November 2023

Global Youtube Statistics with Machine Learning

Kanyasorn Phongphaitoonsin st122983
Nitesh Ghimire st124453

Present to
Professor Chantri Polprasert

Project Stages

Chapter 1

Introduction

Chapter 2

Problem Statement

Chapter 3

Related Works

Chapter 4

Dataset

Chapter 5

Methodology

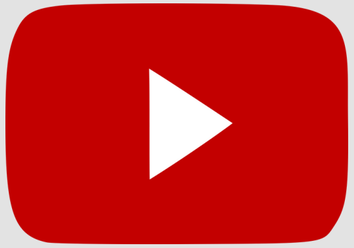
Chapter 6

Result

Chapter 7

Conclusion

Introduction



- Global YouTube Statistics 2023" dataset offers insights into user demographics, content categories, and engagement patterns worldwide.
- Valuable for data scientists, marketers, and those interested in digital culture, providing a comprehensive view of YouTube's global impact.
- Enables researchers to explore and understand YouTube's role in the modern digital landscape.

Problem statement

Emerging Trends and Predictive Analytics

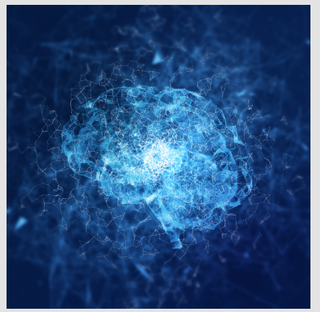
Can the dataset be used to identify emerging trends and make predictions about the future of YouTube in 2023 and beyond?

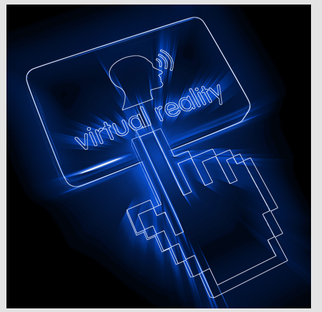
Prescriptive Content Strategies

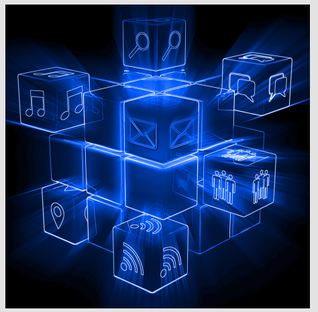
Based on predictive insights, how can we develop prescriptive content strategies that advise content creators on the types of content, formats, and timing that are most likely to attract and retain viewers?

Predictive Analytics for User Engagement

How can predictive models be developed to forecast user engagement and content performance on YouTube in 2023?







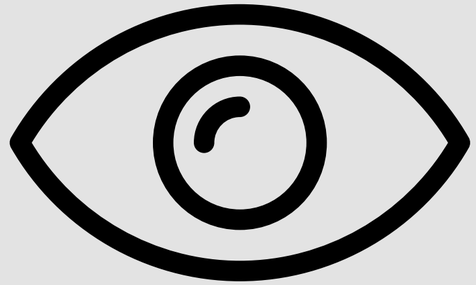
Related works

Predictive analysis of YouTube trending videos using Machine Learning

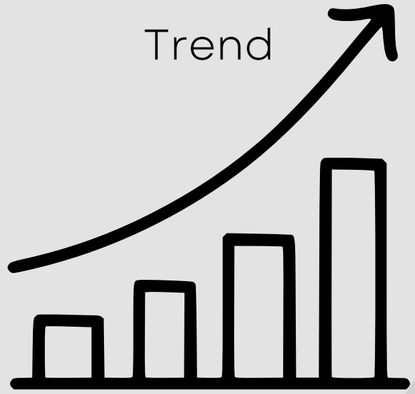
This research focuses on analyzing interactive features contributing to YouTube video trendiness, utilizing machine learning and viewership statistics from over 40,000 trending videos to determine correlations and variables influencing a video's popularity, achieving a maximum accuracy of 62.53% in predicting the lifecycle of trending videos.

Tier

View



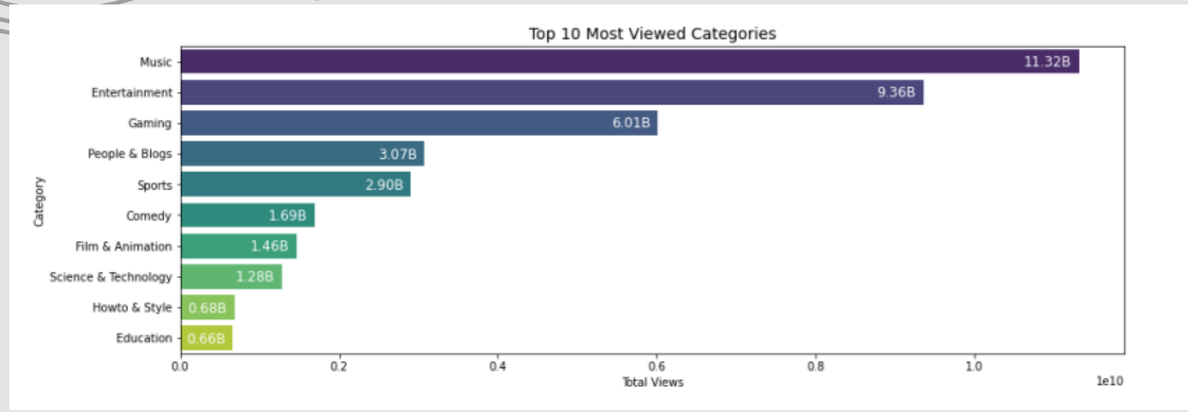
Trend



Related Works

Youtube Trending Videos: Boosting Machine Learning Results Using Exploratory Data Analysis

This paper analyzes YouTube trending video data using Big Data Technologies, data mining, and machine learning, examining 40,000 trending videos over 205 days. Employing exploratory data analysis and statistical methods, it aims to identify patterns in user activity, understand the impact of events on data quality, and predict future trending video categories to aid content creators in optimizing uploads for increased views and performance.



Related Works

Popularity Prediction of Videos in YouTube as Case Study: A Regression Analysis Study

The paper focuses on predicting YouTube video popularity through regression analysis, identifying and analyzing key variables influencing views. Serving as a case study, it employs statistical techniques to model relationships between factors and video views, offering valuable insights for content creators and marketers aiming to optimize strategies on the platform.

Parameters	Meaning
video ID	an 11-digit string, which is unique
uploader	a string of the video uploader's user-name
age	an integer number of days between the date when the video was uploaded
category	a string of the video category chosen by the uploader
length	an integer number of the video length
views	an integer number of the views
rate	a float number of the video rate
ratings	an integer number of the ratings
comments	an integer number of the comments.

Related Works

The Statistical Analysis based on YouTube Channel Dataset

The paper utilizes statistical analysis on a YouTube channel dataset to explore patterns, relationships, and trends within channel-related data. Examining variables like subscriber counts and engagement metrics, it aims to uncover significant correlations and dependencies, offering a data-driven understanding of factors influencing a channel's performance and popularity on the platform.

video ID	an 11-digit string, which is unique
uploader	a string of the video uploader's username
age	an integer number of days between the date when the video was uploaded and Feb.15, 2007 (YouTube's establishment)
category	a string of the video category chosen by the uploader
length	an integer number of the video length
views	an integer number of the views
rate	a float number of the video rate
ratings	an integer number of the ratings
comments	an integer number of the comments
related IDs	up to 20 strings of the related video IDs

Dataset

rank	Position of the YouTube channel based on the number of subscribers
Youtuber	Name of the YouTube channel
subscribers	Number of subscribers to the channel
video views	Total views across all videos on the channel
category	Category or niche of the channel
Title	Title of the YouTube channel
uploads	Total number of videos uploaded on the channel
Country	Country where the YouTube channel originates
Abbreviation	Abbreviation of the country
channel_type	Type of the YouTube channel (e.g., individual, brand)
video_views_rank	Ranking of the channel based on total video views
country_rank	Ranking of the channel based on the number of subscribers within its country
channel_type_rank	Ranking of the channel based on its type (individual or brand)
video_views_for_the_last_30_days	Total video views in the last 30 days
lowest_monthly_earnings	Lowest estimated monthly earnings from the channel
highest_monthly_earnings	Highest estimated monthly earnings from the channel
lowest_yearly_earnings	Lowest estimated yearly earnings from the channel
highest_yearly_earnings	Highest estimated yearly earnings from the channel
subscribers_for_last_30_days	Number of new subscribers gained in the last 30 days
created_year	Year when the YouTube channel was created
created_month	Month when the YouTube channel was created
created_date	Exact date of the YouTube channel's creation
Gross tertiary education enrollment (%)	Percentage of the population enrolled in tertiary education in the country
Population	Total population of the country
Unemployment rate	Unemployment rate in the country
Urban$_{p}opulation$	Percentage of the population living in urban areas
Latitude	Latitude coordinate of the country's location
Longitude	Longitude coordinate of the country's location

Timeline Project

1 Data Preprocessing

1. Import Library
2. Load Data:
3. See Data Information
4. Check Missing Values

2 Exploratory Data Analysis (EDA)

1. Top 10 Channels by Subscribers
2. Subscriber vs Video Views:
3. Video Views vs Highest Monthly Earnings
4. Count of YouTube Channels by Categories
5. Categories with the Highest Number of Subscribers
6. YouTube Channels with Highest Subscribers Group by Country
7. Top 10 Countries with the Highest Number of Channels
8. Top 10 Channels by Views
9. Highest Viewed Channels by Categories
10. Top 10 YouTube Channels with Highest Uploads to Subscriber Ratio
11. Highest Earning Categories
12. Distribution of Channel Creation Year
13. Correlation of the Dataset Within Columns

3 Machine Learning

1. Select Features and Target
2. GridSearchCV for Model and Parameter Selection
3. Train the Model
4. Pickle Model

4 Deployment

1. Create HTML
2. Deploy on AWS

Import Library

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

from sklearn import metrics
from datetime import datetime
import datetime


from sklearn.preprocessing import scale
from sklearn.model_selection import train_test_split
from sklearn.metrics import euclidean_distances
from sklearn.neighbors import KNeighborsClassifier

from sklearn.model_selection import GridSearchCV
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import KFold, cross_val_score
from sklearn.linear_model import LinearRegression,Ridge,Lasso,ElasticNet
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import StandardScaler

from sklearn.compose import ColumnTransformer # transform specific columns
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler, MinMaxScaler
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import OrdinalEncoder, OneHotEncoder
import scipy.stats as stats
```

Load data

```
1  df = pd.read_csv('data.csv', encoding = "ISO-8859-1")
```

✓ 0.0s Python

```
1  df.head()
```

✓ 0.0s Python

	rank	Youtuber	subscribers	video views	category	Title	uploads	Country	Abbreviation	channel_type	...	subscribers_for_last_30_days	created_year	created_month	created_d
0	1	T-Series	245000000	2.280000e+11	Music	T-Series	20082	India	IN	Music	...	2000000.0	2006.0	Mar	1
1	2	YouTube Movies	170000000	0.000000e+00	Film & Animation	youtubemovies	1	United States	US	Games	...	NaN	2006.0	Mar	
2	3	MrBeast	166000000	2.836884e+10	Entertainment	MrBeast	741	United States	US	Entertainment	...	8000000.0	2012.0	Feb	2
3	4	Cocomelon - Nursery Rhymes	162000000	1.640000e+11	Education	Cocomelon - Nursery Rhymes	966	United States	US	Education	...	1000000.0	2006.0	Sep	
4	5	SET India	159000000	1.480000e+11	Shows	SET India	116536	India	IN	Entertainment	...	1000000.0	2006.0	Sep	2

5 rows × 28 columns

See the information of data

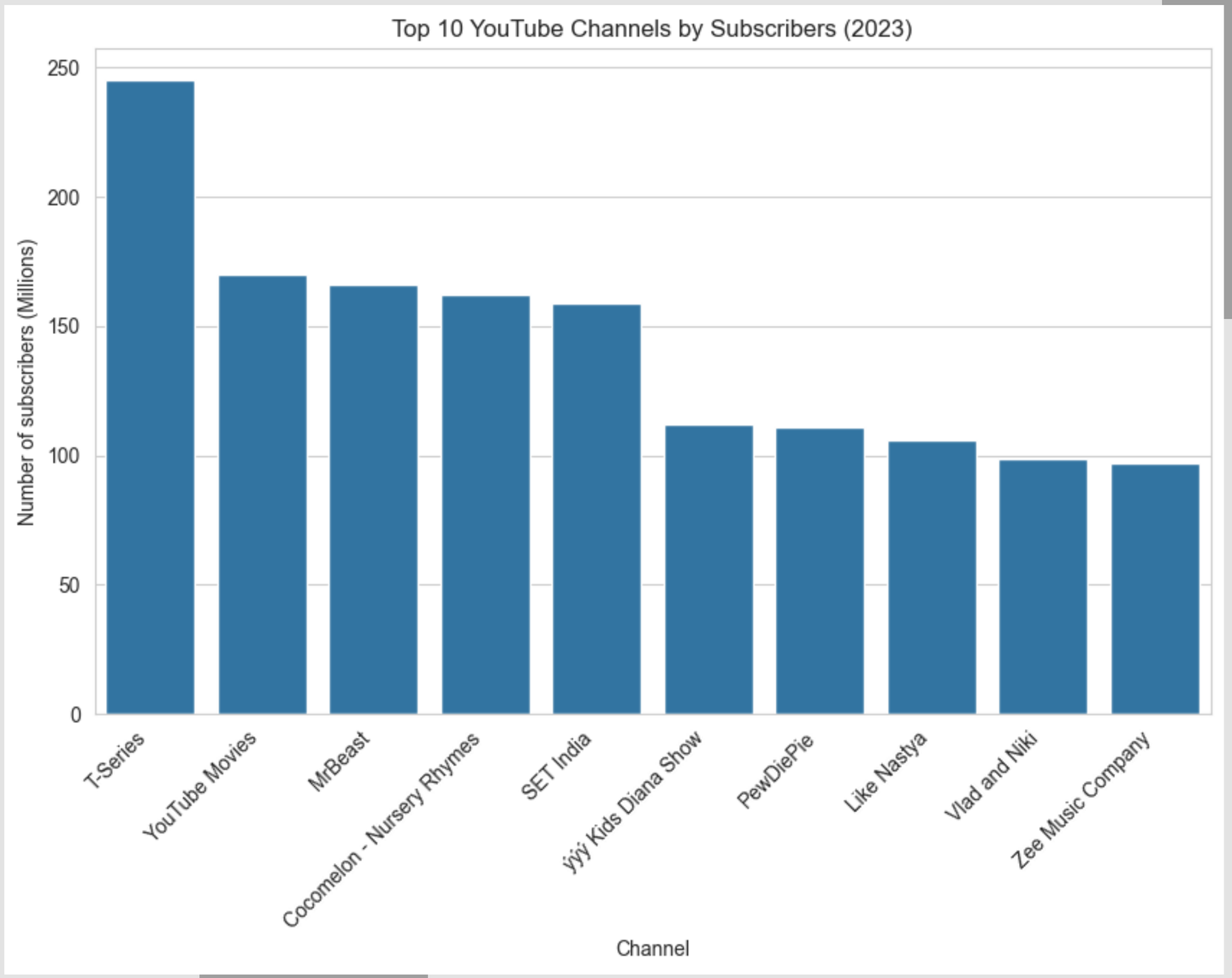
```
1  print(df.info())
```

✓ 0.0s

```
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   rank                                  995 non-null    int64
 1   Youtuber                              995 non-null    object
 2   subscribers                           995 non-null    int64
 3   video views                           995 non-null    float64
 4   category                              949 non-null    object
 5   Title                                 995 non-null    object
 6   uploads                               995 non-null    int64
 7   Country                               873 non-null    object
 8   Abbreviation                          873 non-null    object
 9   channel_type                          965 non-null    object
 10  video_views_rank                      994 non-null    float64
 11  country_rank                          879 non-null    float64
 12  channel_type_rank                     962 non-null    float64
 13  video_views_for_the_last_30_days      939 non-null    float64
 14  lowest_monthly_earnings               995 non-null    float64
 15  highest_monthly_earnings              995 non-null    float64
 16  lowest_yearly_earnings                995 non-null    float64
 17  highest_yearly_earnings               995 non-null    float64
 18  subscribers_for_last_30_days          658 non-null    float64
 19  created_year                          990 non-null    float64
 20  created_month                         990 non-null    object
 21  created_date                          990 non-null    float64
 22  Gross tertiary education enrollment (%)  872 non-null  float64
 23  Population                            872 non-null    float64
 24  Unemployment rate                     872 non-null    float64
 25  Urban_population                      872 non-null    float64
 26  Latitude                              872 non-null    float64
 27  Longitude                             872 non-null    float64
```

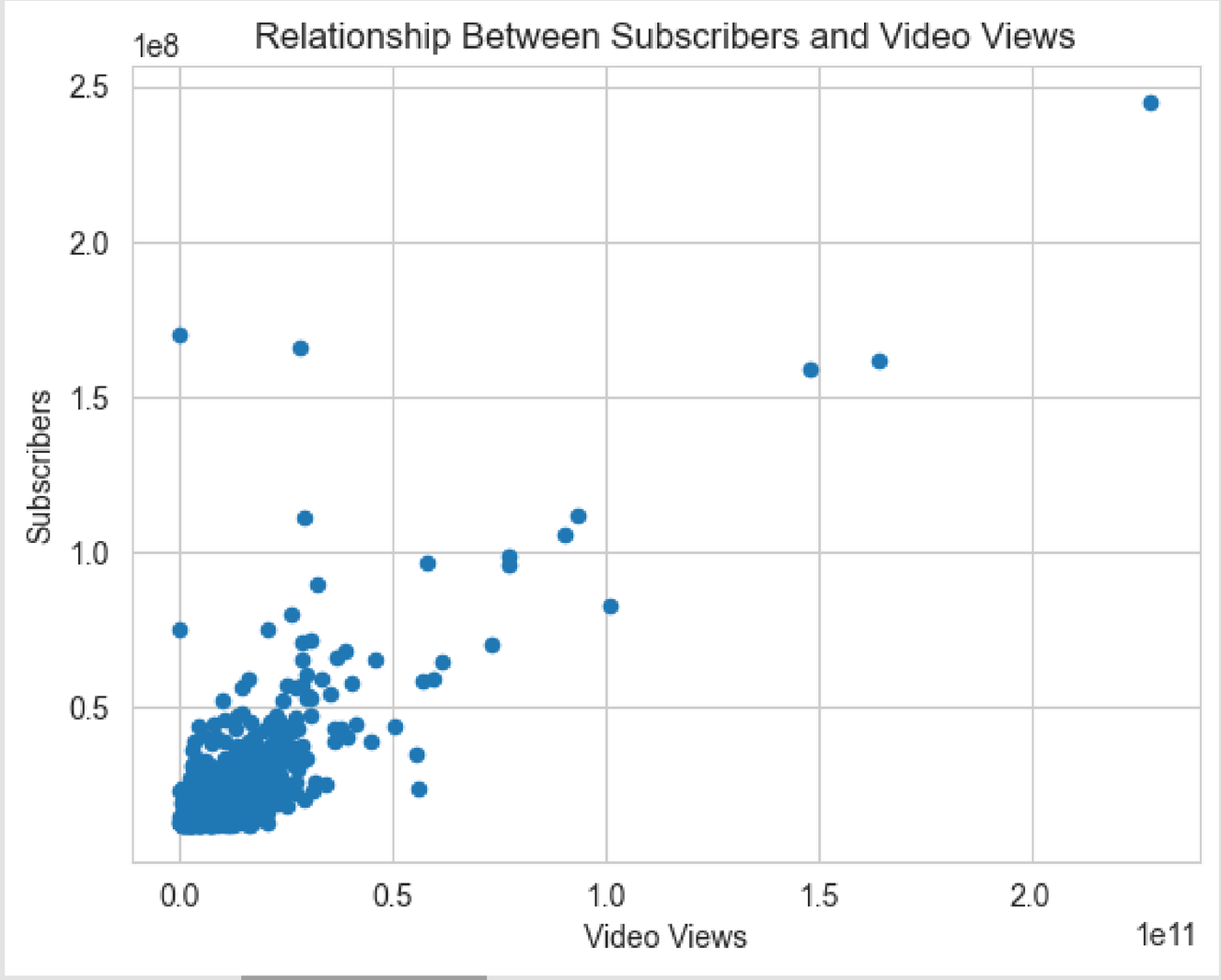
Exploratory Data Analysis (EDA) Result

Top 10 channels by subscribers



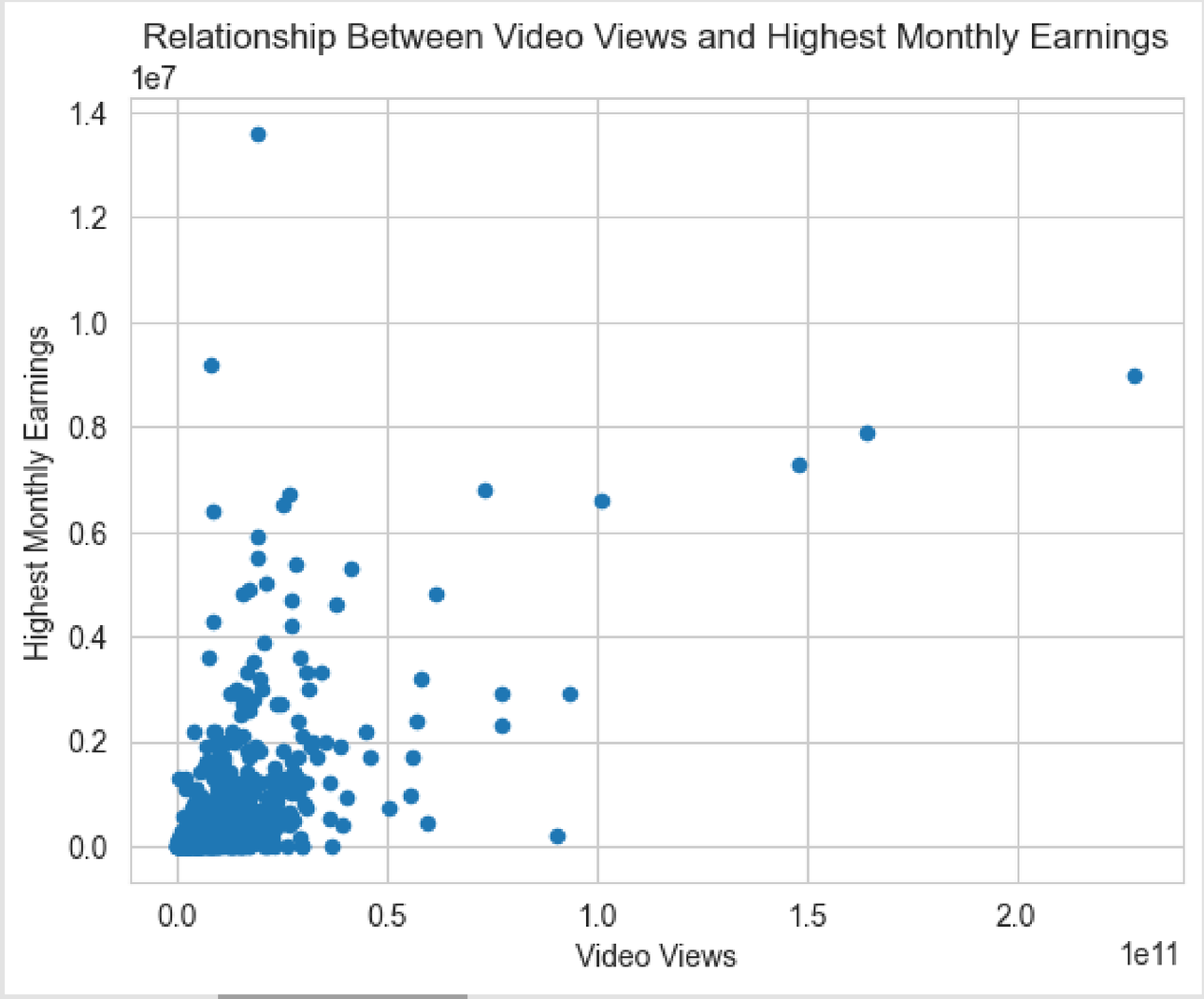
Exploratory Data Analysis (EDA) Result

Subscriber vs Video Views



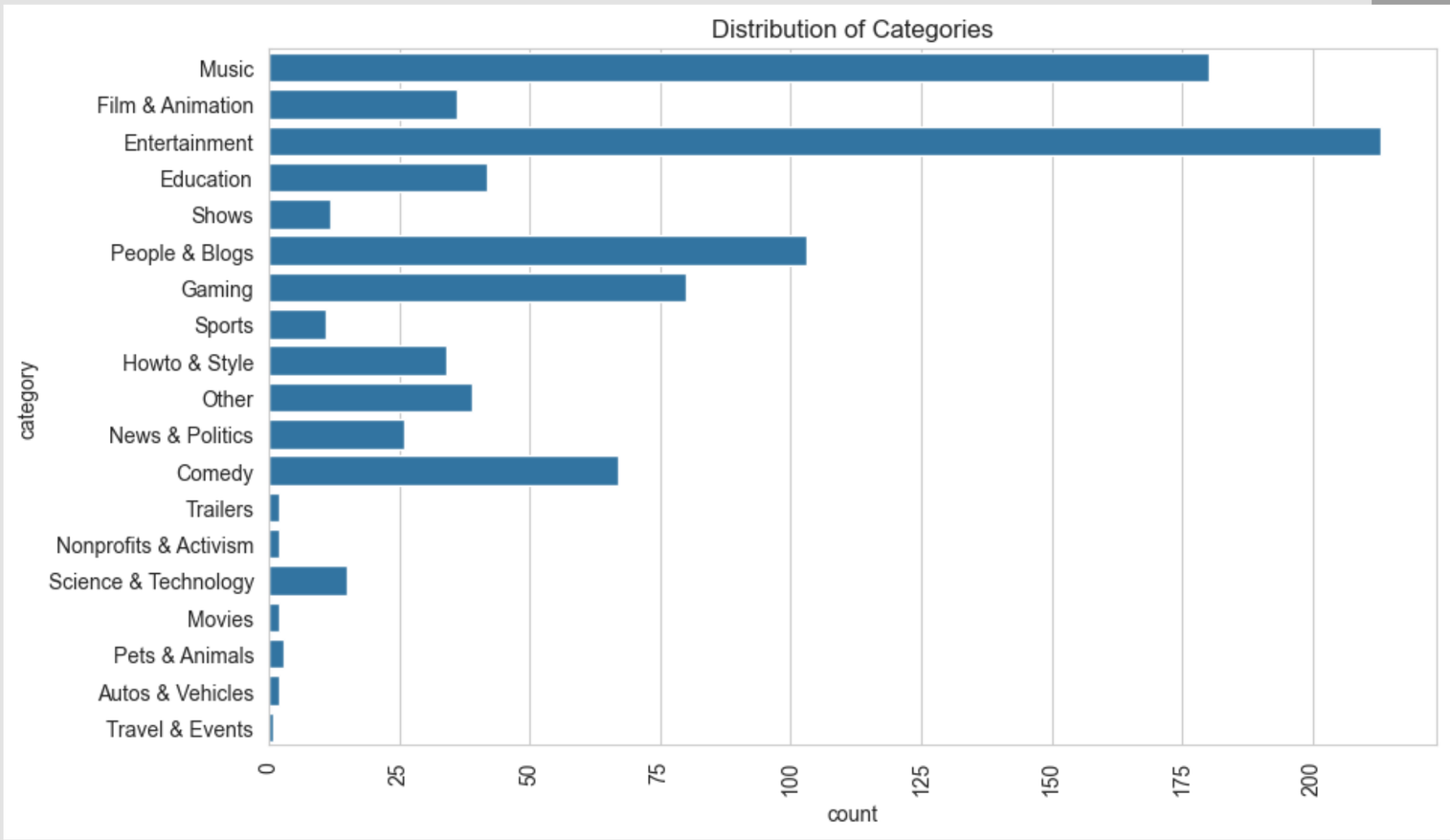
Exploratory Data Analysis (EDA) Result

Video Views vs Highest Monthly Earnings



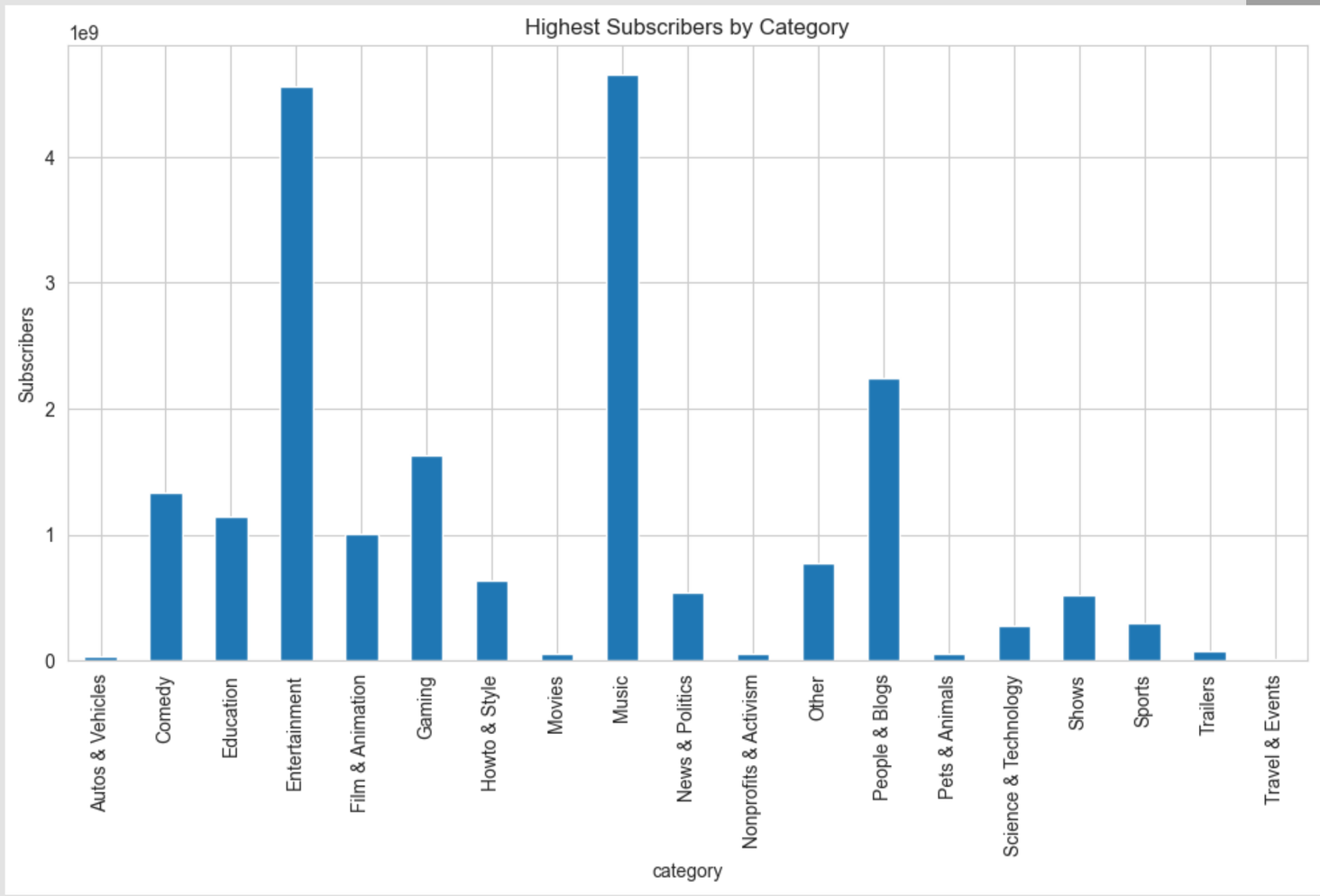
Exploratory Data Analysis (EDA) Result

Count of Youtube channels by categories



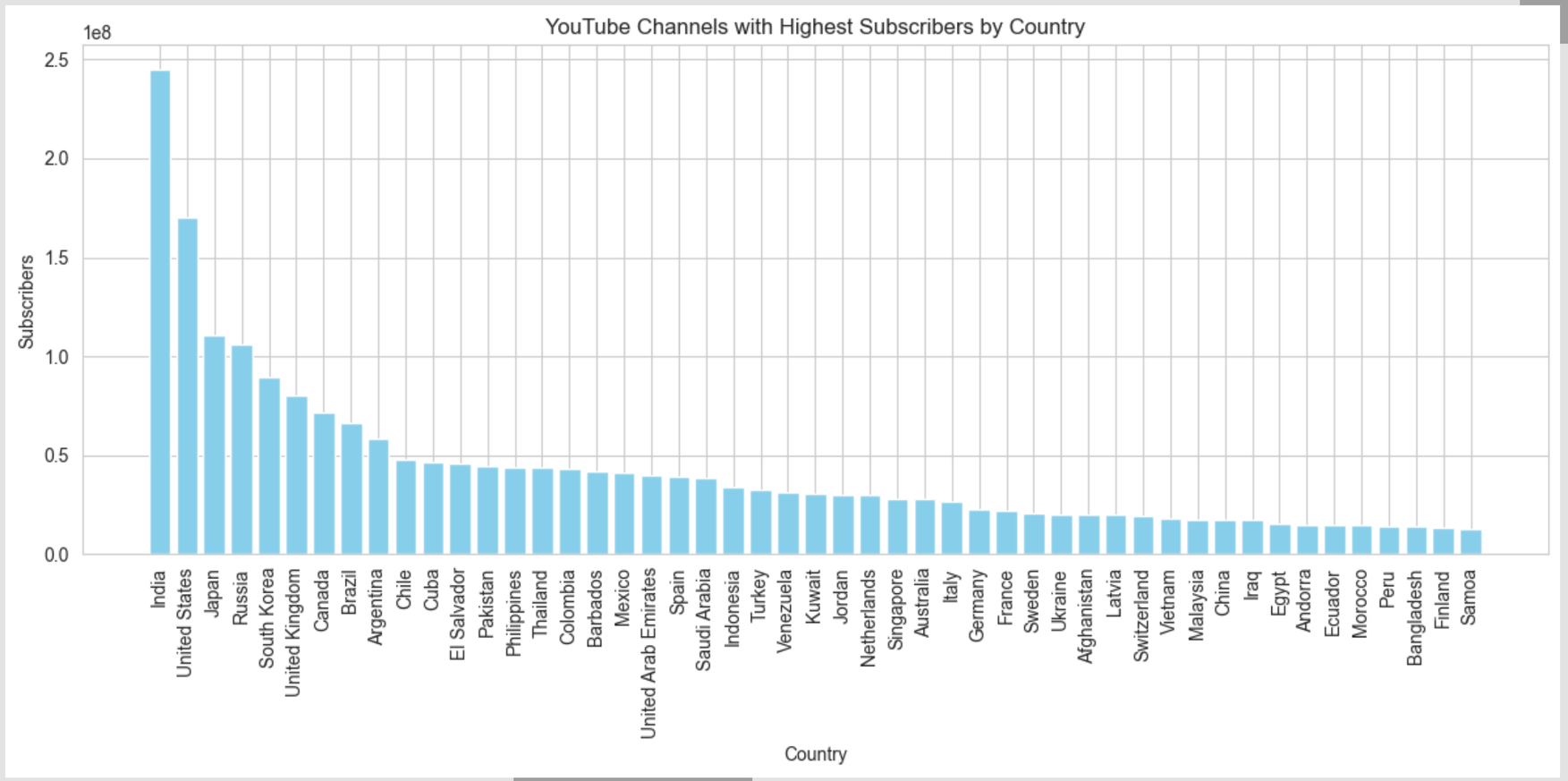
Exploratory Data Analysis (EDA) Result

Categories with the highest number of subscribers



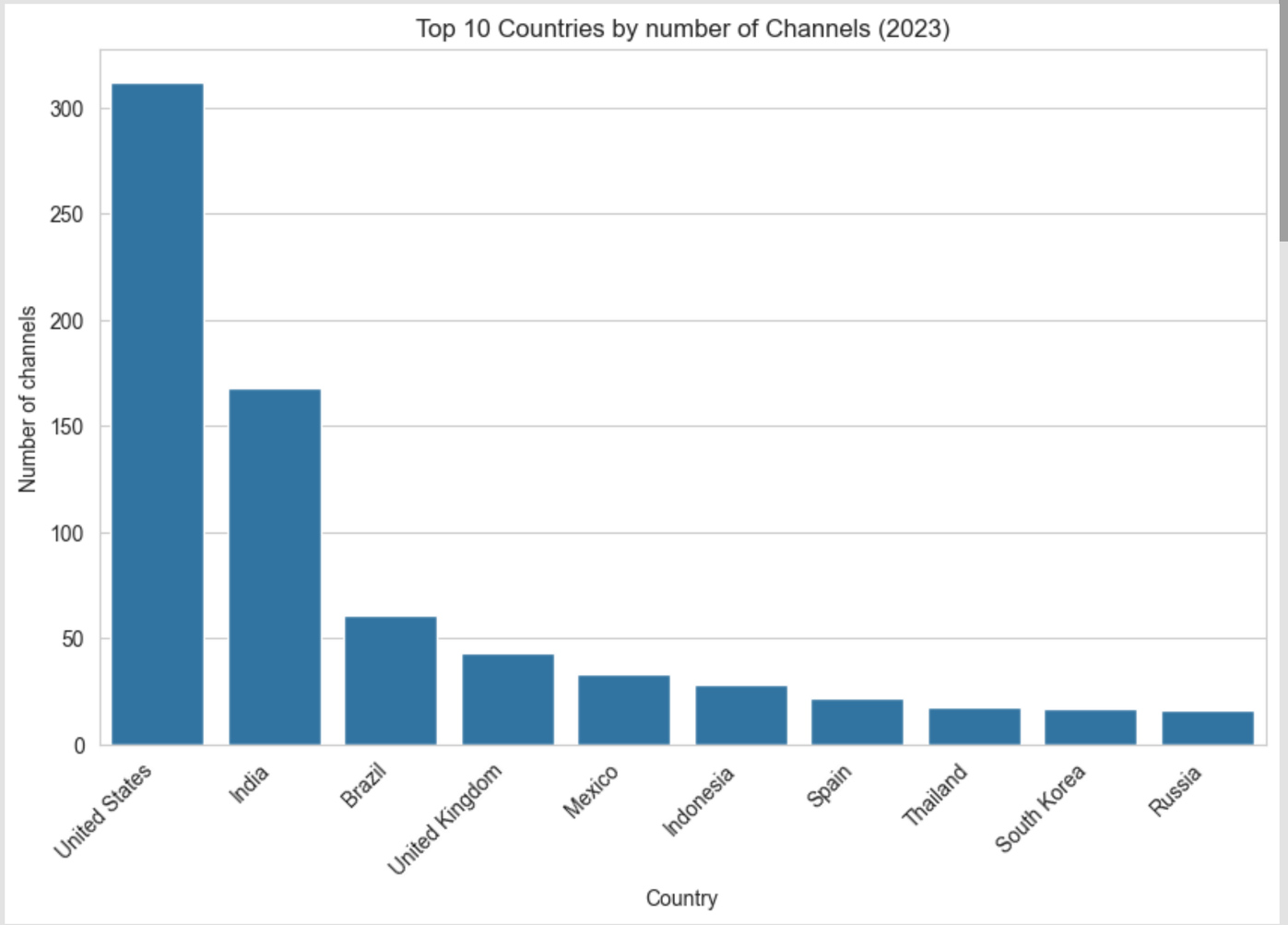
Exploratory Data Analysis (EDA) Result

Youtube Channels with the highest subscriber group by country



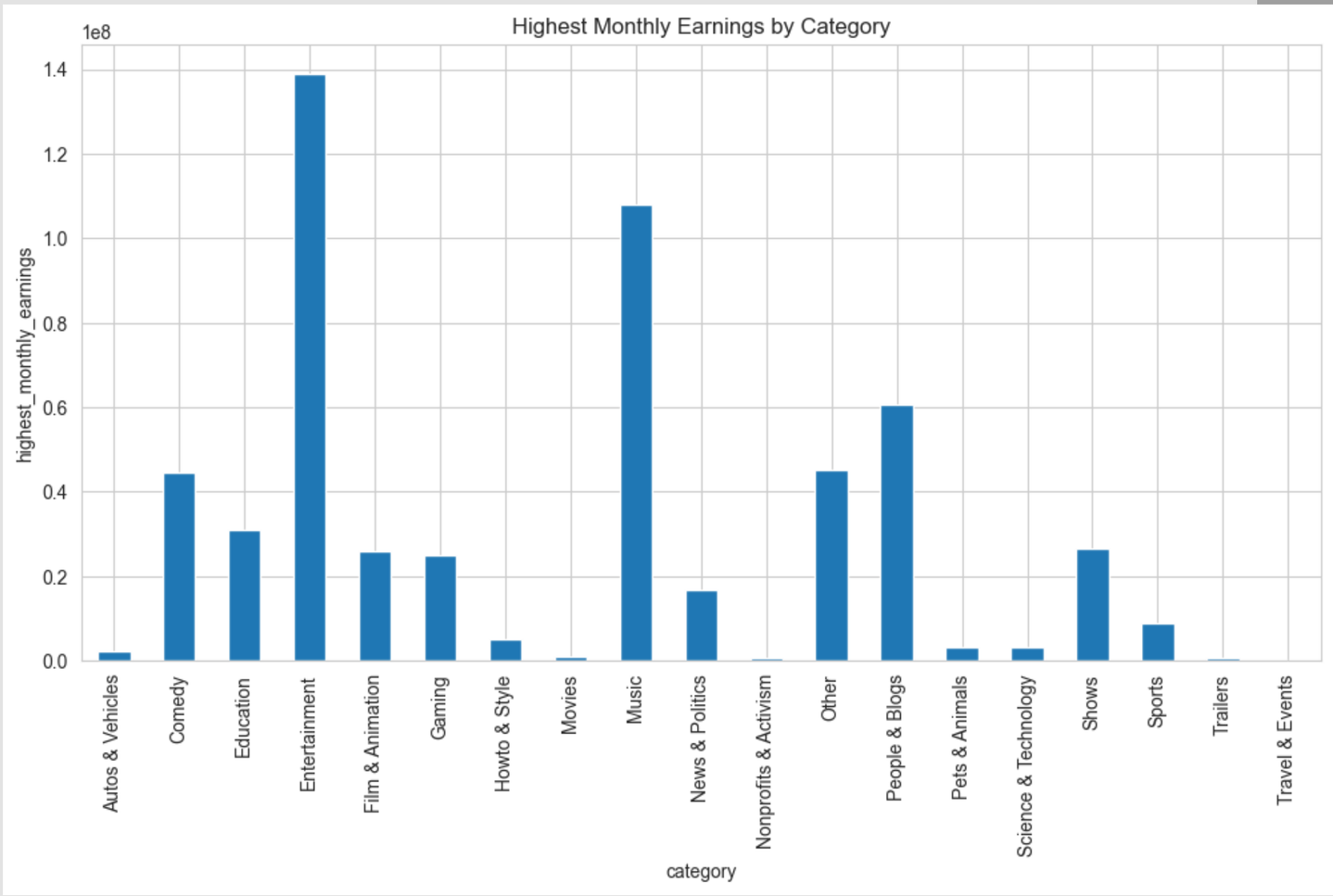
Exploratory Data Analysis (EDA) Result

Top 10 countries with highest number of channel

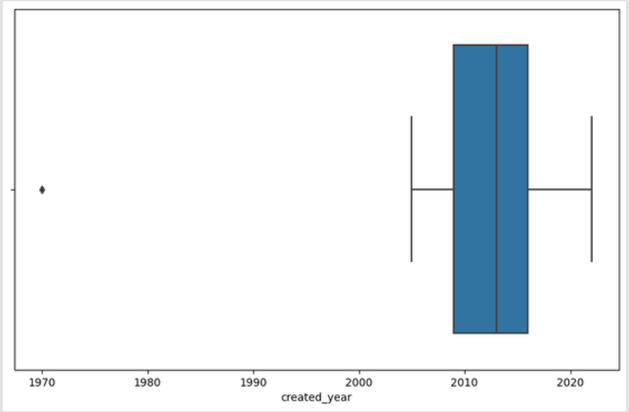


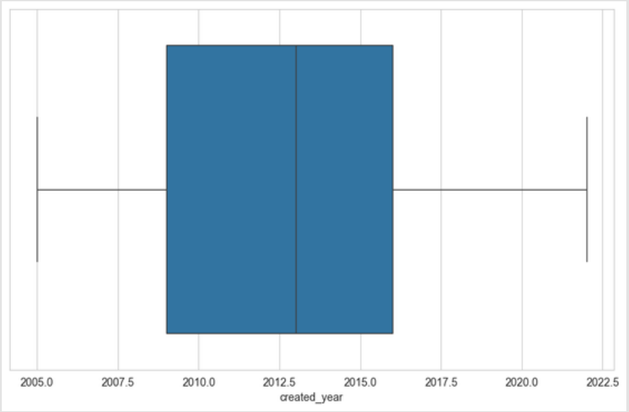
Exploratory Data Analysis (EDA) Result

Highest Earning Categories

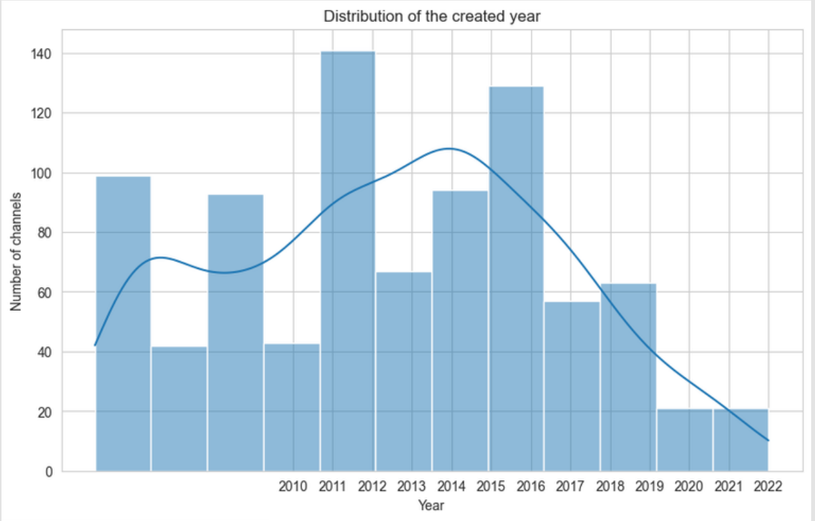


Exploratory Data Analysis (EDA) Result



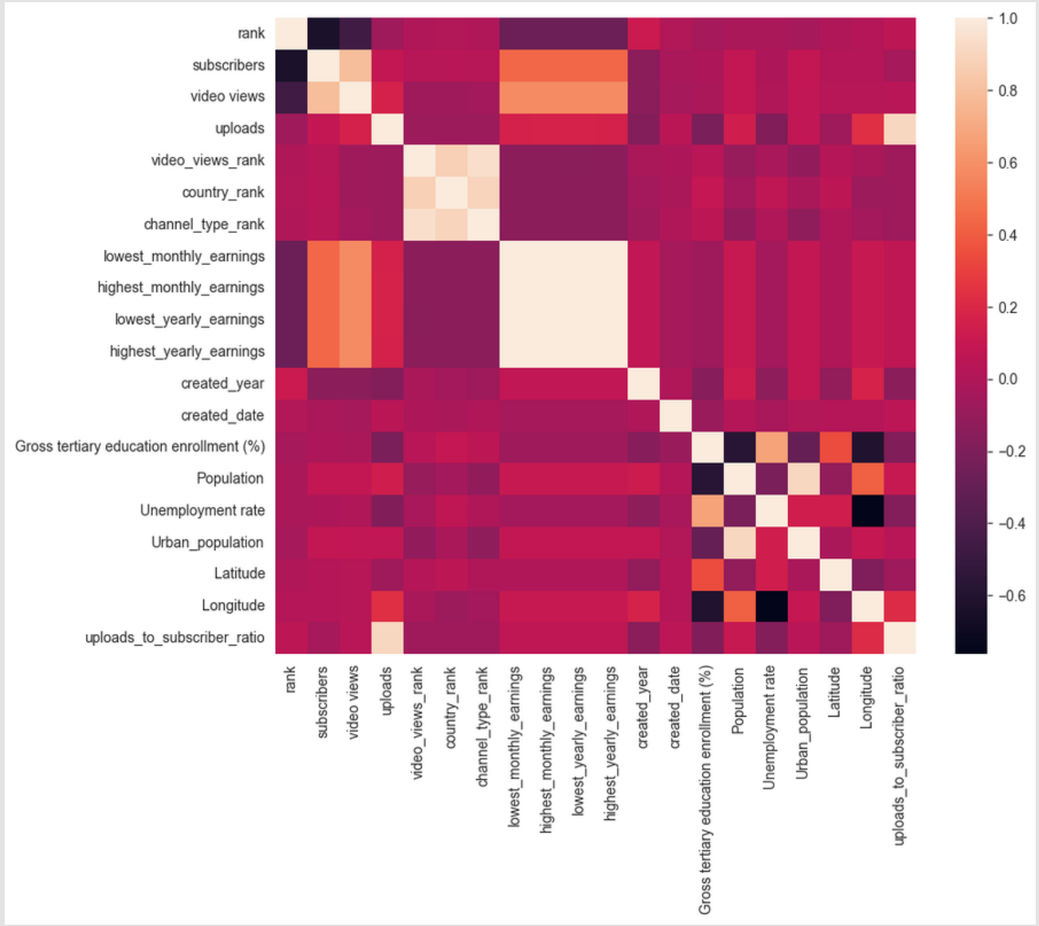


Distribution of the channel creation year



Exploratory Data Analysis (EDA) Result

Correlation of the dataset within the columns




```
[119]
    1  scope_video_views = correlation.loc[correlation['video views'].abs()>=0.1].index[:-1]
    2  print(scope_video_views)

...  Index(['rank', 'subscribers', 'video views', 'uploads',
            'lowest_monthly_earnings', 'highest_monthly_earnings',
            'lowest_yearly_earnings', 'highest_yearly_earnings'],
           dtype='object')
```

Model

Model 1
Target: Video Views
Feature: Subscribers, Uploads, Country, Channel types

Model 2
Target: Video Views
Feature: Subscribers, Uploads, Average monthly earning, Average monthly earning

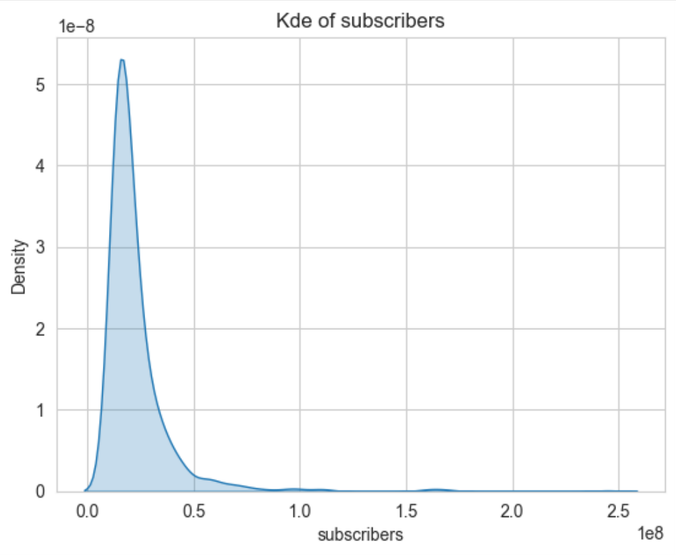
Model 3
Target: Video Views
Feature: Subscribers, Uploads, Average monthly earning, Average monthly earning , Country, Channel types

Model 4
Target: Subscribers
Feature: Video Views, Uploads, Average monthly earning, Average monthly earning

```
1  scope_video_views = correlation.loc[correlation['video views'].abs()>=0.1].index[:-1]
2  print(scope_video_views)
```

✓ 0.0s

```
Index(['rank', 'subscribers', 'video views', 'uploads',
       'lowest_monthly_earnings', 'highest_monthly_earnings',
       'lowest_yearly_earnings', 'highest_yearly_earnings'],
      dtype='object')
```



Numerical features

Simple Imputor

Min/Max Scaler

Categorical features

Simple Imputor

One hot encoder

Pipeline

Cross Validation

Linear Regression
Random Forest Regressor
Decision Tree Regressor

GridSerchCV

max_depth = 5,10,15
n_estimator = 5-15

Train/Test

Random Forest Regressor
check score

Pickle

Deploy on AWS

Result

TABLE II
MEAN SCORES FOR DIFFERENT REGRESSION MODELS

Regression Model	Mean Squared Error Score ($\times 10^{19}$)
Linear Regression	-4.97×10^{24}
Random Forest	-8.14
Decision Tree Regressor	-10.74

TABLE III
MEAN SCORES FOR DIFFERENT REGRESSION MODELS

Regression Model	Mean Squared Error Score ($\times 10^{19}$)
Linear Regression	-6.09
Random Forest	-6.48
Decision Tree Regressor	-12.48

TABLE IV
MEAN SCORES FOR DIFFERENT REGRESSION MODELS

Regression Model	Mean Squared Error Score ($\times 10^{19}$)
Linear Regression	-4.99×10^{24}
Random Forest	-7.15
Decision Tree Regressor	-10.72

TABLE V
MEAN SCORES FOR DIFFERENT REGRESSION MODELS

Regression Model	Mean Squared Error Score ($\times 10^{14}$)
Linear Regression	-1.41
Random Forest	-1.67
Decision Tree Regressor	-2.43

Result

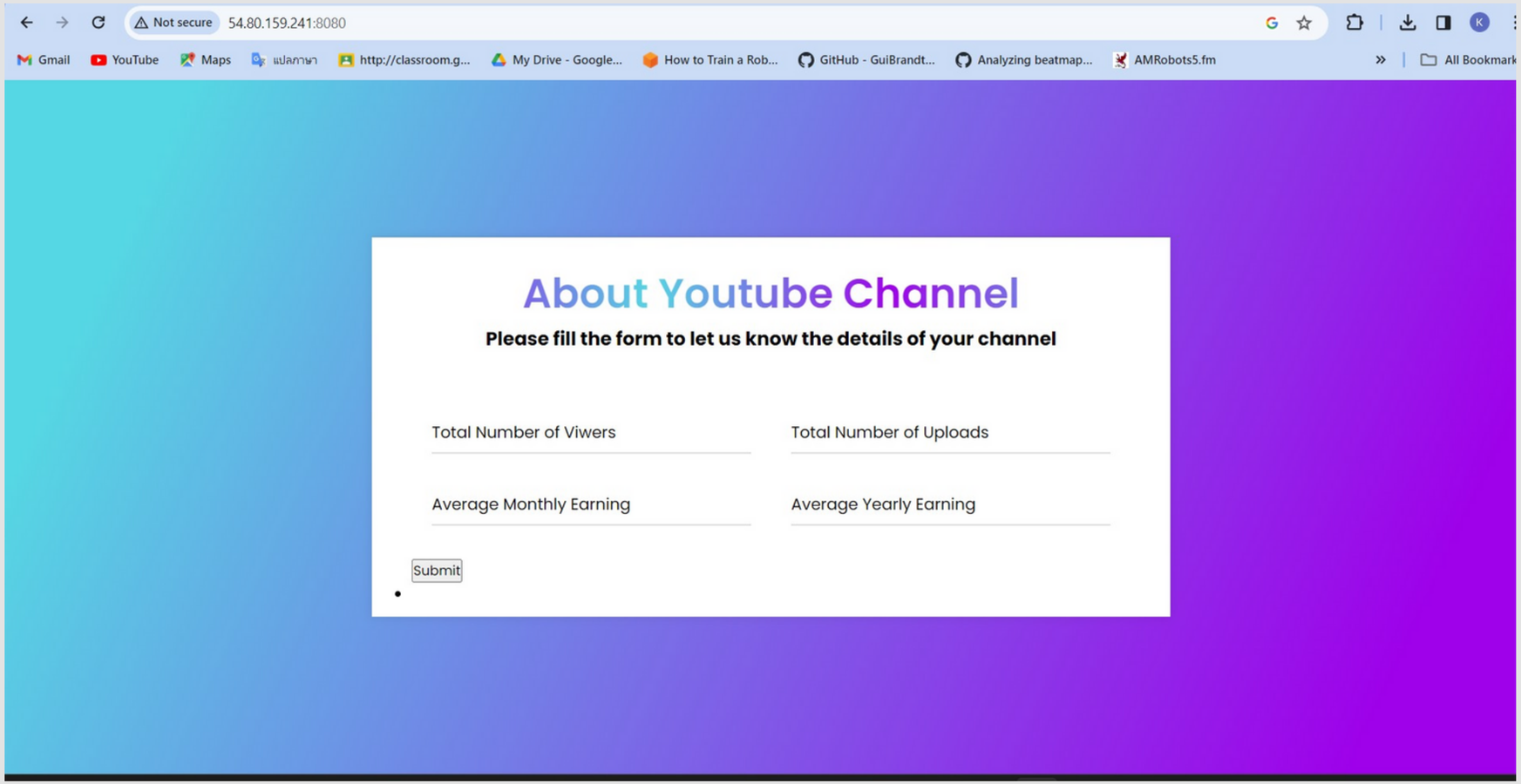
TABLE VI
HYPERPARAMETERS FOR RANDOM FOREST MODELS

Model	Regressor	max_depth	n_estimators
Model 1	RandomForestRegressor	5	6
Model 2	RandomForestRegressor	5	12
Model 3	RandomForestRegressor	10	11
Model 4	RandomForestRegressor	5	11

TABLE VII
MEAN SQUARE ERROR FOR DIFFERENT MODELS

Model	Target Variable	R^2	Mean Square Error ($\times 10^{14}$)
Model 1	Video Views	0.385	3.22×10^6
Model 2	Video Views	0.74	9.32×10^5
Model 3	Video Views	0.69	1.13×10^6
Model 4	Subscribers	0.71	1.80

Deployment


Not secure
54.80.159.241:8080
Gmail
YouTube
Maps
แปลภาษา
http://classroom.g...
My Drive - Google...
How to Train a Rob...
GitHub - GuiBrandt...
Analyzing beatmap...
AMRobots5.fm
All Bookmark
About Youtube Channel
Please fill the form to let us know the details of your channel
Total Number of Viwers
Total Number of Uploads
Average Monthly Earning
Average Yearly Earning
Submit

Conclusion

- Our extensive examination of the "Global YouTube Statistics 2023 Dataset" has yielded crucial findings on YouTube's dynamics in 2023.
- Utilizing exploratory data analysis, we identified trends including the prevalence of the "Entertainment" and "Music" categories
- a positive correlation between subscribers and video views, and the financial strength of top channels measured by monthly earnings.
- Employing machine learning, we developed a predictive model—specifically, the Random Forest model (Model 4)—which demonstrated impressive accuracy and reliability in forecasting YouTube metrics.
- The model, deployed on AWS as a user-friendly web application, allows users to input relevant data for predictions on subscribers. Beyond benefiting content creators, marketers, and YouTube enthusiasts, our project establishes a foundation for future research and applications in predictive analytics for online platforms.

Future Work

- Further refinement and optimization of machine learning models could enhance predictive accuracy.
- Integrating additional datasets, such as social media trends, cultural events, or technological advancements

Thank You