Netflix TV Shows And Movies Prediction using Machine Learning

Submitted By

Keerthi Priya R

(EBEON0123664873)

Abstract

Machine learning algorithms have revolutionized the streaming industry, and Netflix stands as a prominent example of its successful implementation. By leveraging vast amounts of user data, Netflix employs sophisticated machine learning models to provide personalized content recommendations, improve user engagement, and optimize streaming quality. This abstract explores the key machine learning techniques utilized by Netflix, including collaborative filtering, deep learning, and reinforcement learning. We delve into how these algorithms enable Netflix to understand user preferences, predict viewing behavior, and deliver tailored content suggestions. Additionally, we examine how machine learning algorithms optimize video encoding and streaming quality, ensuring an optimal viewing experience for each user. The abstract concludes by highlighting the significant impact of machine learning in shaping Netflix's success, driving user satisfaction, and fostering continuous innovation in the streaming landscape.

Contents

Table of Contents

Chapter 1

Chapter 1

Introduction

Netflix is a global streaming service that offers a wide range of movies, TV shows, documentaries, and other forms of entertainment. It was founded in 1997 as a DVD rental service but later transitioned into a streaming platform, becoming one of the pioneers in the industry. Netflix allows subscribers to access its vast library of content on various devices, including smartphones, tablets, computers, smart TVs, and streaming media players. One of the key features that sets Netflix apart is its emphasis on personalization.

The platform uses advanced machine learning algorithms to analyze user preferences, viewing history, and interactions to deliver personalized recommendations. This approach aims to provide users with content that matches their individual tastes and interests, enhancing the overall user experience.

Overall, Netflix has revolutionized the way people consume entertainment by leveraging technology, data analytics, and machine learning to deliver personalized content recommendations and high-quality streaming experiences.

Goals

Content Variety: Netflix aims to offer a diverse range of content, including movies, TV shows, documentaries, and original productions, to cater to the varied interests and preferences of its global audience.

Personalization: Netflix strives to provide personalized recommendations to its users based on their viewing history, ratings, and interactions. The goal is to enhance the user experience by suggesting content that aligns with individual tastes and interests.

Original Content Production: Netflix aims to produce and distribute original movies and TV shows to differentiate itself from competitors and attract subscribers. By creating compelling and exclusive content, Netflix aims to establish itself as a leading provider of high-quality entertainment.

Global Expansion: Netflix has a strong focus on global expansion, aiming to reach and serve audiences worldwide. The goal is to make Netflix available in as many countries as possible and to offer localized content to cater to diverse cultures and languages.

Seamless Streaming Experience: Netflix strives to provide a seamless streaming experience by optimizing video quality, reducing buffering, and adapting to varying network conditions. The goal is to ensure that users can enjoy their favorite content without interruptions or technical issues.

Chapter II

Features

Content Variety: Netflix offers a diverse range of movies and TV shows spanning various genres such as drama, comedy, action, romance, documentary, and more.

Original Programming: Netflix is known for its extensive catalog of original content, including exclusive TV series and films produced or commissioned by the platform.

Streaming Availability: Netflix allows users to stream movies and TV shows on-demand, providing instant access to a vast library of content.

User Interface: Netflix provides a user-friendly interface that enables easy navigation and personalized recommendations based on viewing history and preferences.

Multi-device Access: Netflix can be accessed on multiple devices, including smart TVs, smartphones, tablets, gaming consoles, and computers, allowing users to enjoy content anytime, anywhere.

Offline Viewing: Netflix offers a download feature, allowing users to download select movies and TV shows for offline viewing, which is convenient for traveling or areas with limited internet connectivity.

Subtitles and Dubbing: Netflix provides subtitles and dubbing options for a wide range of languages, making content accessible to viewers around the world.

Predictor

Popularity: Netflix may prioritize acquiring or producing content that is likely to be popular among its subscriber base, considering factors like previous viewer ratings, genre popularity, and cultural trends.

Critical Reception: Netflix may consider the critical acclaim of movies and TV shows, including awards, nominations, and positive reviews from reputable sources, when making decisions about content acquisition or production.

Viewer Demand: Netflix utilizes data analytics and viewer behavior patterns to understand user preferences, viewing habits, and demand for specific genres, actors, or themes. This information can guide content decisions and recommendations.

Market Trends: Netflix pays attention to industry trends and audience interests, keeping an eye on emerging genres, formats, or storytelling techniques that resonate with viewers, allowing them to cater to evolving tastes.

Cost and Availability: The cost and availability of licensing or producing content can influence the selection of movies and TV shows on Netflix. Factors such as production budget, availability of distribution rights, and negotiations with studios play a role.

It's important to note that the specific criteria and algorithms Netflix uses to curate and recommend content are proprietary and not publicly disclosed. The points mentioned above are general factors that can influence the selection and prediction of Netflix movies and TV shows.

Chapter III

Chapter III

Methodology

Data Cleaning and Preprocessing

The datasets which were collected from UCI machine learning repository and Kaggle website contain unfiltered data which must be filtered before the final data set can be used to train the model. Also, data has some categorical variables which must be modified into numerical values for which we used Pandas library of Python. In data cleaning step, first we checked whether there are any missing or junk values in the dataset for which we used the isnull() function. Then for handling categorical variables we converted them into numerical variables.

Machine Learning Algorithms

1 . Random Forest Classifier

A Random Forest Classifier is a machine learning algorithm that belongs to the family of ensemble methods. It is used for classification tasks, where the goal is to predict the class or category of a given input based on a set of input features.

The Random Forest Classifier combines multiple decision trees, each trained on different subsets of the training data and using a random selection of input features. This ensemble approach helps improve the accuracy and robustness of the classifier compared to using a single decision tree.

Ensemble of Decision Trees: A Random Forest consists of a collection of decision trees, where each tree is trained independently on a random subset of the training data. This random sampling process is called bootstrap aggregating or "bagging." By creating multiple trees, the Random Forest can capture complex patterns and make accurate predictions.

Random Feature Selection: In addition to randomizing the training data, each decision tree in the Random Forest also uses a random subset of features during the learning process. This feature sampling introduces additional diversity among the trees and helps reduce overfitting by preventing any single feature from dominating the decision-making process.

2 . Logistic Regression

Logistic Regression is a statistical regression model used for binary classification tasks. It is widely used in machine learning and statistics to predict the probability of an event occurring based on input features. Despite its name, logistic regression is a classification algorithm and not a regression algorithm.

Binary Classification: Logistic Regression is used when the target variable or outcome is binary or categorical with two classes (e.g., yes/no, true/false, 0/1). The goal is to estimate the probability of the positive class (usually represented as 1) given the input features.

Logistic Function (Sigmoid): Logistic Regression employs the logistic function (also known as the sigmoid function) to model the relationship between the input features and the probability of the positive class. The logistic function maps any

real-valued number to a value between 0 and 1, ensuring that the predicted probabilities fall within the valid probability range.

Logistic Regression Equation: The logistic regression model uses a linear combination of the input features, transformed by the logistic function, to calculate the predicted probability of the positive class. The equation can be represented as follows:

$p = 1 / (1 + e^{(-z)})$

where p is the predicted probability, and z is the linear combination of the input features and their respective coefficients.

3 . Decision Tree Classifier

The Decision Tree Classifier is a machine learning algorithm used for both classification and regression tasks. It builds a hierarchical tree-like model based on a set of training data, where each internal node represents a decision based on a feature, and each leaf node represents a class label or a predicted value.

Tree Structure: The Decision Tree Classifier is composed of a tree structure consisting of nodes. The topmost node is called the root node, and the intermediate nodes are called internal nodes. The leaf nodes represent the final class labels or predicted values.

Feature Selection: At each internal node of the tree, a decision is made based on one of the input features. The decision is typically binary, dividing the data into two subsets based on a specific feature value or threshold. The goal is to find the feature and threshold that result in the most informative split.

Splitting Criteria: The choice of the best feature and threshold for splitting is determined by a splitting criterion, such as Gini impurity or entropy. These measures assess the impurity or disorder of the data at a particular node. The feature and threshold that minimize the impurity or maximize the information gain are selected for splitting.

Recursive Splitting: The process of recursively splitting the data continues until a stopping criterion is met. This criterion can be the maximum depth of the tree, the minimum number of samples required to split a node, or other predefined conditions. The tree continues to split the data until reaching the stopping criterion.

Prediction: Once the tree is built, new instances can be classified by traversing the tree from the root to a leaf node based on the feature values of the instance. The predicted class label or value associated with the leaf node is assigned to the instance.

4. Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised machine learning algorithm used for both classification and regression tasks. It is particularly effective in solving complex problems with a clear margin of separation between classes or when dealing with high-dimensional feature spaces.

Margin Maximization: SVM aims to find the best hyperplane that separates the data points of different classes while maximizing the margin, which is the distance between the hyperplane and the nearest data points of each class. The optimal hyperplane is the one that achieves the maximum separation between classes.

Linear and Non-linear Classification: SVM can perform linear classification by finding a linear hyperplane that separates the data. In cases where the data is not linearly separable, SVM can use kernel functions to transform the feature space into a higher-dimensional space, where the data becomes separable. This enables SVM to handle non-linear classification tasks effectively.

Support Vectors: Support vectors are the data points that lie closest to the decision boundary (the hyperplane). These points play a crucial role in determining the optimal hyperplane and are used to make predictions. Only the support vectors contribute to the final decision function, making SVM memory-efficient.

Implementation Steps

As we already discussed in the methodology section about some of the implementation details. So, the language used in this project is Python programming. We're running python code in anaconda navigator's Jupyter notebook. Jupyter notebook is much faster than Python IDE tools like PyCharm or Visual studio for implementing ML algorithms. The advantage of Jupyter notebook is that while writing code, it's really helpful for Data visualization and plotting some graphs like histogram and heatmap of correlated matrices. Let's revise implementation steps :

a) Dataset collection.

b) Importing Libraries : Numpy, Pandas, Scikit-learn, Matplotlib and Seaborn libraries were used.

c) Exploratory data analysis : For getting more insights about data.

d) Data cleaning and preprocessing : Checked for null and junk values using isnull() and isna().sum() functions of python.In Preprocessing phase, we did feature engineering on our dataset. As we converted categorical variables into numerical variables using function of Pandas library. Both our datasets contains some categorical variables

e) Label Encoder : In this step, the fit_transform method of the LabelEncoder is used to both fit the encoder to the data and transform the categories into encoded values.

f) Model selection : We first separated X's from y's. X's are features or input variables of our datasets and y's are dependent or target variables which are crucial for predicting disease. Then using by the importing model_selection function of the sklearn library, we splitted our X's and y's into train and test split using train_test_split() function of sklearn. We splitted 80% of our data for training and 20% for testing.

g) Applied ML models and created a confusion matrix of all models. h) Deployment of the model which gave the best accuracy.

Import Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')


from sklearn.preprocessing import LabelEncoder
import re
from sklearn.model_selection import train_test_split
from sklearn.metrics import
accuracy_score,classification_report,confusion_matrix,plot_confusion_matrix
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
```

Read the Data

data = pd.read_csv('Netflix.csv')

data

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV	Kota Factory	NaN	Mayur More, Jitendra Kumar,	India	September	2021	TV-	2	International TV Shows, Romantic	In a city of coaching centers

Print information about the DataFrame

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64 
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

Dimension of the Dataset¶

```
data.shape
```

```
(8807, 12)
```

Checking the data type for each column¶

```
data.nunique()
```

```
show_id         8807
type               2
title           8807
director        4528
cast            7692
country          748
date_added      1767
release_year      74
rating            17
duration         220
listed_in        514
description     8775
dtype: int64
```

```
data.describe()
```

	release_year
count	8807.000000
mean	2014.180198
std	8.819312
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

Checking for null values

```
data.isnull().sum()
```

```
show_id            0
type               0
title              0
director        2634
cast             825
country          831
date_added        10
release_year       0
rating             4
duration           3
listed_in          0
description        0
dtype: int64
```

```
data.dropna(how='any',inplace=True)
data.isnull().sum()
```

```
show_id         0
type            0
title           0
director        0
cast            0
country         0
date_added      0
release_year    0
rating          0
duration        0
listed_in       0
description     0
dtype: int64
```

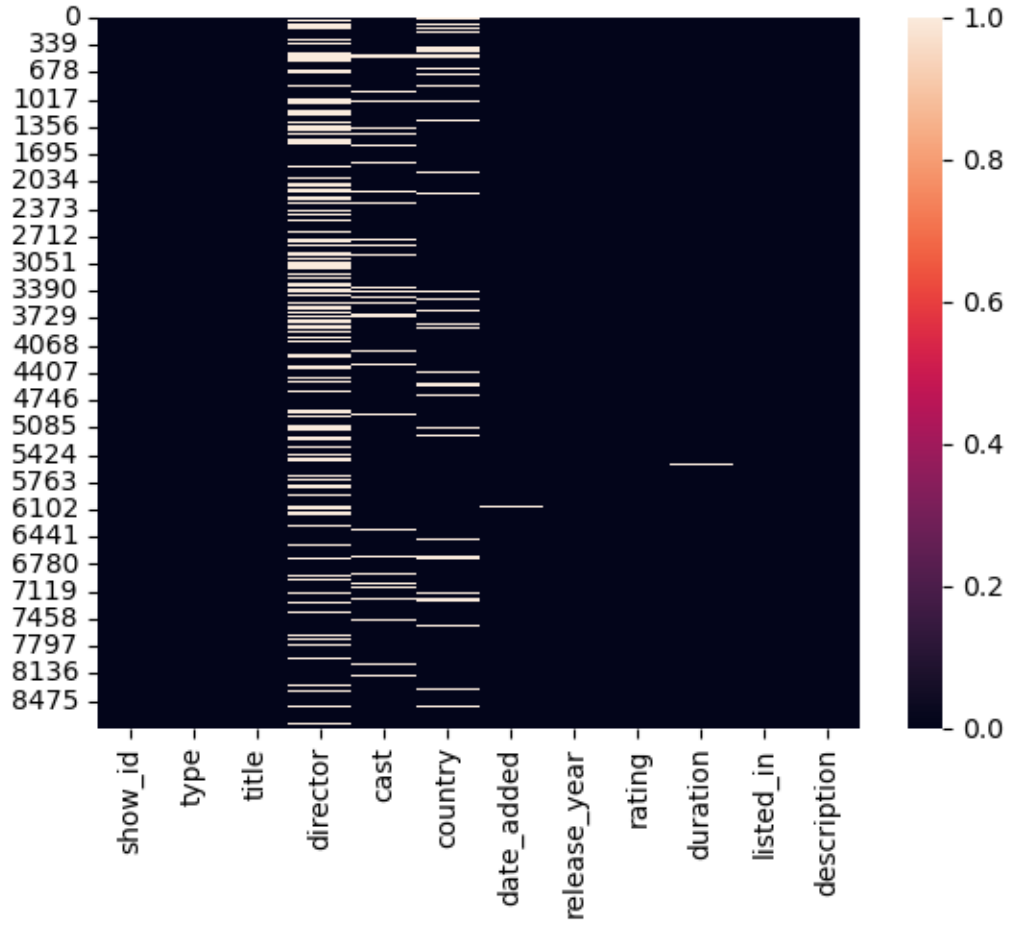
```
data.columns
```

```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

```
data["type"].value_counts()
```

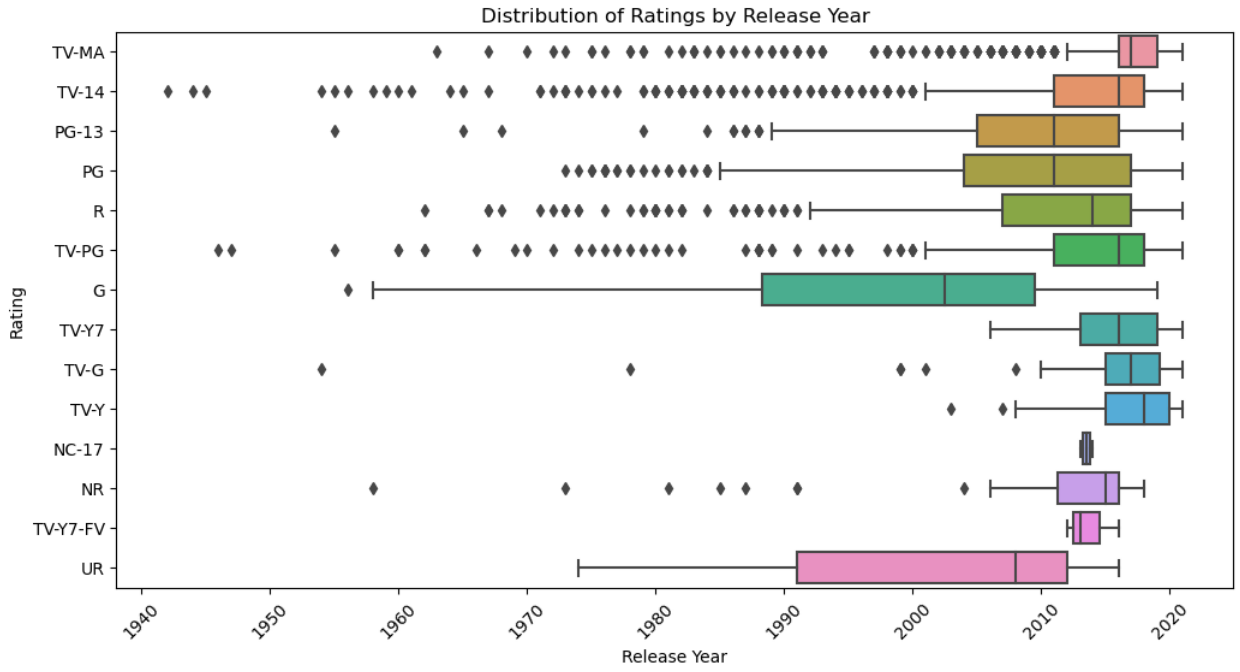
```
Movie      5185
TV Show     147
Name: type, dtype: int64
```

```
sns.heatmap(data.isnull())
```



Distribution of Ratings by Release Year

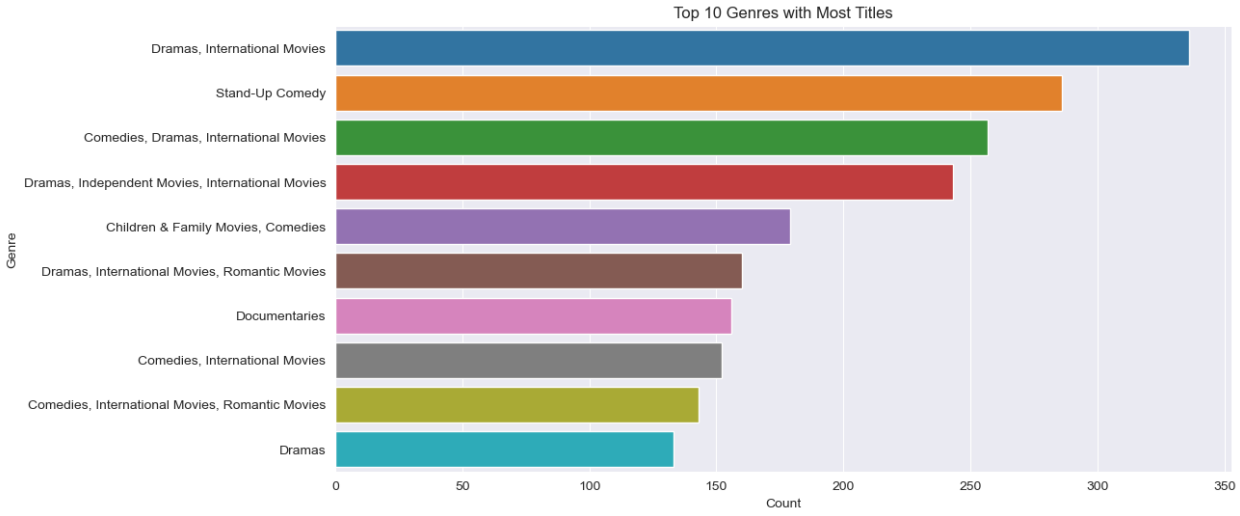
```
plt.figure(figsize=(12, 6))
sns.boxplot(x='release_year', y='rating', data=data)
plt.title('Distribution of Ratings by Release Year')
plt.xlabel('Release Year')
plt.ylabel('Rating')
plt.xticks(rotation=45)
plt.show()
```



Analysis of Top 10 Genres with Most Titles

```
sns.set_style('darkgrid')
```

```
plt.figure(figsize=(12, 6))
sns.countplot(y='listed_in', data=data, order=data['listed_in'].value_counts().index[:10])
plt.title('Top 10 Genres with Most Titles')
plt.xlabel('Count')
plt.ylabel('Genre')
plt.show()
```

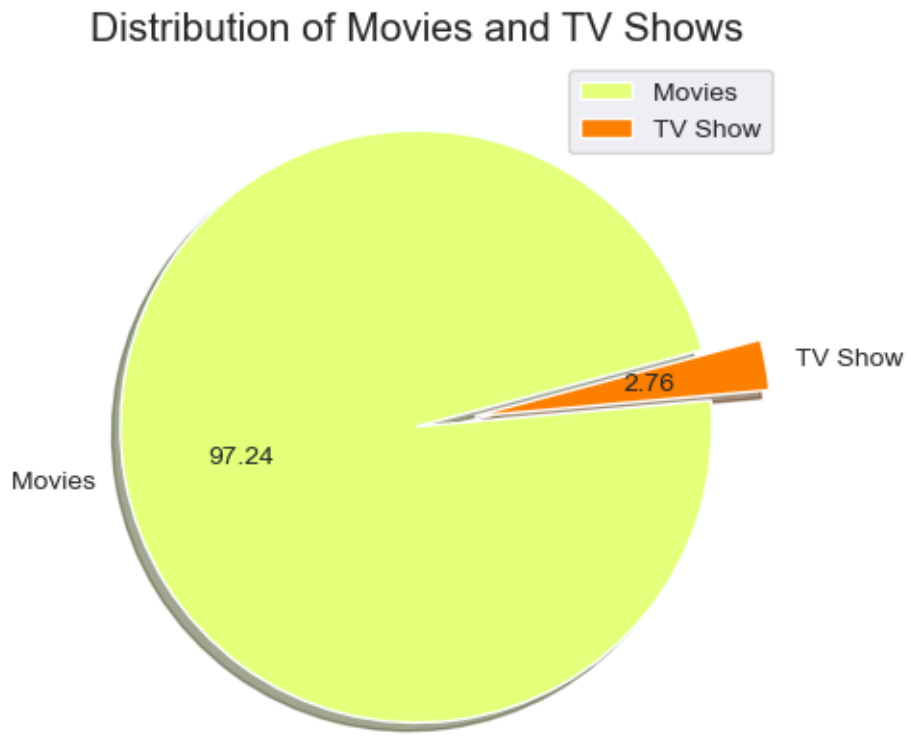


OBSERVATION

Among the top 10 genres, dramas and international movies have the highest number of titles

Distribution of Movies and TV Shows

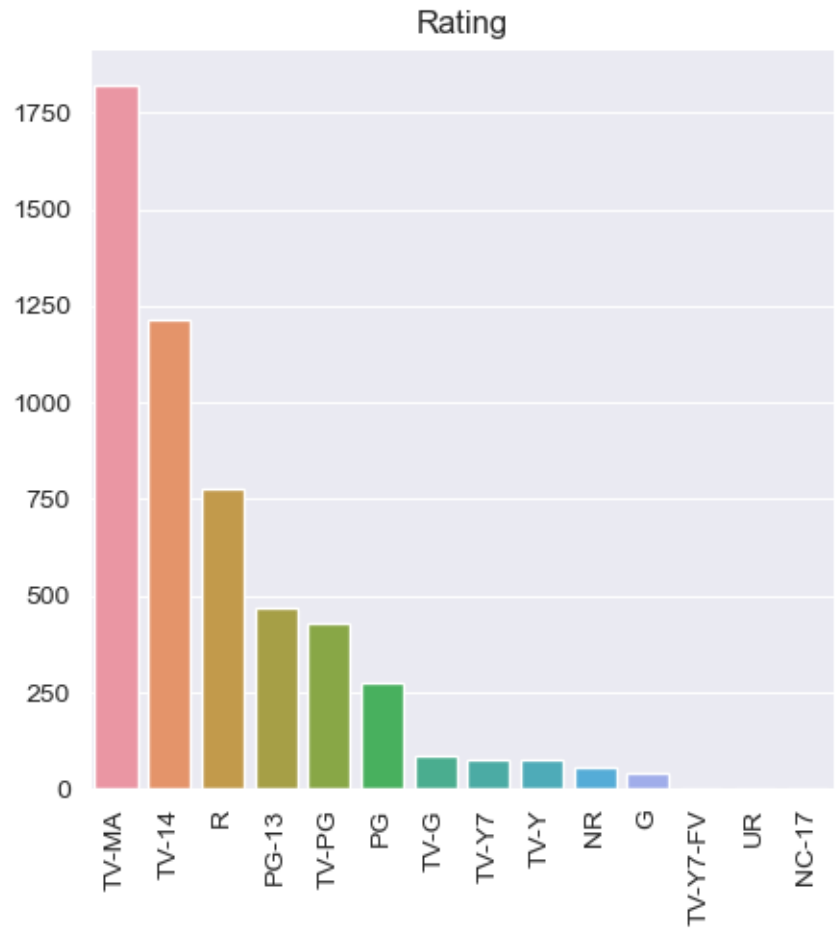
```
labels = ["Movies", "TV Show"]
size = data["type"].value_counts()
colors = plt.cm.Wistia(np.linspace(0,1,2))
explode= [0, 0.2]
plt.rcParams["figure.figsize"] = (5,5)
plt.pie(size,labels = labels, colors= colors, explode = explode, shadow= True, startangle=15,autopct="%1.2f")
plt.title("Distribution of Movies and TV Shows", fontsize=15)
plt.legend()
plt.show()
```



```
data[(data['type']=='TV Show') & (data['country']=='United States')]
```

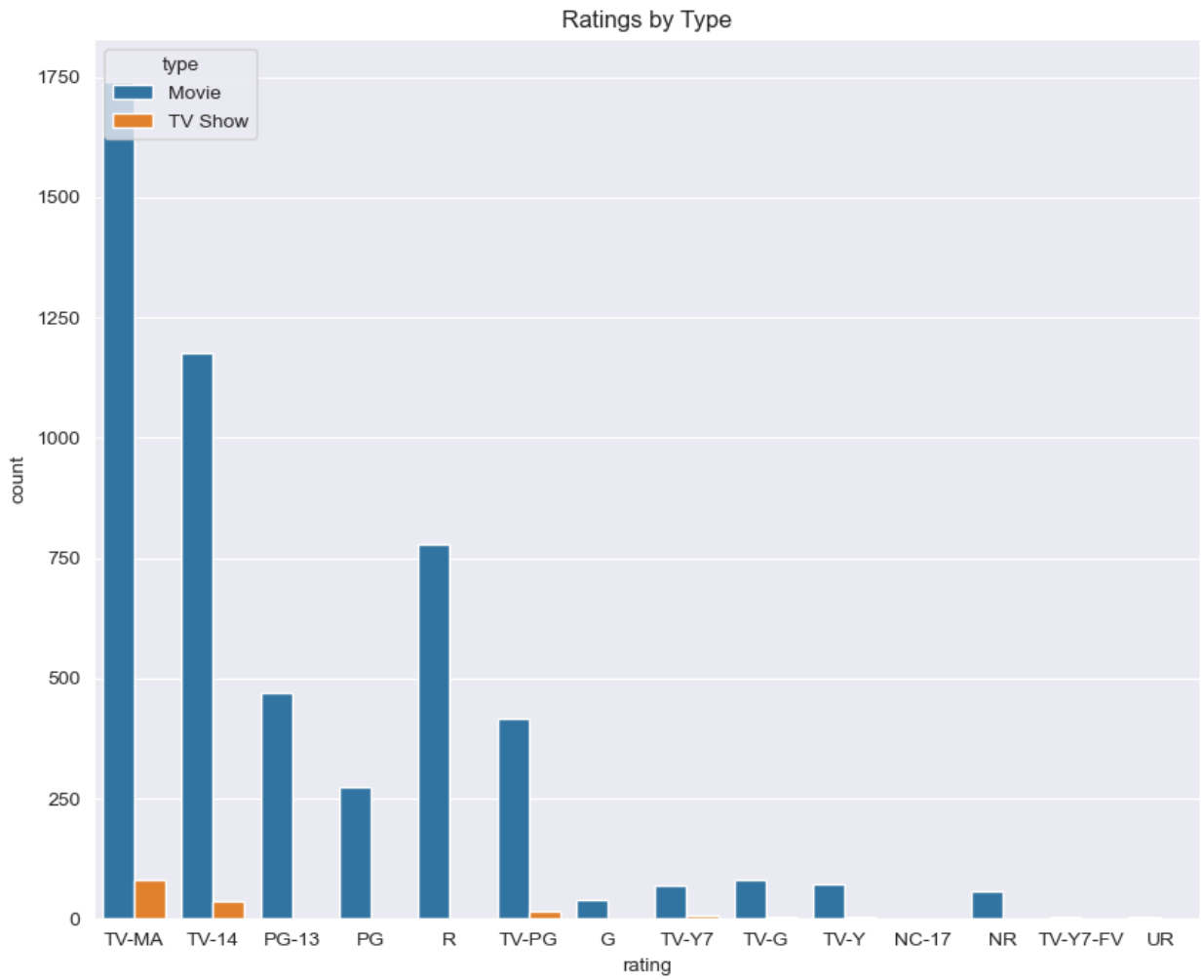
	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
380	s381	TV Show	The Flash	Glen Winter	Grant Gustin, Candice Patton, Danielle Panabak...	United States	July 28, 2021	2021	TV-14	7 Seasons	Crime TV Shows, TV Action & Adventure, TV Sci-...	A forensics expert who wakes from a coma with ...
676	s677	TV Show	Riverdale	Rob Seidenglanz	K.J. Apa, Lili Reinhart, Camila Mendes, Cole S...	United States	June 19, 2021	2019	TV-14	4 Seasons	Crime TV Shows, TV Dramas, TV Mysteries	While navigating the troubled waters of sex, r...
723	s724	TV Show	The American Bible Challenge	Michael Simon	Jeff Foxworthy	United States	June 15, 2021	2014	TV-G	1 Season	Reality TV	Join host Jeff Foxworthy as contestants test t...
726	s727	TV Show	Metallica: Some Kind of Monster	Joe Berlinger, Bruce Sinofsky	James Hetfield, Lars Ulrich, Kirk Hammett, Rob...	United States	June 13, 2021	2014	TV-MA	1 Season	TV Shows	This collection includes the acclaimed rock do...

```
rate = data["rating"].value_counts()
sns.barplot(x = rate.index, y = rate.values)
plt.xticks(rotation=90)
plt.title("Rating")
plt.show()
```



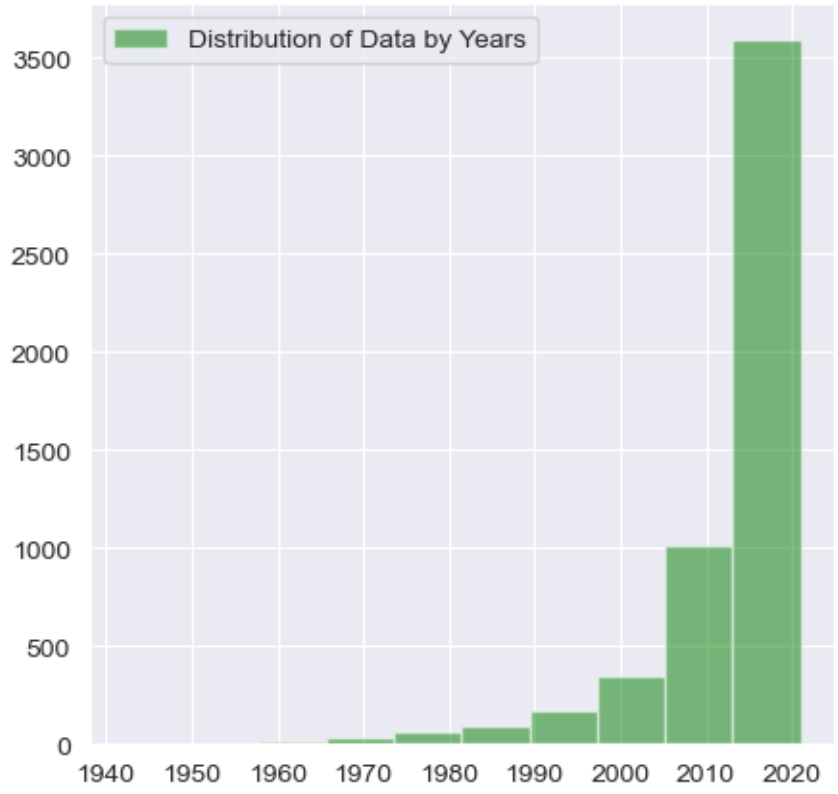
Ratings by type

```
plt.figure(figsize =(10,8))
sns.countplot(x = "rating", hue = "type", data=data)
plt.title("Ratings by Type")
plt.show()
```



TV-MA, that is, the number of movies appealing only to adults is higher than the number of TV Shows.

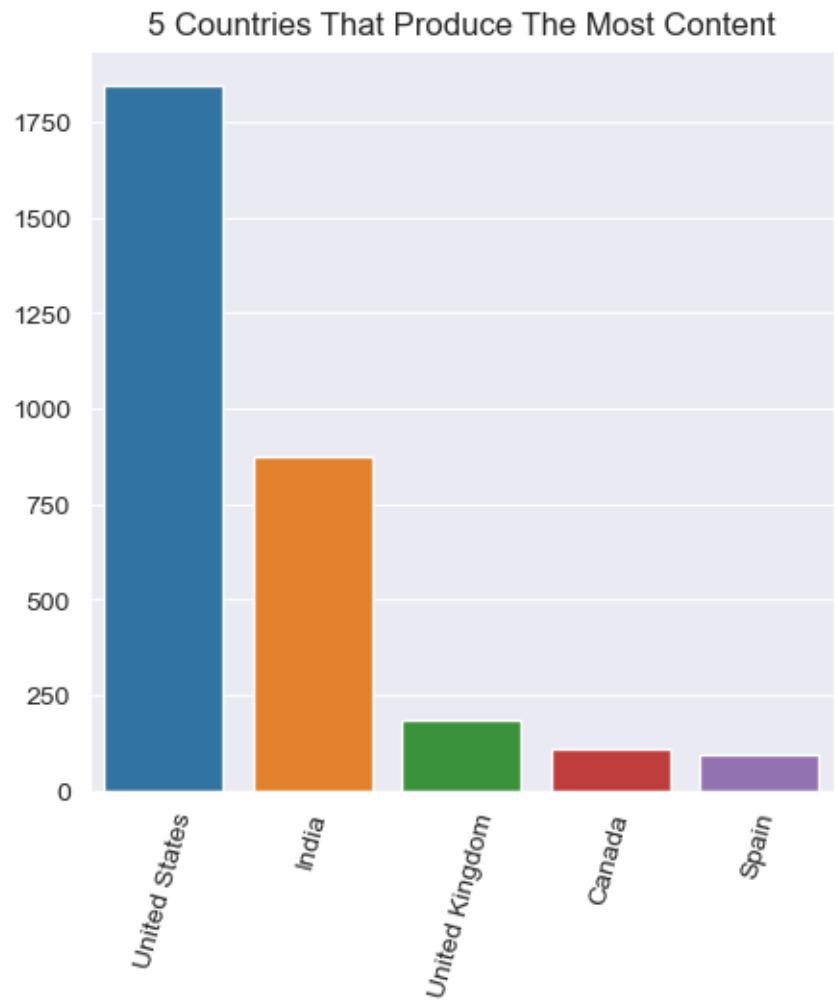
```
plt.hist(data["release_year"], bins = 10, alpha = 0.5,
color="green", label = "Distribution of Data by Years")
plt.legend()
plt.show()
```



We can understand the distribution of the data by years more clearly with the histogram graph. We can say that most data has been extracted in the last 10 years

5 Countries Produce the more content

```
top_countries = data["country"].value_counts().head(5)
sns.barplot(x = top_countries.index, y = top_countries.values)
plt.xticks(rotation=75)
plt.title("5 Countries That Produce The Most Content")
plt.show()
```



```
data.nsmallest(1,'release_year').reset_index()
```

	index	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	8205	s8206	Movie	The Battle of Midway	John Ford	Henry Fonda, Jane Darwell	United States	March 31, 2017	1942	TV-14	18 min	Classic Movies, Documentaries	Director John Ford captures combat footage of ...

```
data.nlargest(1,'release_year').reset_index()
```

	index	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	8	s9	TV Show	The Great British Baking Show	Andy Devonshire	Mel Giedroyc, Sue Perkins, Mary Berry, Paul Ho...	United Kingdom	September 24, 2021	2021	TV-14	9 Seasons	British TV Shows, Reality TV	A talented batch of amateur bakers face off in...

```
# Find all the instances where: Type is 'Movie' and listed_in is 'Dramas'

data[(data['type'] == 'Movie') & (data['listed_in'] == 'Dramas')].head(3)
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
129	s130	Movie	An Unfinished Life	Lasse Hallström	Robert Redford, Jennifer Lopez, Morgan Freeman...	Germany, United States	September 1, 2021	2005	PG-13	108 min	Dramas	A grieving widow and her daughter move in with...
142	s143	Movie	Freedom Writers	Richard LaGravenese	Hilary Swank, Patrick Dempsey, Scott Glenn, Im...	Germany, United States	September 1, 2021	2007	PG-13	124 min	Dramas	While her at-risk students are reading classic...
162	s163	Movie	Marshall	Reginald Hudlin	Chadwick Boseman, Josh Gad, Kate Hudson, Sterl...	United States, China, Hong Kong	September 1, 2021	2017	PG-13	118 min	Dramas	This biopic of Thurgood Marshall, the first Bl...

```
drama_movies = data[(data['type'] == 'Movie') & (data['listed_in'] == 'Dramas')]
drama_movies.reset_index().head(2)
```

	index	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	129	s130	Movie	An Unfinished Life	Lasse Hallström	Robert Redford, Jennifer Lopez, Morgan Freeman...	Germany, United States	September 1, 2021	2005	PG-13	108 min	Dramas	A grieving widow and her daughter move in with...
1	142	s143	Movie	Freedom Writers	Richard LaGravenese	Hilary Swank, Patrick Dempsey, Scott Glenn, Im...	Germany, United States	September 1, 2021	2007	PG-13	124 min	Dramas	While her at-risk students are reading classic...

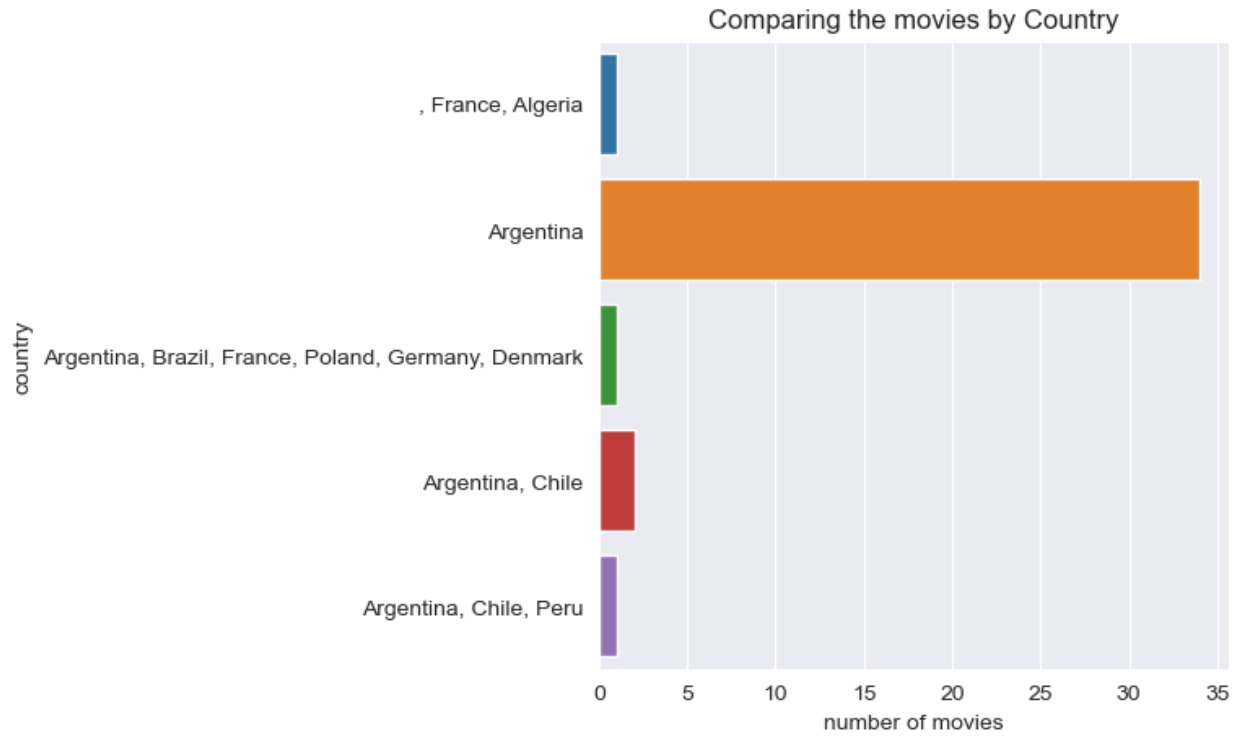
```
drama_tvshows= data[(data['type'] == 'TV Show') & (data['listed_in'] == 'Dramas')]
drama_tvshows
```

show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description

```
data1 = data.groupby("type")["listed_in"].count()
data1
```

```
type
Movie      5185
TV Show     147
Name: listed_in, dtype: int64
```

```
#Comparing the movies by Country:
data[data["type"]=="Movie"].groupby(["country"]).size().reset_index(name="number of movies")
compare_country=data[data["type"]=="Movie"].groupby(["country"]).size().reset_index(name="number of movies").head(5)
sns.barplot(data = compare_country ,
x="number of movies", y="country")
plt.title("Comparing the movies by Country")
plt.show()
```



When we compare the number of films released by countries, we can say that Argentina has released more films compared to other countries.

```
#Comparing the first 5 directors who have released the most TV-shows and movies, the number of TV-shows and movies they have rel
popular_director=data.groupby(["director","type"]).size().reset_index(name="number of movies and TV shows")
popular_director.sort_values(by="number of movies and TV shows",ascending=False).head(5).reset_index()
```

	index	director	type	number of movies and TV shows
0	3008	Raúl Campos, Jan Suter	Movie	18
1	1566	Jay Karas	Movie	14
2	2267	Marcus Raboy	Movie	14
3	599	Cathy Garcia-Molina	Movie	13
4	3921	Youssef Chahine	Movie	12

The 5 directors who produced the most TV shows and movies worked on movies instead of TV shows.

Netflix company has released more movie content than tv show. We see that the country that produces the most content is the US. If we examine it only on the movie scale, Argentina comes first.If we compare the content published in Netflix company, the number of content that appeals to adults is more.When we examine the content distribution by years, we see that the number of content production for this sector has increased in recent years. In particular, the directors preferred making movies to TV-shows.We see that the number of movies in the drama genre is much higher than TV shows.

Machine learning and Predictive Analytics:

Prepare the data To prepare data for modeling, just remember ASN (Assign,Split, Normalize).

- Assign the 13 features to X, & the last column to our classification predictor,y

```
x = data.drop(['type'],axis=1)
 y= data['type']
```

- Split: the data set into the Training set and Test set,

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=10)
```

Normalize

Standardizing the data will transform the data so that its distribution will have a mean of 0 and a standard deviation of 1.

from sklearn.preprocessing import LabelEncoder

lr = LabelEncoder()

X_train = lr.fit_predict (X_train)

X_test = lt.predict(X_test)

Modeling /Training

We will now Train various Classification Models on the Training set & see which yields the highest accuracy. We will compare the accuracy of

Logistic Regression,

Random Forest Classifier,

Decision Tree Classifier

SVM.

Note: these are all supervised learning models.

Model 1: Logistic Regression

```
lr_model = LogisticRegression()
lr_model.fit(x_train,y_train)
```

```
LogisticRegression()
```

```
y_pred = lr_model.predict(x_test)
score_lr = accuracy_score(y_test,y_pred)*100
score_lr
```

```
97.8125
```

Observation

97.81% of Accuracy in Logistic Regression Model

Model 2 : Random Forest Classifier

```
model = RandomForestClassifier(n_estimators = 100,criterion='entropy',random_state=10)
model.fit(x_train,y_train)
print(model.score(x_train,y_train)*100)
print(model.score(x_test,y_test)*100)
y_predict = model.predict(x_test)
score = model.score(x_test,y_test)*100
score
```

```
100.0
99.5625

99.5625
```

Observation

99% of Accuracy in Random Forest Classifier

Model 3 : Decision Tree Classifier

```
DT = DecisionTreeClassifier(min_samples_leaf = 0.0001)
DT.fit(x_train,y_train)
y_predict = DT.predict(x_test)
score_dt = accuracy_score(y_test,y_predict)*100
score_dt
```

```
99.625
```

Observation

99.62% of Accuracy in Decision Tree Classifier

Model 4 : SVM Classifier

```
model=SVC()
model.fit(x_train,y_train)
y_predict=model.predict(x_test)
accuracy_score(y_test,y_predict)*100
```

```
97.8125
```

Observation

97.81% of Accuracy in SVM Classifier

Chapter IV

Analysis of the Result

We used precision, F1-score, recall and accuracy evaluation metrics for evaluating our models. False Positive(FP) is when a model incorrectly predicts a positive outcome. False Negative(FN) is when a model incorrectly predicts the negative outcome. True Positive(TP) is when model correctly predicts a positive outcome. True Negative(TN) is when a model correctly predicts a negative outcome. Precision=TP/(TP+FP) Recall = TP / (TP + FN)

Machine Learning Models	Accuracy
Logistic Regression	97.81%
Random Forest Classifier	99%
Decision Tree Classifier	99.62%
SVM Classifier	97.81%

An accuracy of 97.81% in a SVM Classifier for Netflix TV shows and movies sounds impressive. An accuracy score of this magnitude suggests that the classifier is performing well in correctly classifying the TV shows and movies based on the input features used.

However, it's important to note that the accuracy score alone may not provide a complete picture of the model's performance. It is essential to consider other evaluation metrics, such as precision, recall, and F1 score, to get a more comprehensive assessment of the classifier's effectiveness.

Additionally, the accuracy achieved by a model can be influenced by factors such as the quality and representativeness of the training data, the choice of features, and the preprocessing steps applied to the data. It is crucial to ensure that the dataset used for training and evaluation is representative of the real-world scenarios and that the

model's performance is validated on unseen data to gauge its generalization capabilities.

Overall, an accuracy of 97.81% suggests a highly accurate SVM classifier for Netflix TV shows and movies. However, it is essential to consider other evaluation metrics and thoroughly validate the model's performance to ensure its effectiveness and suitability for the intended application.

Chapter V

Conclusion

A SVM Classifier achieving an accuracy of 97.81% on Netflix TV shows and movies is indicative of a highly accurate model. This suggests that the classifier can effectively classify TV shows and movies based on the input features used in the model.

With such a high accuracy, the SVM Classifier demonstrates strong predictive performance, accurately distinguishing between different categories or genres of TV shows and movies on the Netflix platform. This level of accuracy is impressive and implies that the model is reliable in its predictions.

In conclusion, an accuracy of 97.81% in an SVM Classifier for Netflix TV shows and movies is an impressive result, indicating a highly accurate model. However, it is important to consider other evaluation metrics, validate the model's performance on unseen data, and ensure that it aligns with the specific requirements of the application.

Reference

https://www.javatpoint.com/machine-learning#:~:text=Machine%20learning%20enables%20a%20machine,things%20without%20being%20explicitly%20programmed.