Project Title:

Netflix Data Analysis Project

Author:

Hadisa Fatima Syed [Role: Data Analysis Enthusiast and Student of BE Software]

Project Start Date:

19-June-2025

Project End Date:

25-June-2025

Objective/Problem Statement:

The goal of this project is to answer the following questions:

1. What is the most frequent Genre of movie released on Netflix?
2. What Genre has highest votes?
3. What movie got highest popularity? What it’s Genre?
4. What movie got lowest popularity? What it’s Genre?
5. Which year has the most filmed movies?

Dataset Description:

* Source: .xlsx file from GitHub repo by ‘The iScale”
* File Name: Netflix Data.xlsx
* Size: 9838 rows x 9 Columns
* Columns Overview: Release\_Date, Title, Overview, Popularity, Vote\_Count, Vote\_Average, Original\_Language, Genre and Poster\_Url

Tools and Libraries Used:

* Libraries: Pandas, Seaborn, Matplotlib, Numpy
* Tools: Google Colab, Google Sheets, MS Excel Spreadsheet

Data Cleaning and Preprocessing:

1. Datatype of dataset’s ‘Release\_Date’ column was converted to **datetime** from Object.
2. Datatype of ‘Vote\_Count’ and ‘Vote\_Average’ column was converted from Object to int and float respectively.
3. There were 8 duplicate values in the dataset, and those rows were dropped to prevent data redundancy.
4. Dropping columns Overview, Poster\_Url and Original\_Language as these columns are not useful for answering the asked questions. Dropping them will help keep focus just on relevant data.
5. In Release\_Date column keeping just the year from the mentioned data, as for answering one of the questions we need just year.
6. Changing the datatype of Release\_Date column one more time, this time from datatime64 to Int64 so we can easily fetch year from it.
7. In the Vote\_Average column replacing the float numbers (ratings) to labels and those labels are Not\_popular, below\_Avg, Average and Popular. Doing this so, it will easier for users to interpret the popularity of the movie because numbers can be a little confusing.
8. Checking total number of movies per each label of Vote\_Average column and dropping not\_popular movies as nobody would be interested in watching them and there are total 2467 not popular movies.
9. For Genre Column changing its datatype to str from Object, and in the Genre column, keeping a single Value per cell by splitting genres of a Movie in a list.
10. Converts the **Genre** column to pandas' category data type for efficient memory usage as it has many repeated string values.

**Data Visualization:**

1. Using **Catplot** for answering first question, as the **Genre column** is in categories.
2. For second question, creating a separate data frame just for **Genre** and **vote count per genre** and then sorting them Genre wise (A-Z). Presenting the Genre with highest votes by using a barplot.
3. Fifth Question, i.e which year has highest movies filmed, is answered by using **histogram**.

**Key Findings:**

1. The **Drama Genre** is the most frequent Genre of movies released on Netflix, with **14.539%** of the total Genres on Netflix.
2. Drama Genre has the highest number of Votes.
3. **Spider-Man: No Way Home** is the movie with **highest Popularity** and its Genres are **Action**, **Adventure** and **Science Fiction**.
4. United States vs Billy Holiday and Threads are the movies with lowest Popularity and their Genres are Music, Drama and History and War, Drama and Science Fiction respectively.
5. In **2020**, Netflix filmed **1636** movies which is the **maximum** number of movies in a year.

**Challenges Faced:**

1. Selecting the suitable chart to appropriately visualize the found insights.
2. Changing datatypes from time to time.
3. Categorizing data.
4. Understanding and making sense where data needs to be categorized.
5. Use of user defined function was new for me in Data Analysis.

**Conclusion:**

This project explored various attributes of Netflix movies and revealed audience preferences and release patterns. It was found that genres like Drama and Action dominate the platform both in quantity and user engagement.