Proyecto Final: Introducción a la Ciencia de Datos

Descripción del Problema

En este proyecto tendrá que trabajar con datos reales del departamento de policía de la ciudad de nueva york y se enmarca en el contexto de seguridad y criminalidad en los barrios de la ciudad. El objetivo de este proyecto es predecir la cantidad de delitos de cierto tipo por barrio.

El conjunto de datos a utilizar proviene de la página de datos abiertos de la ciudad de NY (https://opendata.cityofnewyork.us) y comprenden datos desde el 2013 hasta el 2020 de arrestos, colisiones vehiculares y denuncias ciudadanas.

El dataset contiene diversos registros de distintos tipos de crimines, en este estudio debe centrarse en los siguientes:

- **Delitos (Felonies)**
- **Faltas (Misdemeanors)**
- **Violaciones (Violations)**

Esta categorización puede ser encontrada en la variable "LAW_CAT_CD".

Los barrios de estudio son:

- **Staten Island**
- **Brooklyn**
- **Queens**
- **Manhattan**
- **Bronx**

Requerimientos

1. Seleccione al menos dos barrios de la ciudad y obtenga estimaciones de la cantidad de delitos, por tipo, ocurridos en la ciudad de NY semana a semana usando como conjunto de entrenamiento toda la historia de la data hasta el 2019 y como conjunto de pruebas el año 2020 completo. Compare distintas aproximaciones de modelamiento utilizando las métricas y modelos vistos en clases aunque se permite también usar métricas/modelos no discutidos (siempre y cuando se utilicen de manera correcta).
2. Deberá almacenar la información en una base de datos de Postgres con el esquema de datos que usted prefiera.
3. Utilizando Apache Airflow, cree un DAG que contenga al menos las siguientes tasks:
    a. **Data Extraction:** Extraerá la información de la base de datos de Postgres.
    b. **Data Processing:** Procesará la información extraída en la etapa anterior realizando todas la tareas de ingeniería de atributos que usted estime conveniente para sus modelos.
    c. **Model Training:** Utilizando la información generada en el punto anterior esta tasa debe ajustar los modelos a la data histórica con los hiperparámetros y configuración que usted haya estimado conveniente de su experimentación previa, una vez finalizado el entrenamiento debe almacenar el modelo donde usted estime conveniente. Este paso se debe realizar solo si no existe un modelo previamente entrenado, en cuyo caso se debe omitir este paso y pasar directamente al siguiente.
    d. **Model Inference:** Utilizando un modelo ya entrenado realice inferencias sobre el conjunto de pruebas y estas sean guardadas en una tabla de postgres.
4. Genere un informe breve donde se explique la metodología que utilizó para realizar los modelos justificando las decisiones tomadas a partir de lo observado en la experimentación. Explique también la infraestructura y DAG generado.

Consideraciones

- Se le recomienda primero realizar una etapa de experimentación sobre los datos en jupyter Notebook para poder determinar la mejor configuración de sus modelos de series de tiempo, una vez que haya obtenido un modelo con resultados satisfactorios incluya esta configuración en la tasa de Model Training.
- Sus modelo debe hacer predicciones nivel de semana, es decir, debe ser capaz de entregar para cada semana del 2020 disponible en los datos la cantidad de delitos de cada tipo que ocurrirán por barrio.
- Note que no se especificó la forma en la que debe pasar la información de una task a la siguiente, esto queda a criterio de usted ya que hay muchas formas de hacerlo, estudie e investigue cual es la que más le conviene para su caso o considera que es correcta. Lo mismo para el modelo en si.
- El entregable subido a Aula debe contener al menos el Notebook con los experimentos realizados, al menos un archivo .py con la definición del DAG/tasks, el informe en formato PDF, un dump de la base de datos en formato .sql con las tablas que haya utilizado ya pobladas junto con un archivo en formato .txt o .json con la información necesaria para conectarse a la base de datos y, en caso de ser necesario, el modelo entrenado.
- Puesto que le pide usar una base de datos para almacenar información durante las tasks necesitará hacer la conexión desde python a Postgres: Queda prohibido incluir información sensible sobre la conexión al motor en los archivos de código .py, se le sugiere crear un archivo de configuración en formato .json (o el que le guste) y que este sea leído simplemente desde el código, demás está decir que si usa alguna herramienta de versionamiento de código ese archivo no debería ser subido aunque para efectos de la tarea súbalo junto con el resto de los entrañables. Entregar código con información sensible sobre las conexiones al motor incurrirá en descuentos de puntaje.
- Algo que le será útil aunque no es obligación es trabajar con Docker para que los integrantes del grupo puedan trabajar sobre entornos equivalentes.
- **Este proyecto puede ser realizado en grupos de máximo tres personas.**