**MLDL**

**ESTEBAN GALVAN COLONIA**



**UNIVERSIDAD COOPERATIVA DE COLOMBIA**

**FACULTAD DE INGENIERÍA DE SISTEMAS**

**MINERÍA DE DATOS**

**SANTIAGO DE CALI 2024**

**MLDL**

**ESTEBAN GALVAN COLONIA**

**PROFESOR: HAROLD ADRIAN**



**UNIVERSIDAD COOPERATIVA DE COLOMBIA**

**FACULTAD DE INGENIERÍA DE SISTEMAS**

**MINERÍA DE DATOS**

**SANTIAGO DE CALI 2024**

**Tabla de Contenido**

[Contexto](#_Toc24742355) 5

[Nombre del proyecto](#_Toc24742355) 5

[Necesidades](#_Toc24742355) 5

[Requerimientos y especificaciones](#_Toc24742355) 5 – 19

[Conclusión](#_Toc24742355) 19

**Introducción del trabajo:**

Este ejercicio se realizará correspondiendo a la actividad propuesta por el profesor Harold Adrián para la asignatura de Minería de Datos.

**MLDL**

**Presentado por:**

**Esteban Galvan Colonia**

**Estudiante de Ingeniería 10° semestre**

**Contexto:**

MLDL

**Nombre del proyecto:**

NA.

**Necesidades:**

NA

**Descripcion:**

1. Exploratory Data Analysis (EDA): Realiza un análisis exploratorio de los datos para comprender las características generales del conjunto de datos, como la distribución de PRIMARY DESCRIPTION, SECONDARY DESCRIPTION, LOCATION DESCRIPTION, y otros campos categóricos.

Instalamos librerías

!pip install pandas

!pip install matplotlib

!pip install seaborn

!pip install Leafl

Tabla

Descripción generada automáticamenteTabla

Descripción generada automáticamenteTabla

Descripción generada automáticamente

Texto

Descripción generada automáticamente con confianza mediaDiagrama

Descripción generada automáticamenteGráfico

Descripción generada automáticamente2. Cleaning and Preprocessing Data: o Limpia y preprocesa los datos, como manejar valores nulos, convertir datos categóricos a numéricos (codificación one-hot o label encoding), y normalizar los datos numéricos.

Instalamos:

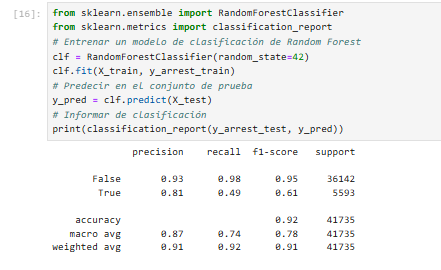
!pip install scikit-learn



Tabla

Descripción generada automáticamenteTabla

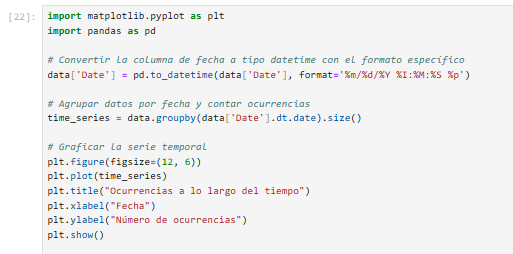
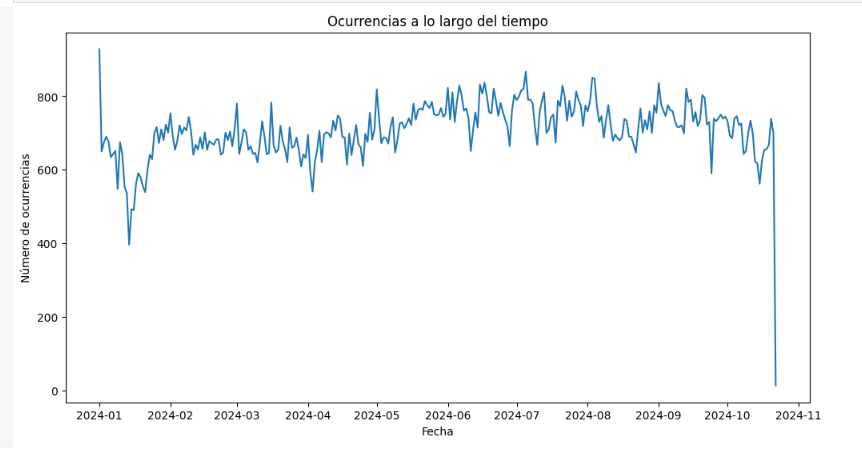
Descripción generada automáticamente

* 1. 3. Classification Task: Desarrolla un modelo de clasificación para predecir si un arresto se realizó (ARREST) utilizando características como PRIMARY DESCRIPTION, LOCATION DESCRIPTION, y WARD.
  2. 

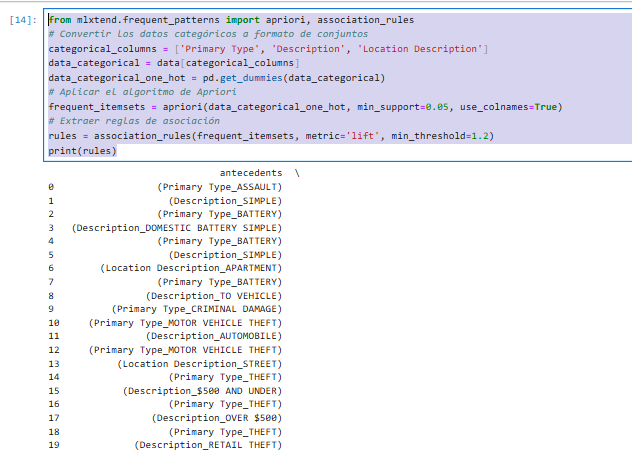
4. Regression Task: Crea un modelo de regresión para predecir la latitud y longitud de un caso en base a otras características del caso.



5. Time Series Analysis: Analiza la fecha y hora de ocurrencia (DATE OF OCCURRENCE) como una serie temporal para identificar patrones o tendencias a lo largo del tiempo.

* 1. 
  2. 
  3. 6. Geospatial Analysis: Visualiza los datos en un mapa geográfico para identificar patrones geoespaciales, como áreas con mayor incidencia de crímenes.
  4. instalamos geopandas:
  5. !pip install geopandas
  6. Actualizamos:
  7. !pip install --upgrade geopandas
  8. Descargamos del siguiente enlace: [Natural Earth » 1:110m Cultural Vectors - Free vector and raster map data at 1:10m, 1:50m, and 1:110m scales](https://www.naturalearthdata.com/downloads/110m-cultural-vectors/)
  9. el archivo: Admin 0 – Countries
  10. Tomamos del zip los siguientes archivos ne\_110m\_admin\_0\_countries.shp .shx y .dbf y los pegamos en la carpeta donde estamos trabajando, en este caso:
  11. C:\Users\ESTEBAN\Documents\esteban\repos-ucc\ucc-data-mining\ejercicios-jupiter\MLDL
  12. Imagen que contiene Gráfico

      Descripción generada automáticamente
  13. 7. Anomaly Detection: Implementa un modelo de detección de anomalías para identificar casos inusuales o fuera de lo común.
  14. Texto

      Descripción generada automáticamente
  15. 8. Association Rule Mining: Utiliza técnicas de minería de reglas de asociación para descubrir relaciones entre PRIMARY DESCRIPTION, SECONDARY DESCRIPTION, y otras características.
  16. Instalamos:
  17. !pip install mlxtend
  18. Tabla

      Descripción generada automáticamenteInterfaz de usuario gráfica, Aplicación, Tabla

      Descripción generada automáticamenteTabla

      Descripción generada automáticamente
  19. 9. Feature Importance Analysis: Utiliza técnicas como SHAP para analizar la importancia de las características en la predicción de ARREST o DOMESTIC.

1. instalamos:
2. !pip install shap
3. !pip install ipywidgets --upgrade

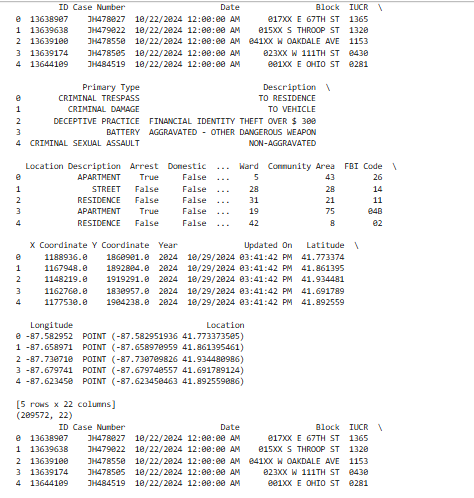
Reinstalamos lo siguiente:

!pip uninstall numpy -y

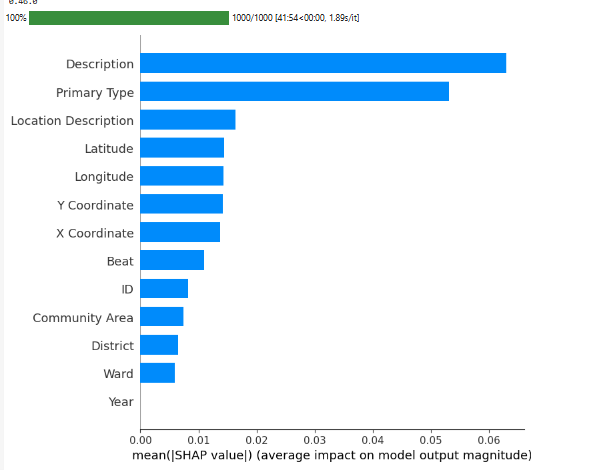
!pip install numpy==1.24.0

!pip install --upgrade shap



 Tabla

Descripción generada automáticamente

10. Neural Network Classification:

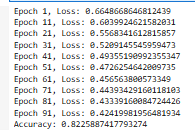
Desarrolla una red neuronal con Keras o PyTorch para clasificar los casos como DOMESTIC o no.

Instalamos:

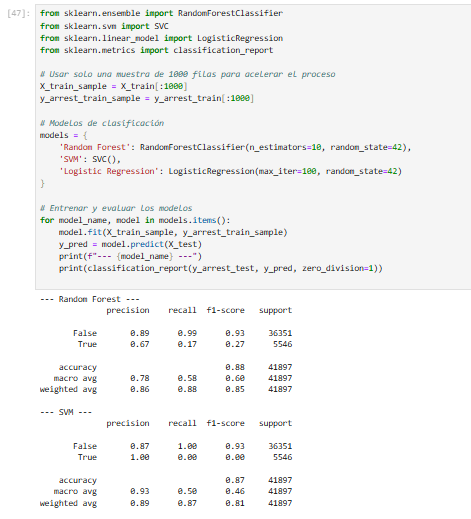
!pip install torch

Texto

Descripción generada automáticamente



11. Model Evaluation and Comparison: Compara diferentes modelos de clasificación (como árboles de decisión, SVM, redes neuronales) para predecir ARREST y evalúa su desempeño.

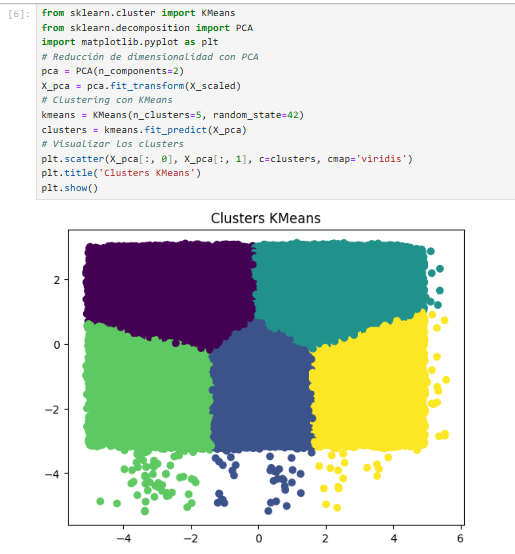
 Una captura de pantalla de un celular

Descripción generada automáticamente

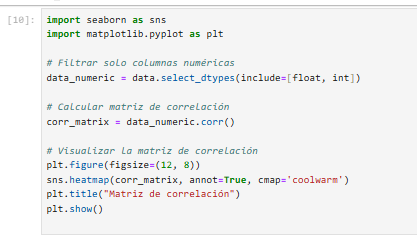
12. Clustering Analysis: Aplica técnicas de clustering (como k-means) para agrupar los datos en diferentes categorías basadas en características como PRIMARY DESCRIPTION, LOCATION DESCRIPTION, LATITUDE, y LONGITUDE.

Instalamos:

!pip install KMeans



13. Correlation Analysis: Analiza las correlaciones entre diferentes características del conjunto de datos, como PRIMARY DESCRIPTION, WARD, y FBI CD.

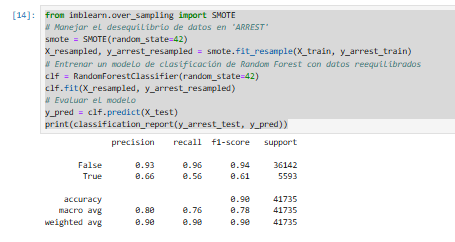
Gráfico

Descripción generada automáticamente

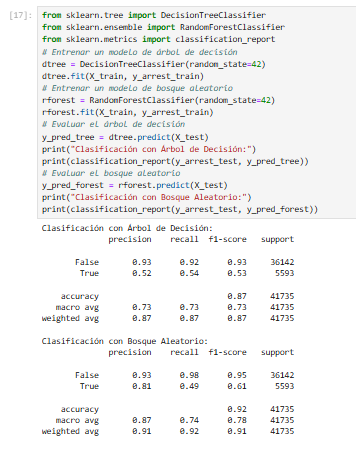
14. Data Imbalance Handling: Maneja los desequilibrios de datos en características como ARREST y DOMESTIC utilizando técnicas de sobremuestreo o submuestreo.

Instalamos:

!pip install imblearn

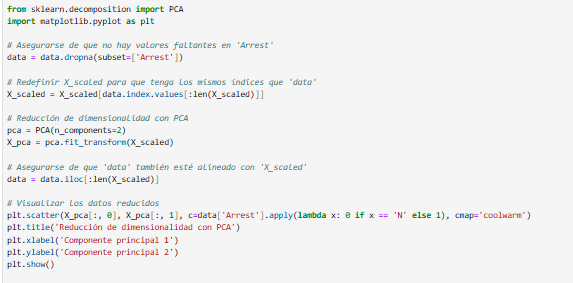


15. Decision Trees and Random Forest: Desarrolla un modelo de árbol de decisión o bosque aleatorio para predecir ARREST y DOMESTIC.



* 1. 16. Text Mining and NLP: Realiza minería de texto en campos como PRIMARY DESCRIPTION y SECONDARY DESCRIPTION para extraer características de texto y utilizarlas en modelos.

17. Dimensionality Reduction: Utiliza técnicas de reducción de dimensionalidad como PCA o t-SNE para visualizar datos complejos y reducir la dimensionalidad para modelos de aprendizaje.

* 1. 
  2. Gráfico, Gráfico de dispersión

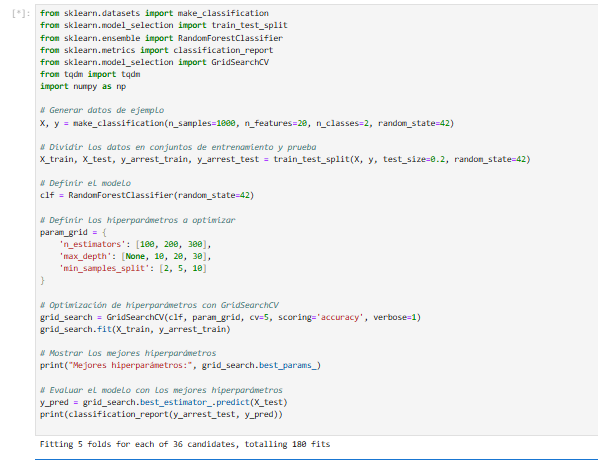
     Descripción generada automáticamente

18. Model Deployment: Implementa un modelo entrenado en una aplicación web sencilla utilizando Flask o Django para predecir ARREST basado en características de entrada.

* Crea un archivo llamado app.py
* Guarda los archivos de modelo (model\_arrest.pkl, scaler.pkl, label\_encoders.pkl) después de entrenar el modelo de ARREST.
* Ejecuta el archivo app.py con python app.py para iniciar el servidor Flask.
* Envía peticiones POST a http://localhost:5000/predict con datos JSON para obtener predicciones.

Instalamos:   
!pip install flask

19. Model Tuning and Hyperparameter Optimization: Realiza optimización de hiperparámetros utilizando técnicas como Grid Search o Random Search para mejorar el rendimiento de un modelo.

* 1. 

20. Time Series Forecasting: Utiliza modelos de series temporales para pronosticar la ocurrencia de casos futuros basados en datos históricos.

Instalamos:

!pip install fbprophet

**Conclusión del trabajo:**

Con base a lo estipulado y enseñado por el profesor Harold Adrián se llevó a cabo la realización de la actividad propuesta en clase.

# Referencias:

Clase 30 de Harold B.