## **Netflix Content Exploration and Recommendation Assignment**

This assignment utilizes the netflix\_titles.csv dataset to explore content trends, user preferences (through ratings), and build a simple recommendation model.

**Learning Objectives:**

* Data exploration and cleaning with Pandas
* Data visualization with Matplotlib
* Building a simple linear regression model with scikit-learn

**Task 1: Data Exploration and Cleaning**

1. **Load the Data:**
   * Use Pandas library to read the CSV file containing the Netflix Titles dataset.
   * Assign the loaded data to a Pandas DataFrame object.
2. **Investigate the Data:**
   * **Missing Values:**
     + Identify columns with missing values using df.isnull().sum().
     + Analyze the number of missing values and their distribution across columns.
     + Decide on an appropriate strategy to handle missing values (e.g., imputation, removal).
   * **Data Types:**
     + Check the data types of each column using df.dtypes.
     + Identify any inconsistencies (e.g., dates stored as strings).
     + Convert columns to appropriate data types if necessary (e.g., converting "date\_added" to datetime format).
   * **Descriptive Statistics:**
     + Calculate descriptive statistics like mean, median, and standard deviation for numerical columns using df.describe().
     + Analyze these statistics to understand the central tendencies and spread of the data.
   * **Categorical Data Analysis:**
     + Identify categorical columns (e.g., "type", "genre").
     + Use methods like df['type'].value\_counts() to explore the distribution of categories within these columns.
     + Visualize the distribution of categorical data using bar charts or pie charts.
3. **Clean the Data:**
   * Implement your chosen strategy to handle missing values (e.g., filling missing values with appropriate values or removing rows with too many missing entries).
   * Address any identified inconsistencies in data types.

**Task 2: Data Visualization**

1. **Release Year Distribution:**
   * Create a histogram or density plot to visualize the distribution of the "release\_year" column using Matplotlib.
   * Analyze the plot to identify trends over time (e.g., are there more recent additions to the platform?).
2. **Rating by Type:**
   * Calculate the average rating for "Movie" and "TV Show" categories.
   * Create a bar chart to compare the average ratings between these categories.
   * Interpret the chart and discuss if there's a preference for certain content types based on ratings.
3. **Duration vs Rating:**
   * Create a scatter plot to visualize the relationship between "duration" and "rating".
   * Analyze the plot to identify any correlation between the length of content and user ratings.
4. **Genre Exploration:**
   * Utilize techniques like df['listed\_in'].explode().value\_counts() to identify the most frequent genres in the dataset.
   * Create a bar chart or word cloud to visualize the distribution of genres.
   * Discuss which genres are most prominent on Netflix.

**Task 3: Building a Simple Recommendation Model**

1. **Identify Features:**
   * Based on your data exploration, choose features you believe might influence user ratings (e.g., release year, duration, genre). Consider including numerical features and categorical features after encoding them (e.g., one-hot encoding).
2. **Preprocess the Data:**
   * Split the data into separate DataFrames for features (X) and target variable (ratings - y).
   * Handle categorical features using encoding techniques (e.g., one-hot encoding with scikit-learn's OneHotEncoder).
3. **Train-Test Split:**
   * Split the preprocessed data into training and testing sets using scikit-learn's train\_test\_split function.
   * This will be used to train the model on the training set and evaluate its performance on the unseen testing set.
4. **Build a Linear Regression Model:**
   * Use scikit-learn's LinearRegression class to create a linear regression model.
   * Train the model on the training data.
5. **Evaluate Model Performance:**
   * Predict ratings on the testing set using the trained model.
   * Calculate metrics like R-squared or Mean Squared Error (MSE) to evaluate the model's accuracy in predicting ratings.
   * Interpret your results

**Task 4: Report and Discussion**

1. **Report:**
   * Create a well-structured report summarizing your findings from data exploration and visualization tasks.
   * Include visualizations and relevant statistics to support your observations.
   * Discuss the limitations of your chosen features and potential future considerations for building a more robust model.
2. **Model Discussion:**
   * Explain the rationale behind your chosen features for the recommendation model. Discuss why you believe these features might influence user ratings.
   * Analyze the performance of your model based on the evaluation metrics.
   * Discuss potential reasons for any limitations in the model's accuracy and how it could be improved.

**Deliverables:**

* Jupyter Notebook or Python script documenting your code, data cleaning steps, analysis, and visualizations.
* A well-structured report summarizing your findings and insights.
* Create a folder including both files and push it to your homework repo following the naming convention for homework 9.

**Bonus (Optional)**

1. **Content-Based Recommendation:**
   * Explore building a recommendation system based on content similarity. This could involve techniques like using cosine similarity between genres to recommend similar content to users.
2. **Model Performance for Subsets:**
   * Investigate how your model performs for different genres or release years. You can split the data by genre or release year and evaluate the model's performance on these subsets.