Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

Actividad 1:

Exploración de Datos en Python

Aplicaciones en Ciencia de Datos e Inteligencia Artificial

Profesor : Francisco Pérez Galarce.
Ayudante : Yesenia Salinas

Fecha : 29 de noviembre de 2024

1 Introducción

Python se ha posicionado como el lenguaje de programación más utilizado para el desarrollo de proyectos de Ciencia de Datos e Inteligencia Artificial. En este lenguaje se ha desarrollado un conjunto de librerías altamente especializadas para las distintas etapas o tareas dentro de un proyecto en esta área, tales como `Pandas`[a], `Numpy`[b], `Matplotlib`[c], `Pytorch`[d], `Sklearn`[e] y `Scipy`[f]. En esta actividad, usted explorará funcionalidades relevantes usando este conjunto de librerías.

2 Instrucciones de la actividad

2.1 Lectura y análisis exploratorio de datos

- Abrir entorno de programación, de preferencia utilizar Visual Studio Code, Google Colab[g] o Jupyter Notebook.
- Importe (e instale en caso de ser necesario) librería `Pandas`.
- Cargar la base de datos de nombre "ejemplo_data.csv". En esta parte recomendamos explorar las diferentes opciones de *read* que tiene disponible la librería `Pandas`, identificando los argumentos disponibles en cada una de ellas.
- Identifique los tipos de varibles que hay disponibles en la base de datos (df.types o df.info()).
- Utilizando la función astype transforme el atributo "ID" a entero y el atributo "Activo" a booleano. Vuelva a consultar el estado de las variables.

[a] https://pandas.pydata.org/
[b] https://numpy.org/
[c] https://matplotlib.org/
[d] https://pytorch.org/
[e] https://pypi.org/project/scikit-learn/
[f] https://scipy.org/
[g] https://colab.research.google.com/notebooks

- Convierta el atributo "unidades" a entero y "2016" a flotante.

2.2 Lectura y análisis exploratorio de datos 2

- Cargar la base de datos de nombre "ecommerce_data.csv". En esta parte recomendamos explorar las diferentes opciones de *read* que tiene disponible la librería `Pandas`, identificando los argumentos disponibles en cada una de ellas.
- Identifique los tipos de varibles que hay disponibles en la base de datos (df.types o df.info()).
- Utilizando la función astype transforme el atributo "InvoiceNo" a entero y el atributo "Description" a string. Vuelva a consultar el estado de las variables.
- Convierta el atributo "Quantity" a entero y "UnitPrice" a flotante.
- La columna "InvoiceDate" contiene un string que representa "fecha-hora", separe la columna en dos columnas que representen cada atributo por separado.
- Añada una nueva columna que represente el monto total para cada boleta.
- Exporte la base de datos procesada en formato ".csv"
- Exploración de funciones *groupby*, *sort_values*, *set_index*, *sample*, *pivot*, *reset_index* y *merge*.

2.3 Estadísticas descriptivas

- Crear un diccionario con 50 datos que contenga al menos tres atributos continuos.
- Transforme dicho diccionario a un dataFrame de `Pandas`.
- Obtenga estadísticas descriptivas de tendencia central.
- Obtenga estadísticas descriptivas de dispersión.

2.4 Transformación e imputación de datos

- Importe la librería sklearn.
- Cargar las bases de datos de nombre "ratings_data.csv" y "books_data.csv".
- Utilizando "ratings_data.csv" genere un diagnóstico de números perdidos. Luego impute los valores de acuerdo a la media y de acuerdo a otro criterio seleccionado por usted. Explore las opciones de imputación del método *fillna()* de `Pandas`.
- En "ratings_data.csv genere una nueva variable que represente el promedio de rating para cada ISBN distinto.
- Utilizando la columna ISBN de cada base de datos consolídelas en una sola base de datos con todos los atributos de "books_data.csv"
- Exporte la base de datos consolidada.

2.5 Entrega

- La actividad deberá entregarse en un archivo jupyter notebook (.ipynb) y subirse a un repositorio en `Github`[h].
- Dicho repositorio debe tener como nombre *apellido-nombre*.
- Dentro de este repositorio usted deberá subir todas las actividades que se realicen durante el curso. Por ejemplo, para la actividad 1 usted deberá subir un archivo jupyter notebook con nombre actividad1.ipynb. El repositorio será evaluado al final del curso, los estudiantes pueden mejorar todas las actividades durante el desarrollo del curso, siendo estas evaluadas después de la última clase.
- Algunas actividades serán evaluadas de acuerdo al trabajo realizado durante la clase, en esos casos se solicitará adicionalmente que las suban a la plataforma del curso.
- La interacción con Github puede ser a través de la web, Github desktop[i] o Command line (CLI)[j]. Sin embargo, es altamente recomendable que usen Github desktop o Command line (CLI).

3 Otras recomendaciones

- Crear un ambiente en conda específico para ejecutar las actividades del curso.
- Priorizar la programación en Visual studio code, e instalar el complemento Live Share[k]. Este complemento ayudará en las actividades grupales y apoyo por parte del equipo docente en la resolución de dudas de programación.

[h] https://github.com/
[i] https://desktop.github.com/download/
[j] https://docs.github.com/es
[k] https://marketplace.visualstudio.com/items?itemName=MS-vsliveshare.vsliveshare