Proyecto Big Data: Análisis de Clima y Precios de Productos Agrícolas en Nepal

Autor: Marco Ferraro
Profesor Guía: Luis Alexander Calvo

Introducción

Este proyecto tiene como objetivo principal analizar y correlacionar dos conjuntos de datos distintos pero relevantes para el contexto nepalí. El primer conjunto proporciona información histórica sobre la temperatura diaria en diferentes distritos de Nepal. Por otro lado, el segundo conjunto ofrece un registro temporal de los precios de una variedad de vegetales y frutas en los mercados de Nepal.

Conjuntos de Datos Utilizados

- **Datos Climáticos Diarios por Distrito**: Este conjunto de datos proporciona registros históricos de temperatura diaria en distintos distritos de Nepal. Puede accederse a este conjunto de datos desde OpenDataNepal.
- **Precios Temporales de Vegetales y Frutas**: Este conjunto de datos ofrece información sobre los precios de varios vegetales y frutas en los mercados nepalíes. Los datos están disponibles en Kaggle.

Objetivos del Proyecto

1. Integrar y combinar estos dos conjuntos de datos utilizando la columna de fecha como clave principal.
2. Realizar transformaciones de datos para prepararlos adecuadamente para su almacenamiento en un contenedor PostgreSQL.
3. Implementar un contenedor PostgreSQL para almacenar los datos transformados.
4. Desarrollar un cuaderno Jupyter que permita montar modelos predictivos y analizar el rendimiento de las predicciones basándose en la información combinada.

Metodología

Transformación de Datos y Almacenamiento

1. **Integración de Datos**: Se llevará a cabo un proceso de integración de datos utilizando la columna de fecha como referencia para combinar la información climática y de precios.
2. **Limpieza y Estructuración**: Se aplicarán técnicas de limpieza y estructuración de datos para garantizar la coherencia y calidad de la información.
3. **Preparación para PostgreSQL**: Los datos transformados y limpios se prepararán para su almacenamiento en un contenedor PostgreSQL, definiendo esquemas adecuados y optimizando el rendimiento de las consultas.
4. **Almacenamiento en PostgreSQL**: Una vez preparados, los datos se cargarán en un contenedor PostgreSQL, asegurando la integridad y accesibilidad para su posterior análisis.

5. **Desarrollo de Modelos en Jupyter**: Utilizando un cuaderno Jupyter, se implementarán y evaluarán modelos predictivos basados en los datos almacenados en PostgreSQL. Esto permitirá analizar el rendimiento de los modelos y su aplicabilidad en escenarios reales.

Guia de usuario

Aca se presenta una guía de como utilizar e iniciar los módulos de forma adecuada.

- Inicie Docker Desktop para asegurarse de que el entorno Docker esté listo para su uso.

Inicializacion de Modulos

Módulo de Transformación de Datos: Preparación y Ejecución

Para garantizar una correcta transformación y almacenamiento de los datos, es esencial seguir una serie de pasos estructurados que permitan una implementación eficiente y efectiva. A continuación, se detallan los pasos necesarios para llevar a cabo este proceso:

Instrucciones de Preparación del Entorno

1. **Construcción de la Imagen Docker**:
   - Navegue hasta el directorio raíz del proyecto y ejecute el script `build_image.sh` ubicado en la carpeta `scripts` mediante el siguiente comando:

     ```
     ./scripts/build_image.sh
     ```

2. **Inicialización del Contenedor PostgreSQL**:
   - Acceda a la carpeta `scripts` y luego a la carpeta `postgresql`.
   - Ejecute el script `build_image.sh` para asegurar la correcta construcción de la imagen de PostgreSQL.

3. **Ejecución del Contenedor PostgreSQL**:
   - Permaneciendo en el directorio `scripts`, ejecute el script `run_image.sh` para inicializar el contenedor de PostgreSQL, garantizando así un entorno de base de datos adecuado para el almacenamiento de los datos transformados.

Transformación y Almacenamiento de Datos

Una vez configurado el entorno, proceda con la transformación y almacenamiento de los conjuntos de datos:

4. **Ejecución del Programa de Transformación**:
   - Utilice el comando a continuación para ejecutar el programa principal `main.py`, el cual realizará la integración, transformación y almacenamiento de los datos.

```
python main.py data/climate_data_nepal.csv kalimati_market_data.csv
```

- Este comando tomará como entrada los conjuntos de datos `climate_data_nepal.csv` y `kalimati_market_data.csv`, realizando las operaciones de transformación y almacenamiento en el contenedor PostgreSQL previamente configurado.

Modulo de Machine Learning.

5. **Servidor de Jupyter**

Finalmente, despues de ejecutar el programa, y estando en el mismo ambiente, corremos el comando del script `load_jupyter_notebook.sh` para alzar un servidor de Jupyter. Podemos accesar al este servidor desde un buscador

6. **Pruebas Unitarias:**

- Ejecute pruebas unitarias con `pytest`.

```
pytest test_functions.py -v
```