

UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE INGENIERÍA INDUSTRIAL Y DE SISTEMAS



TESIS

**APLICACIÓN DE ALGORITMOS DE APRENDIZAJE DE MÁQUINA Y
HEURÍSTICOS PARA EL PATRULLAJE DE ZONAS DELICTIVAS EN UN
DISTRITO**

**PARA OBTENER EL TÍTULO PROFESIONAL DE:
INGENIERO DE SISTEMAS**

**ELABORADO POR:
LUIGI CESAR CASTRO ALVIS**

id [0009-0005-2601-2530](https://orcid.org/0009-0005-2601-2530)

**ASESOR:
DR. LUIS ALBERTO LUJAN CAMPOS**

id [0000-0001-7197-2526](https://orcid.org/0000-0001-7197-2526)

LIMA – PERÚ

2024

Citar / How to cite	(Castro, 2024)
Referencia/Reference	Castro, L. (2024). <i>Aplicación de Algoritmos de Aprendizaje de Máquina y Heurísticos para el Patrullaje de Zonas Delictivas en un Distrito</i> . [Tesis de pregrado, Universidad Nacional de Ingeniería]. Repositorio Institucional Cybertesis UNI.
Estilo / Style: APA (7ma Ed.)	

Citar / How to cite	Castro Alvis [1]
Referencia/Reference	[1] L. Castro Alvis, "Aplicación de Algoritmos de Aprendizaje de Máquina y Heurísticos para el Patrullaje de Zonas Delictivas en un Distrito" [Tesis de pregrado]. Lima (Perú), Universidad Nacional de Ingeniería, 2024.
Estilo / Style: IEEE (2022)	

DEDICATORIA

Esta tesis está dedicada a mi papá Enrique y mi mamá Aurelia, y que durante toda mi vida me brindaron su apoyo y orientación en la consecución de mis logros y objetivos.

AGRADECIMIENTOS

Expreso mi profundo agradecimiento al Dr. Luis Alberto Lujan Campos, mi Asesor de tesis, por la orientación y retroalimentación en la realización de la tesis.

Agradezco a mis compañeros y amigos de la Universidad con los cuales compartí numerosas actividades académicas y recreativas.

RESUMEN

El principal objetivo de la tesis es reducir la distancia del camino de patrullaje entre las zonas críticas de delincuencia en Pueblo Libre. Se busca una eficiente utilización de los recursos de seguridad con los que cuenta la Municipalidad. El diseño de la tesis es el preexperimental, de tipo prueba previa y posterior con un solo grupo. Se usará un caso de estudio, el cual tendrá el ámbito de un distrito. El nivel será explicativo, el enfoque será cuantitativo, y se utilizará el Análisis Documental. Se empleó la ficha de registro como instrumento para la recopilación de datos., y la herramienta a utilizar será Jupyter Notebook. La población y la muestra serán 692 denuncias policiales, el cual es el conjunto de denuncias policiales por robo o hurto en Pueblo Libre del 01/01/2022 al 15/12/2022 ocurridos en los 8 sectores del distrito. El aporte que se buscará en la tesis es la mejora del patrullaje de zonas críticas de delincuencia de un distrito utilizando algoritmos de agrupamiento: algoritmos DBSCAN y K-Medias (Algoritmos No Supervisados de Aprendizaje de Máquina) y algoritmos genéticos para calcular la ruta más corta (Algoritmos Heurísticos).

Palabras clave — Aprendizaje de Máquina, Agrupamiento, Patrullaje, Zona Crítica de Delincuencia, Algoritmos Genéticos

ABSTRACT

The main objective of the thesis is to reduce the patrolling distance between critical crime zones in Pueblo Libre. The aim is to efficiently utilize the security resources available to the Municipality. The design of the thesis is pre-experimental, with a pre- and post-test type using a single group. A case study will be used, covering the scope of a district. The level will be explanatory, the approach will be quantitative, and Documentary Analysis will be utilized. A registration form will be used as the data collection instrument, and the tool to be used will be Jupiter Notebook. The population and the sample will be 692 police reports, which is the set of police reports for robbery or theft in Pueblo Libre from 01/01/2022 to 15/12/2022 occurred in the 8 sectors of the district. The contribution that will be sought in the thesis is the improvement of the patrolling of critical crime areas in a district using clustering algorithms: DBSCAN and K-Means algorithms (Machine Learning Algorithms) and Genetic Algorithms to calculate the shortest route (Heuristic Algorithms).

Keywords — Machine Learning, Clustering, Patrolling, Crime Hot Spot, Genetic Algorithm

ÍNDICE

	Pág.
DEDICATORIA	i
AGRADECIMIENTOS.....	ii
RESUMEN.....	iii
ABSTRACT	iv
ÍNDICE	v
LISTA DE TABLAS	x
LISTA DE FIGURAS.....	xii
LISTA DE ANEXOS.....	xiv
INTRODUCCIÓN.....	1
CAPÍTULO I PARTE INTRODUCTORIA DE LA TESIS	3
1.1. GENERALIDADES	3
1.2. DESCRIPCIÓN DEL PROBLEMA DE INVESTIGACIÓN.....	3
1.2.1. Mapa de Puntos.....	7
1.2.2. Problema General.....	8
1.2.3. Problemas Específicos.....	8
1.3. OBJETIVOS.....	9
1.3.1. Objetivo General.....	9
1.3.2. Objetivos Específicos.....	9
1.4. HIPÓTESIS.....	9

1.4.1. Hipótesis General	9
1.4.2. Hipótesis Específicas	9
1.5. MATRIZ DE CONSISTENCIA.....	10
1.6. MATRIZ DE OPERACIONALIZACIÓN DE VARIABLES	14
1.7. ALCANCE Y LIMITACIONES DE LA TESIS	24
1.8. JUSTIFICACIÓN DEL ESTUDIO	25
1.9. METODOLOGÍA DE INVESTIGACIÓN.....	25
1.9.1. Diseño y Nivel de la Investigación.....	25
1.9.2. Enfoque de la Investigación	26
1.9.3. Población y Muestra	26
1.9.4. Técnica e Instrumento de recolección de datos	26
1.9.4.1. Técnicas de recolección de datos	26
1.9.4.2. Instrumento de recolección de datos.....	27
1.9.4.3. Herramientas para el procesamiento de datos	27
1.9.5. Metodología de Análisis de Datos	28
1.9.6. Aspectos Éticos	30
CAPÍTULO II MARCOS TEÓRICO Y CONCEPTUAL.....	31
2.1. ANTECEDENTES DE INVESTIGACIÓN	31
2.1.1. Artículo 1	31
2.1.2. Artículo 2	32
2.1.3. Artículo 3	33
2.1.4. Artículo 4	36
2.1.5. Tesis 1	37
2.1.6. Tesis 2	38
2.2. MARCO TEÓRICO	39
2.2.1. Patrullaje de Zonas Críticas de Delincuencia	39

2.2.1.1. Patrullaje Dirigido.....	39
2.2.1.2. Concentración del Crimen.....	40
2.2.2. Aprendizaje de Máquina	40
2.2.2.1. Aprendizaje No Supervisado.....	41
2.2.3. Algoritmos Heurísticos	46
2.2.3.1. Algoritmo Genético	47
2.2.3.2. Algoritmo Memético	47
2.3. MARCO CONCEPTUAL	48
2.3.1. Patrullaje de Zonas Críticas	48
2.3.2. Densidad Delictiva	48
2.3.3. Zona Crítica	48
2.3.4. Localización Específica.....	49
2.3.5. Coordenada Geográfica.....	49
2.3.6. Paralelos.....	49
2.3.7. Meridianos	49
2.3.8. Longitud.....	50
2.3.9. L a t i t u d.....	50
CAPÍTULO III DESARROLLO DEL TRABAJO DE INVESTIGACIÓN	52
3.1. COMPRENSIÓN DEL NEGOCIO	52
3.1.1. Determinar los Objetivos de la Seguridad Ciudadana	52
3.1.2. Evaluación de la Situación	53
3.1.3. Determinar los objetivos de la minería de datos.....	56
3.1.4. Producir el plan del proyecto.....	56
3.2. COMPRENSIÓN DE LOS DATOS.....	58
3.2.1. Recolección de datos iniciales	58
3.2.2. Descripción de los datos	59

3.2.3. Exploración de los datos.....	61
3.2.4. Comprobación de la calidad de los datos.....	69
3.3. PREPARACIÓN DE LOS DATOS.....	69
3.3.1. Selección de los datos	69
3.3.2. Limpieza de los datos	69
3.3.3. Integración y Normalización de los datos.....	70
3.3.4. Formateo de los datos	71
3.4. MODELADO	72
3.4.1. Seleccionar las técnicas de modelado	72
3.4.2. Generar el Plan de Prueba	73
3.4.3. Creación de Modelos	73
3.4.3.1. Algoritmo K- Medias.....	73
3.4.3.2. Algoritmo DBSCAN.....	76
3.4.3.3. Algoritmo HDBSCAN	80
3.4.3.4. Análisis de las Densidades Delictivas	84
3.4.3.5. Elección del agrupamiento óptimo y determinación de las zonas críticas de delincuencia.....	106
3.4.3.6. Algoritmo Genético	112
3.4.3.7. Algoritmo Memético	117
3.5. EVALUACIÓN.....	120
3.5.1. Evaluación de los Resultados del Algoritmo Genético y Memético	120
3.5.2. Determinación de la Distancia Total y Recorrido de Patrullaje de Zonas Críticas de Delincuencia Óptimo.....	124
CAPÍTULO IV ANÁLISIS Y DISCUSIÓN DE RESULTADOS.....	130
4.1. CONTEXTUALIZACIÓN DE RESULTADOS.....	130
4.2. PRESENTACIÓN DE RESULTADOS.....	132

4.3. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS	135
4.3.1. Medición de Densidad y Determinación de zonas críticas de delincuencia. 135	
4.3.2. Reducción de la Distancia Total de Patrullaje con la Aplicación de Algoritmos de Aprendizaje de Máquina y Heurísticos	135
4.3.3. Determinación del camino mínimo de las Zonas Crítica de Delincuencia... 136	
4.4. CONTRASTACIÓN DE HIPÓTESIS	136
4.4.1. Contrastación de la Hipótesis Específica 1	136
4.4.2. Contrastación de la Hipótesis Específica 2	137
4.4.3. Contrastación de la Hipótesis General	138
CONCLUSIONES	141
RECOMENDACIONES	143
REFERENCIAS BIBLIOGRÁFICAS.....	145
ANEXOS	151

LISTA DE TABLAS

Tabla 1 Ruta de Patrullaje Antes de la Aplicación de los Algoritmos de Agrupamiento (Aprendizaje de Máquina) y Heurísticos.....	5
Tabla 2 Denuncias por tipo de delitos en Pueblo Libre en 2022	7
Tabla 3 Riesgos y Medidas de Mitigación.....	55
Tabla 4 Atributos del Registro de Datos Delictivos	59
Tabla 5 Tipo de Dato de los Atributos del Conjunto de Datos.....	60
Tabla 6 Tipo de Datos de las Coordenadas Limítrofes de Pueblo Libre	61
Tabla 7 Parámetros y Métricas de Evaluación del Algoritmo DBSCAN	77
Tabla 8 Parámetros y Métricas de Resultados del Algoritmo HDBSCAN	82
Tabla 9 Densidades Delictivas del Agrupamiento K-Medias (k=3).....	86
Tabla 10 Densidades Delictivas del Agrupamiento K-Medias (k=17).....	87
Tabla 11 Densidades Delictivas del Agrupamiento DBSCAN (min_samples=2, eps=2.1)	88
Tabla 12 Densidades Delictivas del Agrupamiento DBSCAN (min_samples = 2, eps = 2.1) con grupos filtrados	90
Tabla 13 Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2).....	93
Tabla 14 Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2) con grupos filtrados	95
Tabla 15 Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 3, min_samples = 2, cluster_selection_epsilon = 2).....	97
Tabla 16 Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 3, min_samples = 2, cluster_selection_epsilon = 2) con grupos filtrados	99

Tabla 17 Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85)	101
Tabla 18 Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85) con grupos filtrados	103
Tabla 19 Análisis Comparativo de las Densidades Delictivas de los Algoritmos K-Medias, DBSCAN y HDBSCAN	105
Tabla 20 Determinación Geográfica de las Zonas Críticas de Delincuencia	108
Tabla 21 Distancias Mínimas en las 10 Ejecuciones del Algoritmo Genético.....	115
Tabla 22 Distancias Mínimas en las 10 Ejecuciones del Algoritmo Memético	119
Tabla 23 Estadísticas de los Valores de la Función Fitness de las últimas 100 generaciones del Algoritmo Genético y Algoritmo Memético	122
Tabla 24 Camino de Patrullaje (Vías) de Patrullaje de Zonas Críticas de Delincuencia de la Ruta Mínima determinada por el Algoritmo Memético	126
Tabla 25 Comparación de resultados obtenidos de la Aplicación de los Algoritmos de Agrupamiento e Indicadores de Densidad Delictiva	132
Tabla 26 Resultados de la Aplicación de los Algoritmos Heurísticos (A. Genético y A. Memético).....	134
Tabla 27 Reducción de la Distancia Total de Patrullaje de Zonas Críticas de Delincuencia de Pueblo Libre	136
Tabla 28 Estadísticas de distancias obtenidas por el Algoritmo Memético	139

LISTA DE FIGURAS

Figura 1 Mapa de Puntos Delictivos de Pueblo Libre	8
Figura 2 Desplazamiento Vectorial.....	18
Figura 3 Esquema del Marco de Trabajo CRISP-DM.....	30
Figura 4 Tipos de Algoritmos de Aprendizaje de Máquina	41
Figura 5 Esquema del Algoritmo K-Medias	43
Figura 6 Esquema de Agrupamiento mediante DBSCAN.....	45
Figura 7 Paralelos y Meridianos.....	50
Figura 8 Latitud y Longitud.....	51
Figura 9 Cantidad de Delitos por Sector Distrital.....	62
Figura 10 Cantidad de Delitos Según el Día de la Semana.....	63
Figura 11 Cantidad de Delitos según el Método de Huida.....	64
Figura 12 Cantidad de Delitos según el Mes.....	65
Figura 13 Cantidad de Delitos según el Lugar de Ocurrencia	66
Figura 14 Cantidad de Delitos según el Medio Utilizado	67
Figura 15 Histograma de la Latitud de Hechos Delictivos	68
Figura 16 Histograma de la Longitud de Hechos Delictivos	68
Figura 17 Entrenamiento y Validación de Algoritmos No Supervisados	73
Figura 18 Gráficas de las Métricas de Evaluación para el Algoritmo K-Medias	74
Figura 19 Mapa de Puntos para el Agrupamiento K-Medias (k=3)	75
Figura 20 Mapa de Puntos para el Agrupamiento K-Medias (k=17)	75
Figura 21 Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 1.95)	77
Figura 22 Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 2.00)	78
Figura 23 Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 2.05)	78
Figura 24 Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 2.10)	79

Figura 25 Agrupamiento HDBSCAN ($m_c_size^* = 2$, $min_samples = 2$, $epsilon = 2$) .	82
Figura 26 Agrupamiento HDBSCAN ($m_c_size^* = 3$, $min_samples = 2$, $epsilon = 2$) .	83
Figura 27 Agrupamiento HDBSCAN ($m_c_size^* = 4$, $min_samples = 2$, $eps^* = 1.85$)	83
Figura 28 Mapa de Puntos Delictivos de cada Zona Crítica de Delincuencia	107
Figura 29 Mapa de Centroides de Zonas Críticas de Delincuencia	107
Figura 30 Evolución de la Distancia de Patrullaje en cada Generación del Algoritmo Genético	116
Figura 31 Evolución de la Distancia de Patrullaje (Función Fitness) en cada Generación del Algoritmo Memético	119
Figura 32 Función Fitness de las Soluciones (Algoritmos Genético y Memético)	121
Figura 33 Mapa de Puntos de la Ruta Mínima de Patrullaje encontrada por el Algoritmo Genético	123
Figura 34 Mapa de Puntos de la Ruta Mínima de Patrullaje encontrada por el Algoritmo Memético	123
Figura 35 Mapa del Camino (Trayectoria) de Patrullaje de Zonas Críticas de Delincuencia de la Ruta Mínima determinada por el Algoritmo Memético.....	125
Figura 36 Proceso de la Aplicación de Algoritmos de Aprendizaje de Máquina y Heurísticos para el Patrullaje de Zonas Críticas de Delincuencia	131
Figura 37 Mapa de Centroides de Zonas Críticas de Delincuencia	133
Figura 38 Distribución t-Student con 9 grados de libertad	140

LISTA DE ANEXOS

Anexo 1	Glosario de la Terminología de Minería de Datos	151
Anexo 2	Código para la Comprensión de los Datos	154
Anexo 3	Código para la Preparación de los Datos	157
Anexo 4	Código para el Modelado de los Algoritmos de Agrupamiento	159
Anexo 5	Código para el Cálculo de Densidades Delictivas	164
Anexo 6	Código para Determinar las Zonas Críticas de Delincuencia.....	171
Anexo 7	Código del Desarrollo del Algoritmo Genético	173
Anexo 8	Código del Desarrollo del Algoritmo Memético.....	178
Anexo 9	Código de Evaluación de los Algoritmos Genético y Memético	180
Anexo 10	Generación del Indicador de Desigualdad en la Distribución Geográfica de delitos (Índice de Gini)	182
Anexo 11	Cuadro Interpretativo de los resultados finales indicados en las tablas	184

INTRODUCCIÓN

El patrullaje de zonas delictivas establece un conjunto de retos como la determinación de las zonas con alto nivel delictivo, conocidas como Zonas Críticas de Delincuencia, exige un proceso riguroso de análisis de los datos geográficos de los incidentes delictivos registrados en un área y tiempo determinado, en este sentido, los decisores de las políticas de seguridad ciudadana no cuentan con procesos estandarizados de análisis de los datos delictivos, a pesar de contar disposición de éstos.

Por esto, resulta importante realizar un análisis de dichos datos delictivos para generar indicadores de densidad delictiva, para la determinación de las Zonas Críticas de Delincuencia y así mejorar el proceso de patrullaje de Zonas Críticas de Delincuencia, con el objetivo potencial de reducir los elevados índices de inseguridad ciudadana que afecta negativamente nuestras ciudades.

En esta tesis, se abordó la aplicación de algoritmos de agrupamiento (Algoritmos No Supervisados de Aprendizaje de Máquina) para determinar las Zonas Críticas de Delincuencias, así como la aplicación de algoritmos heurísticos (Algoritmo Genético y Algoritmo Memético) para minimizar el recorrido de patrullaje de zonas críticas de delincuencia.

La presente tesis está dividida en 4 capítulos. La descripción del problema de investigación se detalla en el Capítulo I, además se mencionan los objetivos e hipótesis de investigación, las matrices de consistencia y operacionalización, los

alcances y limitaciones de la investigación, la justificación del estudio y culmina con la metodología. El Capítulo II, consta de 3 partes, las cuales abarcan los antecedentes investigativos, el marco teórico y el marco de conceptos. El Capítulo III ejecuta la metodología de análisis de datos, la cual se denomina Proceso Estándar Interindustrial para la Minería de Datos (CRISP - DM). El Capítulo IV, consta de la realización del análisis y la discusión de los resultados obtenidos por los algoritmos de Aprendizaje de Máquina y Heurísticos, el cual se extrae toda la información relevante obtenida en el Capítulo 3 y se discute su implicancia con los objetivos del estudio. Posteriormente, se brindan las conclusiones y recomendaciones correspondientes. Por último, se brindan las referencias bibliográficas que alimentaron el desarrollo del estudio y los anexos correspondientes.

CAPÍTULO I

PARTE INTRODUCTORIA DE LA TESIS

1.1. GENERALIDADES

En los últimos años ha existido un avance en los métodos de prevención del delito con respecto a la formulación de técnicas para disuadir el crimen. Entre uno de esos métodos se encuentra el Patrullaje de Zonas Críticas de Delincuencia.

Reisig y Kane (2014), Weisburd y Telep (2014), Weisburd y Eck (2004) definieron el Patrullaje de Zonas Críticas de Delincuencia como un “Enfoque de control de la delincuencia que incluye el patrullaje dirigido a lugares específicos donde el crimen está concentrado”. Este concepto considera al patrullaje dirigido como una estrategia policial en la que los agentes patrullan zonas específicas para disuadir la delincuencia o responder rápidamente cuando se produce un delito. (McKee, 2023).

1.2. DESCRIPCIÓN DEL PROBLEMA DE INVESTIGACIÓN

En los gobiernos locales (municipalidades) se dispuso la creación del serenazgo municipal, el cual brinda “acciones de vigilancia pública y apoyo en atención de emergencias, así como auxilio y asistencia al ciudadano y cooperación con las entidades públicas que la requieran” (Congreso de la República del Perú, 2021, Ley 31297, Artículo 3). Es este cuerpo de serenazgos que, con la dirección de

los gerentes municipales, toman la decisión de realizar el Patrullaje de Zonas Críticas de Delincuencia en su jurisdicción.

En el **patrullaje** se busca “recorrer una cierta zona, principalmente para evitar que se cometan delitos o también que se produzcan hechos con consecuencias negativas” (Porto, 2013); por lo tanto, es importante reducir el camino entre las zonas delictivas que concentran la mayor ocurrencia de delitos ya que los recursos de vigilancia (automóviles, camionetas, motocicletas) suelen ser limitados. Como es el caso de la Municipalidad de Pueblo Libre, la cual cuenta con 34 vehículos motorizados para el servicio de patrullaje (Plan de Acción Distrital de Seguridad Ciudadana, PADSC). Sin embargo, durante los últimos meses, se viene declarando estados de emergencia, en el cual la Gerencia de Coordinación de Seguridad Ciudadana solicitó el aumento de serenos y del número de vehículos (Municipalidad de Pueblo Libre, 2019) lo cual evidencia una escasez de asignación de recursos de seguridad en el distrito.

Asimismo, existe imprecisiones en la determinación de las **zonas específicas de patrullaje**, lo que se evidencia en el PADSC de los diferentes distritos de Lima en el que no se maneja zonas georreferenciadas para el patrullaje de territorios con mayor actividad delictiva y solo se resaltan las avenidas con mayor ocurrencia relativa de incidentes. En estas zonas específicas de patrullaje, el objetivo es vigilar las zonas con crimen concentrado. Esto se refiere a que en una unidad micro geográfica un rango de porcentajes se asocia con una proporción acumulativa de delincuencia. (Weisburd, 2015). La proporción acumulativa de delincuencia es un porcentaje de delincuencia que tiene su ocurrencia en un lugar geográfico determinado. (Weisburd, 2015). En esta situación, la proporción acumulativa de delincuencia que es requerida en el patrullaje que realiza el serenazgo municipal no está medido y es inespecífico.

A partir de los sectores y vías de mayor incidencia de denuncias determinadas en el PADSC de Pueblo Libre (2023), se elaboró la Ruta de Patrullaje antes de la Aplicación de los Algoritmos de Máquina y Heurísticos, la cual se muestra en la Tabla 1, resultando que la Distancia Total de Patrullaje antes de la Aplicación de Algoritmos de Agrupamiento (Aprendizaje de Máquina) y Heurísticos es 20.784 km.

Tabla 1

Ruta de Patrullaje Antes de la Aplicación de los Algoritmos de Agrupamiento (Aprendizaje de Máquina) y Heurísticos.

Nombre de la Vía	Distancia (m)
Avenida Mariano Cornejo	1774.9
Avenida Plaza de la Bandera	249.5
Prolongación 28 de Julio	687.5
Jirón Pedro Ruiz Gallo	630.8
Avenida Brasil	2424.9
Jirón Amazonas	1036.5
Jirón Echenique	247.1
Avenida La Marina	1083.8
Avenida Universitaria	1163.3
Avenida Simón Bolívar	2054.1
Avenida Paso de los Andes	860.1
Avenida General José Leguía y Meléndez	1023.8
Avenida Coronel Juan Valer Sandoval	1014.4
Calle Bernardo de O'Higgins	519.1
Jr. José Santiago Wagner	780.3
Jr. Cabo Gutarra	637.5
Avenida La Marina	578.2
Avenida José de Sucre	220.0
Avenida Brasil	538.0
Avenida La Marina	245.3

Nombre de la Vía	Distancia (m)
Avenida General Manuel Vivanco	233.9
Avenida Coronel Juan Valer Sandoval	715.4
Avenida Simón Bolívar	638.7
Avenida Simón Bolívar	566.8
Avenida Paso de los Andes	860.1
Distancia Total de Patrullaje (Antes)	20784

Nota. Información adaptada del PADSC de Pueblo Libre, 2023. Elaboración Propia.

En el contexto nacional, no existía una recopilación georreferenciada de los incidentes delictivos ocurridos en el territorio peruano; sin embargo, el 2017 el Instituto Nacional de Estadística e Informática (INEI) presentó el Sistema Integrado de Estadísticas de la Criminalidad y Seguridad Ciudadana (DATACRIM), el cual registra los datos georreferenciados de los delitos ocurridos en el territorio peruano. Se presenta la lista de la tipología de los delitos georreferenciados por el DATACRIM (INEI, 2023):

- Robo
- Hurto
- Asalto y robo de vehículos
- Homicidio
- Micro comercialización de drogas
- Estafa y otras defraudaciones

A partir de esta información los decisores de políticas públicas tienen mayores herramientas para la ejecución de acciones para prevenir o castigar el delito. En el caso de Pueblo Libre, existe una preponderancia de la tipología de Robo y Hurto en Pueblo Libre, como se destaca en la Tabla 2.

También se destaca que los delitos contra el patrimonio son uno de los principales casos delictivos que afectan la seguridad de Pueblo Libre (PADSC, 2023), por lo que

es necesario priorizar la vigilancia que tomen en consideración la ocurrencia de este tipo de delitos.

1.2.1. Mapa de Puntos

Es aquel mapa en el cual “la información cuantitativa se representa por medio de la repetición de puntos que, a diferencia de los mapas de símbolos proporcionales, no varían de tamaño” (Universidad Politécnica de Madrid, s.f.). En la Figura 1, se logra mostrar el mapa de puntos delictivos de Pueblo Libre desde el 2017.

Tabla 2

Denuncias por tipo de delitos en Pueblo Libre en 2022

Tipo de Delito	Número de Delitos
D. Ambientales	1
D. de Robo y Hurto	692
Otros D. C. el Patrimonio	674
D. C. el Patrimonio Cultural	1
D. C. la Administración Pública	38
D.C. la Familia	5
D. C. la Fe Pública	8
D. C. la Humanidad	4
D. C. la Libertad	112
D. C. la Seguridad Pública	64
D C. la Vida, el Cuerpo y la Salud	46
D. C. la Voluntad Popular	1
TOTAL	1646

Nota. Extraído del PADSC de Pueblo Libre (2023). Cifras en unidades de delitos reportados. D. y D.C. abrevian es “Delitos” y “Delitos Contra” respectivamente.

Figura 1

Mapa de Puntos Delictivos de Pueblo Libre



Nota. Extraído de DATACRIM, INEI (2023). Delitos de Robo y Hurto reportados del 2017 al 2022.

1.2.2. Problema General

¿Cómo reducir la distancia del camino de patrullaje entre las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de aprendizaje de máquina y heurísticos?

1.2.3. Problemas Específicos

P.E.1: ¿Cómo determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos?

P.E.2: ¿Cómo determinar el camino mínimo de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos?

1.3. OBJETIVOS

1.3.1. Objetivo General

Reducir la distancia del camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos.

1.3.2. Objetivos Específicos

O.E.1: Determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos.

O.E.2: Determinar el camino mínimo de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos.

1.4. HIPÓTESIS

1.4.1. Hipótesis General

La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permiten reducir la distancia del camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre.

1.4.2. Hipótesis Específicas

H.E.1: La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permite determinar las zonas críticas de delincuencia en Pueblo Libre.

H.E.2: La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permite determinar el camino mínimo de patrullaje de las zonas críticas de delincuencia en Pueblo Libre.

1.5. MATRIZ DE CONSISTENCIA

PROBLEMA	OBJETIVOS	HIPÓTESIS	VARIABLES	METODOLOGÍA
PROBLEMA GENERAL	OBJETIVO GENERAL	HIPÓTESIS GENERAL	VARIABLE DEPENDIENTE	Tipo: Aplicada
¿Cómo reducir la distancia de camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos?	Reducir la distancia del camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos.	La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permiten reducir la distancia del camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre.	Patrullaje de las zonas críticas de delincuencia	Nivel: Explicativo Enfoque: Cuantitativo Diseño: Pre – experimental
PROBLEMAS ESPECÍFICOS	OBJETIVOS ESPECÍFICOS	HIPÓTESIS ESPECÍFICAS	VARIABLES INDEPENDIENTES	Población: 692 denuncias policiales por robo o hurto en Pueblo Libre de 2022. Muestra: 692 denuncias policiales por robo o hurto en Pueblo Libre de 2022.
¿Cómo determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos?	Determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos	La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permite determinar las zonas críticas de delincuencia en Pueblo Libre.	Algoritmos de Aprendizaje de Máquina	
¿Cómo determinar el camino mínimo en el patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos?	Determinar el camino mínimo en el patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos	La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permite determinar el camino mínimo de patrullaje de las zonas críticas de delincuencia en Pueblo Libre.	Algoritmos Heurísticos	

Desagregación de los P. Específicos

1. P. Específico 1 (PE1)

El PE1 enuncia: ¿Cómo determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos? se descompone en las siguientes cuestiones:

- No existe un marco de trabajo que determine de forma específica las zonas críticas de delincuencia, las cuales concentran actividades delictivas en base a datos históricos de los delitos cometidos en el distrito.
- No se conoce las densidades delictivas/ densidades delictivas relativas de las zonas críticas de delincuencia.
- No se conoce la densidad delictiva acumulada de las zonas críticas de delincuencia.

2. P. Específico 2 (PE2)

El PE2 enuncia: ¿Cómo determinar el camino mínimo en el patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos? se descompone en las siguientes cuestiones:

- No se conoce con un camino óptimo para el patrullaje de zonas críticas de delincuencia.
- No se conoce la distancia mínima de patrullaje, el cual recorra las zonas críticas de delincuencia.
- No existe una utilización óptima de los recursos escasos de patrullaje para la vigilancia de zonas críticas de delincuencia.

Desagregación de los Objetivos Específicos

1. O. Específico 1

El O. Específico 1 (OE1) que enuncia: Determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos se descompone en las siguientes actividades:

- Aplicar los algoritmos de Aprendizaje de Máquina No Supervisado para el agrupamiento de los datos delictivos.
- Calcular la densidad delictiva/densidad delictiva relativa de los agrupamientos obtenidos a partir de algoritmos de Aprendizaje de Máquina No Supervisado.
- Calcular la densidad delictiva acumulada/ densidad delictiva acumulada relativa de los agrupamientos obtenidos a partir de algoritmos de Aprendizaje de Máquina No Supervisado.
- Comparar los resultados obtenidos a partir de las densidades delictivas y la proporción acumulativa de delincuencia y escoger el algoritmo de agrupamiento idóneo.
- Determinar de forma específica las zonas críticas de delincuencia a partir del algoritmo de agrupamiento idóneo. Indicar las avenidas/calles involucradas en cada zona específica de delincuencia y sus respectivos centroides.

2. O. Específico 2

El O. Específico 2 (OE2) que enuncia: Determinar el camino mínimo en el patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos se descompone en las siguientes actividades:

- Aplicar en las zonas críticas de delincuencia determinadas, el Algoritmo Genético y el Algoritmo Memético adaptado al Problema del Agente Viajero (TSP).
- Determinar el camino con recorrido mínimo para el patrullaje de zonas críticas de delincuencia.
- Calcular la distancia mínima de patrullaje de zonas críticas de delincuencia.

- Comparar la distancia actual de patrullaje (Plan de Acción Distrital de Seguridad Ciudadana 2023) respecto a la distancia mínima de patrullaje calculada anteriormente.

1.6. MATRIZ DE OPERACIONALIZACIÓN DE VARIABLES

VARIABLE DEPENDIENTE	DEFINICIÓN CONCEPTUAL	DIMENSIONES	INDICADORES
Patrullaje de Zonas Críticas de Delincuencia	Según Reisig y Kane (2014), Weisburd y Telep (2014), Weisburd y Eck (2004), el Patrullaje de Zonas Críticas de Delincuencia es “Enfoque de control de la delincuencia que incluye el patrullaje dirigido a lugares específicos donde el crimen está concentrado”	Zona Crítica: Según Sherman (1995) las Zonas Críticas de Delincuencia son “pequeños lugares en los que la delincuencia es tan frecuente que es altamente predecible, al menos durante un período de un año.”	Densidad Delictiva Densidad Delictiva Relativa
		Patrullaje Dirigido: Según McKee (2023), el “patrullaje dirigido es una estrategia policial en la que los agentes patrullan zonas específicas para disuadir la delincuencia o responder rápidamente cuando se produce un delito.”	Camino de Patrullaje (Trayectoria) Distancia Total de Patrullaje

VARIABLES INDEPENDIENTES	DEFINICIÓN CONCEPTUAL	DIMENSIONES	DEFINICIÓN CONCEPTUAL	INDICADORES / MÉTRICAS
Algoritmos de Aprendizaje de Máquina	Según Murphy (2012), define el Aprendizaje Automático como un “conjunto de métodos capaces de detectar automáticamente patrones en los datos y utilizarlos para predecir datos futuros o tomar otro tipo de decisiones en condiciones de incertidumbre”.	Algoritmo K-Medias	El algoritmo de K-Medias se encarga agrupar un grupo de datos en “K” grupos. (Theobald, 2019).	Índice Silueta
		Algoritmo DBSCAN	DBSCAN es un “algoritmo de agrupamiento basado en densidad que produce un agrupamiento particional, en el que el número de grupos es determinado automáticamente por el algoritmo.” (Tan et al., 2019).	Índice Davies-Bouldin
		Algoritmo HDBSCAN	HDBSCAN es “un algoritmo de agrupamiento, el cual no necesita la especificación previa del número de grupos, generar grupos de diferentes formas y tamaños e identifica puntos de ruido.” (Campelo et al., 2013).	Índice Calinski - Harabasz
Algoritmo Heurístico	Según Russell y Norvig (2010), un algoritmo heurístico es el cual “utiliza información que va más allá de la definición del problema para utilizar conocimientos específicos lo que permite encontrar soluciones más rápidamente que una estrategia no informada”	Algoritmo Genético	Los Algoritmos Genéticos combinan la supervivencia del más apto entre las estructuras de cadenas con un intercambio de información estructurado pero aleatorio para formar un algoritmo de búsqueda. (Goldberg, 1989).	Convergencia de la Función Fitness
		Algoritmo Memético	Los Algoritmos Meméticos son “metaheurísticas basadas en poblaciones compuestas de un enfoque evolutivo y un conjunto de algoritmos de búsqueda local, los cuales son activados dentro del ciclo de generaciones.” (Neri et al., 2012)	

Descripción de los Indicadores

Variable Dependiente:

1. Densidad Delictiva:

La Densidad Delictiva se refiere a la cantidad de delitos reportados en un área determinada (en el caso de la investigación, se considera el área del distrito), generalmente medida en términos de delitos por unidad de superficie (por ejemplo: delitos / kilómetro cuadrado). Para calcular la densidad delictiva se utiliza la Ecuación 1. En la presente tesis, se utilizó las tipologías de delitos de Robo y Hurto, los cuales serán contabilizados y utilizados en el numerador de la Ecuación 1.

$$Densidad\ Delictiva = \frac{Número\ total\ de\ delitos\ (robo\ y\ hurto)}{Área\ de\ la\ zona\ (distrito)} \quad (1)$$

Asociado al concepto de Densidad Delictiva, también se puede realizar el cálculo de la Densidad Delictiva Relativa, la cual compara la Densidad Delictiva de una parte específica del distrito (Zona Crítica) con la Densidad Delictiva de la zona a analizar. La formulación de la Densidad Delictiva se muestra en la Ecuación 2.

$$Densidad\ Delictiva\ Relativa = \frac{\frac{Número\ de\ Delitos\ en\ la\ Zona\ Crítica}{Área\ de\ la\ zona\ crítica}}{\frac{Número\ total\ de\ delitos\ (robo\ y\ hurto)}{Área\ de\ la\ zona\ (distrito)}} \quad (2)$$

2. Camino de Patrullaje (Trayectoria):

El Camino de Patrullaje se representa como un grafo, el cual es un conjunto de nodos (centroides de zonas críticas de delincuencia) y aristas (desplazamientos entre las zonas críticas) que conectan estos nodos. Cada nodo en el grafo representa una ubicación específica que debe ser patrullada (áreas críticas en el distrito). El Camino de Patrullaje se denota como un ciclo cerrado (nodo inicial es el mismo al nodo final).

La definición formal de un ciclo cerrado según Teoría de Grafos es la siguiente:

Si $G(V, A)$ es un grafo, entonces:

- V es un conjunto de vértices.
- A es un conjunto de aristas, los cuales unen los vértices (V).

Un ciclo cerrado C (en un grafo $G(V, E)$) se representa en la Ecuación 3.

$$C = \{v_1, v_2, v_3, \dots, v_{k-1}, v_k, v_1\} \quad (3)$$

O también se puede representar como la Ecuación 4.

$$C = \{\{v_i, v_{i+1}\} | i = 1, \dots, n - 1\} \cup \{v_n, v_1\} \quad (4)$$

El Desplazamiento es “una magnitud vectorial que sólo depende de la posición inicial y final del cuerpo y es independiente de la trayectoria” (Fernández, 2023). La Ecuación 5 describe un desplazamiento simple con una posición inicial y una posición final. En la Figura 2 se acompaña un esquema para un mejor entendimiento de la Ecuación 5.

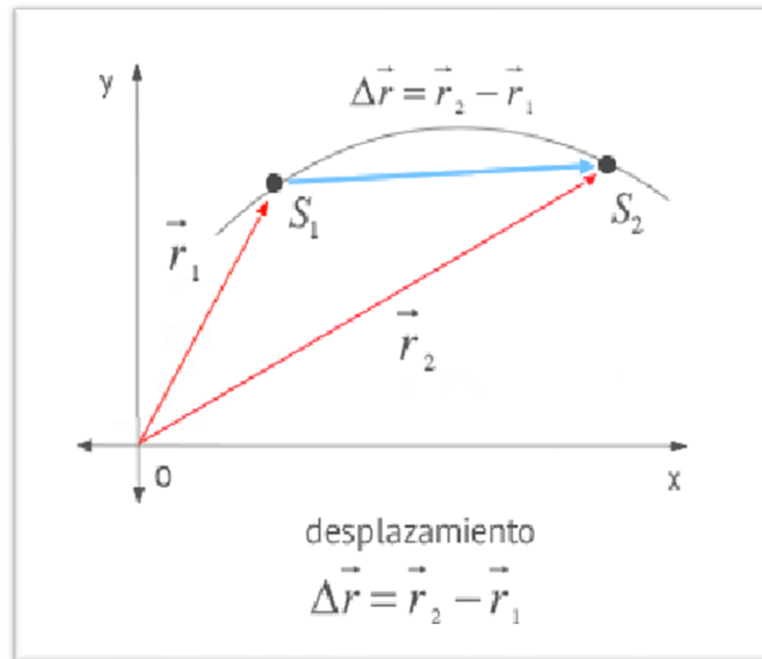
$$\Delta\vec{r} = \vec{r}_2 - \vec{r}_1 \quad (5)$$

Donde:

- \vec{r}_1 : Vector del punto de referencia a la posición inicial.
- \vec{r}_2 : Vector del punto de referencia a la posición final.

Figura 2

Desplazamiento Vectorial



Nota. Adaptado de *FisicaLab*, por Fernández, 2023.

3. Distancia Total de Patrullaje:

La Distancia Total de Patrullaje se refiere a la sumatoria de los desplazamientos entre cada uno de los centroides de las Zonas Críticas de Delincuencia. La Ecuación 6 describe la Distancia Total de Patrullaje según la notación de teoría de Grafos.

$$\text{Distancia Total de Patrullaje} = \sum_{i}^{j \neq i} x_{ij} d_{ij} \quad (6)$$

Considerando una red completa no dirigida $G(N, A)$.

- N : conjunto de nodos i (Centroides de las Zonas Críticas de Delincuencia) de la red.
- A : conjunto de arcos (i, j) en la red.
- x_{ij} : variable binaria (0 o 1) asociada a la decisión de utilizar el arco (i, j) en la solución.
- d_{ij} : distancia euclidiana del arco (i, j) en la solución.

Variable Independiente:

Los siguientes índices de evaluación fueron utilizados para medir la eficacia de los Algoritmos de Aprendizaje de Máquina No Supervisados:

1. Índice Silueta.

Es una métrica utilizada para evaluar agrupamientos que mide la similitud de cada punto de datos con su propio grupo ('cohesión intraclúster') en comparación con otros grupos ('separación interclúster'). Un Índice de Silueta más alto indica una mejor calidad de agrupamiento, donde los grupos están más cohesionados internamente y mejor separados entre sí (Rousseeuw, 1987). El Índice de Silueta para un punto dado se calcula mediante la Ecuación 7.

$$S_i = \frac{b_i - a_i}{\max \{a_i, b_i\}} \quad (7)$$

Donde:

- S_i : Índice de Silueta para el punto "i".
- a_i : Distancia promedio del punto "i" a los otros puntos del mismo clúster. (cohesión intraclúster).
- b_i : Distancia promedio de punto "i" a los puntos del clúster más cerca. (separación interclúster).

El Índice Silueta para la totalidad de los datos se obtiene calculando el promedio del Índice Silueta individual de cada punto mediante la Ecuación 8.

$$S = \frac{1}{N} \sum_{i=1}^N S_i \quad (8)$$

Donde:

- N : Número total de puntos en el conjunto de datos.
- $S_{(i)}$: Índice Silueta individual del punto “ i ”.

Interpretación del Índice de Silueta:

El Índice Silueta puede tomar un valor entre -1 y 1.

- Valores cercanos a 1 indica que los grupos están bien definidos y separados
- Valores cercanos a 0 sugieren grupos superpuestos.
- Valores cercanos a -1 indican una asignación errónea de los grupos.

En general, se busca obtener Índice Silueta más altos para evaluar agrupamientos de mayor calidad.

2. Índice de Davies - Bouldin

Es una métrica para evaluar agrupamientos que ofrece una medida de la “separación” entre grupos y la “cohesión” dentro de ellos en un conjunto de datos. Señala que la similitud de los grupos (que se supone poseen una alta densidad de datos) disminuye a medida que aumenta la distancia al vector característico del grupo. (Davies y Bouldin, 1979).

Sea el conjunto de datos $X = X_1, X_2, X_3, \dots$; el Índice Davies-Bouldin se calcula aplicando la Ecuación 9.

$$DB(i, j) = \frac{1}{k} \sum_{i=1, i \neq j}^k \max\left(\frac{\Delta(X_i) + \Delta(X_j)}{\delta(X_i, X_j)}\right) \quad (9)$$

Donde:

- $\Delta(X_i)$: Distancia promedio intraclúster dentro del grupo X_i .

- $\delta(X_i, X_j)$: Distancia Interclúster entre los grupos X_i para el grupo X_j .
(Distancia entre centros)
- k : Número de Grupos

Interpretación de los Valores Davies - Bouldin:

El valor de Davies - Bouldin puede tomar valores de 0 a más infinito.

- Un valor de 0 indica una configuración ideal.
- Un valor más bajo de Davies-Bouldin indica una mejor calidad del agrupamiento.
- Un valor más alto de Davies-Bouldin indica una peor calidad del agrupamiento.

3. Índice Calinski - Harabasz

Es una métrica que puede ser usada para la evaluación la idoneidad de los grupos en un conjunto de datos. Proporciona una medida de qué tan separados están los grupos y cuán compactos están dentro de sí mismos (Calinski y Harabasz, 1974). Este índice requiere un conjunto de pasos para ser calculado. El primer paso consiste en calcular la Dispersión Interclúster, que es la suma ponderada de las distancias al cuadrado entre los centroides de un grupo y el centroide del conjunto de datos completo. La Dispersión Interclúster (BGSS) se calcula mediante la Ecuación 10.

$$BGSS = \sum_{k=1}^K n_k \times \|C_k - C\|^2 \quad (10)$$

Donde:

- n_k : Número de observaciones (puntos) en el grupo k .
- C_k : Centroide del grupo k .
- C : Centroide de los datos.
- K : Cantidad de grupos.

El segundo paso es calcular la dispersión Intraclúster, que consiste en sumar las distancias al cuadrado entre cada observación y el centroide de su propio grupo. Para cada grupo k se calcula la Suma de Cuadrados dentro del su propio grupo (WGSS) mediante la Ecuación 11.

$$WGSS_k = \sum_{i=1}^{n_k} \|X_{ik} - C_k\|^2 \quad (11)$$

Donde:

- n_k : Número de observaciones (puntos) en el grupo k .
- X_{ik} : La observación número i del grupo k .
- C_k : Centroide del grupo k .

Luego se suma todas las sumas de cuadrados dentro del grupo individual:

$$WGSS = \sum_{k=1}^K WGSS_k$$

Donde:

- $WGSS_k$: Suma de cuadrados dentro del grupo " k ".
- K : Cantidad de grupos.

El tercer paso es el cálculo del Índice Calinski – Harabasz, la fórmula para su cálculo se muestra en la Ecuación 12

$$CH = \frac{BGSS}{WGSS} \times \frac{N - K}{K - 1} \quad (12)$$

Donde:

- *BGSS*: Suma de Cuadrados entre grupos.
- *WGSS*: Suma de Cuadrados dentro del grupo.
- *N*: Número total de observaciones.
- *K*: Número total de grupos.

Interpretación del Índice Calinski – Harabasz:

Calinski – Harabasz puede tomar valores de 0 a más infinito.

- El índice no tiene un rango específico, y la interpretación de qué tan "bueno" es un valor depende del contexto y de comparaciones con otros resultados de agrupamiento.
- Un valor más alto del índice de Calinski-Harabasz indica una mejor separación entre los clústeres.

A continuación, se muestra la métrica que se usará para la evaluación del algoritmo de Genético adaptado al Problema del Agente Viajero:

1. Convergencia de la Función Fitness

El Grado de Convergencia de un Algoritmo Genético (AG) o Algoritmo Memético (AM) se calcula mediante la variación de la función Fitness a lo largo de las generaciones del AG o AM. La Ecuación 13 permite calcular la convergencia de una función Fitness entre 2 generaciones.

$$\text{Grado de Convergencia} = \frac{\text{Fitness}_{\text{anterior}} - \text{Fitness}_{\text{actual}}}{\text{Fitness}_{\text{anterior}}} \quad (13)$$

Donde:

- $\text{Fitness}_{\text{anterior}}$: Función Fitness de la generación anterior.
- $\text{Fitness}_{\text{actual}}$: Función Fitness de la generación actual.

En el caso del presente estudio, al tomar la función Fitness el valor de la distancia de la ruta de patrullaje, el Grado de Convergencia tomará a esta última (distancia de la ruta de patrullaje) para el cálculo del Grado de Convergencia. A continuación, se presenta la Ecuación 14, la cual es la adaptación de la Ecuación 13.

$$\text{Grado de Convergencia} = \frac{\text{Distancia}_{\text{anterior}} - \text{Distancia}_{\text{actual}}}{\text{Distancia}_{\text{anterior}}} \quad (14)$$

Donde:

- $\text{Distancia}_{\text{anterior}}$: Distancia Mínima de Patrullaje de la generación anterior.
- $\text{Distancia}_{\text{actual}}$: Distancia Mínima de Patrullaje de la generación actual.

1.7. ALCANCE Y LIMITACIONES DE LA TESIS

La presente tesis **se centra** en las siguientes consideraciones:

- Se considera la elaboración de un camino de patrullaje que reduzca la distancia entre las zonas críticas de delincuencia.
- La determinación de las zonas crítica de delincuencia será en base a la densidad delictiva.
- El ámbito geográfico será los delitos cometidos dentro de la jurisdicción de Pueblo Libre.

- El ámbito criminológico involucra las denuncias contra el patrimonio (robo – hurto) registrado 01/01/2022 al 15/12/2022.

Dentro de los aspectos que **no se enmarca la tesis** se encuentran:

- La optimización de la ruta de patrullaje en la zona crítica de delincuencia.
- Despliegue de oficiales de serenazgo a pie en las zonas críticas de delincuencia.
- Consideración de áreas de patrullaje policial por sectores.
- Establecimiento de horarios de patrullaje de las zonas críticas de delincuencia.
- El análisis de datos criminológicos en tiempo real.

1.8. JUSTIFICACIÓN DEL ESTUDIO

La presente investigación de tesis proveerá beneficios tanto sociales como económicos. En el ámbito social, mejorará el grado de vigilancia de las zonas delictivas de un distrito, ya que optimizará las distancias recorridas, asimismo, establece metodologías de patrullaje que han mostrado ser efectivos en la prevención de actividades delictivas.

En el ámbito económico, enfocará la asignación de recursos de patrullaje (autos, motos, camionetas) en las zonas en donde el crimen está altamente concentrado y, por otro lado, reducirá el camino total entre las zonas de alto índice de criminalidad.

1.9. METODOLOGÍA DE INVESTIGACIÓN

1.9.1. Diseño y Nivel de la Investigación

En este trabajo de tesis se emplea un diseño preexperimental de tipo preprueba - posprueba con un único grupo. Se usará un caso de estudio, el cual

tendrá el ámbito de un distrito. El nivel de la investigación es **explicativo**, ya que según Arias (2012), “la investigación explicativa se enfoca en indagar las razones detrás de los eventos al establecer conexiones de causa y efecto, explorando tanto las posibles causas como las consecuencias.”

1.9.2. Enfoque de la Investigación

La investigación desarrolla el **enfoque cuantitativo**, ya que es el que más se adapta a las características de la investigación a realizar. Asimismo, se utilizará un caso de estudio para la construcción de la tesis con ámbito distrital. El enfoque cuantitativo es utilizado cuando “queremos estimar las magnitudes u ocurrencia de los fenómenos y probar hipótesis.” (Hernández Sampieri y Mendoza Torres, 2020).

1.9.3. Población y Muestra

La población es 692 denuncias policiales, el cual es el conjunto de denuncias policiales por robo o hurto en Pueblo Libre del 01/01/2022 al 15/12/2022 ocurridos en los 8 sectores del distrito.

La muestra es la misma que la población; es decir, 692 denuncias por robo o hurto ocurridos en Pueblo Libre del 01/01/2022 al 15/12/2022.

1.9.4. Técnica e Instrumento de recolección de datos

1.9.4.1. Técnicas de recolección de datos

Análisis Documental. La técnica de recolección de datos que se utilizará en la investigación será el del Análisis Documental, ya que se utilizarán varios documentos del Estado, Gobiernos Locales y entidades públicas de datos (INEI). Se filtrará y extraerá datos de las coordenadas geográficas de las denuncias policiales por robo y hurto del distrito, en este caso el distrito que será el caso de estudio es Pueblo Libre. Estos datos se encuentran en el PADSC de Pueblo Libre 2023, DATACRIM (INEI) y el Plan Nacional de Seguridad Ciudadana 2019 – 2023. De

acuerdo a Hernández y Mendoza (2020), la recolección de datos “implica elaborar un plan detallado de procedimientos que nos conduzcan a reunir datos con un propósito específico” (p.198).

1.9.4.2. Instrumento de recolección de datos

Fuentes Gubernamentales. Se hizo requerimiento de los datos delictivos de Pueblo Libre al INEI, así como la extracción de los datos delictivos del Observatorio de Seguridad Ciudadana implementado por la Municipalidad de Pueblo Libre.

Ficha de Registro. Se usará la **ficha de registro** como instrumento para la recolección de datos. Los datos recolectados serán de tipo secundario, debido a que “son registros escritos que proceden también de un contacto con la práctica, pero que ya han sido recogidos y muchas veces procesados por otros investigadores” (Sabino, 1992, p.115).

1.9.4.3. Herramientas para el procesamiento de datos

Se usa principalmente la herramienta Jupyter Notebook la cual es una “aplicación web original para crear y compartir documentos computacionales.” (Project Jupyter, s.f.) debido a que es una herramienta especializada en el manejo, visualización y análisis; así como, al estar embebida del lenguaje de programación Python permite usar librerías para utilizar algoritmos de aprendizaje de máquina y algoritmos heurísticos. Dentro de las librerías de Python, se utilizarán las de Pandas, NumPy, Matplotlib (Para el manejo de DataFrames, el cálculo matemático y el uso de gráficas respectivamente). También se utilizará la Librería de SkLearn que provee varias funciones de preprocesamiento de datos, algoritmos de aprendizaje no supervisado, etc.

1.9.5. Metodología de Análisis de Datos

La metodología a utilizar en la presente investigación es el Proceso Estándar Interindustrial para la Minería de Datos, más conocido como CRISP-DM, por sus siglas en inglés. Shearer (2000) define a la metodología CRISP-DM: “una metodología integral de minería de datos y un modelo de proceso que proporciona a cualquier persona, desde principiantes hasta expertos en minería de datos, un plan completo para llevar a cabo un proyecto de minería de datos”, en este sentido, aporta una robusta metodología para el análisis de los datos geográficos de la delincuencia.

La metodología CRISP-DM consta de 6 fases las cuales se detallan a continuación:

1. Comprensión del Negocio

La primera etapa se enfoca en comprender los objetivos del proyecto desde una perspectiva empresarial, traduciendo este entendimiento en una definición del problema de minería de datos, seguido por la elaboración de un plan preliminar diseñado para alcanzar dichos objetivos (Shearer, 2000).

2. Comprensión de los Datos

La segunda fase consiste en la comprensión de los datos, comenzando con el conjunto de datos inicial. Se procede a familiarizarse con los datos, identificar problemas de calidad de los mismos, y descubrir ideas iniciales a partir de la información disponible (Shearer, 2000).

3. Preparación de los Datos

La tercera etapa abarca la preparación de los datos, que incluye "todas las actividades necesarias para construir el conjunto de datos final o los datos que se introducirán en las herramientas de modelado" (Shearer, 2000).

4. Modelado

En la cuarta fase, el modelado, se seleccionan y aplican diversas técnicas de modelado (algoritmos), calibrando sus parámetros con valores óptimos.

5. Evaluación

La quinta fase, la evaluación, implica un análisis exhaustivo del modelo y una revisión de su construcción para garantizar que cumpla adecuadamente con los objetivos empresariales. Al final de esta etapa, "el líder del proyecto debe decidir exactamente cómo utilizar los resultados de la minería de datos" (Shearer, 2000).

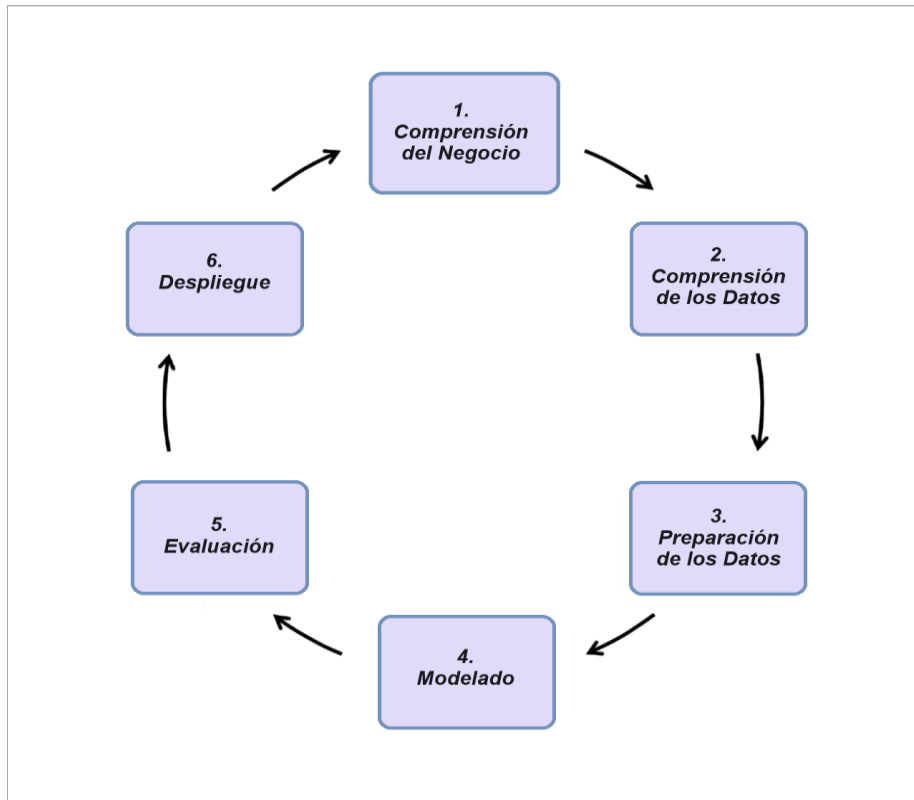
6. Despliegue

Finalmente, en la sexta etapa, el despliegue, el conocimiento adquirido se organiza y presenta de manera que la organización pueda utilizarlo. El resultado puede variar desde la generación de un informe hasta la implementación de un proceso de minería de datos repetible en toda la organización. (Shearer, 2000).

En la Figura 3 se presenta un esquema de las 6 fases de la metodología CRISP-DM.

Figura 3

Esquema del Marco de Trabajo CRISP-DM



Nota. Adaptado de "The CRISP-DM model: the new blueprint for data mining" (p. 14),

C. Shearer, 2000.

1.9.6. Aspectos Éticos

La presente investigación utiliza como fuente los datos las coordenadas geográficas de los delitos cometidos en un distrito. Estos datos, son de acceso público y no involucran información sensible de personas o empresas, por lo que el riesgo respecto a la seguridad de la información es bajo.

CAPÍTULO II

MARCOS TEÓRICO Y CONCEPTUAL

2.1. ANTECEDENTES DE INVESTIGACIÓN

2.1.1. Artículo 1

“Predictive police patrolling to target hotspots and cover response demand”

En este artículo, Leigh et al. (2017) estudian el problema del planeamiento de caminos de patrullaje para la respuesta de incidentes en Leicester. Se menciona que el patrullaje policial tiene el potencial de disuadir el crimen cuando está dirigido a las áreas correctas (áreas críticas de delincuencia), así como la capacidad de posicionar policías con acceso a dichas áreas para anticiparse a cualquier incidente delictivo que surja. Con estas consideraciones, los autores presentan un algoritmo que dirige rutas de patrullaje en tiempo real enfocándose en las áreas de alta criminalidad mientras que también maximiza la cobertura de la demanda ante incidentes. Para la identificación y delimitación de los puntos críticos utiliza los métodos de Densidad de Kernel (KDE) y una función de maximización de cobertura de la demanda para el posicionamiento de policías. Para la que la actualización de las soluciones se dé a tiempo real utilizaron un enfoque heurístico llamado Búsqueda Tabú. En base a una Simulación Basada en Agentes, en los cuales utilizan a los oficiales de policía como agentes, se usa como base los mapas delictivos caracterizados por el método de KDE y se utiliza el 3% del área de la ciudad de Leicester como las zonas críticas. De

esta simulación, se obtuvo una reducción del muestran una disminución potencial de la delincuencia callejera en un 22% durante la simulación de 1 mes, lo cual traería una disminución en el tiempo de respuesta a incidentes.

De este artículo se tomará en consideración el modelo de determinación de las zonas críticas de delincuencia (en base a un algoritmo de agrupamiento de densidad) para el patrullaje.

2.1.2. Artículo 2

“Towards Optimal Police Patrol Routes with Genetic Algorithms”

En este artículo, Reis et al. (2006) presentan una herramienta de simulación de evolutiva basada en multiagentes (GAPatrol) que es utilizada por los gestores policiales en el diseño de efectivas rutas de patrullaje. En este artículo, se revisa la hibridación de Sistemas Multiagente (MAS) con algoritmos evolutivos (EA), de forma que combina sus propiedades positivas y complementarias. Asimismo, toma en cuenta otros estudios relacionados a la seguridad pública que usan los sistemas de simulación basados en multiagentes. La solución planteada es llamada GAPatrol, que tuvo el propósito de brindar respaldo en la toma de decisiones para la delimitación y caracterización de puntos de concentración de la delincuencia. Se realizaron algunos experimentos de simulación, considerando 2 escenarios sobre un entorno simulado que imita un barrio ubicado en Fortaleza, Brasil. Los meta-parámetros del algoritmo genético fueron 95% de tasa de cruce, 5% de tasa de mutación, tamaño de población de 30 individuos y un número máximo de iteraciones de 100. En el primer escenario, consideraron como punto de salida de los delincuentes en el centro de 4 cuadrantes; en cambio, en el segundo escenario, consideraron como punto de salida una única fuente geográfica. Como resultado, el artículo presentó los mejores conjuntos de rutas para ambos escenarios, junto con

los tipos de objetivos ordenados según su frecuencia de visita por parte de los delincuentes. Indican que, a mayor frecuencia del incidente, mayor probabilidad de que el objetivo se convierta en una zona crítica (punto caliente). Asimismo, indican que el GAPatrol asigna un equipo de patrulla permanente a las zonas críticas de la delincuencia o aparecen 2 o más veces en cualquier ruta. También, resaltan el hecho que el GAPatrol asignó más recursos a los objetivos que están relacionados con delincuentes principiantes (plazas, ventas de droga y loterías) lo que impediría que los delincuentes se conviertan en expertos. Finalmente, indicaron que al final del proceso evolutivo se determinó certeramente las zonas de críticas de delincuencia, centrándose en las zonas más importantes. Finalmente, su principal conclusión que el sistema evolutivo es eficaz en este tipo de simulación para la delimitación de las zonas delictivas; así como, la determinación de las rutas que deben tomar las patrullas. Los autores mencionaron describir más exhaustivamente las propiedades y características de los mejores conjuntos de rutas encontradas por el GAPatrol.

Respecto a este artículo, considero que es un enfoque novedoso para la determinación de las rutas de patrullaje y la delimitación de las zonas delictivas, sin embargo, no trabaja con un conjunto de datos reales, lo que puede ser un inconveniente a la hora de aplicar los resultados en los escenarios reales de seguridad pública.

2.1.3. Artículo 3

“Improving the Creation of Hot Spot Policing Patrol Routes: Comparing Cognitive Heuristic Performance to an Automated Spatial Computation Approach”

En este artículo, Chainey et al. (2021) presentan el problema de la creación de rutas de patrullaje de puntos calientes, el cual se menciona que es principalmente un proceso manual que involucra utilizar los análisis de patrones espaciales de la

delincuencia la cual ayuda en la determinación de las zonas delictivas y trazar las rutas que los agentes de policía deben patrullar. En este artículo, se cuestiona el trabajo de los analistas policiales junto con el análisis académico para la determinación de las zonas delictivas. Resalta que la aplicación del patrullaje de zonas delictivas en América Latina, especialmente en Brasil, la cual indica que es limitada debido a la poca capacitación y acceso a Sistemas de Información Geográfica (SIG), requerido para realizar dicho análisis. La solución que plantearon los autores fue diseño de una aplicación de análisis espacial computacional que cree rutas de patrulla para la vigilancia de puntos calientes que supere las limitaciones de acceso/capacitación de los analistas policiales. Los autores describen al proceso manual de creación de rutas de patrullaje como un enfoque heurístico cognitivo; mientras que, el enfoque de cálculo espacial automatizado involucró el diseño de un algoritmo que identifica las concentraciones espaciales de delincuencia y a continuación, automatiza la creación de las rutas de las patrullas de zonas delictivas. En el artículo, se menciona que se utilizó 2 procesos de creación automática de rutas de patrulla de zonas delictivas que se diferenciaban en la forma en que decidían tomar los segmentos de calles con mayores niveles de delincuencia, llamados HotStar y HotSee. Las ciudades en las cuales hicieron el estudio fueron Florianópolis y Joinville. Tomaron los datos georreferenciados de robos a peatones del 1/02/2019 al 31/01/2020, los cuales estaban referenciados a segmentos de calle. Los indicadores para comparar los resultados fueron el número de delitos cometidos previamente en las rutas (W), la proporción de rutas de patrulla que eran segmentos calientes anteriormente (HLF), el Índice de Densidad Delictiva (IDC) y la proporción de la longitud total de todas las rutas que no se repiten en todas las rutas (NREF). Como resultado, los autores obtuvieron niveles críticos de concentración delictiva para ambos enfoques. Para la métrica W , entre los enfoques automatizados, HotStar

creó rutas que contenían más delitos que HotSee. Los enfoques computacionales incluyeron en sus rutas un 19% y 44% de delitos para las ciudades de Florianópolis y Joinville respectivamente. También se mejoró el indicador HLF; en Joinville, el indicador HLF llegó a 42%(HotStar) y 49% (HotSee) mientras que el enfoque manual logró un 35%. Los valores de CDI también mostraron mayor densidad en las rutas de patrulla, por ejemplo, en Joinville eran 21 más calientes, mientras que el enfoque manual eran 14 más calientes. El indicador NREF, también resultó mejor para los enfoques computacionales que el enfoque manual. En general, los enfoques computacionales automatizados superaron al enfoque heurístico cognitivo en cada una de las medidas que se utilizaron para comparar las rutas de patrulla que se crearon. También, el tiempo empleado para generar las rutas de patrulla fue significativamente menor para los enfoques computacionales, los cuales demoraron minutos comparados con los 3 días que demoró el enfoque manual. Finalmente, concluyeron que a pesar que los enfoques computacionales mejoran la creación de rutas de patrullaje en las zonas delictivas, éstas no consideran los conocimientos locales de los agentes de policía sobre el entorno y paisaje a patrullar. Indican que las rutas producidas por los enfoques computacionales deberían complementarse con un perfeccionamiento manual tras una visita a los lugares de patrullaje propuesto. Respecto a este artículo, considero que desarrolla un enfoque innovador en la generación de rutas de patrullaje de zonas críticas de delincuencia, sin embargo, considero que el factor de los agentes de policía es importante para la orientación del patrullaje en zonas pequeñas (menor a 1km), por lo que el trabajo de patrullaje dentro de una zona caliente debería tener en cuenta el contexto situacional de esas zonas.

Se tomará como aporte, la hipótesis de que el estudio de puntos de delitos mejora significativamente el patrullaje en zonas delictivas críticas, así como la definición del Índice de Densidad Delictiva.

2.1.4. Artículo 4

“Automatization Search for the Shortest Routes in the Transport Network Using the Floyd-Warshall Algorithm”

Sakharov et al. (2021) se centran en hallar las rutas más cortas para un grupo de buques que tienen el objetivo de alcanzar un objetivo con determinadas coordenadas dentro de un espacio delimitado. Menciona trabajos sobre algoritmos de Dijkstra, Bellman-Ford; así mismo, menciona que los métodos de inteligencia artificial basados en redes neuronales "no son menos eficaces", y finalmente los algoritmos genéticos. Recapitula que los problemas en los cuales se aplican estas herramientas, entre las cuales se encuentran la formación de rutas seguras en las zonas de separación con diferentes direcciones las cuales converjan en un lugar pequeño, la conexión entre todos los nodos de red utilizando los caminos de menor longitud, el diseño de las rutas de tráfico más fiables, formación de rutas de buques con menos tiempo de tránsito, entre otras. Emplea la aplicación de un algoritmo de redes conocido como Floyd-Warshall que se diferencia de los algoritmos Dijkstra y Bellman-Ford porque no necesita un determinado nodo fijo, indicando que es más general y proporciona la determinación simultánea de los caminos más cortos entre cualquier par de vértice de un grafo. El principal resultado es la presentación de las matrices finales de distancia y de secuencia de nodos producto de la ejecución del algoritmo Floyd-Warshall, así como la ejemplificación de una ruta más corta entre 2 buques. Entre las conclusiones de los autores se destacan la implementación de un algoritmo que automatice las operaciones de determinación de las rutas más cortas entre 2 buques, lo cual proporciona la elección de rutas marítimas lo que es extrapolable al

transporte terrestre y aéreo. Las ventajas del procedimiento propuesto de la aplicación del algoritmo Floyd-Warshall en la sencillez de implementación y la automatización de determinación de las rutas más cortas entre 2 nodos. Sin embargo, el algoritmo de Floyd-Warshall no se adecúa para problema de ruteo al ser un camino cerrado y no un camino abierto, por lo tanto, no se utilizará en la presente tesis.

2.1.5. Tesis 1

“Aplicación de la Metodología CRISP-DM a un Proyecto de Minería de Datos en el Entorno Universitario”

Galán (2015) realizó la tesis titulada “Aplicación de la Metodología CRISP-DM a un Proyecto de Minería de Datos en el Entorno Universitario”, en el cual el principal objetivo es el de realizar la aplicación de las etapas de la metodología de minería de datos CRISP-DM en los datos de los alumnos almacenados por la Universidad Carlos III para mejorar los servicios universitarios. Para lograr esto, revisa exhaustivamente cada una de las fases involucradas dentro del Marco de Trabajo CRISP-DM, las cuales son:

- Paso 1: Comprensión del Negocio
- Paso 2: Comprensión de los Datos
- Paso 3: Preparación de los Datos
- Paso 4: Modelado
- Paso 5: Evaluación
- Paso 6: Implantación.

Los modelos algorítmicos que usó el autor fueron modelos de regresión (Algoritmo de Aprendizaje de Máquina Supervisado) y utilizó la herramienta Oracle Data Mining para realizar el proceso de minería de datos, específicamente para los pasos 4, 5 y 6. En los resultados, indica que el modelo predictivo dio un porcentaje

de fiabilidad del 68% y 50% para el primer y segundo objetivo (Predicción del tiempo que un alumno en acabar la carrera y Predicción del promedio de notas de la carrera que un alumno); para el tercer objetivo (Predicción de la nota obtenida en una asignatura dada) se obtuvo un 12% de confianza predictiva, por lo que descartó el tercer modelo. En síntesis, el autor aceptó los modelos de los objetivos 1 y 2 y rechazó el modelo del objetivo 3. El principal aporte de la tesis a la presente investigación es que muestra un ejemplo claro de la aplicación de CRISP-DM utilizando algoritmos de Aprendizaje Automático.

2.1.6. Tesis 2

“Aplicación de la minería de datos para determinar la respuesta del dictamen de los clientes de una entidad financiera”

Guerra (2022) realizó la tesis titulada “Aplicación de la minería de datos para determinar la respuesta de dictamen de los clientes de una entidad financiera”, en la cual el principal objetivo fue de “diseñar, construir y evaluar 4 algoritmos para predecir la respuesta a reclamos de una entidad financiera, con la finalidad de disminuir la cantidad de reclamos a ser atendidos por los asesores de la entidad financiera, que fueron ingresados por los teleoperadores de empresa a petición de los clientes”. Para lograr dicho objetivo, Guerra (2022) utilizó las herramientas de minería de datos; asimismo, utilizó la metodología de minería de datos CRISP-DM, la cual le proveyó un marco de trabajo para poder desarrollar el proyecto de minería de datos. La herramienta que utilizó para la limpieza y el tratamiento de los datos fue Rstudio. Respeto a los resultados, Guerra encontró que el modelo predictivo reduciría de 147555 a 78522 la cantidad de reclamos ingresados para los meses de enero hasta junio, es decir una reducción del 46.78%. De la tesis analizada, se rescata la aplicación del marco de trabajo CRISP-DM en el proceso de minería de

datos en un determinado conjunto de datos; metodología que será utilizada en la presente tesis.

2.2. MARCO TEÓRICO

2.2.1. Patrullaje de Zonas Críticas de Delincuencia

El Patrullaje de Zonas Críticas de Delincuencia ha sido descrito por varios autores; por ejemplo, según Weisburd y Eck (2004) definen el "patrullaje de zonas críticas de delincuencia como un nuevo e importante enfoque para el control de la delincuencia que implica el patrullaje dirigido a los lugares específicos de una jurisdicción en los que se concentra la delincuencia.". Por otro lado, Reisig y Kane (2014) indican que el "patrullaje de zonas críticas de delincuencia es un enfoque de control de la delincuencia que incluye el concentrar recursos policiales en los lugares de alta criminalidad". Finalmente, Weisburd y Telep (2014) definen el "patrullaje de zonas críticas de delincuencia como el rango de respuestas policiales que todas comparten en común un enfoque de recursos en los lugares donde el crimen está altamente concentrado. Por lo tanto, de lo anterior, se puede afirmar que el patrullaje de zonas críticas de delincuencia es el "Enfoque de control de la delincuencia que incluye el patrullaje dirigido a lugares específicos donde el crimen está concentrado." (Weisburd et al., 2014)

2.2.1.1. Patrullaje Dirigido.

Según McKee (2023), el "patrullaje dirigido es una estrategia policial en la que los agentes patrullan zonas específicas para disuadir la delincuencia o responder rápidamente cuando se produce un delito". El patrullaje dirigido requiere que se definan las zonas específicas a patrullar (zonas críticas) en las que se produce la mayor cantidad de delitos en un espacio urbano.

Patrullar: Según Porto (2013), patrullar es "recorrer una cierta zona, por lo general para evitar que se cometan delitos o que se produzcan hechos con consecuencias negativas.". En este sentido, lo que se encontrar un recorrido óptimo para prevenir el delito, el cual tiende a concentrarse en zonas específicas.

Zona específica: Según Eck, et al. (2005), las "zonas específicas son el nivel más bajo de análisis donde ocurren crímenes". Se refiere a las coordenadas geográficas de los incidentes de delitos. Con respecto a las coordenadas geográficas, éstas están constituidas por latitud (la distancia en grados, minutos y segundos o decimales que hay con respecto al ecuador (paralelo 0°)) y longitud (la distancia en grados, minutos y segundos o decimales que hay con respecto al meridiano de Greenwich (meridiano 0°)).

2.2.1.2. Concentración del Crimen.

Según Weisburd (2015), la "concentración del crimen indica que en una unidad micro geográfica un rango de porcentajes se asocia con una proporción acumulativa de delincuencia". Por lo tanto, la concentración del crimen implica que un porcentaje que suele ir del 25% al 50% de los delitos de una zona determinada se concentra en lugares específicos en un espacio urbano.

Proporción acumulativa de delincuencia: Según Weisburd (2015), una "proporción acumulativa de delincuencia es un porcentaje de la delincuencia que ocurre en un espacio geográfico". Por ejemplo, Weisburd encontró que respecto a las denuncias provenientes de incidentes delictivos relacionados al tráfico representaron solo el 4.5% de los segmentos de calle de Seattle (EE.UU.), pero suponen el 44% de las denuncias en las intersecciones.

2.2.2. Aprendizaje de Máquina

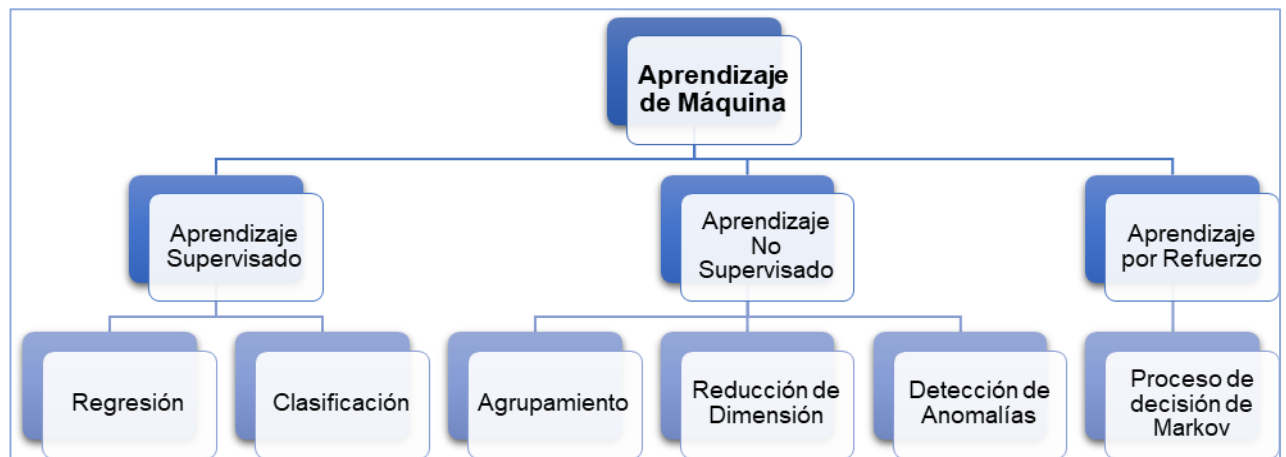
El término Aprendizaje de Máquina (Aprendizaje Automático o Machine Learning) fue acuñado por Arthur Samuel y se refiere al campo de estudio que da a

las computadoras la capacidad de aprender sin ser explícitamente programados (Samuel, 1969). Según Murphy (2012), define el Aprendizaje Automático como un “conjunto de métodos capaces de detectar automáticamente patrones en los datos y utilizarlos para predecir datos futuros o tomar otro tipo de decisiones en condiciones de incertidumbre”.

Los algoritmos de Aprendizaje de Máquina “pueden clasificarse en 3 grupos en función del resultado deseado y del tipo de entrada necesaria” (Swamynathan,2017). Estos 3 grupos son Aprendizaje Supervisado, Aprendizaje No supervisado y Aprendizaje por Refuerzo. En la Figura 4 se muestra un gráfico jerárquico con los 3 grupos de algoritmos de Aprendizaje de Máquina.

Figura 4

Tipos de Algoritmos de Aprendizaje de Máquina



Nota. Adaptado de “*Mastering Machine Learning with Python in Six Steps*”. (p. 67), por M. Swamynathan, 2017, Apress eBooks.

2.2.2.1. Aprendizaje No Supervisado

Es una familia de algoritmos de Aprendizaje de Máquina (Machine Learning). En el Aprendizaje No Supervisado “no hay un resultado conocido ni una guía que

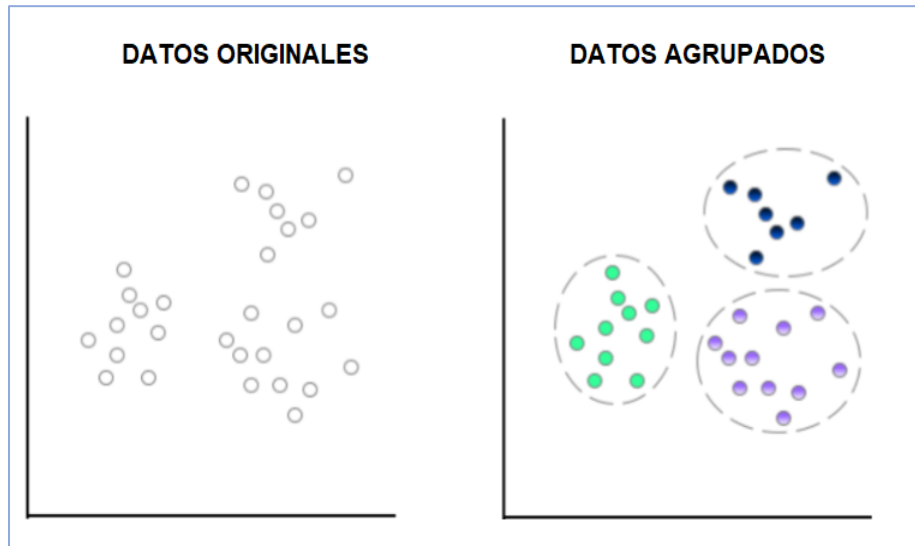
instruya al algoritmo, sino que simplemente se le muestran los datos de entrada y se le pide extraer conocimiento” (Müller & Guido, 2016).

Algoritmos de Agrupamiento. Los algoritmos de Agrupamiento (Clustering Algorithms) son un tipo de Algoritmos de Aprendizaje No Supervisado, en el cual “el objetivo clave es identificar grupos distintos (llamados “clusters” en inglés) basándose en alguna noción de similitud (semejanza) dentro de un conjunto de datos dado.” (Swamynathan, 2017). Dentro de estos Algoritmos de Agrupamiento se encuentran el Algoritmo K-Medias, DBSCAN (Agrupamiento Espacial Basado en Densidad de Aplicaciones con Ruido), Agrupamiento Aglomerativo, entre otros.

Algoritmo de Agrupamiento K-Medias. El Algoritmo K-Medias pertenece a los tipos de Algoritmos No Supervisados de Agrupamiento (observar Figura 4). Es una técnica popular para reducir la complejidad de los datos, que se utiliza para identificar grupos de puntos de datos sin conocimiento previo de las clases existentes. (Theobald, 2019). El agrupamiento de K-Medias se encarga de dividir o agrupar el conjunto de datos en un número k de grupos, siendo k el número de grupos (Theobald, 2019). En la Figura 5, se muestra un esquema que ejemplifica un agrupamiento obtenido por el algoritmo K-Medias con un k igual a 3.

Figura 5

Esquema del Algoritmo K-Medias



Nota. Extraído de *“Machine learning with Python: A Practical Beginners’ Guide”* (p. 61), por O. Theobald, 2019.

A continuación, se presenta los pasos para la aplicación del Algoritmo K-Medias:

1. Escoge el número de grupos (K) y obtener los datos de los puntos.
2. Seleccionar k puntos al azar del conjunto de datos como los centroides iniciales de cada clúster:
3. Asignar cada punto del conjunto de datos al grupo cuyo centroide esté más cerca.
4. Recalcular los centroides de cada grupo como la media de todos los puntos del grupo
5. Repetir los pasos 3 y 4 hasta que los centroides de los grupos ya no cambien o hasta que se alcance el número máximo de iteraciones.

Agrupamiento Espacial Basado en Densidad de Aplicaciones con Ruido

(DBSCAN). El algoritmo de Agrupamiento Espacial Basado en Densidad de Aplicaciones con Ruido, conocido por sus siglas en inglés como DBSCAN (Density - Based Spatial Clustering of Applications with Noise), fue presentado por Ester et al. en 1996. DBSCAN es un algoritmo que está diseñado para descubrir los clusters y el ruido en una base de datos espacial (Ester et al., 1996). Según Tan et al. (2019), DBSCAN es un “algoritmo de agrupamiento basado en densidad que produce un agrupamiento particional, en el que el número de grupos es determinado automáticamente por el algoritmo.” Por otro lado, los puntos ubicados en regiones de baja densidad son clasificados como ruido y no son agrupados en ningún grupo, por lo tanto, DBSCAN no produce una agrupación completa (Tan et al., 2019). Los 2 parámetros más importantes del algoritmo DBSCAN son los siguientes:

- ‘min_samples’: Número mínimo de puntos en una vecindad para considerar un punto central.
- ‘eps’: Distancia épsilon para la selección de puntos centrales.

El algoritmo DBSCAN define los grupos mediante la estimación de la densidad local. El proceso algorítmico se puede dividir en las siguientes etapas:

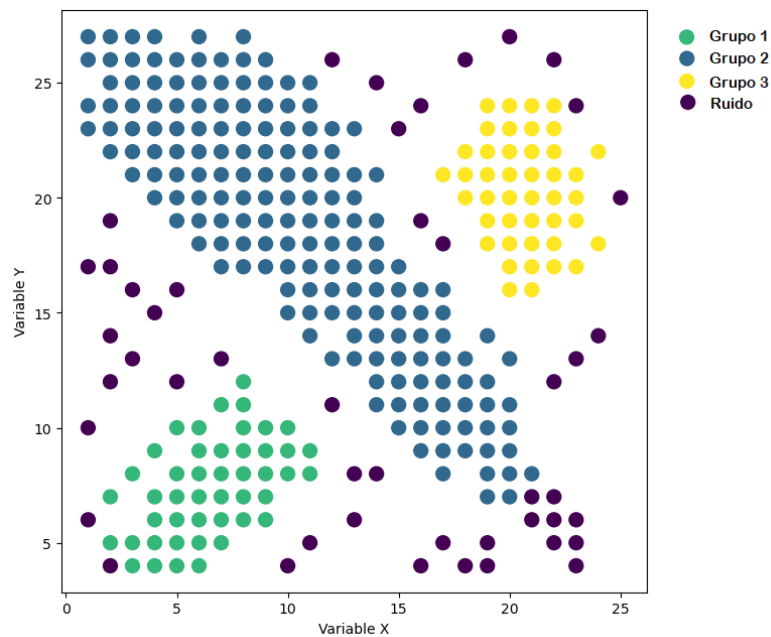
1. Para cada punto miramos el número de puntos adyacentes a una distancia máxima épsilon (ϵ), la cual es definida previamente.
2. Si un punto tiene al menos un cierto número de vecinos (MinPts), incluida ella misma, se considera una observación central. En este caso, se ha detectado una observación de alta densidad.
3. Todos los puntos en la vecindad de una observación central pertenecen al mismo grupo. Puede haber puntos centrales cercanos entre sí. Por lo tanto, de un paso a otro, se obtiene una larga secuencia de observaciones centrales que constituyen un único grupo.

4. Cualquier punto que no sea un punto central y que no tenga ningún punto central en su vecindad se considera ruido.

En la Figura 6, se muestra un ejemplo de un Agrupamiento DBSCAN.

Figura 6

Esquema de Agrupamiento mediante DBSCAN.



Nota. Extraído de “Identifica Clusters con DBSCAN: Algoritmo paso a paso e implementación con Python”, por Gutiérrez-García, 2022.

Agrupamiento Espacial Jerárquico Basado en Densidad de Aplicaciones con Ruido (HDBSCAN). El algoritmo de Agrupamiento Espacial Jerárquico Basado en Densidad de Aplicaciones con Ruido, conocido por sus siglas en inglés como HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise), fue presentado por Campello, R. et al. en el 2013. Este algoritmo se caracteriza por no necesitar la especificación previa del número de grupos, generar grupos de diferentes formas y tamaños e identifica puntos de ruido. (Campello et al., 2013). Dentro de sus parámetros principales se encuentran:

- *min_cluster_size*: Representa el tamaño mínimo del grupo.
- *min_samples*: Número mínimo de puntos en una vecindad para considerar un punto central.
- *cluster_selection_epsilon*: Distancia épsilon para la selección de grupos.

2.2.3. Algoritmos Heurísticos

Los algoritmos heurísticos involucran estrategias que utilizan información de fácil acceso, aunque poco aplicable, para controlar los procesos de resolución de problemas en los seres humanos y las máquinas (Pearl, 1984). Por otro lado, Russell y Norvig (2010) indicaron que los algoritmos heurísticos utilizan información que va más allá de la definición del problema para utilizar conocimientos específicos del mismo lo que permite encontrar soluciones más rápidamente que una estrategia no informada. En otras palabras, estos algoritmos se caracterizan por usar heurísticas (técnicas de indagación y descubrimiento) para llegar a resultados buenos o aproximados a la solución óptima de forma más rápida que las formas tradicionales las cuales brindan soluciones exactas.

Heurística

El término heurístico(a) tiene como significado “Técnica de la indagación y del descubrimiento” (Real Academia Española, s.f., 2da acepción), también tiene como significado “Manera de buscar la solución de un problema mediante métodos no rigurosos, como por tanteo, reglas empíricas, etc.” (Real Academia Española, s.f., 4ta acepción).

Con los anteriores significados del término heurístico(a), se puede formular que una heurística es una técnica de indagación y descubrimiento de soluciones a un problema mediante métodos no rigurosos (tanteo, reglas empíricas, etc.).

2.2.3.1. Algoritmo Genético

Los Algoritmos Genéticos son algoritmos de búsqueda basados en la teoría de la selección natural y la genética (Goldberg, 1989). Estos algoritmos combinan la supervivencia del más apto entre las estructuras de cadenas con un intercambio de información estructurado pero aleatorio para formar un algoritmo de búsqueda. (Goldberg, 1989). Los Algoritmos Genéticos se han considerado en gran medida como métodos de optimización de funciones (Eiben y Smith, 2015).

Genética

Genético(a) se refiere a todo lo relacionado con los genes y la herencia biológica, incluyendo la transmisión de características hereditarias de una generación a otra. (Hartl y Ruvolo, 2011).

2.2.3.2. Algoritmo Memético

Los Algoritmos Meméticos (MA) son “metaheurísticas basadas en poblaciones compuestas de un enfoque evolutivo y un conjunto de algoritmos de búsqueda local, los cuales son activados dentro del ciclo de generaciones del marco externo.” (Neri et al., 2012). Los Algoritmos de Búsqueda Local, según Levitin (2012):

Comienzan con un recorrido inicial, por ejemplo, construido aleatoriamente o mediante un algoritmo de aproximación. En cada iteración, el algoritmo explora un vecindario alrededor del recorrido actual sustituyendo algunas aristas del recorrido actual por otras aristas. Si los cambios producen un recorrido más corto, el algoritmo lo convierte en el recorrido actual y continúa explorando su vecindario de la misma manera; de lo contrario, el recorrido actual se devuelve como salida del algoritmo y éste se detiene. (p. 449)

Los Algoritmos de Búsqueda Local que se suelen usar dentro de los Algoritmos Meméticos suelen los Algoritmos 2-Opt y 3-Opt.

Algoritmo 2-Opt. El algoritmo de Búsqueda Local 2-Opt “funciona eliminando un par de aristas no adyacentes en un recorrido y volviendo a conectar sus puntos finales mediante otro par de aristas para obtener otro recorrido” (Levitin, 2012).

2.3. MARCO CONCEPTUAL

2.3.1. Patrullaje de Zonas Críticas

El patrullaje de zonas críticas es el “Enfoque de control de la delincuencia que incluye el patrullaje dirigido a lugares específicos donde el crimen está concentrado.” (Reisig y Kane, 2014; Weisburd y Telep, 2014; Weisburd y Eck, 2004).

2.3.2. Densidad Delictiva

La Densidad Delictiva se refiere a la cantidad de delitos reportados en un área determinada (en el caso de la investigación, se considera el área del distrito), generalmente medida en términos de delitos por unidad de superficie (por ejemplo: delitos / kilómetro cuadrado).

2.3.3. Zona Crítica

Según Sherman y Weisburd (1995), las “Zonas Críticas de Delincuencia son pequeñas agrupaciones de direcciones con frecuentes llamadas por delitos graves y numerosas llamadas por delitos leves”; también, según Sherman (1995) las Zonas Críticas de Delincuencia son “pequeños lugares en los que la delincuencia es tan frecuente que es altamente predecible, al menos durante un período de un año”. Estos pequeños lugares, en el contexto criminológico son “localizaciones específicas dentro de un entorno social más amplio; pueden ser tan pequeños como la zona contigua a un cajero automático o tan grandes como la fachada de una manzana, un centro comercial o un edificio de departamentos.” (Eck & Weisburd, 2015).

2.3.4. Localización Específica

Una "localización específica" se refiere a un lugar o posición concreta y detallada en un espacio geográfico. Una localización específica es un lugar en particular donde algo ocurre o se ubica. Una localización específica se caracteriza por su alto nivel de precisión para identificar un lugar o posición en el espacio geográfico. En el contexto de seguridad ciudadana, una localización específica se refiere a una dirección exacta, coordenadas geográficas o una descripción detallada de un lugar donde se ha producido un incidente delictivo.

2.3.5. Coordenada Geográfica

El término "coordenadas geográficas" generalmente se refiere a valores numéricos utilizados para representar la latitud y longitud exacta de un punto en la superficie de la Tierra. (Robinson et al.,1995) Estas coordenadas, expresadas en grados, minutos y segundos, determinan la ubicación de un lugar con respecto al ecuador (latitud) y al meridiano de Greenwich (longitud). En pocas palabras, las "coordenadas geográficas", tal como se definen en los principios de la cartografía, son los valores angulares (latitud y longitud) necesarios para determinar con precisión una ubicación en la superficie de la Tierra.

2.3.6. Paralelos

Los paralelos son un conjunto de líneas que van de este-oeste equidistantes las cuales fueron propuestas por el astrónomo y matemático Hiparco (Kimerling et al., 2016). A estas líneas, añadió un conjunto de líneas norte-sur llamadas meridianos que están igualmente espaciadas en el ecuador y convergen en los polos norte y sur.

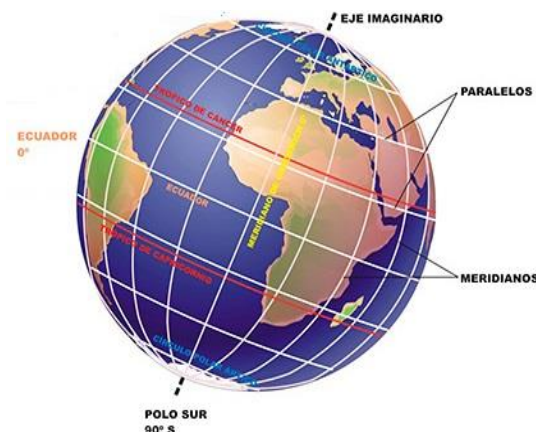
2.3.7. Meridianos

Los meridianos son un conjunto de líneas norte-sur que están igualmente espaciadas en el ecuador y convergen en los polos norte y sur. (Kimerling et al.,

2016). El Meridiano de Greenwich es el cual “se cuentan los grados de longitud” (García et al. ,s.f.). La Figura 7 muestra el globo terráqueo y paralelos y meridianos.

Figura 7

Paralelos y Meridianos



Nota. Bermúdez, J (s. f.). Currículum Nacional

2.3.8. Longitud

La longitud es el ángulo, medido en el ecuador, entre la intersección del meridiano de referencia (Meridiano de Greenwich) y el meridiano del punto de interés se cruza con el ecuador (Kimerling et al., 2016). El valor numérico de la longitud va de 0° a 180° al este y al oeste del meridiano de referencia. Al igual que con la latitud sur, la longitud oeste también puede indicarse con un signo menos, pero no se utiliza un signo más para la longitud este (Kimerling et al., 2016). En la presente investigación, la longitud es un atributo que representa el eje X (eje de las ordenadas) y posee valores negativos porque el área de estudio —Pueblo Libre, Lima — se encuentra en longitudes oeste.

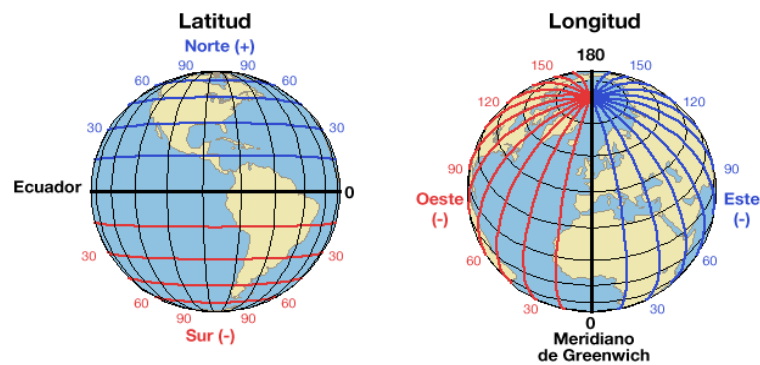
2.3.9. Latitud

La latitud en la Tierra esférica es la distancia angular norte-sur desde el ecuador hasta el lugar de interés (Kimerling et al., 2016). El rango numérico de la latitud va de 0° en el ecuador a 90° en los polos. Las letras N y S, como 48° N, se utilizan para indicar las latitudes norte y sur, respectivamente. En lugar de la letra S,

se puede ver un signo menos (-) para las latitudes sur; sin embargo, no se utiliza un signo más (+) para las latitudes norte (Kimerling et al., 2016). En la presente investigación, es un atributo que representa el eje Y (eje de las ordenadas) y tendrá valores negativos porque el área de estudio —Pueblo Libre, Lima— se encuentra en latitudes sur. En la Figura 8 se ilustran los conceptos de latitud y longitud descritos anteriormente.

Figura 8

Latitud y Longitud



Nota. Extraído de “Las Coordenadas”, Junta de Andalucía (s.f.)

CAPÍTULO III

DESARROLLO DEL TRABAJO DE INVESTIGACIÓN

El Marco de Trabajo utilizado para la presente tesis es el Proceso Estándar Interindustrial para la Minería de Datos, más conocido como CRISP-DM, por sus siglas en inglés. El Marco de Trabajo CRISP-DM tiene 6 fases que se aplicaron con la guía del artículo presentado por Colin Shearer (2000).

3.1. COMPRENSIÓN DEL NEGOCIO

3.1.1. Determinar los Objetivos de la Seguridad Ciudadana

Dentro de los objetivos que se plantea la Gerencia de Seguridad Ciudadana de Pueblo Libre están los indicados por el Plan Nacional de Seguridad Ciudadana (2019-2023) en los cuales se introduce nuevos enfoques para la gestión de las amenazas que afectan la seguridad ciudadana (PADSC, 2023). En dicho plan, se establecen 5 objetivos estratégicos:

1. Reducir los homicidios a nivel nacional;
2. Reducir el número de fallecidos y heridos en accidentes de tránsito;
3. Reducir la violencia contra la mujer e integrantes del grupo familiar;
- 4. Promover espacios libres de hurto y robo;**
5. Reducir la victimización cometida por bandas criminales hacia personas naturales y jurídicas.

Al ser la Seguridad Ciudadana un problema sumamente amplio y complejo, solo se considerará la consideración del objetivo N°4: Promover espacios libres de hurto y robo. En este sentido, una pregunta que se plantea al problema de seguridad ciudadana es: ¿Cómo reducir la distancia de camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos?

En alineamiento con este planteamiento, una medida de éxito es reducir en un 10% la distancia del camino de patrullaje de las zonas críticas de delincuencia.

- Evaluación de la Situación
- Determinar los objetivos de la minería de datos
- Producir el plan del proyecto

3.1.2. Evaluación de la Situación

Para la ejecución del proyecto de minería de datos, se tiene acceso a los datos del registro de las 692 denuncias georreferenciadas de robo y hurto ocurridos en Pueblo Libre del año 2022 provenientes del INEI y el Observatorio de Seguridad Ciudadana de Pueblo Libre. Este conjunto de datos contiene atributos clave para el trabajo de minería de datos como —por ejemplo— longitud (Eje X), latitud (Eje Y), Ubigeo, ID. La base de datos se encuentra en formato Excel (.xlsx).

Inventario de Recursos

Referido al software se dispone de la plataforma web para análisis de datos Jupyter Notebook, la cual proporciona herramientas de almacenamiento, selección, preprocesamiento de base de datos. También permite la generación de gráficas y modelado de algoritmos de distinta naturaleza (Aprendizaje de Máquina, A. Genéticos, etc.)

Los recursos de hardware que se disponen son los siguientes:

- Computadora Portátil (Redacción del Informe)

- Marca: Hewlett-Packard
- Procesador: Intel(R) Core (TM) i3-1125G4 a 2.00GHz
- RAM instalada: 8.00 GB
- Tipo de sistema: Sistema operativo de 64 bits.
- Sistema Operativo: Windows 11 Home Single Language
- Computadora Personal (Minería de Datos)
 - Procesador: Intel(R) Core (TM) i5-9400F a 2.90GHz
 - RAM instalada: 16.00 GB
 - Tipo de sistema: Sistema operativo de 64 bits.
 - Sistema Operativo: Windows 10 Pro
 - Tarjeta Gráfica: NVIDIA GeForce GTX 1650 4GB
 - Monitor: LG 22MK400 de 22"

Supuestos y restricciones

Debido a que algunos distritos con mayor nivel delictivo no cuentan con los datos de los delitos debidamente georreferenciados, no se pudo analizar dichos distritos.

Riesgos y Contingencias

Los riesgos identificados en la implementación del proyecto de minería de datos se especifican en la Tabla 3, junto con sus respectivas medidas para mitigarlos:

Tabla 3

Riesgos y Medidas de Mitigación

Riesgo Identificado	Nivel de Impacto	Medidas de mitigación
Retraso en la obtención de la base de datos a analizar.	Medio	Solicitar con anticipación las bases de datos a las organizaciones públicas respectivas.
Datos con valores nulos u equívocos	Bajo	Realizar un proceso de limpieza y preprocesamiento de los datos.
Dificultad en la implementación de algoritmos.	Media	Buscar material bibliográfico de confiabilidad que apliquen algoritmos en ámbitos similares.

Nota. Elaboración Propia.

Terminología

Los términos del negocio se describen en el marco conceptual y los términos de minería de datos se enlistan en el Glosario del Proyecto de Minería de Datos (Anexo 1)

Costos y Beneficios

La generación del conjunto de datos no requiere un costo adicional, así como por parte del INEI ya que estos datos son recopilados con anterioridad en las comisarías de la jurisdicción distrital.

En relación a los beneficios, no se puede afirmar que los resultados que provengan de este proyecto generen un beneficio económico de una manera directa, sin embargo, al buscar reducirse la distancia del camino de patrullaje entre zonas críticas, sí puede repercutir indirectamente a los gastos asociados en el transporte

de los vehículos de patrullaje, así como incrementar la presencia del serenazgo en las zonas donde el hurto/robo se hace más presente.

3.1.3. Determinar los objetivos de la minería de datos

Los objetivos de la minería de datos en términos de la organización son los siguientes:

- Determinar las zonas críticas de delincuencia (agrupamientos o clusters) de Pueblo Libre dadas la latitud y longitud de las denuncias policiales de robo y hurto del año 2022.
- Identificar los lugares específicos (avenidas, coordenadas geográficas) que delimitan las zonas críticas de delincuencia de Pueblo Libre.
- Determinar el camino de recorrido mínimo del patrullaje de las zonas críticas de delincuencia de Pueblo Libre.
- Calcular la distancia del recorrido mínimo de patrullaje de las zonas críticas de delincuencia de Pueblo Libre.

Criterios de éxito

Desde el punto de vista de la minería de datos, se establece como criterios de éxito el resultado de reducir en un 10% la distancia de recorrido de patrullaje entre las zonas críticas de delincuencia de Pueblo Libre, así como la consideración de la densidad delictiva relativa mayor a 2 veces la Densidad Delictiva del Distrito.

3.1.4. Producir el plan del proyecto

El proyecto de minería de datos tendrá las siguientes etapas y duración:

- Fase 1: Análisis del conjunto de datos y los atributos de la base de datos.

Duración prevista: 1 semana.

- Fase 2: Preparación de los datos (incluye la selección, depuración y el ajuste de escala) para facilitar su uso en la minería de datos sobre ellos. Duración prevista: 3 semanas
- Fase 3: Selección y aplicación de técnicas de modelado sobre los datos recopilados. Duración prevista: 2 semanas.
- Fase 4: Análisis de los resultados obtenidos en la fase anterior. Duración prevista: 1 semana.
- Fase 5: Elaboración de informes que resuman los resultados según los objetivos comerciales y los criterios de éxito establecidos. Duración prevista: 1 semana.
- Etapa 6: Presentación de los resultados finales. Duración prevista: 1 semana.

Análisis inicial de herramientas y métodos

La herramienta de análisis de datos que se va a utilizar es Jupyter Notebook, como se indicó en el apartado de Herramientas para el procesamiento de datos (1.9.4.3), esta herramienta se adecua a las actividades descritas anteriormente en las 6 etapas del plan del proyecto. En cuanto a las técnicas de modelado para la extracción de patrones (conocimiento), Jupyter Notebook nos provee de las siguientes librerías:

- Pandas: Librería enfocada en el análisis de datos, que proporciona estructuras como DataFrames y Series, así como operaciones para manejar tablas numéricas.
- NumPy: Librería para la computación científica en Python. Provee el objeto de matriz multidimensional, objetos derivados (matrices) y una variedad de

rutinas para operaciones rápidas en matrices. El núcleo del paquete Python es el objeto ndarray (matriz con “n” dimensiones).

- Scikit - Learn: Librería que provee de numerosos algoritmos de Aprendizaje basada en el lenguaje Python. Incluye algoritmos de clasificación, regresión, agrupamiento, etc. Provee de varios módulos como: cluster (DBSCAN, KMedias), preprocessing (Preprocesamiento), neighbors (Vecinos más cercanos), metrics (Métricas de Evaluación de Algoritmos de Agrupamiento) entre otros.
- Matplotlib: Librería completa especializada en crear visualizaciones estáticas, animadas e interactivas en Python.
- Seaborn: Librería apoyada en Matplotlib que ofrece un entorno de alto nivel para crear gráficos estadísticos atractivos e informativos.
- Geopy: Librería de Python que simplifica el cálculo de distancias geográficas entre 2 puntos.

Para la evaluación de los algoritmos de aprendizaje no supervisado (algoritmos de agrupamiento), se utilizarán los índices Silueta, Davies-Bouldin y Calinski y Harabasz. Para la evaluación del algoritmo genético adaptado al Problema del Agente Viajero, medirá la Convergencia de la Función Fitness.

3.2. COMPRENSIÓN DE LOS DATOS

3.2.1. Recolección de datos iniciales

El conjunto de datos recolectados, es el registro de denuncias por robo y hurto en Pueblo Libre del año 2022. Consta de 692 registros de datos delictivos de Robo y Hurto. Este conjunto de datos, fue recibido en formato CSV (Valores separados por comas) y fue cargado a un Dataframe (Estructura de datos tabular) de nombre

'delitos'. El código aplicado se muestra en el Anexo 2. También se recolectó las coordenadas geográficas de los puntos que delimitan el distrito de Pueblo Libre, mediante la georreferenciación del mapa de Pueblo Libre en el software ArcGis Pro. Los atributos que se obtuvieron de los datos delictivos se detallan en la Tabla 4.

Tabla 4

Atributos del Registro de Datos Delictivos

Nombre del Atributo	Descripción del Atributo
Sector	Indica el sector en el cual ocurrió el hecho delictivo.
Día	Indica el día de la semana en el cual ocurrió el hecho delictivo.
Mes	Indica el mes en el cual ocurrió el hecho delictivo.
Dirección	Nombre de la Avenida, Jirón, Pasaje donde ocurrió el hecho delictivo
Tipo	Indica si el delito fue de Robo o Hurto
Objeto	Indica el objeto de valor que fue arrebatado en el hecho delictivo.
MedioUtilizado	Indica la Herramienta (Arma) utilizada en la comisión del hecho delictivo.
Lugar	Indica el tipo de ambiente en el cual se desarrolló el hecho delictivo
MétodoDeHuida	Método utilizado en la fuga de los delincuentes.
Latitud	Distancia angular norte-sur desde el ecuador hasta el lugar del hecho delictivo.
Longitud	Distancia angular, medido en el ecuador, entre la intersección del meridiano Greenwich y el meridiano del lugar del hecho delictivo.

Nota. Conjunto de datos obtenido del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

3.2.2. Descripción de los datos

En la Tabla 5, se muestra la clase de dato correspondiente a cada uno de los atributos en los registros delictivos:

Tabla 5

Tipo de Dato de los Atributos del Conjunto de Datos

Nombre del Atributo	Tipo de Dato
Sector	Cadena
Día	Cadena
Mes	Cadena
Dirección	Cadena
Tipo	Cadena
Objeto	Cadena
MedioUtilizado	Cadena
Lugar	Cadena
MétodoDeHuida	Cadena
Latitud	Flotante
Longitud	Flotante

Nota. Datos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

A continuación, se detalla las características de cada atributo:

- Sector: Variable Categórica. Indica en qué sector del distrito se cometió el hecho delictivo. Toma 8 valores: Sector 1, Sector 2, ..., Sector 8.
- Día: Variable Categórica. Indica el día de la semana se cometió el hecho delictivo.
- Mes: Variable Categórica. Indica el mes en el cual se cometió el hecho delictivo.
- Dirección: Indica la dirección exacta del delito cometido.
- Tipo: Variable Categórica. Indica si el delito fue 'Robo' o 'Hurto'.

- Objeto: Variable Categórica. Entre los objetos robados/hurtados se encuentran Celular, Autopartes, Accesorios o Herramientas, Vehículo Mayor, Dinero, Vehículo Menor, Vehículo Menor, Computadoras y Robo de Mascotas.
- Medio Utilizado: Variable Categórica.
- Método De Huida: Variable Categórica.
- Latitud: Variable Continua. Indica la latitud en la cual se cometió el hecho delictivo. Tiene valores negativos.
- Longitud: Variable Continua. Indica la longitud en la cual se cometió el hecho delictivo. Tiene valores negativos.

Seguidamente, en la Tabla 6 se describe el tipo de dato de los atributos (Longitud y Latitud) de las coordenadas que conforman el límite distrital:

Tabla 6

Tipo de Datos de las Coordenadas Limítrofes de Pueblo Libre

Nombre del Atributo	Tipo de Dato
Longitud	Cadena
Latitud	Cadena

Nota. Conjunto de datos obtenido del software ArcGIS Pro. Elaboración Propia.

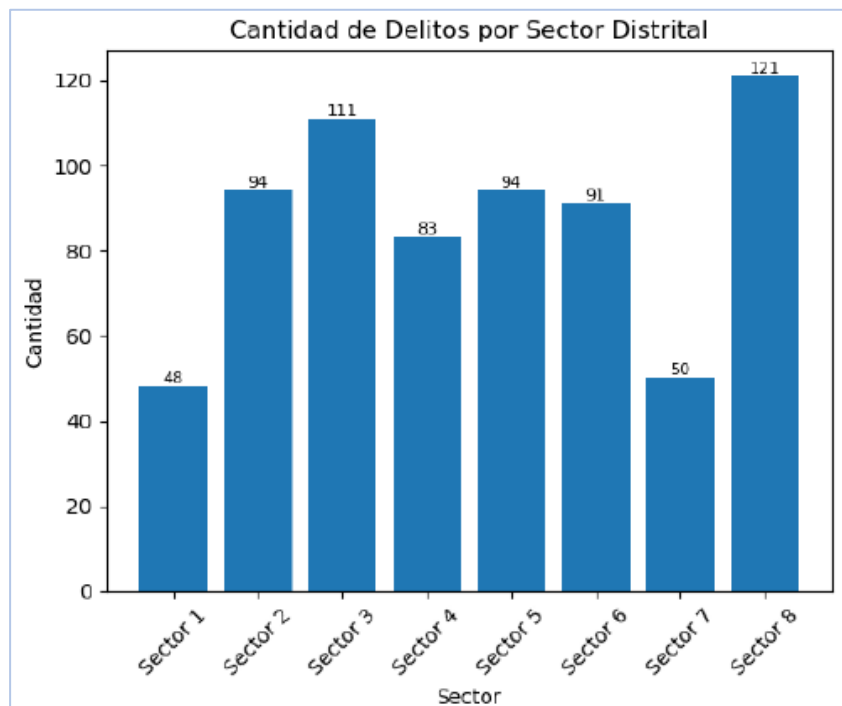
3.2.3. Exploración de los datos

A partir de los datos obtenidos, se procedió a realizar la exploración de los datos, en el cual se aplicó estadísticas de distribución de datos. El código utilizado para la exploración de los datos se detalla en el Anexo 2.

En relación a la distribución de delitos en los sectores distritales, se observa en la Figura 9 que en el Sector 8 y Sector 3 se concentran la mayor cantidad de hechos delictivos. En el Sector 1 y el Sector 7 se muestra una baja cantidad de hechos delictivos.

Figura 9

Cantidad de Delitos por Sector Distrital

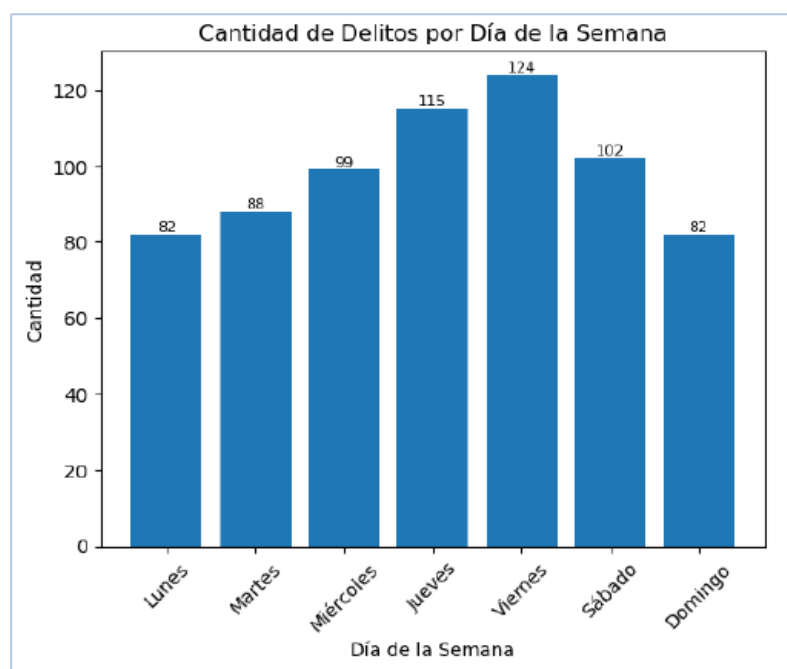


Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, por Municipalidad de Pueblo Libre, 2023. Elaboración Propia.

En relación a la distribución de delitos según el día de la semana, se observa un aumento de los delitos cometidos los viernes y una baja incidencia los días lunes y domingo. En la Figura 10 se muestra un gráfico de barras que detalla la cantidad de delitos por día de la semana.

Figura 10

Cantidad de Delitos Según el Día de la Semana

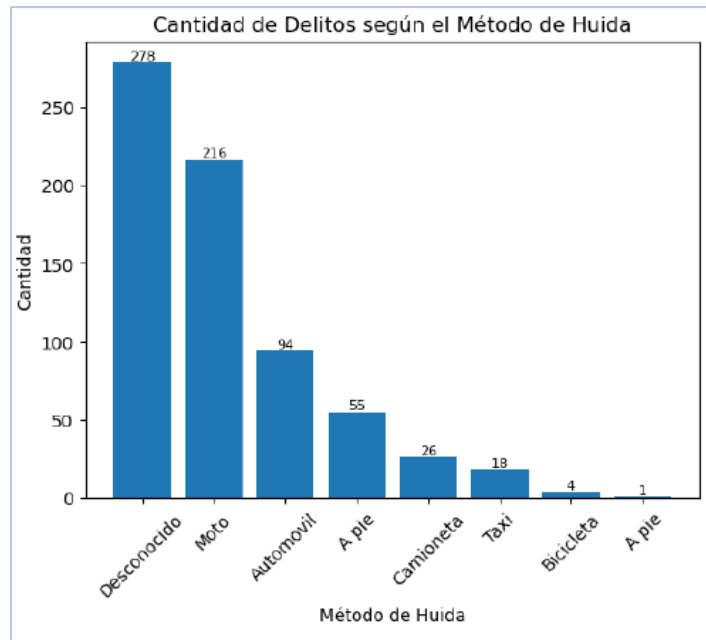


Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

Con respecto a la distribución de delitos según el método de huida, se observa que se desconoce en un número elevado de incidentes delictivos. El método de huida conocido que más veces ocurrió fue vía moto. La cantidad de delitos según el método de Huida se brinda en la Figura 11

Figura 11

Cantidad de Delitos según el Método de Huida

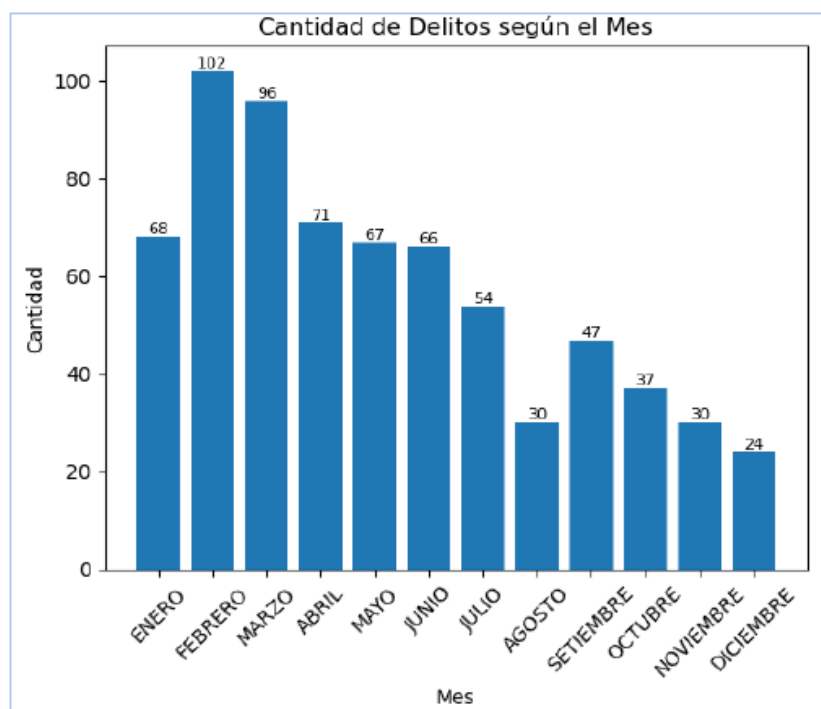


Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

En relación a la distribución de delitos según el mes, se observa que en un elevado número de incidentes delictivos en febrero y marzo y un continuo descenso de los casos delictivos hacia fin de año. La Figura 12 presenta el gráfico que detalla la cantidad de delitos según el Mes.

Figura 12

Cantidad de Delitos según el Mes



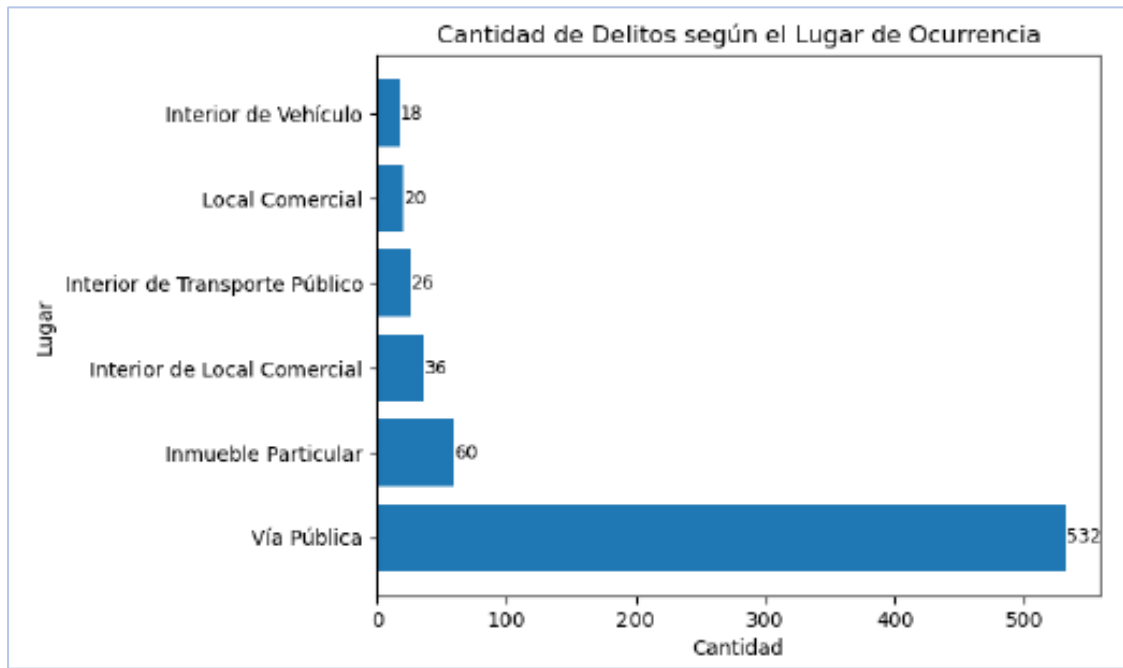
Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

En relación a la distribución de delitos según el lugar de ocurrencia, se observa que gran parte de los incidentes delictivos ocurrieron exactamente en la vía pública, siguiendo lugares como inmueble particular e interior de local comercial. En la Figura 13 se muestra el gráfico que detalla la cantidad de delitos según el Lugar de Ocurrencia.

Respecto a la distribución de delitos según el medio utilizado para la comisión del delito, se observa que gran parte de los incidentes delictivos se produjeron por la habilidad del delincuente, seguidos de arma de fuego y el arrebato. En la Figura 14 se muestra el gráfico que detalla la cantidad de delitos según el Medio Utilizado.

Figura 13

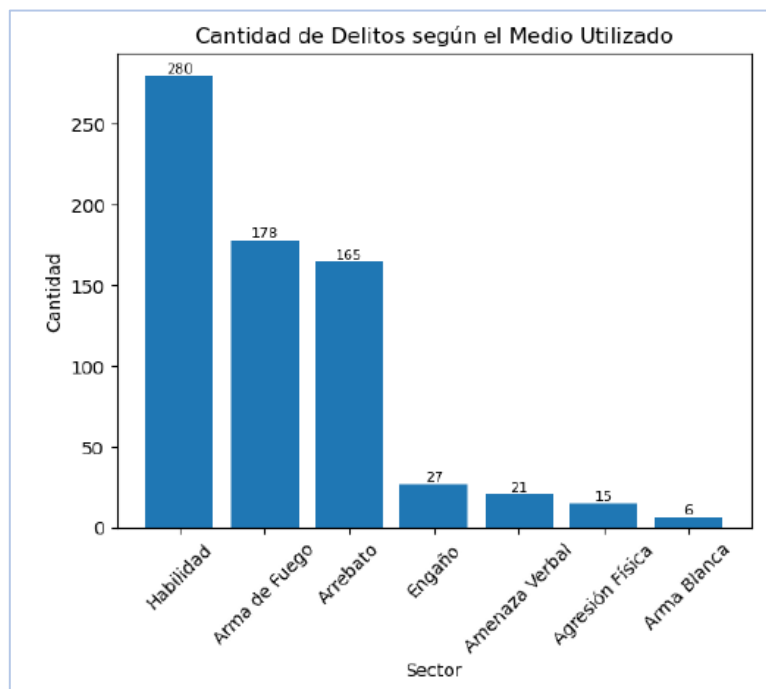
Cantidad de Delitos según el Lugar de Ocurrencia



Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

Figura 14

Cantidad de Delitos según el Medio Utilizado



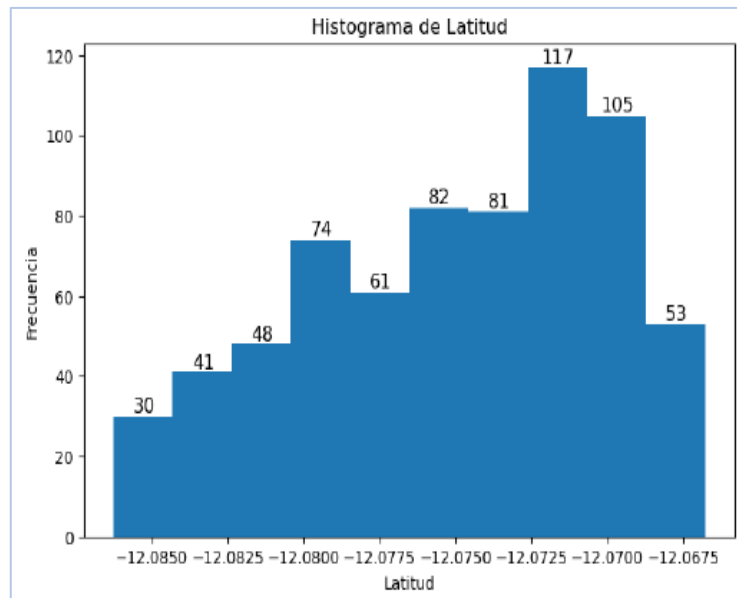
Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

En relación a la distribución de delitos según la latitud, se observa que gran parte de los incidentes delictivos ocurrieron cerca de la zona norte del distrito, entre latitudes -12.0725 y -12.0675 . Se brinda el histograma de la distribución de hechos delictivos según la latitud en la Figura 15.

En relación a la distribución de delitos según la longitud, se observa que gran parte de los incidentes delictivos ocurrieron en la zona de longitud media del distrito, entre latitudes -77.070 y -77.060 . Se brinda el histograma de la distribución de hechos delictivos según la longitud en la Figura 16.

Figura 15

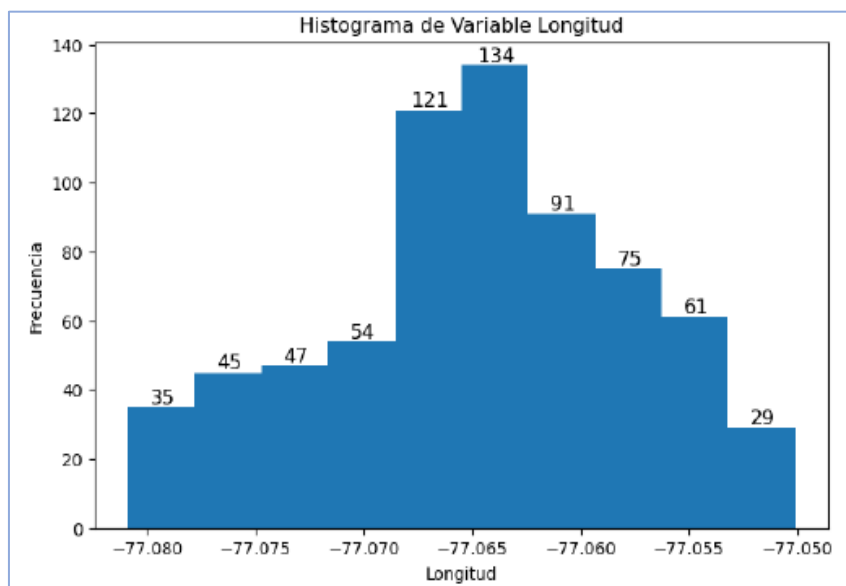
Histograma de la Latitud de Hechos Delictivos



Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

Figura 16

Histograma de la Longitud de Hechos Delictivos



Nota. Datos Delictivos obtenidos del Observatorio de Seguridad Ciudadana de Pueblo Libre, 2023. Elaboración Propia.

3.2.4. Comprobación de la calidad de los datos

Con los resultados de la exploración de datos (visto en el apartado 3.2.3.) se puede afirmar que no se ha encontrado valores erróneos ni valores atípicos. No hay valores vacíos. En el atributo Dirección, se observa algunos campos que tienen direcciones con valores no estandarizados, como espacios en blanco al inicio y final, texto en mayúsculas y minúsculas y múltiples espacios en blanco. Para el tratamiento de este atributo se realizó un proceso de limpieza de datos que se detallará en el apartado número 3.3.2.

3.3. PREPARACIÓN DE LOS DATOS

3.3.1. Selección de los datos

Todas las 692 filas que componen el conjunto de datos serán seleccionadas debido a que los algoritmos de agrupamiento a aplicar no poseen etiquetas como guía (Aprendizaje No Supervisado), por lo que no procede la división del conjunto de datos en datos de prueba y entrenamiento. Considerando el conjunto de datos explorado en la fase anterior, es importante mencionar que los atributos que presentan relevancia en la localización espacial de zonas críticas de delincuencia para el presente proyecto de minería de datos son los siguientes: Latitud, Longitud y Dirección. No se consideró el campo "Sector" por su bajo nivel de especificidad en la localización geográfica. Para esto, se seleccionó dichos atributos y se creó otro Dataframe con nombre "loc_delitos". El código utilizado en esta tarea se muestra en el Anexo 3 (Preparación de Datos). También se seleccionaron las 118 coordenadas geográficas (latitud y longitud) de la línea que delimita el distrito de Pueblo Libre.

3.3.2. Limpieza de los datos

En la limpieza de los datos, se consideró el tratamiento del atributo "Dirección" de los datos delictivos, para lo cual se ejecutó el código Python presentado en el

Anexo 3 (Preparación de Datos). Se realizó uso de los siguientes criterios de limpieza:

- Eliminar los espacios en blanco iniciales y finales de cada entrada en la columna 'Dirección'. (Función `str.strip()`)
- Convertir todo el texto en la columna 'Dirección' a letras mayúsculas. (Función `str.upper()`)
- Eliminar los caracteres '0 ' al final de cada entrada en la columna 'Dirección'. (Función `str.rstrip('0 ')`)
- Reemplaza las secuencias de espacios en blanco (incluyendo espacios, tabulaciones) con un solo espacio en blanco. (Función `str.replace(r'\s+', ' ', regex=True)`)
- Reemplaza la subcadena “EQUIVOCADA” con la subcadena “CORRECTA” en la columna “Dirección”. (Función `str.replace (“EQUIVOCADA”, “CORRECTA”, regex=False)`)

También se realizó un preprocesamiento de las coordenadas geográficas del límite de Pueblo Libre. El código utilizado en esta tarea se muestra en el Anexo 3 (Preparación de Datos).

3.3.3. Integración y Normalización de los datos

Al conjunto de datos limpiado, se añadió registros (concatenó mediante filas) de los puntos limítrofes de Pueblo Libre, así como de un campo 'Dirección' que indica las avenidas o cruces de dichos puntos limítrofes. Adicionalmente, se añadió un nuevo atributo con nombre 'Indicador' el cual toma valor 0 para un punto limítrofe o 1 para un punto delictivo. El código Python utilizado se muestra en el Anexo 3.

También se realizó la normalización de las coordenadas geográficas. Para lograr esta normalización se usó el escalado los atributos 'Longitud' y 'Latitud' a valores entre 0 y 1. Se realizó el proceso de escalado utilizando la Ecuación 15.

Para cada columna X en el conjunto de datos:

$$X_{escalado} = \frac{x - X_{min}}{X_{max} - x_{min}} \quad (15)$$

- $X_{escalado}$ es el valor escalado de x .
- x es el valor inicial del atributo X .
- X_{min} es el valor mínimo del atributo X .
- X_{max} es el valor máximo del atributo X .

El proceso de escalado de datos se realiza utilizando el utilizando la función MinMaxScaler, la cual es una Función de Preprocesamiento que ejecuta el escalado (normalización) de datos en un rango de 0 a 1. Luego, para facilitar la optimización de parámetros en el modelado de los algoritmos se multiplicó por 100 cada valor obtenido, obteniéndose un rango de 0 a 100. A los atributos derivados creados a partir del proceso de escalado se le asignaron los nombres: 'X' (Longitud escalada) y 'Y' (Latitud escalada)'. El código Python para el proceso anterior se muestra en el Anexo 3.

3.3.4. Formateo de los datos

En esta etapa, se agregó un identificador el cual sea único para cada coordenada geográfica, el cual será importante para la agrupación de cada uno de los datos delictivos. Posteriormente, para facilitar la optimización de parámetros en el modelado de los algoritmos se multiplicó por 100 a los atributos derivados 'X' e 'Y', consiguiéndose un rango de 0 a 100. Para la asignación del atributo 'Id' se utilizó el código mostrado en el Anexo 3. A partir de la selección, limpieza, construcción, integración y formateo de los datos se puede iniciar adecuadamente con la siguiente

fase de modelado de los algoritmos de aprendizaje de máquina y posteriormente aplicar los algoritmos genéticos.

3.4. MODELADO

3.4.1. Seleccionar las técnicas de modelado

Como primer paso para el modelado algorítmico, se enlista los algoritmos de agrupamiento y el criterio de selección:

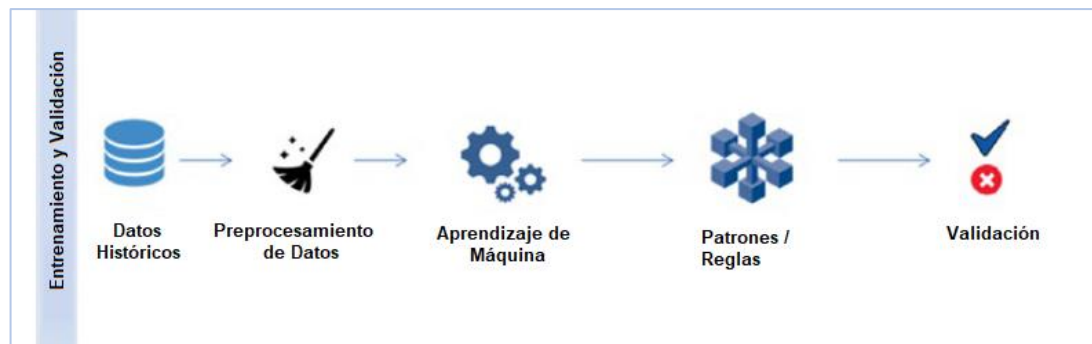
- Algoritmo K-Medias: Otorga un agrupamiento sencillo a un conjunto de observaciones. Permite agrupar en base a distancias euclidianas y su parametrización solo necesita la evaluación del valor 'K' (Número de agrupaciones o 'clústeres').
- Algoritmo DBSCAN: Otorga un agrupamiento espacial en base a densidad. Genera grupos y etiqueta ciertas observaciones sin agrupar como ruido. Su parametrización es más compleja, ya que requiere encontrar el valor óptimo ϵ (épsilon) y la cantidad mínima de puntos (MinPts).
- Algoritmo HDBSCAN: Se basa en el concepto de densidad alcanzable y utiliza una estructura de árbol (Árbol de Expansión Mínima) para representar la relación entre los clústeres a diferentes escalas de densidad.
- Algoritmo Genético Adaptado al Problema del Agente Viajero. La solución consiste en encontrar un individuo cuya combinación de genes, de solución al problema de visitar todas las zonas críticas de delincuencia y retornar al punto de origen.

3.4.2. Generar el Plan de Prueba

En la aplicación de los A. de Aprendizaje de Máquina No Supervisados se aplicará el proceso de entrenamiento y validación mostrado en la Figura 17.

Figura 17

Entrenamiento y Validación de Algoritmos No Supervisados



Nota. Extraído de “*Mastering Machine Learning with Python in Six Steps*”. (p. 195), por M. Swamynathan, 2017, Apress eBooks.

Respecto a los indicadores e índices para validar los algoritmos de agrupamiento se indican a continuación:

- Índice de Silueta
- Índice Calinski – Harabasz
- Índice Davies – Boulding

3.4.3. Creación de Modelos

En este apartado se mostrarán las gráficas de las agrupaciones obtenidas producto de la aplicación de los algoritmos de Agrupamiento K-Medias, DBSCAN, HDBSCAN y Propagación de Afinidad. Asimismo, el proceso de optimización de sus hiperparámetros aplicando las métricas de evaluación para cada algoritmo individualmente.

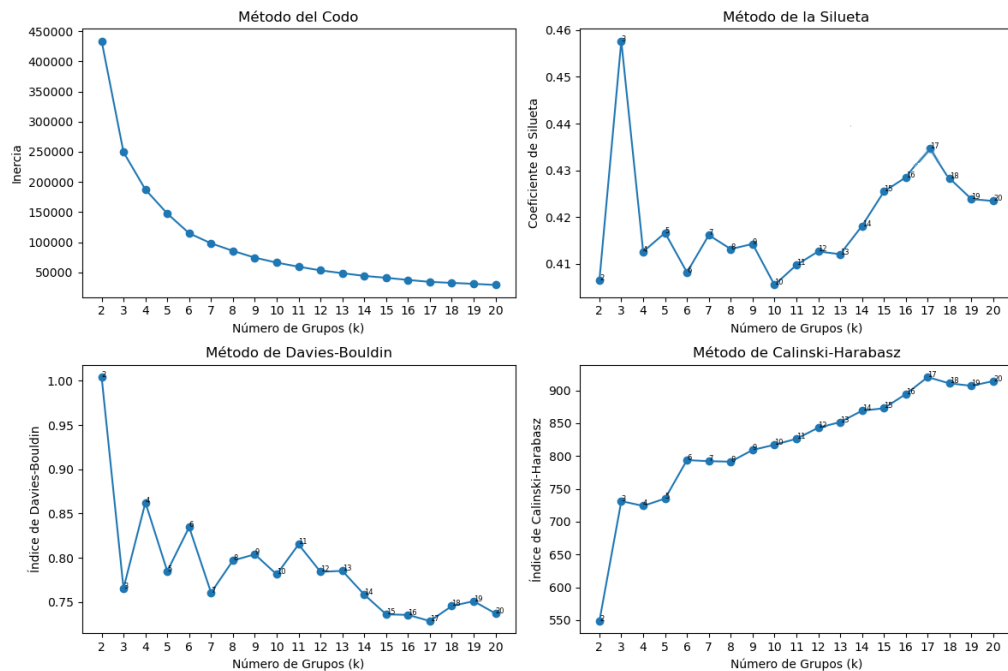
3.4.3.1. Algoritmo K- Medias

Se procedió a ejecutar el algoritmo K-Medias en el conjunto de datos delictivos para un rango de número de grupos, es decir valores “ k ” (2,3,4, ...

20), con los cuales se evaluó los resultados de sus agrupamientos por medio del método del codo (Inercia) y las 3 métricas de evaluación (Índice Silueta, Índice Davies – Bouldin e Índice de Calinski – Harabasz). El código utilizado para la ejecución y validación individual del algoritmo K-Medias se encuentra en el Anexo 4. En la Figura 18, se muestran las gráficas del método del codo y las métricas de evaluación para los distintos valores de “ k ”.

Figura 18

Gráficas de las Métricas de Evaluación para el Algoritmo K-Medias



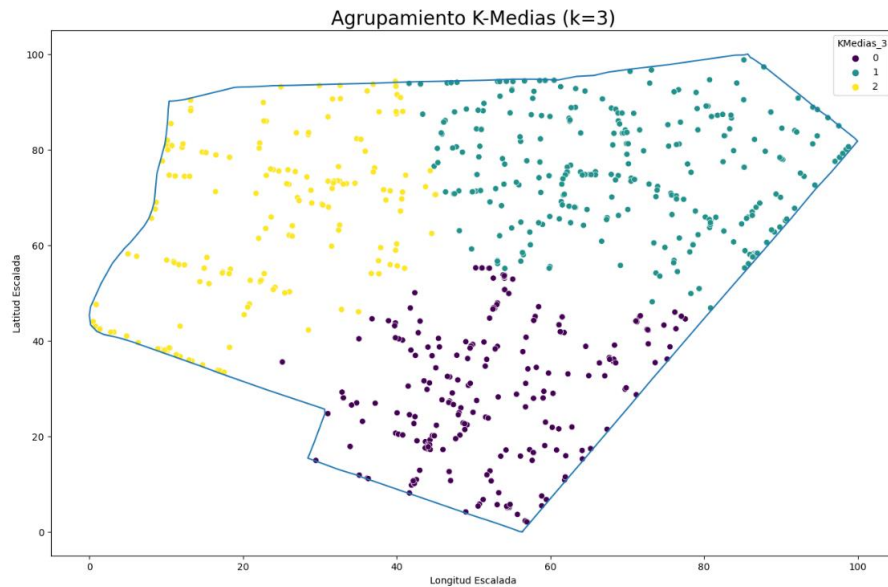
Nota. Gráficas obtenidas con el Software Jupyter Notebook. Elaboración Propia.

En el conjunto de datos, se encontró que los valores de “ k ” igual a 3 y 17 consiguen métricas de evaluación superiores a otros valores de “ k ”.

A continuación, se procedió a graficar el Mapa de Puntos de los agrupamientos obtenidos con los valores de “ k ” iguales a 3 y 17 (Figura 19 y Figura 20 respectivamente),

Figura 19

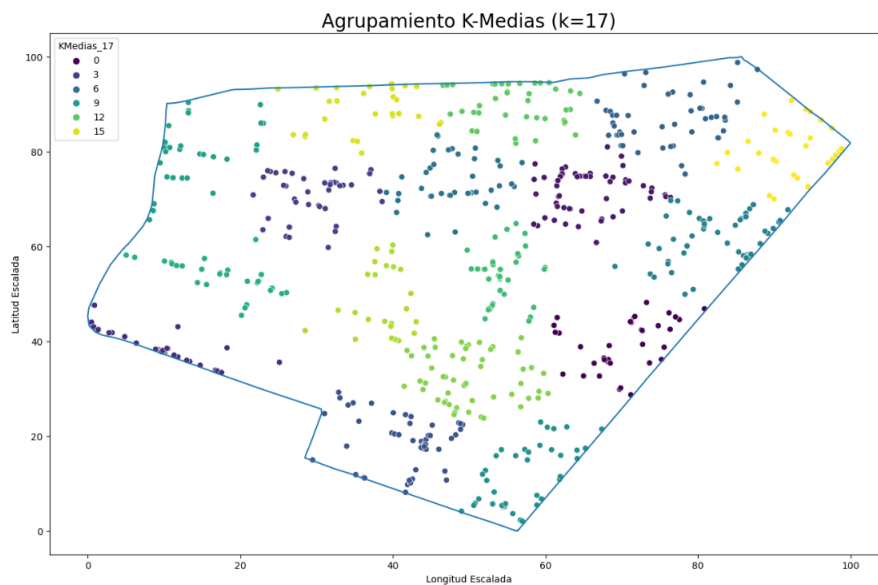
Mapa de Puntos para el Agrupamiento K-Medias (k=3)



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

Figura 20

Mapa de Puntos para el Agrupamiento K-Medias (k=17)



Nota. Gráfica generada a partir del software Jupyter Notebook. Elaboración Propia.

En la Figura 19 (Agrupación vía K-Medias con 3 grupos) se observa que el agrupamiento logrado no es específico, en términos de localización específica. Por lo tanto, no contribuye a la determinación de las zonas críticas de delincuencia. En cambio, en la Figura 20 (Agrupación vía K-Medias con 17 grupos), se observa un mejor agrupamiento, con lo que se podría lograr la determinación de las zonas críticas de delincuencia.

3.4.3.2. Algoritmo DBSCAN

Se procedió a ejecutar el algoritmo DBSCAN en el conjunto de datos delictivos con los siguientes valores de los parámetros `min_samples` y `épsilon (eps)`: rango de números enteros de 2 a 9 para `min_samples` y números reales de 1 a 5 con saltos de 0.05; luego se ejecutó el algoritmo DBSCAN y se obtuvieron las métricas de evaluación Índice Silueta, Índice Davies – Bouldin e Índice de Calinski – Harabasz. También se calculó la cantidad puntos no clasificados (ruido) y el número de agrupaciones. El código utilizado para este proceso se encuentra en el Anexo 4. En la Tabla 7, se presentan los 10 resultados con mejor Silueta, siendo este indicador el que mejor se ajusta a la evaluación del algoritmo DBSCAN. Se observa que el Índice Davies-Bouldin no presenta mucha variabilidad, por lo cual no se consideró como una métrica relevante. Se observa que las combinaciones de los parámetros `min_samples` toma un valor 2 y `épsilon` toma valores de 1.8 a 2.45. Al considerar el Índice Calinski – Harabasz, se detectó que al aumentaba conjuntamente con el valor de `épsilon` lo cual disminuye la densidad de puntos para la determinación de grupos de DBSCAN; por lo que se consideró solo los mayores valores del Índice Calinski – Harabasz que hayan sido producto de una `épsilon` menor a 2.1. Graficando las 4 combinaciones de `min_samples` y `épsilon`, se obtienen la Figura 21 (`min_samples=2`, `eps=1.95`), la Figura 22 (`min_samples=2`, `eps=2.0`), la Figura 23 (`min_samples=2`, `eps=2.05`) y la Figura 24 (`min_samples=2`, `eps=2.10`) de los mapas de puntos.

Tabla 7

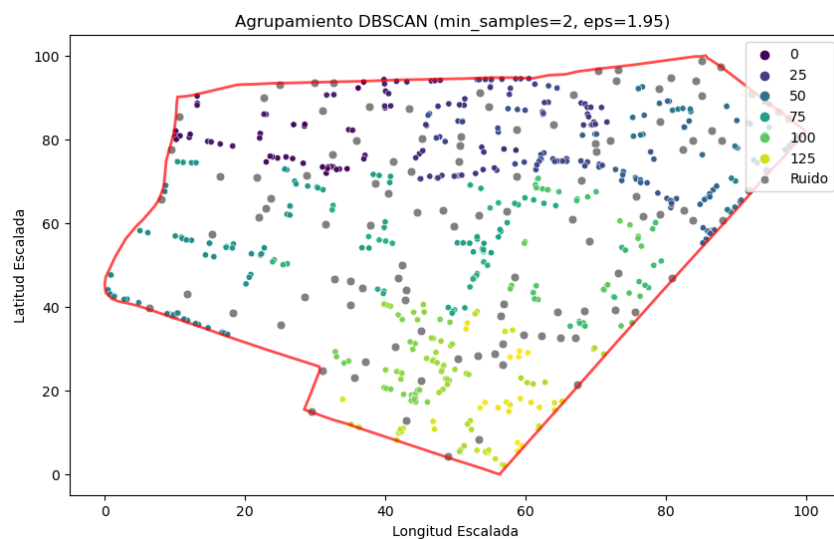
Parámetros y Métricas de Evaluación del Algoritmo DBSCAN

Min_samples	Epsilon	Índice Silueta	Índice Davies Bouldin	Índice Calinski-Harabasz	Nro. de Grupos	Nro. Filas "Ruido"
2	1.95	0.367	1.359	21.498	136	113
2	2.00	0.360	1.382	23.64	133	108
2	1.90	0.353	1.38	19.848	139	118
2	2.10	0.352	1.445	27.582	130	95
2	1.85	0.347	1.368	18.731	140	122
2	2.05	0.346	1.405	26.318	130	101
2	1.8	0.346	1.374	17.639	141	128
2	1.75	0.343	1.439	15.586	145	135
2	2.45	0.34	1.354	48.838	106	66
2	2.4	0.336	1.345	46.168	108	69

Nota. Se muestran los 10 registros con mayor valor del Índice Silueta. Los valores de los índices de evaluación fueron truncados a 3 decimales. Elaboración Propia.

Figura 21

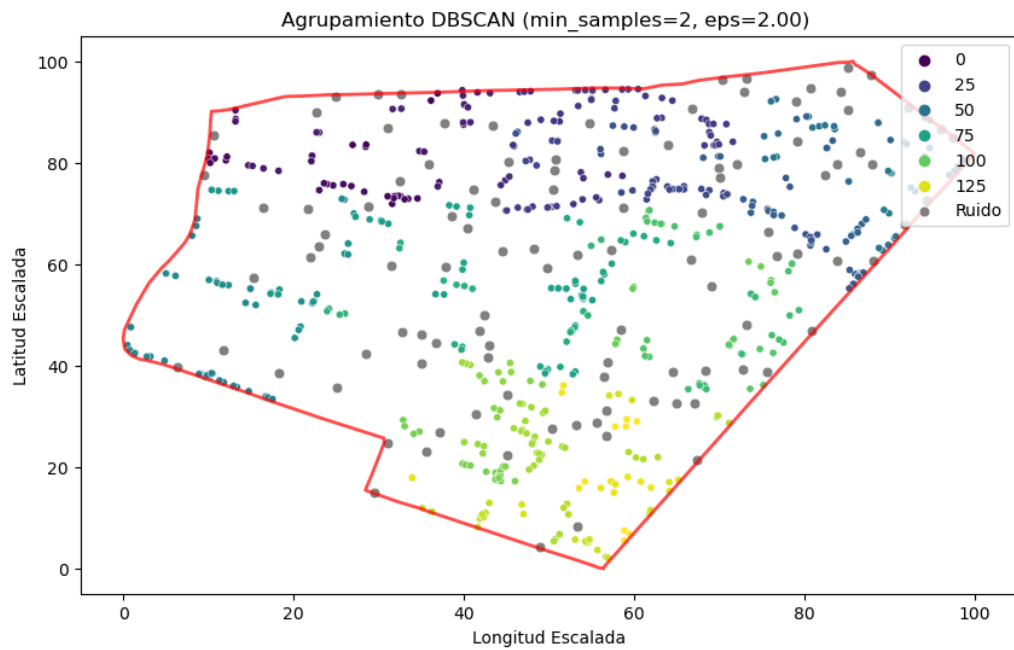
Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 1.95)



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

Figura 22

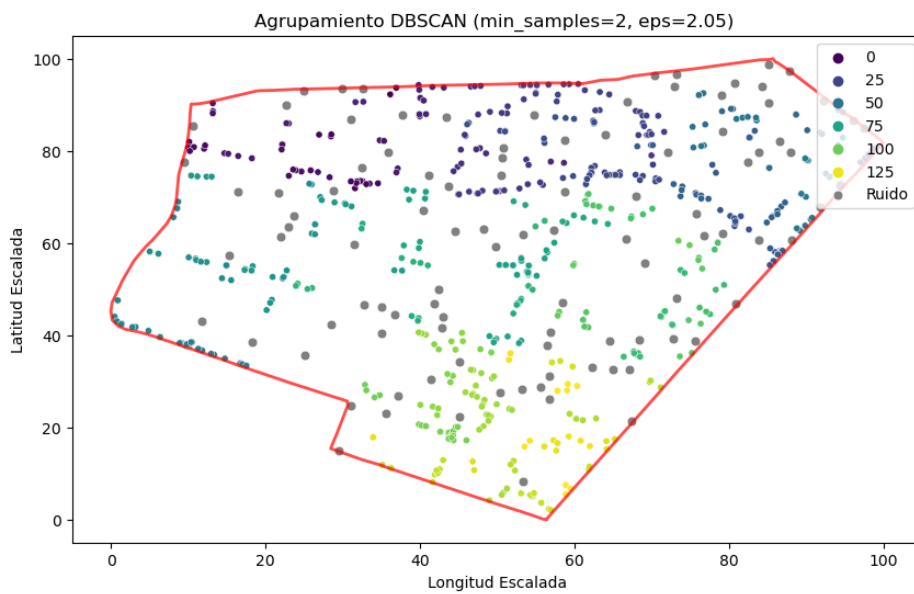
Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 2.00)



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

Figura 23

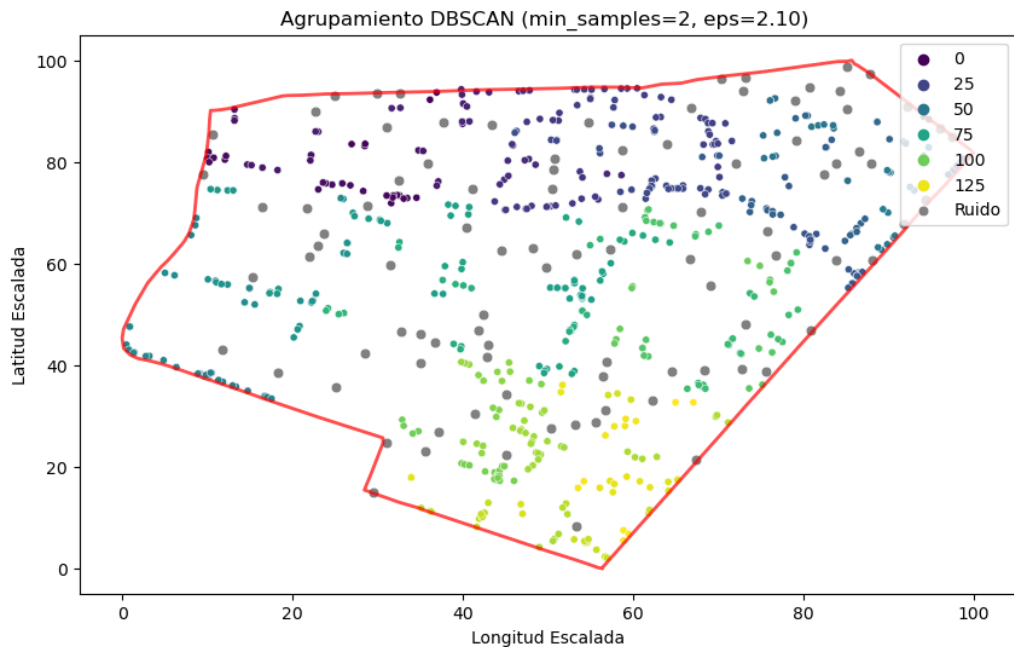
Mapa de Puntos usando DBSCAN (min_samples = 2, eps = 2.05)



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

Figura 24

Mapa de Puntos usando DBSCAN ($min_samples = 2$, $eps = 2.10$)



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

En la Figura 21 y la Figura 22, se observan que algunos puntos en el extremo oeste del distrito no fueron clasificados dentro de un grupo a pesar de estar en una zona de alta densidad, considerándose como ruido, lo que es indicativo de un agrupamiento inadecuado, por lo que se descartan las combinaciones de parámetros $min_samples$ y eps de (2,1.95) y (2,2.00). En las Figuras 23 y 24, se observan mapas de puntos que tienen agrupaciones adecuadas y presentan ambos 130 grupos definidos, teniendo la combinación de $min_samples$ y eps de (2, 2.10) 95 puntos ruido respecto a la combinación (2, 2.05), la cual tiene 101 puntos ruido. Por lo expuesto anteriormente, se considera que la combinación de $min_samples$ y eps igual a 2, 2.10 es la combinación idónea para el algoritmo de agrupamiento DBSCAN.

3.4.3.3. Algoritmo HDBSCAN

En esta etapa, se procedió a ejecutar Algoritmo HDBSCAN en el conjunto de datos delictivos para distintos valores de parámetros *min_cluster_size* (rango de números enteros de 2 a 4), *min_samples* (rango de números enteros de 2 a 4) y *cluster_selection_epsilon* (rango de números reales de 1 a 5 con saltos de 0.05) y se obtuvieron las métricas de evaluación Índice Silueta, Índice Davies – Bouldin e Índice de Calinski – Harabasz. También se calculó la cantidad puntos no clasificados (ruido) y el número de agrupaciones. Posteriormente se calculó un Función “Score” con las 3 métricas de evaluación siguiendo la Ecuación 16. El código utilizado para este proceso se encuentra en el Anexo 4.

$$Score = (F_{Silueta}S_{Normalizada} + F_{DaviesBouldin}DB_{Normalizada} + F_{CalinskiHarabasz}CH_{Normalizada}) / \sum F \quad (16)$$

Donde:

- $F_{Silueta}$: Factor de Ponderación del Índice Silueta
- $S_{Normalizada}$: Índice Silueta Normalizado de 0 a 1
- $F_{DaviesBouldin}$: Factor de Ponderación del Índice Davies-Bouldin
- $DB_{Normalizada}$: Índice Davies-Bouldin Normalizado de 0 a 1
- $F_{CalinskiHarabasz}$: Factor de Ponderación del Índice Calinski-Harabasz
- $CH_{Normalizada}$: Índice Calinski-Harabasz Normalizado de 0 a 1
- $\sum F$: Sumatoria de Factores de Ponderación

En la Ecuación 17 se muestra la normalización del Índice Silueta.

$$S_{Normalizada} = \frac{|Silueta| + Silueta}{2} \quad (17)$$

En la Ecuación 18 se muestra la normalización del Índice Davies-Bouldin.

$$DB_{Normalizada} = \frac{\max(DaviesBouldin) - DaviesBouldin}{\max(DaviesBouldin) - \min(DaviesBouldin)} \quad (18)$$

En la Ecuación 19 se muestra la normalización del Índice Calinski-Harabasz.

$$CH_{Normalizada} = \frac{CalinskiHarabasz - \min(CalinskiHarabasz)}{\max(CalinskiHarabasz) - \min(CalinskiHarabasz)} \quad (19)$$

Donde:

- *Silueta*: Índice Silueta
- *DaviesBouldin*: Índice Davies-Bouldin
- *CalinskiHarabasz*: Índice Calinski-Harabasz

Los factores de ponderación fueron 5,1,1 para el Índice Silueta, Davies-Bouldin y Calinski-Harabasz respectivamente. En base al cálculo de las métricas de evaluación, se obtuvo los 14 registros con los mejores valores del Score. Los resultados se muestran en la Tabla 8.

De las combinaciones obtenidas en la Tabla 8, se observa que el valor de *min_samples* óptimo es 2. El valor *min_cluster_size* toma valores de 2, 3 y 4; a partir de estos valores, se consideraron las *epsilon* (*cluster_selection_epsilon*) más pequeños (que brindan agrupaciones más densas): estos valores fueron 2, 2 y 1.85 respectivamente. A continuación, se procedió a graficar las agrupaciones resultantes con las 3 combinaciones de parámetros del Algoritmo HDBSCAN; los resultados obtenidos producto del Agrupamiento HDBSCAN se brindan en la Figura 25 (*min_cluster_size*=2, *min_samples*=2, *epsilon* = 2), Figura 26 (*min_cluster_size*=3, *min_samples*=2, *epsilon* = 2) y la Figura 27 (*min_cluster_size* = 4, *min_samples*=2, *epsilon* = 1.85).

Tabla 8

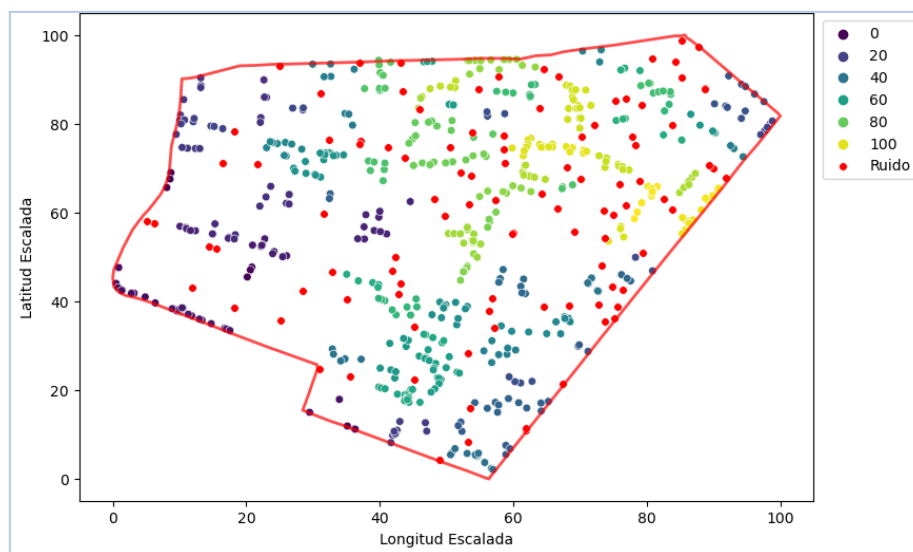
Parámetros y Métricas de Resultados del Algoritmo HDBSCAN

min cluster size	min samples	cluster selection epsilon	Índice Silueta	Índice Davies Bouldin	Índice Calinski Harabasz	Nro. De Grupos	Nro. Filas "Ruido"	Score
3	2	2.55	0.317	1.213	66.652	78	79	0.411
3	2	2.45	0.323	1.209	61.701	82	82	0.411
3	2	2.4	0.323	1.222	60.896	83	82	0.410
2	2	2.45	0.320	1.258	58.990	86	79	0.406
2	2	2.55	0.314	1.267	63.571	82	76	0.405
2	2	2.4	0.320	1.271	58.253	87	79	0.405
3	2	2.5	0.312	1.199	62.364	81	82	0.404
2	2	2	0.348	1.396	36.794	106	102	0.404
2	2	1.95	0.349	1.364	35.291	108	104	0.403
3	2	2	0.343	1.363	38.723	99	106	0.402
4	2	1.85	0.248	1.352	37.938	65	136	0.334
4	2	1.9	0.248	1.352	37.938	65	136	0.334
4	2	1.95	0.248	1.352	37.938	65	136	0.334
4	2	2	0.248	1.352	37.938	65	136	0.334

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

Figura 25

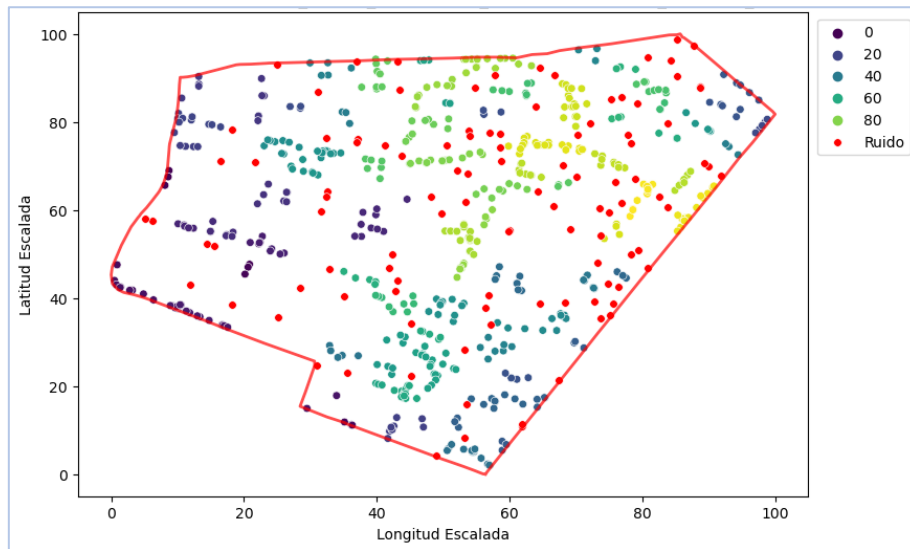
Agrupamiento HDBSCAN ($m_c_size^*=2$, $min_samples=2$, $epsilon=2$)



Nota. Gráfica generada con el software Jupyter Notebook. *El parámetro $min_cluster_size$ se abrevió como m_c_size . Elaboración Propia.

Figura 26

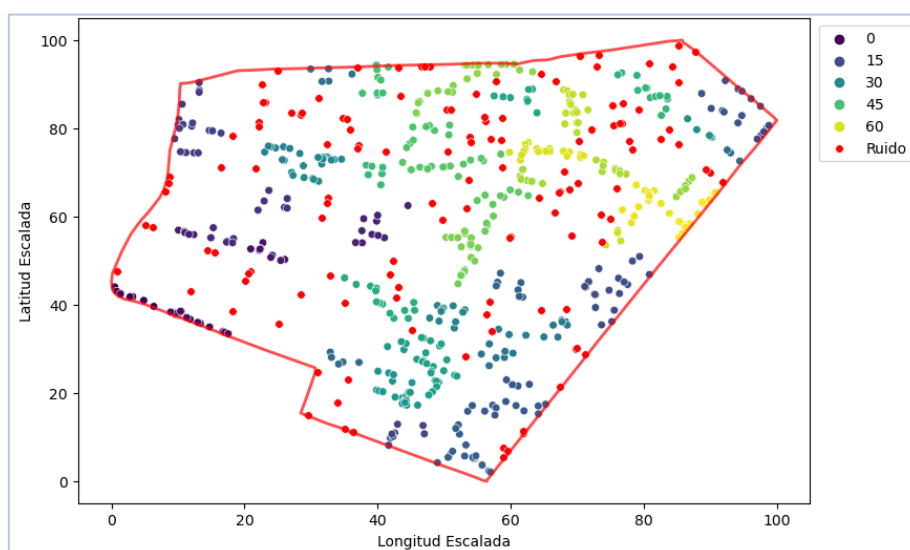
Agrupamiento HDBSCAN ($m_c_size^=3$, $min_samples=2$, $epsilon=2$)*



Nota. Gráfica generada con el software Jupyter Notebook. *El parámetro $min_cluster_size$ se abrevió como m_c_size . Elaboración Propia.

Figura 27

Agrupamiento HDBSCAN ($m_c_size^=4$, $min_samples=2$, $eps^*=1.85$)*



Nota. Gráfica generada con el software Jupyter Notebook. *El parámetro $min_cluster_size$ se abrevió como m_c_size y $epsilon$ como eps . Elaboración Propia.

3.4.3.4. Análisis de las Densidades Delictivas

Cálculo de densidad delictiva del distrito. Para calcular la densidad delictiva del distrito, se aplicó la Ecuación 1 descrita en el acápite “Descripción de los Indicadores” de la Matriz de Operacionalización de Variables. Posteriormente, se usó las coordenadas geográficas de los puntos limítrofes de Pueblo Libre, para generar el objeto Polígono a partir de la función Polígono, luego se calculó su área en grados cuadrados, en kilómetros cuadrados y luego en hectómetros cuadrados. El código empleado para el cálculo de la Densidad Delictiva del Distrito (DDD) se presenta en el Anexo 5. La Densidad Delictiva del Distrito (DDD) de Pueblo Libre se determina en la Ecuación 20.

$$DDD = \frac{\text{Número de delitos}}{\text{Área del Distrito}} = \frac{692 \text{ delitos}}{472.28 \text{ hm}^2} = 1.47 \text{ delitos/hm}^2 \quad (20)$$

Cálculo de la densidad delictiva/ densidad delictiva acumulada de los agrupamientos. Para el cálculo de la densidad delictiva de los agrupamientos, también se utilizó de Ecuación 1, la cual se adecuó a cada grupo determinado por los algoritmos de agrupamiento, obteniéndose la Ecuación 21. Luego, teniendo la Densidad Delictiva del Distrito, se procedió a calcular la Densidad Delictiva Relativa utilizando la Ecuación 2 descrita en el acápite “Descripción de los Indicadores” de la Matriz de Operacionalización de Variables. A partir de la Ecuación 21, se construyó la función '**calcular_densidad**' (cuyo código Python se presenta en el Anexo 5), la cual realiza el cálculo de la densidad delictiva para cada grupo en un determinado DataFrame.

$$DD_i = \frac{\text{Número de delitos}_i}{\text{Área Convexa de las Coordenadas}_i} \quad (21)$$

Donde:

- DD_i : Densidad Delictiva del Grupo i .
- $\text{Número de delitos}_i$: Número de Delitos del Grupo i .
- $\text{Área Convexa de las Coordenadas}_i$: Área Convexa de las Coordenadas de los Delitos pertenecientes al Grupo i .

Posteriormente, se observó que algunas Área Convexa de las Coordenadas de los Delitos pertenecientes determinados grupos tenían áreas muy pequeñas (menores a 100m^2), las cuales alteraban los resultados de las densidades delictivas, por este motivo, se filtró las áreas de las agrupaciones que sean menores a 0.01hm^2 . Para determinar la Densidad Acumulada, se utilizó la Ecuación 22, descrita a continuación. El código utilizado para realizar los cálculos de las densidades delictivas se muestra en el Anexo 5.

$$\text{Densidad Delictiva}_{acum} = \frac{\sum_i ND_i}{\sum_i AC_i} \quad (22)$$

Donde:

- $\text{Densidad Delictiva}_{acum}$: Densidad Delictiva Acumulada.
- ND_i : Número de Delitos pertenecientes al Grupo i .
- AC_i : Área Convexa de las Coordenadas de los Delitos pertenecientes al Grupo i .

Densidad Delictiva del Agrupamiento K-Medias (K = 3).

La Densidad Delictiva de cada grupo resultante de la aplicación del algoritmo de agrupamiento K-Medias (K = 3) se muestra en la Tabla 9.

Tabla 9

Densidades Delictivas del Agrupamiento K-Medias (k=3)

Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	1.76 delitos/hm ²	1.20 veces
1	1.60 delitos/hm ²	1.09 veces
2	1.24 delitos/hm ²	0.84 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Densidad Delictiva Relativa se calculó en base a la Densidad Delictiva del Distrito de 1.47 delitos/hm².
Elaboración Propia.

Como los 3 grupos poseen un área convexa con un tamaño mayor a 0.01 hm², se calculó la Densidad Delictiva Acumulada para el Agrupamiento K-Medias (k=3) resultando el valor de 1.54 delitos/hm² y la Densidad Delictiva Acumulada relativa de 1.05 veces la Densidad Delictiva del Distrito (1.47 delitos/hm²).

Densidad Delictiva del Agrupamiento K-Medias (K = 17).

La Densidad Delictiva de cada grupo resultante de la aplicación del algoritmo de agrupamiento K-Medias (K = 17) se muestra en la Tabla 10.

Tabla 10*Densidades Delictivas del Agrupamiento K-Medias (k=17)*

Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	2.74 delitos/hm ²	1.87 veces
1	2.84 delitos/hm ²	1.94 veces
2	2.20 delitos/hm ²	1.50 veces
3	2.09 delitos/hm ²	1.43 veces
4	3.01 delitos/hm ²	2.06 veces
5	2.38 delitos/hm ²	1.62 veces
6	2.52 delitos/hm ²	1.72 veces
7	2.03 delitos/hm ²	1.39 veces
8	3.92 delitos/hm ²	2.67 veces
9	1.65 delitos/hm ²	1.13 veces
10	2.78 delitos/hm ²	1.90 veces
11	2.78 delitos/hm ²	1.90 veces
12	2.15 delitos/hm ²	1.47 veces
13	1.96 delitos/hm ²	1.34 veces
14	1.80 delitos/hm ²	1.23 veces
15	3.60 delitos/hm ²	2.45 veces
16	1.86 delitos/hm ²	1.27 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Densidad Delictiva Relativa medida en base a la Densidad Delictiva del Distrito (DDD) que es 1.47 delitos/hm². Elaboración Propia.

Como los 17 grupos poseen un área convexa con un tamaño mayor a 0.01 hm², se calculó la Densidad Delictiva Acumulada para el Agrupamiento K-Medias (k=3) resultando el valor de 2.43 delitos/hm² y la Densidad Delictiva Acumulada relativa de 1.66 veces la Densidad Delictiva del Distrito (1.47 delitos/hm²).

Densidad Delictiva del Agrupamiento DBSCAN (min_samples = 2, eps = 2.10).

La Densidad Delictiva de cada grupo resultante de la aplicación del algoritmo de agrupamiento DBSCAN (min_samples=2, eps= 2.10) se muestra en la Tabla 11.

Tabla 11*Densidades Delictivas del Agrupamiento DBSCAN (min_samples=2, eps=2.1)*

Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	14.51 delitos/hm ²	9.9 veces
1	114.57 delitos/hm ²	78.19 veces
2	39.22 delitos/hm ²	26.76 veces
3	31.97 delitos/hm ²	21.82 veces
4	16.63 delitos/hm ²	11.35 veces
5	3285.92 delitos/hm ²	2242.61 veces
6	1383.45 delitos/hm ²	944.19 veces
7	65.8 delitos/hm ²	44.91 veces
8	283.24 delitos/hm ²	193.31 veces
9	65.18 delitos/hm ²	44.48 veces
10	209.51 delitos/hm ²	142.99 veces
11	15.31 delitos/hm ²	10.45 veces
12	20.14 delitos/hm ²	13.75 veces
13	201.9 delitos/hm ²	137.79 veces
14	7.18 delitos/hm ²	4.9 veces
15	33.61 delitos/hm ²	22.94 veces
16	19022.4 delitos/hm ²	12982.58 veces
17	988.98 delitos/hm ²	674.97 veces
18	671.69 delitos/hm ²	458.42 veces
19	32.94 delitos/hm ²	22.48 veces
20	22.6 delitos/hm ²	15.43 veces
21	12.19 delitos/hm ²	8.32 veces
22	12.56 delitos/hm ²	8.57 veces
23	7.43 delitos/hm ²	5.07 veces
24	12.04 delitos/hm ²	8.22 veces
25	67.06 delitos/hm ²	45.77 veces
26	23.34 delitos/hm ²	15.93 veces
27	1113.34 delitos/hm ²	759.84 veces
28	11.97 delitos/hm ²	8.17 veces
29	1672.3 delitos/hm ²	1141.33 veces
30	55.84 delitos/hm ²	38.11 veces
31	1664.3 delitos/hm ²	1135.87 veces
32	239.74 delitos/hm ²	163.62 veces
33	25.05 delitos/hm ²	17.09 veces
34	392.43 delitos/hm ²	267.83 veces
35	14.26 delitos/hm ²	9.73 veces
36	59.69 delitos/hm ²	40.74 veces
37	807.45 delitos/hm ²	551.08 veces
38	536.85 delitos/hm ²	366.39 veces
39	22.01 delitos/hm ²	15.02 veces
40	97.0 delitos/hm ²	66.20 veces
41	190.12 delitos/hm ²	129.75 veces
42	481.48 delitos/hm ²	328.61 veces
43	5316.3 delitos/hm ²	3628.32 veces
44	7151.49 delitos/hm ²	4880.81 veces
45	7.57 delitos/hm ²	5.17 veces
46	8575.85 delitos/hm ²	5852.92 veces
47	48.18 delitos/hm ²	32.88 veces
48	4435.1 delitos/hm ²	3026.9 veces
49	4660.03 delitos/hm ²	3180.42 veces
50	56.67 delitos/hm ²	38.68 veces

Grupo	Densidad Delictiva	Densidad Delictiva Relativa
51	64.63 delitos/hm ²	44.11 veces
52	3840.48 delitos/hm ²	2621.09 veces
53	14.18 delitos/hm ²	9.68 veces
54	39.76 delitos/hm ²	27.13 veces
55	70.25 delitos/hm ²	47.95 veces
56	8846.02 delitos/hm ²	6037.31 veces
57	170.01 delitos/hm ²	116.03 veces
58	22441.17 delitos/hm ²	15315.86 veces
59	31.21 delitos/hm ²	21.3 veces
60	63243.31 delitos/hm ²	43162.87 veces
61	223.48 delitos/hm ²	152.53 veces
62	14.07 delitos/hm ²	9.6 veces
63	15.61 delitos/hm ²	10.65 veces
64	149.1 delitos/hm ²	101.76 veces
65	50.99 delitos/hm ²	34.8 veces
66	122.47 delitos/hm ²	83.58 veces
67	1614.63 delitos/hm ²	1101.97 veces
68	394.57 delitos/hm ²	269.29 veces
69	97.38 delitos/hm ²	66.46 veces
70	49.14 delitos/hm ²	33.53 veces
71	21.06 delitos/hm ²	14.37 veces
72	22.41 delitos/hm ²	15.29 veces
73	15.74 delitos/hm ²	10.74 veces
74	69.05 delitos/hm ²	47.13 veces
75	20.41 delitos/hm ²	13.93 veces
76	28.85 delitos/hm ²	19.69 veces
77	8.3 delitos/hm ²	5.66 veces
78	15.01 delitos/hm ²	10.24 veces
79	211.7 delitos/hm ²	144.49 veces
80	134.68 delitos/hm ²	91.91 veces
81	43.36 delitos/hm ²	29.59 veces
82	160.8 delitos/hm ²	109.74 veces
83	1239.75 delitos/hm ²	846.12 veces
84	55.76 delitos/hm ²	38.06 veces
85	10.34 delitos/hm ²	7.06 veces
86	47.74 delitos/hm ²	32.58 veces
87	60.97 delitos/hm ²	41.61 veces
88	57.12 delitos/hm ²	38.98 veces
89	13.27 delitos/hm ²	9.06 veces
90	2089.12 delitos/hm ²	1425.8 veces
91	58.51 delitos/hm ²	39.93 veces
92	450.31 delitos/hm ²	307.33 veces
93	25.53 delitos/hm ²	17.42 veces
94	59.92 delitos/hm ²	40.9 veces
95	7.9 delitos/hm ²	5.39 veces
96	7.93 delitos/hm ²	5.41 veces
97	234.01 delitos/hm ²	159.71 veces
98	361.2 delitos/hm ²	246.51 veces
99	848.42 delitos/hm ²	579.04 veces
100	8.66 delitos/hm ²	5.91 veces
101	30.55 delitos/hm ²	20.85 veces
102	1094.81 delitos/hm ²	747.2 veces
103	30.8 delitos/hm ²	21.02 veces
104	9.58 delitos/hm ²	6.53 veces
105	69.79 delitos/hm ²	47.63 veces
106	24.7 delitos/hm ²	16.86 veces
107	19.19 delitos/hm ²	13.1 veces
108	26.88 delitos/hm ²	18.34 veces

Grupo	Densidad Delictiva	Densidad Delictiva Relativa
109	13527.04 delitos/hm ²	9232.06 veces
110	71.19 delitos/hm ²	48.59 veces
111	8.76 delitos/hm ²	5.98 veces
112	73.73 delitos/hm ²	50.32 veces
113	27.91 delitos/hm ²	19.05 veces
114	127.33 delitos/hm ²	86.9 veces
115	81.0 delitos/hm ²	55.28 veces
116	2635.14 delitos/hm ²	1798.45 veces
117	101.02 delitos/hm ²	68.94 veces
118	39.93 delitos/hm ²	27.25 veces
119	126.98 delitos/hm ²	86.66 veces
120	42.62 delitos/hm ²	29.09 veces
121	13.88 delitos/hm ²	9.47 veces
122	48.04 delitos/hm ²	32.79 veces
123	2066779.5 delitos/hm ²	1410554 veces
124	10.1 delitos/hm ²	6.89 veces
125	24.33 delitos/hm ²	16.61 veces
126	22.36 delitos/hm ²	15.26 veces
127	18.31 delitos/hm ²	12.5 veces
128	61.88 delitos/hm ²	42.23 veces
129	4.8 delitos/hm ²	3.28 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Densidad Delictiva Relativa medida en base a la Densidad Delictiva del Distrito (DDD) que es 1.47 delitos/hm².

Como se puede observar en la Tabla 11, algunos grupos tienen una Densidad Delictiva atípica, lo que se debe a que el tamaño de estos grupos es menor a 100m² (0.01hm²), por lo que se procedió a filtrar dichos grupos y se obtuvo los grupos de la Tabla 12, con su respectiva densidad delictiva y densidad delictiva relativa.

Tabla 12

Densidades Delictivas del Agrupamiento DBSCAN (min_samples = 2, eps = 2.1) con grupos filtrados

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	0	14.51 delitos/hm ²	9.9 veces
1	1	114.57 delitos/hm ²	78.19 veces
2	2	39.22 delitos/hm ²	26.76 veces
3	3	31.97 delitos/hm ²	21.82 veces
4	4	16.63 delitos/hm ²	11.35 veces
7	5	65.8 delitos/hm ²	44.91 veces
8	6	283.24 delitos/hm ²	193.31 veces
9	7	65.18 delitos/hm ²	44.48 veces
11	8	15.31 delitos/hm ²	10.45 veces
12	9	20.14 delitos/hm ²	13.75 veces

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
13	10	201.9 delitos/hm ²	137.79 veces
14	11	7.18 delitos/hm ²	4.9 veces
15	12	33.61 delitos/hm ²	22.94 veces
19	13	32.94 delitos/hm ²	22.48 veces
20	14	22.6 delitos/hm ²	15.43 veces
21	15	12.19 delitos/hm ²	8.32 veces
22	16	12.56 delitos/hm ²	8.57 veces
23	17	7.43 delitos/hm ²	5.07 veces
24	18	12.04 delitos/hm ²	8.22 veces
25	19	67.06 delitos/hm ²	45.77 veces
26	20	23.34 delitos/hm ²	15.93 veces
28	21	11.97 delitos/hm ²	8.17 veces
30	22	55.84 delitos/hm ²	38.11 veces
33	23	25.05 delitos/hm ²	17.09 veces
34	24	392.43 delitos/hm ²	267.83 veces
35	25	14.26 delitos/hm ²	9.73 veces
36	26	59.69 delitos/hm ²	40.74 veces
39	27	22.01 delitos/hm ²	15.02 veces
40	28	97.0 delitos/hm ²	66.2 veces
41	29	190.12 delitos/hm ²	129.75 veces
45	30	7.57 delitos/hm ²	5.17 veces
47	31	48.18 delitos/hm ²	32.88 veces
50	32	56.67 delitos/hm ²	38.68 veces
51	33	64.63 delitos/hm ²	44.11 veces
53	34	14.18 delitos/hm ²	9.68 veces
54	35	39.76 delitos/hm ²	27.13 veces
55	36	70.25 delitos/hm ²	47.95 veces
57	37	170.01 delitos/hm ²	116.03 veces
59	38	31.21 delitos/hm ²	21.3 veces
62	39	14.07 delitos/hm ²	9.6 veces
63	40	15.61 delitos/hm ²	10.65 veces
64	41	149.1 delitos/hm ²	101.76 veces
65	42	50.99 delitos/hm ²	34.8 veces
66	43	122.47 delitos/hm ²	83.58 veces
68	44	394.57 delitos/hm ²	269.29 veces
69	45	97.38 delitos/hm ²	66.46 veces
70	46	49.14 delitos/hm ²	33.53 veces
71	47	21.06 delitos/hm ²	14.37 veces
72	48	22.41 delitos/hm ²	15.29 veces
73	49	15.74 delitos/hm ²	10.74 veces
74	50	69.05 delitos/hm ²	47.13 veces
75	51	20.41 delitos/hm ²	13.93 veces
76	52	28.85 delitos/hm ²	19.69 veces
77	53	8.3 delitos/hm ²	5.66 veces
78	54	15.01 delitos/hm ²	10.24 veces
79	55	211.7 delitos/hm ²	144.49 veces
80	56	134.68 delitos/hm ²	91.91 veces
81	57	43.36 delitos/hm ²	29.59 veces
82	58	160.8 delitos/hm ²	109.74 veces
84	59	55.76 delitos/hm ²	38.06 veces
85	60	10.34 delitos/hm ²	7.06 veces
86	61	47.74 delitos/hm ²	32.58 veces
87	62	60.97 delitos/hm ²	41.61 veces
88	63	57.12 delitos/hm ²	38.98 veces
89	64	13.27 delitos/hm ²	9.06 veces
91	65	58.51 delitos/hm ²	39.93 veces
93	66	25.53 delitos/hm ²	17.42 veces
94	67	59.92 delitos/hm ²	40.9 veces

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
95	68	7.9 delitos/hm ²	5.39 veces
96	69	7.93 delitos/hm ²	5.41 veces
97	70	234.01 delitos/hm ²	159.71 veces
100	71	8.66 delitos/hm ²	5.91 veces
101	72	30.55 delitos/hm ²	20.85 veces
103	73	30.8 delitos/hm ²	21.02 veces
104	74	9.58 delitos/hm ²	6.53 veces
105	75	69.79 delitos/hm ²	47.63 veces
106	76	24.7 delitos/hm ²	16.86 veces
107	77	19.19 delitos/hm ²	13.1 veces
108	78	26.88 delitos/hm ²	18.34 veces
110	79	71.19 delitos/hm ²	48.59 veces
111	80	8.76 delitos/hm ²	5.98 veces
112	81	73.73 delitos/hm ²	50.32 veces
113	82	27.91 delitos/hm ²	19.05 veces
114	83	127.33 delitos/hm ²	86.9 veces
115	84	81.0 delitos/hm ²	55.28 veces
117	85	101.02 delitos/hm ²	68.94 veces
118	86	39.93 delitos/hm ²	27.25 veces
119	87	126.98 delitos/hm ²	86.66 veces
120	88	42.62 delitos/hm ²	29.09 veces
121	89	13.88 delitos/hm ²	9.47 veces
122	90	48.04 delitos/hm ²	32.79 veces
124	91	10.1 delitos/hm ²	6.89 veces
125	92	24.33 delitos/hm ²	16.61 veces
126	93	22.36 delitos/hm ²	15.26 veces
127	94	18.31 delitos/hm ²	12.5 veces
128	95	61.88 delitos/hm ²	42.23 veces
129	96	4.8 delitos/hm ²	3.28 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Densidad Delictiva Relativa medida en base a la Densidad Delictiva del Distrito (DDD) que es 1.47 delitos/hm².

A partir de la Tabla 12, se obtuvieron 97 grupos y luego se calculó la Densidad Delictiva Acumulada para el Agrupamiento DBSCAN (min_samples=2, eps= 2.10) resultando el valor de 22.16 delitos/hm² y la Densidad Delictiva Acumulada relativa de 15.13 veces la Densidad Delictiva del Distrito (1.47 delitos/hm²).

Densidad Delictiva del Agrupamiento HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2)

La Densidad Delictiva de cada grupo según el agrupamiento HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2) se muestra en la Tabla 13.

Tabla 13

Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2)

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
0	170.01 delitos/hm ²	116.03 veces
1	1614.63 delitos/hm ²	1101.97 veces
2	2635.14 delitos/hm ²	1798.45 veces
3	22441.17 delitos/hm ²	15315.86 veces
4	31.21 delitos/hm ²	21.3 veces
5	3488.35 delitos/hm ²	2380.76 veces
6	8846.02 delitos/hm ²	6037.31 veces
7	70.25 delitos/hm ²	47.95 veces
8	50.99 delitos/hm ²	34.8 veces
9	55.76 delitos/hm ²	38.06 veces
10	149.1 delitos/hm ²	101.76 veces
11	4.18 delitos/hm ²	2.86 veces
12	1267.42 delitos/hm ²	865.0 veces
13	97.38 delitos/hm ²	66.46 veces
14	8.26 delitos/hm ²	5.63 veces
15	63.21 delitos/hm ²	43.14 veces
16	3285.92 delitos/hm ²	2242.61 veces
17	18.24 delitos/hm ²	12.45 veces
18	19.33 delitos/hm ²	13.19 veces
19	7.62 delitos/hm ²	5.2 veces
20	31.97 delitos/hm ²	21.82 veces
21	714.46 delitos/hm ²	487.61 veces
22	392.43 delitos/hm ²	267.83 veces
23	114.57 delitos/hm ²	78.19 veces
24	2.04 delitos/hm ²	1.39 veces
25	31.02 delitos/hm ²	21.17 veces
26	27.91 delitos/hm ²	19.05 veces
27	5.88 delitos/hm ²	4.01 veces
28	99.43 delitos/hm ²	67.86 veces
29	61.88 delitos/hm ²	42.23 veces
30	127.33 delitos/hm ²	86.9 veces
31	81.0 delitos/hm ²	55.28 veces
32	43.36 delitos/hm ²	29.59 veces
33	48.04 delitos/hm ²	32.79 veces
34	8.62 delitos/hm ²	5.88 veces
35	2089.12 delitos/hm ²	1425.8 veces
36	59.92 delitos/hm ²	40.9 veces
37	848.42 delitos/hm ²	579.04 veces
38	2166.25 delitos/hm ²	1478.44 veces
39	39.93 delitos/hm ²	27.25 veces
40	13.13 delitos/hm ²	8.96 veces
41	404.98 delitos/hm ²	276.39 veces
42	33127.45 delitos/hm ²	22609.12 veces
43	58.73 delitos/hm ²	40.08 veces
44	1758.66 delitos/hm ²	1200.27 veces
45	6.35 delitos/hm ²	4.33 veces
46	7.71 delitos/hm ²	5.27 veces
47	10.21 delitos/hm ²	6.97 veces
48	21.39 delitos/hm ²	14.6 veces
49	11.66 delitos/hm ²	7.96 veces
50	2.86 delitos/hm ²	1.95 veces

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
51	6.79 delitos/hm ²	4.63 veces
52	8.37 delitos/hm ²	5.71 veces
53	39.22 delitos/hm ²	26.76 veces
54	15.17 delitos/hm ²	10.36 veces
55	39.57 delitos/hm ²	27.01 veces
56	26.88 delitos/hm ²	18.34 veces
57	19.62 delitos/hm ²	13.39 veces
58	1113.34 delitos/hm ²	759.84 veces
59	7.9 delitos/hm ²	5.39 veces
60	19022.4 delitos/hm ²	12982.58 veces
61	1094.81 delitos/hm ²	747.2 veces
62	30.8 delitos/hm ²	21.02 veces
63	71.19 delitos/hm ²	48.59 veces
64	103611.36 delitos/hm ²	70713.63 veces
65	26.01 delitos/hm ²	17.75 veces
66	75.39 delitos/hm ²	51.45 veces
67	4660.03 delitos/hm ²	3180.42 veces
68	13527.04 delitos/hm ²	9232.06 veces
69	10.11 delitos/hm ²	6.9 veces
70	536.85 delitos/hm ²	366.39 veces
71	68.62 delitos/hm ²	46.83 veces
72	1664.3 delitos/hm ²	1135.87 veces
73	56.67 delitos/hm ²	38.68 veces
74	481.48 delitos/hm ²	328.61 veces
75	11.57 delitos/hm ²	7.89 veces
76	145.14 delitos/hm ²	99.06 veces
77	73.1 delitos/hm ²	49.89 veces
78	22.55 delitos/hm ²	15.39 veces
79	2.49 delitos/hm ²	1.7 veces
80	11.89 delitos/hm ²	8.11 veces
81	988.98 delitos/hm ²	674.97 veces
82	56.34 delitos/hm ²	38.45 veces
83	76.44 delitos/hm ²	52.17 veces
84	69.05 delitos/hm ²	47.13 veces
85	211.7 delitos/hm ²	144.49 veces
86	10.34 delitos/hm ²	7.06 veces
87	17.96 delitos/hm ²	12.26 veces
88	8.73 delitos/hm ²	5.96 veces
89	1239.75 delitos/hm ²	846.12 veces
90	22.41 delitos/hm ²	15.29 veces
91	9.61 delitos/hm ²	6.56 veces
92	97.0 delitos/hm ²	66.2 veces
93	25.53 delitos/hm ²	17.42 veces
94	7.48 delitos/hm ²	5.11 veces
95	30.33 delitos/hm ²	20.7 veces
96	22.01 delitos/hm ²	15.02 veces
97	22.96 delitos/hm ²	15.67 veces
98	12.04 delitos/hm ²	8.22 veces
99	17.74 delitos/hm ²	12.11 veces
100	16.99 delitos/hm ²	11.59 veces
101	42.02 delitos/hm ²	28.68 veces
102	36.89 delitos/hm ²	25.17 veces
103	64.63 delitos/hm ²	44.11 veces
104	36.1 delitos/hm ²	24.64 veces
105	1306.19 delitos/hm ²	891.46 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

Como se puede observar en la Tabla 13, algunos grupos tienen una Densidad Delictiva atípica, lo que se debe a que el tamaño de estos grupos es menor a un área mínima requerida (0.01hm^2), por lo que se procedió a filtrar dichos grupos y se obtuvo los grupos de la Tabla 14, con su respectiva densidad delictiva y densidad delictiva relativa.

Tabla 14

Densidades Delictivas del Agrupamiento HDBSCAN ($\text{min_cluster_size} = 2$, $\text{min_samples} = 2$, $\text{cluster_selection_epsilon} = 2$) con grupos filtrados

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	0	170.01 delitos/hm ²	116.03 veces
4	1	31.21 delitos/hm ²	21.3 veces
7	2	70.25 delitos/hm ²	47.95 veces
8	3	50.99 delitos/hm ²	34.8 veces
9	4	55.76 delitos/hm ²	38.06 veces
10	5	149.1 delitos/hm ²	101.76 veces
11	6	4.18 delitos/hm ²	2.86 veces
13	7	97.38 delitos/hm ²	66.46 veces
14	8	8.26 delitos/hm ²	5.63 veces
15	9	63.21 delitos/hm ²	43.14 veces
17	10	18.24 delitos/hm ²	12.45 veces
18	11	19.33 delitos/hm ²	13.19 veces
19	12	7.62 delitos/hm ²	5.2 veces
20	13	31.97 delitos/hm ²	21.82 veces
22	14	392.43 delitos/hm ²	267.83 veces
23	15	114.57 delitos/hm ²	78.19 veces
24	16	2.04 delitos/hm ²	1.39 veces
25	17	31.02 delitos/hm ²	21.17 veces
26	18	27.91 delitos/hm ²	19.05 veces
27	19	5.88 delitos/hm ²	4.01 veces
28	20	99.43 delitos/hm ²	67.86 veces
29	21	61.88 delitos/hm ²	42.23 veces
30	22	127.33 delitos/hm ²	86.9 veces
31	23	81.0 delitos/hm ²	55.28 veces
32	24	43.36 delitos/hm ²	29.59 veces
33	25	48.04 delitos/hm ²	32.79 veces
34	26	8.62 delitos/hm ²	5.88 veces
36	27	59.92 delitos/hm ²	40.9 veces
39	28	39.93 delitos/hm ²	27.25 veces
40	29	13.13 delitos/hm ²	8.96 veces
43	30	58.73 delitos/hm ²	40.08 veces
45	31	6.35 delitos/hm ²	4.33 veces
46	32	7.71 delitos/hm ²	5.27 veces
47	33	10.21 delitos/hm ²	6.97 veces
48	34	21.39 delitos/hm ²	14.6 veces
49	35	11.66 delitos/hm ²	7.96 veces
50	36	2.86 delitos/hm ²	1.95 veces
51	37	6.79 delitos/hm ²	4.63 veces

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
52	38	8.37 delitos/hm ²	5.71 veces
53	39	39.22 delitos/hm ²	26.76 veces
54	40	15.17 delitos/hm ²	10.36 veces
55	41	39.57 delitos/hm ²	27.01 veces
56	42	26.88 delitos/hm ²	18.34 veces
57	43	19.62 delitos/hm ²	13.39 veces
59	44	7.9 delitos/hm ²	5.39 veces
62	45	30.8 delitos/hm ²	21.02 veces
63	46	71.19 delitos/hm ²	48.59 veces
65	47	26.01 delitos/hm ²	17.75 veces
66	48	75.39 delitos/hm ²	51.45 veces
69	49	10.11 delitos/hm ²	6.9 veces
71	50	68.62 delitos/hm ²	46.83 veces
73	51	56.67 delitos/hm ²	38.68 veces
75	52	11.57 delitos/hm ²	7.89 veces
76	53	145.14 delitos/hm ²	99.06 veces
77	54	73.1 delitos/hm ²	49.89 veces
78	55	22.55 delitos/hm ²	15.39 veces
79	56	2.49 delitos/hm ²	1.7 veces
80	57	11.89 delitos/hm ²	8.11 veces
82	58	56.34 delitos/hm ²	38.45 veces
83	59	76.44 delitos/hm ²	52.17 veces
84	60	69.05 delitos/hm ²	47.13 veces
85	61	211.7 delitos/hm ²	144.49 veces
86	62	10.34 delitos/hm ²	7.06 veces
87	63	17.96 delitos/hm ²	12.26 veces
88	64	8.73 delitos/hm ²	5.96 veces
90	65	22.41 delitos/hm ²	15.29 veces
91	66	9.61 delitos/hm ²	6.56 veces
92	67	97.0 delitos/hm ²	66.2 veces
93	68	25.53 delitos/hm ²	17.42 veces
94	69	7.48 delitos/hm ²	5.11 veces
95	70	30.33 delitos/hm ²	20.7 veces
96	71	22.01 delitos/hm ²	15.02 veces
97	72	22.96 delitos/hm ²	15.67 veces
98	73	12.04 delitos/hm ²	8.22 veces
99	74	17.74 delitos/hm ²	12.11 veces
100	75	16.99 delitos/hm ²	11.59 veces
101	76	42.02 delitos/hm ²	28.68 veces
102	77	36.89 delitos/hm ²	25.17 veces
103	78	64.63 delitos/hm ²	44.11 veces
104	79	36.1 delitos/hm ²	24.64 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

A partir de la Tabla 14, se obtuvieron 80 grupos y luego se calculó la Densidad Delictiva Acumulada para el Agrupamiento HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2) resultando el valor de 16.14 delitos/hm² y la Densidad Delictiva Acumulada relativa de 11.02 veces la Densidad Delictiva del Distrito (1.47 delitos/hm²).

***Densidad Delictiva del Agrupamiento HDBSCAN (min_cluster_size = 3,
min_samples = 2, cluster_selection_epsilon = 2)***

La Densidad Delictiva de cada grupo según el agrupamiento HDBSCAN (min_cluster_size = 3, min_samples = 2, cluster_selection_epsilon = 2) se muestra en la Tabla 15.

Tabla 15

*Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 3,
min_samples = 2, cluster_selection_epsilon = 2)*

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
0	170.01 delitos/hm ²	116.03 veces
1	1614.63 delitos/hm ²	1101.97 veces
2	2635.14 delitos/hm ²	1798.45 veces
3	22441.17 delitos/hm ²	15315.86 veces
4	31.21 delitos/hm ²	21.3 veces
5	3488.35 delitos/hm ²	2380.76 veces
6	8846.02 delitos/hm ²	6037.31 veces
7	70.25 delitos/hm ²	47.95 veces
8	50.99 delitos/hm ²	34.8 veces
9	55.76 delitos/hm ²	38.06 veces
10	149.1 delitos/hm ²	101.76 veces
11	4.18 delitos/hm ²	2.86 veces
12	1267.42 delitos/hm ²	865.0 veces
13	97.38 delitos/hm ²	66.46 veces
14	8.26 delitos/hm ²	5.63 veces
15	63.21 delitos/hm ²	43.14 veces
16	3285.92 delitos/hm ²	2242.61 veces
17	18.24 delitos/hm ²	12.45 veces
18	19.33 delitos/hm ²	13.19 veces
19	7.62 delitos/hm ²	5.2 veces
20	31.97 delitos/hm ²	21.82 veces
21	714.46 delitos/hm ²	487.61 veces
22	392.43 delitos/hm ²	267.83 veces
23	114.57 delitos/hm ²	78.19 veces
24	31.02 delitos/hm ²	21.17 veces
25	27.91 delitos/hm ²	19.05 veces
26	5.88 delitos/hm ²	4.01 veces
27	99.43 delitos/hm ²	67.86 veces
28	61.88 delitos/hm ²	42.23 veces
29	127.33 delitos/hm ²	86.9 veces
30	81.0 delitos/hm ²	55.28 veces
31	48.04 delitos/hm ²	32.79 veces
32	8.62 delitos/hm ²	5.88 veces
33	2089.12 delitos/hm ²	1425.8 veces
34	59.92 delitos/hm ²	40.9 veces
35	848.42 delitos/hm ²	579.04 veces
36	2166.25 delitos/hm ²	1478.44 veces
37	39.93 delitos/hm ²	27.25 veces
38	13.13 delitos/hm ²	8.96 veces

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
39	404.98 delitos/hm ²	276.39 veces
40	33127.45 delitos/hm ²	22609.12 veces
41	58.73 delitos/hm ²	40.08 veces
42	1758.66 delitos/hm ²	1200.27 veces
43	6.35 delitos/hm ²	4.33 veces
44	7.71 delitos/hm ²	5.27 veces
45	10.21 delitos/hm ²	6.97 veces
46	21.39 delitos/hm ²	14.6 veces
47	18.91 delitos/hm ²	12.91 veces
48	11.66 delitos/hm ²	7.96 veces
49	8.37 delitos/hm ²	5.71 veces
50	39.22 delitos/hm ²	26.76 veces
51	15.17 delitos/hm ²	10.36 veces
52	39.57 delitos/hm ²	27.01 veces
53	26.88 delitos/hm ²	18.34 veces
54	19.62 delitos/hm ²	13.39 veces
55	1113.34 delitos/hm ²	759.84 veces
56	7.9 delitos/hm ²	5.39 veces
57	19022.4 delitos/hm ²	12982.58 veces
58	1094.81 delitos/hm ²	747.2 veces
59	30.8 delitos/hm ²	21.02 veces
60	71.19 delitos/hm ²	48.59 veces
61	103611.36 delitos/hm ²	70713.63 veces
62	26.01 delitos/hm ²	17.75 veces
63	75.39 delitos/hm ²	51.45 veces
64	4660.03 delitos/hm ²	3180.42 veces
65	13527.04 delitos/hm ²	9232.06 veces
66	10.11 delitos/hm ²	6.9 veces
67	536.85 delitos/hm ²	366.39 veces
68	68.62 delitos/hm ²	46.83 veces
69	1664.3 delitos/hm ²	1135.87 veces
70	56.67 delitos/hm ²	38.68 veces
71	481.48 delitos/hm ²	328.61 veces
72	11.57 delitos/hm ²	7.89 veces
73	145.14 delitos/hm ²	99.06 veces
74	73.1 delitos/hm ²	49.89 veces
75	22.55 delitos/hm ²	15.39 veces
76	11.89 delitos/hm ²	8.11 veces
77	988.98 delitos/hm ²	674.97 veces
78	7.99 delitos/hm ²	5.45 veces
79	56.34 delitos/hm ²	38.45 veces
80	76.44 delitos/hm ²	52.17 veces
81	69.05 delitos/hm ²	47.13 veces
82	211.7 delitos/hm ²	144.49 veces
83	8.94 delitos/hm ²	6.1 veces
84	8.73 delitos/hm ²	5.96 veces
85	1239.75 delitos/hm ²	846.12 veces
86	22.41 delitos/hm ²	15.29 veces
87	97.0 delitos/hm ²	66.2 veces
88	25.53 delitos/hm ²	17.42 veces
89	22.01 delitos/hm ²	15.02 veces
90	22.96 delitos/hm ²	15.67 veces
91	12.04 delitos/hm ²	8.22 veces
92	17.74 delitos/hm ²	12.11 veces
93	16.99 delitos/hm ²	11.59 veces
94	42.02 delitos/hm ²	28.68 veces
95	36.89 delitos/hm ²	25.17 veces
96	64.63 delitos/hm ²	44.11 veces

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
97	36.1 delitos/hm ²	24.64 veces
98	1306.19 delitos/hm ²	891.46 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

Como se puede observar en la Tabla 15, algunos grupos tienen una Densidad Delictiva atípica, lo que se debe a que el tamaño de estos grupos es menor a un área mínima requerida (0.01hm²), por lo que se procedió a filtrar dichos grupos y se obtuvo los grupos de la Tabla 16, con su respectiva densidad delictiva y densidad delictiva relativa.

Tabla 16

Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 3, min_samples = 2, cluster_selection_epsilon = 2) con grupos filtrados

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	0	170.01 delitos/hm ²	116.03 veces
4	1	31.21 delitos/hm ²	21.3 veces
7	2	70.25 delitos/hm ²	47.95 veces
8	3	50.99 delitos/hm ²	34.8 veces
9	4	55.76 delitos/hm ²	38.06 veces
10	5	149.1 delitos/hm ²	101.76 veces
11	6	4.18 delitos/hm ²	2.86 veces
13	7	97.38 delitos/hm ²	66.46 veces
14	8	8.26 delitos/hm ²	5.63 veces
15	9	63.21 delitos/hm ²	43.14 veces
17	10	18.24 delitos/hm ²	12.45 veces
18	11	19.33 delitos/hm ²	13.19 veces
19	12	7.62 delitos/hm ²	5.2 veces
20	13	31.97 delitos/hm ²	21.82 veces
22	14	392.43 delitos/hm ²	267.83 veces
23	15	114.57 delitos/hm ²	78.19 veces
24	16	31.02 delitos/hm ²	21.17 veces
25	17	27.91 delitos/hm ²	19.05 veces
26	18	5.88 delitos/hm ²	4.01 veces
27	19	99.43 delitos/hm ²	67.86 veces
28	20	61.88 delitos/hm ²	42.23 veces
29	21	127.33 delitos/hm ²	86.9 veces
30	22	81.0 delitos/hm ²	55.28 veces
31	23	48.04 delitos/hm ²	32.79 veces
32	24	8.62 delitos/hm ²	5.88 veces
34	25	59.92 delitos/hm ²	40.9 veces
37	26	39.93 delitos/hm ²	27.25 veces
38	27	13.13 delitos/hm ²	8.96 veces
41	28	58.73 delitos/hm ²	40.08 veces
43	29	6.35 delitos/hm ²	4.33 veces

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
44	30	7.71 delitos/hm ²	5.27 veces
45	31	10.21 delitos/hm ²	6.97 veces
46	32	21.39 delitos/hm ²	14.6 veces
47	33	18.91 delitos/hm ²	12.91 veces
48	34	11.66 delitos/hm ²	7.96 veces
49	35	8.37 delitos/hm ²	5.71 veces
50	36	39.22 delitos/hm ²	26.76 veces
51	37	15.17 delitos/hm ²	10.36 veces
52	38	39.57 delitos/hm ²	27.01 veces
53	39	26.88 delitos/hm ²	18.34 veces
54	40	19.62 delitos/hm ²	13.39 veces
56	41	7.9 delitos/hm ²	5.39 veces
59	42	30.8 delitos/hm ²	21.02 veces
60	43	71.19 delitos/hm ²	48.59 veces
62	44	26.01 delitos/hm ²	17.75 veces
63	45	75.39 delitos/hm ²	51.45 veces
66	46	10.11 delitos/hm ²	6.9 veces
68	47	68.62 delitos/hm ²	46.83 veces
70	48	56.67 delitos/hm ²	38.68 veces
72	49	11.57 delitos/hm ²	7.89 veces
73	50	145.14 delitos/hm ²	99.06 veces
74	51	73.1 delitos/hm ²	49.89 veces
75	52	22.55 delitos/hm ²	15.39 veces
76	53	11.89 delitos/hm ²	8.11 veces
78	54	7.99 delitos/hm ²	5.45 veces
79	55	56.34 delitos/hm ²	38.45 veces
80	56	76.44 delitos/hm ²	52.17 veces
81	57	69.05 delitos/hm ²	47.13 veces
82	58	211.7 delitos/hm ²	144.49 veces
83	59	8.94 delitos/hm ²	6.1 veces
84	60	8.73 delitos/hm ²	5.96 veces
86	61	22.41 delitos/hm ²	15.29 veces
87	62	97.0 delitos/hm ²	66.2 veces
88	63	25.53 delitos/hm ²	17.42 veces
89	64	22.01 delitos/hm ²	15.02 veces
90	65	22.96 delitos/hm ²	15.67 veces
91	66	12.04 delitos/hm ²	8.22 veces
92	67	17.74 delitos/hm ²	12.11 veces
93	68	16.99 delitos/hm ²	11.59 veces
94	69	42.02 delitos/hm ²	28.68 veces
95	70	36.89 delitos/hm ²	25.17 veces
96	71	64.63 delitos/hm ²	44.11 veces
97	72	36.1 delitos/hm ²	24.64 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

A partir de la Tabla 16, se obtuvieron 73 grupos y luego se calculó la Densidad Delictiva Acumulada para el Agrupamiento HDBSCAN (`min_cluster_size = 3`, `min_samples = 2`, `cluster_selection_epsilon = 2`) resultando el valor de 16.92

delitos/hm² y la Densidad Delictiva Acumulada relativa de 11.55 veces la Densidad Delictiva del Distrito (1.47 delitos/hm²).

Densidad Delictiva del Agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85)

La Densidad Delictiva de cada grupo según el agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85) se muestra en la Tabla 17.

Tabla 17

Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85)

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
0	31.21 delitos/hm ²	21.3 veces
1	8846.02 delitos/hm ²	6037.31 veces
2	70.25 delitos/hm ²	47.95 veces
3	50.99 delitos/hm ²	34.8 veces
4	55.76 delitos/hm ²	38.06 veces
5	149.1 delitos/hm ²	101.76 veces
6	4.18 delitos/hm ²	2.86 veces
7	1267.42 delitos/hm ²	865.0 veces
8	97.38 delitos/hm ²	66.46 veces
9	8.26 delitos/hm ²	5.63 veces
10	18.24 delitos/hm ²	12.45 veces
11	19.33 delitos/hm ²	13.19 veces
12	18.22 delitos/hm ²	12.43 veces
13	7.62 delitos/hm ²	5.2 veces
14	392.43 delitos/hm ²	267.83 veces
15	3.09 delitos/hm ²	2.11 veces
16	3.42 delitos/hm ²	2.33 veces
17	27.91 delitos/hm ²	19.05 veces
18	6.76 delitos/hm ²	4.61 veces
19	6.05 delitos/hm ²	4.13 veces
20	59.92 delitos/hm ²	40.9 veces
21	848.42 delitos/hm ²	579.04 veces
22	13.13 delitos/hm ²	8.96 veces
23	1758.66 delitos/hm ²	1200.27 veces
24	6.35 delitos/hm ²	4.33 veces
25	7.71 delitos/hm ²	5.27 veces
26	21.39 delitos/hm ²	14.6 veces
27	18.91 delitos/hm ²	12.91 veces
28	11.66 delitos/hm ²	7.96 veces
29	8.37 delitos/hm ²	5.71 veces
30	39.22 delitos/hm ²	26.76 veces
31	15.17 delitos/hm ²	10.36 veces

Grupo	Densidad Delictiva según HDBSCAN	Densidad Delictiva Relativa
32	39.57 delitos/hm ²	27.01 veces
33	26.88 delitos/hm ²	18.34 veces
34	13.33 delitos/hm ²	9.1 veces
35	19.62 delitos/hm ²	13.39 veces
36	7.9 delitos/hm ²	5.39 veces
37	71.19 delitos/hm ²	48.59 veces
38	7.17 delitos/hm ²	4.89 veces
39	26.01 delitos/hm ²	17.75 veces
40	7.65 delitos/hm ²	5.22 veces
41	75.39 delitos/hm ²	51.45 veces
42	10.55 delitos/hm ²	7.2 veces
43	11.57 delitos/hm ²	7.89 veces
44	73.1 delitos/hm ²	49.89 veces
45	22.55 delitos/hm ²	15.39 veces
46	5.82 delitos/hm ²	3.97 veces
47	6.01 delitos/hm ²	4.1 veces
48	7.99 delitos/hm ²	5.45 veces
49	69.05 delitos/hm ²	47.13 veces
50	211.7 delitos/hm ²	144.49 veces
51	17.02 delitos/hm ²	11.62 veces
52	8.94 delitos/hm ²	6.1 veces
53	8.73 delitos/hm ²	5.96 veces
54	97.0 delitos/hm ²	66.2 veces
55	25.53 delitos/hm ²	17.42 veces
56	22.01 delitos/hm ²	15.02 veces
57	22.96 delitos/hm ²	15.67 veces
58	12.04 delitos/hm ²	8.22 veces
59	17.74 delitos/hm ²	12.11 veces
60	16.99 delitos/hm ²	11.59 veces
61	42.02 delitos/hm ²	28.68 veces
62	6.11 delitos/hm ²	4.17 veces
63	36.89 delitos/hm ²	25.17 veces
64	64.63 delitos/hm ²	44.11 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

Como se puede observar en la Tabla 17, algunos grupos tienen una Densidad Delictiva atípica, lo que se debe a que el tamaño de estos grupos es menor a un área mínima requerida (0.01hm²), por lo que se procedió a filtrar dichos grupos y se obtuvo los grupos de la Tabla 18, con su respectiva densidad delictiva y densidad delictiva relativa.

Tabla 18

Densidades Delictivas del Agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85) con grupos filtrados

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
0	0	31.21 delitos/hm ²	21.3 veces
2	1	70.25 delitos/hm ²	47.95 veces
3	2	50.99 delitos/hm ²	34.8 veces
4	3	55.76 delitos/hm ²	38.06 veces
5	4	149.1 delitos/hm ²	101.76 veces
6	5	4.18 delitos/hm ²	2.86 veces
8	6	97.38 delitos/hm ²	66.46 veces
9	7	8.26 delitos/hm ²	5.63 veces
10	8	18.24 delitos/hm ²	12.45 veces
11	9	19.33 delitos/hm ²	13.19 veces
12	10	18.22 delitos/hm ²	12.43 veces
13	11	7.62 delitos/hm ²	5.2 veces
14	12	392.43 delitos/hm ²	267.83 veces
15	13	3.09 delitos/hm ²	2.11 veces
16	14	3.42 delitos/hm ²	2.33 veces
17	15	27.91 delitos/hm ²	19.05 veces
18	16	6.76 delitos/hm ²	4.61 veces
19	17	6.05 delitos/hm ²	4.13 veces
20	18	59.92 delitos/hm ²	40.9 veces
22	19	13.13 delitos/hm ²	8.96 veces
24	20	6.35 delitos/hm ²	4.33 veces
25	21	7.71 delitos/hm ²	5.27 veces
26	22	21.39 delitos/hm ²	14.6 veces
27	23	18.91 delitos/hm ²	12.91 veces
28	24	11.66 delitos/hm ²	7.96 veces
29	25	8.37 delitos/hm ²	5.71 veces
30	26	39.22 delitos/hm ²	26.76 veces
31	27	15.17 delitos/hm ²	10.36 veces
32	28	39.57 delitos/hm ²	27.01 veces
33	29	26.88 delitos/hm ²	18.34 veces
34	30	13.33 delitos/hm ²	9.1 veces
35	31	19.62 delitos/hm ²	13.39 veces
36	32	7.9 delitos/hm ²	5.39 veces
37	33	71.19 delitos/hm ²	48.59 veces
38	34	7.17 delitos/hm ²	4.89 veces
39	35	26.01 delitos/hm ²	17.75 veces
40	36	7.65 delitos/hm ²	5.22 veces
41	37	75.39 delitos/hm ²	51.45 veces
42	38	10.55 delitos/hm ²	7.2 veces
43	39	11.57 delitos/hm ²	7.89 veces
44	40	73.1 delitos/hm ²	49.89 veces
45	41	22.55 delitos/hm ²	15.39 veces
46	42	5.82 delitos/hm ²	3.97 veces
47	43	6.01 delitos/hm ²	4.1 veces
48	44	7.99 delitos/hm ²	5.45 veces
49	45	69.05 delitos/hm ²	47.13 veces
50	46	211.7 delitos/hm ²	144.49 veces
51	47	17.02 delitos/hm ²	11.62 veces
52	48	8.94 delitos/hm ²	6.1 veces
53	49	8.73 delitos/hm ²	5.96 veces
54	50	97.0 delitos/hm ²	66.2 veces

ID	Grupo	Densidad Delictiva	Densidad Delictiva Relativa
55	51	25.53 delitos/hm ²	17.42 veces
56	52	22.01 delitos/hm ²	15.02 veces
57	53	22.96 delitos/hm ²	15.67 veces
58	54	12.04 delitos/hm ²	8.22 veces
59	55	17.74 delitos/hm ²	12.11 veces
60	56	16.99 delitos/hm ²	11.59 veces
61	57	42.02 delitos/hm ²	28.68 veces
62	58	6.11 delitos/hm ²	4.17 veces
63	59	36.89 delitos/hm ²	25.17 veces
64	60	64.63 delitos/hm ²	44.11 veces

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Elaboración Propia.

A partir de la Tabla 18, se obtuvieron 61 grupos y luego se calculó la Densidad Delictiva Acumulada para el Agrupamiento HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85) resultando el valor de 10.36 delitos/hm² y la Densidad Delictiva Acumulada relativa de 7.07 veces la Densidad Delictiva del Distrito (1.47 delitos/hm²).

Comparación de los resultados obtenidos en los distintos agrupamientos.

A continuación, se procedió a comparar los indicadores de densidad delictiva de los agrupamientos obtenidos por los Algoritmos K-Medias (K=3), K-Medias (K=17), DBSCAN (min_samples=2, eps = 2.1), HDBSCAN (m_c_size = 2, min_samples = 2, c_s_epsilon = 2), HDBSCAN (m_c_size = 3, min_samples = 2, c_s_epsilon = 2), HDBSCAN (m_c_size = 4, min_samples = 2, c_s_epsilon = 1.85). Los resultados se muestran en la Tabla 19.

Tabla 19

Análisis Comparativo de las Densidades Delictivas de los Algoritmos K-Medias, DBSCAN y HDBSCAN

ALGORITMO	K-Medias (K=3)	K-Medias (K=17)	DBSCAN (min_samples=2, eps = 2.1)	HDBSCAN (m_c_size = 2, min_samples = 2, c_s_epsilon = 2)	HDBSCAN (m_c_size = 3, min_samples = 2, c_s_epsilon = 2)	HDBSCAN (m_c_size = 4, min_samples = 2, c_s_epsilon = 1.85)
# DE GRUPOS	3	17	97	80	73	61
# DE REGISTROS RUIDO	0	0	189	185	189	152
DENSIDAD DELICTIVA ACUMULADA	1.54 delitos/hm ²	2.43 delitos/hm ²	22.16 delitos/hm ²	16.14 delitos/hm ²	16.92 delitos/hm ²	10.36 delitos/hm ²
DENSIDAD DELICTIVA RELATIVA	1.05 veces	1.66 veces	15.13 veces	11.02 veces	11.55 veces	7.07 veces
ÁREA ACUMULADA DE LOS GRUPOS	449.66 hm ²	293.2 hm ²	22.7 hm ²	31.41 hm ²	29.73 hm ²	52.14 hm ²
% DEL ÁREA ACUMULADA DE LOS GRUPOS	95.2%	62.1%	4.8%	6.7%	6.3%	11.0%
PROPORCIÓN ACUMULATIVA DE DELINCUENCIA	95.2% área → 100.0% delitos	62.1% área → 100.0% delitos	4.8% área → 72.7%delitos	6.7% área → 73.3% delitos	6.3% área → 72.7% delitos	11.0% área → 78.0% delitos

Nota. Métricas obtenidas a partir del software Jupyter Notebook. Los términos “c_s_epsilon” y “m_c_size” se refieren a los parámetros de “cluster_selection_epsilon” y “min_cluster_size” de HDBSCAN respectivamente. Elaboración Propia.

En base a los resultados obtenidos de los resultados de los algoritmos de agrupamiento, se puede observar que las agrupaciones realizadas por el algoritmo K-Medias, no obtuvieron resultados adecuados debido a que su **Densidad Delictiva Acumulada de 1.54 delitos/hm² (K=3) y 2.43 delitos/hm² (K=17)** y su **Densidad Delictiva Relativa Acumulada es 1.05 veces (K=3) y 1.66 veces (K=17)** la Densidad Delictiva Promedio del distrito, además de considerar que las agrupaciones

involucran el 95.2% y 62.1% del área del distrito, siendo estas áreas inespecíficas para el caso de estudio.

Respecto a los resultados del Algoritmo DBSCAN, se obtiene un mejor resultado tanto en el porcentaje de área acumulada (4.8%) así como del porcentaje de delitos agrupados con valor de 72.7% de delitos totales, con una Densidad Delictiva Acumulada de 22.16 delitos/hm² y una Densidad Delictiva Acumulada Relativa de 15.13 veces la Densidad Delictiva Promedio del distrito.

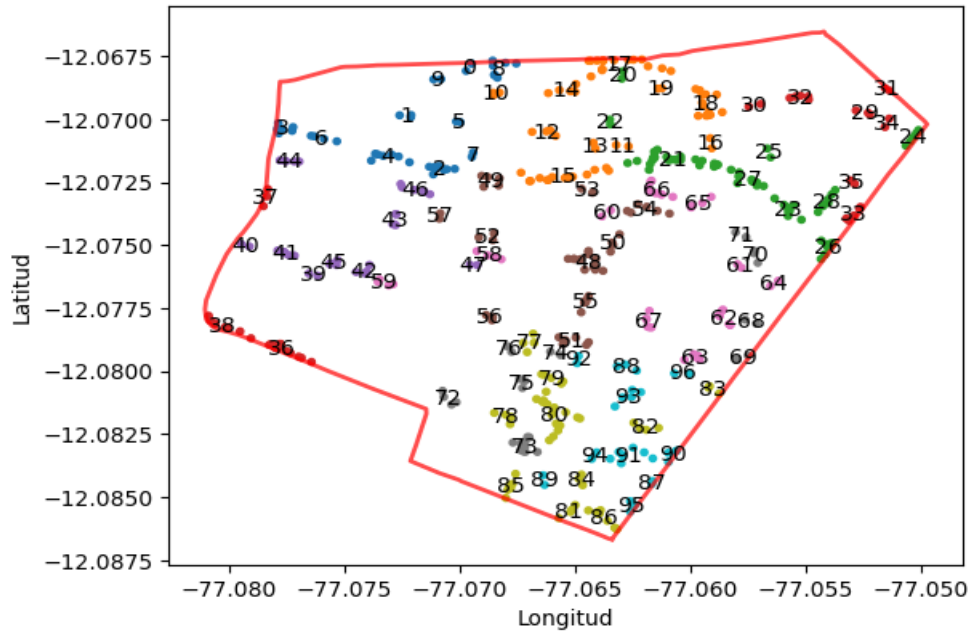
Por otro lado, los resultados del Algoritmo HDBSCAN, se obtiene un peor resultado, siendo el más cercano al resultado del Algoritmo DBSCAN el agrupamiento con los parámetros *min_cluster_size* = 3, *min_samples* = 2, *cluster_selection_epsilon* = 2, con un porcentaje de área acumulada (4.8%) así como del porcentaje de delitos del 72.7%, con una Densidad Delictiva Acumulada de 16.92 delitos/hm² y una Densidad Delictiva Acumulada Relativa de 11.02 veces la Densidad Delictiva Promedio del distrito.

3.4.3.5. Elección del agrupamiento óptimo y determinación de las zonas críticas de delincuencia.

En función de los resultados anteriores, se determinó al algoritmo **DBSCAN (min_samples=2, eps = 2.1)** como el algoritmo óptimo para la determinación de las zonas críticas de delincuencia. Luego, se graficó los grupos en el mapa de puntos de los delitos agrupados por el algoritmo DBSCAN (min_samples=2, eps = 2.1), ver la Figura 28. Posteriormente se calculó sus respectivos centroides, los resultados se muestran en la Figura 29 y en la Tabla 20. El código utilizado tanto para la preparación de la información de las Zonas Críticas de Delincuencia, la generación del mapa de puntos delictivos y los respectivos centroides de cada zona crítica de delincuencia se muestra en el Anexo 6.

Figura 28

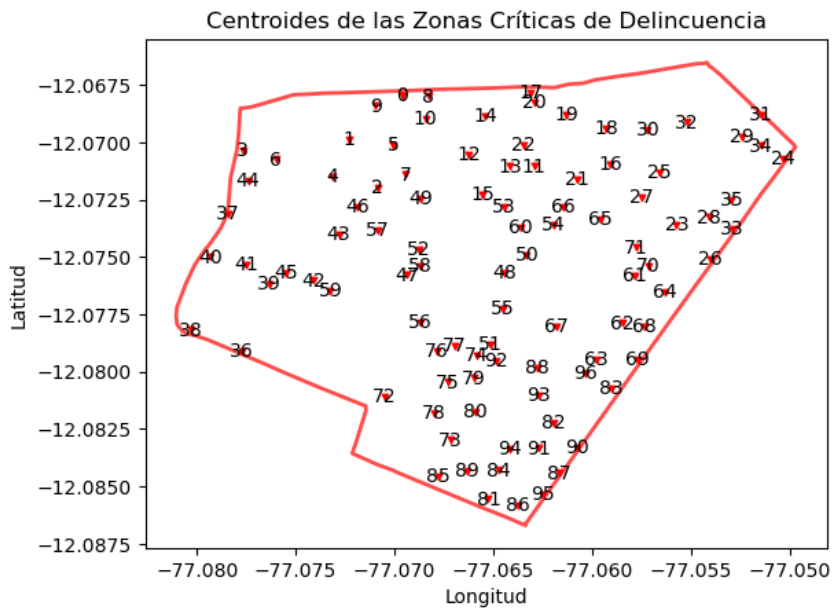
Mapa de Puntos Delictivos de cada Zona Crítica de Delincuencia



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

Figura 29

Mapa de Centroides de Zonas Críticas de Delincuencia



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia.

Tabla 20*Determinación Geográfica de las Zonas Críticas de Delincuencia*

Grupo ID	Longitud del Centroide	Latitud del Centroide	Dirección /Referencia
0	-77.069603	-12.067945	Avenida Mariano Cornejo cuadra 17 cruce con Jirón Cartagena
1	-77.072283	-12.069872	Calle Río Tambo cuadra 4 cruce con Jirón Río Huanta
2	-77.070878	-12.071963	Avenida Simón Bolívar cuadra 15 cruce con Calle Navarra
3	-77.077670	-12.070373	Avenida Simón Bolívar cuadra 22 cruce con Avenida Universitaria
4	-77.073167	-12.071498	Avenida Simón Bolívar cuadra 18 cruce con Calle Río Moche
5	-77.070064	-12.070110	Calle Río Tambo cuadra 2 cruce con Calle La Coruña
6	-77.075991	-12.070752	Avenida Simón Bolívar cuadra 21 cruce con Calle Abraham Lincoln
7	-77.069446	-12.071414	Jirón Santander cuadra 2
8	-77.068294	-12.067999	Avenida Mariano Cornejo cuadra 15 cruce con Calle Valencia cuadra 1
9	-77.070976	-12.068419	Calle Valencia cuadra 4 cruce con Calle Galicia
10	-77.068448	-12.068984	Calle Toledo cuadra 1 cruce con Jirón General José Ramón Pizarro
11	-77.062950	-12.071056	Avenida José Mariano Arce cuadra 2
12	-77.066258	-12.070525	Calle Abraham Valdelomar cuadra 5 cruce con Jirón José Mariano Arce
13	-77.064163	-12.071056	Jirón José Mariano Arce cuadra 3 cruce con Calle Andalucía
14	-77.065401	-12.068850	Jirón Granada cuadra 4
15	-77.065544	-12.072264	Avenida Simón Bolívar cuadra 11
16	-77.059104	-12.070956	Avenida Paso de los Andes Cuadra 9 con Calle Copacabana
17	-77.063119	-12.067827	Avenida Mariano Cornejo cuadra 12 cruce con Calle Talavera
18	-77.059320	-12.069418	Avenida Paso de los Andes cuadra 10
19	-77.061321	-12.068784	Avenida José de San Martín cuadra 1

Grupo ID	Longitud del Centroide	Latitud del Centroide	Dirección /Referencia
20	-77.062950	-12.068251	Calle Talavera cuadra 1 cruce con Jirón Zaragoza cuadra 1
21	-77.060755	-12.071634	Avenida Simón Bolívar cuadra 8 cruce con Jirón Morelia y Riglos
22	-77.063480	-12.070116	Calle Parque José de San Martín cuadra 2
23	-77.055738	-12.073590	Avenida del Río cuadra 5 cruce Avenida Simón Bolívar
24	-77.050352	-12.070729	Avenida Brasil cuadra 11
25	-77.056592	-12.071340	Calle Maipú cuadra 6 cruce Calle Bahía
26	-77.054034	-12.075077	Avenida Simón Bolívar cuadra 1 cruce con Avenida Brasil cuadra 16
27	-77.057521	-12.072400	Avenida Simón Bolívar cuadra 4 cruce con Pasaje Salta
28	-77.054083	-12.073266	Calle Coraceros cuadra 2 con Calle Juan Pablo Fernandini
29	-77.052442	-12.069743	Avenida Colombia cuadra 2
30	-77.057209	-12.069456	Jirón Juan José Pasos cuadra 3
31	-77.051460	-12.068820	Jirón Pedro Ruiz Gallo cuadra 2
32	-77.055225	-12.069136	Jirón Juan José Pasos cuadra 1 cruce con Calle El Gondal
33	-77.052911	-12.073766	Avenida Brasil cuadra 14 cruce con Calle Coraceros
34	-77.051468	-12.070148	Calle Juan Pablo Fernandini cuadra 11 cruce con Avenida Colombia
35	-77.052964	-12.072500	Jirón Pedro Morales Bermúdez cuadra 1 cruce con Calle Juan Pablo Fernandini
36	-77.077734	-12.079107	Avenida La Marina cuadra 15 cruce con Calle Kenko
37	-77.078400	-12.073099	Avenida Universitaria cuadra 19
38	-77.080286	-12.078180	Avenida La Marina cuadra 16 cruce con Avenida Universitaria
39	-77.076359	-12.076165	Avenida Jirón Abraham Lincoln cuadra 4 cruce con Calle Martín de Oviedo
40	-77.079299	-12.075003	Avenida Manuel Cipriano cuadra 20 cruce con Avenida Universitaria
41	-77.077534	-12.075324	Avenida Manuel Cipriano cuadra 19
42	-77.074118	-12.076022	Avenida Coronel Juan Valer Sandoval cuadra 5 cruce con Avenida Manuel Cipriano Dulanto

Grupo ID	Longitud del Centroide	Latitud del Centroide	Dirección /Referencia
43	-77.072815	-12.074018	Avenida Coronel Juan Valer Sandoval cuadra 4 cruce con Avenida General José Leguía y Meléndez
44	-77.077408	-12.071669	Calle Paracas cuadra 5
45	-77.075472	-12.075714	Avenida Manuel Cipriano Dulanto cuadra 18 cruce con Pasaje Cipriano Dulanto 2
46	-77.071908	-12.072802	Calle Salinar cuadra 1
47	-77.069400	-12.075784	Calle Santa Fe cuadra 3
48	-77.064439	-12.075678	Avenida José Leguía y Meléndez cuadra 9
49	-77.068680	-12.072443	Avenida Simón Bolívar cuadra 13 cruce con Jirón General José Ramón Pizarro
50	-77.063320	-12.074928	Avenida Antonio José de Sucre cuadra 4 cruce con Calle Alejandro Pedro Murillo
51	-77.065148	-12.078807	Avenida Manuel Cipriano cuadra 10 cruce con Calle Bartolomé Figari
52	-77.068754	-12.074673	Jirón Daniel Hernández cuadra 7
53	-77.064484	-12.072842	Jirón Yauli cuadra 1
54	-77.061970	-12.073599	Avenida San Martín cuadra 7
55	-77.064548	-12.077254	Avenida Antonio José de Sucre cuadra 6 cruce con Avenida General Manuel Vivanco
56	-77.068677	-12.077870	Avenida Manuel Cipriano cuadra 13
57	-77.070866	-12.073844	Calle Huari cuadra 1
58	-77.068686	-12.075414	Avenida José Leguía y Meléndez cuadra 14 cruce con Jirón Daniel Hernández
59	-77.073250	-12.076488	Avenida Manuel Cipriano Dulanto cuadra 16
60	-77.063648	-12.073738	Calle General Artigas cuadra 6
61	-77.057873	-12.075824	Calle 29 de Diciembre cuadra 3
62	-77.058512	-12.077886	Calle Loreto cuadra 2 cruce con Jirón José Santiago Wagner
63	-77.059816	-12.079447	Avenida General Manuel Vivanco cuadra 4
64	-77.056358	-12.076518	Calle José Manuel Ugarteche cuadra 1
65	-77.059622	-12.073336	Calle Rodríguez de Mendoza cuadra 1
66	-77.061468	-12.072820	Calle Rodríguez de Mendoza cuadra 3 cruce con Avenida San Martín
67	-77.061821	-12.078038	Jirón Carlos de los Heros cuadra 2
68	-77.057381	-12.078030	Jirón Domingo Nieto cuadra 1
69	-77.057739	-12.079460	Avenida Brasil cuadra 21 cruce con Calle Antonio Polo

Grupo ID	Longitud del Centroide	Latitud del Centroide	Dirección /Referencia
70	-77.057172	-12.075417	Jirón José Santiago Wagner cuadra 17
71	-77.057812	-12.074583	Calle Isla San Lorenzo cuadra 1
72	-77.070488	-12.081086	Pasaje La Marina cuadra 1
73	-77.067195	-12.082969	Avenida La Marina cuadra 6 cruce con Avenida José de Sucre cuadra 11
74	-77.065850	-12.079293	Calle Acevedo cuadra 8 cruce con Avenida José de Sucre
75	-77.067291	-12.080452	Calle Coral cuadra 1 cruce con Calle Príncipe
76	-77.067848	-12.079119	Calle Carlos Vidal cuadra 2 cruce con Calle Acevedo
77	-77.066980	-12.078875	Jirón Eleazar Blanco cuadra 1
78	-77.068024	-12.081823	Calle Santa Isabel cuadra 1
79	-77.065993	-12.080283	Calle Nicolás Alcázar cuadra 6 con Avenida Antonio José de Sucre
80	-77.065898	-12.081740	Calle David Matto cuadra 1
81	-77.065262	-12.085552	Jirón José Santiago Wagner cuadra 28
82	-77.061981	-12.082220	Calle Nicolás Alcázar cuadra 3 cruce con Calle José Morales Alpaca
83	-77.059044	-12.080711	Calle Toribio Ara cuadra 2
84	-77.064743	-12.084292	Avenida La Marina cuadra 3 cruce con Jirón José Santiago Wagner
85	-77.067810	-12.084575	Avenida Antonio José de Sucre cuadra 13
86	-77.063759	-12.085817	Avenida La Marina cuadra 1 cruce con Calle Gregorio Paredes
87	-77.061661	-12.084414	Avenida Brasil cuadra 27 cruce con Jirón Cabo Nicolás Gutarra
88	-77.062777	-12.079837	Jirón Carlos de los Heros cuadra 5
89	-77.066352	-12.084316	Calle Paseo de la Libertad cuadra 3 cruce con Calle Carlos Bondy
90	-77.060780	-12.083288	Calle Nicolás Alcázar cuadra 1 cruce con Avenida Brasil
91	-77.062698	-12.083336	Calle José Morales Alpaca cuadra 1
92	-77.064880	-12.079540	Calle Acevedo cuadra 2
93	-77.062688	-12.081006	Calle Agustín Gamarra cuadra 5 cruce con Parque Amoretti
94	-77.064171	-12.083346	Calle Marcos Dongo cuadra 2 cruce con Jirón San José
95	-77.062543	-12.085343	Jirón San José cuadra 1 cruce con Avenida Brasil cuadra 28

Grupo ID	Longitud del Centroide	Latitud del Centroide	Dirección /Referencia
96	-77.060364	-12.080088	Avenida Manuel Cipriano Dulanto cuadra 4 cruce con Jirón José Santiago Wagner

Nota. Información Geográfica determinada con los softwares Jupyter Notebook y ArcGis Pro. Elaboración Propia.

3.4.3.6. Algoritmo Genético

A continuación, se describe las fases de aplicación del Algoritmo Genético Adaptado al Problema del Agente Viajero:

Fase 1: Definir la Representación de las Soluciones (rutas) del Algoritmo Genético. En esta fase, se define cada una de las soluciones que genera el Algoritmo Genético adaptado al Problema del Agente Viajero. En el caso de estudio, cada una de las soluciones halladas por el Algoritmo Genético, representará la ruta de patrullaje en un circuito cerrado, que pase por todos los centroides de las Zonas Críticas de Delincuencia. Los centroides están almacenados en el Dataframe “centroides”, en los cuales el índice de cada registro del Dataframe será el identificador de cada centroide. En el caso de estudio, el número de centroides es de 97, con valores del índice (identificador) de 0 a 96.

La representación según la teoría de Grafos es la siguiente:

Si $G = (V, E)$ es un grafo completo, donde:

- V es un conjunto de nodos (vértices), donde cada vértice representa un índice del Dataframe “centroides”.
- E es el conjunto de aristas, donde cada arista (i, j) representa la distancia entre los centroides i y j .

Dado un conjunto de vértices $V = \{v_0, v_1, v_2, \dots, v_{96}\}$, donde $n=97$ es el número de centroides del Dataframe “centroides”, una solución óptima es representada como una permutación de los vértices, que define el orden en el que

se visitan los centroides minimizando la distancia total entre los centroides. Por lo tanto, una solución (ruta) se representa en la Ecuación 23.

$$\pi = \{v_{\pi(0)}, v_{\pi(1)}, v_{\pi(2)}, v_{\pi(3)}, \dots, v_{\pi(96)}\} \quad (23)$$

Donde:

- $\pi(i)$ es el vértice en la posición i de la permutación.

Fase 2: Definir la Función Fitness (Función de Aptitud). En esta fase, se definirá las Función Fitness (Función de Aptitud), la cual se encarga de evaluar qué tan buena es una ruta de patrullaje en términos de su longitud total. En el caso de estudio, se define como la longitud total de patrullaje, ya que la función fitness minimizará la longitud total de patrullaje. La representación matemática de la función fitness se muestra en la Ecuación 24.

$$f(\pi) = L(\pi) \quad (24)$$

Donde:

- π es la ruta (permutación de vértices) que se evaluará.
- $L(\pi)$ es la longitud total de la ruta π , que se define como la suma de distancias entre cada par de la ruta.
- $f(\pi)$ es la función Fitness para una ruta π en el problema del viajante.

La Longitud total de la ruta π se calcula mediante la Ecuación 25:

$$L(\pi) = \sum_{i=0}^{n-2} d(\pi_i, \pi_{i+1}) + d(\pi_{n-1}, \pi_0) \quad (25)$$

Donde:

- $d(u, v)$ es la distancia entre los nodos u y v (2 zonas críticas).
- n es el número de nodos en el grafo.

Fase 3: Implementar el Algoritmo Genético. En esta fase se codificaron las funciones necesarias para la implementación del Algoritmo Genético, así como,

los operadores genéticos de selección, cruce y mutación para la generación de los individuos (rutas) y poblaciones (conjunto de rutas) necesarios para la ejecución del algoritmo genético. El código utilizado para la implementación del Algoritmo Genético se ubica en el Anexo 7.

Fase 4: Descripción de los Parámetros del Algoritmo Genético. A continuación, se describen los 4 parámetros del Algoritmo Genético.

- Tamaño de la Población ('tamanoPoblacion'): Se refiere al número de rutas (individuos) de patrullaje evaluadas en cada generación del Algoritmo Genético.
- Tamaño de Elite ('tamanoElite'): Indica la cantidad de individuos con las mejores función Fitness de cada generación se seleccionarán automáticamente para pasar a la siguiente generación sin cambios.
- Ratio de Mutación('ratioMutacion'): Representa la tasa de mutación, que determina la probabilidad de que ocurra una mutación en cada individuo durante la generación de la siguiente población.
- Cantidad de Generaciones('generaciones'): Define el número total de generaciones (iteraciones) que el algoritmo genético llevará a cabo durante su ejecución.

Fase 5: Ejecución del Algoritmo Genético. En esta fase ocurre la ejecución del Algoritmo Genético, cuyos valores de los parámetros son los siguientes:

- 'tamanoPoblacion' = 2000
- 'tamanoElite' = 1000
- 'ratioMutacion' = 0.00005
- 'generaciones' = 1000

Después de ejecutar 10 veces el Algoritmo Genético, se obtuvo la Tabla 21, en la que se muestra las distancias mínimas de cada una de las ejecuciones, junto con el tiempo de procesamiento. Posteriormente, se procedió a graficar la evolución

de la Distancia Total de Patrullaje en cada generación para cada una de las 10 ejecuciones del Algoritmo Genético.

El código relacionado al Algoritmo Genético se ubica en el Anexo 7.

Tabla 21

Distancias Mínimas en las 10 Ejecuciones del Algoritmo Genético

Ejecución	Distancia Mínima de Patrullaje en la Última Generación (metros)	Tiempo de Ejecución (segundos)
1	20113	219
2	20397	230
3	21345	228
4	20196	243
5	21344	236
6	20264	242
7	19917	235
8	20059	223
9	19535	225
10	20245	224

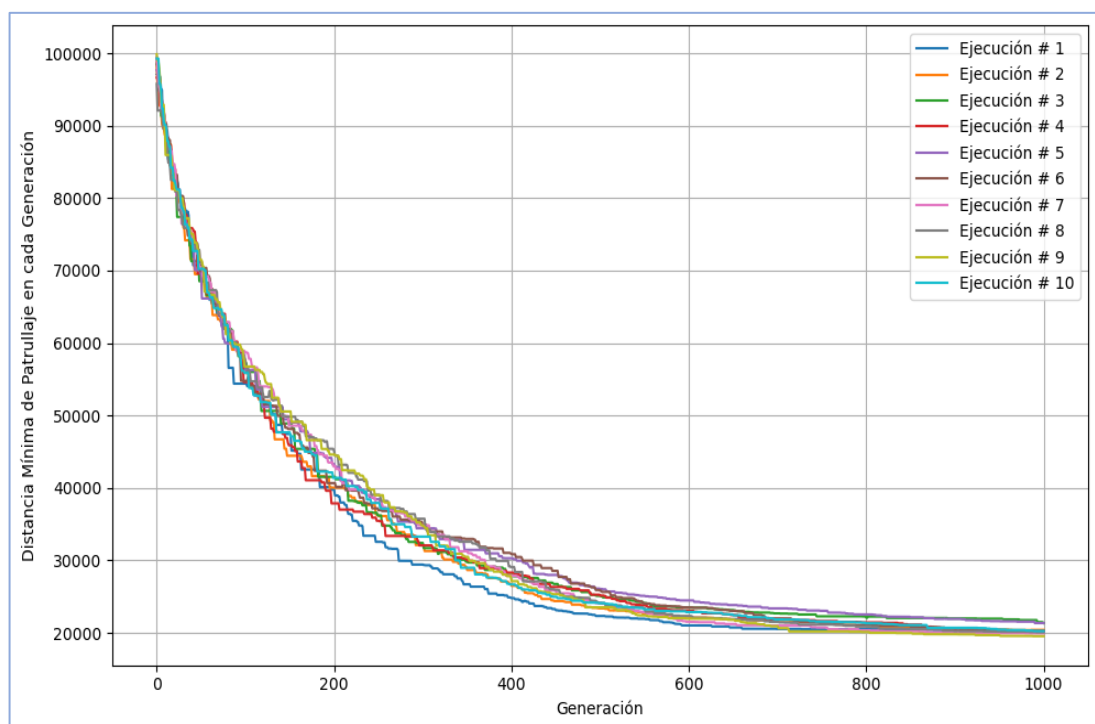
Nota. Resultados obtenidos a partir del software Jupyter Notebook. Elaboración Propia.

Como resultado se obtuvo en la Ejecución 9 del Algoritmo Genético, la menor distancia total de patrullaje de Zonas Críticas de Delincuencia encontrada tuvo un valor de 19.535 km y con la ruta [53, 60, 50, 48, 11, 22, 20, 17, 14, 12, 10, 8, 0, 9, 1, 5, 2, 7, 49, 52, 58, 59, 42, 45, 39, 41, 36, 38, 40, 37, 44, 3, 6, 4, 43, 46, 57, 47, 56, 75, 80, 79, 74, 92, 51, 55, 77, 76, 72, 78, 73, 85, 89, 81, 86, 95, 84, 94, 91, 90, 87, 82, 93, 88, 67, 63, 96, 83, 69, 62, 68, 64, 70, 61, 71, 23, 26, 28, 33, 35, 24, 34, 31, 29, 32, 30, 25, 27, 65, 16, 18, 19, 21, 66, 54, 13, 15]. Cabe destacar que las zonas críticas inicial y final se unen para formar un circuito.

En la Figura 30, se muestra la evolución de la distancia de la ruta de Patrullaje, en la cual se muestra la reducción de la distancia total de patrullaje para cada generación del algoritmo genético.

Figura 30

Evolución de la Distancia de Patrullaje en cada Generación del Algoritmo Genético



Nota. Gráfica elaborada con el software Jupyter Notebook. Las distancias mínimas de Patrullaje están expresadas en metros. Elaboración Propia.

3.4.3.7. Algoritmo Memético

Fase 1: Definir la Representación de las Soluciones (rutas) del Algoritmo

Memético. En esta fase, se define cada una de las soluciones que genera el Algoritmo Memético adaptado al Problema del Agente Viajero. La representación de las soluciones (rutas de patrullaje) es la misma que la utilizada en el Algoritmo Genético

Fase 2: Definir la Función Fitness (Función de Aptitud). En esta fase, se definirá las Función Fitness (Función de Aptitud), la cual se encarga de evaluar qué tan buena es una ruta de patrullaje en términos de su longitud total. En el caso de estudio, la función fitness es la longitud total de patrullaje. La formulación matemática es la misma que la definida en el Algoritmo Genético.

Fase 3: Implementar el Algoritmo Memético. En esta fase se codificaron las funciones necesarias para la implementación del Algoritmo Memético, así como, los operadores genéticos de selección, cruce y mejora para la generación de los individuos (rutas) y poblaciones (conjunto de rutas) necesarios para la ejecución del algoritmo genético. El código utilizado para la implementación del Algoritmo Genético se ubica en el Anexo 8.

Fase 4: Descripción de los Parámetros del Algoritmo Memético. A continuación, se describen los 4 parámetros del Algoritmo Memético.

- Tamaño de la Población ('tamanoPoblacion'): Se refiere al número de rutas (individuos) de patrullaje evaluadas en cada generación del Algoritmo Memético.
- Tamaño de Elite ('tamanoElite'): Indica la cantidad de individuos con las mejores función Fitness de cada generación se seleccionarán automáticamente para pasar a la siguiente generación sin cambios.

- Ratio de Mejora('ratioMejora'): Representa el factor de mejora local (2-Opt) aplicado a un porcentaje determinado de las soluciones, en el caso particular se aplica el algoritmo de búsqueda local 2-Opt. El valor de 1 indica que se aplicará a todos los individuos el algoritmo 2-Opt.
- Cantidad de Generaciones('generaciones'): Define el número total de generaciones (iteraciones) que el algoritmo memético llevará a cabo durante su ejecución.

Fase 5: Ejecución del Algoritmo Memético. En esta fase ocurre la ejecución del Algoritmo Memético, cuyos valores de los parámetros son los siguientes:

- 'tamanoPoblacion' = 200
- 'tamanoElite' = 100
- 'ratioMejora' = 1
- 'generaciones' = 200

El número de generaciones de 200 fue determinado en base a la rápida convergencia entorno a ese número. Después de ejecutar 10 veces el Algoritmo Memético, se obtuvo la Tabla 22, en la que se muestra las distancias mínimas de patrullaje en cada una de las ejecuciones, junto con el tiempo de procesamiento. Posteriormente, se procedió a graficar la evolución de la Función Fitness en cada generación para cada una de las 10 ejecuciones del Algoritmo Memético mostrándose en la Figura 31. El código relacionado al Algoritmo Memético se ubica en el Anexo 8.

Tabla 22

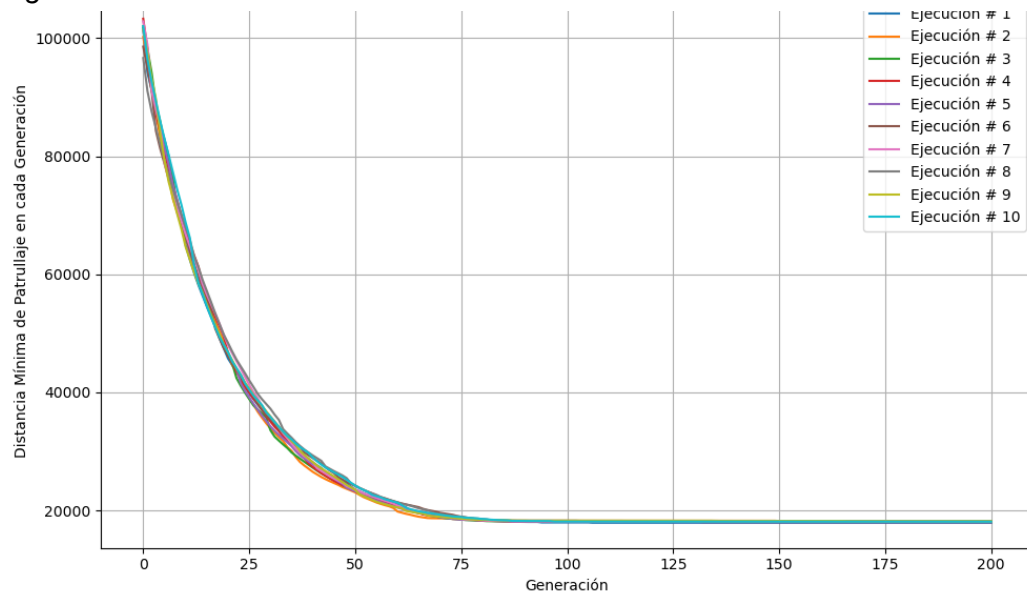
Distancias Mínimas en las 10 Ejecuciones del Algoritmo Memético

Ejecución	Distancia Mínima de Patrullaje en la Última Generación (metros)	Tiempo de Ejecución (segundos)
1	17907	169
2	18195	178
3	18050	174
4	18025	170
5	18025	172
6	18075	172
7	18140	172
8	18145	164
9	18180	168
10	18092	167

Nota. Resultados obtenidos a partir del software Jupyter Notebook. Elaboración Propia.

Figura 31

Evolución de la Distancia de Patrullaje (Función Fitness) en cada Generación del Algoritmo Memético



Nota. Gráfica elaborada con el software Jupyter Notebook. Elaboración Propia.

En la Ejecución #1 del Algoritmo Memético, se obtuvo la menor distancia total de patrullaje de Zonas Críticas de Delincuencia con un valor de 17.907 km y con la

ruta [34, 31, 29, 32, 30, 25, 27, 65, 54, 66, 21, 16, 18, 19, 17, 20, 22, 11, 13, 14, 12, 15, 53, 60, 50, 48, 55, 51, 92, 74, 77, 76, 56, 47, 58, 52, 57, 43, 46, 2, 49, 7, 5, 10, 8, 0, 9, 1, 4, 6, 3, 44, 37, 40, 38, 36, 41, 39, 45, 42, 59, 72, 78, 75, 79, 80, 73, 85, 89, 94, 84, 81, 86, 95, 87, 90, 91, 82, 93, 88, 67, 63, 96, 83, 69, 62, 68, 64, 70, 61, 71, 23, 28, 26, 33, 35, 24]. Cabe destacar que las zonas críticas inicial y final se unen para formar un circuito.

3.5. EVALUACIÓN

3.5.1. Evaluación de los Resultados del Algoritmo Genético y Memético

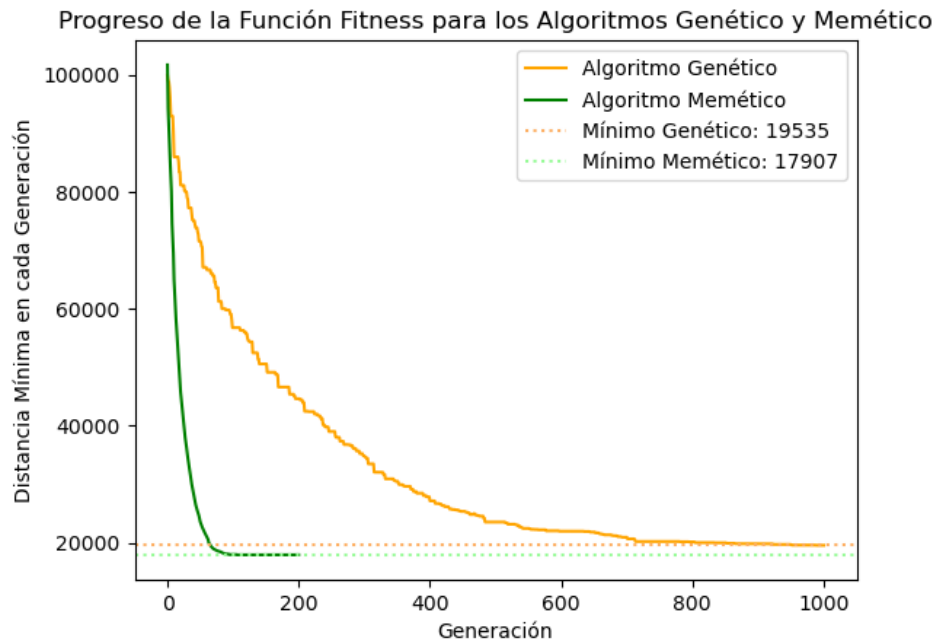
En este apartado se mostrarán la evaluación comparativa de los resultados obtenidos a partir de la aplicación de los Algoritmos Genéticos y Algoritmo Memético. Para dicha evaluación, se realizó comparó la convergencia de las Funciones Fitness de la solución con la menor distancia total de patrullaje encontrada por el Algoritmo Genético y el Algoritmo Memético (ver Tabla 21 y Tabla 22).

Evaluación de la Convergencia de la Función Fitness (Distancia)

Se procedió a graficar la Función Fitness de cada solución encontrada en la Figura 32, en la que se observa que la línea verde, que representa la Función Fitness del Algoritmo Memético, tiene una convergencia mayor que la Función Fitness del Algoritmo Genético, además de converger a un valor menor que el Algoritmo Genético.

Figura 32

Función Fitness de las Soluciones (Algoritmos Genético y Memético)



Nota. Gráfica elaborada con el software Jupyter Notebook. Distancia Mínima en cada Generación expresada en metros. Elaboración Propia.

Luego, se realizó el cálculo de la Convergencia Promedio de generaciones consecutivas de las Funciones Fitness mediante el uso de la Ecuación 26, obteniéndose los valores de **80.7** y **418.9** como Convergencia Promedio de las Funciones Fitness de los Algoritmos Genético y Memético respectivamente, lo cual indica que la Función Fitness obtenida por el Algoritmo Memético converge más rápido que el Algoritmo Genético.

$$\text{Convergencia Promedio} = \frac{1}{n} \sum_{i=1}^{n-1} |f(x_{i+1}) - f(x_i)| \quad (26)$$

Donde:

- ✓ n : Número de Generaciones
- ✓ $f(x_i)$: Valor de la Función Fitness en la generación i .

- ✓ $|f(x_{i+1}) - f(x_i)|$: Valor absoluto de la diferencia entre 2 valores de Función Fitness de generaciones consecutivas.

A continuación, se realizó un Análisis Estadístico de mínimo, máximo, promedio y desviación estándar de las últimas 100 generaciones de los Algoritmo Genético y Algoritmo Memético y se obtuvo la Tabla 23, en la cual se observa mejores resultados obtenidos por el Algoritmo Memético que el Algoritmo Genético.

Tabla 23

Estadísticas de los Valores de la Función Fitness de las últimas 100 generaciones del Algoritmo Genético y Algoritmo Memético

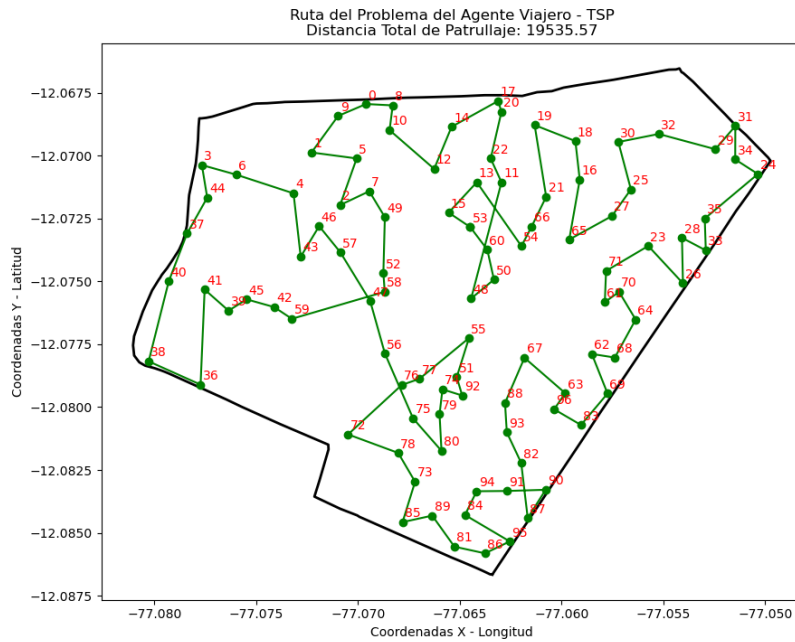
Medida Estadística	Algoritmo Genético	Algoritmo Memético
Mínimo	19535	17907
Máximo	19797	17951
Promedio	19643	17909
Desviación Estándar	93	9

Nota. Resultados obtenidos a partir del software Jupyter Notebook. Valores en metros. Elaboración Propia.

Posteriormente, se procedió a mapear el recorrido de patrullaje de Zonas Críticas de Delincuencia, para lo cual se realizó un Mapa de Puntos de las soluciones encontradas por ambos algoritmos. Se muestra el Mapa de Puntos de la Ruta Mínima de Patrullaje encontrada por el Algoritmo Genético en la Figura 33 y el Mapa de Puntos de la Ruta Mínima de Patrullaje encontrada por el Algoritmo Memético en la Figura 34. El código relacionado a la evaluación de los Algoritmos Genético y Memético se encuentra en el Anexo 9.

Figura 33

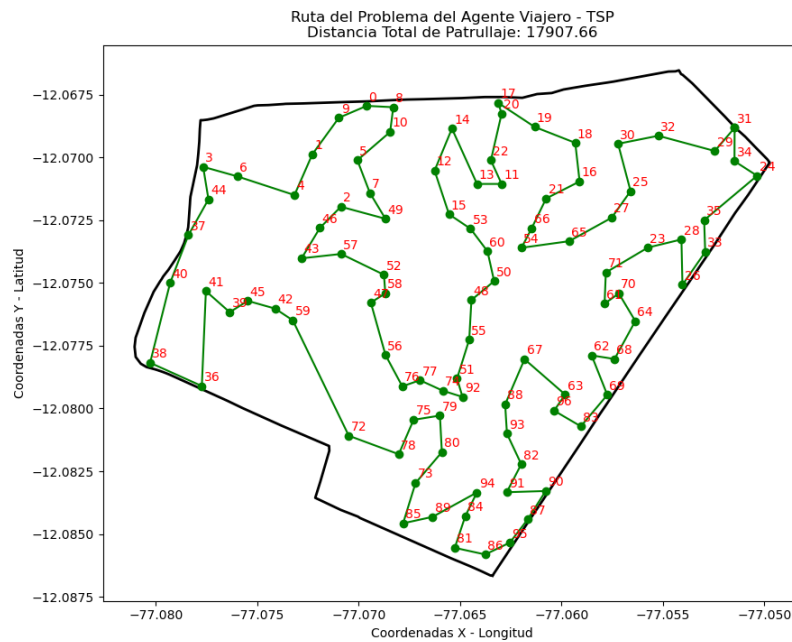
Mapa de Puntos de la Ruta Mínima de Patrullaje encontrada por el Algoritmo



Nota. Gráfica elaborada con el software Jupyter Notebook. Elaboración Propia.

Figura 34

Mapa de Puntos de la Ruta Mínima de Patrullaje encontrada por el Algoritmo



Nota. Gráfica elaborada con el software Jupyter Notebook. Elaboración Propia.

Se puede observar en la Figura 33, se generaron cruces en la ruta de patrullaje encontrado por el Algoritmo Genético; sin embargo, en la Figura 34 no se observó ningún cruce en la ruta de patrullaje encontrado por el Algoritmo Memético.

Evaluación de los Criterios de Éxito de la Minería de Datos

En el Acápite 3.1.3., se determinó los Criterios de éxito de minería de datos, entre los cuales se especifican: reducir en un 10% la distancia de recorrido de patrullaje entre las zonas críticas de delincuencia de Pueblo Libre, así como la consideración de la densidad delictiva relativa de 2 veces la Densidad Delictiva del Distrito. En este sentido, el resultado obtenido por el Algoritmo Memético cumple con una reducción del 13.8% de la distancia de patrullaje (del 20.784 km a 17.907km) y la una densidad delictiva relativa de 15.13 veces (ver Tabla 19) la densidad delictiva promedio del distrito. Sin embargo, el resultado obtenido por el Algoritmo Genético, obtuvo una reducción del 6.0% (del 20.784 km al 19.535 km) no cumpliéndose el criterio de la reducción del 10% de la distancia de recorrido de patrullaje. A partir de las comparaciones de la convergencia, tiempos de ejecución y menores valores distancias mínimas de patrullajes encontradas, las medidas estadísticas de las últimas 100 generaciones de la función Fitness y los criterios de éxito de la minería de datos se afirma que la ruta obtenida por el Algoritmo Memético es cuantitativa y cualitativamente más adecuada que la ruta obtenida por el Algoritmo Genético.

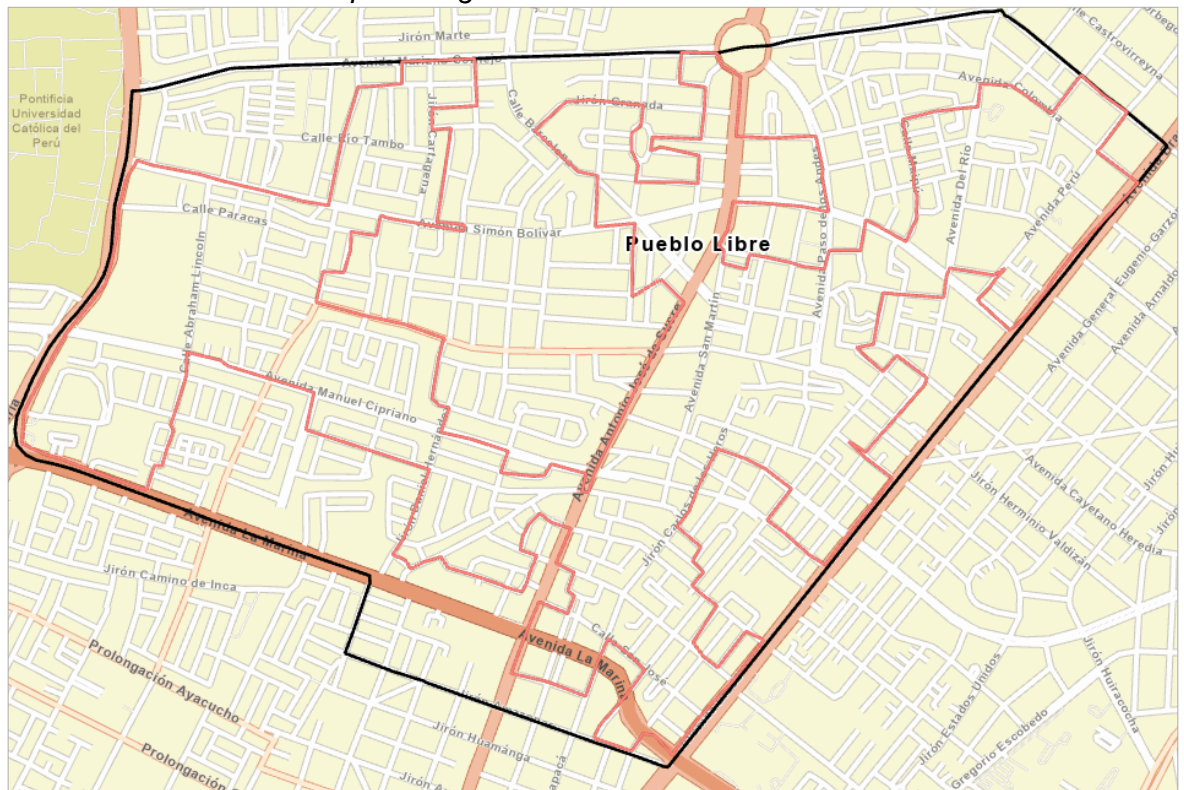
3.5.2. Determinación de la Distancia Total y Recorrido de Patrullaje de Zonas Críticas de Delincuencia Óptimo.

Respecto a la Distancia Total de Patrullaje, se determinó como distancia mínima de patrullaje la encontrada por el Algoritmo Memético de 17.907 km. Posteriormente, con la ayuda de la Tabla 20 (Dirección exacta de los centroides de las Zonas Críticas de Delincuencia), se procedió a graficar en el Software ArcGis Pro

el recorrido de Patrullaje de las Zonas Críticas de Delincuencia. El recorrido del patrullaje resultante se muestra en la Figura 35. Así mismo, se elaboró la Tabla 24, en la cual se muestra cada una de las vías (Avenidas, Calles, Pasajes, etc.) que recorre la ruta de patrullaje propuesta.

Figura 35

Mapa del Camino (Trayectoria) de Patrullaje de Zonas Críticas de Delincuencia de la Ruta Mínima determinada por el Algoritmo Memético



Nota. Elaborado con el software ArcGis Pro. La línea anaranjada representa la ruta de patrullaje determinada por el Algoritmo Memético.

Tabla 24

Camino de Patrullaje (Vías) de Patrullaje de Zonas Críticas de Delincuencia de la Ruta Mínima determinada por el Algoritmo Memético

Nombre de la Vía	Distancia (m)
Avenida La Marina	289
Avenida Universitaria	754
Avenida Simón Bolívar	641
Jirón Huaura	323
Calle Valencia	114
Calle Galicia	53
Avenida Mariano Cornejo	174
Jirón F. Valle Riestra	126
Calle Toledo	114
Jirón Cartagena	44
Calle América	50
Jirón Santander	211
Avenida Simón Bolívar	136
Calle Navarra	52
Calle Paracas	70
Calle Salinar	109
Calle Nepeña	52
Avenida Coronel Juan Valer Sandoval	84
Avenida General José Leguía y Meléndez	321
Jirón General José Ramón Pizarro	195
Jirón Daniel Hernández	70
Avenida Manuel Cipriano	194
Calle Carlos Vidal	47
Jirón Eleazar Blanco	168
Avenida Manuel Cipriano	105
Avenida Antonio José de Sucre	466
Calle Olmedo	95
Calle General Artigas	88

Nombre de la Vía	Distancia (m)
Calle Andalucía	155
Avenida Simón Bolívar	124
Calle Sevilla	148
Calle Barcelona	201
Jirón Granada	253
Calle Andalucía	195
Avenida José Mariano Arce	155
Calle General Artigas	73
Parque San Martín	253
Calle Triana	134
Jirón Zaragoza	81
Calle Talavera	85
Avenida Mariano Cornejo	102
Plaza La Bandera	95
Avenida José de San Martín	136
Jirón Juan José Paso	205
Avenida Paso de los Andes	182
Avenida Simón Bolívar	215
Avenida José de Sucre	108
Avenida San Martín	61
Pasaje D	70
Pasaje F	43
Calle Rodríguez de Mendoza	203
Avenida Paso de los Andes	48
Pasaje Paso de los Andes	74
Calle Isla Hawai	77
Avenida Simón Bolívar	113
Calle Maipú	318
Jirón Juan José Paso	200
Avenida del Río	67
Avenida Colombia	202
Jirón Huaraz	135

Nombre de la Vía	Distancia (m)
Jirón Pedro Ruiz Gallo	142
Avenida Perú	114
Avenida Colombia	118
Avenida Brasil	461
Avenida Simón Bolívar	118
Avenida Perú	103
Calle Coraceros	145
Avenida del Río	72
Calle San Lorenzo	152
Calle Luna Iglesias	145
Prolongación Cayetano Heredia	50
Calle 29 de Diciembre	95
Jirón José Santiago Wagner	64
Prolongación Cayetano Heredia	87
Calle José Manuel Ugarteche	273
Calle Loreto	201
Avenida Brasil	296
Avenida Manuel Cipriano	202
Jirón José Santiago Wagner	103
Avenida General Manuel Vivanco	210
Jirón Carlos de los Heros	232
Calle Gallese	88
Calle Agustín Gamarra	52
Calle Zoila Amoretti	42
Calle Agustín Gamarra	51
Calle Nicolás Alcázar	96
Calle José Morales Alpaca	123
Jirón Cabo Nicolás Gutarra	92
Calle Odriozola	142
Calle Nicolás Alcázar	62
Avenida Brasil	344
Avenida La Marina	112

Nombre de la Vía	Distancia (m)
Calle Gregorio Paredes	51
Jirón Amazonas	97
Jirón José Santiago Wagner	214
Calle San José	147
Calle Manuel de Guirior	170
Jirón Tarapacá	222
Calle Carlos Bondy	188
Avenida Antonio José de Sucre	90
Calle San José	54
Jirón Libertad	55
Calle Lino Alarco	24
Calle Cabo Gutarra	56
Jirón Cabo Nicolás Gutarra	59
Avenida Antonio José de Sucre	103
Calle Nicolás Alcázar	84
Calle Príncipe	107
Jirón Cabo Nicolás Gutarra	50
Calle Santa Ana	104
Calle Santa Isabel	223
Jirón Pedro Torres Malarin	84
Pasaje La Marina	64
Jirón Daniel Hernández	255
Calle Acevedo	341
Avenida Herrera Ardiles	107
Avenida Manuel Cipriano	320
Calle Abraham Lincoln	202
Calle Kenko	193
Distancia Total de Patrullaje (Después)	17907

Nota. Elaborado a partir de la ruta determinada con el software ArcGis Pro.

Distancias en metros (m).

CAPÍTULO IV

ANÁLISIS Y DISCUSIÓN DE RESULTADOS

4.1. CONTEXTUALIZACIÓN DE RESULTADOS

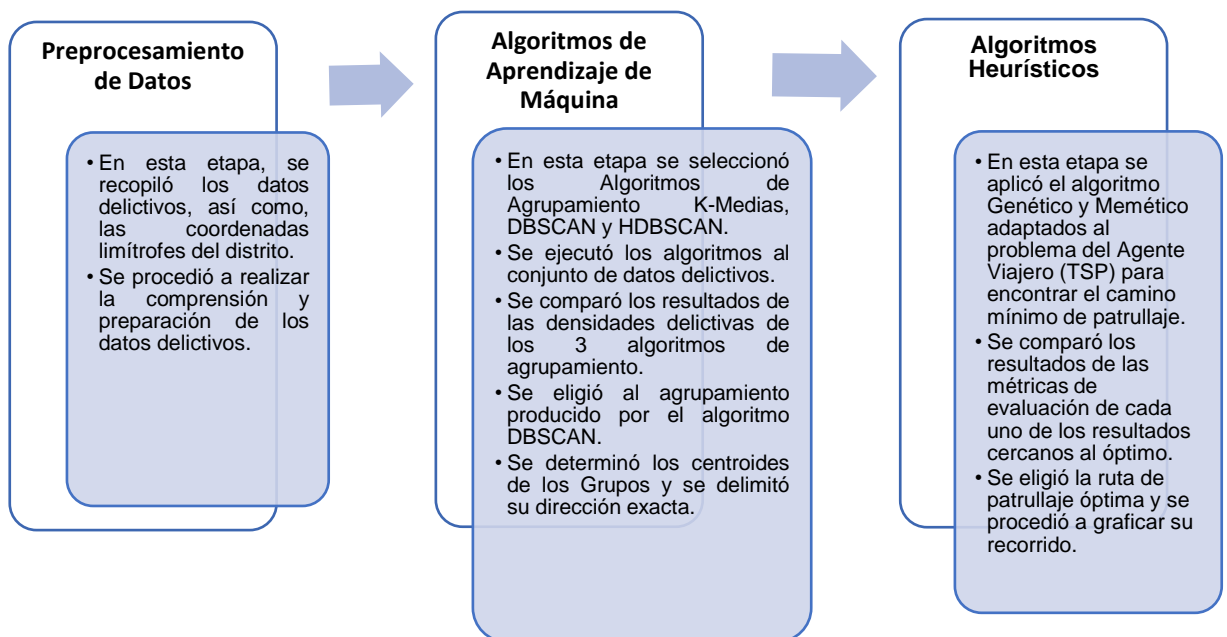
El objetivo principal de la investigación fue reducir la distancia del camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos. Para lograrlo, se necesitó determinar las zonas críticas de delincuencia en Pueblo Libre mediante la aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos, entre los cuales se utilizaron los algoritmos de agrupamiento K-Medias, DBSCAN y HDBSCAN. El conjunto de datos a partir del cual se realizó la aplicación de los algoritmos de agrupamientos fueron los delitos reportados de hurto y robo ocurridos en el año 2022 en Pueblo Libre, específicamente los datos relacionados a medidas geográficas de latitud y longitud. A partir de la aplicación de estos algoritmos de agrupamiento, se obtuvo las agrupaciones de delitos, posteriormente se comparó los indicadores (densidades delictivas) para cada uno de los 3 algoritmos de agrupamiento, y se determinó que el algoritmo DBSCAN agrupaba el conjunto de datos delictivos con mejor calidad de acuerdo a su densidad delictiva. Posteriormente, se procedió a graficar las Zonas Críticas de Delincuencia con un Mapa de Puntos, y se determinó su dirección exacta. A continuación, se procedió a calcular el centroide de cada uno de las Zonas Crítica de Delincuencia, para posteriormente encontrar el camino que recorra todas las

Zonas Crítica de Delincuencia mediante la aplicación de Algoritmos Heurísticos (Algoritmos Genéticos y Algoritmos Meméticos) adaptados al Problema de Agente Viajero (conocido como TSP por sus siglas en inglés). Finalmente, se encontraron varios recorridos cercanos al camino mínimo de patrullaje de las Zonas Críticas de Delincuencia, de los cuales se analizó su factibilidad y conveniencia en el patrullaje del distrito.

En la Figura 36 se muestra el proceso que se siguió para la consecución de los resultados de la presente tesis.

Figura 36

Proceso de la Aplicación de Algoritmos de Aprendizaje de Máquina y Heurísticos para el Patrullaje de Zonas Críticas de Delincuencia



Nota. Elaboración Propia.

4.2. PRESENTACIÓN DE RESULTADOS

Con la aplicación de los Algoritmos de Agrupamiento K-Medias, DBSCAN y HDBSCAN se obtuvo las agrupaciones y los índices de densidad delictiva de cada uno de los agrupamientos. En la Tabla 25, se presentan el mejor agrupamiento resultante de la aplicación de cada algoritmo, así como los indicadores de densidad delictiva para cada agrupamiento.

Tabla 25

Comparación de resultados obtenidos de la Aplicación de los Algoritmos de Agrupamiento e Indicadores de Densidad Delictiva

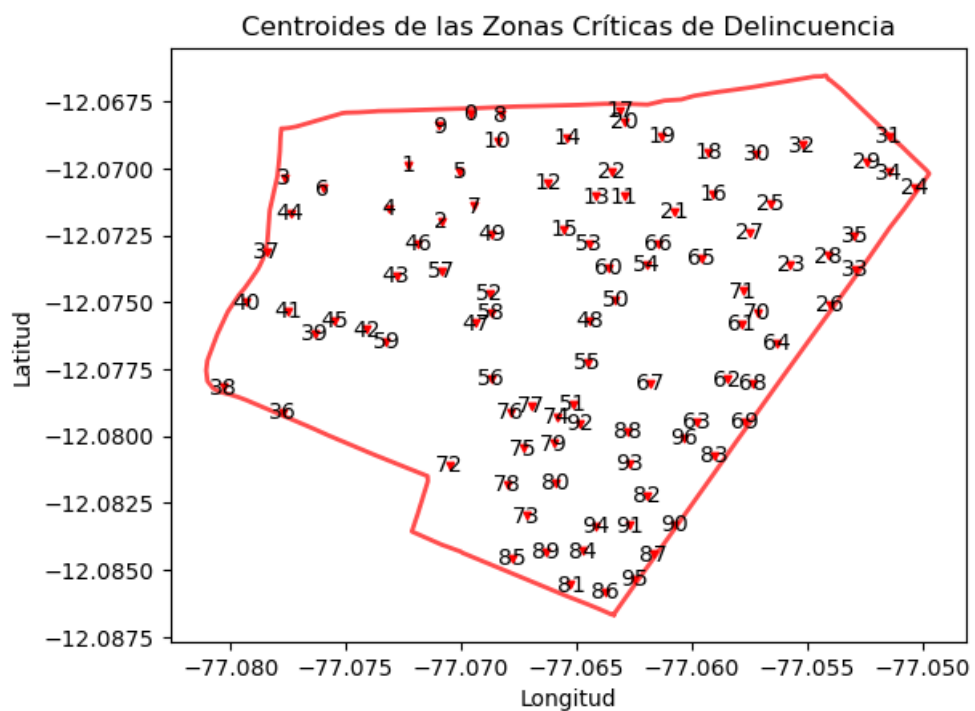
ALGORITMO	K-Medias (K=17)	DBSCAN (min_samples=2, eps = 2.1)	HDBSCAN (m_c_size = 3, min_samples = 2, c_s_epsilon = 2)
Número De Grupos	17	97	73
Número De Registros Ruido	0	189	189
Densidad Delictiva Acumulada	2.43 delitos/hm ²	22.16 delitos/hm ²	16.92 delitos/hm ²
Densidad Delictiva Relativa	1.66 veces	15.13 veces	11.55 veces
Área Acumulada De Los Grupos	293.2 hm ²	22.7 hm ²	29.73 hm ²
Porcentaje Del Área Acumulada De Los Grupos	62.1%	4.8%	6.3%
Proporción Acumulativa De Delincuencia	62.1% área → 100.0% delitos	4.8% área → 72.7% delitos	6.3% área → 72.7% delitos

Nota. Datos extraídos de la Tabla 19. Densidad Delictiva Relativa medida en base a la Densidad Delictiva del Distrito (DDD), la cual es 1.47 delitos/hm².

Al obtener con el algoritmo DBSCAN la densidad delictiva y densidades delictivas relativas más altas con valor de 22.16 delitos/hm² y 15.13 veces la DDD, se procedió a calcular los centroides de las 97 agrupaciones resultantes. El mapa de puntos de la Figura 37 muestra los centroides de cada una de las Zonas Críticas de Delincuencia. La dirección exacta de cada una de las Zonas Crítica de Delincuencia se encuentra en la Tabla 20.

Figura 37

Mapa de Centroides de Zonas Críticas de Delincuencia



Nota. Gráfica generada con el software Jupyter Notebook. Elaboración Propia

Posteriormente, se aplicó el Algoritmo Genético y el Algoritmo Memético para encontrar la ruta mínima de patrullaje. En la Figura 32 se muestra la diferencia en la convergencia de las mejores soluciones de cada algoritmo. En la Tabla 26 se muestran los resultados de los Algoritmo Memético y Genéticos, así como las métricas de evaluación que se utilizó para la elección del Algoritmo Memético.

Tabla 26

Resultados de la Aplicación de los Algoritmos Heurísticos (A. Genético y A. Memético)

	Algoritmo Genético	Algoritmo Memético
Parámetros	<ul style="list-style-type: none"> • 'tamanoPoblacion' = 2000 • 'tamanoElite' = 1000 • 'ratioMutacion' = 0.00005 • 'generaciones' = 1000 	<ul style="list-style-type: none"> • 'tamanoPoblacion' = 200 • 'tamanoElite' = 100 • 'ratioMejora' = 1 • 'generaciones' = 200
Distancia Mínima Encontrada	19.535 km	17.907 km
Ruta Mínima Encontrada	[53, 60, 50, 48, 11, 22, 20, 17, 14, 12, 10, 8, 0, 9, 1, 5, 2, 7, 49, 52, 58, 59, 42, 45, 39, 41, 36, 38, 40, 37, 44, 3, 6, 4, 43, 46, 57, 47, 56, 75, 80, 79, 74, 92, 51, 55, 77, 76, 72, 78, 73, 85, 89, 81, 86, 95, 84, 94, 91, 90, 87, 82, 93, 88, 67, 63, 96, 83, 69, 62, 68, 64, 70, 61, 71, 23, 26, 28, 33, 35, 24, 34, 31, 29, 32, 30, 25, 27, 65, 16, 18, 19, 21, 66, 54, 13, 15]*	[34, 31, 29, 32, 30, 25, 27, 65, 54, 66, 21, 16, 18, 19, 17, 20, 22, 11, 13, 14, 12, 15, 53, 60, 50, 48, 55, 51, 92, 74, 77, 76, 56, 47, 58, 52, 57, 43, 46, 2, 49, 7, 5, 10, 8, 0, 9, 1, 4, 6, 3, 44, 37, 40, 38, 36, 41, 39, 45, 42, 59, 72, 78, 75, 79, 80, 73, 85, 89, 94, 84, 81, 86, 95, 87, 90, 91, 82, 93, 88, 67, 63, 96, 83, 69, 62, 68, 64, 70, 61, 71, 23, 26, 33, 35, 24]*
Tiempo De Ejecución Total	225 s	169 s
Convergencia Promedio	80.7	418.9

Nota. Datos obtenidos con el software Jupyter Notebook. *Cada uno de los números representa una Zona Crítica de Delincuencia, los números inicial y final se llegan a cerrar formando un ciclo.

4.3. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.3.1. Medición de Densidad y Determinación de zonas críticas de delincuencia.

A partir de la aplicación de los algoritmos agrupamiento K-Medias, DBSCAN y HDBSCAN presentados en la Tabla 25, se resalta que el algoritmo DBSCAN obtuvo la mayor densidad delictiva de los 3 algoritmos con un valor de 22.16 delitos/hm², sobre una densidad delictiva de 2.43 delitos/hm² y 16.92 delitos/hm² determinados por los algoritmos K-Medias y HDBSCAN. Adicionalmente, el algoritmo DBSCAN obtuvo una densidad delictiva relativa de 15.13 veces la Densidad Delictiva del Distrito, valor muy superior a la densidad delictiva relativa de 1.66 y 11.55 veces la Densidad Delictiva del Distrito determinados por los algoritmos K-Medias y HDBSCAN. En el caso del agrupamiento hallado por el algoritmo DBSCAN, las áreas involucran un 4.8% área del distrito, sin embargo, concentran un 72.7% de los delitos de hurto y robo, lo cual demuestra la existencia del concepto de la Proporción Acumulativa de Delincuencia propuesto por Weisburd (2015).

4.3.2. Reducción de la Distancia Total de Patrullaje con la Aplicación de Algoritmos de Aprendizaje de Máquina y Heurísticos

La distancia de la ruta de patrullaje de zonas de alto nivel delictivo antes de la Aplicación de los Algoritmos de Máquina y Heurísticos es de 20.784 km. Mediante la Aplicación del Algoritmo Memético, se logró reducir a 17.907 km (13.8 % de reducción sobre la distancia inicial) la distancia total de patrullaje, sin embargo, mediante la Aplicación del Algoritmo Genético se logró reducir a 19.535 km (6% de reducción sobre la distancia inicial). Por lo tanto, la mayor reducción de la distancia total de patrullaje de zonas críticas de delincuencia se obtuvo con la Aplicación del Algoritmo Memético y fue de 2.877 km o 13.8% respecto a la distancia inicial de patrullaje, esta información se observa en la Tabla 27.

Tabla 27

Reducción de la Distancia Total de Patrullaje de Zonas Críticas de Delincuencia de Pueblo Libre

Distancia Total de Patrullaje <u>Antes</u> de la Aplicación de los Algoritmos de Aprendizaje de Máquina y Heurísticos	20.784 km
Distancia Total de Patrullaje <u>Después</u> de la aplicación del Algoritmo Memético	17.907 km
Reducción de la Distancia de Patrullaje de Zonas Críticas de Delincuencia (A. Memético)	2.877 km
Porcentaje de Reducción de la Distancia de Patrullaje de Zonas Críticas de Delincuencia (A. Memético)	13.8%
Distancia Total de Patrullaje <u>Después</u> de la aplicación del Algoritmo Genético	19.535 km
Reducción de la Distancia de Patrullaje de Zonas Críticas de Delincuencia (A. Genético)	1.249 km
Porcentaje de Reducción de la Distancia de Patrullaje de Zonas Críticas de Delincuencia (A. Genético)	6.0%

Nota. Distancia inicial extraída a partir del PADSC de Pueblo Libre, 2023. Distancia final obtenidos a partir del software Jupyter Notebook. Elaboración Propia.

4.3.3. Determinación del camino mínimo de las Zonas Crítica de Delincuencia

A partir de la ruta de patrullaje de zonas críticas obtenida por el Algoritmo Memético, se desarrolló la Tabla 24, en la cual se presentan las vías junto con sus distancias necesarias para realizar todo el camino de patrullaje teniendo una distancia de 17.907 km.

4.4. CONTRASTACIÓN DE HIPÓTESIS

4.4.1. Contrastación de la Hipótesis Específica 1

En la sección 1.4.2 de esta tesis, se formuló la Hipótesis Específica 1 con el siguiente enunciado: “La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permite determinar las zonas críticas de delincuencia en Pueblo Libre”. Y a partir del empleo del algoritmo DBSCAN, se obtuvo los siguientes resultados:

- 97 Zonas Críticas de Delincuencia que involucraron el 72.7% del total de delitos de robo y hurto (Figura 37, Tabla 25).
- Un valor de densidad delictiva promedio de 22.16 delitos/hm² (Tabla 25).
- Un valor de densidad delictiva relativa de 15.13 veces la Densidad Delictiva del Distrito (Tabla 25).
- Una Proporción Acumulativa de Delincuencia en la cual en el 4.8% área del distrito se concentran el 72.7% de los delitos de robo y hurto del distrito (Tabla 25).

Por lo anterior se infiere que:

- ✓ La aplicación de algoritmos de aprendizaje de máquina, en específico el Algoritmo DBSCAN, determinó 97 Zonas Críticas de Delincuencia en Pueblo Libre, las cuales tuvieron una densidad delictiva promedio de 22.16 delitos/hm², una densidad delictiva relativa de 15.13 veces la Densidad Delictiva del Distrito y una Proporción Acumulativa de Delincuencia en la cual el 4.8% del área del distrito concentró el 72.7% de los delitos de robo y hurto del distrito.

4.4.2. Contrastación de la Hipótesis Específica 2

En la sección 1.4.2 de esta tesis, se formuló la Hipótesis Específica 1 con el siguiente enunciado: “La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permite determinar el camino mínimo de patrullaje de las zonas críticas de delincuencia en Pueblo Libre.”

La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos, en específico, el Algoritmo Memético, determinó la ruta del camino mínimo de patrullaje de las Zonas Críticas de Delincuencia en Pueblo Libre (Tabla 24, Tabla 26) considerando la selección de la ejecución con mejor distancia encontrada de las 10 ejecuciones del Algoritmo Memético (Tabla 22), mostrándose el Mapa de la

Trayectoria Mínima del Patrullaje de las Zonas Críticas de Delincuencia en la Figura 35.

4.4.3. Contrastación de la Hipótesis General

La Hipótesis General planteada en el acápite 1.4.1. de la presente tesis fue: “La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos permiten reducir la distancia del camino de patrullaje de las zonas críticas de delincuencia en Pueblo Libre.”

De acuerdo con los datos obtenidos en el Plan de Acción Distrital de Seguridad Ciudadana de Pueblo Libre, se muestran zonas de alta incidencia delictiva, a través de las cuales se presentó en la Tabla 1, en el cual, el camino de patrullaje de las zonas críticas de delincuencia inicial tuvo una distancia de 20.784 km. Posteriormente, a partir de la determinación de las zonas críticas de delincuencia obtenidos por el algoritmo DBSCAN, se aplicaron los algoritmos heurísticos del tipo Algoritmo Genético y Algoritmo Memético, mediante los cuales se obtuvo distancias del camino de patrullaje de teniendo este una distancia de 19.535 km y 17.907 km respectivamente.

A partir de estos resultados, se procedió a realizar una prueba estadística T-Student con los resultados obtenidos por el Algoritmo Memético:

Si $X = \text{Distancia Total de Patrullaje resultante por el Algoritmo Memético}$

A partir de los resultados mostrados en la Tabla 22, se obtuvieron las medidas estadísticas de las distancias obtenidas en las 10 ejecuciones del Algoritmo Memético (Tabla 28):

Tabla 28

Estadísticas de distancias obtenidas por el Algoritmo Memético

MEDIDAS ESTADÍSTICAS	VALOR
Media	18083.4
Moda	18025
Mediana	18083.5
Varianza	7553.6
Desviación Estándar	86.9

Nota. Resultados obtenidos a partir del software Jupyter Notebook. Elaboración Propia

Hipótesis:

$$H_0: \mu \geq 20.784 \text{ km}$$

$$H_1: \mu < 20.784 \text{ km}$$

Nivel de significancia:

$$\alpha = 0.01$$

Estadístico de Prueba:

$$T_c = \frac{\bar{X} - \mu_0}{S/\sqrt{n}} \sim t_{n-1} \quad (27)$$

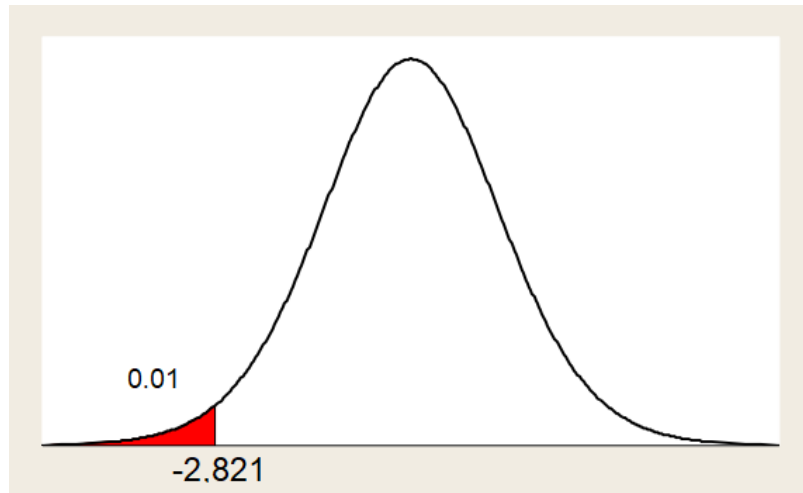
Donde:

- \bar{X} : Promedio de la muestra.
- μ_0 : Valor asumido de la media poblacional en la hipótesis nula.
- S : Desviación estándar.
- n : Tamaño de la muestra.
- $t_{(n-1)}$: D. t de Student con $n - 1$ grados de libertad
- α : Nivel de Significancia.

Calculando el valor $t_{n-1} = t_{10-1} = t_9$ con $\alpha = 0.01$ resultando: $t_9 = -2.821$

Figura 38

Distribución t-Student con 9 grados de libertad



Nota. Elaboración Propia.

Reemplazando los datos en la Ecuación 27, se obtuvo en la Ecuación 28 el valor de T_c igual a -98.27, el cual está dentro de la Zona de Rechazo de la Hipótesis Nula.

$$T_c = \frac{\bar{X} - \mu_0}{S/\sqrt{n}} = \frac{18083.4 - 20784}{86.9/\sqrt{10}} = -98.27 \quad (28)$$

Al obtener un valor menor a -2.821, la decisión fue rechazar H_0 .

Se concluye que, existe suficiente evidencia estadística, con un nivel de significancia de 0.01, para afirmar que la distancia promedio obtenida por el Algoritmo Memético redujo la distancia del camino de patrullaje de zonas críticas de delincuencia a un valor inicial fue de 20.784 km

A partir del análisis anterior, se concluyó:

- ✓ La aplicación del algoritmo heurístico, en específico el algoritmo memético redujo la distancia del camino de patrullaje de las zonas críticas de delincuencia en un 13.8%, de una distancia inicial de 20.784 km a la distancia final de 17.907 km.

CONCLUSIONES

A partir de la contrastación de hipótesis realizada en el capítulo anterior, se brindan como conclusiones de la presente tesis:

- La aplicación del algoritmo heurístico, en específico el algoritmo memético redujo la distancia del camino de patrullaje de las zonas críticas de delincuencia en un 13.8%, de una distancia inicial de 20.784 km a la distancia final de 17.907 km., esta reducción en la distancia del camino de patrullaje puede traducirse en ahorros de tiempo y recursos para el cuerpo de serenazgo, lo que les permitiría cubrir las áreas de mayor densidad delictiva en menos tiempo y con menos recursos.
- La aplicación de algoritmos de Aprendizaje de Máquina y Heurísticos, en específico, el Algoritmo Memético, determinó la ruta del camino mínimo de patrullaje de las Zonas Críticas de Delincuencia en Pueblo Libre (Tabla 24, Tabla 26) considerando la selección de la ejecución con mejor distancia encontrada de las 10 ejecuciones del Algoritmo Memético (Tabla 22), mostrándose el Mapa de la Trayectoria Mínima del Patrullaje de las Zonas Críticas de Delincuencia en la Figura 35.
- La aplicación de algoritmos de aprendizaje de máquina, en específico el Algoritmo DBSCAN, determinó 97 Zonas Críticas de Delincuencia en Pueblo Libre, las cuales tuvieron una densidad delictiva promedio de 22.16 delitos/hm², una densidad delictiva relativa de 15.13 veces la Densidad Delictiva del

Distrito y una Proporción Acumulativa de Delincuencia en la cual el 4.8% del área del distrito concentró el 72.7% de los delitos de robo y hurto del distrito.

RECOMENDACIONES

- Se recomienda que la Gerencia de Seguridad Ciudadana de Pueblo Libre designe a un conjunto de patrullas y motocicletas de serenazgo, para la ejecución de estrategias de patrullaje de zonas críticas de delincuencia desarrollado en la presente investigación, el cual involucran un patrullaje específico en las Zonas Críticas de Delincuencia.
- Se recomienda la implementación de la estrategia de patrullaje de zonas críticas de delincuencia mediante la aplicación de algoritmos de aprendizaje de máquina y heurísticos por parte de los gobiernos locales con alto índice delictivo en los diferentes distritos de Lima, como por ejemplo San Juan de Lurigancho, Los Olivos y Comas, con el objetivo posterior de medir la eficacia de dichas medidas en la reducción de índice delictivos en dichos distritos.
- Se recomienda la implementación de la estrategia de patrullaje de zonas críticas de delincuencia mediante la aplicación de algoritmos de aprendizaje de máquina y heurísticos por parte de las provincias y distritos que han sido oficialmente declarados en situación de emergencia ante el excesivo incremento de la delincuencia, tales como: Trujillo, La Libertad, Arequipa y Chimbote.
- Se recomienda para siguientes investigaciones la aplicación de nuevos algoritmos de agrupamiento, para medir su eficacia en la determinación de Zonas Críticas de Delincuencias en áreas distritales o locales.

- Se recomienda la implementación de nuevas tecnologías de georreferenciación de delitos a nivel gobierno local para un adecuado tratamiento de datos delictivos con el fin de conseguir más patrones de seguimiento y así determinar nuevas estrategias de prevención del delito.

REFERENCIAS BIBLIOGRÁFICAS

- Arias, F. (2012). *El proyecto de investigación. Introducción a la metodología científica. 6ta. edición*. Editorial Episteme.
- Bermúdez, J. *Paralelos y Meridianos* Currículum Nacional.
<https://www.curriculumnacional.cl/portal/Autor/Jose-Alberto-Bermudez/>
- Calinski T. y Harabasz J. (1974). A dendrite method for cluster analysis, *Communications in Statistics-Theory and Methods*, 3(1), 1-27.
- Campello, R. J. G. B., Moulavi, D., & Sander, J. (2013). Density-Based clustering based on hierarchical density estimates. *Lecture Notes in Computer Science* (pp. 160-172). https://doi.org/10.1007/978-3-642-37456-2_14
- Chainey, S., Matias, J., Nunes, F., Da Silva, T., De Macêdo, J., Magalhães, R., De Queiroz Neto, J., y Silva, W. (2021). Improving the creation of hot spot policing patrol routes: Comparing cognitive heuristic performance to an automated spatial computation approach. *ISPRS international journal of geo-information*, 10(8), 560. <https://doi.org/10.3390/ijgi10080560>
- Congreso de la Republica del Perú. (2021). *Ley 31297 de 2021. Por lo cual se expide Ley del Servicio de Serenazgo Municipal*.
- Davies, D., y Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2), 224-227.
<https://doi.org/10.1109/tpami.1979.4766909>

- Eck, J., Chainey, S., Cameron, J., Leitner, M. y Wilson, R. (2005). Mapping Crime: Understanding Hot Spots. Department of Justice, USA.
<https://www.ojp.gov/pdffiles1/nij/209393.pdf>
- Eck, J., y Weisburd, D. (2015). Crime Places in Crime Theory. *Crime and Place: Crime Prevention Studies*, 4. (pp. 1-33) <https://ssrn.com/abstract=2629856>
- Eiben, A. y Smith, J. (2015). Introduction to Evolutionary Computing. Springer.
- Ester, M., Kriegel, H., Sander, J., y Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial Databases with Noise. *Knowledge Discovery and Data Mining*, 226–231.
<https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf>
- Fernández, J. (2023). *Diferencia entre desplazamiento y espacio camino*. Fisicalab.
<https://www.fisicalab.com/apartado/desplazamiento-vs-espacio>
- Galán, V. (2015). Aplicación de la metodología CRISP-DM a un proyecto de minería de datos en el entorno universitario [Tesis de Licenciatura, Universidad Carlos III]. https://uc3m.es/PFC_Victor_Galan_Cortina.pdf
- García, A., Pardo-Balmonete, S., y Saint-Supéry, A. (s. f.). *Diseño Cartográfico*. Universidad Politécnica de Madrid.
<http://redgeomatica.rediris.es/cart02/portada/porta.htm#>
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley Professional.
- Guerra, E., (2022). Aplicación de la minería de datos para determinar la respuesta del dictamen de los clientes de una entidad financiera [Tesis, Universidad Nacional de Ingeniería]. <http://hdl.handle.net/20.500.14076/22901>

- Gutiérrez-García, J.O. [Código Máquina]. (2022). Identifica Clusters con DBSCAN: Algoritmo paso a paso e implementación con Python. YouTube. <https://www.youtube.com/watch?v=HMis89IGdkA>
- Hartl, D. y Ruvolo, M. (2011). *Genetics. Analysis of genes and genomes Burlington*. 8^{va} Edición. Jones & Bartlett Learning.
- Hernández-Sampieri, R., y Mendoza, C. (2020). Metodología de la investigación: las rutas cuantitativa, cualitativa y mixta. McGraw - Hill.
- Instituto Nacional de Estadística e Informática (s.f.) DATACRIM – Sistema Integrado de Estadísticas de la Criminalidad y Seguridad Ciudadana. <https://datacrim.inei.gob.pe/ciudadano/>
- Junta de Andalucía (s.f.) Las Coordinadas. <https://edea.juntadeandalucia.es/>
- Kimerling, A. J., Muehrcke, P. C., Muehrcke, J. O., y Muehrcke, P. M. (2016). *Map use: reading, analysis, interpretation*. ESRI Press Academic.
- Leigh, J., Dunnett, S., y Jackson, L. (2017). Predictive police patrolling to target hotspots and cover response demand. *Annals Of Operations Research*, 283(1-2), 395-410. <https://doi.org/10.1007/s10479-017-2528-x>
- Levitin, A. (2012). *Introduction to The Design and Analysis of Algorithms*. (3^o Ed.) Pearson Education.
- McKee, A. J. (2023). Doc's CJ Glossary. <https://www.docmckee.com/WP/cj/docs-criminal-justice-glossary/>
- Müller, A. C., y Guido, S. (2016). *Introduction to machine learning with Python: A Guide for Data Scientists*. O'Reilly Media, Inc.
- Municipalidad de Pueblo Libre (2019). Declaratoria de Emergencia. <https://portal.muniplibre.gob.pe/pueblo-libre-declara-en-emergencia-la-seguridad-ciudadana-por-90-dias/>

- Municipalidad de Pueblo Libre (2023). Observatorio de Seguridad Ciudadana. <https://portal.muniplibre.gob.pe/observatorio-de-seguridad-ciudadana/>
- Municipalidad de Pueblo Libre (2023). Plan de Acción Distrital de Seguridad Ciudadana de Pueblo Libre (PADSC). <https://portal.muniplibre.gob.pe/wp-content/uploads/2022/12/PADSC-2023-FINAL.pdf>
- Murphy, K. (2012). *Machine Learning: A Probabilistic Perspective*. MIT press.
- Neri, F., Cotta, C., & Moscato, P. (2012). Handbook of Memetic Algorithms. En *Studies in computational intelligence*. <https://doi.org/10.1007/978-3-642-23247-3>
- Pearl, J. (1984). *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison Wesley Publishing Company.
- Project Jupyter. (s. f.). Home. <https://jupyter.org/>
- Real Academia Española. (s.f.) *Diccionario de la lengua española*, 23° Ed., [versión 23.6 en línea]. Recuperado en 31 de octubre de 2023, de <https://dle.rae.es>
- Reis, D., Melo, A., Coelho, A., y Furtado, V. (2006). Towards optimal police patrol routes with genetic algorithms. *Lecture Notes in Computer Science* (pp. 485-491). https://doi.org/10.1007/11760146_45
- Reisig, M. D., y Kane, R. L. (2014). *The Oxford Handbook of Police and Policing*. Oxford University Press eBooks. <https://doi.org/10.1093/oxfordhb/9780199843886.001>
- Robinson, A. H., Morrison, J. L., Muehrcke, P. C., Kimerling, A. J., & Guptill, S. C. (1995). *Elements of cartography*. 6° Ed. John Wiley and Sons. New York.
- Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20, 53-65. [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7)

- Russell, S., & Norvig, P. (2010). *Artificial intelligence: A Modern Approach*. Prentice Hall.
- Sabino, C. A. (1992). El proceso de investigación.
- Sakharov, V. V., Chernyi, S., Saburov, S., & Chertkov, A. A. (2021). Automatization Search for the Shortest Routes in the Transport Network Using the Floyd-Warshall Algorithm. *Transportation Research Procedia*, 54, 1-11. <https://doi.org/10.1016/j.trpro.2021.02.041>
- Samuel, A. (1959). Some studies in machine learning using the game of checkers. *IBM journal of research and development*, 3(3), 210-229. <https://doi.org/10.1147/rd.33.0210>
- Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. *Journal of data warehousing*, 5(4), 13-22.
- Sherman, L. W. (1995). Hot spots of crime and criminal careers of places. *Crime and place*, 4, 35-52.
- Sherman, L. W., & Weisburd, D. (1995). General Deterrent Effects of Police Patrol in Crime "Hot Spots": a randomized, controlled trial. *Justice Quarterly*, 12(4), 625-648. <https://doi.org/10.1080/07418829500096221>
- Swamynathan, M. (2017). *Mastering machine learning with Python in six steps*. Apress eBooks. <https://doi.org/10.1007/978-1-4842-2866-1>
- Tan, P., Steinbach, M., Karpatne, A., y Kumar, V. (2019). *Introduction to Data Mining* (2° Ed.). Pearson.
- Theobald, O. (2019). *Machine learning with Python: A Practical Beginners' Guide*. Independently Published.
- Weisburd, D. (2015). The Law of Crime Concentration And The Criminology Of Place. *Criminology*, 53(2), 133-157. <https://doi.org/10.1111/1745-9125.12070>

- Weisburd, D. y Eck, J., (2004). What can police do to reduce crime, disorder, and fear? En *Annals of the American Academy of Political and Social Science*. Vol. 593 (pp. 42-65). <https://doi.org/10.1177/0002716203262548>
- Weisburd, D., Eck, J. E., y Braga, A. A. (2014). *Place matters*. Cambridge University Press.
- Weisburd, D., & Telep, C. W. (2014). Hot spots policing. *Journal of Contemporary Criminal Justice*, 30(2), 200-220. <https://doi.org/10.1177/1043986214525083>

ANEXOS

Anexo 1 Glosario de la Terminología de Minería de Datos

- **Algoritmo DBSCAN:** Algoritmo de agrupamiento utilizado en minería de datos que agrupa puntos de datos en función de su densidad en un espacio.
- **Algoritmo K-Medias:** Método de agrupamiento utilizado en minería de datos que agrupa puntos de datos en “K” grupos basados en similitudes entre ellos.
- **Algoritmos de Agrupamiento:** Técnicas utilizadas en minería de datos para dividir datos en grupos o clústeres basados en similitudes.
- **Algoritmos de Aprendizaje de Máquina:** Métodos que permiten a las computadoras aprender patrones y tomar decisiones basadas en datos.
- **Algoritmos Genéticos:** Los Algoritmos Genéticos son “técnicas de optimización inspiradas en la evolución biológica que se utilizan en problemas de búsqueda y optimización.”
- **Algoritmos Heurísticos:** Los Algoritmos Heurísticos son “métodos de solución de problemas que utilizan reglas prácticas o enfoques aproximados para encontrar soluciones aceptables en un tiempo razonable.”
- **Algoritmos Meméticos:** Los Algoritmos Meméticos son “un marco computacional basado en poblaciones y comparten una serie de características con métodos como los algoritmos evolutivos y la búsqueda dispersa.”

- **Aprendizaje No Supervisado:** Rama del aprendizaje de máquina en la que el modelo se entrena con datos no etiquetados y encuentra patrones y estructuras en los datos.
- **CRISP-DM (Cross-Industry Standard Process for Data Mining):** Proceso estándar utilizado en minería de datos que define las fases clave para llevar a cabo proyectos de minería de datos.
- **DataFrame:** Estructura de datos tabular utilizada en programación y análisis de datos, a menudo asociada con librerías como Pandas en Python.
- **Datos anormales:** Valores inusuales o atípicos en un conjunto de datos que pueden desviarse significativamente del comportamiento típico.
- **Escalado de Datos:** Proceso de ajustar los valores en un conjunto de datos para que tengan una escala similar, lo que puede ser útil en algoritmos sensibles a la escala. Generalmente el escalado se realiza en una escala de 0 a 1.
- **Genético(a):** Refiere a la analogía con la genética en biología y la evolución. Todo lo relacionado con los genes y la herencia biológica, incluyendo la transmisión de características hereditarias de una generación a otra.
- **Heurística(o):** Regla o método práctico que puede ser útil para resolver problemas, especialmente cuando no se dispone de una solución algorítmica exacta. Técnica de indagación y descubrimiento de soluciones a un problema mediante métodos no rigurosos (tanteo, reglas empíricas, etc.).
- **Librerías:** En el contexto de la programación, las librerías son conjuntos de funciones y herramientas predefinidas que se pueden utilizar para realizar tareas específicas.
- **Minería de Datos:** La minería de datos es el proceso de descubrir patrones y conocimientos útiles en grandes conjuntos de datos utilizando técnicas de análisis y estadísticas.

- **Modelado:** En el contexto de la minería de datos, el modelado se refiere a la creación y evaluación de modelos predictivos basados en datos.
- **Preprocesamiento de Datos (Data Preprocessing):** El preprocesamiento de datos es la etapa en la minería de datos donde se limpian, transforman y preparan los datos para su análisis posterior.

Anexo 2 Código para la Comprensión de los Datos

Importación de Librerías y Carga de Conjunto de Datos

```
import numpy as np
import pandas as pd
import hdbscan
import seaborn as sns
import matplotlib.pyplot as plt
from docx import Document
from docx.shared import Pt
from docx.enum.table import WD_ALIGN_VERTICAL
from geopy.distance import geodesic
from sklearn.preprocessing import MinMaxScaler
from shapely.geometry import Polygon
from scipy.spatial import ConvexHull
from sklearn.cluster import DBSCAN, KMeans, AffinityPropagation
from sklearn.metrics import silhouette_score, davies_bouldin_score, \
    calinski_harabasz_score

#Carga de Conjunto de Datos
delitos = pd.read_csv("datos/Observatorio Seguridad Ciudadana.csv")
limite = pd.read_csv("datos/LimitesDistritoArcGis.csv")
```

Cantidad de Delitos según el Sector Distrital

```
# Calcular la cantidad de delitos por sector
contador_sector = delitos['Sector'].value_counts().sort_index()
# Crear el gráfico de barras
plt.bar(contador_sector.index, contador_sector.values)
plt.xlabel('Sector')
plt.ylabel('Cantidad')
plt.title('Cantidad de Delitos por Sector Distrital')
plt.xticks(rotation=45)
for i, v in enumerate(contador_sector.values):
    plt.text(i, v, str(v), ha='center', va='bottom', fontsize=8)

plt.savefig('Salidas/grafico_barras_sectores.png')
plt.show()
```

Cantidad de Delitos según el Día de la Semana

```
# Asigna datos categóricos al atributo Dia
dias_ordenados = ["Lunes", "Martes", "Miércoles", "Jueves", "Viernes", "Sábado", "Domingo"]
delitos.Dia = pd.Categorical(delitos.Dia, dias_ordenados)
# Calcular la cantidad de delitos por día
contador_dia = delitos['Dia'].value_counts().sort_index()
# Crear el gráfico de barras
plt.bar(contador_dia.index, contador_dia.values)
plt.xlabel('Día de la Semana')
plt.ylabel('Cantidad')
plt.title('Cantidad de Delitos por Día de la Semana')
plt.xticks(rotation=45)
for i, v in enumerate(contador_dia.values):
    plt.text(i, v, str(v), ha='center', va='bottom', fontsize=8)

plt.show()
```

Cantidad de Delitos según el Método de Huida

```
# Calcular la cantidad de delitos según el método de huida
contador_huida = delitos['MetodoDeHuida'].value_counts()
# Crear el gráfico de barras
plt.bar(contador_huida.index, contador_huida.values)
plt.xlabel('Método de Huida')
plt.ylabel('Cantidad')
plt.title('Cantidad de Delitos según el Método de Huida')
plt.xticks(rotation=45)
for i, v in enumerate(contador_huida.values):
    plt.text(i, v, str(v), ha='center', va='bottom', fontsize=8)

plt.show()
```

Cantidad de Delitos según el Mes

```
# Asigna datos categóricos al atributo Mes
meses_ordenados = ["ENERO", "FEBRERO", "MARZO", "ABRIL", "MAYO", "JUNIO", "JULIO", "AGOSTO", "SEPTIEMBRE", "OCTUBRE", "NOVIEMBRE", "DICIEMBRE"]
delitos.Mes = pd.Categorical(delitos.Mes, meses_ordenados)
# Calcular la cantidad de delitos por mes
contador_mes = delitos['Mes'].value_counts().sort_index()
# Crear el gráfico de barras
plt.bar(contador_mes.index, contador_mes.values)
plt.xlabel('Mes')
plt.ylabel('Cantidad')
plt.title('Cantidad de Delitos según el Mes')
plt.xticks(rotation=45)
for i, v in enumerate(contador_mes.values):
    plt.text(i, v, str(v), ha='center', va='bottom', fontsize=8)

plt.show()
```

Cantidad de Delitos según el Lugar de Ocurrencia

```
# Calcular la cantidad de delitos según el lugar de ocurrencia
contador_Lugar = delitos['Lugar'].value_counts().sort_index()
# Crear el gráfico de barras
plt.barh(contador_Lugar.index, contador_Lugar.values)
plt.xlabel('Cantidad')
plt.ylabel('Lugar')
plt.title('Cantidad de Delitos según el Lugar de Ocurrencia')
for i, v in enumerate(contador_Lugar.values):
    plt.text(v, i, str(v), ha='left', va='center', fontsize=9)
plt.show()
```

Cantidad de Delitos según el Medio Utilizado

```
# Calcular la cantidad de delitos según objeto robado/hurtado
contador_objeto = delitos['Objeto'].value_counts().sort_values(ascending=False)
contador_objeto
# Crear el gráfico de barras
plt.barh(contador_objeto.index, contador_objeto.values)
plt.xlabel('Objeto')
plt.ylabel('Cantidad')
plt.title('Cantidad de Delitos según el Objeto Robado/Hurtado')
for i, v in enumerate(contador_objeto.values):
    plt.text(v, i, str(v), ha='left', va='center', fontsize=9)
plt.show()
```

Histograma de la variable longitud

```
# Calcular el histograma de la variable de latitud
hist, edges = np.histogram(delitos['Latitud'], bins=10)
# Crear el histograma
fig, ax = plt.subplots(figsize=(8, 5))
plt.hist(delitos['Latitud'], bins=bins)
for i in range(bins):
    plt.text(edges[i]+0.001, hist[i], str(hist[i]), ha='center', va='bottom',
             fontsize=12)
plt.xlabel('Latitud')
plt.ylabel('Frecuencia')
plt.title('Histograma de Latitud')
plt.show()
```

Histograma de la variable latitud

```
# Calcular el histograma de la variable de longitud
hist, edges = np.histogram(delitos['Longitud'], bins=10)
# Crear el histograma
fig, ax = plt.subplots(figsize=(8, 5))
plt.hist(delitos['Longitud'], bins=bins)
for i in range(bins):
    plt.text(edges[i]+0.0016, hist[i], str(hist[i]), ha='center', va='bottom',
             fontsize=12)
plt.xlabel('Longitud')
plt.ylabel('Frecuencia')
plt.title('Histograma de Variable Longitud')
plt.show()
```

Anexo 3 Código para la Preparación de los Datos

Limpieza de Datos Delictivos

```
# Limpiar "Direccion" de la base de delitos
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.strip()
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.upper()
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.rstrip(' 0')
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace(r'\s+', ' ',
↳ regex=True)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('ABRAM',
↳ 'ABRAHAM', regex=False)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('MANUL',
↳ 'MANUEL', regex=False)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('AV. ',
↳ 'AVENIDA ', regex=False)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('AV ', 'AVENIDA',
↳ regex=False)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('-PUEBLO',
↳ 'LIBRE', '', regex=False)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('PUEBLO LIBRE',
↳ '', regex=False)
loc_delitos['Direccion'] = loc_delitos['Direccion'].str.replace('C.', 'CALLE ',
↳ regex=False)
```

Limpieza de Datos de las Coordenadas Limítrofes

```
# Pre-procesamiento de las coordenadas limítrofes del Distrito
limite['Longitud'] = limite['Longitud'].str.strip()
limite['Latitud'] = limite['Latitud'].str.strip()
limite['Longitud'] = limite['Longitud'].str.replace('W', '', regex=False)
limite['Latitud'] = limite['Latitud'].str.replace('S', '', regex=False)
limite['Latitud'] = pd.to_numeric(limite['Latitud'], errors='coerce')
limite['Longitud'] = pd.to_numeric(limite['Longitud'], errors='coerce')
limite['Longitud'] = limite['Longitud'].apply(lambda x: -x)
limite['Latitud'] = limite['Latitud'].apply(lambda x: -x)
```

Integración y Construcción de Datos

```
# Agregar Datos de Puntos Limítrofes del Distrito de Pueblo Libre
delitosYlimites = pd.concat([loc_delitos, limite], ignore_index=True)
# Definir y Aplicar una función Lambda para crear el indicador de delito o
↳ límite
crea_indicador = lambda x: 1 if pd.notna(x) and x != '' else 0
delitosYlimites['Indicador'] = delitosYlimites['Direccion'].
↳ apply(crea_indicador)
#Utilizar la Función MinMaxScaler para el Escalado de Datos
scaler = MinMaxScaler()
columnas = ['Longitud', 'Latitud']
delitosYlimites[['X', 'Y']] = pd.DataFrame(scaler.
↳ fit_transform(delitosYlimites[columnas]))
```

Formateo de Datos

```
# Agregar una columna 'Id' (Identificador)
delitosYlimites.insert(0, 'Id', range(1, len(delitosYlimites) + 1))
# Multiplicar por 100 a los atributos derivados X e Y
delitosYlimites['X']=delitosYlimites['X']*100
delitosYlimites['Y']=delitosYlimites['Y']*100
```

Anexo 4 Código para el Modelado de los Algoritmos de Agrupamiento

Aplicación y Evaluación del Algoritmo K-Medias

```
#Ejecutar el Algoritmo K-Medias con k = 2,3,4, ... 20
rango = range(2,21)
inercias_KMedias = []
siluetas_KMedias = []
calinski_harabasz_KMedias = []
davies_bouldin_KMedias = []

for k in rango:
    kmeans = KMeans(n_clusters = k, n_init = 100, max_iter = 100).
    ↵fit(seleccion_delitos[['X','Y']])
        # Inercia (Suma de distancias al cuadrado)
        inercias_KMedias.append(kmeans.inertia_)
        # Índice Silueta
        siluetas_KMedias.append(silhouette_score(seleccion_delitos[['X','Y']], ↵
    ↵kmeans.labels_))
        # Índice de Calinski-Harabasz
        calinski_harabasz_KMedias.
    ↵append(calinski_harabasz_score(seleccion_delitos[['X','Y']], kmeans.labels_))
        # Índice de Davies-Bouldin
        davies_bouldin_KMedias.
    ↵append(davies_bouldin_score(seleccion_delitos[['X','Y']], kmeans.labels_))

plt.figure(figsize=(12, 8))
# Subgráfica 1: Método del Codo (Inercia)
plt.subplot(2, 2, 1)
plt.plot(rango, inercias_KMedias, marker='o', markersize =5)
plt.xticks(rango)
plt.title('Método del Codo')
plt.xlabel('Número de Grupos (k)')
plt.ylabel('Inercia')
```

```

# Subgráfica 2: Índice Silueta
plt.subplot(2, 2, 2)
plt.plot(rango, siluetas_KMedias, marker='o', markersize =5)
for indice, valor in enumerate(siluetas_KMedias):
    plt.text(indice+rango[0], valor, str(indice+rango[0]),fontsize=6)
plt.xticks(rango)
plt.title('Método de la Silueta')
plt.xlabel('Número de Grupos (k)')
plt.ylabel('Coeficiente de Silueta')

# Subgráfica 3: Índice de Davies-Bouldin
plt.subplot(2, 2, 3)
plt.plot(rango, davies_bouldin_KMedias, marker='o', markersize =5)
for indice, valor in enumerate(davies_bouldin_KMedias):
    plt.text(indice+rango[0], valor, str(indice+rango[0]),fontsize=6)
plt.xticks(rango)
plt.title('Método de Davies-Bouldin')
plt.xlabel('Número de Grupos (k)')
plt.ylabel('Índice de Davies-Bouldin')

# Subgráfica 4: Índice de Calinski-Harabasz
plt.subplot(2, 2, 4)
plt.plot(rango, calinski_harabasz_KMedias, marker='o', markersize =5)
for indice, valor in enumerate(calinski_harabasz_KMedias):
    plt.text(indice+rango[0], valor, str(indice+rango[0]),fontsize=6)
plt.xticks(rango)
plt.title('Método de Calinski-Harabasz')
plt.xlabel('Número de Grupos (k)')
plt.ylabel('Índice de Calinski-Harabasz')
plt.tight_layout()
plt.show()

```

Aplicación y Evaluación del Algoritmo DBSCAN

```

#Seleccionar las Coordenadas Longitud y Latitud Escaladas de los Delitos
datos = seleccion_delitos[['X', 'Y']]
#Crear el Dataframe para almacenar los Parámetros y Métricas de DBSCAN
resultados_dbscan = pd.DataFrame(columns=['min_samples',
                                         'epsilon',
                                         'silhouette',
                                         'davies_bouldin',
                                         'calinski_harabasz',
                                         'num_grupos',
                                         'num_ruido'])

```



```

def ejecutar_dbscan(data, min_samples, eps):
    global resultados_dbscan # Hacer referencia al DataFrame global
    try:
        # Ejecutar DBSCAN
        model = DBSCAN(min_samples=min_samples, eps=eps).fit(data)
        # Calcular métricas de evaluación
        num_ruido = len(data[model.labels_ == -1])
        num_agrupaciones = len(set(model.labels_)) - (1 if -1 in model.labels_
else 0)
        # Crear un DataFrame con los resultados
        resultado = pd.DataFrame({
            'min_samples': [min_samples],
            'epsilon': [eps],
            'silhouette': [silhouette_score(data, model.labels_)],
            'davies_bouldin': [davies_bouldin_score(data, model.labels_)],
            'calinski_harabasz': [calinski_harabasz_score(data, model.labels_)],
            'num_grupos': [num_agrupaciones],
            'num_ruido': [num_ruido]
        })
        # Concatenar el nuevo DataFrame con el DataFrame global
        resultados_dbscan = pd.concat([resultados_dbscan, resultado],
ignore_index=True)
    except Exception as e:
        print(f"Error al ejecutar DBSCAN: {e}")

```

```

# Generación del modelo DBSCAN con distintos valores de min_samples y eps
for m in range(2,10):
    for n in np.arange(1, 5, 0.05):
        ejecutar_dbscan(datos, min_samples = m, eps = n)
resultados_dbscan

```

Aplicación y Evaluación del Algoritmo HDBSCAN

```

#Crear el Dataframe para almacenar los Parámetros y Métricas de HDBSCAN
resultados_hdbscan = pd.DataFrame(columns=['min_cluster_size',
            'min_samples',
            'epsilon',
            'silhouette',
            'davies_bouldin',
            'calinski_harabasz',
            'num_grupos',
            'num_ruido'])

```

```

def ejecutar_hdbscan(data, min_cluster_size, min_samples, epsilon):
    global resultados_hdbscan

    try:
        # Ejecutar HDBSCAN
        clusterer = hdbscan.HDBSCAN(min_cluster_size = min_cluster_size,
                                    min_samples = min_samples,
                                    cluster_selection_epsilon = epsilon)

        labels = clusterer.fit_predict(data.values)
        # Calcular métricas de evaluación
        silhouette = silhouette_score(data, labels)
        davies_bouldin = davies_bouldin_score(data, labels)
        calinski_harabasz = calinski_harabasz_score(data, labels)
        # Calcular números de grupos y ruido
        num_ruido = len(data[labels == -1])
        num_grupos = len(set(labels)) - (1 if -1 in labels else 0)
        # Almacenar resultados en el DataFrame global
        resultados_hdbscan = pd.concat([resultados_hdbscan, pd.DataFrame({
            'min_cluster_size': [min_cluster_size],
            'min_samples': [min_samples],
            'epsilon': [epsilon],
            'silhouette': [silhouette],
            'davies_bouldin': [davies_bouldin],
            'calinski_harabasz': [calinski_harabasz],
            'num_grupos': [num_grupos],
            'num_ruido': [num_ruido]
        })], ignore_index=True)
    except Exception as e:
        print(f"Error al ejecutar HDBSCAN: {e}")

```

```

def graficar_clusters(data, labels, min_cluster_size, min_samples,
                    cluster_selection_epsilon):

    plt.figure(figsize=(9, 6))
    # Graficar sombras alrededor de los grupos
    delitos = data[labels != -1]
    labels_sin_ruido = labels[labels != -1]
    sns.scatterplot(x='X', y='Y', hue=labels_sin_ruido, palette='viridis',
                  data=delitos, s=30, edgecolor='w')
    plt.plot(seleccion_limites['X'], seleccion_limites['Y'], color='red',
            linewidth=2, linestyle='-', alpha=0.7)
    # Graficar los puntos de ruido
    ruido = data[labels == -1]
    sns.scatterplot(x='X', y='Y', data=ruido, color='red', s=30, edgecolor='w',
                  label='Ruido')
    # Modificar leyendas, ejes y título del gráfico
    plt.legend(loc='upper right', bbox_to_anchor=(1.15, 1))
    plt.xlabel('Longitud Escalada')
    plt.ylabel('Latitud Escalada')
    titulo = f'Agrupamiento HDBSCAN (min_cluster_size={min_cluster_size},
    min_samples={min_samples},
    cluster_selection_epsilon={cluster_selection_epsilon})'
    plt.title(titulo)
    plt.show()

```

```

# Generación del modelo DBSCAN con distintos valores de min_cluster_size,
↳ min_samples y cluster_selection_epsilon
for m in range(2,10):
    for m_s in range(2,5):
        for epsilon in np.arange(1,5,0.05):
            epsilon = float(epsilon)
            ejecutar_hdbscan(datos, min_cluster_size = m, min_samples = m_s,
↳ epsilon = epsilon )

# Generar el score y Guardar el DataFrame en un archivo Excel
resultados_hdbscan["score"] =
↳ ((abs(resultados_hdbscan["silhouette"])+resultados_hdbscan["silhouette"])*5+0.
↳ 5 + ((max(resultados_hdbscan["davies_bouldin"]) -
↳ resultados_hdbscan["davies_bouldin"])/
↳ (max(resultados_hdbscan["davies_bouldin"]) -
↳ min(resultados_hdbscan["davies_bouldin"]))) +
↳ 2*((resultados_hdbscan["calinski_harabasz"] -
↳ min(resultados_hdbscan["calinski_harabasz"]))/
↳ (max(resultados_hdbscan["calinski_harabasz"]) -
↳ min(resultados_hdbscan["calinski_harabasz"])))))/8
resultados_hdbscan.to_excel('salidas/hdbscan_FINAL.xlsx', index=False)

```

```

# Aplicar y Graficar HDBSCAN
clusterer = hdbscan.HDBSCAN(min_cluster_size = 2, min_samples = 2,
↳ cluster_selection_epsilon = 2)
labels = clusterer.fit_predict(datos.values)
graficar_clusters(datos, labels, min_cluster_size = 2, min_samples = 2,
↳ cluster_selection_epsilon = 2)

# Aplicar y Graficar HDBSCAN
clusterer = hdbscan.HDBSCAN(min_cluster_size = 3, min_samples = 2,
↳ cluster_selection_epsilon = 2)
labels = clusterer.fit_predict(datos.values)
graficar_clusters(datos, labels, min_cluster_size = 3, min_samples = 2,
↳ cluster_selection_epsilon = 2)

# Aplicar y Graficar HDBSCAN
clusterer = hdbscan.HDBSCAN(min_cluster_size = 4, min_samples = 2,
↳ cluster_selection_epsilon = 1.85)
labels = clusterer.fit_predict(datos.values)
graficar_clusters(datos, labels, min_cluster_size = 4, min_samples = 2,
↳ cluster_selection_epsilon = 1.85)

```

Anexo 5 Código para el Cálculo de Densidades Delictivas

Cálculo de la Densidad Delictiva (robos y hurtos) de Pueblo Libre

```
# CÁLCULO DE LA DENSIDAD DELICTIVA DEL DISTRITO
# Crear el objeto Polígono que contenga los puntos limítrofes del distrito
poligono_distrito = Polygon(limite)

# Calcular el área del Polígono en grados cuadrados, km2 y hm2
area_distrito = poligono_distrito.area
area_distrito_km2 = area_distrito * (111**2)
area_distrito_hm2 = area_distrito_km2 * 100

# Cálculo de la Densidad Delictiva Distrito
numero_total_delitos = len(loc_delitos)
densidad_delictiva_km2 = numero_total_delitos / area_distrito_km2
densidad_delictiva_hm2 = numero_total_delitos / area_distrito_hm2
print(f"El número de delitos es {numero_total_delitos} delitos y el área del
↳ distrito es {area_distrito_km2:.2f} km\u00B2 o {area_distrito_hm2:.2f}
↳ hm\u00B2")
print(f"La densidad delictiva es {densidad_delictiva_km2:.2f} delitos/km\u00B2
↳ o {densidad_delictiva_hm2:.2f} delitos/hm\u00B2")
```

Función para Calcular la Densidad Delictiva de los Agrupamientos

```
# Función para calcular la Densidad Delictiva de los Agrupamientos
def calcular_densidad(dataframe, nombre_grupo):
    resultados_por_grupo = []
    for grupo, datos_grupo in dataframe.groupby(nombre_grupo):
        #Seleccionar las coordenadas del grupo
        coordenadas_grupo = datos_grupo[['Latitud', 'Longitud']].values
        if(len(coordenadas_grupo) >= 3):
            if len(coordenadas_grupo) < 3:
                diametro = np.linalg.norm(coordenadas_grupo[0] -
        coordenadas_grupo[1])
                area_grados = np.pi * (diametro / 2)**2
            else:
                try:
                    convex_hull_grupo = ConvexHull(coordenadas_grupo)
                    area_grados = convex_hull_grupo.volume
                except QhullError:
                    continue
            area_km2 = area_grados * (111**2)
            area_hm2 = area_km2 * 100

            #Contar el número de delitos en el grupo
            numero_delitos = len(datos_grupo)
            # Calcular la densidad delictiva de cada grupo
            densidad_grupo_hm2 = numero_delitos / area_hm2
            # Calcular la densidad delictiva relativa de cada grupo
            densidad_grupo_relativa = densidad_grupo_hm2 /
        densidad_delictiva_hm2

            # Almacenar resultados en la lista resultados_por_grupo
            resultados_por_grupo.append({
                'Grupo': grupo,
                'Densidad_del_grupo': densidad_grupo_hm2,
                'Densidad_relativa_del_grupo': densidad_grupo_relativa,
                'Numero_Delitos': numero_delitos,
                'Area_hm2': area_hm2
            })

    resultados_df = pd.DataFrame(resultados_por_grupo)

    print("La Densidad Delictiva de cada grupo según el agrupamiento",
    nombre_grupo, "es:")
    for index, row in resultados_df.iterrows():
        print(f"Grupo {int(row['Grupo'])}: Densidad delictiva
        {row['Densidad_del_grupo']:.2f} delitos/hm y con Densidad delictiva
        relativa {row['Densidad_relativa_del_grupo']:.2f} veces")
    return resultados_df
```

Funciones para Calcular las Densidades Delictivas Acumuladas / Densidades Delictivas Acumuladas Relativas y la Proporción Acumulativa de Delincuencia

```
# Función para calcular la Densidad Delictiva Acumulada y la Densidad Delictiva
↳ Acumulada Relativa
def calcular_densidad_acumulada(dataframe):
    densidad_acumulada = dataframe['Numero_Delitos'].sum() /
↳ dataframe['Area_hm2'].sum()
    densidad_acumulada_relativa = densidad_acumulada/densidad_delictiva_hm2
    print(f"La Densidad Delictiva Acumulada es: {densidad_acumulada:.2f}
↳ delitos/hm\u00B2")
    print(f"La Densidad Delictiva Acumulada Relativa es:
↳ {densidad_acumulada_relativa:.2f} veces")

# Función para crear un campo dentro de un dataframe dado, que contenga la
↳ variable densidad_relativa
def calcular_densidad_relativa(dataframe):
    dataframe['Densidad_Relativa'] = dataframe['Densidad_del_grupo'] /
↳ densidad_delictiva_hm2

# Función que elimina los registros con Área menores a 0.01 hm2
def limpiar_densidad(dataframe_densidades):
    # Filtrar registros con Area_hm2 >= 0.01
    dataframe_limpiado = dataframe_densidades[dataframe_densidades['Area_hm2']
↳ >= 0.01]
    # Verificar si hay cambios
    if len(dataframe_limpiado) == len(dataframe_densidades):
        print("No hay cambios en la limpieza de datos")
        return dataframe_densidades
    else:
        # Calcular la diferencia en el número de delitos
        diferencia_delitos = dataframe_densidades['Numero_Delitos'].sum() -
↳ dataframe_limpiado['Numero_Delitos'].sum()
        # Calcular la cantidad de grupos eliminados
        grupos_eliminados = len(dataframe_densidades) - len(dataframe_limpiado)
        print(f"Hay cambios - Diferencia en número de delitos:
↳ {diferencia_delitos}, Grupos eliminados: {grupos_eliminados}")
        return dataframe_limpiado

def generar_proporcion_acumulativa_delincuencia(densidades_delictivas_limpiado):
    proporcion_area = densidades_delictivas_limpiado['Area_hm2'].sum() /
↳ area_distrito_hm2 * 100
    proporcion_delitos = densidades_delictivas_limpiado['Numero_Delitos'].sum()
↳ / numero_total_delitos * 100
    texto = f"El {proporcion_area:.1f}% área se concentra en el
↳ {proporcion_delitos:.1f}% de los delitos."
    return texto
```

Función para Crear el Reporte de las Densidades Delictivas / Densidades Delictivas Relativas de cada Agrupamiento

```
def reporteWordAgrupamiento(dataframe, agrupamiento):
    # Crear un documento de Word
    doc = Document()
    doc.add_heading(f'Densidades de Agrupamiento - {agrupamiento}', 0)

    # Filtrar solo los campos 'Grupo', 'Densidad_del_grupo',
    ↪ 'Densidad_relativa_del_grupo'
    densidades_delictivas_filtro = dataframe[['Grupo',
    ↪ 'Densidad_del_grupo', 'Densidad_relativa_del_grupo']]
    # Agregar tabla al documento de Word
    table = doc.add_table(rows=densidades_delictivas_filtro.shape[0] + 1,
    ↪ cols=densidades_delictivas_filtro.shape[1])
    # Establecer estilo
    table.style = 'Table Grid'
    table.autofit = False
    # Establecer fuente Arial 11 para la tabla
    for row in table.rows:
        for cell in row.cells:
            for paragraph in cell.paragraphs:
                for run in paragraph.runs:
                    run.font.name = 'Arial'
                    run.font.size = Pt(11)
    # Agregar encabezados de columna
    for i, column in enumerate(densidades_delictivas_filtro.columns):
        table.cell(0, i).text = column
    # Recorrer cada celda de la Tabla Word para almacenar las Densidades
    ↪ Delictivas
    for i in range(densidades_delictivas_filtro.shape[0]):
        for j in range(densidades_delictivas_filtro.shape[1]):
            value = densidades_delictivas_filtro.iloc[i, j]

            if densidades_delictivas_filtro.columns[j] in
    ↪ ['Densidad_del_grupo', 'Densidad_relativa_del_grupo']:
                value = round(value, 2)
                if densidades_delictivas_filtro.columns[j]
    ↪ == 'Densidad_del_grupo':
                    value = f"{value} delitos/hm\u00B2"
                else:
                    value = f"{value} veces"
                if densidades_delictivas_filtro.columns[j] == 'Grupo' and not pd.
    ↪ isnull(value):
                    value = int(value)
                    table.cell(i + 1, j).text = str(value)

    doc.save(f'reporte_agrupamiento_{agrupamiento}.docx')
```

Cálculo de las Densidades Delictivas / Densidades Delictivas Relativas del Agrupamiento obtenido con K-Medias (K=3)

```
# Integra las coordenadas geográficas y los grupos K-Medias en un solo dataframe
col0 = ['Id','Longitud', 'Latitud']
col1 = ['KMedias_3', 'KMedias_17']
resultado_KMedias = seleccion_delitos.merge(delitosYlimites[col0], on='Id',
↳how='inner')[col0 + col1]
densidades_delictivas = calcular_densidad(resultado_KMedias, 'KMedias_3')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
calcular_densidad_acumulada(densidades_delictivas_limpiado)
calcular_densidad_relativa(densidades_delictivas_limpiado)
```

Cálculo de las Densidades Delictivas / Densidades Delictivas Relativas del Agrupamiento obtenido con K-Medias (K=17)

```
# Integra las coordenadas geográficas y los grupos K-Medias en un solo dataframe
col0 = ['Id','Longitud', 'Latitud']
col1 = ['KMedias_3', 'KMedias_17']
resultado_KMedias = seleccion_delitos.merge(delitosYlimites[col0], on='Id',
↳how='inner')[col0 + col1]
densidades_delictivas = calcular_densidad(resultado_KMedias, 'KMedias_17')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
calcular_densidad_acumulada(densidades_delictivas_limpiado)
calcular_densidad_relativa(densidades_delictivas_limpiado)
```

Cálculo de las Densidades Delictivas / Densidades Delictivas Relativas del Agrupamiento obtenido con DBSCAN (min_samples = 2, eps = 2.10)

```
# Integra las coordenadas geográficas y los grupos DBSCAN en un solo dataframe
col0 = ['X','Y', 'Longitud', 'Latitud']
col1 = ['Latitud','Longitud','DBSCAN']
resultado_DBSCAN = datos_con_labels.merge(delitosYlimites[col0], on=['X', 'Y'],
↳how='inner')[col1]
densidades_delictivas = calcular_densidad(resultado_DBSCAN, 'DBSCAN')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
calcular_densidad_acumulada(densidades_delictivas_limpiado)
densidades_delictivas_limpiado.sort_values(by='Densidad_del_grupo', ascending =
↳False)
reporteWordAgrupamiento(densidades_delictivas, "DBSCAN")
```


Cálculo de las Densidades Delictivas / Densidades Delictivas Relativas del Agrupamiento obtenido con HDBSCAN (min_cluster_size = 2, min_samples = 2, cluster_selection_epsilon = 2)

```
# Integra las coordenadas geográficas y los grupos HDBSCAN en un solo dataframe
col0 = ['X', 'Y', 'HDBSCAN']
col1 = ['Latitud', 'Longitud', 'HDBSCAN']

# Aplicar HDBSCAN con min_cluster_size = 2, min_samples = 2,
↳ cluster_selection_epsilon = 2
clusterer = hdbscan.HDBSCAN(min_cluster_size = 2, min_samples = 2,
↳ cluster_selection_epsilon = 2)
labels1 = clusterer.fit_predict(datos.values)
resultados_HDBSCAN1 = datos.copy()
resultados_HDBSCAN1['HDBSCAN'] = labels1
resultado_HDBSCAN_2_2_2 = delitosYlimites.merge(resultados_HDBSCAN1[col0],
↳ on=['X', 'Y'], how='inner')[col1]
densidades_delictivas = calcular_densidad(resultado_HDBSCAN_2_2_2, 'HDBSCAN')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
calcular_densidad_acumulada(densidades_delictivas_limpiado)
reporteWordAgrupamiento(densidades_delictivas, "HDBSCAN")
```

Cálculo de las Densidades Delictivas / Densidades Delictivas Relativas del Agrupamiento obtenido con HDBSCAN (min_cluster_size = 3, min_samples = 2, cluster_selection_epsilon = 2)

```
# Aplicar HDBSCAN min_cluster_size = 3, min_samples = 2,
↳ cluster_selection_epsilon = 2
clusterer = hdbscan.HDBSCAN(min_cluster_size = 3, min_samples = 2,
↳ cluster_selection_epsilon = 2)
labels2 = clusterer.fit_predict(datos.values)
resultados_HDBSCAN2 = datos.copy()
resultados_HDBSCAN2['HDBSCAN'] = labels2
resultado_HDBSCAN_3_2_2 = delitosYlimites.merge(resultados_HDBSCAN2[col0],
↳ on=['X', 'Y'], how='inner')[col1]
densidades_delictivas = calcular_densidad(resultado_HDBSCAN_3_2_2, 'HDBSCAN')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
calcular_densidad_acumulada(densidades_delictivas_limpiado)
reporteWordAgrupamiento(densidades_delictivas_limpiado, "HDBSCAN2")
```

Cálculo de las Densidades Delictivas / Densidades Delictivas Relativas del Agrupamiento obtenido con HDBSCAN (min_cluster_size = 4, min_samples = 2, cluster_selection_epsilon = 1.85)

```
# Aplicar HDBSCAN min_cluster_size = 4, min_samples = 2,
↳ cluster_selection_epsilon = 1.85
clusterer = hdbSCAN.HDBSCAN(min_cluster_size = 4, min_samples = 2,
↳ cluster_selection_epsilon = 1.85)
labels3 = clusterer.fit_predict(datos.values)
resultados_HDBSCAN3 = datos.copy()
resultados_HDBSCAN3['HDBSCAN'] = labels3
resultado_HDBSCAN_4_2_185 = delitosYlimites.merge(resultados_HDBSCAN3[col0],
↳ on=['X', 'Y'], how='inner')[col1]
densidades_delictivas = calcular_densidad(resultado_HDBSCAN_4_2_185, 'HDBSCAN')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
calcular_densidad_acumulada(densidades_delictivas_limpiado)
```

Generación de los Indicadores de Densidad Delictiva

```
#Generar todos indicadores de la Tabla 18
print("# De Grupos: ")
↳ ", len(densidades_delictivas_limpiado))
print("# Registros Ruido: ", len(loc_delitos) -
↳ densidades_delictivas_limpiado['Numero_Delitos'].sum())
calcular_densidad_acumulada(densidades_delictivas_limpiado)
print("Área Acumulada de los Grupos ",
↳ round(densidades_delictivas_limpiado["Area_hm2"].sum(), 2), "hm\u00B2")
proporcion_acumulativa_delincuencia =
↳ generar_proporcion_acumulativa_delincuencia(densidades_delictivas_limpiado)
print(proporcion_acumulativa_delincuencia)
```

Anexo 6 Código para Determinar las Zonas Críticas de Delincuencia

Preparación de la Información Requerida de las Zonas Críticas de Delincuencia

```
# Preparar el DataFrame de los delitos agrupados con DBSCAN
# Integra las coordenadas geográficas y los grupos DBSCAN en un solo dataframe
col0 = ['X','Y', 'Longitud', 'Latitud']
col1 = ['Latitud','Longitud','DBSCAN']
resultado_DBSCAN = datos_con_labels.merge(delitosYlimites[col0], on=['X', 'Y'],
↳how='inner')[col1]
densidades_delictivas = calcular_densidad(resultado_DBSCAN, 'DBSCAN')
calcular_densidad_acumulada(densidades_delictivas)
densidades_delictivas_limpiado = limpiar_densidad(densidades_delictivas)
grupos_DBSCAN = densidades_delictivas_limpiado[['Grupo']].copy()
grupos_DBSCAN.rename(columns={'Grupo': 'DBSCAN'}, inplace=True)
agrupamiento_DBSCAN = resultado_DBSCAN.merge(grupos_DBSCAN,
↳on='DBSCAN')[['DBSCAN', 'Longitud', 'Latitud']]
agrupamiento_DBSCAN['Grupo_ID'] = agrupamiento_DBSCAN['DBSCAN'].
↳astype('category').cat.codes
agrupamiento_DBSCAN = agrupamiento_DBSCAN[['Grupo_ID', 'Longitud', 'Latitud',
↳'DBSCAN']]
print(agrupamiento_DBSCAN)
```

Graficar los Delitos pertenecientes a cada Zona Crítica de Delincuencia y sus respectivos centroides

```
# Agrupar por Grupo_ID y calcular los centroides de longitud y latitud
centroides = agrupamiento_DBSCAN.groupby('Grupo_ID')[['Longitud', 'Latitud']].
↳mean()
# Graficar las Zonas Críticas de Delincuencia
plt.scatter(agrupamiento_DBSCAN['Longitud'], agrupamiento_DBSCAN['Latitud'],
↳c=agrupamiento_DBSCAN['Grupo_ID'], cmap='tab10', marker='.')
for index, row in centroides.iterrows():
    plt.text(row['Longitud'], row['Latitud'], str(index), ha='center',
↳va='center')
plt.plot(limite['Longitud'], limite['Latitud'], color='red', linewidth=2,
↳linestyle='-', alpha=0.7)
plt.xlabel('Longitud')
plt.ylabel('Latitud')
plt.title('Zonas Críticas de Delincuencia')
plt.show()
# Graficar los Centroides de las Zonas Críticas de Delincuencia
plt.scatter(centroides['Longitud'], centroides['Latitud'], c='red', s=10,
↳marker='v')
for index, row in centroides.iterrows():
    plt.text(row['Longitud'], row['Latitud'], str(index), ha='center',
↳va='center')
plt.plot(limite['Longitud'], limite['Latitud'], color='red', linewidth=2,
↳linestyle='-', alpha=0.7)
plt.xlabel('Longitud')
plt.ylabel('Latitud')
plt.title('Centroides de las Zonas Críticas de Delincuencia')
plt.show()
```

Generar Tabla 20 para determinación de la localización de cada Zona Crítica de Delincuencia

```
# Truncar la longitud y latitud a 6 decimales
centroides = centroides.round(6)
# Crear un documento de Word, agregar título y crear tabla
doc = Document()
doc.add_heading('Centroides', level=1)
tabla = doc.add_table(rows=1, cols=len(centroides.columns) + 1)
# Agregar encabezados a la tabla
encabezados = tabla.rows[0].cells
encabezados[0].text = 'Grupo_ID'
for i, columna in enumerate(centroides.columns):
    encabezados[i+1].text = columna
# Agregar datos a la tabla
for indice, fila in centroides.iterrows():
    fila_nueva = tabla.add_row().cells
    fila_nueva[0].text = str(indice)
    for i, valor in enumerate(fila):
        fila_nueva[i+1].text = str(valor)
# Guardar el documento
doc.save('centroides.docx')
# Generar el archivo csv para exportarlo a ArcGis Pro
centroides.to_csv('centroides.csv')
# Convierte el DataFrame en una Lista
centroides_lista = [(longitud, latitud) for longitud, latitud in
    zip(centroides['Longitud'], centroides['Latitud'])]
```

Generación del Archivo con las Información Geográfica de cada Zona Crítica de Delincuencia

```
# Copia las Coordenadas de los Centroides de las Zonas Críticas de Delincuencia
zonas_criticas = centroides.copy()
# Calcular la menor longitud y latitud y la distancia en metros respecto a ellas
min_longitud = zonas_criticas['Longitud'].min()
min_latitud = zonas_criticas['Latitud'].min()
zonas_criticas['x'] = zonas_criticas.apply(lambda row:
    geodesic((row['Latitud'], min_longitud), (row['Latitud'], row['Longitud'])).
    meters, axis=1)
zonas_criticas['y'] = zonas_criticas.apply(lambda row: geodesic((min_latitud,
    row['Longitud']), (row['Latitud'], row['Longitud'])).meters, axis=1)
zonas_criticas['x'] = zonas_criticas['x'].round(1)
zonas_criticas['y'] = zonas_criticas['y'].round(1)
zonas_criticas.to_csv('zonas_criticas.csv')
```

Anexo 7 Código del Desarrollo del Algoritmo Genético

Creación de Variables Iniciales para los Algoritmos Heurísticos (Algoritmo Genético y Memético)

```
#Carga de Zonas Críticas de Delincuencia
zonas_criticas = pd.read_csv('zonas_criticas.csv', index_col='Grupo_ID')
#Creación de Nodos
lon = zonas_criticas["Longitud"].to_numpy()
lat = zonas_criticas["Latitud"].to_numpy()
n= len(zonas_criticas)
global zonas
zonas = [i for i in range(n)]
#Creación de Arcos
arcos = [(i, j) for i in range(n) for j in range(n) if i != j]
x = zonas_criticas["x"].to_numpy()
y = zonas_criticas["y"].to_numpy()
#Creación de la Matriz de Distancias
global matrizDistancia
matrizDistancia = {(i,j):np.hypot(x[i]-x[j],y[i]-y[j]) for i,j in arcos}
```

Funciones para la Generación de Gráficos

```
#Funciones Complementarias a los Algoritmos Genético y Memético
def graficarRuta(ruta):
    global limite, lon, lat
    distancia_total = calcularDistancia(ruta)
    titulo = f"Ruta del Problema del Agente Viajero - TSP\nDistancia Total de
    ↪Patrullaje: {round(distancia_total,2)}"
    plt.figure(figsize=(10, 8))
    plt.plot(limite["Longitud"], limite["Latitud"], color='black', ↪
    ↪linestyle='-', linewidth=2)
    for n in ruta:
        plt.annotate(str(n), xy=(lon[n], lat[n]), xytext=(lon[n] + 0.0001, ↪
    ↪lat[n] + 0.0002), color='red')
    ruta1 = ruta + [ruta[0]]
    ruta_x = [lon[i] for i in ruta1]
    ruta_y = [lat[i] for i in ruta1]
    plt.plot(ruta_x, ruta_y, linestyle='-', marker='o', color='green')
    plt.xlabel("Coordenadas X - Longitud")
    plt.ylabel("Coordenadas Y - Latitud")
    plt.title(titulo)
    plt.show()
def graficarFuncionFitness(progreso):
    plt.plot(progreso)
    plt.ylabel('Distancia Mínima de Patrullaje en Cada Generación (m)')
    plt.xlabel('Generación')
    plt.title('Evolución de la Distancia Mínima de la Ruta de Patrullaje para
    ↪cada Generación')
    plt.ticklabel_format(style='plain', axis='y')
    plt.tight_layout()
    plt.show()
# Definir función para graficar el progreso de la función fitness
def graficarProgresoEjecuciones(df_resultados):
    plt.figure(figsize=(10, 6))
    for i, row in df_resultados.iterrows():
        plt.plot(row['Progreso'], label=f'Ejecución # {row["Corrida"]}')
    plt.xlabel('Generación')
    plt.ylabel('Distancia Mínima de Patrullaje en cada Generación')
    plt.legend()
    plt.grid(True)
    plt.tight_layout()
    plt.show()
```

Funciones de los Operadores del Algoritmo Genético

```
# Creación de las funciones necesarias
def calcularDistancia(ruta):
    distancia = sum(matrizDistancia[(ruta[i], ruta[i + 1])] for i in
    range(len(ruta) - 1))
    distancia += matrizDistancia[(ruta[-1], ruta[0])]
    return distancia
def crearRuta(zonas):
    return random.sample(zonas, len(zonas))
def crearPoblacion(tamanoPoblacion):
    poblacion = [crearRuta(zonas) for _ in range(tamanoPoblacion)]
    return poblacion
def rankeaRutas(poblacion):
    resultadosFitness = {i: calcularDistancia(poblacion[i]) for i in
    range(len(poblacion))}
    return sorted(resultadosFitness.items(), key=lambda x: x[1])
def seleccionarIndividuos(poblacionRankeada, tamanoElite):
    resultadosSeleccionados = [poblacionRankeada[i][0] for i in
    range(tamanoElite)]
    aptitudes_acumuladas = np.cumsum([aptitud for indice, aptitud in
    poblacionRankeada])
    suma_aptitudes = sum(aptitud for indice, aptitud in poblacionRankeada)
    porcentajes_acumulados = [100 * suma / suma_aptitudes for suma in
    aptitudes_acumuladas]
    for _ in range(len(poblacionRankeada) - tamanoElite):
        pick = 100 * random.random()
        for i in range(len(poblacionRankeada)):
            if pick <= porcentajes_acumulados[i]:
                resultadosSeleccionados.append(poblacionRankeada[i][0])
                break
    return resultadosSeleccionados
def piscinaApareamiento(poblacion, resultadoSeleccion):
    return [poblacion[i] for i in resultadoSeleccion]
def cruce(padre1, padre2):
    genInicial, genFinal = sorted(random.sample(range(len(padre1)), 2))
    hijoP1 = padre1[genInicial:genFinal]
    hijoP2 = [item for item in padre2 if item not in hijoP1]
    hijo = hijoP1 + hijoP2
    return hijo
def cruzarPoblacion(piscinaApareamiento, tamanoElite):
    hijos = piscinaApareamiento[:tamanoElite]
    length = len(piscinaApareamiento) - tamanoElite
    piscina = random.sample(piscinaApareamiento, len(piscinaApareamiento))
    for i in range(length):
        hijo = cruce(piscina[i], piscina[len(piscinaApareamiento)-i-1])
        hijos.append(hijo)
    return hijos
```

```

def mutar(individuo, ratioMutacion):
    for i, ciudad in enumerate(individuo):
        if random.random() < ratioMutacion:
            intercambiar_con = random.randint(0, len(individuo) - 1)
            individuo[i], individuo[intercambiar_con] =
↪ individuo[intercambiar_con], ciudad
    return individuo

def mutarPoblacion(poblacion, ratioMutacion):
    poblacionMutada = []
    for ind in range(0, len(poblacion)):
        indMutado = mutar(poblacion[ind], ratioMutacion)
        poblacionMutada.append(indMutado)
    return poblacionMutada

def genSiguiente(poblacion, tamanoElite, ratioMutacion):
    poblacionRankeada = rankeaRutas(poblacion)
    resultadoSeleccion = seleccionarIndividuos(poblacionRankeada, tamanoElite)
    piscina = piscinaApareamiento(poblacion, resultadoSeleccion)
    hijos = cruzarPoblacion(piscina, tamanoElite)
    poblacionMutada = mutarPoblacion(hijos, ratioMutacion)
    return poblacionMutada

# Función de Implementación del Algoritmo Genético
def algoritmoGenetico(tamanoPoblacion, tamanoElite, ratioMutacion,
↪ generaciones):
    poblacion = crearPoblacion(tamanoPoblacion)
    progreso = [rankeaRutas(poblacion)[0][1]]
    print('Distancia Inicial: ' + str(round(progreso[0] / 1000, 3)) + ' km')
    for i in range(1, generaciones + 1):
        poblacion = genSiguiente(poblacion, tamanoElite, ratioMutacion)
        progreso.append(rankearutas(poblacion)[0][1])
        if i % 50 == 0:
            print('Distancia Mínima de la Generación ' + str(i), ': ',
↪ round(progreso[i] / 1000, 3), 'km')
        indiceMejorRuta = rankeaRutas(poblacion)[0][0]
        mejorRuta = poblacion[indiceMejorRuta]
    return mejorRuta, progreso

```


Función para Ejecutar 10 veces el Algoritmo Genético

```
def ejecutarAlgoritmoGenetico(num_corridas, tamanoPoblacion, tamanoElite,
ratioMutacion, generaciones):
    resultados = []
    for i in range(num_corridas):
        tiempo_inicial = time.time()
        print('Corrida #',i)
        mejorRuta, progreso = algoritmoGenetico(tamanoPoblacion, tamanoElite,
ratioMutacion, generaciones)
        tiempo_ejecucion = time.time() - tiempo_inicial
        resultados.append({
            'Corrida': i + 1,
            'Progreso': progreso,
            'MejorRuta': mejorRuta,
            'DistanciaMejorRuta': calcularDistancia(mejorRuta),
            'TiempoEjecucion': tiempo_ejecucion
        })
    df_resultados = pd.DataFrame(resultados)
    return df_resultados
```

Aplicación de las 10 Ejecuciones del Algoritmo Genético y mostrar mejor solución

```
# Ejecutar el algoritmo genético para 10 ejecuciones
num_corridas = 10
tamanoPoblacion = 2000
tamanoElite = 1000
ratioMutacion = 0.00005
generaciones = 1000
resultadosAGenetico = ejecutarAlgoritmoGenetico(num_corridas, tamanoPoblacion,
tamanoElite, ratioMutacion, generaciones)
resultadosAGenetico.to_csv('resultadosAlgoritmoGenetico.csv', index=False)
print(resultadosAGenetico)
```

```
# Selección de la Ruta con menor Distancia Total de Patrullaje
indice = resultadosAGenetico['DistanciaMejorRuta'].idxmin()
filaMenorRuta = resultadosAGenetico.loc[indice]
mejorRutaAlgoritmoGenético = filaMenorRuta['MejorRuta']
print("La mejor ruta del algoritmo genético es:", mejorRutaAlgoritmoGenético)
# Graficar Las Funciones Fitness de las Ejecuciones del Algoritmo Genético
graficarProgresoEjecuciones(resultadosAGenetico)
```

Anexo 8 Código del Desarrollo del Algoritmo Memético

Funciones de los Operadores del Algoritmo Memético

```
# Algoritmo de Búsqueda Local 2-Opt
def BL_2opt(ruta):
    min_cambio = 0
    ruta_cerrada = ruta + [ruta[0]]
    for i in range(len(ruta_cerrada) - 2):
        for j in range(i + 2, len(ruta_cerrada) - 1):
            costo_actual = matrizDistancia[(ruta_cerrada[i], ruta_cerrada[i +
↵1])] + matrizDistancia[(ruta_cerrada[j], ruta_cerrada[j + 1])]
            costo_nuevo = matrizDistancia[(ruta_cerrada[i], ruta_cerrada[j])] +
↵matrizDistancia[(ruta_cerrada[i + 1], ruta_cerrada[j + 1])]
            cambio = costo_nuevo - costo_actual
            if cambio < min_cambio:
                min_cambio = cambio
                min_i = i
                min_j = j
    if min_cambio < 0:
        ruta_cerrada[min_i + 1:min_j + 1] = ruta_cerrada[min_i + 1:min_j + 1][:
↵-1]
    return ruta_cerrada[:-1]
def mejorarPoblacion(poblacion, ratioMejora):
    poblacionMejorada = []
    for ind in poblacion:
        if random.random() < ratioMejora:
            poblacionMejorada.append(BL_2opt(ind))
        else:
            poblacionMejorada.append(ind)
    return poblacionMejorada
```

```
# Función de Generación Siguiete
def genSiguieteAlgMemetico(poblacion, tamanoElite, ratioMejora):
    poblacionRankeada = rankeaRutas(poblacion)
    resultadoSeleccion = seleccionarIndividuos(poblacionRankeada, tamanoElite)
    piscina = piscinaApareamiento(poblacion, resultadoSeleccion)
    hijos = cruzarPoblacion(piscina, tamanoElite)
    poblacionMejorada = mejorarPoblacion(hijos, ratioMejora)
    return poblacionMejorada
# Función de Implementación del Algoritmo Memético
def algoritmoMemetico(tamanoPoblacion, tamanoElite, ratioMejora, generaciones):
    poblacion = crearPoblacion(tamanoPoblacion)
    progreso = [rankeaRutas(poblacion)[0][1]]
    print('Distancia Inicial: ' + str(round(progreso[0] / 1000, 3)) + ' km')
    for i in range(1, generaciones + 1):
        poblacion = genSiguieteAlgMemetico(poblacion, tamanoElite, ratioMejora)
        progreso.append(rankearutas(poblacion)[0][1])
        if i % 20 == 0:
            print('Distancia Mínima de la Generación ' + str(i), ': ',
↵round(progreso[i] / 1000, 3), 'km')
        indiceMejorRuta = rankeaRutas(poblacion)[0][0]
        mejorRuta = poblacion[indiceMejorRuta]
    return mejorRuta, progreso
```

Función para Ejecutar 10 veces el Algoritmo Memético

```
def ejecutarAlgoritmoMemetico(num_corridas, tamanoPoblacion, tamanoElite,
    ratioMejora, generaciones):
    resultados = []
    for i in range(num_corridas):
        tiempo_inicial = time.time()
        print('Corrida #',i)
        mejorRuta, progreso = algoritmoMemetico(tamanoPoblacion, tamanoElite,
    ratioMejora, generaciones)
        tiempo_ejecucion = time.time() - tiempo_inicial
        resultados.append({
            'Corrida': i + 1,
            'Progreso': progreso,
            'MejorRuta': mejorRuta,
            'DistanciaMejorRuta': calcularDistancia(mejorRuta),
            'TiempoEjecucion': tiempo_ejecucion
        })
    df_resultados = pd.DataFrame(resultados)
    return df_resultados
```

Aplicación de las 10 Ejecuciones del Algoritmo Genético y mostrar mejor Solución

```
# Ejecutar el algoritmo memético para 10 ejecuciones
num_corridas = 10
tamanoPoblacion = 200
tamanoElite = 100
ratioMejora = 1
generaciones = 200
resultadosAMemetico = ejecutarAlgoritmoMemetico(num_corridas, tamanoPoblacion,
    tamanoElite, ratioMejora, generaciones)
resultadosAMemetico.to_csv('resultadosAlgoritmoMemetico.csv', index=False)
print(resultadosAMemetico)
```

```
# Selección de la Ruta con menor Distancia Total de Patrullaje
indice = resultadosAMemetico['DistanciaMejorRuta'].idxmin()
filaMenorRuta = resultadosAMemetico.loc[indice]
mejorRutaAlgoritmoMemetico = filaMenorRuta['MejorRuta']
print("La mejor ruta del algoritmo genético es:", mejorRutaAlgoritmoMemetico)
# Graficar Las Funciones Fitness de las Ejecuciones del Algoritmo Genético
graficarProgresoEjecuciones(resultadosAMemetico)
```

Anexo 9 Código de Evaluación de los Algoritmos Genético y Memético

Generación de la Gráfica de la Función Fitness para la Evaluación de sus Convergencias

```
#Graficar Funciones Fitness de las Mejores Soluciones
indGenetico = resultadosAGenetico['DistanciaMejorRuta'].idxmin()
mejorSolucionGenetico = resultadosAGenetico.loc[indGenetico]
indMemetico = resultadosAMemetico['DistanciaMejorRuta'].idxmin()
mejorSolucionMemetico = resultadosAMemetico.loc[indMemetico]
plt.plot(mejorSolucionGenetico['Progreso'], label='Algoritmo Genético',
        color='orange')
plt.plot(mejorSolucionMemetico['Progreso'], label='Algoritmo Memético',
        color='green')
minFitnessGenetico = min(mejorSolucionGenetico['Progreso'])
minFitnessMemetico = min(mejorSolucionMemetico['Progreso'])
plt.axhline(y=min_fitness_genetico, color='#fdae61', linestyle=':',
        label='Mínimo Genético: {:d}'.format(int(min_fitness_genetico)))
plt.axhline(y=min_fitness_memetico, color='#98FB98', linestyle=':',
        label='Mínimo Memético: {:d}'.format(int(min_fitness_memetico)))
plt.xlabel('Generación')
plt.ylabel('Distancia Mínima en cada Generación')
plt.title('Progreso de la Función Fitness para los Algoritmos Genético y
        Memético')
plt.legend()
plt.show()
```

Cálculo de Medidas Estadísticas en la Función Fitness

```
# Calcular estadísticas de la Función Fitness de 100 generaciones
ultimas100Gen = mejorSolucionGenetico['Progreso'][-100:]
ultimas100Mem = mejorSolucionMemetico['Progreso'][-100:]
statsGenetico = [np.min(ultimas100Gen), np.max(ultimas100Gen), np.
        mean(ultimas100Gen), np.std(ultimas100Gen)]
statsMemetico = [np.min(ultimas100Mem), np.max(ultimas100Mem), np.
        mean(ultimas100Mem), np.std(ultimas100Mem)]
estadisticasGenetico = [int(i) for i in statsGenetico]
estadisticasMemetico = [int(i) for i in statsMemetico]
data = {'Algoritmo Genético': estadisticasMemetico, 'Algoritmo Memético':
        estadisticasMemetico}
df = pd.DataFrame(data, index=['Mínimo', 'Máximo', 'Promedio', 'Desviación
        estándar'])
print("Análisis Estadístico de la Función Fitness en las últimas 100
        generaciones:")
print(df)
```

Cálculo de la Convergencia Promedio de los Algoritmos Genético y Memético

```
# Función para calcular la convergencia promedio
def calcular_convergencia(progreso):
    diferencias = np.abs(np.diff(progreso))
    convergencia = np.mean(diferencias)
    return convergencia, diferencias
# Calcular convergencia promedio para el A.Genético y A.Memético
convergencia_genetico, diferenciasGenetico = \
    calcular_convergencia(mejorSolucionGenetico['Progreso'])
convergencia_memetico, diferenciasMemetico = \
    calcular_convergencia(mejorSolucionMemetico['Progreso'])
print("Convergencia del Algoritmo Genético:", round(convergencia_genetico,1))
print("Convergencia del Algoritmo Memético:", round(convergencia_memetico,1))
```

Generación de la Gráficas del Mapa de Puntos de las Rutas de Patrullaje halladas por el Algoritmo Genético y el Algoritmo Memético

```
graficarRuta(mejorSolucionGenetico["MejorRuta"])
graficarRuta(mejorSolucionMemetico["MejorRuta"])
```

Anexo 10 Generación del Indicador de Desigualdad en la Distribución Geográfica de delitos (Índice de Gini)

El índice de Gini es una medida ampliamente utilizada para evaluar la desigualdad en la distribución de una variable en una población o área geográfica. En el contexto de la investigación, el índice de Gini se empleó para comprender la disparidad en la distribución de delitos. Para calcular el índice de Gini, se siguieron los siguientes pasos:

1. Dividir el área geográfica en una cuadrícula o celdas para segmentar el espacio.(10x10 celdas)
2. Para cada celda, contar el número de delitos que ocurrieron dentro de esa celda.
3. Ordenar las celdas por el número de delitos en orden ascendente.
4. Calcular la fracción acumulada del total de delitos para cada celda, lo que nos permite entender la contribución relativa de cada celda a la distribución total de delitos.
5. Calcular el área bajo la curva de Lorenz, que representa la acumulación de la distribución de delitos sobre la distribución de celdas ordenadas.
6. Finalmente, calcular el índice de Gini como la relación entre el área entre la línea de igualdad perfecta y la curva de Lorenz, y el área total bajo la línea de igualdad perfecta.

El valor resultante del índice de Gini es 0.51, lo cual indica una desigualdad moderada en la distribución espacial de los delitos, donde ciertas áreas experimentan una concentración desproporcionadamente alta de delitos en comparación con otras.

CÓDIGO DE GENERACIÓN DEL ÍNDICE DE GINI

```
num_celdas = 10
lat_min = delitos['Latitud'].min()
lat_max = delitos['Latitud'].max()
lon_min = delitos['Longitud'].min()
lon_max = delitos['Longitud'].max()
lat_celdas = np.linspace(lat_min, lat_max, num_celdas + 1)
lon_celdas = np.linspace(lon_min, lon_max, num_celdas + 1)
frecuencias, _, _ = np.histogram2d(delitos['Latitud'], delitos['Longitud'],
    ↪bins=[lat_celdas, lon_celdas])
delitosAcumulados = np.cumsum(np.sort(frecuencias.ravel())) / np.
    ↪sum(frecuencias)
x = np.linspace(0.01, 1.00, 100)
y = delitosAcumulados
# Calcular el Área bajo la curva de Lorenz e Índice de Gini
areaBajoCurva = np.trapz(y, x)
areaIgualdadPerfecta = 0.5
indiceGini = (areaIgualdadPerfecta - areaBajoCurva) / areaIgualdadPerfecta
print("Índice de Gini de la distribución espacial de delitos:",
    ↪round(indiceGini,2))
```

Anexo 11 Cuadro Interpretativo de los resultados finales indicados en las tablas

En el siguiente cuadro, se presentan los principales resultados finales mostrados en las tablas presentadas en la presente tesis.

Nro. de Tabla	Hallazgo (s)
Tabla 20	Se mostró las coordenadas geográficas y vías de cada uno de los centroides de las Zonas críticas de Delincuencia.
Tabla 21	Del conjunto de 10 ejecuciones del Algoritmo Genético, en la 9 ^{na} ejecución se obtuvo la menor distancia de patrullaje de 19.535 km.
Tabla 22	Del conjunto de 10 ejecuciones del Algoritmo Memético, en la 1 ^{ra} ejecución se obtuvo la menor distancia de patrullaje de 17.907 km.
Tabla 23	Medidas estadísticas las mejores distancias de patrullaje obtenidas a partir de las últimas 100 generaciones de los Algoritmo Memético y Genético. El algoritmo Memético obtiene el mejor en la media de las distancias, 19643m de media del Algoritmo Genético vs. 17909m de media del Algoritmo Memético.
Tabla 24	Muestra el Camino Mínimo de Patrullaje (Vías) de Patrullaje de Zonas Críticas de Delincuencia de la Ruta Mínima determinada por el Algoritmo Memético.
Tabla 25	Se destacan las agrupaciones y las métricas de densidad delictivas obtenidas por el Algoritmo DBSCAN las cuales son: <ul style="list-style-type: none"> ✓ 97 Zonas Críticas de Densidad Delictiva ✓ Densidad Delictiva: 22.16 delitos/hm² ✓ Densidad Delictiva Relativa: 15.13 veces la Densidad Delictiva de Pueblo Libre. ✓ Proporción Acumulativa de Delincuencia: 72.7% delitos ocurridos en el 4.8% del área de Pueblo Libre.
Tabla 26	Se destacan los siguientes resultados obtenidos por el Algoritmo Memético: <ul style="list-style-type: none"> ✓ Distancia mínima de Patrullaje: 17.907 km ✓ Ruta Mínima de Patrullaje (Grafo): [34, 31, 29, 32, 30, 25, 27, 65, 54, 66, 21, 16, 18, 19, 17, 20, 22, 11, 13, 14, 12, 15, 53, 60, 50, 48, 55, 51, 92, 74, 77, 76, 56, 47, 58, 52, 57, 43, 46, 2, 49, 7, 5, 10, 8, 0, 9, 1, 4, 6, 3, 44, 37, 40, 38, 36, 41, 39, 45, 42, 59, 72, 78, 75, 79, 80, 73, 85, 89, 94, 84, 81, 86, 95, 87, 90, 91, 82, 93, 88, 67, 63, 96, 83, 69, 62, 68, 64, 70, 61, 71, 23, 28, 26, 33, 35, 24]* ✓ Tiempo de ejecución: 169 s
Tabla 27	Se destaca la reducción de la distancia de Patrullaje de Zonas Crítica de Delincuencia obtenida con el Algoritmo Memético: <ul style="list-style-type: none"> ✓ Distancia de Patrullaje (Antes): 20.784 km ✓ Distancia de Patrullaje (Después) obtenido por el A. Memético: 17.907 km ✓ Reducción de la Distancia de Patrullaje: 2.877 km o equivalente al 13.8% de la Distancia de Patrullaje (Antes).