
65	100.6	455.3	57.8	1100	1110	1116	67.3	59.3	42	82.1	73.6	1977	2627	553.2	259.4
65	100.6	455	57.6	1100	1124	1112	67.2	59.3	42.1	82.1	73.6	1974	2641	551.1	243.5
65	100.5	454.8	58	1100	1122	1079	67.3	59.4	42.3	82.1	73.8	1981	2647	547.9	256.4
65	100.8	453.3	56.8	1100	1112	1074	67.5	59.5	42.5	82.3	73.9	1974	2646	518.4	256.1
65	100.7	452.8	57.3	1100	1077	1077	67.6	59.7	42.5	82.3	73.9	1976	2631	491.9	240.9
65	100.5	453.8	56.6	1100	1077	1076	67.8	59.7	42.7	82.3	73.9	1973	2602	487.8	241.6
65	100.6	452.5	57.9	1100	1078	1083	67.8	59.8	42.8	82.6	74.1	1975	2566	512.2	239.2
65	100.5	453.9	56.2	1100	1070	1080	67.9	59.8	42.9	82.6	74.1	1971	2530	541.7	252.3
65	100.7	457.5	57.5	1100	1092	1129	68	59.9	43.1	82.6	74.3	2023	2500	533.6	253.9
65	100.7	463.3	57.7	1100	1105	1097	68.2	59.9	43.2	82.8	74.3	2042	2483	526.1	245.2
65	100.5	463.6	56.7	1100	1082	1127	68.2	59.9	43.3	82.8	74.3	2045	2482	526.1	245.9
65	100.6	464.3	56.7	1100	1085	1129	68.3	60	43.4	82.8	74.3	2051	2487	525.2	245.8
65	100.5	464.1	57.6	1100	1082	1113	68.3	60	43.5	82.8	74.3	2052	2491	547.5	244.9
65	100.5	469.3	56.9	1100	1080	1108	68.5	60.1	43.6	82.8	74.3	2080	2496	560.5	266.6
65	100.6	479.2	57.1	1100	1102	1087	68.6	60.1	43.9	83	74.3	2143	2498	553.2	258.7
65	100.8	482.8	58.3	1100	1079	1113	68.6	60.4	44	83	74.3	2158	2495	548.1	257.6
65	100.6	484.4	57.7	1100	1118	1086	68.7	60.6	44.1	83.1	74.1	2170	2490	538.7	252.9
65	100.4	487.4	57.9	1100	1084	1082	68.9	60.6	44.2	83.3	74.1	2186	2479	532.1	249.9
65	100.5	486.3	56	1100	1087	1079	69	60.6	44.4	83.3	74	2188	2455	525.9	246.3
65	100.5	487.1	55.1	1100	1096	1081	69	60.7	44.5	83.3	73.9	2185	2424	524.6	245.2
65	100.6	487.7	54.9	1100	1112	1082	69.2	60.9	44.8	83.5	73.8	2187	2388	522.7	245.1
65	100.5	486.9	55	1100	1110	1083	69.3	60.9	44.9	83.5	73.6	2185	2352	522.3	243.9
65	100.4	486.4	55.5	1100	1101	1080	69.4	61	45	83.5	73.6	2188	2321	511.8	244.4
65	100.6	488.6	53.2	1100	1107	1101	69.4	61	45.2	83.6	73.5	2189	2296	495.7	232.9
65	100.5	488.1	49.8	1100	1108	1087	69.6	61.1	45.3	83.7	73.4	2195	2271	487.8	228
65	100.5	487.3	49.4	1100	1082	1088	69.7	61.1	45.4	83.7	73.4	2198	2244	480.8	217
65	100.5	488.3	47.8	1100	1110	1074	69.7	61.1	45.5	83.7	73.2	2192	2212	477.2	224.3
65	100.4	487.8	45.8	1100	1099	1104	70	61.1	45.8	83.9	73.2	2194	2178	481.2	222.9
65	100.5	488.8	46.3	1100	1081	1094	70	61.3	45.9	84	73	2194	2148	484.4	225.6
65	100.4	485.8	46.2	1100	1080	1092	70.2	61.4	46.1	84	73	2197	2122	477.8	225.4
65	100.5	486.8	45.8	1100	1085	1111	70.2	61.4	46.2	84	73	2193	2105	479.1	221.2
65	100.7	488.1	45.4	1100	1111	1087	70.2	61.4	46.3	84.2	73	2197	2096	483.6	225.5
65	100.5	487.2	45.8	1100	1085	1117	70.4	61.6	46.4	84.2	73	2199	2092	478.6	224
65	100.5	487.1	44.9	1100	1092	1090	70.4	61.7	46.6	84.2	73	2195	2088	473.1	223.1
65	100.4	486.5	45.3	1100	1098	1089	70.7	61.7	46.8	84.2	73	2197	2086	476.1	220.9

65	100.4	486.1	45.3	1100	1095	1085	70.7	61.8	46.9	84.4	73.1	2196	2084	481	225.4
65	100.6	485.4	45.6	1100	1090	1095	70.7	61.8	47	84.4	73.2	2193	2081	478.9	224.2
65	100.4	485.8	46.8	1100	1099	1110	70.7	61.8	47.2	84.4	73.2	2199	2078	473.3	222.4
65	100.4	485	46.4	1100	1097	1089	70.9	62	47.3	84.5	73.2	2196	2076	473.3	219
65	100.6	484.6	44.9	1100	1081	1111	70.9	61.8	47.5	84.5	73.2	2200	2075	481.4	221.4
65	100.5	485.1	47.1	1100	1087	1088	71.1	62	47.7	84.7	73.4	2193	2075	481.6	225
65	100.5	484.8	45.8	1100	1084	1085	71.3	62	47.8	84.7	73.4	2194	2075	478.6	221.4
65	100.5	484.6	45.2	1100	1083	1112	71.3	62	47.9	84.8	73.4	2192	2076	475.6	225.2
65	100.6	484.8	45.5	1100	1109	1091	71.4	62	48	84.9	73.5	2199	2078	474.4	220.6
65	100.7	484	46	1100	1085	1091	71.4	62.1	48.1	84.9	73.6	2199	2078	475.4	223.1
65	100.4	483.4	45.5	1100	1083	1085	71.4	62.1	48.2	84.9	73.6	2194	2076	473.7	221.9
65	100.5	482.1	45.3	1100	1084	1085	71.6	62.1	48.3	84.9	73.6	2195	2070	456.4	220.9
65	100.5	484.5	43.3	1100	1097	1111	71.6	62.1	48.5	85.1	73.8	2214	2053	412.8	214.7
65	100.6	483.6	34.9	1100	1090	1082	71.6	62.1	48.6	85.1	73.8	2215	2018	380.7	179.7
65	100.5	486.8	29.7	1098	1083	1087	71.6	62.3	48.6	85.1	73.9	2217	1966	370.3	174.7
65	100.7	486.8	25.9	1071	1084	1089	71.6	62.1	48.7	85.3	73.9	2222	1892	348.4	168.8
65	100.6	487.3	23.2	1021	1007	986	71.6	62.3	48.7	85.4	74	2227	1794	296.3	153.5
65	100.8	487.7	16.8	900	892	952	71.6	62.3	48.8	85.4	74.1	2243	1679	256.8	129.2
65	100.5	486.8	13.8	900	889	896	71.6	62.4	48.9	85.4	74.1	2247	1557	243.5	112
65	100.8	488.6	11.3	900	892	893	71.6	62.4	48.9	85.5	74.1	2247	1427	214.8	104.5
65	100.8	488.3	9.4	900	902	900	71.6	62.4	48.9	85.6	74.2	2251	1291	186.8	90.3
65	100.6	489.3	8.6	900	900	900	71.4	62.4	49	85.6	74.3	2255	1164	180.6	83.2
65	100.7	488.4	8.1	900	898	899	71.4	62.4	49	85.6	74.3	2259	1049	167.4	81.3
65	100.5	488.4	5.9	900	897	901	71.3	62.4	49.1	85.6	74.3	2260	942	154.1	73.3
65	100.6	487.4	5	900	902	898	71.1	62.4	49.1	85.6	74.3	2248	853	156.9	69
65	100.5	488.7	5.2	900	901	897	70.9	62.3	49.2	85.8	74.3	2245	787	152.9	72.1
65	100.5	488.7	5	900	898	903	70.9	62.3	49.2	85.8	74.3	2247	735	147.7	69.8
65	100.5	486.3	5.2	900	900	898	70.9	62.3	49.2	85.8	74.3	2253	695	145.4	69.3
65	100.5	486.2	5.2	900	898	903	70.7	62.3	49.3	85.8	74.3	2243	668	143.2	68.4
65	100.6	487.8	5.2	900	897	901	70.9	62.3	49.3	85.8	74.3	2237	649	140.9	67.9
65	100.7	487.1	4.6	900	899	895	70.7	62.3	49.3	85.9	74.2	2245	634	140.7	67.4
65	100.6	485.3	4.5	900	892	901	70.7	62.1	49.4	85.9	74.1	2239	623	143.4	66.2
65	100.8	485.3	4.5	900	897	900	70.7	62.1	49.4	86	74.1	2237	616	144.3	72.2
65	100.8	485.1	4	900	896	907	70.7	62.1	49.4	86	74.1	2238	611	143	70.4
65	100.6	487.1	4.1	900	897	899	70.5	62.1	49.4	86	74.1	2248	609	143.9	67
65	100.5	486.8	4.6	900	902	895	70.5	62.1	49.4	86	74.1	2243	609	143.7	67.5
65	100.7	485.8	4.3	900	901	901	70.5	62.1	49.4	86	73.9	2251	610	142.6	64.4
65	100.8	485.6	4.2	900	902	908	70.5	62	49.5	86	73.9	2251	610	141.5	67.7
65	100.6	487.5	4.3	900	897	896	70.5	62	49.5	86.1	73.9	2239	611	147.7	65.4
65	100.7	488.5	4.2	900	903	900	70.5	62	49.5	86.1	73.9	2244	613	157.6	72.2
65	100.7	487.1	4.1	900	895	898	70.4	62	49.5	86.3	73.9	2241	617	150.1	75
65	100.6	485.5	3.7	900	894	894	70.4	62	49.5	86.3	73.8	2253	620	142.1	67.5
65	100.6	487.9	4.5	900	897	899	70.2	62	49.5	86.3	73.8	2247	625	145.8	68
65	100.6	487.2	4.4	900	899	895	70.2	62	49.5	86.3	73.6	2242	629	146.9	68.1
65	100.7	486.3	4.3	900	901	894	70.2	62	49.6	86.3	73.6	2250	628	144.1	67.6
65	100.7	486.2	4.1	900	897	897	70.2	62	49.6	86.3	73.6	2245	626	147.1	68.1
65	100.5	486.4	4.2	900	890	894	70.2	62	49.6	86.3	73.6	2251	625	153.1	72.6
65	100.7	486.3	3.9	900	901	904	70.2	62	49.6	86.3	73.5	2243	625	147.9	72.1
65	100.6	486.3	4.1	900	895	895	70	62	49.7	86.5	73.4	2243	623	142.8	70.3
65	100.4	486.6	4.2	900	894	893	70	62	49.7	86.5	73.4	2249	624	142.2	67.8
65	100.6	485.8	4.2	900	902	900	70	62	49.7	86.5	73.4	2241	624	143	68.2
65	100.5	485.4	4.1	900	893	895	70	62	49.7	86.5	73.4	2252	622	144.1	66.7
65	100.6	487.7	4.5	900	898	892	70	62	49.7	86.5	73.4	2237	618	144.5	66.3
65	100.7	486.3	3.9	900	894	897	70	62	49.7	86.5	73.3	2242	616	151.8	68.4
65	100.7	487.5	4.6	900	890	901	69.9	62	49.7	86.6	73.2	2251	620	170.8	69
65	100.7	486.1	4.5	900	894	895	70	62	49.7	86.8	73.2	2244	631	181.9	87.7
65	100.6	486.7	5.1	900	899	900	69.9	62	49.7	86.8	73.3	2241	652	194.1	84.8
65	100.8	484.5	6.8	900	905	899	69.9	62	49.7	86.8	73.2	2246	686	214.2	98
65	100.8	486.6	7.7	900	905	910	69.9	62	49.7	86.8	73.2	2244	735	252.7	103.9
65	100.5	485.6	11.3	959	947	907	69.7	62	49.7	86.8	73.2	2229	801	296.5	133.4
65	100.7	483.4	13.5	985	977	986	69.7	62	49.7	86.8	73.2	2228	883	313.6	143.4
65	100.6	485.1	15	1001	1003	1021	69.7	62	49.6	86.8	73.2	2228	979	323.9	150.4

65	100.7	482.4	16.2	1016	1039	1032	69.9	62.1	49.6	86.8	73.2	2226	1083	330.9	153.3
65	100.5	482.4	17.9	1048	1032	1046	69.9	62.1	49.6	86.9	73.2	2220	1186	338.8	157
65	100.6	482.6	18.7	1046	1034	1039	69.9	62.1	49.6	86.9	73.2	2222	1275	343.7	159.8
65	100.6	481.4	19.4	1062	1051	1052	69.9	62.1	49.6	87	73.2	2219	1350	360	161.5
65	100.6	482.8	20.7	1100	1088	1069	70	62.3	49.6	87	73.2	2211	1413	386.9	170.5
65	100.6	483.1	23.8	1100	1099	1086	70.2	62.3	49.6	87	73.2	2211	1468	399.8	186.3
65	100.7	480.4	24.4	1100	1088	1092	70.2	62.4	49.6	87	73.2	2209	1520	404.5	188.9
65	100.8	479.7	26.5	1100	1105	1107	70.2	62.4	49.6	87.2	73.2	2210	1576	410.9	190.7
65	100.8	479.9	28.2	1100	1089	1092	70.4	62.5	49.6	87.2	73.2	2207	1632	412.6	193.5
65	100.8	479.9	28.1	1100	1093	1094	70.4	62.5	49.6	87.2	73.3	2205	1683	419.9	193.1
65	100.6	479.1	30.7	1100	1088	1100	70.7	62.5	49.6	87.2	73.4	2203	1728	432.7	198.1
65	100.6	479.3	32.4	1100	1090	1096	70.7	62.7	49.6	87.2	73.4	2207	1768	439.5	204.3
65	100.8	476.7	32.7	1100	1081	1086	70.7	62.7	49.7	87.2	73.4	2200	1801	445.1	215.9
65	100.7	478.6	35.3	1100	1082	1081	70.9	62.8	49.7	87.4	73.4	2202	1832	446.8	209.3
65	100.7	478.3	34.2	1100	1075	1097	70.9	62.8	49.7	87.4	73.4	2201	1861	447.9	209.7
65	100.7	475.7	34.2	1100	1084	1110	70.9	62.8	49.7	87.4	73.5	2199	1890	464.7	210.9
65	100.5	475.7	38	1100	1082	1084	70.9	62.8	49.8	87.4	73.5	2196	1922	487	225.9
65	100.5	472.9	47.1	1100	1082	1084	70.9	63	49.8	87.4	73.6	2192	1956	493.4	232.7
65	100.6	474.8	46.8	1100	1106	1081	71.1	63	49.8	87.4	73.6	2192	1992	491.9	219.1
65	100.7	474.3	47.6	1100	1103	1080	71.1	63	49.8	87.5	73.6	2193	2032	493.6	231.1
65	100.7	475	48.8	1100	1112	1084	71.1	63	49.9	87.6	73.7	2195	2071	496	230.8
65	100.5	475.6	49.2	1100	1106	1085	71.1	63	49.9	87.6	73.7	2190	2104	498.1	222
65	100.6	474.5	48.6	1100	1106	1084	71.3	63.1	49.9	87.7	73.8	2190	2130	500.7	233.4
65	100.8	474.3	50.3	1100	1082	1110	71.4	63	49.9	87.7	73.8	2191	2150	506.9	233.7
65	100.8	473.9	50.5	1100	1088	1103	71.4	63.1	49.8	87.7	73.9	2192	2164	511.1	238
65	100.7	475.6	51.2	1100	1081	1118	71.6	63.1	49.8	87.7	73.9	2191	2173	488.3	236.9
65	100.8	474.1	51.3	1100	1080	1110	71.6	63.2	49.8	87.7	73.9	2193	2176	486.8	227.1
65	100.7	473.5	51.5	1100	1114	1081	71.8	63.2	49.8	87.9	74	2192	2182	521.2	235.9
65	100.7	473.3	53.9	1100	1075	1088	72	63.2	49.8	87.9	74.1	2182	2192	537.7	249.2
65	100.4	474.3	56.4	1100	1096	1089	72	63.4	49.8	87.9	74.1	2184	2203	541.5	251.6
65	100.6	473.6	58.4	1100	1078	1113	72	63.4	49.8	87.9	74.2	2185	2226	544.3	252.7
65	100.7	471.4	57.3	1100	1098	1093	72.1	63.5	49.8	88	74.3	2182	2257	520.3	253.4
65	100.6	471.4	58.4	1100	1099	1083	72.1	63.5	49.8	88.1	74.3	2176	2286	524.4	243.3
65	100.5	465.4	58.1	1100	1086	1078	72.1	63.5	49.9	88.1	74.3	2148	2310	531.4	257.6
65	100.7	462.1	58.1	1100	1095	1097	72.3	63.5	49.9	88.1	74.3	2132	2329	535.3	235.7
65	100.7	458.1	56.2	1100	1077	1076	72.7	63.5	50	88.1	74.5	2113	2344	559.4	250.9
65	100.5	458.9	57.2	1100	1079	1078	72.7	63.7	50	88.1	74.5	2114	2362	558	260
65	100.6	457.8	56.6	1100	1081	1091	72.9	63.7	50.1	88.2	74.5	2104	2389	560.3	261
65	100.6	456.2	56.5	1100	1087	1120	72.9	63.8	50.2	88.2	74.5	2083	2420	540.2	252.3
65	100.7	452.4	57.1	1100	1115	1083	72.9	63.8	50.3	88.4	74.5	2070	2450	520.8	254.5
65	100.6	450.9	56.3	1100	1111	1082	73	63.8	50.4	88.4	74.5	2060	2475	546.8	255.8
65	100.8	450.4	57.3	1100	1079	1105	73	63.8	50.5	88.4	74.5	2059	2490	547.7	255.6
65	100.7	448.2	56.4	1100	1096	1086	73.2	63.9	50.6	88.6	74.5	2038	2493	527	256.6
65	100.5	446	57	1100	1080	1081	73.4	63.9	50.7	88.5	74.5	2027	2492	530.4	247.1
65	100.5	443.7	56.3	1100	1086	1079	73.4	63.9	50.7	88.6	74.5	2016	2501	559.7	248.6
65	100.6	444.3	57.6	1100	1099	1083	73.6	63.9	50.8	88.6	74.5	2022	2515	562.9	261.5
65	100.8	444.7	58.4	1100	1124	1118	73.6	64.1	50.9	88.6	74.5	2018	2528	536.6	263.8
65	100.8	445.7	57.4	1100	1104	1113	73.6	64.1	50.9	88.6	74.3	2017	2546	536.8	250.3
65	100.6	443.5	57.3	1100	1081	1088	73.6	64.1	51	88.6	74.3	2002	2568	543.9	252.2
65	100.5	438.4	58.4	1100	1086	1092	73.6	64.2	51.1	88.8	74.2	1982	2579	545.1	242
65	100.5	438.9	58.5	1100	1083	1095	73.6	64.4	51.2	88.8	74.1	1978	2577	514.8	253.5
65	100.7	439.1	58.1	1100	1085	1078	73.8	64.4	51.2	88.8	74.1	1981	2574	516.3	239.7
65	100.7	436.4	58.4	1100	1109	1076	73.9	64.5	51.2	88.8	74.1	1978	2567	521	254.6
65	100.5	436.7	58	1100	1110	1077	73.9	64.5	51.3	88.9	73.9	1963	2549	499.6	243.4
65	100.5	430.8	58.5	1100	1073	1092	74.1	64.5	51.3	89.1	73.9	1922	2534	535.1	246.7
65	100.7	428.2	57.4	1100	1082	1076	74.3	64.6	51.4	89.1	73.9	1912	2534	538.9	262.9
65	100.5	426.4	58.1	1100	1074	1074	74.3	64.6	51.4	89.1	73.8	1890	2544	533.6	236.1
65	100.5	425.2	57.4	1100	1111	1119	74.5	64.6	51.5	89.1	73.8	1895	2567	556.5	261.7
65	100.4	425.6	57.3	1100	1081	1076	74.5	64.6	51.6	89.2	73.8	1900	2614	570.6	265.6
65	100.6	425.2	56.7	1100	1090	1078	74.5	64.8	51.6	89.1	73.8	1889	2670	575.5	254
65	100.6	425.1	59.1	1100	1119	1123	74.5	64.8	51.8	89.2	73.6	1893	2724	566.7	267.4
65	100.4	424.5	57.4	1100	1104	1086	74.6	64.8	51.9	89.2	73.6	1896	2772	547.9	265.5

65	100.5	425.9	58.2	1100	1084	1083	74.5	64.8	52	89.3	73.6	1895	2800	511.4	245.5
65	100.6	426	57.5	1100	1093	1088	74.6	64.9	52.1	89.3	73.6	1886	2797	504.3	246
65	100.7	424.8	57.1	1100	1117	1119	74.6	64.9	52.1	89.3	73.6	1891	2772	514.4	254.4
65	100.6	424.8	56.7	1097	1082	1089	74.8	64.9	52.2	89.3	73.6	1875	2732	518.4	235.4
65	100.5	424.8	58	1100	1113	1124	75	65.1	52.4	89.3	73.6	1882	2683	516.3	246
65	100.4	424.3	56.8	1099	1118	1116	75.2	65.1	52.5	89.5	73.7	1879	2650	536.2	239.3
65	100.5	425.5	58.6	1097	1081	1120	75.2	65.2	52.6	89.5	73.8	1889	2635	534.2	264.1
65	100.7	426.4	58.3	1100	1069	1097	75.2	65.2	52.7	89.5	73.8	1895	2631	510.3	240.5
65	100.7	427.7	56.9	1100	1112	1090	75.2	65.2	52.7	89.5	73.8	1883	2635	510.9	239
65	100.5	424.8	58.2	1100	1116	1120	75.4	65.2	52.8	89.5	73.8	1886	2642	510.1	253.2
65	100.6	423.8	58.6	1100	1080	1088	75.4	65.2	52.9	89.6	73.8	1885	2639	513.7	241.1
65	100.5	425.4	58.3	1100	1118	1119	75.5	65.4	53	89.8	73.9	1894	2635	564.4	242.1
67	100.7	428.3	60.2	1100	1082	1115	75.5	65.4	53.1	89.8	73.9	1924	2650	606.9	287.5
67	100.6	431.7	64	1100	1121	1084	75.7	65.4	53.2	89.8	74	1919	2683	599.6	262.5
68	100.6	431.7	63.6	1100	1118	1080	75.7	65.4	53.2	89.8	74.1	1932	2726	568.6	267.6
68	100.6	432.7	64.5	1100	1083	1111	75.9	65.4	53.3	89.8	74.1	1945	2783	574	268.6
69	100.4	435.2	68.7	1100	1079	1126	75.9	65.5	53.3	89.8	74.1	1962	2841	619.1	281.4
70	100.5	437.9	70.2	1100	1136	1089	76.1	65.5	53.4	89.8	74.1	1983	2881	624.7	282.2
71	100.6	438.8	71.5	1100	1067	1081	76.1	65.5	53.4	90	74.3	1985	2906	624	293.3
71	100.5	441.9	77.1	1100	1066	1088	76.3	65.6	53.5	90	74.3	2008	2931	640.3	295.7
71	100.8	443.6	78.4	1100	1121	1120	76.3	65.6	53.6	90	74.3	2026	2956	639	321.9
71	100.5	443.6	75.8	1100	1113	1114	76.3	65.6	53.6	90	74.3	2017	2971	624.9	297.1
71	100.5	443.3	15.8	1100	1072	1145	76.4	65.8	53.8	90	74.4	2006	2969	540.2	287.7
71	100.7	424.4	-6.1	1200	1197	1204	76.4	65.8	53.6	90.3	74.5	1815	2937	430.5	213
71	100.6	397.7	-5.3	1200	1207	1184	76.6	65.8	53.4	90.3	74.5	1688	2866	406	189.4
71	100.6	387.9	-5.2	1019	1029	995	76.6	65.9	53.4	90.3	74.5	1637	2762	407.5	190.7
71	100.5	368.5	-4.8	976	972	970	76.8	65.9	53.4	90.3	74.6	1576	2639	406.2	190.6
71	100.4	359.7	-4.3	944	933	949	76.8	65.9	53.4	90.3	74.6	1544	2526	379.9	189.3
71	100.6	372.6	-4.3	927	914	917	76.6	65.9	53.4	90.4	74.6	1527	2446	379	177.6
71	100.4	379.6	-4.3	910	889	912	76.6	65.9	53.4	90.4	74.6	1510	2400	377.3	188.7
71	100.5	383	-4	883	868	878	76.4	65.9	53.4	90.6	74.5	1483	2374	351.9	176.7
71	100.6	383.8	-4.1	874	864	867	76.3	65.8	53.4	90.6	74.5	1474	2361	351.9	164.8
71	100.6	383.8	-4.1	874	864	867	76.3	65.8	53.4	90.6	74.5	1474	2314	195.8	164.8
0	100.7	388.4	-3.1	656	657	641	48.9	47.6	51.3	67.6	34.2	944	2184	39.6	20.8
0	100.7	385.1	-3	659	666	667	48.8	47.4	51	67.6	34.2	946	1969	35.1	20.8
0	100.7	374.3	-3.2	658	660	662	48.7	47.2	50.4	67.7	34.3	946	1682	34.6	18.7
0	100.8	369.9	-2.9	658	660	661	48.7	47.2	50.1	67.6	34.4	945	1326	38.7	20.5
0	100.8	367.9	-3.4	658	657	662	48.6	47.2	49.8	67.6	34.4	946	979	38.7	20.5
0	100.7	365.1	-3.2	658	658	660	48.6	47.2	49.5	67.6	34.4	946	710	38.5	20.4
0	100.8	363.6	-2.8	658	661	664	48.6	47.2	49.2	67.6	34.4	945	522	34.6	18
0	100.7	362.9	-2.9	658	660	658	48.7	47.3	48.9	67.5	34.4	945	411	34.4	18.5
0	100.9	360.4	-3.1	657	658	660	48.8	47.3	48.3	67.3	34.4	945	373	34.2	20.2
0	100.7	362.1	-2.5	655	655	658	48.9	47.3	48.1	67.3	34.5	946	368	34.2	18.4
0	100.9	363.8	-2.4	658	658	653	48.9	47.4	47.8	67.3	34.5	945	363	34.2	20.2
0	100.7	365.1	-2.7	656	660	659	48.9	47.4	47.6	67.2	34.6	946	357	34.2	18.2
0	100.7	366.9	-2.7	656	654	660	49	47.5	47.4	67.2	34.7	945	353	38.1	17.9
0	100.8	368.5	-2.8	657	657	656	49.1	47.6	47.2	67.1	34.7	945	352	37.8	20.1
0	100.8	371	-2.8	656	657	654	49.2	47.6	46.8	67.1	34.9	945	355	37.8	19.9
0	100.8	371.3	-3.1	655	653	655	49.3	47.6	46.6	67.1	35	945	359	34	20
0	100.7	371.7	-2.5	654	658	657	49.3	47.7	46.4	67	35.1	945	363	33.8	18.1
0	100.7	373.4	-2.9	655	659	658	49.4	47.7	46.2	66.9	35.1	945	366	37.6	17.7
0	100.8	373.4	-2.8	655	656	653	49.4	47.8	46	66.9	35.2	945	367	37.6	19.9
0	100.6	373.9	-3	655	656	655	49.5	47.8	45.9	66.9	35.3	945	366	37.6	20
0	100.6	375.2	-2.7	656	655	659	49.5	47.8	45.7	66.9	35.5	944	367	37.6	17.7
0	100.8	375.7	-2.9	655	651	659	49.6	47.9	45.4	66.9	35.6	945	370	37.6	19.9
0	100.8	376.6	-2.5	656	659	652	49.7	47.9	45.3	66.8	35.8	944	373	37.6	19.8
0	100.8	376.7	-3.2	655	655	654	49.7	47.9	45.2	66.8	35.9	945	374	33.8	19.9
0	100.6	377.8	-3.1	655	656	658	49.7	47.9	45	66.8	36	944	374	33.8	18
0	100.6	377	-2.4	655	654	657	49.7	47.9	44.9	66.8	36.1	944	373	37.6	19.9
0	100.6	377.5	-2.7	654	654	655	49.8	48	44.7	66.8	36.3	946	369	37.4	19.8
0	100.7	378.7	-2.8	655	657	652	49.8	48	44.5	66.6	36.3	945	366	37.4	19.8
0	100.6	379.1	-3	654	654	656	49.8	47.9	44.4	66.6	36.5	946	366	37.4	19.9

0	100.8	379.1	-2.9	655	654	656	49.9	48	44.3	66.6	36.5	945	368	37.2	19.7
0	100.9	379.3	-3.2	655	655	653	49.9	48	44.2	66.6	36.6	945	369	33.6	19.8
2	100.7	380.1	-2.9	655	653	655	49.9	48	44.1	66.6	36.7	946	369	33.6	18
13	100.8	379.9	-3	655	654	653	49.9	48	43.9	66.5	36.9	944	370	41.3	17.5
21	100.8	384.2	-3.2	734	710	659	50	48	43.8	66.5	37	981	377	56.9	21.7
25	100.6	406.2	-3.2	815	773	694	50	48	43.6	66.5	37.1	1079	394	74.6	28.5
28	100.6	416.8	-4	944	988	1002	50	48.1	43.5	66.5	37.2	1231	431	103.7	38.9
33	100.8	407	-4.4	1132	1083	1214	50	48.1	43.5	66.5	37.4	1335	493	129.3	53.1
37	100.6	451.9	-5	1200	1204	1205	50	48.1	43.3	66.4	37.5	1484	579	165.9	76.4
44	100.6	503.8	-5.2	1200	1216	1194	50.1	48.1	43.2	66.4	37.6	1708	683	209.9	86.1
47	100.6	545.3	-5.1	1200	1215	1202	50.1	48.1	43	66.4	37.7	1879	802	257	99.8
54	100.7	593.5	-2.5	1200	1206	1204	50.2	48.1	42.6	66.4	38	2219	909	229.6	126.4
58	100.8	601.8	-2.9	1200	1198	1198	50.4	48.2	42.4	66.3	38.1	2256	985	166.1	89.7
61	100.6	597.6	-3.3	1200	1199	1193	50.4	48.2	42.2	66.3	38.4	2235	1020	169.9	80.4
63	100.5	597.8	-2.8	1200	1200	1197	50.5	48.2	41.9	66.2	38.6	2240	1014	176.1	82.4
65	100.8	597.7	-2.8	1200	1197	1207	50.5	48.3	41.7	66.2	38.8	2247	960	177.4	82.5
71	100.8	598.4	-2.7	1200	1199	1196	50.5	48.3	41.5	66.2	39	2246	890	182.4	82.1
80	100.8	596.2	-2.2	1200	1198	1198	50.6	48.3	41.1	66.2	39.4	2244	833	190.9	84.3
84	100.6	595.7	-2.3	1200	1203	1200	50.6	48.3	41	66.2	39.5	2235	792	182.3	90
87	100.7	595.2	-2.5	1200	1198	1203	50.6	48.3	40.8	66.2	39.6	2251	767	172.7	84
92	100.5	595.1	-2.6	1200	1202	1200	50.7	48.4	40.7	66.2	39.8	2242	764	175.3	81.5
94	100.6	592.6	-2.7	1200	1196	1198	50.7	48.4	40.5	66.1	40	2239	768	175.1	83.8
98	100.8	594.2	-2.4	1200	1201	1196	50.7	48.4	40.4	66.1	40.1	2242	766	178.5	84
100	100.5	592.3	-1.8	1200	1196	1199	50.9	48.5	40.2	66.1	40.6	2250	762	182.1	88.7
99	100.6	592.5	-2.5	1200	1193	1197	50.9	48.5	40.1	66.1	40.7	2250	759	178.7	81.5
98	100.4	592.3	-1.6	1200	1195	1201	51	48.6	40	66.1	40.8	2237	757	178.9	82.2
98	100.6	588.8	-2.2	1200	1203	1203	51	48.6	39.9	66.1	41	2247	757	176.1	84
98	100.5	592.7	-2.3	1200	1193	1197	51.2	48.6	39.8	66.1	41.1	2244	759	184.3	81.9
98	100.6	591.7	-2.5	1200	1209	1208	51.3	48.6	39.8	66	41.3	2247	764	194.5	90
98	100.7	588.4	-2.1	1200	1195	1196	51.4	48.6	39.6	66.1	41.6	2237	770	190.1	89.3
98	100.6	588.4	-2.2	1200	1196	1193	51.4	48.7	39.6	66	41.8	2245	776	183	89.5
98	100.6	588.8	-1.7	1200	1198	1200	51.5	48.7	39.5	65.9	41.9	2242	782	179.1	85.7
98	100.5	586.5	-1.9	1200	1197	1198	51.6	48.7	39.5	65.9	42	2245	788	183.2	84
98	100.7	587.2	-2.1	1200	1194	1202	51.6	48.8	39.4	65.9	42.2	2249	789	187	86.2
98	100.8	587.7	-1.7	1200	1195	1193	51.7	48.8	39.4	65.9	42.3	2237	789	190.7	90.8
98	100.7	583.5	-1.7	1200	1206	1195	51.8	48.9	39.3	65.9	42.5	2242	792	202.9	91.3
98	100.7	585.3	-1.4	1200	1198	1204	52	48.9	39.3	65.9	42.7	2237	800	215	98.5
98	100.5	580.5	-0.3	1200	1197	1203	52.1	48.9	39.2	65.9	42.9	2235	816	226	103.8
98	100.6	579.3	0	1200	1206	1190	52.1	49	39.2	65.9	43.1	2243	842	238.6	110.1
98	100.6	580.5	0.5	1200	1197	1201	52.2	49	39.2	65.9	43.3	2242	880	265.5	116.2
98	100.6	580.7	1.3	1200	1197	1197	52.3	49.1	39.2	65.9	43.4	2238	932	299.7	129.8
98	100.7	575.2	2.8	1200	1201	1196	52.4	49.1	39.2	65.8	43.7	2233	993	301	138.4
98	100.6	575.3	3.8	1200	1198	1201	52.5	49.1	39.2	65.9	43.8	2232	1058	298	135.3
98	100.4	575.7	4.5	1200	1196	1192	52.5	49.2	39.1	65.8	43.9	2227	1128	316	140.8
98	100.7	573.8	5.1	1200	1210	1202	52.6	49.2	39.1	65.9	44.1	2224	1197	332	150.6
98	100.6	574	6.4	1200	1193	1209	52.8	49.3	39.1	65.8	44.3	2221	1257	341.2	158.7
98	100.7	573.5	6.9	1200	1194	1191	52.8	49.3	39.1	65.8	44.4	2222	1310	347.6	161.7
98	100.5	572.8	7.7	1200	1192	1210	52.9	49.3	39.1	65.8	44.6	2221	1361	351.9	164.9
98	100.5	570.7	8.2	1200	1204	1199	53	49.4	39.1	65.8	44.8	2219	1408	371.3	168.8
98	100.6	568.4	11.3	1200	1206	1210	53.1	49.5	39.2	65.8	45	2211	1454	392.7	180.7
98	100.8	563.6	11.7	1200	1189	1200	53.2	49.5	39.2	65.8	45.2	2211	1502	399.1	185.5
98	100.5	565.8	13.2	1200	1209	1197	53.3	49.6	39.2	65.8	45.3	2209	1553	404.7	187.4
98	100.6	566.3	13.5	1200	1196	1197	53.5	49.7	39.2	65.8	45.4	2211	1604	407.9	190.6
98	100.6	562.6	13.3	1200	1211	1198	53.6	49.9	39.2	65.9	45.6	2210	1655	425	190.7
98	100.6	560.8	13.6	1200	1196	1193	53.7	50	39.3	65.8	45.7	2208	1700	423.7	198.3
98	100.6	562.1	14.3	1200	1201	1186	53.8	50.1	39.3	65.9	45.9	2209	1741	433.1	192.7
98	100.6	560.5	16.4	1200	1195	1195	54	50.1	39.4	65.9	46.3	2195	1783	465.2	212.6
98	100.6	559.7	18	1200	1193	1194	54.1	50.2	39.4	65.9	46.5	2196	1826	468.2	219.1
98	100.7	558.4	19.2	1200	1215	1213	54.3	50.4	39.5	65.9	46.6	2199	1868	473.5	218.5
98	100.8	558.5	20.2	1200	1201	1191	54.5	50.4	39.5	65.9	46.8	2199	1914	473.9	222.7
98	100.8	560.3	19.5	1200	1195	1187	54.6	50.5	39.6	65.9	46.9	2195	1960	468.8	219.6
98	100.7	559.1	19.2	1200	1215	1207	54.7	50.5	39.7	66	47.1	2198	1997	473.1	220.7

98	100.5	556.3	21.2	1200	1194	1194	54.8	50.5	39.7	66	47.3	2199	2024	475.9	222.7
98	100.6	558.5	20.6	1200	1207	1203	55.2	50.5	39.8	66.1	47.5	2197	2043	476.1	222.3
98	100.6	557.3	21.2	1200	1190	1189	55.3	50.5	39.9	66.1	47.7	2197	2053	473.7	222.3
98	100.6	554.5	19.1	1200	1199	1187	55.5	50.6	40	66.1	47.8	2195	2056	471.2	221.1
98	100.5	557	19.6	1200	1192	1187	55.6	50.7	40	66.1	48	2194	2059	474.6	220.3
98	100.5	553.9	20	1200	1213	1202	55.8	50.7	40.1	66.2	48.1	2199	2060	475.9	223.9
98	100.6	557.4	20.8	1200	1194	1187	55.9	50.8	40.2	66.2	48.3	2198	2061	479.7	221.4
98	100.5	551.3	23.4	1200	1189	1183	56.2	50.9	40.3	66.2	48.6	2183	2070	509.2	234.3
98	100.5	555.9	27.8	1200	1183	1189	56.3	50.9	40.4	66.3	48.8	2183	2089	536.4	250.4
98	100.5	553.2	29.2	1200	1183	1196	56.4	51	40.5	66.3	48.9	2185	2120	541.5	253.4
98	100.5	553.5	30.3	1200	1191	1187	56.7	51.3	40.6	66.3	49.1	2185	2164	542.8	253.3
98	100.6	546.3	31.3	1200	1196	1181	56.8	51.3	40.7	66.3	49.3	2187	2218	541.9	253.3
98	100.7	550.8	29.1	1200	1189	1184	57	51.4	40.8	66.5	49.5	2186	2271	543	254
98	100.7	551.3	30.2	1200	1201	1201	57.2	51.4	41	66.5	49.8	2184	2314	543	252.1
98	100.5	548.1	31.2	1200	1192	1183	57.4	51.5	41.1	66.5	49.9	2181	2344	543	253.5
98	100.7	548.4	32.3	1200	1193	1191	57.5	51.5	41.2	66.6	50	2179	2364	546	254.7
98	100.6	544.9	33.1	1194	1178	1212	57.6	51.6	41.3	66.6	50.2	2177	2375	558.2	257.1
98	100.7	548	34.4	1100	1116	1118	57.8	51.7	41.4	66.6	50.4	2173	2385	565	267.2
98	100.8	546.5	38.3	1200	1178	1196	57.9	51.8	41.6	66.7	50.5	2181	2398	560.5	263.3
98	100.6	544.1	39.6	1173	1117	1134	58.2	51.8	41.8	66.8	50.9	2179	2413	561.6	264.8
98	100.7	544.8	41.5	1127	1136	1088	58.4	52	41.9	66.8	51.1	2176	2432	575.5	265
98	100.7	544.2	43	1100	1096	1140	58.5	52.1	42.1	66.9	51.3	2173	2453	584.5	269.1
98	100.6	542.1	47.1	1100	1090	1083	58.6	52.2	42.3	66.9	51.5	2176	2469	561.6	272.4
98	100.4	546.7	47.4	1100	1089	1116	58.7	52.3	42.4	67	51.7	2164	2485	572.9	268.4
98	100.5	543.8	48.7	1100	1090	1125	58.9	52.4	42.5	67	51.8	2173	2501	579.5	267.8
98	100.5	543.2	52.1	1100	1085	1083	59.2	52.5	42.9	67.2	52.4	2162	2511	570.6	271.1
98	100.5	541.2	55.3	1100	1110	1105	59.3	52.6	43	67.2	52.5	2156	2519	593	270.4
98	100.5	538.3	58.3	1100	1084	1116	59.4	52.6	43.2	67.3	52.7	2157	2535	597.7	293.5
98	100.5	540.6	62.5	1100	1119	1084	59.7	52.9	43.4	67.3	52.9	2153	2551	591.5	278.3
98	100.6	540.6	60.7	1100	1092	1103	59.8	52.9	43.6	67.3	53.1	2151	2568	595.4	277.5
98	100.4	537.7	59.5	1100	1084	1093	60	53	43.8	67.5	53.3	2154	2591	601.1	295
98	100.5	542.5	62	1100	1073	1076	60.1	53	44	67.5	53.5	2152	2613	597.5	279.5
98	100.5	538	60.8	1100	1081	1086	60.6	53.1	44.3	67.6	53.8	2154	2627	592.6	293.6
98	100.7	541.6	56.8	1100	1084	1081	60.7	53.2	44.5	67.6	53.9	2185	2631	566.7	275.8
98	100.7	545.2	52.4	1200	1187	1125	60.9	53.2	44.7	67.8	54.1	2194	2609	478.6	248.3
98	100.5	546.8	37.6	1200	1203	1185	61	53.3	44.9	67.8	54.3	2212	2540	383.5	208.2
98	100.6	544.6	22.2	1200	1192	1203	61.3	53.3	45	67.9	54.6	2226	2424	346.7	174.2
98	100.5	547.8	14.3	1200	1201	1210	61.6	53.6	45.1	68	54.8	2233	2257	308.3	148.8
98	100.6	548.6	8.8	1200	1205	1191	61.8	53.7	45.2	68	55	2239	2041	267.9	127.4
98	100.4	549.7	4.9	1177	1167	1178	62.3	53.8	45.4	68.2	55.4	2246	1808	249.7	118.2
98	100.7	548.9	2.9	1164	1172	1169	62.4	53.8	45.5	68.3	55.6	2249	1587	229.6	110.1
98	100.7	545.8	1.7	1152	1146	1158	62.5	53.9	45.7	68.3	55.8	2255	1387	196.9	105.7
98	100.6	548.9	0.7	1143	1136	1138	62.7	54.1	45.8	68.4	56	2247	1219	186.4	83.4
98	100.5	546.4	0.7	1129	1126	1130	62.8	54.3	45.9	68.5	56.2	2242	1090	189.6	94
98	100.4	547.9	1.1	1111	1116	1119	63.1	54.4	46.1	68.6	56.5	2236	991	180	84.5
98	100.5	545	0.7	1102	1101	1101	63.1	54.5	46.3	68.6	56.6	2245	909	177.2	82.8
98	100.5	545.3	0.8	1093	1093	1102	63.2	54.5	46.4	68.7	56.8	2239	852	183	81.5
98	100.5	547.3	0.8	1086	1084	1077	63.2	54.6	46.5	68.7	56.9	2250	818	193.7	85.8
98	100.7	545.1	0.1	1079	1079	1071	63.4	54.7	46.6	68.7	57	2237	795	182.3	89.8
98	100.5	546.6	1.1	1072	1061	1060	63.4	54.7	46.7	68.9	57.1	2244	783	176.1	77
98	100.6	543.8	1	1066	1058	1053	63.5	54.7	46.8	68.9	57.2	2240	780	173	81.3
98	100.4	543.5	1.2	1052	1055	1052	63.5	54.7	47	69	57.5	2253	774	159.5	77.3
98	100.3	544.2	1.5	1042	1046	1049	63.5	54.7	47.1	69	57.6	2243	761	156.9	74
98	100.5	541.3	1.6	1038	1039	1032	63.5	54.8	47.2	69	57.7	2248	745	156.1	74.5
98	100.6	540.2	1.1	1031	1034	1027	63.5	54.8	47.3	69.1	57.8	2254	726	155.8	75.4
98	100.5	542.9	1.5	1024	1023	1029	63.5	54.8	47.3	69.2	57.9	2244	705	159.7	73.2
98	100.6	540.9	1.1	1018	1018	1013	63.5	54.8	47.5	69.2	58	2248	689	162.2	75.9
98	100.4	541.8	2	1014	1012	1003	63.5	54.8	47.5	69.2	58.1	2248	679	158.2	74.2
98	100.4	541.9	2.3	1003	1007	1003	63.4	54.8	47.7	69.3	58.3	2244	674	158.8	71.9
98	100.5	538.8	1.6	997	996	1009	63.4	54.8	47.7	69.3	58.4	2247	674	163.1	74.2
98	100.7	537.7	1.9	996	989	987	63.4	54.8	47.8	69.3	58.4	2243	677	163.5	74.3
98	100.5	538.3	2.3	990	988	984	63.4	54.8	47.8	69.3	58.5	2240	682	173	80

98	100.5	539.2	2.5	984	981	976	63.4	54.8	47.8	69.4	58.6	2250	691	186.6	83.7
98	100.5	541.4	2.7	983	979	983	63.2	54.8	47.9	69.4	58.6	2247	703	178.9	83.1
98	100.5	536.6	3	980	981	977	63.1	54.8	48	69.4	58.7	2242	717	178.1	82.6
98	100.5	536.7	3	976	975	981	63.1	54.8	48	69.5	58.7	2237	736	202.2	84.5
98	100.6	534.8	4.3	969	957	962	63.1	54.8	48	69.5	58.9	2242	761	216.6	106.2
98	100.5	537.7	5.4	966	956	959	63.1	54.8	48.1	69.6	58.9	2248	787	221.7	101.3
98	100.4	534.9	5.7	962	956	952	63	54.8	48.1	69.6	59	2246	818	229.4	110.7
98	100.4	535.8	7.8	950	946	951	63	54.8	48.1	69.6	59.1	2244	856	234.5	108.6
98	100.5	534.1	7.5	948	950	942	63	54.8	48.2	69.6	59.2	2241	894	237.9	110.2
98	100.5	532.9	7.4	942	935	942	63	54.8	48.2	69.7	59.3	2241	930	247.6	111.5
98	100.7	535.6	8.4	939	934	937	63	54.8	48.2	69.7	59.4	2240	968	274.5	116.7
98	100.4	532.7	10.2	966	950	974	62.8	54.8	48.2	69.7	59.4	2233	1008	287.1	133.1
98	100.6	532.2	10.6	977	964	975	62.8	54.8	48.3	69.7	59.4	2233	1047	282.8	137.7
98	100.5	531.1	10.8	977	982	971	62.7	54.8	48.3	69.7	59.5	2233	1090	291.2	134.5
98	100.6	531.7	13	996	980	989	62.7	54.8	48.3	69.9	59.7	2229	1136	298.2	138.8
98	100.5	529.8	12.5	1003	989	980	62.7	54.8	48.3	69.9	59.8	2227	1181	322.8	145.7
98	100.5	533	13.3	1014	1022	1013	62.7	54.8	48.3	69.9	59.9	2227	1225	340.1	154.2
98	100.5	531.8	14.5	1053	1040	1072	62.7	54.8	48.4	70	59.9	2220	1272	344.6	158.8
98	100.4	530.9	16	1060	1063	1078	62.7	54.8	48.4	70	60	2221	1322	361.3	167.5
98	100.6	529	17	1086	1092	1088	62.7	54.8	48.4	70	60	2217	1376	372.2	180.7
98	100.7	530.9	19.3	1090	1088	1111	62.7	54.8	48.4	70.1	60.1	2218	1431	375.4	175.1
98	100.5	527.7	20.1	1100	1088	1105	62.7	54.8	48.5	70.2	60.3	2213	1483	375	175.1
98	100.7	527.3	20.3	1100	1085	1088	62.7	54.8	48.5	70.2	60.3	2216	1531	386.1	178.3
98	100.4	527	21.1	1100	1092	1089	62.8	54.8	48.5	70.2	60.4	2207	1575	405.1	187.1
98	100.6	527.2	23.4	1100	1091	1085	62.8	54.9	48.5	70.4	60.5	2211	1614	414.3	193.8
98	100.6	525.2	25.2	1100	1086	1095	62.8	54.9	48.5	70.4	60.6	2207	1650	414.1	193.5
98	100.5	523.7	27.1	1100	1097	1099	62.8	54.9	48.6	70.5	60.7	2208	1687	421.1	195
98	100.6	523.5	27.8	1100	1089	1107	62.8	54.9	48.6	70.5	60.9	2204	1727	435.5	197.5
98	100.6	522.7	28.5	1100	1094	1091	63	54.9	48.6	70.5	60.9	2204	1765	445.9	205.7
98	100.6	524.9	31.3	1100	1081	1092	63	54.9	48.6	70.7	61	2203	1803	448.9	209.3
98	100.4	521.3	31.3	1100	1083	1092	63.1	55.1	48.6	70.7	61.1	2198	1840	450.2	209.7
98	100.6	523.1	30.8	1100	1108	1102	63.1	55.1	48.7	70.7	61.2	2203	1874	451.3	211.4
98	100.4	524.8	32.7	1100	1084	1087	63.2	55.1	48.7	70.8	61.3	2203	1902	450.2	211
98	100.5	520.4	32.9	1100	1089	1083	63.2	55.1	48.7	70.9	61.4	2197	1926	459.6	210.4
98	100.5	524.3	33.5	1100	1100	1085	63.4	55.1	48.7	70.9	61.5	2197	1950	488.9	223.1
98	100.5	522.1	43.2	1100	1096	1083	63.4	55.2	48.8	71.1	61.8	2192	1978	506.2	237.8
98	100.5	520.6	45.2	1100	1085	1083	63.5	55.2	48.9	71.1	61.8	2190	2009	501.1	236.3
98	100.8	519	44.9	1100	1110	1112	63.5	55.2	48.9	71.2	62	2193	2047	501.1	232.9
98	100.7	517.1	45.9	1100	1112	1106	63.7	55.3	49	71.3	62.1	2193	2091	502.4	235.1
98	100.5	519.7	45.8	1100	1103	1097	63.7	55.3	49	71.3	62.1	2193	2129	502.6	233.8
98	100.5	519.2	45.3	1100	1082	1093	63.8	55.3	49.1	71.4	62.3	2195	2157	504.5	236.5
98	100.5	519	46.2	1100	1079	1083	63.8	55.3	49.1	71.4	62.4	2187	2177	503.9	235.3
98	100.7	520.6	46.8	1100	1075	1097	63.9	55.4	49.3	71.6	62.5	2193	2188	503.9	223.2
98	100.5	518.4	47.4	1100	1081	1111	64.1	55.4	49.3	71.7	62.7	2195	2191	503.9	235.3
98	100.7	517.6	46.5	1100	1087	1095	64.2	55.5	49.4	71.8	62.7	2189	2192	502.2	235.1
98	100.5	518	47.5	1100	1108	1082	64.2	55.5	49.4	71.8	62.9	2190	2192	498.3	234.7
98	100.5	519.7	46.3	1100	1083	1081	64.4	55.6	49.5	72	63	2190	2187	474.2	231.5
98	100.6	517.7	46.4	1100	1094	1083	64.4	55.6	49.5	72	63.1	2191	2176	474.6	222.3
98	100.6	520.8	46.6	1100	1078	1079	64.5	55.8	49.6	72.1	63.2	2190	2164	497	233
98	100.6	516.5	47.3	1100	1105	1086	64.6	55.8	49.6	72.1	63.4	2193	2151	498.3	221.7
98	100.6	518.9	47	1100	1101	1112	64.8	55.8	49.7	72.3	63.5	2193	2139	500.7	234.1
98	100.7	517.3	47.1	1100	1087	1094	64.8	55.9	49.7	72.3	63.6	2193	2137	501.9	235.1
98	100.5	517.7	47.9	1100	1097	1081	65.1	56	49.8	72.5	63.7	2193	2145	499.6	234.4
98	100.4	516.3	48.9	1100	1083	1085	65.1	56	49.8	72.5	63.8	2195	2153	498.7	233.6
98	100.7	516.8	48.1	1100	1082	1078	65.2	56	49.9	72.7	64	2192	2163	499.2	234.9
98	100.7	514.8	47.7	1100	1092	1093	65.4	56.1	49.9	72.7	64.1	2188	2172	498.5	233.3
98	100.7	516.1	47.3	1100	1115	1104	65.5	56.1	50	72.9	64.1	2189	2176	500	234.4
98	100.5	516.3	47.5	1100	1102	1110	65.5	56.1	50	72.9	64.2	2189	2176	500.2	233.6
98	100.5	517.6	48.5	1100	1089	1095	65.6	56.2	50.1	72.9	64.4	2190	2177	501.1	233.6
98	100.6	514.3	48.8	1100	1084	1091	65.8	56.3	50.3	73	64.5	2190	2178	499.2	234.7
98	100.6	513.9	48	1100	1112	1083	65.8	56.3	50.4	73.2	64.6	2194	2178	498.5	232.5
98	100.5	513.8	47.1	1100	1105	1092	65.9	56.3	50.5	73.1	64.8	2193	2179	498.3	232.9

98	100.7	512.6	47.7	1100	1086	1099	66.1	56.3	50.6	73.4	64.9	2195	2178	499.2	231.7
98	100.6	515.9	47.1	1100	1106	1109	66.1	56.4	50.7	73.4	65	2194	2177	500	235.9
98	100.5	515.1	47	1100	1092	1085	66.2	56.4	50.8	73.4	65.1	2188	2175	499.2	232.5
98	100.5	514	48.4	1100	1088	1117	66.2	56.6	50.9	73.6	65.2	2190	2175	500.2	234
98	100.6	513	47.2	1100	1096	1090	66.5	56.6	51.2	73.7	65.5	2193	2175	501.1	232.4
98	100.6	513.1	48	1100	1081	1109	66.5	56.6	51.3	73.8	65.6	2194	2172	479.7	234
98	100.7	512.2	47.3	1100	1083	1101	66.6	56.7	51.4	73.8	65.6	2192	2166	478.9	221.9
98	100.6	514.5	47.8	1100	1117	1117	66.6	56.7	51.4	73.9	65.8	2190	2160	499.2	233.7
98	100.6	512.9	48.8	1100	1087	1105	66.6	56.8	51.5	74	65.9	2189	2153	495.7	222.8
98	100.8	510.9	48.3	1100	1087	1093	66.8	56.8	51.6	74.1	65.9	2188	2144	494.7	231.8
98	100.8	511.3	48.7	1100	1077	1111	66.9	56.9	51.8	74.3	66.2	2190	2142	496.6	232.5
98	100.6	507.2	48.1	1100	1093	1097	66.9	56.9	51.9	74.3	66.3	2194	2146	496.4	231.7
98	100.6	504.3	48.9	1100	1114	1112	67	57	52	74.3	66.3	2194	2149	493.2	231.9
98	100.8	509.9	47.7	1100	1106	1083	67	57.1	52	74.5	66.3	2193	2153	495.5	219.8
98	100.5	508.4	47.8	1100	1080	1104	67.2	57.1	52.1	74.5	66.5	2193	2158	497.5	233.9
98	100.5	504.3	48.5	1100	1109	1113	67.3	57.2	52.1	74.6	66.6	2195	2160	495.1	231
98	100.8	503.8	48.2	1100	1102	1083	67.5	57.2	52.2	74.7	66.8	2192	2158	494.7	232.4
98	100.6	507.8	48.1	1100	1093	1102	67.5	57.4	52.3	74.8	66.9	2189	2157	492.1	231.4
98	100.7	503.1	47.6	1100	1076	1095	67.6	57.4	52.4	74.8	67	2191	2156	491.3	230.2
98	100.7	503.1	48.7	1100	1108	1094	67.6	57.5	52.4	75	67.1	2194	2154	493.8	231.1
98	100.5	503.9	48.4	1100	1096	1078	67.6	57.5	52.5	75	67.3	2194	2151	484.9	232.9
98	100.7	502.7	49.7	1100	1100	1100	67.6	57.6	52.6	75.2	67.5	2187	2149	499.6	234.1
98	100.7	502.7	52.4	1100	1110	1119	67.8	57.6	52.6	75.2	67.5	2187	2154	530.2	233.5
98	100.7	501.9	54.1	1100	1082	1111	67.8	57.7	52.7	75.3	67.6	2183	2167	538.5	251.9
98	100.7	499.6	56.2	1100	1104	1106	67.9	57.7	52.9	75.4	67.8	2182	2191	542.1	254.1
98	100.7	502.4	57.6	1100	1096	1122	68	57.8	53	75.5	67.9	2179	2227	545.1	255.6
98	100.8	500.5	57.9	1100	1107	1104	68	57.8	53.1	75.5	67.9	2178	2272	547.7	254.8
98	100.7	499.6	58.7	1100	1104	1077	68.2	57.9	53.2	75.7	68	2179	2313	547.5	257.8
98	100.7	498.6	58.2	1100	1074	1097	68.2	57.9	53.3	75.7	68.1	2182	2343	524.8	255.8
98	100.7	497.5	57.2	1100	1101	1096	68.3	57.9	53.4	75.9	68.2	2180	2358	508.6	248.7
98	100.7	500.4	58.8	1100	1076	1110	68.5	58	53.5	75.9	68.3	2173	2362	546	255.2
98	100.6	496.9	63.4	1100	1100	1076	68.5	58	53.7	76.1	68.6	2176	2364	582.6	271.8
98	100.6	498.2	63.5	1100	1081	1098	68.6	58.2	53.8	76.1	68.6	2176	2369	584.3	273.2
98	100.7	497	64.8	1100	1112	1124	68.7	58.2	53.9	76.1	68.7	2177	2386	579.3	258.8
98	100.7	496.8	63.9	1100	1075	1117	68.7	58.3	54	76.3	68.9	2173	2422	578.7	270.2
98	100.6	496	63.3	1100	1077	1121	68.9	58.4	54.1	76.3	68.9	2174	2461	568.2	275
98	100.5	497	67	1100	1100	1083	69	58.4	54.2	76.4	69.1	2170	2494	581.7	270.5
98	100.6	495.8	68.1	1100	1122	1119	69.2	58.5	54.4	76.4	69.2	2170	2524	601.8	282.9
98	100.5	497.1	68.3	1100	1096	1118	69.2	58.5	54.5	76.6	69.3	2171	2547	599.9	282
98	100.5	496.1	67.7	1100	1110	1099	69.2	58.6	54.5	76.6	69.4	2165	2563	603.9	282
98	100.7	494.7	67.7	1100	1116	1081	69.3	58.7	54.6	76.6	69.4	2172	2582	604.1	281.6
98	100.5	496.4	67.8	1100	1109	1080	69.3	58.9	54.6	76.8	69.6	2165	2601	580.8	282.4
98	100.4	495.6	68.7	1100	1085	1077	69.4	58.9	54.7	76.8	69.7	2170	2613	583.4	271.3
98	100.6	494.4	70.8	1100	1098	1115	69.4	58.9	54.8	77	69.9	2170	2623	611	283.4
98	100.6	494.5	69.2	1100	1113	1117	69.4	59	54.9	77.1	70	2171	2629	610.8	285.4
98	100.7	493.5	68.8	1100	1077	1085	69.6	59	54.9	77.1	70	2166	2630	606.9	282.7
98	100.7	494.6	70	1100	1100	1099	69.7	59.1	55	77.1	70.2	2170	2634	603.1	284
98	100.6	495.3	70.4	1100	1077	1102	69.7	59.1	55.1	77.3	70.2	2167	2645	603.3	280.9
98	100.6	493.3	68.6	1100	1081	1104	69.9	59.1	55.2	77.3	70.4	2166	2648	577.2	283.1
98	100.5	493.6	68.2	1100	1081	1102	70.2	59.2	55.3	77.5	70.6	2165	2645	578.9	270.5
98	100.4	493.8	70.2	1100	1119	1102	70.2	59.2	55.4	77.5	70.7	2172	2641	608.4	285.4
98	100.6	492.3	69.8	1100	1080	1106	70.4	59.2	55.5	77.7	70.9	2170	2634	603.5	283.1
98	100.7	493.1	69.6	1100	1114	1097	70.4	59.3	55.6	77.7	71	2173	2622	604.6	281
98	100.6	494.5	-11	1200	1103	1091	70.5	59.3	55.9	77.9	71.2	2143	2616	563.9	286.2
98	100.7	481.8	-10.2	1200	1184	1183	70.5	59.4	55.8	78	71.3	2027	2610	517.1	241.9
98	100.5	469.9	-9.3	1200	1189	1183	70.7	59.4	55.6	78	71.6	1970	2597	515	242.3
98	100.7	460.9	-9.6	1200	1183	1178	70.7	59.5	55.6	78.2	71.6	1936	2574	487.4	240.7
98	100.5	455.5	-9.2	1100	1073	1081	70.9	59.7	55.6	78.2	71.8	1903	2536	458.8	226.7
98	100.6	446.8	-9	1084	1097	1064	70.9	59.7	55.6	78.4	71.8	1854	2499	482.5	225.3
98	100.6	444.5	-8.8	1089	1067	1057	70.9	59.8	55.6	78.5	72	1867	2474	510.9	225.6
98	100.6	456.1	-9.4	1100	1124	1084	70.9	59.8	55.7	78.6	72	1945	2426	361.3	239.6
99	100.7	549.4	-3	1200	1208	1206	55.9	52	38.4	63	54.4	2235	2327	198.2	92.7

99	100.5	548.7	-3.1	1200	1196	1196	56	52	38.4	63.1	54.4	2235	2182	186.6	92.3
99	100.5	547.7	-3.1	1200	1195	1202	56.1	52	38.4	63.1	54.5	2234	1970	186.6	87.5
99	100.7	549.5	-3.8	1200	1193	1192	56.1	52.1	38.4	63.1	54.5	2239	1679	189	88.3
99	100.5	547.8	-3.7	1200	1193	1208	56.1	52.1	38.4	63.1	54.6	2243	1378	192.2	93.7
99	100.6	546.4	-3.4	1200	1205	1212	56.1	52.1	38.4	63.2	54.6	2246	1138	190.9	88.9
99	100.6	547.5	-2.9	1200	1203	1206	56.1	52.1	38.4	63.2	54.7	2246	959	189.8	88.5
99	100.7	545.4	-3.1	1200	1195	1199	56.2	52.2	38.4	63.2	54.8	2238	849	192.6	90.9
99	100.5	547.7	-2.8	1200	1193	1209	56.2	52.3	38.5	63.3	54.8	2244	812	199	96.4
99	100.8	546.5	-3.1	1200	1195	1207	56.3	52.3	38.5	63.4	54.9	2239	816	205.2	94.6
99	100.7	542.3	-2.9	1200	1196	1199	56.3	52.3	38.5	63.4	54.9	2241	825	209.3	96.5
99	100.8	547.4	-2.8	1198	1206	1210	56.3	52.3	38.5	63.4	55	2242	838	218.9	97.7
99	100.8	545.5	-1.9	1189	1186	1193	56.3	52.4	38.5	63.4	55.2	2231	857	231.7	102.8
99	100.5	539	-1.4	1189	1200	1186	56.4	52.4	38.5	63.5	55.2	2237	885	258.5	117.6
99	100.7	544.4	-1.1	1185	1190	1186	56.4	52.5	38.5	63.5	55.3	2239	920	265.9	124.9
99	100.6	545.3	-0.3	1183	1193	1183	56.4	52.5	38.5	63.5	55.3	2239	961	257.6	124.5
99	100.7	541.2	0	1194	1182	1191	56.4	52.5	38.5	63.7	55.3	2237	1006	274.1	122.2
99	100.8	541.4	0.3	1191	1180	1186	56.4	52.5	38.6	63.7	55.3	2237	1053	284.1	128.1
99	100.8	542.2	1.1	1200	1195	1198	56.6	52.6	38.6	63.7	55.5	2230	1096	289	135.7
99	100.7	538.2	1.8	1200	1203	1196	56.6	52.6	38.6	63.8	55.5	2229	1133	298.2	138.4
99	100.7	543.4	3	1200	1200	1188	56.6	52.8	38.6	63.8	55.6	2230	1173	317.9	141.6
99	100.8	536.9	3.4	1200	1196	1201	56.7	52.8	38.6	63.8	55.6	2226	1212	324.7	151.1
99	100.8	538.9	3.9	1200	1197	1196	56.7	52.8	38.6	63.9	55.7	2227	1249	317.2	147.4
99	100.7	539	4.8	1200	1195	1192	56.7	52.9	38.6	63.9	55.8	2228	1285	318.3	148.2
99	100.8	536	5.9	1200	1191	1187	56.8	53	38.7	64.1	55.9	2226	1319	318.5	148.4
99	100.7	539.6	6.1	1200	1200	1204	56.8	53	38.7	64.1	56	2227	1343	323.4	150
99	100.7	539.8	5.8	1200	1200	1193	56.8	53	38.7	64.1	55.9	2222	1360	330.7	153.4
99	100.7	537.5	7.3	1200	1191	1196	56.9	53.1	38.7	64.1	56	2224	1374	336.7	156.4
99	100.8	535.4	7.1	1200	1200	1203	56.9	53.1	38.7	64.2	56.1	2222	1388	345	158.8
99	100.8	535.7	9.2	1200	1202	1202	57	53.2	38.7	64.4	56.2	2219	1404	348.5	163.1
99	100.7	539.2	9.6	1200	1211	1203	57.1	53.3	38.7	64.4	56.2	2221	1426	364.9	162.5
99	100.7	533.7	10.6	1200	1208	1196	57.1	53.5	38.8	64.5	56.3	2219	1454	372	172.6
99	100.7	533.4	10.5	1200	1188	1195	57.2	53.5	38.8	64.5	56.4	2219	1484	366.6	169.5
99	100.8	533	11	1200	1191	1184	57.2	53.6	38.8	64.5	56.4	2213	1515	378	175.3
99	100.6	530.7	13.9	1200	1186	1197	57.4	53.6	38.8	64.6	56.6	2212	1547	384.8	177.3
99	100.8	535.5	15.3	1200	1187	1181	57.4	53.6	38.8	64.6	56.6	2216	1575	386.3	179.8
99	100.8	534.8	15.2	1200	1186	1200	57.5	53.6	38.9	64.7	56.7	2213	1599	384.8	179.8
99	100.8	534	15.5	1200	1202	1196	57.5	53.6	38.9	64.8	56.7	2215	1621	384.2	179.4
99	100.8	534.4	15.6	1200	1207	1203	57.6	53.7	38.9	64.8	56.8	2214	1638	383.5	188
99	100.7	535	16.5	1200	1201	1201	57.7	53.7	38.9	64.9	56.9	2214	1647	382	179.5
99	100.7	533.8	15.6	1200	1188	1190	57.7	53.7	39	64.9	57	2213	1653	381.6	178.5
99	100.7	534.3	16.6	1200	1185	1186	57.8	53.7	39	65.1	57.2	2214	1655	384.8	179.5
99	100.6	534.7	17.1	1200	1187	1193	57.9	53.7	39	65.1	57.2	2215	1655	386.7	180.9
99	100.7	531.7	16.5	1200	1194	1194	57.9	53.7	39	65.2	57.4	2212	1655	387.4	180.8
99	100.7	532.2	16.3	1200	1186	1185	58	53.7	39	65.2	57.4	2216	1656	383.9	181.7
99	100.7	531.1	16	1200	1200	1182	58.2	53.7	39	65.4	57.5	2214	1657	380.1	178.2
99	100.6	532.1	18.6	1200	1191	1200	58.3	53.7	39.1	65.4	57.7	2217	1657	379.9	179.4
99	100.7	532.4	17.4	1200	1194	1189	58.3	53.7	39.1	65.5	57.8	2216	1655	376.2	177.8
99	100.7	528.5	17.6	1200	1210	1204	58.4	53.7	39.2	65.6	57.8	2216	1651	376.7	175.6
99	100.7	527.2	17.7	1200	1207	1203	58.5	53.7	39.2	65.6	57.8	2213	1645	379.7	177.2
99	100.8	533.4	17.3	1200	1189	1206	58.5	53.8	39.2	65.6	57.8	2214	1643	398.5	176.9
99	100.8	531.2	17.7	1200	1186	1205	58.6	53.8	39.3	65.8	57.9	2215	1645	397.6	187
99	100.7	530.1	17.9	1200	1192	1187	58.6	53.8	39.3	65.8	58	2217	1648	374.1	176.6
99	100.9	530.3	18.5	1200	1199	1201	58.9	53.9	39.4	65.9	58.1	2217	1652	372.6	175.4
99	100.7	535	18.5	1200	1199	1196	59	53.9	39.4	66	58.3	2213	1657	376.2	176.2
99	100.6	528.7	18.7	1197	1190	1185	59	53.9	39.5	66.1	58.3	2217	1654	373.9	174.9
99	100.6	529.7	18	1200	1195	1185	59.1	53.9	39.5	66.1	58.3	2212	1643	370.5	181.8
99	100.7	530.7	18.4	1196	1187	1181	59.1	54	39.5	66.2	58.4	2213	1635	385.6	174.2
99	100.7	530.3	18.6	1187	1176	1181	59.2	54	39.6	66.3	58.6	2214	1630	386.7	180
99	100.5	530	19	1176	1177	1179	59.3	54.1	39.6	66.4	58.6	2218	1626	374.1	174.3
99	100.7	526.1	18.6	1176	1167	1153	59.3	54.1	39.7	66.5	58.7	2215	1625	370.5	175.7
99	100.8	530	18	1156	1161	1162	59.4	54	39.7	66.5	58.8	2219	1627	367.1	171.4
99	100.7	530.2	18.7	1165	1149	1144	59.4	54.1	39.7	66.6	58.9	2216	1628	385	172.5

99	100.7	527.6	18.8	1162	1145	1147	59.5	54.1	39.8	66.7	59	2215	1626	385.9	180.3
99	100.6	528.7	19.3	1147	1140	1139	59.5	54.1	39.9	66.8	59	2219	1623	368.5	173.3
99	100.7	529.3	18.8	1145	1139	1148	59.5	54.1	39.9	66.9	59.1	2218	1620	366.8	171.7
99	100.6	528.8	18.5	1135	1148	1144	59.7	54.1	39.9	66.9	59.2	2218	1618	368.3	171.2
99	100.6	530.3	18.3	1138	1121	1126	59.7	54.1	40	66.9	59.3	2217	1614	368.1	172.7
99	100.6	530.3	18.5	1112	1121	1121	59.8	54.3	40	67	59.3	2214	1606	367.3	171.6
99	100.8	527.4	17.7	1102	1122	1120	59.8	54.3	40.1	67.2	59.5	2216	1600	369.2	173.4
99	100.6	527.6	19.4	1120	1105	1098	59.9	54.3	40.2	67.2	59.6	2215	1595	370.7	172.3
99	100.8	527.6	18.6	1121	1108	1105	60	54.3	40.2	67.3	59.7	2214	1593	385.9	173.4
99	100.6	523.5	18.3	1114	1117	1116	60	54.3	40.3	67.3	59.8	2215	1598	391.9	190.2
99	100.7	528.4	19.2	1107	1124	1109	60	54.3	40.3	67.5	59.9	2211	1609	382.9	178.8
99	100.6	527.1	19.4	1101	1098	1121	60.1	54.4	40.4	67.5	60	2215	1622	383.3	179.1
99	100.7	527.9	19.7	1100	1100	1114	60.3	54.4	40.4	67.5	60	2215	1637	385	179.4
99	100.7	526	19.8	1100	1101	1106	60.3	54.4	40.5	67.6	60.1	2211	1649	384.2	180.7
99	100.7	524.8	19.9	1100	1107	1106	60.3	54.4	40.6	67.8	60.3	2211	1654	380.9	178.8
99	100.8	521.5	19.7	1100	1105	1097	60.3	54.4	40.6	67.8	60.4	2215	1657	382.7	178
99	100.8	524.2	19.8	1100	1093	1093	60.3	54.4	40.7	67.9	60.5	2215	1658	386.5	180.1
99	100.7	527.4	19.6	1100	1086	1089	60.3	54.4	40.7	67.9	60.6	2213	1657	388.2	180.7
99	100.7	524.3	19.2	1100	1099	1094	60.3	54.4	40.8	68	60.7	2216	1656	386.3	188.5
99	100.7	525.8	20	1100	1098	1092	60.4	54.4	40.8	68	60.8	2212	1658	383.7	179.4
99	100.8	521.3	20.1	1100	1100	1110	60.4	54.4	40.9	68.2	60.8	2215	1660	384.8	180
99	100.6	520.6	20.3	1100	1091	1106	60.6	54.5	41	68.3	60.9	2215	1662	386.1	179.3
99	100.7	525.8	20.3	1100	1106	1100	60.6	54.5	41	68.3	61	2217	1662	384.4	180.6
99	100.6	525.4	19.5	1100	1101	1088	60.7	54.5	41.1	68.4	61	2215	1662	385.2	179.3
99	100.6	517.7	20.2	1100	1105	1085	60.7	54.6	41.1	68.5	61.1	2211	1664	400.8	181.7
99	100.7	522.4	22	1100	1096	1113	60.9	54.6	41.2	68.5	61.3	2204	1672	418.8	194.5
99	100.7	522.2	24.8	1100	1086	1095	60.9	54.6	41.3	68.6	61.4	2202	1688	426.3	198.6
99	100.7	519.4	26.6	1100	1089	1101	60.9	54.7	41.4	68.7	61.4	2207	1714	439.1	200.3
99	100.8	514.7	28.2	1100	1085	1092	60.9	54.7	41.4	68.7	61.6	2197	1755	465	213.1
99	100.8	516.7	32.4	1100	1076	1089	61	54.7	41.5	68.8	61.7	2193	1808	487.4	215.2
99	100.7	516.4	38.5	1100	1079	1086	61.1	54.8	41.6	68.9	61.8	2194	1869	495.1	231.8
99	100.7	519.3	40.6	1100	1078	1089	61.1	54.8	41.7	69	61.8	2189	1935	494.7	232.7
99	100.8	514.4	42.9	1100	1082	1113	61.1	54.8	41.7	69	62	2190	2007	525.5	232
99	100.7	508.7	48.2	1100	1092	1113	61.3	54.8	41.9	69.2	62.1	2178	2077	534.2	254.4
99	100.7	513.2	53.9	1100	1076	1081	61.4	54.8	42	69.3	62.2	2182	2136	512.8	250
99	100.7	508.4	53.4	1100	1081	1082	61.4	54.9	42.1	69.3	62.3	2178	2186	512.4	251.4
99	100.8	509.6	55	1100	1089	1119	61.6	54.9	42.2	69.3	62.3	2182	2230	532.3	250
99	100.7	509	53.5	1100	1096	1081	61.7	55.1	42.3	69.4	62.4	2177	2264	555.4	259.7
99	100.7	506.7	55.5	1100	1093	1094	61.7	55.1	42.3	69.5	62.5	2177	2288	549.6	262
99	100.7	507.8	56.7	1100	1100	1086	61.8	55.2	42.5	69.6	62.7	2174	2317	566.9	251.9
99	100.7	506	59.4	1100	1082	1095	62	55.3	42.7	69.7	62.8	2172	2357	594.1	277.8
99	100.5	509.1	61.1	1100	1094	1087	62.1	55.3	42.8	69.7	62.9	2172	2401	594.3	280.6
99	100.8	510.7	61.1	1100	1104	1105	62.1	55.3	42.9	69.9	63	2175	2446	590.5	275.8
99	100.8	505.3	62.2	1100	1073	1086	62.3	55.4	43	69.9	63	2172	2490	563.1	275.6
99	100.6	506.3	61.2	1100	1090	1091	62.4	55.4	43.2	70	63.2	2168	2526	572.5	265.1
99	100.7	507.1	62.2	1100	1085	1119	62.5	55.4	43.4	70.2	63.4	2165	2547	589.2	280.7
99	100.6	505	65.7	1100	1091	1107	62.5	55.5	43.5	70.4	63.5	2167	2558	597.3	279.6
99	100.6	505.3	66.9	1100	1085	1121	62.7	55.5	43.6	70.4	63.7	2162	2564	600.7	293.8
99	100.5	502.5	69.5	1100	1081	1094	62.8	55.5	43.8	70.5	63.8	2166	2576	605.2	281.8
99	100.7	504.5	68.1	1100	1125	1086	62.8	55.6	43.9	70.5	63.8	2159	2600	648.2	300.5
99	100.7	504.8	72.4	1100	1081	1088	63	55.6	44.1	70.7	63.9	2155	2639	665.7	309.4
99	100.7	501.3	75.4	1100	1121	1117	63.2	55.6	44.4	70.9	64.1	2146	2690	668.7	311.5
99	100.5	501.4	74.3	1100	1110	1093	63.2	55.8	44.5	70.9	64.2	2145	2752	665.5	299.8
99	100.5	502.5	74.9	1100	1080	1077	63.4	55.8	44.7	71.1	64.4	2143	2821	682.4	312.6
99	100.6	502.9	77.7	1100	1117	1094	63.5	55.9	44.8	71.1	64.5	2121	2886	690.7	317.1
99	100.5	499.4	82.4	1100	1106	1079	63.7	55.9	45	71.3	64.6	2118	2941	677.5	330.1
99	100.6	500.8	85.8	1100	1121	1124	63.8	56	45.2	71.3	64.6	2109	2987	686.2	333.4
99	100.7	498.1	84.7	1100	1079	1084	63.9	56	45.6	71.4	64.9	2107	3029	694.3	324
99	100.8	499.8	89.3	1100	1093	1094	63.9	56.2	45.8	71.6	64.9	2109	3065	724.5	321
99	100.6	499	87.9	1100	1072	1067	64.1	56.3	46	71.6	65.1	2107	3097	718.9	351.8
99	100.6	499.3	87.3	1100	1121	1121	64.2	56.3	46.1	71.8	65.2	2101	3133	713.4	322.1
99	100.6	499.6	89.3	1100	1066	1069	64.4	56.4	46.3	71.8	65.2	2106	3175	753.3	351.7

99	100.7	498.8	89.9	1100	1084	1127	64.5	56.6	46.7	72	65.6	2108	3216	725.3	337.5
99	100.7	498.2	90	1100	1098	1111	64.6	56.6	47	72.1	65.6	2103	3243	695.2	337
99	100.8	496.2	89.7	1100	1122	1113	64.8	56.7	47.1	72.2	65.8	2094	3263	709.7	331.8
99	100.7	495.8	94.1	1100	1087	1072	64.8	56.8	47.3	72.3	65.9	2083	3284	759.1	354.1
99	100.8	493.9	96.1	1100	1080	1099	65.1	56.8	47.6	72.5	66.1	2086	3298	769.4	372.6
99	100.6	492.7	96	1100	1105	1071	65.2	56.9	47.8	72.5	66.1	2080	3310	740.5	360.1
99	100.7	492.5	97.6	1100	1107	1079	65.5	57	48.3	72.8	66.3	2074	3348	796.3	347.3
99	100.5	487.9	101	1100	1085	1116	65.6	57	48.5	72.9	66.5	2052	3411	825.8	383.6
99	100.5	485.3	104.4	1100	1080	1106	65.8	57	48.8	73	66.6	2040	3482	841	374.8
99	100.6	483.3	105.5	1100	1129	1067	65.8	57	49	73	66.7	2035	3567	842.7	395.6
99	100.6	481.4	106.8	1100	1072	1129	65.9	57.1	49.3	73.2	66.8	2044	3663	810.6	376.2
99	100.6	479.4	108.7	1100	1068	1069	66.2	57.1	49.8	73.4	67	2025	3753	855.1	378.4
99	100.7	480.7	106.9	1100	1135	1072	66.3	57.2	50	73.5	67.1	2029	3825	851.1	399.4
99	100.7	480.4	107.6	1100	1075	1075	66.5	57.2	50.2	73.6	67.3	2039	3875	817.3	379
99	100.7	477.2	107.3	1100	1076	1136	66.6	57.2	50.5	73.8	67.4	2030	3911	862.6	384.7
99	100.5	478.5	109.2	1100	1135	1072	66.8	57.2	50.8	73.8	67.6	2027	3941	854.5	401.6
99	100.5	478.3	109	1100	1067	1070	66.9	57.4	51.1	73.9	67.7	2041	3948	810.6	378.1
99	100.5	478.8	110.9	1100	1111	1129	67.2	57.4	51.7	74.1	68	2034	3939	810.8	379.8
99	100.5	478.3	109.8	1100	1133	1077	67.3	57.4	51.9	74.2	68.1	2035	3936	815.3	378.3
99	100.5	476	109.8	1100	1082	1067	67.5	57.5	52.2	74.3	68.2	2024	3920	818.3	384.4
99	100.6	476.8	109.6	1100	1067	1107	67.6	57.5	52.5	74.5	68.3	2035	3886	815.3	382.6
99	100.5	475.8	113.2	1100	1091	1109	67.8	57.5	53	74.6	68.5	2033	3864	816	384.6
99	100.5	476.9	111.5	1100	1134	1066	67.9	57.6	53.3	74.8	68.6	2034	3859	859.6	381.3
99	100.7	473.5	109.9	1100	1124	1076	68	57.7	53.6	74.8	68.7	2020	3863	859.8	403.4
99	100.7	473.4	111.2	1100	1073	1069	68.2	57.7	53.8	75	68.9	2036	3875	821.8	381.9
99	100.6	472.8	110.2	1100	1071	1112	68.2	57.7	54.1	75	69	2021	3898	822.8	386.4
99	100.6	472	111	1100	1117	1130	68.5	57.7	54.6	75.4	69.4	2032	3919	819.2	383
99	100.8	471.9	110.5	1100	1119	1096	68.5	57.8	54.8	75.4	69.4	2027	3932	860.2	384.3
99	100.6	473.4	110.9	1100	1140	1068	68.6	57.9	55.1	75.5	69.6	2029	3943	901.7	401.6
99	100.8	469.5	111.3	1100	1122	1068	68.7	57.9	55.3	75.5	69.7	2021	3962	859.8	403.2
99	100.6	471.6	112.7	1100	1063	1062	68.9	58	55.5	75.7	69.9	2029	3978	817.1	382.4
99	100.5	470.5	111.8	1100	1071	1108	69	58	56	75.9	70.3	2034	3992	821.5	383.9
99	100.5	469	113.7	1100	1068	1136	69.2	58.2	56.2	76.1	70.4	2027	3999	823.3	385.8
99	100.6	470.4	111.1	1100	1135	1064	69.3	58.3	56.4	76.2	70.5	2019	3984	817.5	383.9
99	100.5	469.1	113.8	1100	1092	1065	69.4	58.3	56.6	76.3	70.7	2028	3953	820.7	382.2
99	100.5	468.9	114.5	1100	1074	1079	69.6	58.3	56.8	76.4	70.9	2026	3924	822.2	386.2
99	100.5	468.6	110.5	1100	1065	1130	69.7	58.4	57.1	76.4	71	2034	3896	822.6	381.8
99	100.5	467.7	115.1	1100	1114	1124	69.9	58.4	57.3	76.6	71.1	2028	3874	822.2	404.2
99	100.5	467.5	114.5	1100	1072	1073	70.2	58.4	57.7	76.8	71.4	2035	3867	822.8	383.2
99	100.5	466.4	111.2	1100	1071	1071	70.4	58.4	57.9	77	71.6	2017	3870	826	387.7
99	100.4	466.2	114.8	1100	1073	1072	70.4	58.4	58.1	77	71.8	2026	3872	823.9	384.5
99	100.7	465.7	115.5	1100	1068	1127	70.5	58.5	58.3	77.1	71.9	2029	3874	822	383.7
99	100.5	464.6	113.6	1100	1117	1129	70.7	58.5	58.5	77.2	72.1	2030	3883	860.7	404.5
99	100.4	465.5	112.9	1100	1124	1074	70.7	58.6	58.7	77.3	72.3	2032	3899	857.9	401.9
99	100.6	463.6	113.6	1100	1067	1068	70.9	58.6	59.1	77.5	72.7	2027	3912	820.7	382.6
99	100.5	464.6	114.6	1100	1075	1068	71.1	58.7	59.2	77.7	72.8	2027	3924	821.1	386.2
99	100.7	464.4	113.8	1100	1068	1099	71.3	58.7	59.4	77.7	72.8	2027	3935	821.3	382.1
99	100.6	462.1	115.3	1100	1072	1138	71.3	58.7	59.5	77.9	73	2021	3933	820	385.8
99	100.6	463.8	114.6	1100	1131	1067	71.4	58.9	59.7	78	73.2	2027	3916	816	382.1
99	100.6	461.5	115.8	1100	1114	1079	71.4	58.9	59.9	78	73.2	2027	3901	820.3	382.4
99	100.5	462.3	116.9	1100	1069	1070	71.8	59	60.2	78.3	73.5	2029	3887	826.7	385.5
99	100.5	461.9	117.3	1100	1065	1098	71.8	59	60.4	78.4	73.7	2025	3875	825.2	388.2
99	100.6	462.5	115.3	1100	1065	1114	72	59	60.6	78.6	73.9	2024	3869	824.3	383.2
99	100.6	458.8	114.9	1100	1072	1136	72.1	59	60.8	78.6	74.1	2027	3873	822.8	386.4
99	100.5	461.2	115.3	1100	1113	1101	72.1	59.1	61	78.8	74.1	2033	3882	851.7	402.3
99	100.6	461.3	113.3	1100	1088	1067	72.3	59.1	61.1	78.9	74.3	2031	3894	850	398.3
99	100.7	458.9	113.3	1100	1070	1069	72.5	59.1	61.5	79.1	74.5	2024	3905	818.3	382.8
99	100.6	457.1	115.1	1100	1066	1133	72.7	59.2	61.7	79.3	74.5	2020	3914	822	385.8
99	100.4	457.6	112	1100	1110	1122	72.7	59.3	61.8	79.3	74.6	2026	3922	820.7	403.9
99	100.6	457.7	112.4	1100	1134	1088	72.9	59.3	61.9	79.5	74.8	2029	3927	861.3	383.4
99	100.5	458.6	114.4	1100	1132	1076	73	59.4	62.1	79.5	74.8	2023	3931	863.7	403.7
99	100.6	456.5	116.6	1100	1067	1067	73.2	59.4	62.4	79.6	74.8	2019	3937	822.6	383.7

99	100.5	460	116.3	1100	1067	1064	73.2	59.4	62.5	79.8	74.7	2026	3942	818.8	384.9
99	100.7	459.2	118.3	1100	1073	1138	73.4	59.5	62.6	80	74.5	2031	3946	817.5	382.9
99	100.7	458.8	114.8	1100	1132	1072	73.4	59.7	62.8	80	74.3	2030	3940	816.8	383.3
99	100.8	459.9	116.3	1100	1120	1077	73.6	59.5	62.9	80	73.9	2030	3923	821.1	381.1
99	100.6	457.4	117.1	1100	1069	1066	73.6	59.7	63.1	80.2	73.7	2030	3903	826	385.1
99	100.5	460.1	116.2	1100	1068	1106	73.9	59.8	63.4	80.5	73.2	2020	3886	822	384.7
99	100.6	460	114.3	1100	1134	1133	74.1	59.8	63.6	80.5	72.9	2027	3838	643.9	383.5
99	100.6	450.7	1.4	1200	1178	1186	74.1	59.9	63.8	80.7	72.7	1827	3741	464.5	216.4
99	100.6	416.1	-6.7	1200	1184	1186	74.1	59.9	63.5	80.8	72.6	1685	3602	458.3	202.4
99	100.5	413.7	-7.5	1200	1195	1198	74.3	60	63.3	80.9	72.5	1697	3405	432.3	214.4
99	100.6	417.7	-7.9	1200	1190	1160	74.3	60	63.1	80.9	72.4	1725	3147	429.3	201.5
99	100.5	431.3	-8.1	1057	1067	1043	74.3	60	62.9	81.2	72.3	1811	2903	485.7	224.5
99	100.5	476.3	-9.1	1100	1091	1085	74.3	60	62.5	81.3	72.5	2096	2707	490	241.9
99	100.8	496.6	-10.1	1094	1108	1078	74.5	60.1	62.3	81.4	72.5	2228	2522	376.7	193.7
99	100.7	502.3	-10.6	946	964	940	74.3	60.1	62.2	81.4	72.6	2240	2345	271.9	134.3
99	100.7	502.3	-10.9	900	884	898	74.3	60.1	61.9	81.4	72.7	2244	2173	238.1	115.9
99	100.6	500.7	-11.2	900	898	898	74.3	60	61.6	81.6	72.9	2247	1955	213.6	103.7
99	100.6	496.4	-11.4	900	897	898	74.1	60	61.3	81.6	72.9	2253	1687	193.7	94.1
99	100.6	496.1	-11.1	900	902	896	73.9	60	60.6	81.8	73.2	2259	1423	169.7	86.4
99	100.5	499.1	-11.2	900	891	894	73.8	59.9	60.2	81.9	73.3	2262	1187	138.5	71.9
99	100.4	494.7	-11.1	900	899	901	73.6	59.9	59.9	81.9	73.4	2246	989	137.5	62.7
99	100.4	495	-11	900	903	907	73.4	59.9	59.6	81.9	73.5	2236	845	147.9	72.6
99	100.4	493.6	-11.4	900	895	896	73.4	59.9	59.3	82	73.6	2239	750	143.9	65.3
99	100.5	496.1	-11.8	900	901	898	73	59.9	58.8	82.1	73.8	2252	686	144.5	67.9
99	100.5	496.1	-11.8	900	901	900	73	59.8	58.8	82.1	73.8	2252	650	162	67.5
53	100.5	609.6	-4.2	1200	1198	1191	51	48.3	42.3	64	42.2	2246	637	176.1	81.4
53	100.6	607.8	-4.4	1200	1192	1201	51.2	48.3	42.1	64	42.3	2237	641	183	80.5
53	100.8	606.4	-3.9	1200	1199	1197	51.2	48.4	42	64	42.5	2239	661	187.1	86.7
53	100.6	604.9	-3.7	1200	1190	1199	51.2	48.4	41.9	63.9	42.6	2249	691	178.1	87.3
53	100.5	604	-4	1200	1195	1200	51.2	48.5	41.8	63.8	42.7	2238	719	176.8	86.8
53	100.7	603.1	-3.8	1200	1203	1196	51.2	48.4	41.7	63.8	42.8	2246	741	176.8	82.4
53	100.7	602	-3.9	1200	1195	1205	51.3	48.5	41.6	63.8	43.1	2244	757	186	81.7
53	100.8	600.9	-4	1200	1195	1198	51.3	48.5	41.5	63.8	43.2	2235	768	195.8	87
53	100.7	600.8	-3.7	1200	1197	1193	51.4	48.6	41.4	63.8	43.3	2242	776	196.3	86.9
53	100.5	599.3	-3.8	1200	1199	1192	51.4	48.6	41.4	63.7	43.5	2243	782	186.2	88.7
53	100.7	598.9	-3.5	1200	1194	1205	51.4	48.6	41.3	63.7	43.6	2236	790	179.4	81.3
53	100.6	597.9	-3.9	1200	1198	1198	51.5	48.6	41.3	63.7	43.7	2245	796	180.4	87.7
53	100.6	595.4	-3.3	1200	1203	1198	51.5	48.6	41.2	63.6	43.8	2236	797	178.1	86.4
53	100.8	596	-3.2	1200	1196	1206	51.6	48.7	41.2	63.5	44	2243	792	179.6	85.2
53	100.9	594.9	-3.5	1200	1197	1199	51.6	48.7	41.1	63.5	44.2	2249	784	176.6	83.2
53	100.6	594.3	-3.4	1200	1194	1207	51.7	48.7	41.1	63.5	44.3	2250	776	182.1	81.5
53	100.7	594.4	-3.3	1200	1193	1198	51.7	48.8	41	63.5	44.4	2236	770	189.4	85.7
53	100.6	592.8	-3.6	1200	1200	1199	51.8	48.8	41	63.5	44.5	2235	768	183.2	84.2
53	100.6	594.3	-3.5	1200	1197	1202	51.8	48.8	41	63.5	44.6	2239	770	179.1	87.5
53	100.5	592.6	-3.1	1200	1198	1200	51.8	48.9	40.9	63.4	44.7	2246	775	194.3	83.3
53	100.7	590.4	-3.1	1200	1210	1199	52	48.9	40.9	63.4	44.9	2239	785	204.8	92.6
53	100.6	589.5	-3.1	1200	1199	1199	52.1	48.9	40.9	63.4	45	2234	795	201.6	93.6
53	100.6	589.6	-2.6	1200	1195	1199	52.1	49	40.9	63.4	45.1	2235	808	206.1	94.6
53	100.8	587.6	-2	1200	1197	1199	52.2	49	40.9	63.4	45.2	2232	829	224.7	99.4
53	100.6	586.3	-1.3	1200	1201	1202	52.2	49	40.8	63.4	45.3	2242	855	242	117
53	100.8	585.3	0.1	1200	1208	1196	52.3	49	40.8	63.4	45.4	2238	885	246.3	120.2
53	100.8	587	0.3	1200	1200	1194	52.3	49	40.8	63.4	45.5	2232	920	252.7	115.3
53	100.6	577.3	0.9	1200	1206	1194	52.4	49.1	40.8	63.4	45.7	2206	962	260.2	119.3
53	100.8	574.8	1.5	1200	1189	1192	52.4	49.1	40.8	63.3	45.8	2191	1010	298.6	122.5
60	100.5	577.8	2.9	1200	1199	1198	52.5	49.1	40.8	63.3	46	2229	1062	318.9	151.2
64	100.7	580.3	2.9	1200	1210	1193	52.5	49.2	40.8	63.3	46	2233	1116	292.2	136.2
69	100.7	572.8	3.3	1200	1196	1197	52.6	49.2	40.8	63.3	46.2	2232	1168	295.4	131.9
72	100.6	577.8	3.3	1200	1207	1189	52.6	49.2	40.8	63.3	46.3	2232	1216	298.4	139.2
79	100.6	574.3	3.5	1200	1200	1195	52.8	49.2	40.8	63.3	46.4	2232	1248	286	133.8
87	100.8	570.8	3.8	1200	1198	1195	52.9	49.3	40.8	63.2	46.6	2235	1263	300.4	133.8
96	100.8	569.9	4	1200	1194	1196	53	49.3	40.8	63.2	46.7	2232	1272	302.9	141.6
97	100.6	572.4	3.8	1200	1196	1197	53	49.4	40.8	63.2	46.8	2230	1272	289.2	135.6

100	100.6	568.3	4.7	1200	1191	1195	53.1	49.4	40.7	63.2	46.9	2232	1265	289.5	141.6
100	100.7	570.5	4.6	1200	1199	1203	53.1	49.5	40.7	63.2	46.9	2235	1262	291.4	135.7
100	100.7	568.3	4.2	1200	1192	1189	53.2	49.5	40.7	63.2	47	2231	1260	289.2	136.2
100	100.7	568.3	4.7	1200	1195	1203	53.2	49.5	40.7	63.2	47.2	2233	1254	288.2	141.2
100	100.5	566.4	5.2	1200	1192	1201	53.3	49.6	40.7	63.2	47.3	2234	1251	303.8	142.5
100	100.7	566.7	5.1	1200	1198	1195	53.5	49.6	40.7	63.2	47.5	2232	1252	303.6	142
100	100.7	565.3	4.8	1200	1194	1200	53.5	49.7	40.7	63.2	47.6	2232	1252	289.7	135.5
100	100.5	566.2	4.8	1200	1195	1206	53.5	49.7	40.7	63.2	47.7	2234	1256	301.4	141.4
100	100.6	564.8	4.8	1200	1206	1206	53.6	49.7	40.8	63.2	47.8	2234	1265	301	140.8
100	100.6	565.8	4.6	1200	1207	1193	53.7	49.7	40.8	63.2	48	2231	1269	288.6	135.8
100	100.8	561.5	4.7	1200	1200	1197	53.9	49.8	40.8	63.2	48.2	2232	1269	295.4	135.1
100	100.6	565.9	5.5	1200	1196	1201	53.9	49.9	40.8	63.2	48.3	2228	1274	316.2	144
100	100.8	557.6	6.8	1200	1200	1200	54	49.9	40.8	63.2	48.4	2225	1283	334.1	153.5
100	100.7	559.8	7.6	1200	1207	1198	54	50	40.8	63.2	48.6	2221	1297	343.5	158.8
100	100.6	558.4	8.1	1200	1197	1195	54.1	50	40.8	63.3	48.7	2219	1325	351.2	162.8
100	100.7	556	9.3	1200	1206	1187	54.3	50.1	40.9	63.3	48.8	2218	1365	357.2	165.2
100	100.7	557.3	10.5	1200	1202	1194	54.4	50.1	40.9	63.3	48.9	2220	1409	362.6	167.1
100	100.6	555.5	10.9	1200	1192	1204	54.5	50.2	40.9	63.3	49	2212	1455	380.3	181
100	100.8	555.8	12.4	1200	1204	1190	54.6	50.4	40.9	63.4	49.1	2210	1504	406	187
100	100.6	553.2	14.3	1200	1208	1193	54.6	50.4	41	63.4	49.2	2203	1555	418.8	196.3
100	100.6	550.4	16.5	1200	1202	1186	54.7	50.5	41	63.4	49.3	2204	1608	427.8	198.8
100	100.7	555.7	17.9	1200	1197	1192	54.9	50.5	41	63.4	49.5	2205	1665	435.9	201.9
100	100.7	548.2	18.7	1200	1201	1204	54.9	50.6	41	63.4	49.6	2203	1724	435.5	204
100	100.6	550.2	19.2	1200	1196	1185	55.1	50.6	41.1	63.4	49.7	2207	1776	434	203.9
100	100.4	550	19.4	1200	1191	1192	55.2	50.7	41.1	63.4	49.8	2203	1823	456.9	203.4
100	100.4	553.7	19	1200	1218	1193	55.2	50.7	41.1	63.5	49.9	2201	1866	477.6	217.2
100	100.5	547.6	21.2	1200	1199	1187	55.3	50.8	41.1	63.5	50.1	2196	1905	480.6	221.8
100	100.5	548.4	23.4	1200	1194	1206	55.4	50.9	41.2	63.5	50.2	2195	1942	486.1	227.8
100	100.7	546.7	24.8	1200	1190	1190	55.6	51	41.2	63.6	50.5	2192	1985	490.6	228.4
100	100.5	547.2	26.5	1200	1210	1185	55.6	51.2	41.3	63.7	50.7	2190	2028	494.7	231.9
100	100.6	546.3	25.9	1200	1198	1190	55.8	51.2	41.3	63.7	50.8	2190	2064	480.8	231.9
100	100.6	541	28.7	1200	1211	1189	55.9	51.3	41.3	63.7	50.9	2181	2095	495.8	231.4
100	100.6	547.3	32.3	1200	1208	1181	56	51.4	41.4	63.7	51	2188	2124	525.9	245.6
100	100.6	544.5	33.1	1200	1180	1204	56.1	51.5	41.5	63.8	51.2	2182	2149	527.6	245.8
100	100.6	542.1	35.7	1200	1213	1178	56.2	51.6	41.5	63.8	51.3	2182	2173	527	248.1
100	100.6	542.5	37.7	1200	1193	1185	56.3	51.6	41.6	63.9	51.5	2183	2205	527	245.8
100	100.6	535.1	37.2	1200	1213	1193	56.4	51.7	41.7	63.9	51.6	2182	2239	527.8	248.1
100	100.7	543.9	39.4	1200	1197	1183	56.6	51.8	41.7	63.9	51.8	2182	2267	530.4	246
100	100.8	533.9	38.3	1200	1179	1207	56.7	51.8	41.8	63.9	51.9	2185	2288	532.1	249.3
100	100.8	540.7	38.9	1200	1204	1199	56.8	52	41.8	64.1	52.1	2182	2304	528.7	247.3
100	100.6	540.3	39.8	1200	1213	1193	56.9	52.1	41.9	64.1	52.3	2180	2312	528.9	247
100	100.6	538	40.6	1200	1187	1176	57	52.1	41.9	64.1	52.5	2181	2316	534.4	250.1
100	100.7	539.6	42.9	1200	1185	1186	57.1	52.2	42.1	64.2	52.7	2182	2320	537.7	251.8
100	100.5	533.3	42.4	1200	1194	1175	57.2	52.2	42.2	64.3	52.8	2182	2324	534.7	252.9
100	100.6	541.4	44	1200	1191	1183	57.4	52.3	42.3	64.4	52.9	2183	2328	536.4	251.3
100	100.8	538.4	44	1200	1210	1194	57.5	52.4	42.3	64.4	53.1	2179	2335	541.7	252.7
100	100.6	535.3	44.3	1200	1190	1216	57.7	52.4	42.5	64.4	53.2	2183	2338	521.4	253.6
100	100.6	537.8	46.2	1200	1184	1192	57.8	52.5	42.6	64.5	53.5	2182	2338	523.5	246
100	100.6	538.3	47	1200	1174	1178	58	52.5	42.7	64.6	53.6	2182	2339	547.5	256.3
100	100.8	532.3	48.3	1161	1170	1182	58.2	52.6	42.8	64.6	53.8	2177	2342	554.8	257.2
100	100.6	531.1	49.3	1100	1114	1111	58.3	52.6	42.9	64.6	54	2172	2349	577.2	264.3
100	100.5	530.4	55.4	1100	1078	1101	58.3	52.8	43	64.8	54.1	2171	2368	567.1	277.2
100	100.7	532.6	58.9	1100	1081	1075	58.4	52.8	43.2	64.8	54.3	2177	2397	568.4	263
100	100.6	528.8	59.2	1100	1091	1083	58.4	52.8	43.3	64.9	54.4	2176	2431	592.8	277.1
100	100.8	533	61.5	1100	1086	1099	58.6	52.9	43.5	65	54.8	2169	2470	595.4	276.2
100	100.8	529	61.3	1100	1094	1087	58.7	52.8	43.7	65.1	54.9	2166	2509	609.5	267
100	100.6	531.6	63.3	1100	1111	1124	58.7	52.9	43.8	65.1	55.1	2165	2549	623.4	289.1
100	100.7	527.7	65.4	1100	1117	1091	58.9	52.8	43.9	65.2	55.3	2161	2592	626.8	295.2
100	100.8	525.1	67.1	1100	1086	1082	59	52.9	44.1	65.3	55.4	2168	2627	594.9	291
100	100.6	526.3	66.7	1100	1075	1081	59.1	52.9	44.2	65.4	55.5	2162	2656	595.4	280.2
100	100.8	526.1	67.6	1100	1093	1097	59.2	52.9	44.3	65.4	55.6	2166	2681	627	291.8
100	100.7	522.7	67.6	1100	1071	1116	59.3	52.9	44.6	65.6	55.9	2168	2697	629.2	293

100	100.5	525.7	67.3	1100	1114	1125	59.4	53	44.8	65.6	56	2165	2705	642.2	294
100	100.6	523.9	69.5	1100	1096	1076	59.5	53	44.9	65.8	56.1	2155	2718	628.1	307.4
100	100.4	521.2	73.4	1100	1118	1109	59.7	53	45.1	65.8	56.3	2145	2739	636.8	297.9
100	100.4	517.3	75.3	1100	1122	1125	59.8	53	45.2	65.8	56.4	2142	2766	670.6	314.3
100	100.6	518.5	76.2	1100	1099	1123	59.9	53	45.4	65.9	56.6	2138	2801	669.5	316.3
100	100.7	514.3	77.1	1100	1086	1121	60	53	45.7	66.1	56.9	2138	2835	642	315.2
100	100.7	515.2	77.6	1100	1096	1116	60.1	53.1	45.9	66.2	57	2137	2866	646.7	303.2
100	100.5	514.6	77.8	1100	1108	1081	60.3	53.1	46	66.2	57.1	2143	2899	686.9	316.9
100	100.6	513.8	78.5	1100	1074	1096	60.4	53.1	46.2	66.3	57.2	2135	2924	677.7	324.3
100	100.7	511.7	81.1	1100	1117	1078	60.6	53.2	46.4	66.3	57.4	2124	2946	693.9	321.2
100	100.6	509.6	84.5	1100	1119	1101	60.7	53.2	46.8	66.5	57.7	2118	2983	730	342.8
100	100.6	510.2	83.9	1100	1131	1075	60.9	53.3	47	66.6	57.8	2115	3027	702	343.7
100	100.5	513.2	84.9	1100	1096	1113	61	53.3	47.2	66.7	58	2120	3066	702.7	312.5
100	100.6	510.5	85.8	1100	1095	1111	61.1	53.5	47.3	66.8	58.2	2107	3118	748.4	346.8
100	100.6	507.7	88.3	1100	1076	1123	61.3	53.5	47.5	66.9	58.3	2101	3174	736.7	342.8
100	100.7	505.5	91.7	1100	1077	1118	61.6	53.6	47.9	67	58.6	2085	3212	712.3	333.1
100	100.8	505.9	94	1100	1093	1080	61.7	53.6	48.2	67	58.8	2082	3249	750.8	353.1
100	100.6	506.7	92.8	1100	1084	1070	62	53.7	48.4	67.2	58.9	2084	3300	788	370.7
100	100.6	508.5	95.5	1100	1081	1093	62.1	53.7	48.6	67.3	59.1	2086	3343	771.7	355.4
100	100.6	503.1	96.3	1100	1066	1104	62.3	53.7	48.8	67.3	59.3	2067	3386	780.5	356.9
100	100.6	503.5	101.1	1100	1072	1061	62.4	53.8	49	67.5	59.3	2030	3455	819.4	379.3
100	100.4	501.9	108.2	1100	1067	1061	62.5	53.8	49.3	67.6	59.5	2023	3532	832.7	390.2
100	100.8	501.6	112.4	1100	1066	1064	62.7	53.8	49.6	67.6	59.7	2015	3604	831.4	389.2
100	100.7	500.4	111.1	1100	1071	1114	63	53.9	50.1	67.8	60	2018	3694	872.9	390
100	100.6	496.1	110.8	1100	1140	1065	63.1	54	50.4	67.9	60.1	2009	3798	883.8	410.9
100	100.5	499.3	111.2	1100	1069	1082	63.2	54	50.7	67.9	60.3	1998	3888	855.3	398.4
100	100.6	493.1	111.8	1100	1065	1078	63.2	54.1	51	68	60.4	1997	3967	867.1	402.8
100	100.7	491.9	112.7	1100	1064	1138	63.4	54.1	51.3	68.2	60.7	1996	4043	875.6	408.6
100	100.5	492.3	113.3	1100	1086	1066	63.5	54.3	51.5	68.2	60.8	1995	4097	876	409.6
100	100.5	491.4	112	1100	1063	1134	63.7	54.1	51.8	68.3	61.1	1984	4129	879.9	411.1
100	100.5	490	112.7	1100	1072	1075	63.9	54.3	52.4	68.5	61.4	1975	4149	837.2	412.8
100	100.7	490.7	114.4	1100	1110	1077	64.2	54.4	52.7	68.6	61.5	1982	4157	836.9	391.1
100	100.5	489.7	116	1100	1126	1073	64.4	54.5	53	68.6	61.8	1974	4159	887.4	413.4
100	100.6	485.7	114.7	1100	1065	1066	64.5	54.5	53.3	68.7	62	1970	4163	889.9	416.9
100	100.6	487.8	115.9	1100	1062	1104	64.6	54.6	53.5	68.9	62.1	1981	4168	887.4	416.1
100	100.5	487.7	117.3	1100	1067	1133	64.8	54.6	53.9	68.9	62.3	1979	4183	887.2	415.6
100	100.8	487.1	115.5	1100	1110	1135	65.1	54.7	54.4	69.2	62.7	1987	4214	896.4	415.3
100	100.8	485.1	115	1100	1082	1130	65.2	54.8	54.7	69.3	62.9	1967	4248	918.8	423.6
100	100.6	480.4	118.7	1100	1114	1123	65.5	54.8	55	69.3	63.1	1953	4287	931.8	435.4
100	100.6	476.3	117.5	1100	1073	1067	65.6	54.8	55.3	69.4	63.2	1953	4335	932.5	413.4
100	100.5	476.4	117.5	1100	1131	1134	65.6	54.9	55.5	69.6	63.4	1936	4387	936.8	438.9
100	100.6	475.5	121.6	1100	1087	1134	65.8	54.9	55.8	69.6	63.5	1926	4442	939.8	439.3
100	100.5	473.1	120.8	1100	1128	1135	66.2	55.1	56.3	69.9	63.9	1928	4490	897.4	438.7
100	100.5	471.5	121.6	1100	1070	1063	66.3	55.2	56.5	69.9	64.1	1910	4527	906.8	400.7
100	100.4	466.7	123.1	1100	1069	1123	66.3	55.2	56.8	70	64.2	1895	4556	916.9	449.5
100	100.4	465.7	120.7	1100	1071	1112	66.5	55.2	57	70.2	64.5	1886	4573	920.3	407.2
100	100.5	464.5	120.7	1093	1079	1067	66.6	55.2	57.3	70.3	64.6	1870	4587	969.9	429.9
100	100.6	461.8	122.1	1089	1067	1121	66.8	55.3	57.5	70.4	64.9	1863	4614	926.5	433.1
100	100.5	458.9	122	1093	1060	1058	66.9	55.4	57.9	70.7	65.2	1866	4640	877.6	410.6
100	100.6	459.2	121.1	1089	1114	1071	67	55.4	58.1	70.7	65.4	1863	4669	926.1	433.6
100	100.5	458.8	121.5	1086	1099	1116	67.2	55.4	58.3	70.9	65.6	1858	4700	927.4	433.9
100	100.6	458.2	124.6	1084	1115	1113	67.3	55.5	58.5	71	65.7	1854	4713	929.3	411.9
100	100.4	456.7	121.2	1082	1072	1113	67.5	55.5	58.7	71.1	65.9	1851	4727	980.4	435
100	100.6	456.2	122.6	1081	1057	1065	67.6	55.6	58.9	71.3	66	1849	4765	981.5	435.5
100	100.4	456.8	125.2	1086	1080	1117	67.9	55.6	59.3	71.4	66.3	1858	4799	931.4	435.6
100	100.5	457.2	123	1086	1055	1103	68	55.6	59.5	71.6	66.5	1858	4820	881.6	411.2
100	100.5	455.6	123.9	1089	1051	1065	68	55.8	59.7	71.8	66.7	1862	4847	928.9	411.2
100	100.6	456.3	123.7	1089	1102	1112	68.2	55.8	59.9	71.8	66.9	1863	4849	926.3	433.4
100	100.8	453.9	123.3	1087	1122	1053	68.3	55.8	60.1	72	67	1859	4805	874.8	409.5
100	100.6	454.9	121.4	1094	1066	1116	68.5	55.9	60.3	72.1	67.2	1871	4745	869	430.3
100	100.6	458.5	124.4	1100	1066	1077	68.6	55.9	60.7	72.3	67.6	1886	4699	915.4	407.1
100	100.6	459.8	123	1100	1070	1133	68.7	56	60.9	72.4	67.7	1900	4657	962.8	451.1

100	100.5	459.3	126.3	1100	1113	1066	68.7	56	61.1	72.5	67.9	1894	4621	910.5	426.7
100	100.4	459.6	123.5	1100	1069	1134	68.9	56.1	61.3	72.7	68.1	1889	4603	867.7	425.8
100	100.7	456.8	123.2	1100	1114	1064	69	56.1	61.4	72.9	68.2	1898	4597	869.9	406
100	100.6	460.1	123.3	1100	1066	1085	69	56.1	61.6	72.9	68.4	1904	4576	865.4	403
100	100.5	456.9	124.1	1100	1071	1124	69.2	56.2	61.9	73.2	68.6	1903	4531	862	426.7
100	100.5	457.7	126.5	1100	1133	1071	69.3	56.2	62.1	73.2	68.6	1902	4490	906.2	403.3
100	100.6	457	123.5	1100	1134	1073	69.4	56.2	62.3	73.4	68.7	1911	4469	953.9	447.5
100	100.5	456.9	126.1	1100	1117	1071	69.6	56.3	62.5	73.6	69	1902	4467	954.9	424.9
100	100.4	457.5	122.9	1100	1112	1072	69.6	56.3	62.7	73.6	69.1	1899	4486	908.1	425.6
100	100.5	456.3	121	1100	1067	1130	69.7	56.4	62.9	73.8	69.3	1902	4535	909.8	402.7
100	100.5	455	125.8	1100	1129	1065	69.9	56.4	63.1	73.9	69.4	1899	4587	913	449.1
100	100.5	452.4	125.3	1100	1069	1072	70.2	56.6	63.5	74.1	69.7	1900	4617	912.6	402.7
100	100.5	451.9	122.7	1100	1090	1095	70.4	56.7	63.8	74.3	69.9	1900	4631	960.5	449.2
100	100.6	452.3	123.8	1100	1125	1074	70.4	56.7	63.9	74.4	70	1903	4637	910	427.6
100	100.4	453.3	124.3	1100	1138	1069	70.5	56.8	64.1	74.5	70.3	1896	4622	857.5	403.5
100	100.6	451	123.6	1100	1114	1072	70.5	56.8	64.3	74.6	70.4	1892	4606	907.5	402.3
100	100.4	450.7	128.8	1100	1083	1138	70.9	56.9	64.6	74.8	70.9	1900	4607	959.4	448.8
100	100.4	450.4	129.3	1100	1130	1063	70.9	57	64.7	74.9	71.1	1909	4595	913	426
100	100.4	451.2	125.3	1100	1079	1071	71.1	57	64.9	75.2	71.3	1908	4583	909.8	403
100	100.6	449.6	125.8	1100	1122	1074	71.1	57	65	75.2	71.3	1904	4591	910.5	424.4
100	100.7	448.2	124.2	1100	1071	1132	71.3	57.1	65.1	75.4	71.4	1903	4599	911.1	404.9
100	100.6	447.9	125.5	1100	1123	1081	71.3	57.1	65.2	75.5	71.6	1900	4597	956.2	447.9
100	100.4	447	122.7	1100	1082	1066	71.6	57.2	65.4	75.7	72.1	1897	4606	907.9	425.7
100	100.4	447.4	125.2	1100	1079	1115	71.8	57.2	65.4	75.8	72.3	1895	4616	908.3	425.1
100	100.6	445.3	128.5	1100	1129	1071	71.8	57.4	65.6	75.8	72.5	1902	4616	912.2	448.9
100	100.5	445.2	125.8	1100	1073	1078	71.8	57.4	65.6	76.1	72.7	1906	4617	910.5	403.7
100	100.6	445.2	125.2	1100	1061	1114	72	57.4	65.7	76.1	72.9	1903	4609	909.6	446.8
100	100.6	445.3	124.9	1100	1074	1128	72.1	57.5	65.8	76.3	73	1888	4592	864.1	427.9
100	100.5	443.6	126.3	1100	1068	1103	72.5	57.5	66	76.4	73.2	1907	4573	908.6	404.8
100	100.5	443.3	126.6	1100	1067	1069	72.7	57.6	66.3	76.6	73.5	1896	4554	908.1	448.1
100	100.5	441.5	125.1	1100	1082	1085	72.7	57.8	66.4	76.6	73.6	1891	4537	911.3	426.8
100	100.6	443	124.9	1100	1122	1070	72.9	57.8	66.5	76.8	73.8	1893	4540	959.6	450.2
100	100.5	442.5	123.4	1100	1117	1062	73	57.8	66.7	76.9	73.9	1913	4556	898.3	448.1
100	100.4	449.9	124.2	1100	1083	1108	73	57.9	66.9	77	74.1	1954	4558	857.9	413.4
100	100.6	456.5	125.1	1100	1070	1133	73.2	57.9	67.1	77.1	74.3	1997	4545	870.9	410.3
100	100.4	457.5	120	1100	1109	1068	73.4	58	67.5	77.3	74.5	1993	4511	862.6	404.6
100	100.5	458.8	120.3	1100	1112	1063	73.6	58.2	67.7	77.5	74.6	2006	4437	860.5	404.1
100	100.6	460.1	122.5	1100	1109	1077	73.8	58.2	67.9	77.6	74.6	1998	4342	859.8	402.7
100	100.6	461.4	119.3	1100	1122	1073	73.8	58.3	68.1	77.7	74.6	1998	4259	859.4	402.7
100	100.6	461.2	120.4	1100	1118	1077	73.9	58.4	68.3	77.9	74.5	1996	4190	861.7	401.6
100	100.5	459.4	121.2	1100	1115	1074	73.9	58.4	68.5	78	74.2	1994	4140	862.4	403.8
100	100.5	460.4	120.6	1100	1120	1072	74.1	58.5	68.6	78	73.9	2010	4120	858.5	401.3
100	100.6	460	121	1100	1135	1069	74.3	58.6	68.9	78.3	73.3	2010	4112	854.9	401.2
100	100.5	459.5	119	1100	1129	1068	74.5	58.7	69	78.4	73.1	2011	4105	852.5	398.7
100	100.5	460.2	117.9	1100	1115	1071	74.5	58.9	69.2	78.4	72.7	1998	4098	854.7	398.9
100	100.5	459.7	118.4	1100	1082	1069	74.6	58.9	69.3	78.6	72.5	1997	4091	856.4	400.4
100	100.5	459.8	119.2	1100	1103	1070	74.8	59	69.3	78.7	72.4	2010	4083	854.7	400.1
100	100.4	457.4	121.2	1100	1081	1066	74.8	59	69.4	78.8	72.3	2005	4077	853.4	399.2
100	100.6	459	119.4	1100	1069	1120	75	59.1	69.4	78.9	72.2	1998	4075	853.8	399.2
100	100.4	457.3	118.9	1100	1068	1120	75.4	59.1	69.6	79.1	72.3	1998	4074	853.6	400.1
100	100.5	457.5	118.3	1100	1077	1116	75.5	59.1	69.6	79.3	72.3	2010	4073	854	399
100	100.5	456.6	121.2	1100	1078	1122	75.7	59.3	69.6	79.3	72.4	1997	4074	855.5	399.6
100	100.4	457.1	120.9	1100	1064	1133	75.7	59.3	69.7	79.5	72.5	1997	4075	854.7	400.8
100	100.5	456.2	119.1	1100	1065	1135	75.9	59.4	69.7	79.6	72.7	1997	4076	855.5	399.5
100	100.5	456.3	118.1	1100	1064	1133	76.1	59.4	69.7	79.6	72.8	2007	4077	855.5	420.7
100	100.4	456.3	119.6	1100	1072	1130	76.3	59.4	69.8	79.8	73	2011	4063	773.6	398.8
100	100.5	479.4	102.9	1100	1121	1093	76.3	59.5	70	80	73.2	2134	4001	645.6	331.8
100	100.6	485.6	90.8	1100	1110	1122	76.4	59.7	70	80	73.4	2145	3886	630	295.5
100	100.5	487	86	1100	1110	1107	76.4	59.7	70	80.2	73.7	2151	3725	661	308.3
100	100.5	484.5	83.6	1100	1077	1076	76.6	59.8	70.2	80.4	73.8	2155	3518	658.6	308.1
100	100.5	485.3	84.8	1100	1087	1108	76.6	59.8	70.2	80.5	73.9	2151	3296	658.9	294.6
100	100.5	485.8	84.5	1100	1104	1110	76.8	59.8	70.4	80.7	74.1	2157	3123	656.7	307.3

100	100.4	484.7	83.3	1100	1123	1116	76.8	59.9	70.5	80.7	74.3	2152	3006	653.9	306.6
100	100.4	484	82.8	1100	1069	1095	77	59.9	70.9	81	74.6	2157	2932	653.5	306.4
100	100.5	484.5	79.8	1100	1127	1101	77	60	71	81.2	74.7	2156	2904	653.3	304.8
100	100.5	482.6	80.8	1100	1098	1080	77	60	71.1	81.4	74.8	2155	2905	653.1	305.7
100	100.4	481.5	81.8	1100	1075	1080	77.1	60	71.3	81.4	75	2158	2905	653.5	304.5
100	100.5	483.1	81	1100	1115	1119	77	60	71.3	81.6	75	2156	2900	651.6	305.6
100	100.5	483.1	80.7	1100	1075	1083	77	60.1	71.3	81.6	75	2154	2896	649.5	303.5
100	100.6	484	81.5	1100	1092	1085	77.1	60.1	71.4	81.9	75	2157	2892	650.9	304
100	100.5	484	81.1	1100	1086	1114	77.1	60.3	71.4	81.9	74.8	2163	2889	649.7	303.8
100	100.4	484	81.3	1100	1113	1105	77.1	60.4	71.5	82.1	74.7	2161	2884	644.7	303.7
100	100.7	483.1	80.1	1100	1095	1117	77.1	60.4	71.6	82.1	74.6	2168	2870	593.2	295.9
100	100.6	487.8	72.7	1100	1105	1101	77.1	60.4	71.6	82.3	74.3	2191	2834	536.2	254.1
100	100.7	490.2	63.1	1100	1114	1092	77.1	60.6	71.6	82.3	74.1	2190	2774	521.8	244.9
100	100.6	488.2	57.6	1100	1089	1096	77.1	60.6	71.3	82.6	73.8	2185	2692	519.3	231.7
100	100.7	489.9	56.9	1100	1105	1083	77.3	60.7	71.2	82.6	73.6	2188	2588	520.1	232.9
100	100.6	490.2	57.2	1100	1106	1095	77.3	60.9	71	82.8	73.4	2184	2481	518.8	243
100	100.5	488.9	55	1100	1094	1091	77.5	60.9	71	82.8	73.3	2183	2352	284.5	243.4
16	100.8	279.6	-3	550	617	595	60.6	55.1	48.2	76.4	67	923	2175	43.8	29.9
20	100.8	318	-3	522	534	522	60.6	55.1	48.2	76.4	66.9	979	1954	61.4	18.4
25	100.8	349.5	-3.2	687	649	711	60.6	54.9	48.2	76.4	66.9	1062	1707	112.4	43.4
37	100.8	406.4	-4.1	792	800	769	60.7	55.1	48	76.4	66.9	1477	1425	165	65.7
42	100.7	452.8	-2.8	800	807	806	60.9	55.2	48	76.4	66.9	1711	1185	215.1	73.3
48	100.8	499.8	-2.1	913	975	861	61	55.3	47.9	76.4	66.9	1937	1055	278.1	105.5
55	100.7	536.8	1	1059	1047	955	61.1	55.4	47.7	76.4	66.9	2175	1006	222.1	134.3
55	101	556.9	0.8	900	895	897	61.4	55.8	47.6	76.4	66.9	2267	992	141.7	61.2
59	100.8	555.4	0.3	900	893	902	61.6	55.8	47.4	76.4	66.8	2262	1009	158.6	64.9
60	100.9	553.3	-0.6	900	898	896	61.8	56	47.2	76.3	66.8	2234	1007	153.7	71.7
67	100.8	550.2	-0.2	900	901	901	62.1	56.1	46.9	76.3	66.8	2246	938	148.6	65.9
72	100.7	548.5	0.1	900	896	898	62.3	56.2	46.7	76.3	66.8	2245	840	146.4	70
77	100.8	546.9	-0.2	900	896	898	62.4	56.3	46.6	76.3	66.8	2237	759	142.4	67
83	100.8	545.3	-0.1	900	896	906	62.4	56.4	46.4	76.3	66.6	2239	692	144.9	69
94	100.9	545.5	-0.3	900	899	909	62.7	56.7	46.3	76.3	66.6	2240	646	146.7	68.1
98	100.7	542.6	0	900	896	898	62.7	56.8	46.2	76.3	66.6	2241	629	150.7	70.4
100	100.8	539.9	0.3	900	901	905	62.8	56.8	46	76.3	66.6	2245	628	155	71.7
100	100.7	540.7	0.3	900	898	896	62.8	56.9	45.9	76.3	66.5	2245	629	156.1	73
99	100.9	538.3	0.6	900	898	901	63	56.9	45.7	76.3	66.5	2236	633	151.1	71.9
99	100.9	537.9	0.2	900	894	895	63	57	45.6	76.3	66.5	2247	640	156.1	69.1
99	100.8	538.8	0.8	900	896	904	63	57	45.5	76.3	66.5	2242	649	166.5	74.2
99	100.9	534.1	0.8	900	899	891	63.1	57.1	45.5	76.3	66.5	2235	657	158	74.2
99	100.8	536.6	0.4	900	899	898	63.1	57.2	45.4	76.3	66.4	2239	661	148.6	69.2
99	100.8	531.8	-0.2	900	897	893	63.2	57.2	45.3	76.3	66.4	2236	664	148.6	70.1
99	100.8	534.7	0.3	900	895	894	63.2	57.2	45.2	76.3	66.4	2241	664	146.4	70.6
99	100.8	529.9	0.8	900	897	896	63.4	57.2	45.1	76.4	66.4	2247	658	145.6	69.2
99	100.8	529.5	0.4	900	896	893	63.4	57.4	45.1	76.3	66.3	2247	650	147.5	68.2
99	100.8	529.2	0.2	900	899	899	63.4	57.4	45	76.4	66.3	2252	642	149.8	69.8
99	100.7	527.7	0.8	900	894	902	63.5	57.5	45	76.4	66.3	2250	636	154.1	74.1
99	100.9	528	0.4	900	900	902	63.5	57.5	44.9	76.3	66.3	2248	634	162.7	73.4
99	100.9	523.7	1.1	900	903	892	63.7	57.6	44.9	76.4	66.2	2243	639	167.6	75.6
99	100.7	524.9	1.1	900	902	908	63.7	57.6	44.8	76.4	66.2	2248	650	164.4	76.5
99	100.7	528.7	1.3	900	892	894	63.7	57.6	44.8	76.4	66.2	2247	662	163.3	76.6
99	100.8	520.7	1.1	900	902	897	63.8	57.7	44.7	76.4	66.2	2242	676	166.3	78.4
99	100.7	523	1.3	900	893	906	63.8	57.7	44.7	76.4	66.2	2243	688	166.3	77.7
99	100.8	524.6	1.3	900	903	901	63.9	57.7	44.7	76.4	66.2	2250	695	165.7	81.8
99	100.8	520.3	1.5	900	894	904	63.9	57.8	44.6	76.4	66.2	2238	702	177	82.2
99	100.8	523.3	1.7	900	891	899	63.9	57.8	44.6	76.4	66.2	2245	711	187.5	87.7
99	100.8	523.6	2.5	900	900	907	63.9	57.8	44.5	76.4	66.2	2245	725	205.7	88
99	100.8	519	3	900	903	896	64.1	57.8	44.5	76.4	66.2	2249	747	221.7	100.4
99	101.1	522.3	3.3	900	894	907	64.1	57.8	44.5	76.4	66.2	2251	778	215.5	98.3
99	100.9	520	3.8	900	901	899	64.1	57.8	44.5	76.4	66.2	2247	812	209.9	98
99	100.9	522	3.9	900	905	893	64.1	57.9	44.5	76.5	66.2	2250	846	212.5	103
99	100.8	520.8	4.2	900	891	902	64.1	57.9	44.5	76.4	66.2	2247	877	229.2	102
99	100.9	517.8	5.9	900	891	892	64.2	58	44.5	76.4	66.2	2239	904	246.5	111.9

99	100.8	519.2	6.5	900	905	898	64.2	58	44.4	76.6	66.2	2240	928	255.2	115.6
99	100.8	515.4	7.2	911	891	901	64.4	58	44.4	76.6	66.2	2236	959	273.8	120.9
99	100.9	514.1	7.6	910	922	910	64.4	58	44.4	76.6	66.3	2237	997	278.6	130
99	100.7	517.8	7.7	930	912	925	64.5	58.2	44.4	76.6	66.3	2234	1038	268.7	125.6
99	100.9	515.4	8.6	910	927	924	64.5	58.2	44.4	76.6	66.3	2238	1080	282.4	126.7
99	100.8	511.4	9.4	959	957	942	64.6	58.3	44.4	76.6	66.3	2232	1124	302.3	142.3
99	100.8	512.9	10.5	989	976	970	64.6	58.3	44.4	76.6	66.3	2228	1163	310.4	142.1
99	100.7	515.5	11.9	999	997	989	64.6	58.3	44.4	76.6	66.3	2228	1198	317.7	145.9
99	100.8	510.6	12.8	1021	1001	992	64.8	58.4	44.4	76.8	66.3	2224	1236	324.5	150.1
99	100.7	510.7	13.2	1013	1010	1010	64.8	58.4	44.4	76.8	66.3	2226	1275	327.3	159.6
99	100.6	512.2	13.3	1028	1038	1027	64.8	58.5	44.4	76.8	66.3	2224	1312	335.8	154.3
99	100.7	509.6	13.9	1057	1054	1061	64.9	58.5	44.4	76.8	66.5	2219	1351	356.6	163.4
99	100.8	507.7	17.2	1099	1092	1074	65.1	58.5	44.4	76.8	66.5	2214	1395	384.6	172.1
99	100.8	508.9	17.6	1088	1101	1084	65.2	58.6	44.4	76.9	66.5	2218	1439	386.1	178.9
99	100.8	508.8	18.9	1100	1099	1101	65.4	58.7	44.4	76.9	66.5	2213	1484	371.8	174
99	100.8	508	19.5	1098	1088	1089	65.4	58.7	44.4	77	66.5	2217	1532	388.4	173.9
99	100.6	508.8	19.4	1097	1111	1088	65.5	58.9	44.4	77	66.6	2217	1579	393.8	182.2
99	100.8	505.1	20.2	1094	1083	1108	65.6	58.9	44.4	77	66.6	2218	1615	394.6	176.6
99	100.6	500.9	20.6	1100	1087	1097	65.8	58.9	44.4	77	66.6	2216	1643	402.1	183.5
99	100.9	502.7	20.7	1100	1090	1088	65.9	59	44.4	77	66.6	2210	1668	403	181.6
99	100.7	500.7	22.8	1100	1083	1094	65.9	59.1	44.4	77.1	66.8	2204	1690	416.4	192.2
99	100.8	500.3	25.6	1100	1084	1111	65.9	59.1	44.4	77.1	66.8	2209	1711	424.3	197.5
99	100.7	500.1	26.3	1100	1086	1095	66.1	59.1	44.4	77.1	66.8	2205	1735	425.6	197.8
99	100.8	498.8	28.2	1100	1096	1097	66.1	59.1	44.5	77.2	66.9	2202	1762	425.2	200.9
99	100.9	500.5	28.2	1100	1075	1094	66.2	59.1	44.5	77.2	66.9	2205	1787	425.8	198.1
99	100.9	497.9	27.6	1100	1086	1094	66.2	59.2	44.5	77.3	66.9	2202	1809	426.9	200.9
99	100.8	497.7	27.7	1100	1081	1090	66.2	59.2	44.5	77.3	66.9	2206	1825	426.1	200.9
99	100.6	497.2	28.7	1100	1111	1084	66.3	59.3	44.6	77.3	67	2205	1843	463.2	201.1
99	100.6	495.4	31	1100	1077	1081	66.5	59.3	44.6	77.4	67	2201	1865	479.9	221.2
99	100.6	495.5	33.6	1100	1080	1117	66.6	59.3	44.6	77.5	67.1	2198	1890	461.1	216.1
99	100.7	495.8	35.3	1100	1092	1091	66.6	59.4	44.7	77.5	67.2	2200	1919	464.3	216
99	100.6	497.3	36.8	1100	1093	1085	66.6	59.5	44.7	77.5	67.2	2199	1956	466.5	218.5
99	100.6	497.8	37.4	1100	1107	1083	66.8	59.5	44.8	77.5	67.3	2195	1986	466.2	217.7
99	100.7	492.9	37.2	1100	1076	1112	66.8	59.5	44.8	77.5	67.3	2197	2004	463.5	216.4
99	100.6	495	36.7	1100	1109	1086	66.9	59.7	44.8	77.6	67.3	2199	2017	464.7	216.7
99	100.7	496.4	37.5	1100	1084	1094	67	59.7	44.9	77.7	67.5	2199	2024	467.5	218.5
99	100.7	494.8	37.3	1100	1076	1105	67	59.7	45	77.7	67.6	2201	2023	466.9	218.3
99	100.8	495.7	37.3	1100	1115	1086	67.2	59.8	45	77.7	67.6	2199	2021	463.9	217.6
99	100.7	493.3	37.4	1100	1086	1083	67.2	59.8	45.1	77.8	67.6	2195	2023	470.7	217.5
99	100.9	497.8	38.4	1100	1084	1106	67.3	59.8	45.2	77.9	67.8	2189	2031	500	220.1
99	100.7	494.4	44	1100	1088	1085	67.3	59.9	45.2	77.9	67.8	2182	2048	520.5	240
99	100.8	491.3	47.9	1100	1096	1107	67.5	59.9	45.3	78	67.9	2185	2075	520.3	244.4
99	100.8	494.6	48.5	1100	1094	1089	67.5	59.9	45.4	78	67.9	2182	2113	520.5	242.4
99	100.7	492.7	50	1100	1108	1109	67.6	60	45.5	78	67.9	2187	2159	521.2	243.4
99	100.8	492.7	49.1	1100	1109	1111	67.8	60	45.6	78	68	2183	2203	521.4	244.4
99	100.7	491.6	49.3	1100	1097	1092	67.8	60.1	45.6	78.1	68	2184	2236	519.7	244.4
99	100.7	491.3	50.5	1100	1077	1088	67.9	60.1	45.7	78.2	68.1	2186	2259	518	241.2
99	100.7	493	49.7	1100	1096	1082	68	60.1	45.9	78.2	68.2	2187	2272	521.8	243.6
99	100.8	487.9	51	1100	1096	1089	68.2	60.1	45.9	78.3	68.3	2187	2276	525.9	244.3
99	100.6	490.1	51.8	1100	1082	1086	68.3	60.3	46	78.4	68.3	2191	2276	523.1	245.1
99	100.7	490.7	51.2	1100	1099	1117	68.3	60.3	46.1	78.4	68.3	2182	2281	544.7	251.6
99	100.7	488.5	54.3	1100	1084	1104	68.5	60.3	46.2	78.4	68.5	2179	2295	568.9	265.7
99	100.7	488.6	58.3	1100	1104	1095	68.6	60.4	46.4	78.6	68.6	2177	2319	571.4	264.4
99	100.6	485.7	58.1	1100	1077	1112	68.7	60.4	46.5	78.6	68.7	2178	2350	573.1	268.5
99	100.7	488.3	59.8	1100	1102	1108	68.7	60.6	46.6	78.7	68.7	2174	2391	574.2	268.5
99	100.7	488.9	59.5	1100	1084	1083	68.9	60.6	46.7	78.7	68.9	2179	2429	550.7	269.1
99	100.8	486.3	60.4	1100	1085	1099	69	60.7	46.8	78.8	68.9	2178	2451	525.9	244.1
99	100.6	488.8	60.2	1100	1113	1126	69	60.7	46.9	78.8	68.9	2176	2461	560.3	257.4
99	100.8	485.7	61.3	1100	1079	1091	69.2	60.9	47	78.9	69	2167	2468	604.4	281.7
99	100.8	486.5	65.3	1100	1080	1122	69.3	60.9	47.2	78.9	69.1	2166	2474	615.7	287.9
99	100.7	485.5	67.3	1100	1107	1080	69.4	60.9	47.5	78.9	69.3	2165	2491	618.2	289.4
99	100.6	483	69.3	1100	1110	1078	69.6	61	47.5	79.1	69.4	2166	2532	618.7	289.5

99	100.7	481	69.6	1100	1107	1080	69.6	61	47.7	79.1	69.4	2170	2580	589.6	288.6
99	100.6	486.5	69.5	1100	1090	1091	69.7	61.1	47.8	79.2	69.6	2163	2621	593.7	277.9
99	100.6	485	70.3	1100	1077	1116	69.9	61.1	47.9	79.3	69.6	2169	2650	592.6	290.5
99	100.5	482.5	69.6	1100	1080	1112	70	61.1	48.1	79.3	69.7	2163	2666	613.3	276.4
99	100.7	484.3	74.4	1100	1080	1079	70	61.3	48.3	79.4	69.9	2149	2675	674.3	308.2
99	100.8	483.2	77.1	1100	1071	1108	70.2	61.3	48.4	79.5	69.9	2142	2697	681.1	317.6
99	100.8	481	79	1100	1096	1081	70.4	61.3	48.6	79.5	70	2136	2737	680.5	319.1
99	100.7	481.5	81.5	1100	1114	1083	70.4	61.4	48.8	79.6	70.1	2136	2794	650.5	306.1
99	100.7	479.6	80.5	1100	1130	1124	70.5	61.4	48.9	79.6	70.2	2136	2859	653.5	304.2
99	100.8	480.7	79.8	1100	1122	1092	70.5	61.6	49.1	79.6	70.3	2135	2918	691.6	321.1
99	100.8	479.9	82.4	1100	1089	1095	70.7	61.6	49.4	79.8	70.5	2129	2960	680.5	324.1
99	100.7	477.2	83.7	1100	1122	1079	70.9	61.7	49.6	79.8	70.5	2121	2984	668.9	305.4
99	100.7	475.1	87.7	1100	1116	1097	71.1	61.7	49.7	79.9	70.6	2109	3007	706.3	331
99	100.8	474.3	89.9	1100	1078	1125	71.3	61.8	49.9	80	70.7	2107	3041	746.7	334.1
99	100.7	473.8	90.4	1100	1094	1106	71.3	61.8	50.1	80	70.9	2112	3076	714	335.6
99	100.6	474.1	92.1	1100	1080	1115	71.4	62	50.3	80	71	2106	3117	711	332.7
99	100.7	473.8	92.4	1100	1068	1077	71.6	62	50.7	80.2	71.3	2109	3168	712.1	335.6
99	100.8	472.4	91.9	1100	1119	1123	71.6	62.1	51	80.2	71.3	2109	3203	682.2	317
99	100.8	472.1	92.6	1100	1118	1078	71.8	62.1	51.2	80.5	71.4	2113	3208	685.1	333.8
99	100.9	473.9	93	1100	1106	1125	72	62.1	51.4	80.5	71.6	2105	3205	698	322.4
99	100.7	469.2	94.6	1100	1113	1126	72	62.3	51.6	80.5	71.6	2089	3204	758.3	336.4
99	100.5	466.6	96.8	1100	1073	1079	72.1	62.3	51.8	80.7	71.8	2078	3209	775.6	379.3
99	100.7	465.4	98.6	1100	1117	1072	72.1	62.4	52.1	80.7	72	2078	3239	783.1	349.5
99	100.8	465.3	100.7	1100	1132	1072	72.3	62.4	52.3	80.7	72	2074	3301	787.5	383
99	100.7	463.9	102.6	1100	1069	1076	72.5	62.4	52.8	80.9	72.3	2070	3379	755.1	371.1
99	100.8	462	103.9	1100	1079	1076	72.7	62.4	53.1	81.2	72.4	2071	3447	755.7	368.2
99	100.7	462.3	102.1	1100	1076	1077	72.7	62.5	53.3	81.2	72.5	2064	3499	754.2	371.2
99	100.6	459.8	105.7	1100	1075	1099	72.7	62.5	53.5	81.3	72.5	2075	3533	790.3	370.4
99	100.6	461.7	104.4	1100	1077	1135	72.9	62.7	53.8	81.3	72.6	2066	3548	805.1	371.5
99	100.6	458.1	105.4	1100	1078	1074	73	62.7	54.2	81.5	72.9	2048	3569	843.6	359.9
99	100.6	451.3	111.3	1100	1127	1066	73	62.8	54.5	81.6	73	2020	3615	872	403.2
99	100.8	450.7	114.1	1100	1110	1073	73.2	62.8	54.8	81.6	73.2	2022	3683	838.4	389.9
99	100.8	448.8	115	1100	1067	1067	73.4	62.8	55	81.9	73.3	2017	3763	838.6	392.9
99	100.7	447.8	116.3	1100	1062	1075	73.4	62.8	55.3	81.8	73.4	2010	3849	838.4	392.2
99	100.6	447.4	116.7	1100	1072	1135	73.6	63	55.5	81.9	73.6	2016	3919	842.1	393.3
99	100.7	446.3	116.9	1100	1119	1066	73.8	63	55.8	82	73.8	2006	3959	841.6	395
99	100.8	444.8	116.4	1100	1065	1094	73.9	63	56.3	82.1	73.9	2010	3983	843.8	392.9
99	100.7	443.9	116.5	1100	1063	1076	74.1	63.1	56.5	82.3	74.1	2010	3997	855.5	395.9
99	100.6	441.1	118.3	1100	1067	1066	74.1	63.1	56.8	82.3	74.3	1989	4007	865.8	402.9
99	100.7	440.6	116.6	1100	1142	1132	74.3	63.1	57	82.5	74.4	1992	4022	871.1	387.5
99	100.8	441.2	118.3	1100	1072	1070	74.5	63.2	57.3	82.6	74.5	1983	4051	873.7	408.8
99	100.9	439.4	119.3	1100	1072	1072	74.5	63.2	57.5	82.6	74.6	1992	4084	875	409
99	100.7	440.9	118.6	1100	1064	1120	74.8	63.2	57.9	82.8	74.8	1983	4120	876.7	389.2
99	100.8	438.1	118	1100	1114	1072	75	63.4	58.1	82.8	74.8	1982	4154	884.8	409.5
99	100.8	437.8	118.8	1100	1068	1121	75.2	63.4	58.3	83	74.8	1970	4187	898.9	397.7
99	100.8	436.2	121.8	1100	1140	1076	75.2	63.4	58.5	83	74.8	1970	4220	909.6	426.2
99	100.7	434.4	122.3	1100	1084	1072	75.4	63.5	58.8	83.2	74.8	1963	4257	915	430.3
99	100.5	433.5	121.3	1100	1110	1076	75.4	63.5	59	83.3	74.7	1956	4297	917.5	431.7
99	100.6	433.2	123.3	1100	1095	1080	75.5	63.5	59.2	83.3	74.5	1957	4342	919.7	430.8
99	100.6	434.6	122.4	1100	1125	1075	75.7	63.5	59.6	83.5	74.1	1951	4385	922	432.2
99	100.6	433.8	120.7	1100	1127	1080	75.7	63.7	59.8	83.5	73.9	1949	4427	930.6	433.8
99	100.6	433	124.2	1100	1069	1132	75.9	63.7	60	83.7	73.6	1925	4458	894.4	438.4
99	100.9	430.5	126.2	1100	1064	1077	75.9	63.8	60.2	83.7	73.4	1920	4470	855.3	399.2
99	100.8	428.2	123.3	1100	1069	1116	76.1	63.8	60.5	83.7	73.2	1913	4464	860.2	425.2
99	100.8	427	123.3	1100	1069	1069	76.1	63.8	60.7	84	73	1903	4446	864.7	426.2
99	100.8	423.3	123.4	1100	1071	1101	76.4	63.8	61.1	84.1	72.8	1888	4415	865.8	404.3
99	100.8	423.2	124.4	1100	1124	1063	76.6	63.8	61.3	84.2	72.8	1892	4381	864.7	403.6
99	100.8	423.2	126	1100	1072	1085	76.6	63.8	61.5	84.2	72.7	1890	4371	915.2	427.5
99	100.7	423.2	124.2	1100	1072	1065	76.6	63.9	61.7	84.2	72.8	1886	4393	965.2	428.4
99	100.7	423.7	124.3	1100	1116	1109	76.8	63.9	61.9	84.4	72.9	1895	4432	914.1	429.2
99	100.8	425.1	123.7	1100	1062	1077	77	63.9	62.1	84.4	72.9	1905	4424	605.4	404.7
99	100.7	380.3	-2.5	1200	1178	1178	77	63.9	62	84.7	73.2	1390	4334	323.2	176.4

99	100.7	375.5	-2.5	1192	1186	1171	77	63.9	61.8	84.7	73.4	1290	4132	297.6	150.7
99	100.7	367	-2.5	1133	1127	1161	77	63.9	61.6	84.9	73.4	1233	3803	298.4	150.6
99	100.6	349.4	-1.7	800	774	774	77	63.9	61.6	84.9	73.6	1146	3379	298.6	139.3
99	100.7	354.6	-2.5	800	798	797	77	63.8	61.6	84.9	73.8	1212	2989	323	138.9
99	100.7	394.7	-3.3	828	808	919	76.8	63.8	61.5	85	73.8	1428	2703	399.3	175.8
99	100.5	422.1	-6.1	1071	1053	1074	76.6	63.7	61.5	85.1	73.9	1834	2521	457.5	199.9
99	100.7	448	-6.9	1100	1087	1126	76.6	63.7	61.5	85.1	74.2	1985	2449	515.8	227.6
99	100.7	470.9	-7.5	1100	1099	1095	76.6	63.7	61.5	85.2	74.3	2108	2455	514.6	265.9
99	100.8	489	-8.4	1100	1094	1104	76.6	63.8	61.5	85.4	74.4	2224	2418	230.9	240

Anexo XV. Acceso digital a la documentación de la tesis.

El código fuente, base de datos, y tablas de resultados se pueden revisar en el siguiente enlace:

<https://github.com/Cesar-Mendoza-M6/Desempeno-de-MCI/releases/tag/archivos>