Data Exploration

Objective: Those exercises are to familiarize yourself with the manipulation of a complex dataset, having multiple types of features. We will use python. I recommend working with notebooks. You can work either by installing python on your computer or using google colab. If you are not familiar with pandas library, here is a short introduction: https://colab.research.google.com/github/Yquetzal/Teaching_notebooks/blob/main/Pandas_hands_on.ipynb

1. Fundamentals

1.1 Loading the data

- Download the dataset `movies_metadata.csv` found on the class website.
- Using pandas, load the file and check its content using for instance `.head(2)`

1.2 Column types

- Using `df.info()`, check the type that pandas assigned automatically to each column.
- One column has not been converted to the expected numerical type. Try to force conversion using `pd.to_numeric`. An error should occur. This is because a row is problematic. You can see the option `errors="coerce"` to ignore those errors (nb.: you will certainly introduce new eros doing so, but let's start with a quick and dirty approach)

1.3 Data quality

- Compute the classic descriptors of the `budget` column using pandas' `describe` function. Check the mean, std, percentile, and extreme values…
- You should observe suspicious values, too low and too high. Keeping false values in the dataset would bias the results. We will replace them later with `np.nan`, but we need to explore the data to know which values to remove.

1.4 Missing values

- Check the number of missing values in the color column. This value was already present when you did the `df.info`, but you can also use `df[col].isna().sum()` to compute it for one column.

- For columns with few missing values, remove the corresponding rows. You can use the dropna() function. It has a subset parameter to take only some columns into account. For columns with many missing values, keep them for now.

1.5 Data exploration

- Using a plotting library (easiest: seaborn, interactive, etc), plot the distribution of the budget variable using a histogram. You can directly use pandas plotting tools (df[col].plot.hist()). Vary the number of bins using the bin parameter and observe the changes.
- Do the same with other numerical values. Which ones are , visually, following a bell curved, and which ones are not?

1.6 Correlation, Covariance

- For the following questions, we will focus on the revenue, runtime, vote_average and vote_count variables. It might be easier to create a new dataframe with only those variables. You can use df[['col1', 'col2']]. Keep only lines in which all values are not NaN.
- Compute the variance, the standard deviation and the mean average deviation.
- Compute the covariance matrix, e.g., with cov function from pandas. Check the relation with the variance. Can you say something about the other values in this matrix?
- Compute the correlation coefficient between those variables, for instance the corr function from pandas. By default, it uses the Pearson correlation coefficient. Check how it is computed from the covariance matrix. Interpret those coefficients.
- Remember that the assumption made when computing Pearson correlation is that the relation between the two variables is linear. Use sns.pairplot to have a look at the relation between those variables.

2. Advanced

- On the class page, you can find a dataset from the website. Download it.
- Apply a similar analysis on the dataset.