A) Project Name - Netflix Movies And TV Shows Clustering: This is the name given to the project focused on analyzing and clustering Netflix movies and TV shows based on their characteristics to provide personalized recommendations to users.

B) Project Summary: The project involves using data analysis and machine learning techniques to cluster Netflix movies and TV shows into similar categories. By analyzing various characteristics such as genre, cast, and plot, Netflix aims to identify patterns and similarities among titles. This clustering process helps in improving user engagement and satisfaction by providing personalized content recommendations based on their viewing history and preferences. Additionally, the project helps Netflix make data-driven decisions regarding content production and licensing.

Github link

Problem statement: The ultimate goal of this clustering is to improve the Netflix user experience by providing personalised content recommendations based on viewing history and preferences.

General guideline :

Here we start importing libraries.

C) Import Libraries: This section includes the necessary libraries and packages imported for data analysis and visualization, natural language processing (NLP), clustering algorithms, and warnings filtering.

D) Dataset Loading: The project loads the dataset from a Google Drive location. Data is a variable where we store our csv file with the help of panda .The dataset file used is 'NETFLIX MOVIES AND TV SHOWS CLUSTERING.csv'.

.sample is used to display the random rows .

Data.shape is used to display the no of rows and col.The dataset contains 7787 rows and 12 columns

E) Dataset Information:is a dataset Summary. Each row represents a movie or TV show. The columns provide information such as the show ID, type (movie or TV show), title, director, cast, country, date added, release year, rating, duration, listed genres, and description. The data types range from objects (strings) to an integer for the release year.

F) Duplicate Values: The information does not mention any duplicate values in the dataset.

G) Missing Values/Null Values: The table shows the number of missing values (null values) in each column of the dataset. For example, the 'director' column has 2389 missing values, 'cast' has 718 missing values, 'country' has 507 missing values, and so on. The 'show\_id', 'type', 'title', 'release\_year', and 'duration' columns have no missing values.

Msno.matrix and msno.baris used to Visualizing the missing data.

### What did you know about your dataset?

We have already discuss about this.

## ***Understanding Variable or column name***

Data.columns display the coln name

data.describe(include='all') The describe method is used to generate descriptive statistics of the data within the DataFrame.

It would provide summary statistics for each column, including count, mean, standard deviation, minimum, quartiles, and maximum values for numeric columns. For non-numeric columns, it would include the count, unique values, top (most frequent) value, and frequency of the top value.

### variables Description this is our variables Description about each column name.

### Unique Values for each variable.

This loop will iterate over each column in data.columns.tolist() and print a statement indicating the number of unique values in that column.

### What all manipulations have you done and insights you found?

We have already discuss about this

H) Data Wrangling:

This section focuses on preparing the dataset for analysis by addressing missing values. The following steps are taken:

1. Converting the 'cast' column into a list: The 'cast' column is converted into a list format, to facilitate further processing or analysis.
2. Checking for missing values: The code snippet data.isnull().sum() is used to determine the number of missing values in each column of the dataset.
3. Next step will be–Filling missing values: The missing values in the 'director', 'cast', and 'country' columns are filled with the value 'Unknown' using the fillna() method. This ensures that these columns have a non-null value for all entries.
4. Dropping unnecessary columns: The 'date\_added' and 'rating' columns are deemed to have lower importance and are dropped from the analysis. This is done using the dropna() method with the subset parameter specifying the columns to consider.
5. Dropping rows with missing values: The dropna() method is used again, this time without specifying any subset, to drop rows that still have missing values after the previous steps.
6. Finally, data.isnull().sum() is used again to check if any missing values remain in the dataset. If the output shows zeros for all columns, it means that missing values have been successfully handled.

By performing these data wrangling steps, the dataset is prepared for further analysis without missing values, allowing for more reliable and accurate insights.

I) Data Visualization, Storytelling & Experimenting with charts:

1. Chart - 1: Bar Plot - Show Count of Movies and TV Shows

* Reason for picking the chart: The bar plot is chosen to visualize the count of movies and TV shows because it effectively displays the comparison between the two categories using vertical bars.
* Insight(s) from the chart: **The majority of the content in the dataset is likely to be movies, as indicated by the higher count in the 'Movies' category compared to the 'TV Shows' category.**
* Chart - 2: Percentage Distribution of Ratings
* Reason for picking the chart: A pie chart is used to represent the percentage distribution of ratings because it provides a visual representation of each rating category's proportion.
* Insight(s) from the chart: The chart displays the distribution of ratings across the dataset..
* Data shows that rating TV-MA is most frequent.

1. Chart - 3: Distribution of Ratings by Type

* Reason for picking the chart: A stacked bar chart is employed to show the distribution of ratings by type because it enables the comparison of ratings within each content type and provides an overview of how ratings are distributed.
* Insight(s) from the chart: The chart showcases the distribution of ratings for both movies and TV shows. Each bar represents a rating category, and the stacked segments within each bar represent the count of movies and TV shows with that rating. It helps to analyze how different ratings are distributed across the two content types.

1. Chart - 4: Trends in the Number of Netflix Releases over the Years

* Reason for picking the chart: A line plot is chosen to depict the trends in the number of Netflix releases over the years as it shows the change in release counts over time.
* Insight(s) from the chart: The line plot demonstrates the variation in the number of Netflix releases across different years. By observing the line's trajectory, we can identify any upward or downward trends in the number of releases. It provides insights into the growth or decline of content production by Netflix over time.
* Chart demonstrates an abrupt increase in the number of Netflix releases from 2000 to 2020, followed by an abrupt decline, which may be related to a pandemic outbreak.

1. Chart - 5: Distribution of the Duration of Movies and TV Shows based on their Ratings

* Reason for picking the chart: Violin plots are used to visualize the distribution of durations across different ratings for both movies and TV shows. It allows for comparing the distributions and identifying any variations based on the ratings.
* Insight(s) from the chart: The violin plots illustrate the distribution of durations for movies and TV shows within each rating category. The width of the violin indicates the density of titles, and the shape shows the distribution. It helps identify any patterns or differences in the duration distribution based on the ratings for movies and TV shows.

1. Chart - 6: Bar Plot - Top 10 Genre in Movies

* Reason for picking the chart: A bar plot is utilized to visualize the top 10 genres in movies because it allows for easy comparison and ranking of the genres based on their counts.
* Insight(s) from the chart: The bar plot presents the top 10 genres in movies based on their frequency in the dataset. Each bar represents a genre, and the height indicates the count of movies in that genre. It helps identify the most popular genres among the movies in the dataset.
* ‘Documentaries’ is the most popular genre in movies.

7) Chart - 7: Top 20 Movies/TV shows by Release Year

- Reason for picking the chart: A horizontal bar plot is chosen to display the top 20 movies and TV shows by release year because it provides a clear comparison of the number of titles released in each year.

- Insight(s) from the chart:

* Maximum number of movies released or added is in the year 2018.
* Maximum number of TV shows released is in the year 2020, as it was covid time period and everyone is at their home, so netflix use this opportunity to give offers and quality content to the audiences that led to more release in movies and TV shows.
* We saw a huge increase in the number of movies and television episodes after 2014.

8) Chart - 8: Check the count of TV shows and movies over the month.

- Reason for picking the chart: A countplot is used to visualize the count of TV shows and movies over the months because it provides a clear comparison of the frequency of releases in different months.

- Insight(s) from the chart: The chart displays the count of TV shows and movies for each month. The bars represent the count of titles, and each hue within the bars represents TV shows and movies. By observing the bars, it is possible to identify any variations in the number of releases across different months and determine if there are any months with a higher concentration of releases.

Maximum number of content is released in the month of December followed by january.

9) Chart - 9: Distribution of Movie/TV shows Durations

- Reason for picking the chart: A histogram is used to visualize the distribution of durations for movies and TV shows because it provides an overview of how the durations are distributed across different values.

- Insight(s) from the chart: The chart showcases the distribution of durations for movies and TV shows. The x-axis represents the duration values, and the y-axis represents the count of titles with each duration. By observing the histogram, it is possible to identify the most common durations and the overall distribution of durations for both movies and TV shows.

The variable named duration follow a distribution which is close to normal distribution, while there is highly skewed distribution for TV shows.

10) Chart - 10: Pie plot showing the percentage of originals and others in movies

- Reason for picking the chart: A pie plot is used to visualize the percentage distribution of originals and other movies because it provides a clear comparison of the proportions.

- Insight(s) from the chart: The chart represents the percentage distribution of movies categorized as "Originals" and "Others." The two slices of the pie chart represent the proportion of movies that fall into each category. By observing the chart, it is possible to understand the relative frequency of original movies compared to other movies in the dataset.

Netflix has over 30% original content which shows that netflix invests heavily in creating and promoting their own original content. This can be seen as part of Netflix's strategy to differentiate themselves from other streaming platforms and to maintain a competitive advantage.

11) Chart - 11: Check the distribution of TV shows and movies across years

- Reason for picking the chart: A line plot is chosen to display the distribution of TV shows and movies across years because it shows the trend and variation in the number of titles released each year.

- Insight(s) from the chart: The line plot illustrates the variation in the number of TV shows and movies released across different years. The x-axis represents the release years, and the y-axis represents the count of titles. By observing the line plot, it is possible to identify any significant changes in the production of TV shows and movies over time.

* The insight that can be drawn is that the number of TV shows and movies on Netflix has increased significantly since 2010.
* The significant increase in the number of TV shows and movies on Netflix since 2010 could reflect the evolution of the streaming industry. As more consumers have shifted away from traditional cable and satellite TV, the demand for on-demand streaming services has increased. Netflix has responded to this trend by investing in its content library and expanding its offerings to attract more subscribers.

12) Chart - 12: Word Cloud

- Reason for picking the chart: A word cloud is used to visually represent the frequency of words in different text features (director, description, title, cast, listed\_in, country) because it provides an intuitive representation of the most common words in each category.

- Insight(s) from the chart: The word cloud provides a visual representation of the most frequently occurring words in different text features. The size of each word in the cloud represents its frequency, with larger words indicating higher occurrence. By observing the word cloud, it is possible to identify the prominent keywords or themes associated with the specific text feature.

13) Chart - 13: Compare the number of movies and TV shows produced by different countries

- Reason for picking the chart: A bar plot with grouped bars is chosen to compare the number of movies and TV shows produced by different countries. This chart type allows for a clear visual comparison of the counts between the two categories (movies and TV shows) within each country.

- Insight(s) from the chart: The chart presents a comparison of the number of movies and TV shows produced by different countries. Each bar represents a country, and the grouped bars within each country represent movies and TV shows. By observing the chart, it is possible to identify countries that have a higher production of either movies or TV shows and understand the overall distribution of content production across different countries.

USA and India are topmost countries in producing content over the platform.

* As both the USA and India have very distinct cultural identities, the high production volume of movies and TV shows from these countries could indicate that their culture and values are widely consumed and popular among audiences worldwide.
* The fact that USA and India are the top producers of movies and TV shows could indicate that these markets are more attractive for investment in the industry.

14) Chart - 14: Correlation Heatmap

- Reason for picking the chart: A heatmap is used to visualize the correlation between the target age ratings and countries. This chart type helps identify any patterns or relationships between the target age ratings of content and the countries where they are produced.

- Insight(s) from the chart: The heatmap showcases the correlation between the target age ratings and countries. Each cell in the heatmap represents the percentage of content in a particular target age rating category produced by a specific country. The color intensity of each cell indicates the strength of the correlation. By observing the heatmap, it is possible to identify countries that produce content targeting different age groups and understand the distribution of content across age categories and countries.

US and UK are closely aligned with their Netflix target ages, means that people in both USA and UK prefer to watch similar type of content.

In similar way US and Canada are also highly aligned with their Netflix target ages.

The code provided demonstrates three hypothesis tests performed on the dataset.

1) Hypothesis Test - Average Durations of Movies and TV Shows:

- Null Hypothesis: The average duration of movies is equal to the average duration of TV shows.

- Alternative Hypothesis: The average duration of movies is different from the average duration of TV shows.

- Method: Independent t-test (two-sample t-test) is used to compare the average durations of movies and TV shows.

- Result: The t-statistic is calculated as 249.8698635138264, and the p-value is obtained as 0.0. Since the p-value is less than the significance level (e.g., 0.05), there is strong evidence to reject the null hypothesis. This suggests that the average durations of movies and TV shows are significantly different.

2) Hypothesis Test - Ratings Distribution in Movies and TV Shows:

- Null Hypothesis: There is no significant difference in the ratings between movies and TV shows.

- Alternative Hypothesis: There is a significant difference in the ratings between movies and TV shows.

- Method: Chi-square test is used to compare the ratings distribution between movies and TV shows.

- Result: The chi-square statistic is calculated as 930.2390474658954, and the p-value is obtained as 1.6546187394507098e-190. Since the p-value is significantly smaller than the chosen significance level, there is strong evidence to reject the null hypothesis. This suggests that there is a significant difference in the ratings between movies and TV shows.

3) Hypothesis Test - Release Years of Movies and TV Shows:

- Null Hypothesis: There is no significant difference in the release years of movies and TV shows.

- Alternative Hypothesis: There is a significant difference in the release years of movies and TV shows.

- Method: Mann-Whitney U test (non-parametric test) is used to compare the release years of movies and TV shows.

- Result: The Mann-Whitney U statistic is calculated as 4619200.5, and the p-value is obtained as 1.8772186034373436e-89. Since the p-value is significantly smaller than the chosen significance level, there is strong evidence to reject the null hypothesis. This suggests that there is a significant difference in the release years of movies and TV shows.

In all three tests, the p-values obtained indicate that there is a statistically significant difference between the compared variables, leading to the rejection of the corresponding null hypotheses.

The code provided demonstrates several steps of text preprocessing and feature engineering.

a) Textual Data Preprocessing:

1) Expand Contraction:

- Ratings categories are assigned to grouped categories using a dictionary called `ratings`.

- The 'rating' column in the 'data' DataFrame is replaced with the corresponding grouped category using the `replace()` function.

2) Lower Casing:

- The 'organized' column in the 'data' DataFrame is converted to lowercase using the `str.lower()` function.

3) Removing Punctuations:

- The `remove\_punctuation()` function is defined to remove punctuations from a text string using the `string.punctuation` module.

- The `translate()` method is applied to the text using a translation table created by `str.maketrans()` to remove the punctuations.

- The 'organized' column in the 'data' DataFrame is updated by applying the `remove\_punctuation()` function to it using the `apply()` function.

4) Removing Stopwords & Removing White spaces:

- The necessary NLTK (Natural Language Toolkit) libraries and modules are imported.

- The `remove\_stopwords()` function is defined to remove stopwords from a text string using the NLTK `stopwords.words('english')` module.

- The 'organized' column in the 'data' DataFrame is updated by applying the `remove\_stopwords()` function to it using the `apply()` function.

5) Text Normalization:

- The `SnowballStemmer` from NLTK is used for stemming the words.

- The `stem\_text()` function is defined to stem each word in a text string using the Snowball stemmer.

- The 'organized' column in the 'data' DataFrame is updated by applying the `stem\_text()` function to it using the `apply()` function.

6) Text Vectorization:

- The 'title' and 'org\_new' columns are selected from the 'data' DataFrame and stored in a new DataFrame called 'new\_df'.

- The TF-IDF vectorization is applied to the 'org\_new' column using the `TfidfVectorizer()` from `sklearn.feature\_extraction.text`.

- The resulting vectorized text data is stored in the variable 'X' using the `fit\_transform()` function.

b) Dimensionality Reduction:

- The `PCA` class from `sklearn.decomposition` is imported.

- The PCA model is fitted on the vectorized text data 'X' using the `fit()` function.

- The explained variance ratio is plotted against the number of components using `plt.plot()` to determine the optimal number of components to retain.

- Based on the plot, a desired level of explained variance is chosen (e.g., 0.95).

- Another PCA model is created with the desired number of components using `PCA(n\_components=0.95)`.

- The PCA model is fitted on the vectorized text data 'X' using the `fit()` function.

- The vectorized text data is transformed to the reduced dimension using the `transform()` function and stored in 'X\_transformed'.

- The shape of 'X\_transformed' is printed to observe the reduced dimensionality.

Overall, the code performs various preprocessing steps such as expanding contractions, lowercasing, removing punctuation, stopwords, and normalizing text. It then vectorizes the text using TF-IDF and reduces the dimensionality using PCA if needed. These steps are commonly employed in text processing and feature engineering to prepare textual data for further analysis or modeling tasks.

Then the following code snippet demonstrates the implementation of various machine learning models and techniques for analyzing and clustering a dataset. Let's break down the code into different sections:

1) ML Model - 1: K-Means Clustering

- The code uses the KElbowVisualizer from the Yellowbrick library to determine the optimal number of clusters using the silhouette score metric.

- It then performs K-Means clustering using the selected number of clusters and visualizes the results using a scatter plot.

- The cluster labels are added to the 'data' dataframe.

- The code also includes a function, func\_select\_Category, that generates word clouds based on cluster labels for the 'listed\_in' and 'description' columns.

2) ML Model - 2: Dendrogram (Hierarchical Clustering)

- The code uses the linkage and dendrogram functions from the scipy.cluster.hierarchy module to perform hierarchical clustering.

- It computes the distances and generates a dendrogram visualization.

3) ML Model - 3: Agglomerative Clustering

- The code performs Agglomerative Clustering with different numbers of clusters.

- It computes the silhouette score for each number of clusters and prints the results.

- It then visualizes the clusters using a scatter plot and adds the cluster labels to the 'data' dataframe.

- Similar to K-Means clustering, the code includes a function, func\_select\_Category\_agg\_clustering, for generating word clouds based on cluster labels.

4) Recommender System

- The code implements a content-based recommender system using TF-IDF (Term Frequency-Inverse Document Frequency) to compute similarity scores between programs.

- It calculates the cosine similarity between the TF-IDF matrix of program descriptions.

- The recommend function takes a program title as input, finds the index of the program in the dataset, and returns a table of recommended movies based on similarity scores.

5) Pickle Serialization

- The code saves the recommend function, tfidf transformer, and tfidf matrix as pickle files for later use.

Overall, this code showcases the implementation of different clustering algorithms (K-Means, Hierarchical, and Agglomerative) and a content-based recommender system using TF-IDF. It also includes visualizations and word cloud generation to analyze and interpret the clustering results.