Práctica 1
Preparación y Visualización de Datos

Objetivo

Desarrollar un Jupyter-Notebook por entregable (3 en total) que permitan explicar diferentes hipótesis a partir de la preparación y visualización de varios datasets.

Herramientas

- Lenguaje de programación: Python
- Librerías: Numpy, pandas, scikit-learn, matplotlib
- Entorno de gestion de librerías: Anaconda
- Editor: Jupyter

Información de la entrega

La entrega se realizará a través de la tarea LAB1 disponible en la página Canvas de la asignatura.
Consistirá en un fichero comprimido (.zip,.tar.gz) con nombre LAB01-GRUPOxx.zip que contendrá:

1- Un Jupyter-Notebook por cada entregable (archivos con extensión .ipynb)
2- La memoria del laboratorio se entregará integrada en el Notebook de manera que explique y complemente el código entregado
3- El código entregado tiene que ser funcional, correcto y completo.

Las entregas que no se ajusten exactamente a este formato NO SE EVALUARÁN.

Rúbrica

Código

El valor de cada entregable para la nota final de la práctica se indica en el enunciado, así como el valor de cada uno de los apartados.
Todos los aspectos de programación se dan por supuestos.
El código debe ser:

- Funcional: debe ejecutar sin errores y el resultado debe ser el esperable
- Original: el código no puede ser una copia de trabajos publicados en Internet o de otros compañeros. Grupos con código igual serán suspendidos.
- No redundante: se penalizará el código que no sea útil o redundante
- Comentado: es obligatorio incluir comentarios en el código, en su justa medida
- Gráficas: deben incluir todos los datos que sean necesarios

Memoria

La memoria estará incluida en los Jupyter-Notebook que se entreguen de manera que complementen el código entregado. La redacción debe ser clara y correcta ortográfica y gramaticalmente. Debe incluir la justificación de cada paso que se realice para la resolución de los problemas planteados.

Entregable 1 – Limpieza de datos

Este entregable vale 1.5 puntos de la nota final de la práctica 1.

El primer paso cuando nos enfrentamos a un conjunto de datos nuevos es hacer una limpieza para descartar registros ruidosos.

Para practicar esto, vamos a trabajar con un dataset que recoge datos de pacientes con hepatitis. Toda la información sobre el dataset lo encontrarás en Canvas. El archivo hepatitis.zip contiene el propio dataset hepatitis_data.csv. También encontrarás el archivo hepatitis.info con la descripción de los campos que componen el dataset y toda la información adicional necesaria.

1.1- (1 punto) Descripción y limpieza de los datos.

Carga los datos en un DataFrame. Detecta y elimina los registros redundantes o con *missing values*. Recuerda que los *missing values* están marcados con el carácter '?'. ¿Cuántos registros has eliminado de cada tipo? ¿Qué campos tienen registros con '*missing values*'? Presenta un gráfico que muestre el número de missing values por atributo.

1.2-(0.5 puntos) Estudio preliminar de los datos.

Cómo se distribuyen los pacientes que viven (class=2) de los que mueren (class=1) en el dataset. Presenta un gráfico de tarta que muestre esta distribución de la clase *'class'* en los datos.

Entregable 2 – Netflix

Este entregable vale 4.5 puntos de la nota final de la práctica 1.

Netflix es una plataforma de streaming que ofrece contenido audiovisual a nivel mundial. Aunque en sus inicios ofrecía más películas, en los últimos años parece que ha habido un cambio de tendencia y la oferta de series ha tomado más fuerza en su catálogo. Además, los datos también nos pueden dar una idea del crecimiento de la plataforma tanto en contenidos como en importancia y repercusión.

Para eso, trabajaremos sobre el dataset disponible en Canvas en el archivo netflix.zip. Este archivo contiene el propio dataset netflix.csv y su descripción en netflix.info.

2.1- Confirma o desmiente la hipótesis de que las series han desbancado a las películas en la oferta de Netflix.

2.1.1- (1 punto) Encuentra el número de películas vs número de series disponibles totales ¿Qué porcentaje del contenido representa cada categoría? Muéstralo en un gráfico justificando la elección del tipo de gráfico.

2.1.2- (1 punto) Estudia el número de películas vs número de series disponibles por año de publicación en Netflix. Para eso, crea un gráfico que muestre la evolución temporal (por años) de la cantidad de series y de películas disponibles en la plataforma. Justifica el gráfico elegido y explica la tendencia que observas en los datos. ¿Qué conclusión puedes extraer?

2.2- (0.5 puntos) Estudia la viabilidad de la plataforma Netflix ¿se sigue añadiendo contenido o hay una tendencia a la baja? Es decir, ¿la plataforma sigue creciendo o se pueden apreciar signos de desaceleración?

Para esto muestra la cantidad de contenido añadido a la plataforma por años. Justifica el gráfico elegido y explica la tendencia observada.

2.3- Estudio del momento de aparición del contenido en Netflix.

2.3.1- (1 punto) ¿Cuál es el retraso medio en publicar una película en Netflix? ¿Y para una serie? ¿hay variación en este retraso a lo largo de los años? Visualiza los datos utilizando los gráficos más adecuados justificando su elección.

¿Crees que este reflejo de la actividad de la plataforma es un indicativo de su viabilidad y proyección? Justifica la respuesta preferiblemente apoyándote en los datos.

2.3.2- (1 punto) ¿Hay un momento preferido en el año para publicar un contenido? Es decir, ¿hay algún momento del año en el que se ponga más contenido disponible? Para averiguar esto, estudia la media de contenido publicado para cada mes durante el tiempo de vida de Netflix. ¿Hay diferencias si se trata de una película o de una serie? Utiliza de manera justificada los gráficos correspondientes para respaldar tus respuestas.

Entregable 3 – Principal Component Analysis (PCA)

Este entregable vale 4 puntos de la nota final de la práctica 1.

Cuando se trata de trabajar con datos reales nos encontramos que la mayoría de las veces hay que trabajar con datos de dimensionalidad muy alta. Tener que trabajar con tantas dimensiones hace difícil entender los datos y además hace necesarios más recursos computacionales (memoria, etc.) para procesarlos.
Existen varias técnicas de Machine Learning diseñadas para encontrar una representación "más pequeña" de los datos. Dicho de otra manera, hay métodos para encontrar una representación comprimida de los datos de manera que las reconstrucciones de los datos originales a partir de esta representación sean lo más parecidas posible a los datos originales. En este entregable vamos a trabajar uno de estos métodos, el Principal Component Analysis (PCA), aplicándolo a un conjunto de datos para obtener su proyección/transformación a un espacio de 2 o 3 dimensiones.

En particular, vamos a trabajar con un dataset que recoge las características de varios futbolistas para el videojuego FIFA19. El dataset lo encontraréis como en Canvas en el archivo fifa19.zip. Este archivo contiene el dataset como fifa19.csv, además de una descripción de los diferentes campos en fifa19.info.

3.1 – (0.5 puntos) Estandarización y normalización de datos.
Para evitar que las diferencias de rango en los datos supongan un problema a la hora de procesar la información, el primer paso es estandarizar y normalizar los datos. Usad el método *StandardScaler* de la librería *scikit-learn*.

3.2 – (2 puntos) Autovalores y Autovectores
El siguiente paso es obtener los autovalores y autovectores que nos permitan caracterizar el espacio de llegada de la proyección donde vamos a reducir la dimensión de nuestros datos, estudiando cuántos componentes (dimensiones) necesitamos para representar los datos iniciales. Para esto, primero hay que obtener la matriz de covarianza utilizando el método cov de la librería numpy, y luego obtener los autovalores/autovectores de la matriz utilizando el método linalg.eig también de la librería numpy.
Al final de este apartado, tienes que obtener un DataFrame con el porcentaje de varianza y el porcentaje acumulado por cada componente. Además, tienes que explicar qué quieren decir estos datos y cómo se relacionan con los datos.

3.3 – (1 punto) Representar gráficamente los elementos del dataset
Finalmente, queremos representar gráficamente los elementos de nuestro dataset utilizando la descomposición que hemos calculado en el apartado 3.2. Obtén un diagrama de dispersión en 2 dimensiones y comenta/interpreta el resultado. Es necesario que el diagrama contenga toda la información necesaria. Además, tienes que interpretar la información que proporciona el eje X y el eje Y. Para terminar, elige al menos 4 individuos ¿qué puedes decir de ellos después del análisis que has realizado?

3.4- (0.5 punto) Repite los pasos 3.2 y 3.3 utilizando las funcionalidades disponibles en la librería scikit-learn. Compara los resultados y coméntalos. ¿Hay alguna diferencia notable entre las dos implementaciones?