DataScience-IndividualDeliveryGuide
Agosto 2020 - El puente
INSTRUCTOR: Gabriel Vázquez Torresgabriel@thebridgeschool.es
MAESTRO: Clara Piniella Martinezclara.piniella@thebridgeschool.es

Explicación de la entrega
Esta entrega individual tiene por objeto practicar diferentes conceptos sobre la AED y los API. Además, se presentará el proyecto.

El estudiante debe elegir una asignatura que prefiera. Idóneamente, esta entrega se ampliará con la Unidad 2 y 3 del temario. Esto es, aparte de esta entrega EDA, habrá más entregas enfocadas en Aprendizaje de la Máquina (Unidad 2) y Ciencia de los Datos como Producto (Unidad 3).

Requisitos
Los siguientes requisitos son obligatorios:

1. El proyecto debe dar respuesta a una hipótesis (que se explica a continuación).
2. El estudiante hará una presentación y deberá documentar todos los pasos que realice.
3. El estudiante debe dividir las tareas que tiene que hacer.
4. Es obligatorio que el estudiante utilice trello (u otro relacionado) para manejar las tareas en estado indiferente: TODO, HACER y HECHO. BACKLOG y REVIEW son opcionales.
5. La entrega debe ser enviada antes del 31/08/2020 a las 23:59.
6. La entrega debe ser enviada en un archivo .zip por correo electrónico con esta estructura:

    a. Una carpeta src/ que contiene todo el código fuente.
    b. Una carpeta documentation/ que contiene todos los documentos relacionados con la documentación.
    c. Una carpeta resources/ que contiene otros contenidos útiles (imágenes,...)
    d. Una carpeta src/utils/ que contiene todos los módulos utilizados por el archivo principal.
    e. Un archivo src/main.ipynb que contiene toda la funcionalidad. Este fichero sólo debe contener importaciones, pandas, matplotlib, peticiones,... y llamadas a sus src/utils/*módulos.
    f. Hay, al menos, cuatro módulos dentro de src/utils/ :

        i. "folders_tb.py" que contiene la funcionalidad genérica relacionada con abrir, crear, leer y escribir archivos.
        ii. "visualization_tb.py" que contiene la funcionalidad genérica relacionada con topandas, matplotlib, seaborn y otras bibliotecas centradas en visualizaciones.
        iii) "mining_data_tb.py", que contiene la funcionalidad genérica relacionada con la recopilación de datos, la limpieza de datos y otros métodos (métodos de discusión como el trabajo con múltiples jsons).
        iv. "apis_tb.py" que contiene la funcionalidad genérica relacionada con el trabajo con las API.
        v. Otras que el estudiante necesita.

    g. Un archivo src/api/server.py que contiene la funcionalidad que inicia la FlaskAPI. Hay, al menos, una función GET:

        i. Uno que debe permitir recibir un valor de token_id y, si token_id es igual a S, devolver los jsons que contienen la lógica explicada más abajoOtros, devolver una cadena con un mensaje de error.

            1. S es el DNI del estudiante que comienza con la letra: Ejemplo: "B80070012"

        ii) Otras que son pertinentes para el proyecto.

    h. Los jsons que se devuelven son estos por lo menos:

        i. Al menos, el estudiante debe devolver un json que represente los datos tratados y limpiados.
        ii. Dependiendo de los datos y el problema, el estudiante debe devolver datos interesantes con el objetivo de hacer su programa útil.

Hipótesis
Normalmente, el objetivo de un proyecto de la AED es responder a una pregunta o demostrar un axioma, es decir, dar todas las razones necesarias para explicar por qué la respuesta a la pregunta es específica y refutar o reafirmar un axioma. Un ejemplo de una hipótesis en el proyecto de covid-19 podría ser: Creemos que el estado de alarma de cada país tiene un impacto en la progresión de la infección diaria.

Presentación

Todos los estudiantes deben hacer una presentación sobre su proyecto. El presentador del grupo utilizará un archivo de presentación para explicar todos los pasos del flujo de trabajo con gráficos. La duración de la presentación no será superior a 7 minutos por lo que es realmente importante y necesario explicar los puntos esenciales del trabajo.

Los pasos del proyecto

La idea del proyecto consiste en diferentes pasos:

1. Encontrar el tema: el estudiante debe encontrar el proyecto en sí mismo. Esto es algo que quiere hacer.
2. Encontrar los datos relacionados con el proyecto: investigar dónde puede estar y si es accesible al público.
3. Definir una hipótesis: encontrar algo que pueda concluir con sus datos.
4. Defina los pasos necesarios para demostrar o no su hipótesis.
5. Con la estructura de código definida y usando Python:
    a. Obtener sus datos. Tal vez necesites usar una API, tal vez un archivo. Data Wrangling.
    b. Limpia tus datos. Detectar valores atípicos, valores raros y reemplace los valores NaN si es necesario.
    c. Dibuja todos los gráficos que necesites tanto para entender tus datos como para mostrar los resultados necesarios.
    d. Crear una API que devuelva lo explicado en la sección de requisitos.tal vez te resulte útil hacer más de un punto final.
    e. Explicar por qué de tus gráficos y otros resultados se puede argumentar la conclusión.
6. Documente todos los pasos, comprima los archivos necesarios y envíelos a los correos de los profesores.

NOTA: Haga todos los pasos terminando los requisitos de los criterios.

Los recursos

Con el objetivo de encontrar todos los recursos necesarios, el estudiante puede buscar en toda la red.
Hay páginas en las que puede encontrar tanto buenos ejemplos de proyectos EDA como conjuntos de datos:

-Aquí puedes encontrar millones de ejemplos con millones de conjuntos de datos. Hay diferentes partes donde se puede aprender de los novatos o de los expertos.

-Búsqueda de conjuntos de datos en Google: aquí puedes encontrar millones de conjuntos de datos. Es una buena página si quieres encontrar los datos que necesitas.

-GoogleApis: aquí tienes muchos apis de diferentes temas para obtener datos.

-Páginas de ayuntamientos, páginas de estadísticas y miles de APIs que puedes encontrar en Internet.

Si el estudiante no tiene ninguna inspiración, entonces podemos recomendarle las próximas asignaturas de la EDA:

-Analizar cómo la situación de la pandemia ha cambiado las vidas de algunos sujetos.

-Analizar los tweets para determinar si algún evento cambia las tendencias.

-Analizar los datos de las películas para determinar si hay más actrices o películas románticas.

-Analizar los datos deportivos para determinar si Messi, Lebron James o Fernando Alonso son los mejores en sus deportes.

-Analizando los datos de enfermedades para concluir si hay relación entre los síntomas y el número de muertes (u otra relación).

-Analizar los datos climáticos para concluir si el cambio climático es real.

-Analizar los datos de los videos para concluir si los videos divertidos tienen más audiencia.

Criterios de evaluación

Para esta entrega, hay diferentes opciones de entrega. Cada estudiante debe elegir qué entrega quiere hacer. C es el requisito mínimo para esta entrega. Hay una jerarquía en las opciones: B→ B→ A→ A+*

No está permitido hacerlo:
-B sin C
-A sin B y C
-A+ sin A, B y C

Opción C

Aparte de todos los requisitos que se escriben en la sección de requisitos, hay los siguientes ejercicios obligatorios:

1.Documentar todos los pasos. 2.Estructurar su código para mantenerlo limpio usando buenas prácticas.
2.Recopilar los datos. Intenta hacer cada llamada, recoge los últimos datos actualizados.
3.Determine y explique si los datos están limpios. Si no, entonces límpialos.
4.Crea una API que devuelva un Json con la lógica explicada. El servidor del frasco debe ejecutarse ejecutando el archivo src/api/server.py.
5.Mostrar diferentes tendencias para cada columna de su conjunto de datos.
6.Representar, en un gráfico circular, el tiempo necesario para cada punto de la sección Los pasos del proyecto.
7.Responda a las preguntas:
    a.¿Fue posible demostrar la hipótesis? ¿Por qué?
    b.¿Qué puede concluir sobre su estudio de datos?
    c.¿Qué cambiaría si tuviera que hacer otro proyecto EDA?
    d.¿Qué aprendes haciendo este proyecto?

Opción B

1.Mostrar el histograma de cada columna de su conjunto de datos con bins=5. ¿Cómo se pintan los rangos?
2.¿Cuáles son las columnas con mayor correlación? 3.Dibuja la matriz de correlación.
3.Usa las funciones de Matplotlib para mostrar todos los gráficos. No con los pandas directamente.

Opción A

1.Investigación para guardar cada parcela en archivos locales.
2.Usar módulos de distribución para cada funcionalidad. Los cuadernos de jupyter no deben tener ningún bucle ni funciones. Sólo debe tener las iniciales "imports" y la llamada a las funciones necesarias.
3.Aparte de matplotlib, usa seaborn para mostrar las gráficas.
4.Responde a las preguntas:
    a.¿Hay valores atípicos o algunos datos raros?
    b.¿Cuáles son las columnas que tienen más valores repetidos?

Opción A+

Hay diferentes A+. Puedes hacer los que quieras:
1.Crear una solicitud de extracción para todo el proyecto.
2.¿Cómo puedes poner el servidor de la petaca con una IP pública sin Heroku? se da cuenta de que la petaca inicia el servidor en una red privada por defecto (localhost)
3.¿Cómo puedes poner tu servidor de frascos con una URL pública sin Heroku? 4.¿Cómo puedes poner tu servidor de frascos con una IP pública con Heroku?
5.¿Cómo puedes poner tu servidor de matraces con una URL pública con Heroku?
6.¿Hay más urls de donde recoger sus datos?. Explica por qué. Si es así, entonces recogedla y fusionadla con vuestros datos.
7.Con el fin de practicar OOP y conceptos de ingeniería/arquitectura de incomputación, definir todas las funciones dentro de las clases y hacer que el programa funcione con ellas. Después de eso, usa un programa para crear el diagrama de clase.
8.Usando tu propio API url, usa web scraping para obtener el json y mostrar los datos.