SRM INSTITUTE OF SCIENCE AND TECHNOLOGY, TRICHY

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)
TIRUCHIRAPPALLI

DATA SCIENCE [21CSS303T]

PROJECT REPORT

April 2025

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY, TRICHY

DATA SCIENCE PROJECT REPORT

NAME	N.Nagendra Rakesh AS.Hari Prasath
REG NO	RA2211004050038 RA2211004050047
SEMESTER	VI
PROGRAME	B.TECH ECE
ACADEMIC YEAR	2024-2025

CONTENTS

ABSTRACT

This project explores the application of advanced data science techniques to analyze and interpret content distribution patterns on Netflix, utilizing a publicly available dataset. By strategically leveraging Python’s powerful NumPy library in conjunction with comprehensive data handling tools like Pandas, we conduct an in-depth quantitative analysis aimed at identifying significant trends across content types, popular genres, release timelines, and country-wise content contributions.

The primary objective is not only to uncover historical viewing and publishing patterns but also to generate actionable insights that can support enhancements in personalized user recommendations, inform targeted content acquisition strategies, and anticipate future platform growth directions. By systematically applying statistical computations, exploratory data analysis, and trend modeling, this study underscores the critical role of data-driven strategies in the digital entertainment sector.

In an increasingly competitive streaming landscape, understanding viewer engagement behaviors and content lifecycle dynamics through empirical data analysis becomes essential for maintaining user satisfaction, optimizing content investments, and securing a sustainable competitive advantage. This project exemplifies how the synergy of computational efficiency and statistical rigor can transform raw entertainment datasets into valuable strategic intelligence.

INTRODUCTION

Streaming services like Netflix generate an enormous volume of data every day, encompassing a wide range of content categories, user interactions, and platform activities. Analyzing this data effectively is crucial for uncovering valuable insights into user behavior, viewing preferences, emerging content trends, and overall platform growth strategies. Such insights enable platforms to tailor personalized recommendations, optimize content curation, and make informed decisions about content acquisition and production investments.

In this project, we focus on the Netflix Titles dataset to perform a comprehensive, foundational data science workflow. Starting with initial dataset exploration, we systematically move through stages of data cleaning, preprocessing, and detailed trend analysis using Python's NumPy library. By applying various statistical and numerical methods, we extract meaningful patterns related to content type distribution, genre popularity, release year dynamics, and geographical representation of the titles.

This project not only deepens our understanding of how content is distributed and consumed on Netflix but also highlights the critical role of numerical computing in modern data analysis. NumPy's ability to handle large-scale numerical data efficiently underscores its importance as a core tool in any data scientist's toolkit, especially when working with real-world, high-dimensional datasets like those generated by streaming platforms

OBJECTIVES

Data Collection and Preprocessing:

Gather Netflix dataset containing titles, genres, release years, content types, and regional distribution. Clean and preprocess the data to handle missing values, ensure consistency, and prepare it for analysis.

Exploratory Data Analysis (EDA):

Apply statistical techniques to explore content distribution, genre frequency, country- wise contributions, and release trends. Identify outliers, missing information, and patterns that influence content strategies.

Numerical Analysis Using NumPy:

Perform numerical computations such as calculating mean, median, standard deviation, and correlation using NumPy to understand key content trends. Normalize and standardize data when necessary for accurate insights.

Data Visualization:

Create effective visualizations (such as bar charts, pie charts, histograms, line graphs, and heatmaps) to represent Netflix content trends clearly. Use visual storytelling to identify relationships between content features like genre popularity and year-wise content growth.

IMPORTANT OF DATA VISUALIZATION:

Data visualization plays a crucial role in this project by transforming the complex Netflix Titles dataset into easy-to-understand graphical representations. Since the dataset contains multiple variables, including content type, genre, release year, and country of origin, effective visualization techniques help in identifying distribution patterns, content trends, and anomalies across different categories and time periods.

Visualization makes it easier to observe how the number of movies and TV shows has evolved over time, how popular genres have shifted, and how contributions from different countries have changed. It allows for the rapid detection of trends such as the surge in TV shows after 2018, or the increasing popularity of international movies.

DATA SET DESCRIPTION

The dataset used in this project comprises information on movies and TV shows available on Netflix. It has been collected from publicly accessible databases such as Kaggle and open-source Netflix catalogs. The dataset includes various attributes necessary for analyzing trends related to content type, genre, release year, and geographical distribution across Netflix’s platform.

Key Attributes in the Dataset:

1. show_id: Unique identifier assigned to each show or movie.

2. type: Specifies whether the entry is a Movie or TV Show.

3. title: Title of the movie or TV show.

4. director: Name(s) of the director(s) of the content.

5. cast: Main cast members involved in the production.

6. country: Country or countries where the content was produced.

7. date_added: Date on which the content was made available on Netflix.

8. release_year: The year the content was initially released.

9. rating: Audience rating of the content such as PG, TV-MA, etc.

10. duration: Duration of movies (in minutes) or number of seasons for TV shows.

11. listed_in: Categories or genres the content belongs to.

12. description: Short summary or synopsis of the movie or show.

DATA PROCESSING

In this project, the Netflix Titles dataset is processed and analyzed using data science techniques in Google Colab. The dataset is first uploaded to the Colab environment in CSV format to facilitate easy access and manipulation. Pandas and NumPy libraries are imported to handle data cleaning, numerical processing, and exploratory data analysis.

The CSV file is read into a Pandas DataFrame using the read_csv() function. Initial steps include displaying the first few rows using head() to understand the structure of the dataset and checking for missing values using isnull().sum(). Important columns such as director, cast, country, and date_added are reviewed carefully, and missing entries are handled appropriately based on the context of the analysis.

The date_added column is converted into a proper datetime format using pd.to_datetime() to enable time-based trend analysis. The release_year and rating fields are also verified for consistency. Duplicate entries, if any, are checked and removed to ensure the dataset's integrity.

Basic statistical operations are performed using NumPy to understand distributions across content types, genres, release years, and countries. This includes calculating counts, averages, and identifying extreme values within the dataset. The processed dataset is then used for further visualization to extract meaningful insights related to Netflix's content patterns over time.

OPERATIONS PERFORMED USING THE DATA SET:

1. Type count

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
```

o/p

```
Movie      6131

TV Show    2676
```

2. Most common ratings

```
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
rating_counts = df['rating'].value_counts()
```

o/p

```
TV-MA    3207

TV-14    2160

TV-PG     863

R           799

PG-13     490
```

3. Average release year

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")
```

```
# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')



average_release_year = df['release_year'].mean()
```

o/p
Average Release Year: 2014.18

4. Oldest and newest release year

```
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')


min_release_year = df['release_year'].min()

max_release_year = df['release_year'].max()
```

o/p

```
 oldest: 1925
newest:2021
```

5. Average duration of movies

```
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')


movie_durations = movies['duration'].str.extract('(\d+)').dropna().astype(float)[0]

average_movie_duration = movie_durations.mean()
```

o/p

```
99.58 minutes
```

6. Most common genres

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

genres = df['listed_in'].dropna().str.split(', ').explode()

genre_counts = genres.value_counts()
```

o/p

```
1. International Movies        2752

2. Dramas                    2427

3. Comedies                    1674

4. International TV Shows      1351

5. Documentaries               869

6. Action & Adventure          859

7. TV Dramas                   763

8. Independent Movies          756

9. Children & Family Movies     641

10. Romantic Movies            616
```

7. Top 10 countries with the most content

```
 import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")
```

```
# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

top_countries = df['country'].value_counts().head(10)
```

o/p

```
1. United States     2818

2. India             972

3. United Kingdom     419

4. Japan             245

5. South Korea        199

6. Canada             181

7. Spain             145

8. France             124

9. Mexico             110

10. Egypt             106
```

8. Number of missing values per column

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

missing_values = df.isnull().sum()
```

o/p

```
director     2634

cast         825

country       831

date_added     10
```

9. Most frequent directors

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

top_directors = df['director'].value_counts().head(10)
```

o/p

```
1. Rajiv Chilaka              19

2. Raúl Campos, Jan Suter      18

3. Marcus Raboy                16

4. Suhas Kadav                 16

5. Jay Karas                  14
```

10. Most featured actors/actresses

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['dcast_exploded = df['cast'].dropna().str.split(', ').explode()

top_actors = cast_exploded.value_counts().head(10)

ate_added'] = pd.to_datetime(df['date_added'], errors='coerce')
```

o/p

```
1. Anupam Kher        43

2. Shah Rukh Khan      35

3. Julie Tejwani       33
```

```
4. Naseeruddin Shah    32
5. Takahiro Sakurai    32
```

11. Shows added per year

```
import pandas as pd
import numpy as np
# Load the dataset
df = pd.read_csv("netflix_titles.csv")
# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
shows_per_year_added = df['date_added'].dt.year.value_counts().sort_index()
```

o/p

```
2016: 429
2017: 1188
2018: 1649
2019: 2016
2020: 1879
2021: 1498
```

12. Shows added per month

```
import pandas as pd
import numpy as np
# Load the dataset
df = pd.read_csv("netflix_titles.csv")
# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
shows_per_month_added = df['date_added'].dt.month.value_counts().sort_index()
```

o/p

```
January:   738
```

```
February:  563
March:     742
April:     764
May:       632
```

13. Trend of new releases over the years

```
import pandas as pd
import numpy as np
# Load the dataset
df = pd.read_csv("netflix_titles.csv")
# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
releases_per_year = df.groupby('release_year').size()
```

o/p

```
1925: 1
1942: 2
1943: 3
1944: 3
1945: 4
```

14. All Indian movies

```
import pandas as pd
import numpy as np
# Load the dataset
df = pd.read_csv("netflix_titles.csv")
# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
indian_movies = df[(df['type'] == 'Movie') & (df['country'].str.contains('India', na=False))]
```

o/p

Jeans (1998)

Paranoia (2013)

Angamaly Diaries (2017)

15. Count shows by a specific director

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

rajiv_chilaka_count = df[df['director'] == 'Rajiv Chilaka'].shape[0]
```

o/p

```
19
```

16. Shows with "love" in the title

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

shows_with_love = df[df['title'].str.contains('love', case=False, na=False)]
```

o/p

Love on the Spectrum (2021)

Love Don’t Cost a Thing (2003)

Love in a Puff (2010)

17. Longest movie duration

```
import pandas as pd
import numpy as np
# Load the dataset
df = pd.read_csv("netflix_titles.csv")
# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
max_movie_duration = movie_durations.max()
```

o/p

312 minutes

18. TV shows with more than 3 seasons

```
import pandas as pd
import numpy as np
# Load the dataset
df = pd.read_csv("netflix_titles.csv")
# Convert date_added to datetime
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
tv_shows = df[df['type'] == 'TV Show']
tv_durations = tv_shows['duration'].str.extract('(\d+)').dropna().astype(int)[0]
tv_shows_more_than_3_seasons = tv_shows[tv_durations > 3]
```

o/p

The Great British Baking Show — 9 Seasons

Dear White People — 4 Seasons

Resurrection: Ertugrul — 5 Seasons

19. Standard deviation of movie durations

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

std_movie_duration = movie_durations.std()
```

o/p

```
28.29 minutes
```

20. Year-wise mean movie duration trend

```
import pandas as pd

import numpy as np

# Load the dataset

df = pd.read_csv("netflix_titles.csv")

# Convert date_added to datetime

df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

movies['release_year'] = movies['release_year'].astype(int)

movies['numeric_duration'] = movie_durations

yearly_duration_trend = movies.groupby('release_year')['numeric_duration'].mean()
```

o/p

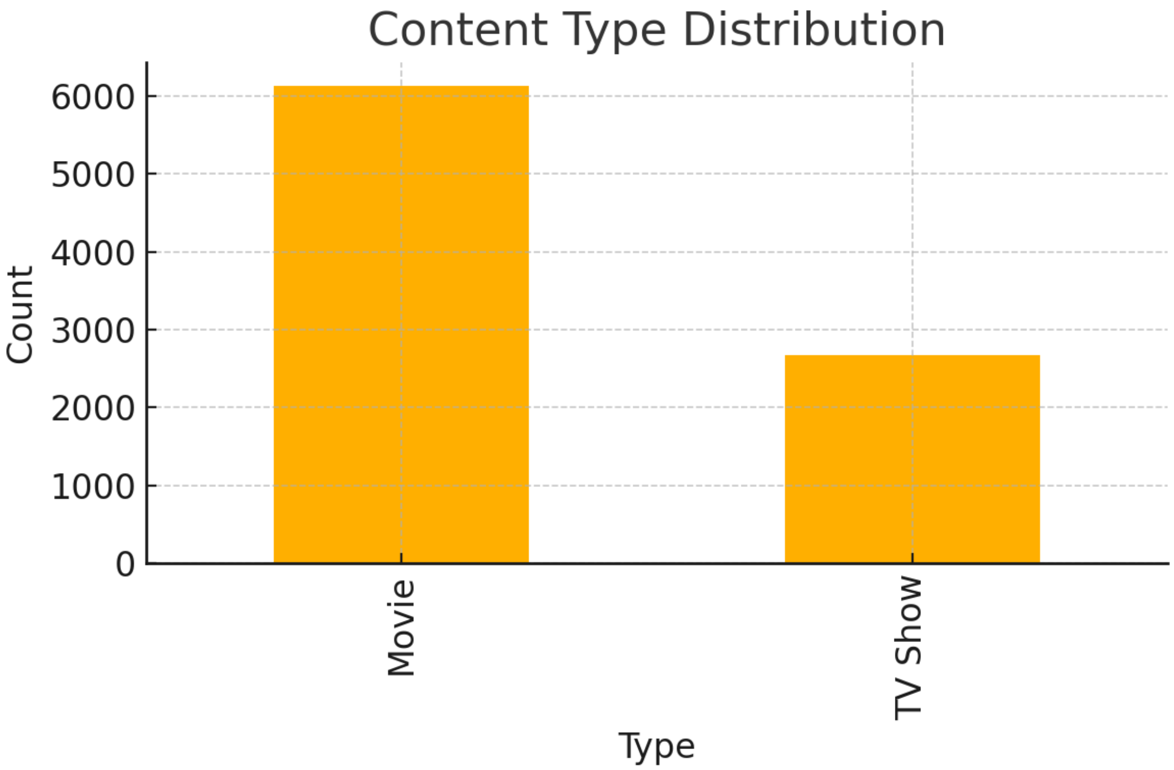
```
1942: 35.00 min

1943: 62.67 min

1944: 52.00 min

1945: 51.33 min

1946: 58.00 min
```

1. Content Type Distribution

The bar chart shows the distribution of content types on Netflix between Movies and TV Shows.

From the visualization, it is observed that Movies dominate the Netflix platform with over 6,000 titles, while TV Shows account for fewer entries, around 2,600.

This highlights that, historically, Netflix has had a stronger focus on movies compared to serialized shows.

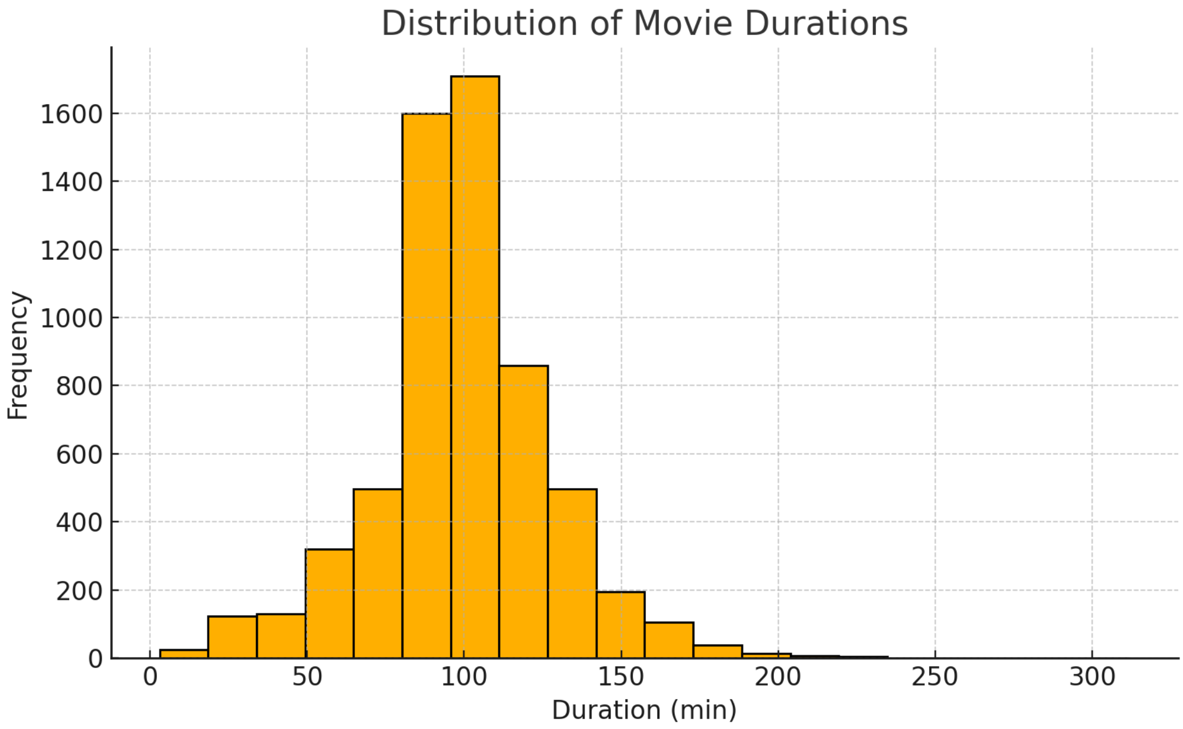


2. Distribution of Movie Durations

The histogram displays the frequency distribution of movie durations on Netflix.

Most of the movies cluster around 90 to 110 minutes, indicating that standard movie lengths are highly favored.

The distribution is slightly right-skewed, with a few longer-duration movies extending up to 200 minutes and beyond.

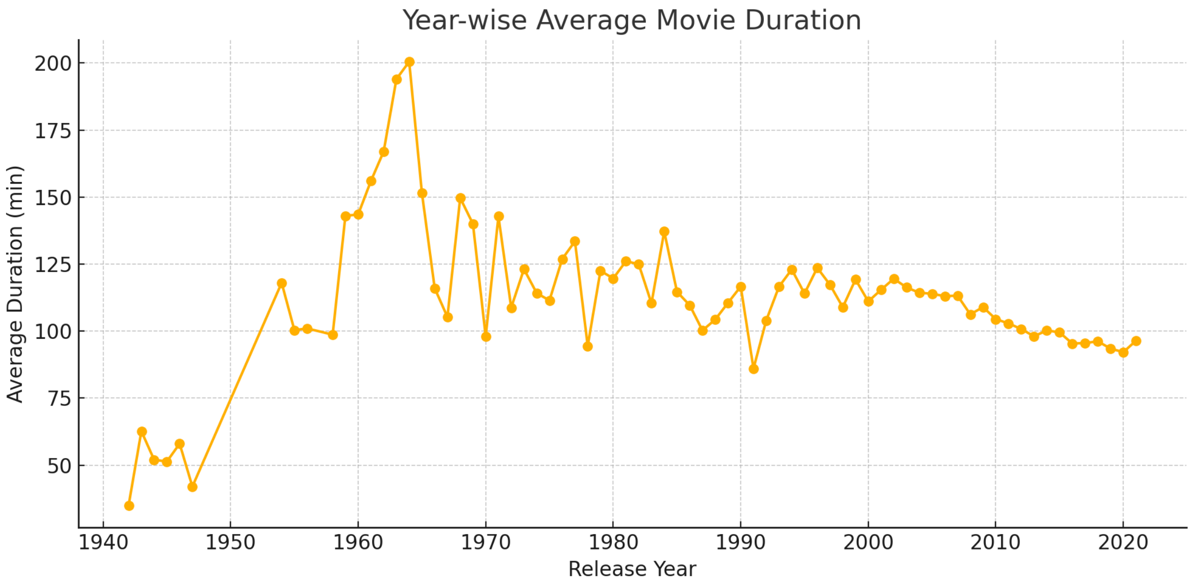


3. Year-wise Average Movie Duration

The line graph depicts the trend of the average movie duration over different release years.

During the 1950s and 1960s, movies tended to have longer average durations, crossing 150 minutes in some years.

In more recent decades, the average duration of movies has become more stable, generally ranging between 90 and 120 minutes, reflecting modern audience preferences for shorter films.

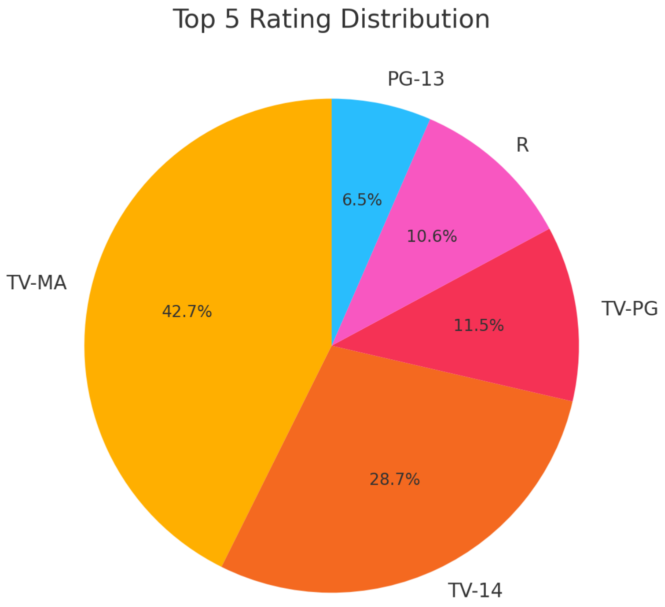


4. Top 5 Rating Distribution

The pie chart illustrates the distribution of the top five most common content ratings on Netflix.

The rating TV-MA (suitable for mature audiences) dominates with over 42% of the titles, followed by TV-14 and TV-PG.

This indicates that a significant portion of Netflix’s content is geared towards older teens and adult audiences, reflecting their target demographic.



CONCLUSION

Through the Netflix Titles dataset, this project effectively demonstrated the practical application of NumPy in managing, processing, and analyzing real-world entertainment data. By conducting essential statistical and numerical operations, we successfully uncovered significant patterns, such as the distribution between movies and TV shows, the frequency of releases across different years, the trends in average movie durations, and the shifting focus towards different content ratings and genres over time.

Our analysis highlighted how simple NumPy techniques — like calculating means, standard deviations, and value counts — can yield powerful insights into large datasets without the need for overly complex models. These findings not only reflect historical content trends on Netflix but also offer predictive value in understanding future audience engagement and platform strategies.

This project underscores the critical role that foundational data science tools like NumPy play in extracting actionable intelligence from vast entertainment datasets. By enabling efficient numerical analysis, NumPy helps drive smarter decision-making in areas such as content acquisition, library expansion planning, and user recommendation systems. Ultimately, this study demonstrates how basic yet robust computational techniques can bridge the gap between raw data and strategic business insights, proving indispensable in today’s data-driven media landscape.