

---

# Modelo de Predicción de Zonas de Riesgo de Accidentes de Tráfico y Congestión Vehicular en la Localidad de Kennedy, con un Enfoque Espacio – Temporal

---

**ESCUELA TECNOLÓGICA INSTITUTO TÉCNICO CENTRAL**

**Facultad de Ingeniería de Sistemas**

**PROYECTO DE INVESTIGACIÓN ELECTIVA IV CIENCIAS DE DATOS**

**Autores:**

Luis Angel Garzón Pérez

Jaicel Arley Soto Álvarez

# Contenido

- I. INTRODUCCIÓN
  - II. LIMPIEZA Y PREPARACIÓN DE DATOS
  - III. BASES DE DATOS
  - IV. RECOLECCIÓN DE INFORMACIÓN
  - V. CARGAR DATOS
  - VI. ANÁLISIS DE ACCIDENTES EN LA CIUDAD: TABLAS Y SERIE TEMPORAL
  - VII. DEFINICIÓN DE POLÍGONOS Y LIMITES ZONALES A TRAVÉS DE COORDENADAS
  - VIII. ZONAS MAPAS DE CALOR
  - IX. IMPLEMENTACIÓN DEL ALGORITMO PROPHET PARA PREDICCIÓN DE DATOS
  - X. EVALUACIÓN DEL MODELO DE PREDICCIÓN CON PROPHET
  - XI. REVISIÓN CALIDAD DE LOS DATOS
  - XII. RESULTADOS DEL MODELO
  - XIII. CONFIGURACIÓN DEL ENTORNO DE DESARROLLO
  - XIV. PROBLEMAS O DIFICULTADES
  - XV. BIBLIOGRAFÍA
  - XVI. CONSULTA
-

## I. INTRODUCCIÓN

El presente proyecto se enfocó en el desarrollo e implementación de un modelo de predicción de zonas de riesgo de accidentes de tráfico y congestión vehicular en la localidad de Kennedy, utilizando un enfoque espacio-temporal. Para lograr este objetivo, se siguió un proceso integral que abarcó desde la identificación de la problemática hasta la implementación y evaluación del modelo predictivo. A lo largo de este proyecto, se mantuvo un enfoque riguroso en mantener el cronograma definido en la propuesta inicial, lo que permitió ejecutar todas las actividades necesarias de manera efectiva y oportuna.

En la fase inicial del proyecto, se llevó a cabo la identificación y recopilación de los datos necesarios para el análisis. Se utilizó un archivo histórico de siniestros en la localidad de Kennedy como fuente principal de información, el cual fue procesado y preparado para su posterior análisis. Este proceso incluyó la limpieza y transformación de los datos utilizando herramientas como Pentaho y Google Colab, lo que permitió obtener un conjunto de datos limpio y listo para su análisis.

Una vez preparados los datos, se procedió a la exploración y análisis de los mismos para identificar patrones y tendencias que pudieran ser relevantes para el modelo predictivo. Se utilizaron técnicas de visualización de datos para comprender mejor la distribución de los accidentes de tráfico y la congestión vehicular en la localidad de Kennedy a lo largo del tiempo.

Con base en el análisis exploratorio de los datos, se procedió a la construcción y ajuste de un modelo de predicción utilizando la biblioteca Prophet de Facebook. Este modelo fue entrenado utilizando datos históricos y luego evaluado utilizando métricas comunes como el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación ( $R^2$ ) para medir su precisión y capacidad predictiva.

Además, se experimentó con diferentes hiperparámetros del modelo Prophet para mejorar su rendimiento y precisión. Se realizaron ajustes en el modelo y se evaluaron los resultados para determinar si las mejoras

realizadas habían resultado en una mejor capacidad predictiva.

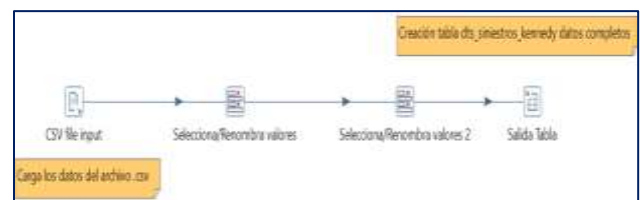
Finalmente, se evaluaron los resultados del modelo mejorado y se compararon con los del modelo original para determinar si las mejoras realizadas habían sido efectivas. Se encontró que, si bien hubo algunas mejoras en la precisión del modelo, aún existen áreas de mejora que podrían explorarse en futuros proyectos. En general, el proyecto logró cumplir con los objetivos establecidos en la propuesta inicial, manteniendo el cronograma definido y ejecutando todas las actividades necesarias para llevar a cabo el proyecto con éxito.

## II. LIMPIEZA Y PREPARACIÓN DE DATOS

La limpieza y preparación de datos es un proceso fundamental en el tratamiento de información antes de realizar análisis o modelado predictivo. En tu caso, el proceso realizado a través de Pentaho y Google Colab implicó varias etapas clave:

### Proceso en Pentaho

*Ilustración 1. Transformación archivo fuente accidentalidad Bogotá*



Fuente: Elaboración propia 2024

- Cargue del archivo fuente: Utilizaste Pentaho para cargar el archivo "historico\_siniestros\_bogota\_d.c\_-.csv", lo cual es un paso común para acceder a los datos.
- Limpieza de datos: Mediante pasos de "Selecciona/Renombra valores", realizaste la limpieza de las características de la información. Este paso es crucial para asegurar que los datos estén en un formato adecuado y consistente.

## Transformación de datos

Ilustración 2. Transformación tabla siniestros Kennedy



Fuente: Elaboración propia 2024

- Consolidación de información: Llevaste a cabo una transformación adicional para consolidar únicamente la información correspondiente a la localidad de Kennedy. Esta es una práctica común para enfocar el análisis en un subconjunto específico de datos.
- Almacenamiento: La tabla resultante de la transformación, "dts\_siniestros\_kennedy", fue almacenada para su posterior uso en el proceso de predicción. El almacenamiento adecuado de los datos preparados es esencial para su acceso y utilización eficiente.

Ilustración 3. Transformación registros accidentes



Fuente: Elaboración propia 2024

## Uso de Google Colab

- Extracción y preparación de datos: Utilizaste Python en Google Colab para la extracción, limpieza y preparación de los datos. Este entorno proporciona herramientas efectivas para trabajar con datos, especialmente en proyectos que requieren análisis y procesamiento complejo.

Ilustración 4. Resultados Consulta Archivo Fuente GoogleColab

```
# Impresión de datos, lectura de archivo, nombre de las columnas y tipo de dato
print(data.info())

# Creación de pandas.core.frame.DataFrame
RangeIndex: 23441 entries, 0 to 23440
Data columns (total 17 columns):
 #   Column                Non-Null Count  Dtype  
```

Fuente: Elaboración propia 2024

## III. BASES DE DATOS

Se implementó una base de datos para la gestión eficiente de la información relacionada con los accidentes de tráfico en la localidad de Kennedy. Esta base de datos cumple una función crucial al almacenar los datos originales provenientes de la fuente, lo que garantiza la integridad y trazabilidad de estos.

Una vez que los datos son sometidos a los procesos de limpieza y transformación necesarios, se conservan en esta base de datos para su posterior análisis y uso en la predicción de accidentes. Este enfoque de almacenamiento de datos asegura que se mantenga un registro detallado y organizado de todas las etapas del proceso, desde la recolección inicial hasta la aplicación de modelos predictivos.

Ilustración 5. Pantallazo Base de Datos

Fuente: Elaboración propia 2024

Además, se creó un archivo que contiene los datos unificados de la localidad de Kennedy, el cual se almacena en la tabla `dts_siniestros_kennedy`. Este archivo consolidado proporciona una visión general de la situación de los accidentes de tráfico en la localidad, lo que facilita el análisis y la identificación de patrones.

La tabla `dts_registros_acci` también desempeña un papel fundamental al almacenar información detallada sobre los registros de accidentes, incluidos los datos necesarios para el proceso de predicción. Esta tabla se convierte en una fuente crucial de información para los modelos predictivos, ya que contiene los datos históricos necesarios para predecir la ocurrencia de accidentes en el futuro.

#### IV. RECOLECCIÓN DE INFORMACIÓN

En este estudio se abarcan los accidentes de tránsito ocurridos en la jurisdicción de la localidad de Kennedy en la ciudad de Bogotá, Colombia, la cual tiene una población de 1.034.293 habitantes (Secretaría General, 2023). En este estudio se consideran todos los tipos de accidentes ocurridos en la localidad mencionada, incluyendo bicicletas y vehículos motorizados, ya que se consideran medios de transporte.

Se utilizó un sistema de información integrado para recopilar datos, fuente que es alimentada por la información de la Secretaría Distrital de Movilidad, el cual se constituye en la base y soporte para la logística de la ciudad. Este sistema facilita la comunicación y el intercambio de información entre los actores (peatones, ciclistas, pasajeros y conductores) y los componentes de la movilidad (entidades del sector de movilidad, infraestructura, vehículos, empresas) cuando resultan daños a personas, vehículos, inmuebles, muebles o animales.

El archivo de datos contiene información georreferenciada de siniestro viales desde 2015 hasta 2022 en la ciudad de Bogotá. Los campos relevantes después del proceso de limpieza de información, para el estudio:

- Fecha de ocurrencia
- Año de ocurrencia

- Dirección
- Gravedad
- Clase de accidente
- Localidad
- Fecha y hora
- Latitud
- Longitud

El valor de las cantidades de accidentes ocurridos es interpretado en conjuntos diarios que se identifican en la fuente de información, constituyendo una serie de valores a lo largo del tiempo. De acuerdo con lo anterior, el registro de los accidentes de tránsito es esencialmente una serie de tiempo, donde el valor numérico de la cantidad de accidentes de tránsito es registrado en intervalos de tiempo no uniformes, sin embargo, pueden ser resumidos y tratados en cualquier unidad temporal.

El registro sistematizado de la ocurrencia de los accidentes de tránsito, y la información disponible a través de los datos abiertos con toda la información almacenada relacionada con los accidentes de tránsito de la ciudad, la cual posteriormente se disminuyó a la localidad de Kennedy. La información recibida contiene un conjunto de datos de accidentes desde el año 2015 hasta el año 2022, y contiene la información de las coordenadas de latitud y longitud del sitio de cada accidente. Para este trabajo, se seleccionó el conjunto de datos de acuerdo con la relevancia.

Se analizaron los datos para identificar patrones y tendencias en la ocurrencia de accidentes de tránsito en la localidad de Kennedy, mediante la creación de un conjunto de datos de exploración para examinar la información en mayor detalle. Luego, se segmentó la información para enfocarse en la localidad específicamente, toda vez que es la finalidad del estudio. Utilizando Python y herramientas de visualización, se crearán gráficos para mostrar la ocurrencia de accidentes en diferentes períodos de tiempo, incluyendo anual, mensual, diario y por horas.

Se incluyen todos los datos de accidentes de tránsito en la localidad de Kennedy para el periodo comprendido entre el 1 de enero de 2015 y el 31 de diciembre de 2021, incluyendo accidentes con heridos o daños materiales en cualquier tipo de vehículo. La información se obtuvo a través de un sistema de información integrado y se procesó para analizar la ocurrencia de accidentes en la localidad.

Dentro del proceso llevado a cabo, se realiza la validación de la cantidad promedio por año, de accidentes de tránsito, con relación a la cantidad de datos con la que se contó como fuente, lo cual permitió establecer un promedio de 3.380, 1 anula.

## V. CARGAR DATOS

Se procede a realizar el cargue inicial de la información, almacenando los datos utilizados para el entrenamiento, evaluación y validación en un directorio de Google Drive del usuario actual de Google Colaboratory. Para obtener un análisis detallado de los datos, se recopila, procesa y agrupa información específica de la localidad de Kennedy. La fuente principal de estos datos es la Secretaría de Movilidad, la cual proporciona un archivo con el histórico de siniestros viales de toda la ciudad de Bogotá.

El proceso comienza con la filtración de la información, seleccionando únicamente los datos relevantes a la localidad de Kennedy para construir un conjunto de datos exploratorio. A continuación, se generan diversas gráficas utilizando Python y la biblioteca matplotlib.pyplot, las cuales muestran la frecuencia de accidentes tanto en general como por zonas específicas de la localidad, a lo largo de diferentes periodos de tiempo (anual, mensual, diario y por hora).

Para acceder a los archivos y carpetas de Google Drive desde el entorno de desarrollo interactivo Google Colab, se realiza la conexión al directorio correspondiente. Luego, se lee el archivo de datos desde Google Drive y se carga en un DataFrame utilizando la biblioteca Pandas. El archivo se encuentra en la ruta especificada del directorio de Google Drive y el separador de columnas en el archivo es el carácter. El resultado es un DataFrame almacenado en la variable data.

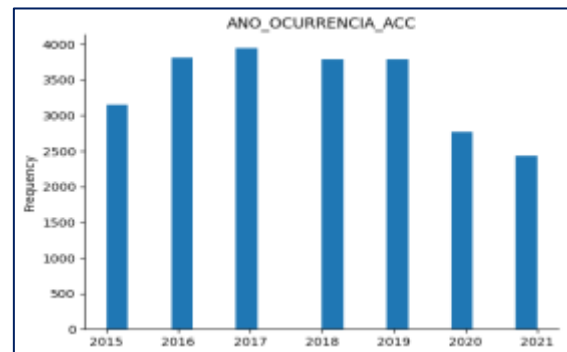
Este enfoque proporciona una visualización clara de la distribución y acumulación de accidentes de tránsito a lo largo de los años, facilitando una comprensión más profunda de las tendencias y patrones en la localidad de Kennedy. Las gráficas generadas permiten interpretar de manera detallada los datos y ofrecen una base sólida para la toma de decisiones informadas en materia de movilidad y seguridad vial.

### Gráfico de Histograma

Mediante un gráfico de histograma utilizando la biblioteca Matplotlib, se muestra la distribución inicial de los valores en la columna

ANO\_OCURRENCIA\_ACC del DataFrame data. El gráfico tiene 20 intervalos (bins) y muestra el año que indica el periodo de ocurrencia de accidentes. Además, se eliminan los ejes superiores y derechos del gráfico para darle un aspecto más limpio.

*Ilustración 6. Cantidad Registro Año Ocurrencia*

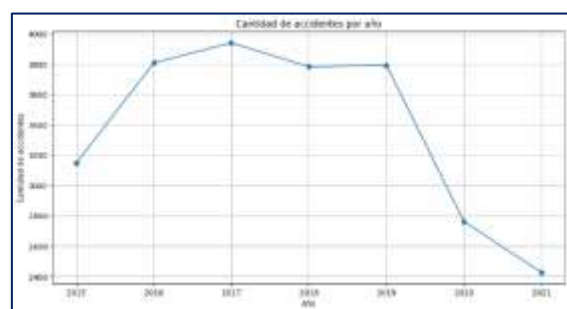


Fuente: Elaboración propia 2024

### Gráfico de Línea de Acumulación

A través de un gráfico de línea, se muestra la acumulación de datos agrupados por año de ocurrencia. El gráfico muestra la cantidad de accidentes que ocurrieron en cada año, lo que permite una interpretación más detallada de los datos y la identificación de tendencias y patrones en la ocurrencia de accidentes a lo largo del tiempo.

*Ilustración 7. Comportamiento Cantidad de Accidentes por Año*



Fuente: Elaboración propia 2024

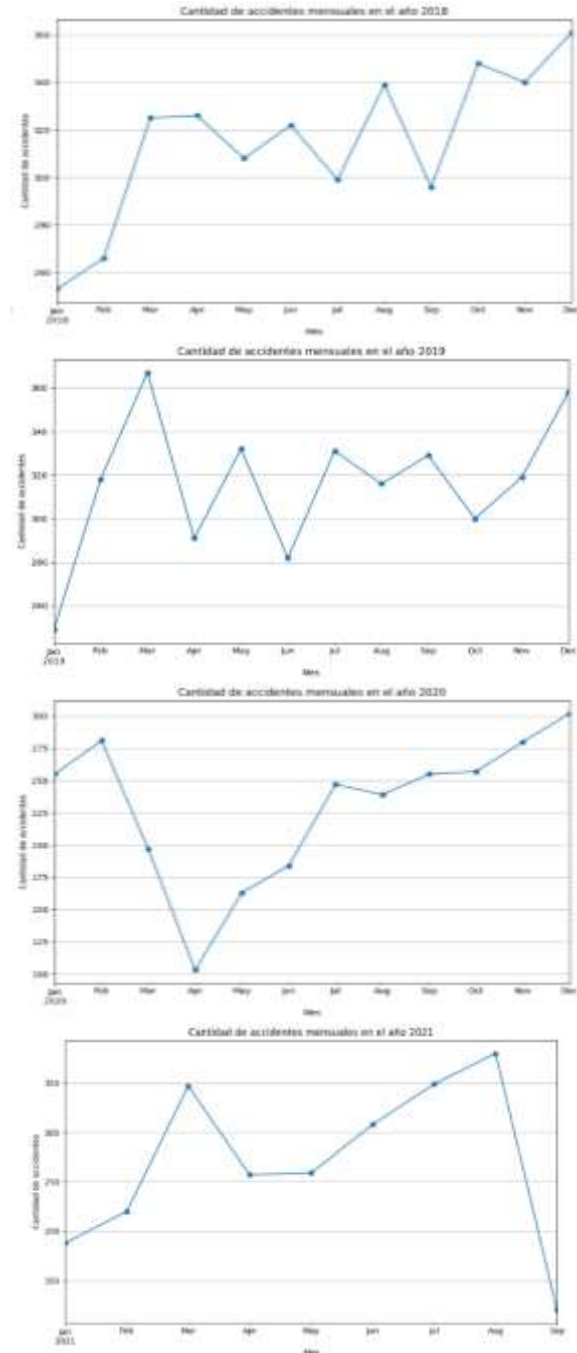
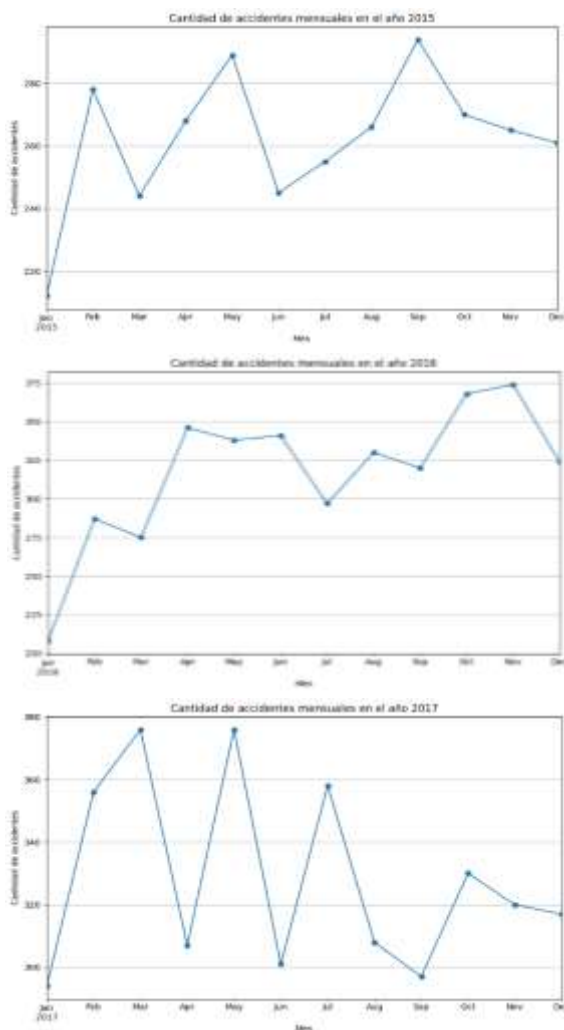
### Análisis de la frecuencia de accidentes mensuales

Por otra parte, se observa que, de manera general, la ocurrencia de accidentes de tránsito aumenta durante los meses de temporada alta, presentando valores más constantes hacia finales del año 2020. En los meses de

enero a marzo, se registra una menor cantidad de accidentes, lo cual podría ser consecuencia de diversas medidas implementadas por la administración local, tales como la construcción de diferentes obras de infraestructura en toda la ciudad. Sin embargo, hacia finales del año 2021, la tendencia muestra una disminución en los últimos meses.

Se evidencian fluctuaciones a lo largo del año, con picos y valles en diferentes meses, lo que indica que los valores de accidentes no son constantes. Esta variabilidad sugiere la necesidad de un análisis más detallado para identificar los factores específicos que influyen en estos cambios, permitiendo así la implementación de estrategias más efectivas para reducir la incidencia de accidentes de tránsito en la localidad de Kennedy.

*Ilustración 8. Comportamiento Cantidad de Accidentes Mes-Año*



Fuente: Elaboración propia 2024

### Análisis de la frecuencia de accidentes por día de la semana.

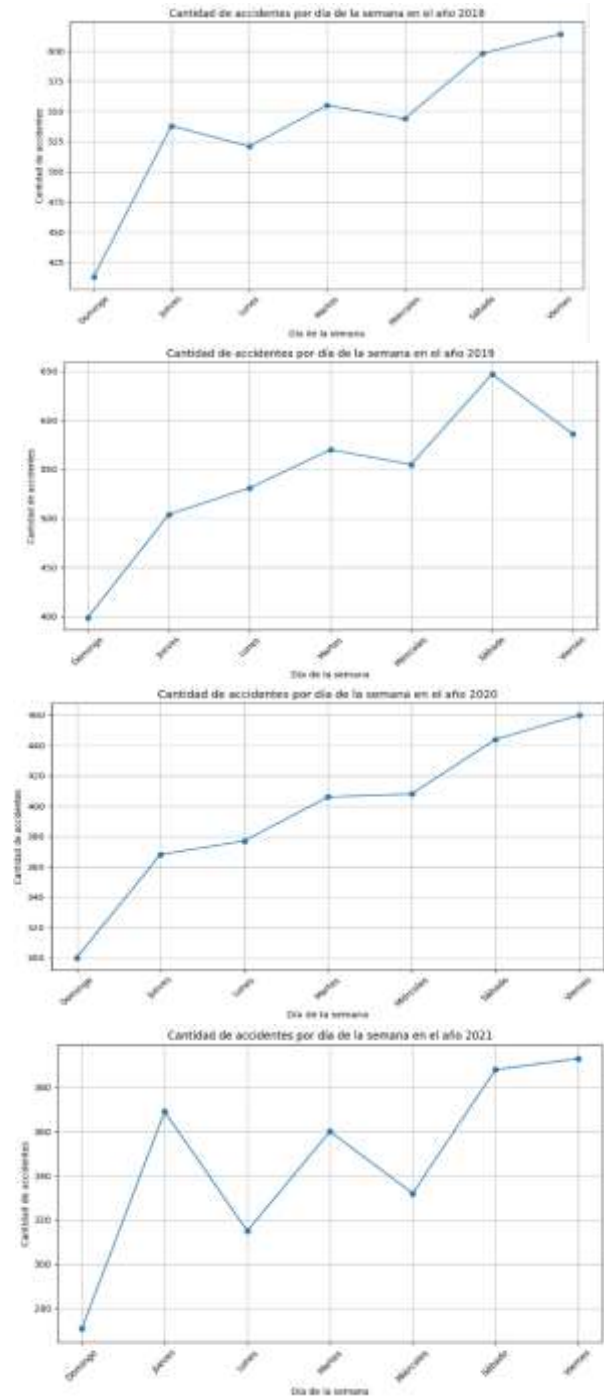
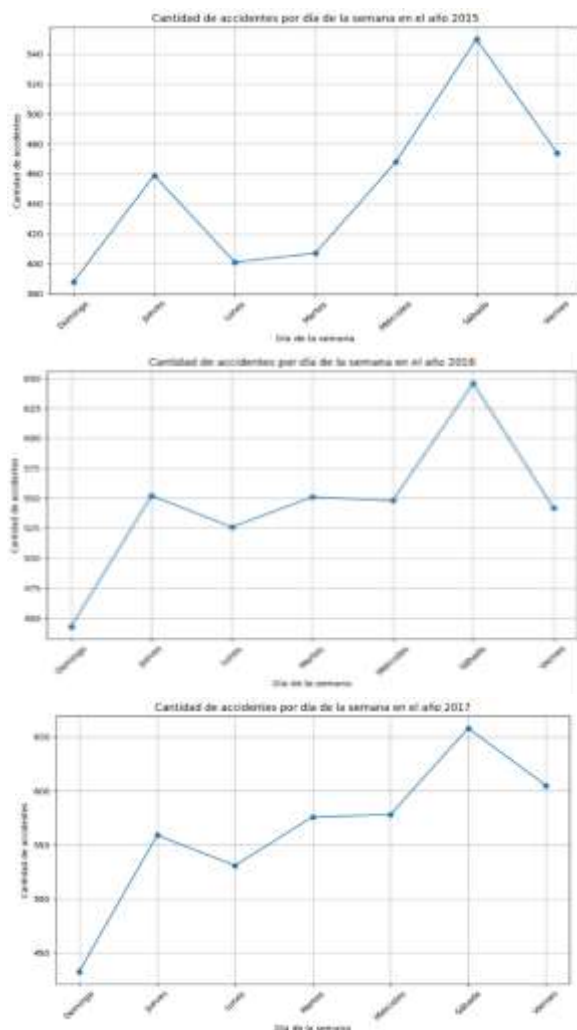
De la misma forma, se realiza un análisis de la frecuencia de accidentes por día de la semana. Primero, se convierte la columna de fechas al tipo datetime y se crea una nueva columna que contiene el día de la semana (0 = lunes, 6 = domingo) para cada fecha. Luego, se mapean los números a los nombres correspondientes de los días de la semana para facilitar la lectura.



Posteriormente, se agrupan los datos por día de la semana y se cuenta la cantidad de accidentes ocurridos en cada día. Finalmente, se crea un gráfico de líneas que muestra la cantidad de accidentes por día de la semana. El gráfico incluye marcadores de puntos para cada día y su título es "Cantidad de accidentes por día de la semana". Además, se etiquetan adecuadamente los ejes, con el eje x representando el día de la semana y el eje y indicando la cantidad de accidentes.

Se observa en la gráfica que los domingos presentan menos accidentes, mientras que los sábados tienen la mayor cantidad de incidentes, lo cual podría deberse al mayor volumen de tráfico vehicular en este día. De lunes a jueves, se nota una variabilidad constante en la cantidad de accidentes, sugiriendo que los fines de semana son los periodos con mayor afectación en términos de accidentes de tránsito.

*Ilustración 9. Comportamiento Cantidad de Accidentes por Día de la Semana - Año*



Fuente: Elaboración propia 2024

### Análisis de Accidentes por Hora del Día y Año

A través de este análisis, se pretendió identificar patrones y tendencias en la frecuencia de accidentes en función de la hora del día y del año. Para lograrlo, se utilizó un conjunto de datos que incluye la fecha y hora de ocurrencia de cada accidente. La columna de fechas se convirtió al formato datetime, y se crearon dos nuevas columnas que asignan el año y la hora del día correspondientes a cada accidente.

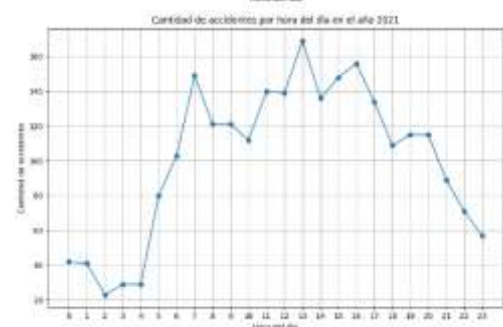
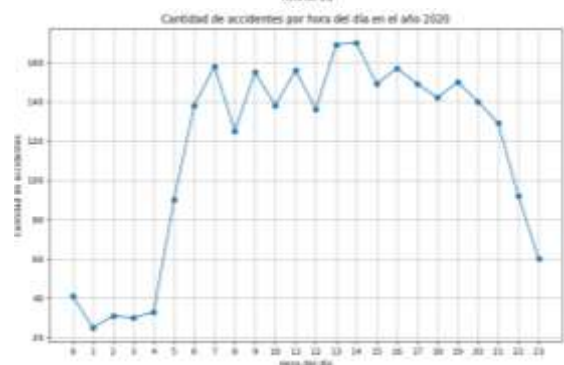
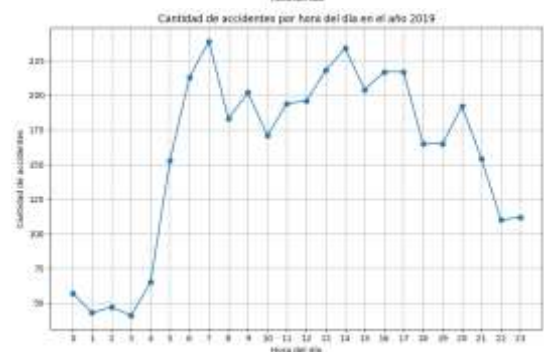
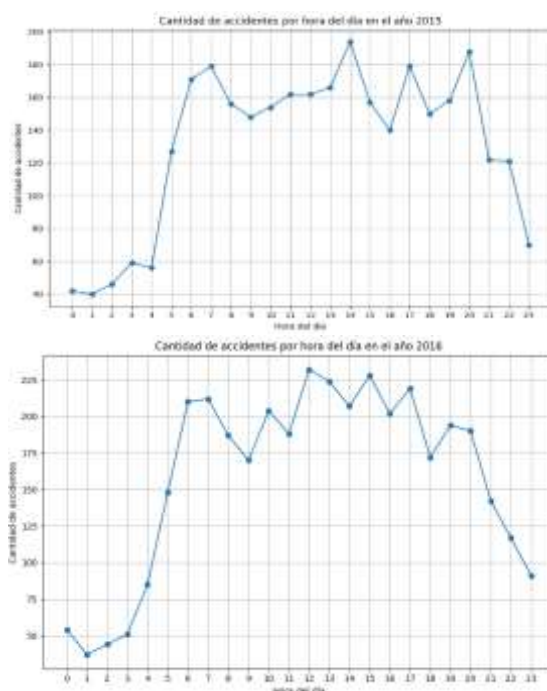


Los resultados muestran que la mayoría de los accidentes ocurren durante el día, entre las 7:00 y las 20:00, con un pico de accidentes a las 10:00 y a las 16:00. Por otro lado, las horas con menor cantidad de accidentes son las 00:00 y las 6:00. Durante la noche, entre las 21:00 y las 7:00, se producen menos accidentes en general, con un pico a las 14:00.

Además, se observó un aumento de accidentes a las 7:00 de la mañana, lo cual puede deberse a la congestión en las vías y al estrés causado por el tráfico y la prisa para llegar a destino. La distribución de accidentes a lo largo de los años es relativamente estable, lo que sugiere patrones de comportamiento constantes a lo largo del tiempo, como los picos de tráfico durante las horas pico de trabajo o las congestiones en las principales vías de la ciudad.

Para identificar zonas específicas de la ocurrencia de accidentes dentro de la localidad de Kennedy, se exploraron diferentes enfoques experimentales. Se establecieron polígonos de diferentes tamaños, pero se encontró que no había suficientes datos para crear una serie de tiempo en un espacio tan reducido. Por lo tanto, se crearon 9 zonas en las que se clasificaron nuevamente los accidentes, permitiendo un análisis más detallado y preciso de las áreas con mayor incidencia de accidentes.

*Ilustración 10. Comportamiento Cantidad de Accidentes por Hora de Día - Año*



Fuente: Elaboración propia 2024

## VI. ANÁLISIS DE ACCIDENTES EN LA CIUDAD: TABLAS Y SERIE TEMPORAL

A continuación, se presenta el análisis de los accidentes de tránsito en la ciudad desde el 1 de enero de 2015 hasta el 31 de diciembre de 2021. Para este propósito, se creó un DataFrame con fechas diarias, inicialmente llenas con ceros. Se añadieron nuevas columnas para identificar si el día corresponde a un fin de semana o entre semana, así como columnas que indican el año, el mes y el día de la semana.

Posteriormente, los datos se agruparon por semana, contando la cantidad de accidentes en cada categoría. También se crearon tablas quincenales y mensuales para analizar la tendencia de los accidentes en diferentes periodos de tiempo. Además, se generó una serie temporal de accidentes totales en toda la ciudad.

Finalmente, se imprimieron los resultados de las tablas diaria, semanal, quincenal y mensual, así como la serie temporal total de accidentes, proporcionando una visión integral de las tendencias y patrones de accidentes en la ciudad durante el periodo de estudio.

### Tabla Diaria:

Esta tabla muestra la cantidad de accidentes de tránsito registrados en la ciudad de manera diaria. Cada fila representa un día específico dentro del periodo de estudio, y la columna correspondiente indica el número de accidentes que ocurrieron ese día. Esta tabla permite identificar patrones diarios y variaciones en la frecuencia de accidentes, proporcionando una visión detallada de los días con mayor y menor incidencia de siniestros.

FECHA	ACCIDENTES	FIN_DE_SEMANA
ENTRE_SEMANA	AÑO	MES
DÍA_SEMANA		
0 2015-01-01	0	False
True 2015	1	3
1 2015-01-02	0	False
True 2015	1	4
2 2015-01-03	0	True
False 2015	1	5
3 2015-01-04	0	True
False 2015	1	6
4 2015-01-05	0	False
True 2015	1	0

Ilustración 11. Resultados Cantidad de Accidentes Diarios

FECHA	ACCIDENTES	FIN_DE_SEMANA	ENTRE_SEMANA	AÑO	MES	DÍA_SEMANA
0 2015-01-01	0	False	True	2015	1	3
1 2015-01-02	0	False	True	2015	1	4
2 2015-01-03	0	True	False	2015	1	5
3 2015-01-04	0	True	False	2015	1	6
4 2015-01-05	0	False	True	2015	1	0
...	...	...	...	...	...	...
2552 2021-12-27	0	False	True	2021	12	0
2553 2021-12-28	0	False	True	2021	12	1
2554 2021-12-29	0	False	True	2021	12	2
2555 2021-12-30	0	False	True	2021	12	3
2556 2021-12-31	0	False	True	2021	12	4

2557 rows × 7 columns

Fuente: Elaboración propia 2024

### Tabla Semanal:

Esta tabla agrupa los datos por semana, mostrando la cantidad total de accidentes que ocurrieron en cada semana durante el periodo de estudio. Cada fila representa una semana específica, con la cantidad de accidentes ocurridos en ese intervalo. Este agrupamiento semanal facilita el análisis de tendencias y patrones que se desarrollan en periodos de siete días, ayudando a identificar semanas particularmente críticas en términos de accidentes.

FECHA	ACCIDENTES
FIN_DE_SEMANA	ENTRE_SEMANA
AÑO	
MES	DÍA_SEMANA
0 2015-01-05	0
2	3 2015 1
3	
1 2015-01-12	0
2	5 2015 1
1	
2 2015-01-19	0
2	5 2015 1
1	
3 2015-01-26	0
2	5 2015 1
1	
4 2015-02-02	0
2	5 2015 1
1	

366 rows × 7 columns

*Ilustración 12. Resultados Cantidad de Accidentes Semanales*

	FECHA	ACCIDENTES	FIN DE SEMANA	ENTRE SEMANA	AÑO	MES	DÍA SEMANA
0	2015-01-05	0	2	3	2015	1	2
1	2015-01-12	0	2	3	2015	1	3
2	2015-01-19	0	2	3	2015	1	4
3	2015-01-26	0	2	3	2015	1	5
4	2015-02-02	0	2	3	2015	1	6
...	...	...	...	...	...	...	...
361	2021-12-06	0	2	3	2021	11	1
362	2021-12-13	0	2	3	2021	12	2
363	2021-12-20	0	2	3	2021	12	3
364	2021-12-27	0	2	3	2021	12	4
365	2022-01-03	0	0	4	2021	12	5

Fuente: Elaboración propia 2024

### Tabla Quincenal:

En esta tabla, los datos están agrupados en intervalos de quince días, mostrando la cantidad total de accidentes para cada quincena. Cada fila corresponde a un periodo de dos semanas, y la columna muestra el número de accidentes que ocurrieron durante ese tiempo. Este formato de agrupamiento ayuda a observar tendencias y fluctuaciones en periodos quincenales, ofreciendo una perspectiva intermedia entre los análisis semanal y mensual.

	FECHA	ACCIDENTES	AÑO	MES
0	2015-01-05	0	2015	1
1	2015-01-19	0	2015	1
2	2015-02-02	0	2015	1
3	2015-02-16	0	2015	2
4	2015-03-02	0	2015	2

184 rows × 4 columns

*Ilustración 13. Resultados Cantidad de Accidentes Quincenales*

	FECHA	ACCIDENTES	AÑO	MES
0	2015-01-05	0	2015	1
1	2015-01-19	0	2015	1
2	2015-02-02	0	2015	1
3	2015-02-16	0	2015	2
4	2015-03-02	0	2015	2
...	...	...	...	...
179	2021-11-15	0	2021	11
180	2021-11-29	0	2021	11
181	2021-12-13	0	2021	11
182	2021-12-27	0	2021	12
183	2022-01-10	0	2021	12

Fuente: Elaboración propia 2024

### Tabla Mensual:

Esta tabla presenta la cantidad de accidentes agrupados por mes. Cada fila representa un mes dentro del periodo de estudio, con la cantidad total de accidentes registrado

s en ese mes. Este agrupamiento mensual permite un análisis a mayor escala, identificando tendencias estacionales y mensuales, y observando cómo varía la frecuencia de accidentes a lo largo de los meses del año.

	FECHA	ACCIDENTES	AÑO	MES
0	2015-01-31	0	2015	1
1	2015-02-28	0	2015	2
2	2015-03-31	0	2015	3
3	2015-04-30	0	2015	4
4	2015-05-31	0	2015	5

84 rows × 4 columns

*Ilustración 14. Resultados Cantidad de Accidentes Mensuales*

	FECHA	ACCIDENTES	AÑO	MES
0	2015-01-31	0	2015	1
1	2015-02-28	0	2015	2
2	2015-03-31	0	2015	3
3	2015-04-30	0	2015	4
4	2015-05-31	0	2015	5
...	...	...	...	...
79	2021-08-31	0	2021	8
80	2021-09-30	0	2021	9
81	2021-10-31	0	2021	10
82	2021-11-30	0	2021	11
83	2021-12-31	0	2021	12

Fuente: Elaboración propia 2024

### Serie Temporal Total:

La serie temporal total muestra una visión general de la cantidad de accidentes acumulados a lo largo del tiempo. Cada punto en la serie representa el total acumulado de accidentes hasta una fecha determinada, permitiendo identificar la evolución de los accidentes de tránsito en la ciudad durante el periodo de estudio. Esta serie es útil para observar tendencias generales y cambios en la frecuencia de accidentes a lo largo de varios años.

FECHA	ACCIDENTES
2015-01-01	0
2015-01-02	0
2015-01-03	0
2015-01-04	0
2015-01-05	0

Name: ACCIDENTES, dtype: int64

## VII. DEFINICIÓN DE POLÍGONOS Y LÍMITES ZONALES A TRAVÉS DE COORDENADAS

En la segunda fase del análisis, se enriqueció el conjunto de datos con variables exógenas relevantes que podrían influir en la ocurrencia de accidentes. Estas variables incluyen días festivos, eventos que generan alteración del orden público, día de la semana y zona geográfica donde ocurrió el accidente. La inclusión de estas variables exógenas permitió capturar patrones y tendencias adicionales que podrían mejorar la precisión y utilidad de las predicciones del modelo.

Para identificar zonas geográficas, se crearon polígonos utilizando el sistema de coordenadas decimales. Se utilizó la herramienta gratuita Google My Maps para crear las capas y extraer las coordenadas correspondientes a cada polígono. Luego, se cargó la información de cada polígono en el entorno de programación Python utilizando la librería `shapely.geometry`. Se utilizó el algoritmo de Ray Casting para identificar a qué zona geográfica pertenecía cada accidente registrado en el conjunto de datos.

*Ilustración 15. Mapa Georreferenciación Por Zona - Polígonos*



Fuente: Elaboración propia 2024

Una vez identificadas y clasificadas todas las incidencias de accidentes en función de las zonas geográficas establecidas, se cargó la información en una base de datos para su posterior análisis. Es importante destacar que solo se utilizaron accidentes ocurridos a partir del 1 de enero de 2015, ya que a partir de esa fecha se tiene la información de las coordenadas del accidente y los polígonos fueron adaptados varias veces para asegurarse de que ningún accidente quedara fuera del área de los polígonos.

Este enfoque permitió realizar un estudio detallado de las características y distribución de los accidentes en función de las diferentes zonas geográficas, proporcionando información valiosa para la implementación de los algoritmos propuestos. La metodología de preparación de datos permite generar una estructura adecuada y la inclusión de variables relevantes disponibles para desarrollar un modelo predictivo en el ámbito de la ingeniería computacional aplicada a la seguridad vial.

Se llevaron a cabo múltiples experimentos con diferentes tipos de modelos, pero se descartaron debido a su bajo desempeño. Se transformó el dataset de diferentes maneras, creando modelos globales para cada día y hora del año, agrupando los datos a nivel de día y hora. También se desarrolló un modelo que consideró características como la zona de ocurrencia del accidente, el día de la semana, el mes y el año. Sin embargo, los resultados para este tipo de modelo no fueron concluyentes.

La información disponible en la base de datos aportada por la secretaría de movilidad de la ciudad de Bogotá incluye la descripción del tipo de accidente presentado, la característica de la vía, los vehículos involucrados y las causas del accidente. Sin embargo, esta información no permite obtener variables que se puedan asociar a una ocurrencia futura, por lo que solo se consideró la información de la fecha, hora, cantidad y ubicación georreferenciada de los accidentes de tránsito de la ciudad.

El código define ocho polígonos en coordenadas geográficas utilizando la librería `Polygon` de `shapely.geometry` en Python. Cada polígono está especificado por una secuencia de puntos, donde cada punto representa un par de coordenadas de longitud y latitud. Estos polígonos son útiles para representar áreas geográficas específicas en un mapa.

Además, el código emplea la librería `pandas` para la manipulación de datos en forma de `DataFrame`. Se crea una columna adicional en el `DataFrame` que indica la cantidad de registros evidenciados en cada polígono, reflejando la cantidad de accidentes en esas áreas. Esta integración de `shapely` y `pandas` facilita el análisis y visualización de datos geoespaciales, permitiendo una comprensión más profunda de la distribución y

frecuencia de accidentes en diferentes zonas geográficas.

La creación de zonas georreferenciadas en Google My Maps ha permitido visualizar áreas urbanas y periféricas en la ciudad para analizar y predecir la ocurrencia de accidentes de tráfico. Estas zonas consideran factores como la topografía, densidad poblacional e infraestructura vial para identificar patrones y tendencias en la distribución de accidentes.

Se observa una relación inversa entre la distribución de accidentes y el tamaño de las zonas, sugiriendo que la densidad poblacional y complejidad vial influyen en la probabilidad de accidentes. La mayor densidad poblacional en áreas urbanas y periféricas puede aumentar los desplazamientos y, por ende, la probabilidad de accidentes.

Inicialmente, se crearon zonas con alta resolución espacial para analizar la ocurrencia de accidentes en períodos cortos. Sin embargo, la baja frecuencia de accidentes en muchas zonas dificultó la capacidad predictiva de los modelos. Por ello, se diseñaron áreas con menor resolución espacial y se emplearon gráficos de calor para identificar puntos críticos de accidentalidad en cada zona.

Aunque la predicción se realiza generalmente para una zona específica, también es posible visualizar los puntos con mayor probabilidad de accidentes dentro de esa área. Esto permite a las autoridades de tránsito concentrar sus esfuerzos en zonas específicas, lo que puede mejorar la eficacia en la prevención de accidentes.

A continuación, se presenta la siguiente tabla que muestra la cantidad de accidentes ocurridos en cada zona entre el año 2015 y 2021:

*Tabla 1. Cantidad de Accidentes por Zona -Año*

AÑO	2015	2016	2017	2018	2019	2020	2021
Zona 1	209	263	269	324	315	188	187
Zona 2	435	591	628	652	679	506	463
Zona 3	665	647	707	693	580	403	386
Zona 4	23	26	31	21	32	17	15
Zona 5	578	690	698	636	702	525	453
Zona 6	1	2	0	1	3	1	0
Zona 7	155	195	202	183	203	152	110
Zona 8	115	159	155	134	145	111	88

Fuente: Elaboración propia 2024

Los datos representan la cantidad de accidentes de tránsito registrados en las diferentes zonas durante los años 2015 a 2021. Cada fila de la tabla corresponde a una zona específica, y cada columna corresponde a un año específico. Por ejemplo, en la fila "Zona 1" y la columna "2015", el valor de 209 indica que en el año 2015 se registraron 209 accidentes de tránsito en la Zona 1.

La tabla muestra cómo la cantidad de accidentes de tránsito ha variado a lo largo de los años en cada una de las zonas. Por ejemplo, se puede observar que en la Zona 3 hubo un pico en la cantidad de accidentes de tránsito en 2017, seguido de una disminución en los años siguientes. Por otro lado, en la Zona 5, la cantidad de accidentes de tránsito ha sido más variable, con aumentos y disminuciones en diferentes años.

## VIII. ZONAS MAPAS DE CALOR

Si bien se llevó a cabo la investigación y el establecimiento de las condiciones necesarias para el manejo de la fuente, con el fin de poder realizar una interpretación precisa de los datos, se definieron los polígonos con el resultado de la cantidad de accidentes en cada una de las zonas. De la misma manera, se realizó un establecimiento de los datos por cada una de las zonas, lo que permite observar cuáles son los lugares en el mapa con mayor afectación, identificados con mapas de calor en dichos puntos.

Se observa entonces que la Zona 1 se encuentra comprendida dentro de los siguientes límites: al norte con el Río Fucha, al sur con la Av. Calle 6, al este con la Av. Carrera 86 y al oeste con el Río Bogotá. En esta zona se encuentran los barrios El Tintal III, El Tintal IV, La Magdalena, Osorio II, Tintalá, Vereda El Tintal Rural y Vereda El Tintal Urbano. Se observa que el área de mayor accidentalidad se da principalmente en la Av. Ciudad de Cali, una calle bastante concurrida por vehículos, siendo una de las vías principales de la ciudad.



Ilustración 16. Mapa de Calor Zona 1



Fuente: Elaboración propia 2024

En la Zona 2, que se encuentra comprendida entre los límites: al norte con la Calle 16C, al sur con la Calle 3 y 26 Sur, al este con la Av. Carrera 72 y al oeste con la Av. Carrera 86. En esta zona se encuentran los barrios Bavaria, Castilla, Ciudad Kennedy, Ciudad Techo II, El Vergel Oriental, La Pampa, Las Dos Avenidas, Mandalay, Nuevo Techo, Pío XII, Valladolid, Vergel Occidental, Villa Alsacia II y Visión de Oriente. Se observa que los puntos relacionados con los accidentes son dispersos, en diferentes áreas de toda la zona, incluyendo algunos puntos de la Avenida Boyacá como vía principal, y en puntos de la Avenida Cali, siendo estas vías principales de tránsito vehicular.

Ilustración 17. Mapa de Calor Zona 2



Fuente: Elaboración propia 2024

En la Zona 3, que se encuentra comprendida entre los límites: al norte con la Avenida Calle 13, al sur con la Avenida Calle 1 de Mayo, al este con la Av. Carrera 68 y al oeste con la Av. Carrera 72. En esta zona se encuentran los barrios Cooperativa de Sub Oficiales, Hipotecho, Hipotecho Occidental, Hipotecho Sur, Lusitania, Marsella, Provienda Oriental y Villa Alsacia. Se observa que los puntos relacionados con los accidentes son dispersos, pero se presentan puntos rojos en la Avenida Boyacá, principal vía de esta zona, y en

sectores limítrofes con la Calle 13 o Calle 30. Como se refleja, siguen siendo las vías principales las que presentan la mayor cantidad de accidentes.

Ilustración 18. Mapa de Calor Zona 3



Fuente: Elaboración propia 2024

En la Zona 4, que se encuentra comprendida entre los límites: al norte con la Avenida Calle 1 de Mayo, al sur con la Autopista Sur y el Río Tunjuelito, al este con la Av. Carrera 68 y al oeste con la Calle 44 Sur. En esta zona se encuentran los barrios Alquilería La Fragua, Alquilería La Fragua Norte, Boita, La Campiña, Las Delicias, Nueva York, Provienda, Provienda Occidental, Renania Urapanes, Santa Catalina, Timiza, Timiza A, Timiza B, Timiza C y Tundama. Se observa entonces que la ubicación de los accidentes no se presenta por todo el sector de manera dispersa; sin embargo, en el límite con la Av. Villavicencio es donde se muestran los puntos en rojo, indicando una mayor concentración de accidentes en esta vía principal.

Ilustración 19. Mapa de Calor Zona 4

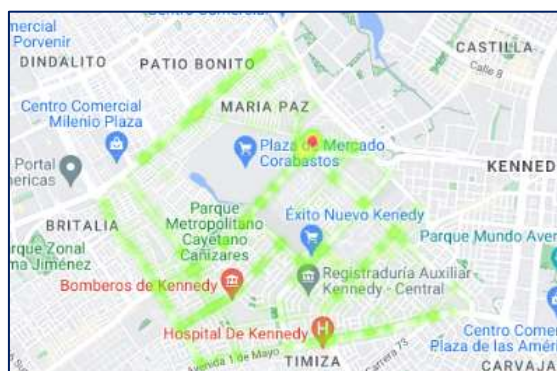


Fuente: Elaboración propia 2024



Para el caso de la Zona 5, que se encuentra comprendida entre los límites: al norte con la Calle 13, al sur con la Av. Villavicencio, al este con la Avenida 1 de Mayo y al oeste con la Avenida Calle 86. Está conformada por los barrios Chucua de La Vaca I, Chucua de La Vaca II, Chucua de La Vaca III, Ciudad Kennedy Central, Ciudad Kennedy Norte, Ciudad Kennedy Occidental, Ciudad Kennedy Oriental, Ciudad Kennedy Sur, Corabastos, Llano Grande, María Paz, Saucedal, Techo y Villa Nelly III S. Es uno de los sectores con mayor afluencia, ya que se encuentran varias vías principales de la ciudad. Pese a ello, se muestra que el sector con mayor accidentalidad es cerca de la plaza de Corabastos, lugar donde se encuentra mayor movimiento económico y, por ende, es transitado a diario, siendo este el punto de mayor accidentalidad.

*Ilustración 20. Mapa de Calor Zona 5*



Fuente: Elaboración propia 2024

La Zona 6 se encuentra delimitada al norte por la Avenida Villavicencio, al sur por el Río Tunjuelito, al este por el mismo río y al oeste por la Avenida Calle 86 y la Carrera 82B. Los barrios que la componen incluyen Casa Blanca Sur, Casablanca, Catalina, Catalina II, Class, El Carmelo, El Rubí, Gran Britalia, Gran Britalia I, Jacqueline, Jorge Uribe Botero, La Cecilia, Pastrana, Roma y Tocarema. A pesar de contar con vías principales como la Avenida Ciudad de Cali o la Avenida Villavicencio, donde en el límite se observan los puntos con mayor accidentalidad, según la fuente, esta zona es la menos afectada en términos de incidencia de accidentes. Aunque es una zona con varias vías principales, se destaca por su menor incidencia de accidentes en comparación con otras áreas de la ciudad.

*Ilustración 21. Mapa de Calor Zona 6*



Fuente: Elaboración propia 2024

La Zona 7 se encuentra delimitada al norte por la Avenida Calle 6, al sur con la Calle 40 Sur, al este por la Avenida Carrera 86 y al oeste por el Río Bogotá. Los barrios que la componen incluyen Calandaima, Campo Hermoso, El Paraíso, El Paraíso Bosa, Galán, Galán Rural, Osorio III, Patio Bonito, Patio Bonito II, Patio Bonito III y Tairona. Es uno de los sectores de la localidad que tiene mayor afluencia vehicular, principalmente debido a su carácter comercial y a su alta densidad poblacional. Aunque las vías principales se presentan únicamente sobre la Avenida Ciudad de Cali, varias áreas del sector presentan numerosos incidentes, lo que lo convierte en uno de los sectores más afectados en términos de accidentes de tránsito.

*Ilustración 22. Mapa de Calor Zona 7*



Fuente: Elaboración propia 2024

La Zona 8 está delimitada al norte por la Calle 40 Sur, al sur con la Calle 49 Sur, al este por la Avenida Carrera 86 y al oeste por el Río Bogotá. Los barrios que la componen incluyen Ciudad de Cali, Dindalito, El Jazmín, Las Acacias, Las Margaritas, Los Almendros, Osorio XII, Provivienda Occidental y Tintalito. Al igual

que otras zonas, presenta dispersión en los datos con respecto a su localización, siendo las avenidas principales como la Avenida Tintal las áreas con mayor incidencia de accidentes. Aunque en todo el sector se visualiza que son varios sectores afectados en este sentido, lo que sugiere una problemática generalizada en cuanto a la seguridad vial en la zona.

Ilustración 23. Mapa de Calor Zona 4



Fuente: Elaboración propia 2024

## IX. IMPLEMENTACIÓN DEL ALGORITMO PROPHET PARA PREDICCIÓN DE DATOS

Prophet es un innovador modelo de predicción de series temporales que emplea un enfoque aditivo para abordar los desafíos inherentes a este tipo de datos. Su estructura se compone de tres elementos fundamentales: una tendencia general, componentes de estacionalidad y efectos de días festivos.

La tendencia general tiene la capacidad de capturar patrones de crecimiento no lineales y cambios en las tendencias, lo que facilita la modelización de patrones complejos presentes en las series temporales. Por otro lado, los componentes de estacionalidad permiten modelar patrones que se repiten en el tiempo, como ciclos diarios, semanales o anuales. Además, los efectos de días festivos son esenciales para incorporar eventos especiales que afectan las observaciones en fechas específicas.

Prophet es un modelo de código abierto desarrollado por el equipo de Core Data Science de Facebook, lanzado en 2017 y disponible en PyPI. Su instalación se inicia con la carga de la herramienta según las especificaciones en la página principal de Prophet, lo que permite comenzar a trabajar con ella de inmediato.

Este modelo es especialmente útil para predecir series temporales que presentan fuertes efectos estacionales y una gran cantidad de datos históricos. Además, es resistente a datos faltantes y cambios en la tendencia, y puede manejar valores atípicos con facilidad. Prophet fue diseñado específicamente para el ámbito empresarial, donde la precisión en la predicción es fundamental.

El modelo aditivo de Prophet consta de cuatro componentes clave:

$$y(t)=g(t)+s(t)+h(t)+\epsilon t$$

donde,

- La tendencia ( $g(t)$ ), que captura la dirección general de la serie temporal.
- La estacionalidad ( $s(t)$ ), que modela las fluctuaciones periódicas que pueden afectar a la serie temporal.
- Los efectos de las vacaciones ( $h(t)$ ), que toman en cuenta los días festivos y otros eventos recurrentes que pueden influir en la serie temporal.
- El término de error ( $\epsilon t$ ), que representa las fluctuaciones aleatorias que el modelo no puede explicar.

Los hiperparámetros de Prophet pueden ser personalizados para adaptarse a las necesidades específicas de cada serie temporal. Algunos de los parámetros más importantes son:

- Puntos de cambio: que definen los cambios en la tendencia de la serie temporal.
- Estacionalidad: que define las funciones periódicas que pueden afectar a la serie temporal.
- Días festivos: que modelan los días especiales y otros eventos recurrentes que pueden influir en la serie temporal.

Implementamos Prophet utilizando la biblioteca correspondiente. Primero, realizamos la instalación de Prophet utilizando pip: `pip install prophet`.

Prophet sigue el sklearnmodelo API. Creamos una instancia de la Prophetclase y luego llamamos a sus métodos fit y predict.

La entrada a Prophet es siempre un marco de datos con dos columnas: ds y y. La ds columna (marca de fecha) debe tener el formato esperado por Pandas, idealmente AAAA-MM-DD para una fecha o AAAA-MM-DD HH:MM:SS para una marca de tiempo. La y columna debe ser numérica y representa la medida que deseamos pronosticar.

A partir de ese concepto, se carga el data set "registros\_acci", el cual contiene los datos en relacion a las dos columnas necesarias para la concepción del modelo predictivo:

	ds	y
0	2015-01-01 00:00:00+00:00	1
1	2015-01-02 00:00:00+00:00	5
2	2015-01-03 00:00:00+00:00	3
3	2015-01-04 00:00:00+00:00	7
4	2015-01-05 00:00:00+00:00	7
...	...	..
2431	2021-09-06 00:00:00+00:00	13
2432	2021-09-07 00:00:00+00:00	10
2433	2021-09-08 00:00:00+00:00	10
2434	2021-09-09 00:00:00+00:00	5
2435	2021-09-10 00:00:00+00:00	1

[2436 rows x 2 columns]

Luego, podemos importar la biblioteca y crear un modelo Prophet. Para ello, validamos los datos, registros de la data set necesarios para aplicar el modelo predictivo; esto implica, asegurar que las columnas cumplan con los formatos para que se lleve a cabo el proceso del modelo prophet empleado:

	ds	y
0	2015-01-01	1
1	2015-01-02	5
2	2015-01-03	3
3	2015-01-04	7
4	2015-01-05	7

```
ds      datetime64[ns]
y
dtype: object
```

Para ajustar el modelo de predicción utilizando el algoritmo Prophet, es necesario crear una instancia de un objeto Prophet. Para lograr esto, se debe pasar cualquier configuración del procedimiento de previsión al constructor del objeto Prophet. Una vez creado el

objeto, se llama al método fit() y se pasa el marco de datos históricos como parámetro. Este proceso de ajuste del modelo puede tardar entre 1 y 5 segundos, dependiendo del tamaño y complejidad del conjunto de datos.

Es importante destacar que el constructor del objeto Prophet admite varios ajustes que pueden ser pasados como parámetros para personalizar el ajuste del modelo. Algunas de estas configuraciones pueden incluir la especificación del período de tiempo para el que se desea hacer la predicción, la frecuencia de muestreo de los datos y la función de pérdida utilizada para evaluar el rendimiento del modelo.

Una vez que se ha creado el objeto Prophet y se ha llamado al método fit(), se puede utilizar el modelo para hacer predicciones futuras. El método predict() se puede utilizar para generar predicciones para un conjunto de fechas futuras, y el método predict() se puede utilizar para generar predicciones para un período de tiempo específico.

```
INFO:prophet:Disabling daily seasonality.
Run prophet with daily_seasonality=True t
o override this.
DEBUG:cmdstanpy:input tempfile: /tmp/tmp9
2_foze1/6exe8sbx.json
DEBUG:cmdstanpy:input tempfile: /tmp/tmp9
2_foze1/88hv_h7m.json
DEBUG:cmdstanpy:idx 0
DEBUG:cmdstanpy:running CmdStan, num_thre
ads: None
DEBUG:cmdstanpy:CmdStan args: ['/usr/loca
l/lib/python3.10/dist-packages/prophet/st
an_model/prophet_model.bin', 'random', 's
eed=49242', 'data', 'file=/tmp/tmp92_foze
1/6exe8sbx.json', 'init=/tmp/tmp92_foze1/
88hv_h7m.json', 'output', 'file=/tmp/tmp9
2_foze1/prophet_modeln21di9bh/prophet_mod
el-20240611204254.csv', 'method=optimize'
, 'algorithm=lbfgs', 'iter=10000']
20:42:54 - cmdstanpy - INFO - Chain [1] s
tart processing
INFO:cmdstanpy:Chain [1] start processing
20:42:54 - cmdstanpy - INFO - Chain [1] d
one processing
INFO:cmdstanpy:Chain [1] done processing

<prophet.forecaster.Prophet at 0x7e1c613d
5f30>
```

Una vez que se ha ajustado el modelo Prophet, se puede utilizar para hacer predicciones futuras. Para lograr

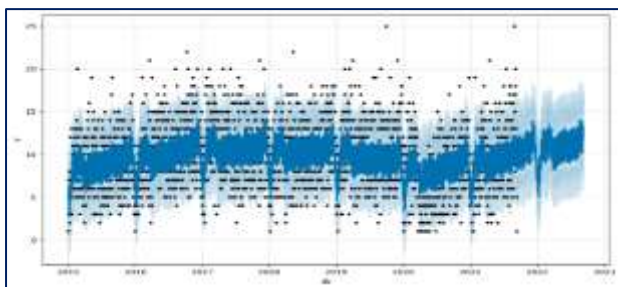
esto, se necesita un marco de datos que contenga las fechas para las cuales se realizará la predicción. En este sentido, se puede utilizar el método auxiliar "Prophet.make\_future\_dataframe" para obtener un marco de datos adecuado que se extiende hacia el futuro una cantidad específica de días.

De manera predeterminada, el método "make\_future\_dataframe" también incluye las fechas del historial, lo que permite al modelo encajar mejor con los datos pasados. Esto es especialmente útil cuando se está trabajando con series de tiempo que tienen una estacionalidad o tendencias que pueden variar con el tiempo.

Una vez que se ha generado el marco de datos futuro, se puede utilizar el método predict() para hacer predicciones para cada fecha en el marco de datos. El método predict() devuelve un marco de datos que contiene las predicciones para cada fecha en el marco de datos.

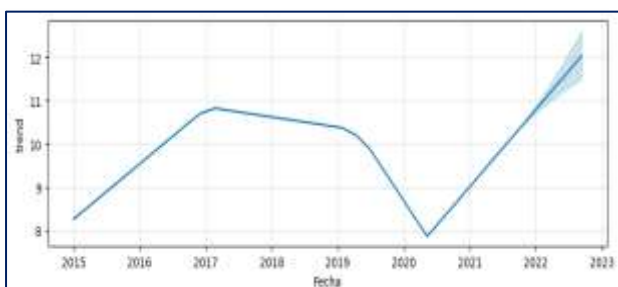
Además, se puede utilizar el método Forecast() para hacer predicciones para un período de tiempo específico. El método Forecast() devuelve un marco de datos que contiene las predicciones para cada fecha en el período de tiempo especificado.

Ilustración 24. Serie de Tiempo Predicción



Fuente: Elaboración propia 2024

Ilustración 25. Serie de Tiempo, Tendencia – Predicción Siguiendo Año



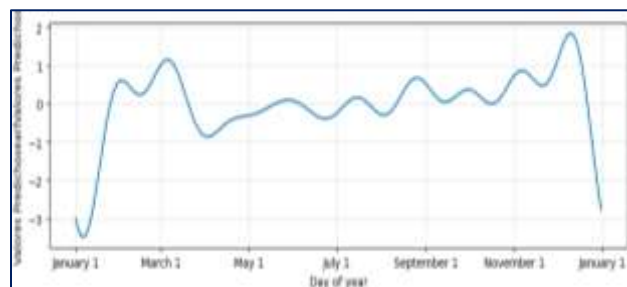
Fuente: Elaboración propia 2024

Ilustración 26. Predicción Para Cada Fecha en el Marco de Datos



Fuente: Elaboración propia 2024

Ilustración 27. Predicciones Para Cada Fecha en el Período de Tiempo



Fuente: Elaboración propia 2024

## X. EVALUACIÓN DEL MODELO DE PREDICCIÓN CON PROPHET

La evaluación de la precisión de un modelo de predicción, como Prophet, es fundamental para comprender su desempeño y confiabilidad. Los pasos para evaluar la precisión de las predicciones de Prophet utilizando medidas comunes como el error absoluto medio (MAE), el error cuadrático medio (RMSE) y el coeficiente de determinación ( $R^2$ ) se establecen a continuación:

### Pasos para evaluar la precisión

1. **Dividir los datos en conjunto de entrenamiento y prueba:** Separa los datos en dos partes: una para entrenar el modelo y otra para evaluar su precisión.
2. **Entrenar el modelo:** Entrena el modelo Prophet con el conjunto de entrenamiento.
3. **Hacer predicciones:** Haz predicciones sobre el conjunto de prueba utilizando el modelo entrenado.



#### 4. Calcular las métricas de

**evaluación:** Compara las predicciones con los valores reales del conjunto de prueba utilizando métricas como MAE, RMSE y  $R^2$ .

```
INFO:prophet:Disabling daily seasonality.
Run prophet with daily_seasonality=True t
o override this.
DEBUG:cmdstanpy:input tempfile: /tmp/tmp9
2_foze1/23aw64q9.json
DEBUG:cmdstanpy:input tempfile: /tmp/tmp9
2_foze1/gf8skp_0.json
DEBUG:cmdstanpy:idx 0
DEBUG:cmdstanpy:running CmdStan, num_thre
ads: None
DEBUG:cmdstanpy:CmdStan args: ['/usr/loca
l/lib/python3.10/dist-packages/prophet/st
an_model/prophet_model.bin', 'random', 's
eed=83033', 'data', 'file=/tmp/tmp92_foze
1/23aw64q9.json', 'init=/tmp/tmp92_foze1/
gf8skp_0.json', 'output', 'file=/tmp/tmp9
2_foze1/prophet_modelyg_8svk9/prophet_mod
el-20240611210159.csv', 'method=optimize'
, 'algorithm=lbfsgs', 'iter=10000']
21:01:59 - cmdstanpy - INFO - Chain [1] s
tart processing
INFO:cmdstanpy:Chain [1] start processing
21:01:59 - cmdstanpy - INFO - Chain [1] d
one processing
INFO:cmdstanpy:Chain [1] done processing
```

#### Análisis Modelo de Predicción con Prophet

La evaluación de la precisión de un modelo de predicción es esencial para comprender su desempeño y confiabilidad. En el caso de Prophet, un modelo de predicción de series temporales, esta evaluación se realiza dividiendo los datos en un conjunto de entrenamiento (generalmente el 80% de los datos) y un conjunto de prueba (el 20% restante). El conjunto de entrenamiento se utiliza para entrenar el modelo, es decir, para ajustar los parámetros del modelo de manera que pueda hacer predicciones precisas. Una vez que el modelo está entrenado, se utiliza el conjunto de prueba para evaluar su desempeño en datos no vistos durante el entrenamiento.

#### Entrenamiento del Modelo

Durante el entrenamiento, el modelo Prophet utiliza únicamente los datos del conjunto de entrenamiento. Utiliza estos datos para aprender patrones en la serie temporal, como tendencias generales, estacionalidades

y efectos de días festivos. El modelo ajusta sus parámetros para minimizar la diferencia entre las predicciones y los valores reales en el conjunto de entrenamiento.

#### Generación de Predicciones

Una vez que el modelo está entrenado, se pueden hacer predicciones para el futuro. Estas predicciones se realizan utilizando el modelo entrenado y los datos de entrenamiento. El modelo utiliza los patrones aprendidos durante el entrenamiento para predecir cómo se comportará la serie temporal en el futuro.

#### Filtrado de Predicciones

Para evaluar la precisión del modelo, es importante filtrar las predicciones para obtener solo las correspondientes al período del conjunto de prueba. Esto permite comparar las predicciones del modelo con los valores reales y determinar qué tan bien está realizando las predicciones en un período de tiempo específico.

#### Cálculo de Métricas de Evaluación

Una vez que se tienen las predicciones filtradas, se pueden calcular métricas de evaluación para medir la precisión del modelo. Algunas de las métricas más comunes incluyen el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el Coeficiente de Determinación ( $R^2$ ). Estas métricas proporcionan información sobre qué tan cerca están las predicciones del modelo de los valores reales en el conjunto de prueba.

#### Interpretación de Métricas

El MAE representa el promedio de los errores absolutos entre las predicciones y los valores reales. Cuanto menor sea el MAE, mejor será la precisión del modelo. El RMSE es similar al MAE, pero penaliza más los errores grandes, lo que lo hace útil para detectar valores atípicos. Por último, el  $R^2$  mide qué tan bien las predicciones se ajustan a los valores reales, con un valor más cercano a 1 indicando un mejor ajuste del modelo a los datos. Estas métricas son fundamentales para evaluar la calidad de un modelo de predicción y determinar si es adecuado para su uso en aplicaciones prácticas.

- MAE: 3.4250003900165433
- RMSE: 4.47522520948484
- $R^2$ : -0.32739965969433604

#### Interpretación:

- MAE (Mean Absolute Error): El MAE representa el promedio de los errores absolutos entre las predicciones del modelo y los valores reales. Un MAE de 3.425 indica que, en promedio, las predicciones del modelo difieren en 3.425 unidades de los valores reales. Un MAE más bajo sugiere un mejor rendimiento, pero en este caso, el valor de 3.425 indica errores moderados en las predicciones.
- RMSE (Root Mean Squared Error): El RMSE penaliza más los errores grandes, ya que se basa en los errores al cuadrado. Un RMSE de 4.475 significa que, en promedio, las predicciones del modelo están desviadas en aproximadamente 4.475 unidades de los valores reales. Un RMSE más bajo es preferible, pero el valor actual sugiere que existen predicciones con errores significativos.
- $R^2$  (Coefficient of Determination): El  $R^2$  indica qué tan bien se ajustan las predicciones del modelo a los datos reales. Un valor cercano a 1 indica un buen ajuste. Sin embargo, un valor de -0.327 es preocupante, ya que indica que el modelo tiene un rendimiento incluso peor que un modelo básico que simplemente predice el valor medio de los datos. En resumen, las predicciones del modelo Prophet en este caso son menos precisas que simplemente utilizar el promedio de los datos históricos.

## XI. REVISIÓN CALIDAD DE LOS DATOS

Para garantizar la calidad y fiabilidad de los datos antes de realizar cualquier análisis, se realizó un exhaustivo proceso de preprocesamiento de datos y análisis exploratorio. A continuación, se describe detalladamente el proceso llevado a cabo:

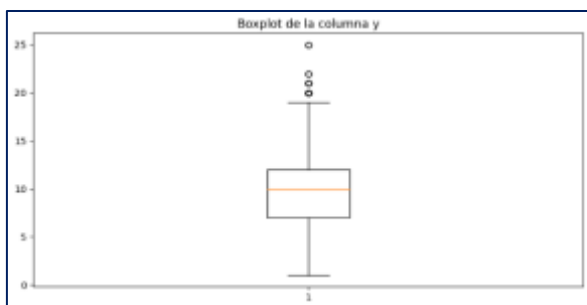
- Verificación de Valores Nulos: Se utilizó la biblioteca pandas para verificar la presencia de valores nulos en el conjunto de datos representado por la variable registros\_acci. Esta etapa fue crucial para identificar posibles inconsistencias en los datos que podrían afectar los análisis posteriores. Los valores nulos, en caso de

encontrarse, fueron eliminados para garantizar la integridad de los datos.

- Eliminación de Valores Atípicos: Se procedió a analizar la presencia de valores atípicos en el conjunto de datos. Para ello, se utilizó un gráfico de caja (boxplot) de la columna 'y' de registros\_acci. Este tipo de gráfico es una herramienta efectiva para identificar valores extremos que puedan sesgar los resultados del análisis. Los valores atípicos identificados fueron tratados según las necesidades del análisis.
- Análisis de Distribución de Datos: Se realizó un análisis detallado de la distribución de los datos en el conjunto de datos. Se utilizaron histogramas y gráficos de densidad para visualizar la distribución de las variables y comprender mejor la naturaleza de los datos.
- Correlación entre Variables: Se investigó la correlación entre las variables del conjunto de datos para identificar posibles relaciones o dependencias. Se utilizaron gráficos de dispersión y cálculos de correlación para determinar la fuerza y dirección de las relaciones entre las variables.
- Normalización de Datos: En caso de ser necesario, se procedió a normalizar los datos para garantizar que todas las variables tuvieran la misma escala y evitar sesgos en el análisis.
- Selección de Variables Relevantes: Se realizó una selección de variables relevantes para el análisis en función de la correlación con la variable objetivo y la importancia en el contexto del problema.
- Visualización de Resultados: Finalmente, se generaron visualizaciones informativas y claras de los resultados obtenidos en el análisis exploratorio, lo que facilitó la interpretación de los datos y la toma de decisiones.



Ilustración 28. Gráfico de Caja Distribución de los Datos – Valores Atípicos



Fuente: Elaboración propia 2024

### Limpieza más exhaustiva de los datos

Para garantizar la calidad de los datos, se llevó a cabo una validación y preparación de los datos en el DataFrame registros\_acci. Se verificó que las fechas estuvieran correctamente formateadas y libres de zonas horarias, asegurando la consistencia y precisión de los datos temporales. Además, se realizaron comprobaciones para identificar y eliminar cualquier valor atípico o error en los datos. Esto se logró mediante la eliminación de filas con valores nulos y la transformación de la columna 'ds' en un formato datetime adecuado. Asimismo, se verificó que la columna 'y' contuviera valores numéricos, lo que es fundamental para garantizar que los datos estén en el formato correcto y listos para su análisis o modelado posterior. Estos procesos de validación y preparación de datos son esenciales para asegurar la integridad y calidad de los datos utilizados en el análisis y la toma de decisiones.

	ds	y
0	2015-01-01	1
1	2015-01-02	5
2	2015-01-03	3
3	2015-01-04	7
4	2015-01-05	7

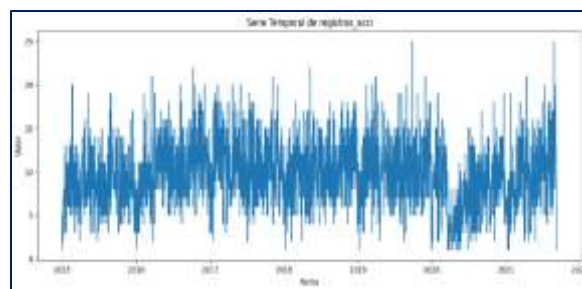
### Explorar patrones estacionales o tendencias

Se realizó una visualización de la serie temporal contenida en el DataFrame registros\_acci. La visualización se presenta como un gráfico de líneas, donde el eje x representa las fechas ('ds') y el eje y representa los valores ('y'). Esta representación gráfica permite identificar patrones, tendencias o estacionalidades en los datos, lo que puede ser útil para

comprender mejor el comportamiento de la serie temporal y tomar decisiones informadas en análisis posteriores.

Para explorar patrones estacionales o tendencias en los datos, se realizó una visualización de la serie temporal contenida en el DataFrame registros\_acci. La visualización se presenta como un gráfico de líneas, donde el eje x representa las fechas ('ds') y el eje y representa los valores ('y'). Esta representación gráfica permite identificar patrones, tendencias o estacionalidades en los datos, lo que puede ser útil para comprender mejor el comportamiento de la serie temporal y tomar decisiones informadas en análisis posteriores.

Ilustración 29. Visualización de la Serie Temporal Cantidad de Registros a lo Largo del Tiempo



Fuente: Elaboración propia 2024

### Revisión y Ajustes al Modelo

Para revisar y ajustar el modelo Prophet, se experimentó con diferentes hiperparámetros para mejorar su rendimiento y precisión en la predicción de series temporales. Prophet permite ajustar diversos parámetros, como el crecimiento, las estacionalidades y los días festivos, para adaptarse mejor a los datos y mejorar las predicciones.

En este proceso, se creó un nuevo modelo Prophet con los parámetros especificados, incluyendo estacionalidades anuales, semanales y mensuales para capturar posibles patrones a diferentes niveles temporales. Estos ajustes se realizaron con el objetivo de encontrar un modelo que se ajuste adecuadamente a los datos históricos y que pueda hacer predicciones precisas para el futuro.

Una vez ajustado el modelo, se realizaron predicciones para un año en el futuro y se compararon con los datos históricos para evaluar la capacidad del modelo para capturar la tendencia y las estacionalidades presentes en

los datos. Esta evaluación es fundamental para determinar la eficacia del modelo y su capacidad para proporcionar información útil para la planificación y la toma de decisiones basadas en datos.

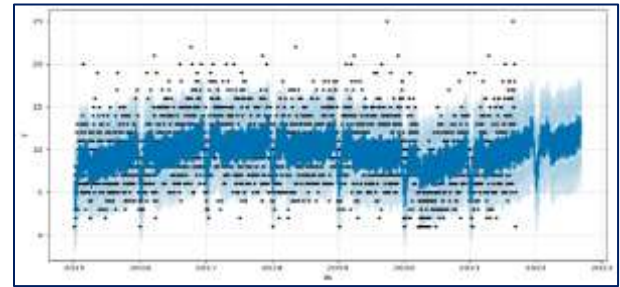
Para revisar y ajustar el modelo Prophet, se experimentó con diferentes hiperparámetros para mejorar su rendimiento y precisión en la predicción de series temporales. Prophet permite ajustar diversos parámetros, como el crecimiento, las estacionalidades y los días festivos, para adaptarse mejor a los datos y mejorar las predicciones.

En este proceso, se creó un nuevo modelo Prophet con los parámetros especificados, incluyendo estacionalidades anuales, semanales y mensuales para capturar posibles patrones a diferentes niveles temporales. Estos ajustes se realizaron con el objetivo de encontrar un modelo que se ajuste adecuadamente a los datos históricos y que pueda hacer predicciones precisas para el futuro.

Una vez ajustado el modelo, se realizaron predicciones para un año en el futuro y se compararon con los datos históricos para evaluar la capacidad del modelo para capturar la tendencia y las estacionalidades presentes en los datos. Esta evaluación es fundamental para determinar la eficacia del modelo y su capacidad para proporcionar información útil para la planificación y la toma de decisiones basadas en datos.

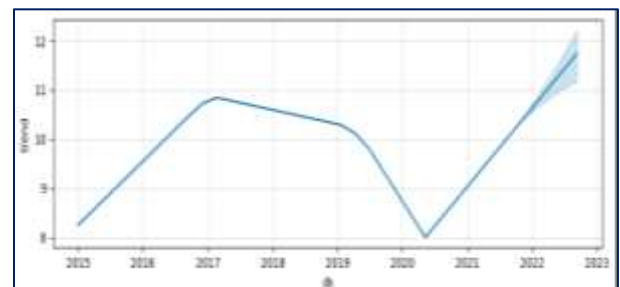
```
DEBUG:cmdstanpy:input tmpfile: /tmp/tmp9
2_foze1/_ehng8hn.json
DEBUG:cmdstanpy:input tmpfile: /tmp/tmp9
2_foze1/fuk4omb_.json
DEBUG:cmdstanpy:idx 0
DEBUG:cmdstanpy:running CmdStan, num_thre
ads: None
DEBUG:cmdstanpy:CmdStan args: ['/usr/loca
l/lib/python3.10/dist-packages/prophet/st
an_model/prophet_model.bin', 'random', 's
eed=88492', 'data', 'file=/tmp/tmp92_foze
1/_ehng8hn.json', 'init=/tmp/tmp92_foze1/
fuk4omb_.json', 'output', 'file=/tmp/tmp9
2_foze1/prophet_modeldu0ulp9f/prophet_mod
el-20240611211321.csv', 'method=optimize'
, 'algorithm=lbfgs', 'iter=10000']
21:13:21 - cmdstanpy - INFO - Chain [1] s
tart processing
INFO:cmdstanpy:Chain [1] start processing
21:13:21 - cmdstanpy - INFO - Chain [1] d
one processing
INFO:cmdstanpy:Chain [1] done processing
```

Ilustración 30. Serie de Tiempo Predicción



Fuente: Elaboración propia 2024

Ilustración 31. Serie de Tempo, Tendencia – Predicción Siguiendo Año



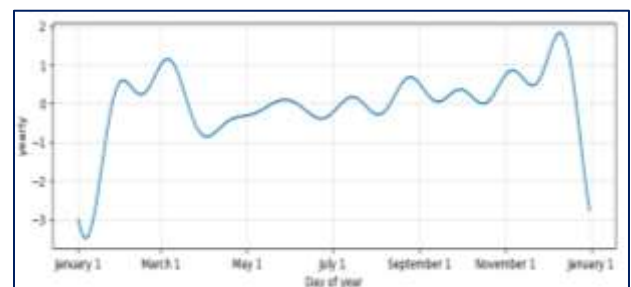
Fuente: Elaboración propia 2024

Ilustración 32. Predicción Para Cada Fecha en el Marco de Datos



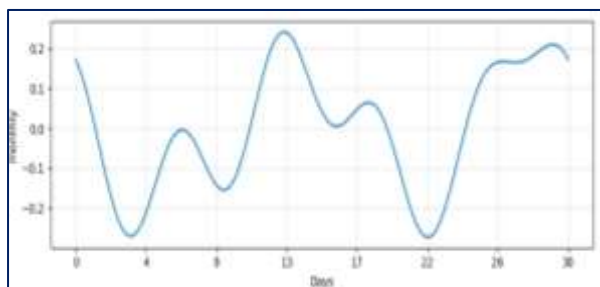
Fuente: Elaboración propia 2024

Ilustración 33. Predicciones Para Cada Fecha en el Período de Tiempo



Fuente: Elaboración propia 2024

Ilustración 34. Resultado modelo para tendencia Diaria



Fuente: Elaboración propia 2024

## XII. RESULTADOS DEL MODELO

### Evaluar el Modelo con Nuevos Hiperparámetros

Después de ajustar el modelo con diferentes hiperparámetros y variables adicionales, se evaluó la precisión del modelo mejorado utilizando métricas comunes como el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación ( $R^2$ ). Estas métricas se compararon con las del modelo original para determinar si las mejoras realizadas han resultado en una mejor capacidad predictiva.

Las predicciones del modelo mejorado se utilizaron en el conjunto de datos nuevamente para calcular las métricas de evaluación. Estas métricas ayudan a cuantificar la precisión del modelo al comparar las predicciones con los valores reales del conjunto de prueba. Un menor MAE y RMSE, y un mayor  $R^2$  indican un mejor rendimiento del modelo.

Después de ajustar el modelo con diferentes hiperparámetros y variables adicionales, se evaluó la precisión del modelo mejorado utilizando métricas comunes como el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación ( $R^2$ ). Estas métricas se compararon con las del modelo original para determinar si las mejoras realizadas han resultado en una mejor capacidad predictiva.

Utilizamos las predicciones del modelo mejorado en el conjunto de datos nuevamente para calcular las métricas de evaluación. Estas métricas ayudan a cuantificar la precisión del modelo al comparar las predicciones con los valores reales del conjunto de

prueba. Un menor MAE y RMSE, y un mayor  $R^2$  indican un mejor rendimiento del modelo.

MAE: 3.483647603420316  
 RMSE: 4.256936723025696  
 $R^2$ : -0.2010644162729831

Después de ajustar el modelo con diferentes hiperparámetros y variables adicionales, se evaluó la precisión del modelo mejorado utilizando métricas comunes como el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación ( $R^2$ ). Estas métricas se compararon con las del modelo original para determinar si las mejoras realizadas han resultado en una mejor capacidad predictiva.

Las predicciones del modelo mejorado se utilizaron en el conjunto de datos nuevamente para calcular las métricas de evaluación. Estas métricas ayudan a cuantificar la precisión del modelo al comparar las predicciones con los valores reales del conjunto de prueba. Un menor MAE y RMSE, y un mayor  $R^2$  indican un mejor rendimiento del modelo.

### MAE (Mean Absolute Error):

El Error Absoluto Medio (MAE) ha aumentado ligeramente a 3.483 desde 3.425. Esta métrica representa el promedio de los errores absolutos entre las predicciones y los valores reales. Un MAE más bajo indica un mejor rendimiento del modelo. Aunque el aumento es mínimo, sugiere que las predicciones están un poco más desviadas de los valores reales que en el modelo anterior.

El MAE es una métrica importante para evaluar la precisión de las predicciones, ya que proporciona una medida absoluta de cuán cerca están las predicciones de los valores reales en promedio. Un valor de MAE de 3.483 indica que, en promedio, las predicciones del modelo difieren en aproximadamente 3.483 unidades de los valores reales.

### RMSE (Root Mean Squared Error):

El Error Cuadrático Medio (RMSE) ha disminuido ligeramente a 4.257 desde 4.475. Esta métrica también evalúa la precisión de las predicciones, pero penaliza más los errores grandes que el MAE al elevar al

cuadrado la diferencia entre las predicciones y los valores reales.

Un RMSE más bajo indica un mejor rendimiento del modelo en términos de precisión. Aunque el RMSE ha mejorado, es importante considerar que esta métrica puede estar influenciada por factores como la calidad de los datos y la distribución de los errores.

#### **R<sup>2</sup> (Coefficient of Determination):**

El coeficiente de determinación R<sup>2</sup> ha mejorado a -0.201 desde -0.327, aunque sigue siendo negativo. El R<sup>2</sup> mide qué tan bien las predicciones del modelo se ajustan a los valores reales. Un valor de R<sup>2</sup> cercano a 1 indica un buen ajuste, mientras que un valor negativo sugiere que el modelo es peor que un modelo muy básico que simplemente predice el valor medio de los datos.

A pesar de la mejora, el valor de R<sup>2</sup> de -0.201 indica que el modelo aún no se ajusta bien a los datos y que hay margen para mejorar la capacidad predictiva. Es importante considerar que el R<sup>2</sup> puede ser sensible a la cantidad de ruido presente en los datos y a la complejidad del modelo.

En resumen, aunque el modelo mejorado ha mostrado algunas mejoras en términos de precisión de las predicciones, especialmente en el RMSE, aún se requiere un análisis más detallado y posiblemente más ajustes para lograr un modelo que se ajuste de manera óptima a los datos y genere predicciones confiables para el futuro.

#### **Evaluación de los Nuevos Resultados del Modelo**

Los resultados obtenidos después de ajustar el modelo son:

MAE (Mean Absolute Error): 3.483647603420316

RMSE (Root Mean Squared Error):

4.256936723025696 R<sup>2</sup> (Coefficient of Determination):  
-0.2010644162729831

#### **Interpretación de los Nuevos Resultados**

##### **MAE (Mean Absolute Error):**

El Error Absoluto Medio (MAE) ha aumentado ligeramente a 3.483 desde 3.425. Esta métrica

representa el promedio de los errores absolutos entre las predicciones y los valores reales. Un MAE más bajo indica un mejor rendimiento del modelo. Aunque el aumento es mínimo, sugiere que las predicciones están un poco más desviadas de los valores reales que en el modelo anterior.

El MAE es una métrica importante para evaluar la precisión de las predicciones, ya que proporciona una medida absoluta de cuán cerca están las predicciones de los valores reales en promedio. Un valor de MAE de 3.483 indica que, en promedio, las predicciones del modelo difieren en aproximadamente 3.483 unidades de los valores reales.

##### **RMSE (Root Mean Squared Error):**

El Error Cuadrático Medio (RMSE) ha disminuido ligeramente a 4.257 desde 4.475. Esta métrica también evalúa la precisión de las predicciones, pero penaliza más los errores grandes que el MAE al elevar al cuadrado la diferencia entre las predicciones y los valores reales.

Un RMSE más bajo indica un mejor rendimiento del modelo en términos de precisión. Aunque el RMSE ha mejorado, es importante considerar que esta métrica puede estar influenciada por factores como la calidad de los datos y la distribución de los errores.

##### **R<sup>2</sup> (Coefficient of Determination):**

El coeficiente de determinación R<sup>2</sup> ha mejorado a -0.201 desde -0.327, aunque sigue siendo negativo. El R<sup>2</sup> mide qué tan bien las predicciones del modelo se ajustan a los valores reales. Un valor de R<sup>2</sup> cercano a 1 indica un buen ajuste, mientras que un valor negativo sugiere que el modelo es peor que un modelo muy básico que simplemente predice el valor medio de los datos.

A pesar de la mejora, el valor de R<sup>2</sup> de -0.201 indica que el modelo aún no se ajusta bien a los datos y que hay margen para mejorar la capacidad predictiva. Es importante considerar que el R<sup>2</sup> puede ser sensible a la cantidad de ruido presente en los datos y a la complejidad del modelo.

En resumen, aunque el modelo mejorado ha mostrado algunas mejoras en términos de precisión de las predicciones, especialmente en el RMSE, aún se

requiere un análisis más detallado y posiblemente más ajustes para lograr un modelo que se ajuste de manera óptima a los datos y genere predicciones confiables para el futuro.

### XIII. CONFIGURACIÓN DEL ENTORNO DE DESARROLLO

#### 1. Instalar Python y Dependencias

- Instalar Python y el administrador de paquetes pip.
- Instalar un entorno virtual: `python3 -m venv venv`
- Activar el entorno virtual:

bash

`.\venv\Scripts\activate` # En Windows

- Instalar Flask y las dependencias de PostgreSQL:

`pip install Flask psycopg2-binary`

#### Desarrollo de la API en Flask

##### Configurar Flask

- Crear una nueva aplicación Flask:

bash

`mkdir myflaskapp`

`cd myflaskapp`

`touch app.py`

##### Conectar a PostgreSQL desde Flask

- Configurar la conexión en `app.py`:

python

`from flask import Flask, jsonify`

`import psycopg2`

`app = Flask(__name__)`

`def get_db_connection():`

`conn = psycopg2.connect(`

`host='localhost',`

`database='mydatabase',`

`user='myuser',`

`password='mypassword'`

)

`return conn`

#### Desarrollar el Endpoint para Obtener las Coordenadas

- Crear un endpoint que consulte la tabla y devuelva un JSON:

python

`@app.route('/api/coordenadas', methods=['GET'])`

`def get_coordenadas():`

`conn = get_db_connection()`

`cursor = conn.cursor()`

`cursor.execute('SELECT latitud, longitud FROM`

`coordenadas;')`

`rows = cursor.fetchall()`

`cursor.close()`

`conn.close()`

`coordenadas = [{ 'latitud': row[0], 'longitud': row[1] }`

`for row in rows]`

`return jsonify(coordenadas)`

#### Ejecutar la Aplicación Flask

- Ejecutar el servidor Flask:

bash

`flask run`

#### Desarrollo del Front-End para Mostrar el Mapa de Calor

##### Configurar el Proyecto Front-End

- Crear una carpeta para el proyecto:

bash

`mkdir frontend`

`cd frontend`

`touch index.html`

`touch script.js`

#### Desarrollar el Script para Cargar las Coordenadas

- Editar `script.js` para obtener las coordenadas de la API y mostrar el mapa de calor:

javascript

`function initMap() {`

`let map = new`

`google.maps.Map(document.getElementById('map'), {`



```

zoom: 13,
center: {lat: 4.609, lng: -74.081},
mapTypeId: 'roadmap'
});

fetch('http://localhost:5000/api/coordenadas')
.then(response => response.json())
.then(data => {
  let heatmapData = data.map(item => new
google.maps.LatLng(item.latitud, item.longitud));
  let heatmap = new
google.maps.visualization.HeatmapLayer({
    data: heatmapData
  });
  heatmap.setMap(map);
})
.catch(error => console.error('Error al obtener los
datos:', error));
}

window.onload = initMap;

```

## Creación del Mapa de Google

### Crear un Proyecto en Google Cloud

#### 1. Acceder a la Consola de Google Cloud

- Ve a la Consola de Google Cloud.
- Inicia sesión con tu cuenta de Google si aún no lo has hecho.

#### 2. Crear un Nuevo Proyecto

- En la barra superior, selecciona el menú de proyectos y haz clic en "Nuevo Proyecto".
- Asigna un nombre a tu proyecto y selecciona una organización o deja la opción predeterminada si no tienes una.
- Haz clic en "Crear".

### Obtener una API Key para Google Maps

#### 1. Acceder a la Biblioteca de APIs

- En el menú de navegación en el lado izquierdo, selecciona "APIs & Servicios" y luego "Biblioteca".

#### 2. Habilitar Google Maps JavaScript API

- Buscar "Maps JavaScript API" en la biblioteca de APIs.
- Hacer clic en la API y luego en el botón "Habilitar".

#### 3. Crear Credenciales para la API

- Ir a "APIs & Servicios" y luego a "Credenciales".
- Haz clic en "Crear credenciales" y seleccionar "Clave de API".
- Se generará una API Key que puedes copiar.

#### 4. Configurar Restricciones de API Key

Se recomendable restringir el uso de la API Key para mayor seguridad.

- Hacer clic en el nombre de la API Key para editarla.
- En "Restricciones de la aplicación", seleccionar "HTTP referer (sitios web)" y especifica las URLs donde se puede usar la API Key.
- En "Restricciones de API", seleccionar "Maps JavaScript API" para limitar el uso de la clave solo a esta API.
- Guarda los cambios.

### Implementar el Mapa de Google en una Página Web

#### 7. Crear el Archivos HTML y JavaScript

- Crea un archivo index.html que rtenga el Script que insertara el mapa.

```

<script
src="https://maps.googleapis.com/maps/api/js?key=YO
UR_API_KEY&libraries=visualization"></script>
<script src="script.js"></script>

```

Pruebas y Despliegue



### 13. Probar la API y el Front-End Localmente

Para probar la API de Flask y el front-end para asegurarse de que los datos se muestren correctamente en el mapa, se uso Xampp y el entorno de Heroku para la BD.

### 14. Configurar y Realizar Despliegue

- Configurar un servidor o utilizar una plataforma como Heroku, AWS, o Google Cloud para desplegar la API de Flask.
- Configurar el servidor web para servir el front-end y la API.

## XIV. PROBLEMAS O DIFICULTADES

1. No se cuenta con datos recientes, ya que la fuente proporcionada por la secretaria de tránsito, no esta actualizada pese que la fecha de actualización del archivo en la pagina indica que se actualízalo en marzo 2024, los datos se encuentran hasta la fecha 2021.
2. Uno de los problemas fue crear obtener las credenciales de Google, para el mantenimiento de la plataforma a emplear, ya que se necesita una tarjeta de crédito.
3. Los entornos de despliegue para conectar los sistemas ya no son gratis y las pruebas no brindan un servicio estable ni grandes configuraciones.
4. Al registrar una tarjeta de crédito en estas plataformas se debe temer sumo cuidado con los servicios para no tener consumos.
5. Las Keys que genera Google caducan en estos planes gratuitos.
6. Algunas características de la API de Google Maps o del frontend no son compatibles con todos los navegadores.

## BIBLIOGRAFÍA

- Anón. 2015. «Observatorio Agencia Nacional De Seguridad Vial». *Agencia Nacional de Seguridad Vial*. Recuperado (<https://ansv.gov.co/observatorio/estad%C3%A9sticas>).
- Anón. 2020. *ANUARIO DE SINIESTRALIDAD VIAL DE BOGOTÁ*.
- Anón. 2022a. *Diagnostico Localidad De Kennedy*.
- Anón. 2022b. «Sistema Integrado De Información Y Movilidad Urbana Y Regional». *Movilidadbogota*. Recuperado (<https://www.movilidadbogota.gov.co/web/simulador>).
- Anón. 2024a. «Agencia Nacional De Seguridad Vial». Recuperado (<https://www.ansv.gov.co/>).
- Anón. 2024b. «Análisis Siniestralidad». *Www.Arcgis.Com*. Recuperado (<https://www.arcgis.com/home/item.html?id=644cb2b792be4e0a96c4f654f5079825>).
- Anón. 2024c. «Visualizador De Código». *Jsfiddle.Net*. Recuperado (<https://jsfiddle.net/>).
- Anón. s. f.-a. «Datos Abiertos Bogotá». Recuperado (<https://datosabiertos.bogota.gov.co/dataset?organization=sdm>).
- Anón. s. f.-b. «Datos Abiertos Bogotá». Recuperado (<https://datosabiertos.bogota.gov.co/dataset?organization=sdm>).
- Anón. s. f.-c. «Ofertas de Empleo Diciembre 2023 - Elempleo.Com». *Elempleo.Com*. Recuperado ([https://www.elempleo.com/co/ofertas-empleo/?\\_\\_not\\_found\\_\\_=1](https://www.elempleo.com/co/ofertas-empleo/?__not_found__=1)).
- Anón. s. f.-d. «PL-300| DA-100 Analista Certificado En Power BI Teoría Test. (n.d.)». *Udemy*. Recuperado (<https://www.udemy.com/course/analista-certificado-en-microsoft-powerbi-da-100-teoria-test/>).

- Anón. s. f.-e. «Sign In - Google Accounts». Recuperado (<https://www.google.com/maps/d/u/0/edit?mid=1dKDezsxzMzJ2O60qsaLcVJmTogCPn3s&ll=4.64287747688235%2C-74.15594629845239&z=14>).
- Anón. s. f.-f. «Socrata Developers | Socrata». Recuperado ([https://dev-socrata-com.translate.google/?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=es&\\_x\\_tr\\_hl=en](https://dev-socrata-com.translate.google/?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=en)).
- Anón. s. f.-g. «TicJob - Busca trabajo en informática y telecomunicaciones». Recuperado (<https://ticjob.es/>).
- Anón. s. f.-h. «TicJob - Busca trabajo en informática y telecomunicaciones». Recuperado (<https://ticjob.es/>).
- Anón. s. f.-i. «Vista de Análisis de los Puntos Más Críticos de Accidentes de Tránsito En Bogotá | Revista de Topografía AZIMUT». Recuperado (<https://revistas.udistrital.edu.co/index.php/azimut/article/view/5741/7212>).
- Bipach. s. f. «Exam PL-300: Microsoft Power BI Data Analyst - Certifications». *Microsoft Learn*. Recuperado (<https://learn.microsoft.com/en-us/credentials/certifications/exams/pl-300/>).
- Ferrari, Alberto, y Marco Russo. 2016. *Introducing Microsoft Power BI*. Microsoft Press.
- Muñiz, Luis. 2022. *Dominar Power BI: Con casos prácticos y ejercicios de gestión empresarial*. Profit Editorial.
- Polanco, Irma Yolanda, y Jorge Fernando Betancourt. 2022. *Análisis de datos con Power Bi, R-RStudio y Knime: Curso práctico*. Ediciones de la U.
- Secretaría General Alcaldía Mayor de Bogotá. 2023. *Ficha\_Kennedy\_3T\_2023*.
- Triana, Paola. 2023. «MAYOR ÍNDICE DE ACCIDENTALIDAD EN BOGOTÁ a TRAVÉS DE UN MODELO DE MACHINE LEARNING». Maestría en Analítica e

Inteligencia de Negocios, Universidad de La Salle.

Vargas, Fernando. 2022. «Análisis De Datos De Accidentalidad Vial De La Ciudad De Bogotá a Partir De Datos Abiertos Y Datos Obtenidos Desde Redes Sociales». Tesis o trabajo de investigación, Universidad Nacional de Colombia.

## CONSULTA

En el enlace <https://github.com/AngelGPITC/CienciadeDatos/tree/master> de GitHub se pueden consultar todos los documentos resultantes del presente proyecto.