MOVIE BOX OFFICE GROSS PREDICTION

AN INDUSTRY ORIENTED MINI REPORT

Submitted to

JAWAHARLAL NEHRU TECNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING(AI&ML)

Submitted By

GOKUL NEREDUKOMMA	**21UK1A6699**
ANOOHYA RAYIDI	**21UK1A66B8**
SHRAVANI	**21UK1A6673**
SAI TEJA SADULA	**21UK1A66A8**

Under the guidance of

Mr. T. DHAYAKAR

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VAAGDEVI ENGINEERING COLLEGE

Affiliated to JNTUH, HYDERABAD

BOLLIKUNTA, WARANGAL (T.S) – 506005

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING(AI&ML)

VAAGDEVI ENGINEERING COLLEGE(WARANGAL)

CERTIFICATE OF COMPLETION
INDUSTRY ORIENTED MINI PROJECT

This is to certify that the UG Project Phase-1 entitled “MOVIE BOX OFFICE GROSS PREDICTION” is being submitted by GOKUL.NEREDUKOMMA(21UK1A6699),ANOOHYA.RAYIDI(21UK1A66B8), SHRAVANI.MANDA(21UK1A6673),SAITEJA.SADULA(21UK1A66A8) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering to Jawaharlal Nehru Technological University Hyderabad during the academic year 2023- 2024.

Project Guide	**HOD**
Mr.T.DHAYAKAR	**Dr. K. SHARMILA**
(Assistant Professor)	(Professor)

External

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr.P.PRASAD RAO,** Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this UG Project Phase-1 in the institute.

We extend our heartfelt thanks to **Dr.K.SHARMILA**, Head of the Department of CSE, Vaagdevi Engineering College for providing us necessary infrastructure and thereby giving us freedom to carry out the UG Project Phase-1.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the UG Project Phase-1 and for their support in completing the UG Project Phase-1.

We express heartfelt thanks to the guide, **T.SANATH KUMAR ,** Assistant professor, Department of CSE for his constant support and giving necessary guidance for completion of this UG Project Phase-1.

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

GOKUL NEREDUKOMMA(21UK1A6699),

ANOOHYA RAYIDI(21UK1A66B8),

SHRAVANI MANDA(21UK1A6673),

SAITEJA SADULA (21UK1A66A8) .

ABSTRACT

The accurate prediction of gross box-office markets is of great benefit for investment and management in the movie industry. In this work, we propose a machine learning-based method for predicting the movie box-office revenue of a country based on the empirical comparisons of eight methods with diverse combinations of economic factors. Specifically, we achieved a prediction performance of the relative root mean squared error of 0.056 in the US and of 0.183 in China for the two case studies of movie markets in time-series forecasting experiments from 2013 to 2016. We concluded that the support-vector-machine-based method using gross domestic product reached the best prediction performance and satisfies the easily available information of economic factors. The computational experiments and comparison studies provided evidence for the effectiveness and advantages of our proposed prediction strategy. In the validation process of the predicted total box-office markets in 2017, the error rates were 0.044 in the US and 0.066 in China. In the consecutive predictions of nationwide box-office markets in 2018 and 2019, the mean relative absolute percentage errors achieved were 0.041 and 0.035 in the US and China, respectively. The precise predictions, both in the training and validation data, demonstrate the efficiency and versatility of our proposed method.

TABLE OF CONTENTS:-

1.INTRODUCTION

1.1.OVERVIEW

Our project aims to predict movie box office gross using machine learning techniques. By leveraging historical data on various movie attributes such as genre, cast, budget, release date, and marketing strategies, we develop a predictive model to estimate a movie's potential box office performance. This predictive tool can assist movie production companies in making informed decisions regarding budget allocation, marketing strategies, and release timing to maximize revenue.

Real-Time Scenarios:

Budget Allocation Optimization:

A movie production company is planning its upcoming film projects for the year. They have a limited budget and want to allocate it optimally to maximize box office revenue. By utilizing our machine learning model, they can predict the potential gross revenue of different movie projects based on various factors such as genre, cast, and production budget. This helps them prioritize projects with higher revenue potential and allocate resources accordingly to ensure the best return on investment.

Marketing Strategy Enhancement:

A film studio is preparing to launch a new blockbuster movie and wants to devise an effective marketing strategy to maximize its box office success. Our machine learning model can analyze historical data on successful movie marketing campaigns, including social media engagement, trailer views, and promotional events, to predict the impact of different marketing strategies on the movie's box office performance. This enables the studio to tailor their marketing efforts to target the most promising audience segments and channels, ultimately driving higher ticket sales.

Release Date Optimization: A movie studio is planning the release date for its upcoming summer blockbuster and wants to choose the optimal timing to maximize box office revenue. By utilizing our predictive model, the studio can analyze historical data on movie release dates and seasonal trends to forecast the potential box office performance of the movie based on different release dates. This allows them to strategically schedule the movie's release to avoid competing with other major releases and leverage peak moviegoing periods, such as holidays or weekends, to attract larger audiences and increase ticket sales.

The culture industry is one of the most important and influential parts of the world economy . Movies are a The popular relaxation method for young and diverse-aged people in all individual countries. The development status of the movie industry specifically reflects the consumption patterns and economic growth of a country . During the modernization processes, the movie industry is expected to play increasingly crucial roles in the capital and commerce of cultural economics due to its popularity. The total box-office revenue of a country is a valuable index of investment and finance for decision-makers, such as producers and moviemakers. In this sense, early prediction of box-office incomes is very beneficial for movie market-related capitalization and data-driven administration.

Accurate prediction of movie box-offices is still significant and challenging economically due to its extreme dynamics. The complexity is highly related to the entropy of the underlying cultural-economic system,especially in a time-series study over a temporal range . Due to its importance, Litman proposed a prediction study for the financial success of theatrical movies as early as 1983. He focused on various aspects of the movie industry, such as the marketing and distribution pattern and cinema arrangement. From the perspective of prediction methods, some studies so far have been proposed to meet this challenge , although most of them have not reached satisfying prediction performances and applicable practices. The econometric methods, e.g., linear regression and log-linear regression, have been introduced to box-office prediction. Elberse and Eliashberg proposed such a method for a motion picture in domestic markets and in foreign markets by examining the interrelationship between the different determinants of box-office revenue . The machine learning-based methods, such as neural networks, have been employed to forecast the box-office revenue of a movie before its theatrical release . Ahmed et al. built up such a prediction system using hybrid methods for box-office success quotient forecasting . Recently, some studies have been proposed for the instant prediction of the success of a movie at the box-office based on the big data of social media , Google search and Wikipedia page activity . However, these available methods focus only on the box-office success of one movie. There are still few studies for predicting the gross movie market of a country . Moreover, the predictor variables in these methods consider few effects of the economic factors. Box-office markets have very complicated relationships with many factors, ranging from macroeconomics of gross domestic product, per capita income, number of theater screen and online-storytelling media to the director, screenwriter, costars, moviegoers and consumption traditions . Effective predictions need to be conducted using reasonable and easily-accessible predictor variable(s). Furthermore, the prediction method needs to consider the time-course data of box-offices rationally and thoroughly. The effective and efficient prediction of gross box-office revenues in one country will provide a deep understanding and valuable direction of investments in the economy and management of the culture industry.

To this end, we aim to propose a machine learning method for predicting a nationwide box-office market through its associated economic factors. To select the prediction algorithm, we investigated various machine learning-based methods, e.g., support vector machines, random forest and neural networks and econometrics-based models (e.g., linear, ridge and auto regressions), to acquire the relationship between these economic indices (predictor variables) and the box-office (response variable). When selecting the prediction variables, we encoded the box-office markets using various predictor variables, e.g., gross domestic product (GDP) and the number of movie screens (NMS) and identified the most reliable and easily-available indices for practical predictions. For proof of concept, we empirically tested our prediction performances using a sliding-window encoding strategy for these time-series studies of box-offices in the US and China. We compared multiple methods with alternative dependent variables to identify the most efficient prediction strategy. We concluded that the gross box-office markets are predicable using a support vector machine with GDP in the case study of two countries. Furthermore, the accurate predictions over the past several years from 2017 to 2019, prior to the COVID-19 pandemic, provide more evidence for the effectiveness and advantage of our proposed prediction method.

For clarification, the contributions and novelties of this paper are summarized as follows:

- We propose an SVM-based method to predict the global box-office market of a country by its economic factor of GDP.

- We implemented four machine learning methods and four econometric methods with diverse combinations of economic factors as prediction variables. The comparison results in both the US and China box-office markets highlight the selected prediction strategy according to prediction performances.

- The time-series cross-validation and the mimicked prediction of the box-office market in real application scenarios prove the effectiveness and efficiency of our proposed method of predicting nationwide box-offices. The easy availability of economic factors also implies its flexibility.

- The empirical experiments with different combinations of economic factors indicate their diverse effects on box-office prediction. The selected prediction variable of GDP proves its interpretable close relationship with box-office revenues.

1.2.PURPOSE

Project Objectives

By the end of this project:

You’ll be able to understand the problem to classify if it is a regression or a classification kind of problem.

You will be able to know how to pre-process/clean the data using different data pre-processing techniques.

You will able to analyze or get insights of data through visualization.

Applying different algorithms according to the dataset and based on visualization.

You will able to know how to find the accuracy of the model.

You will be able to know how to build a web application using the Flask framework.

Materials and methods

Data-

To implement our prediction methods, we collected the statistical data of the two largest movie markets, i.e., the US and China, from a variety of data sources. Due to a reform policy released in 2001 in China after joining the WTO (World Trade Organization), the operational management of movies and their related release policies are significantly different from the previous era. Thus, we collected data from the box-office markets in these two counties from 2002 onwards. In the prediction models, the box-office refers to the outcome or response variable. For easy accessibility, we collected GDP and NMS in each country as the predictor or dependent variable. In detail, the box-office and the NMS data of the US were collected from NATO (National Association of Theatre Owners) and the GDP data of the US were obtained from the IMF (International Monetary Fund). The corresponding values of China were obtained from the National Bureau of Statistics of China and was adjusted to the same units and scale of the data from the US.

It is worth noting that the COVID-19 pandemic is currently threatening all human health. Most countries are still implementing social distancing and movie theaters are still shut down. That is to say, movie box-offices as well as current economic factors are not in their normal states. We began this project before the pandemic. Thus, we performed our study on the data collected before the ongoing pandemic caused by the SARS-CoV-2 virus. The proposed strategy will be still valid when the world economy and consumption recovers from the pandemic.

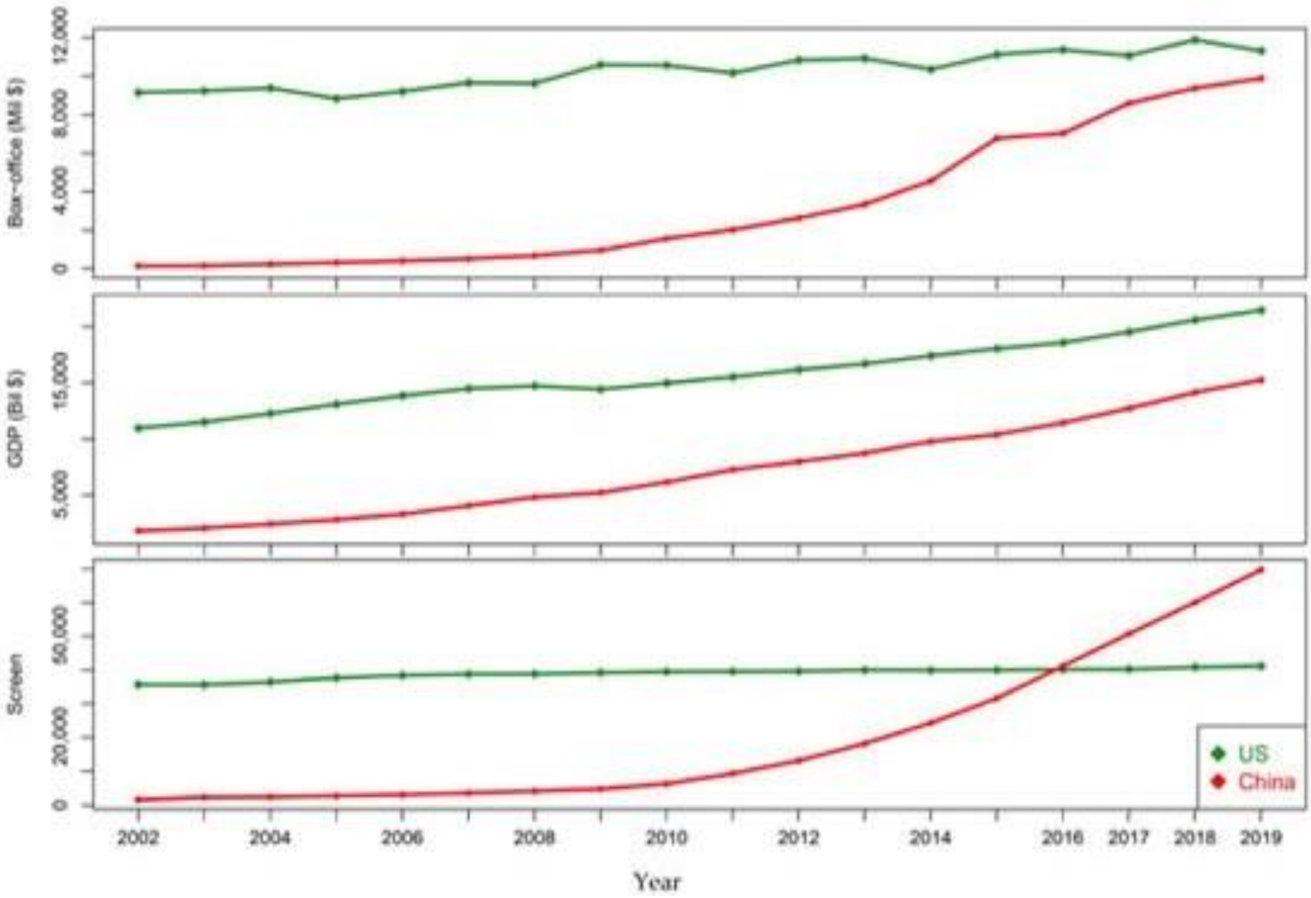


Figure 1. The box-offices over the period 2002 to 2019 in US and China. The corresponding GDP and NMS are also illustrated simultaneously .

Figure 1 records the gross movie box-office markets from 2002 to 2019 in the US and China. In total, we collected 18 continuous time-series data of movie box-offices, GDP, and NMS for the two countries. The task here is to predict the time-series of box-offices using these economic factors. We used the sequential data from 2002 to 2016 for the training and selection the prediction methods and used the data from 2017 to 2019 for testing and validation purposes.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

Predicting movie box office gross accurately can be challenging due to several factors:

1. **Audience Preferences and Trends**: Audience tastes are constantly evolving, making it difficult to predict which genres, actors, or styles will resonate at any given time.
2. **Competition**: The number and strength of competing movies released around the same time can significantly impact a movie's box office performance.
3. **Marketing Efforts**: Effective marketing campaigns can influence audience turnout, but the success of these efforts can be unpredictable.
4. **Critical Reception**: Reviews and word-of-mouth play a crucial role in a movie's success. Predicting how critics and audiences will react can be uncertain.
5. **Economic Factors**: Economic conditions can affect discretionary spending on entertainment, impacting box office numbers.
6. **Seasonality**: Certain times of the year (e.g., summer, holidays) tend to attract more moviegoers, but this can vary based on cultural and regional factors.
7. **Franchise and Brand Strength**: Established franchises or well-known brands may have built-in audiences, but this isn't always a guarantee of success.
8. **Platform Diversity**: With the rise of streaming platforms, predicting how a movie will perform in theaters versus on-demand can be complex.
9. **Global Market Variability**: Movies often have different release dates and strategies across international markets, each with unique audience preferences and economic conditions.
10. **Unforeseen Events**: Unexpected events, such as scandals involving cast or crew, can impact a movie's reception and box office performance.

These factors illustrate why predicting movie box office grosses requires a nuanced understanding of both industry trends and unpredictable human behavior.

2.2 PROPOSED SOLUTION

Predicting movie box office gross accurately is a complex task, but several approaches can enhance prediction models:

1. Data-driven Analysis: Utilize historical box office data combined with factors such as genre, cast, director, release date, marketing budget, and critical reception. Machine learning models like regression, random forests, or neural networks can learn from this data to make predictions.

2. Sentiment Analysis: Analyze social media, reviews, and audience sentiment leading up to and following a movie's release to gauge public interest and reception.

3. Market Segmentation: Segment audiences based on demographics, geographical location, and viewing habits to understand which segments are likely to drive box office sales for specific types of movies.

4. Competitive Analysis: Evaluate the competitive landscape by considering other movies releasing around the same time, their genres, and anticipated audience overlap.

5. Economic Indicators: Incorporate economic factors such as consumer spending trends, disposable income, and general economic conditions that may influence moviegoing behavior.

6. Marketing Analytics: Measure the effectiveness of marketing campaigns through metrics like social media engagement, trailer views, and pre-release ticket sales to forecast initial box office performance.

7. Cross-platform Analysis: Consider the impact of simultaneous releases across different platforms (theaters, streaming services) on overall box office revenue.

8. Collaborative Filtering: Apply techniques from recommendation systems to predict box office success based on similarities with past successful movies in terms of genre, cast, director, etc.

9. Ensemble Methods: Combine predictions from multiple models or sources (e.g., expert opinions, statistical models, machine learning algorithms) to improve accuracy and reliability.

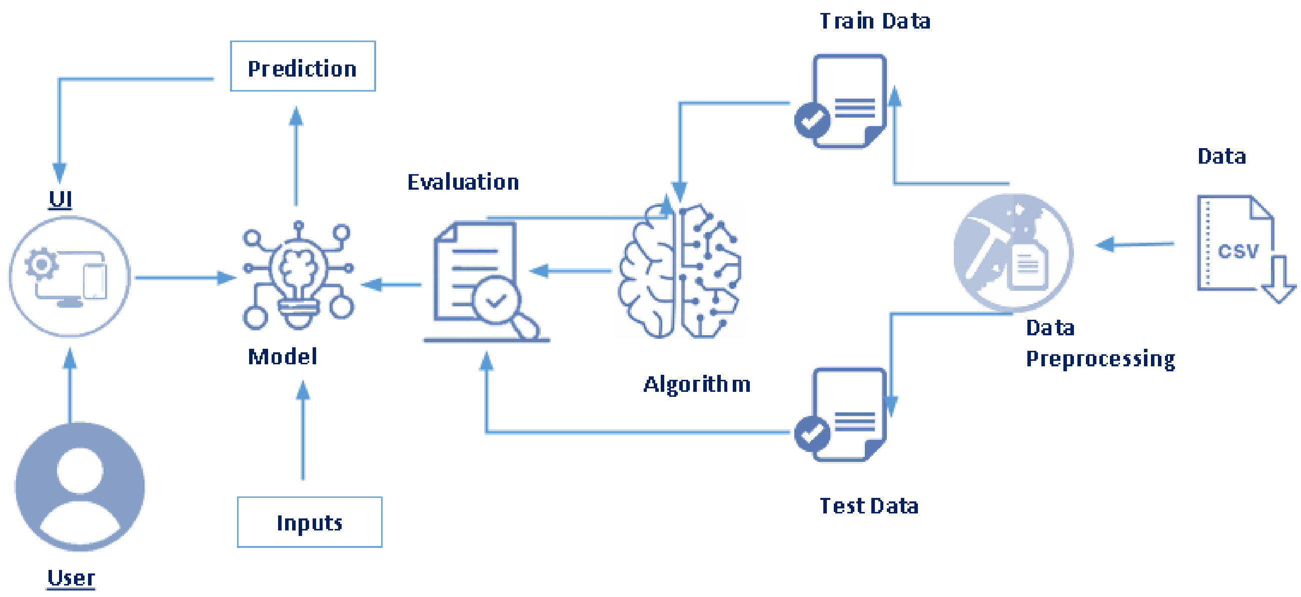
10. Continuous Learning: Adapt models over time with new data and feedback to refine predictions and account for changing audience preferences and market dynamics.

By integrating these approaches, movie studios, analysts, and researchers can develop more robust models for predicting box office gross, thereby improving decision-making and resource allocation in the film industry.

3.THEORITICAL ANALYSIS

3.1. BLOCK DIAGRAM

Architecture:



Creating a block diagram for predicting movie box office gross involves breaking down the process into key components and their interactions. Here’s a simplified block diagram that outlines the major steps involved:

1. Data Collection:

- Historical Data: Gather past box office gross data for movies.

- Movie Details: Collect information about each movie (genre, cast, director, budget, release date, etc.).

- External Factors: Include relevant external data (competition, holidays, economic factors, etc.).

2. Data Preprocessing:

- Cleaning: Handle missing values, outliers, and inconsistencies in the data.

- Normalization/Scaling:Standardize numerical features.

- Feature Engineering: Create new features or transform existing ones (e.g., extract release month from release date).

3.Feature selection:

- Identify the most relevant features that contribute to predicting box office gross.

4. Model Selection:

- Regression Models:Choose appropriate regression algorithms (e.g., linear regression, random forest, gradient boosting).

- Neural Networks:Consider deep learning models if complex patterns are expected.

5. Training:

- Splitting Data: Divide data into training and validation sets.

- Model Training: Train the selected models on the training data.

6. Model Evaluation:

- Validation: Assess model performance using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), etc.

- Cross-Validation: Validate model robustness using techniques like k-fold cross-validation.

7. Prediction:

- Input New Data: Input features of a new movie.

- Model Prediction:Use the trained model to predict the box office gross.

8. Deployment:

- Integrate the predictive model into a production environment.

- Continuously update the model with new data for improved accuracy.

9. Feedback Loop:

- Incorporate user feedback and model performance metrics to refine the prediction process.

Each block in this diagram represents a stage or process in the workflow of predicting movie box office gross. Adjustments can be made based on specific requirements or additional factors considered important in your prediction model.

3.2. SOFTWARE DESIGNING

The following is the Software required to complete this project:

- **Google Colab**: Google Colab will serve as the development and execution environment for your predictive modeling, data preprocessing, and model training tasks. It provides a cloud-based Jupyter Notebook environment with access to Python libraries and hardware acceleration.

- **Dataset (CSV File)**: The dataset in CSV format is essential for training and testing your predictive model. It should include historical air quality data, weather information, pollutant levels, and other relevant features.
- **Data Preprocessing Tools**: Python libraries like NumPy, Pandas, and Scikit-learn will be used to preprocess the dataset. This includes handling missing data, feature scaling, and data cleaning.
- **Feature Selection/Drop**: Feature selection or dropping unnecessary features from the dataset can be done using Scikit-learn or custom Python code to enhance the model's efficiency.
- **Model Training Tools**: Machine learning libraries such as Scikit-learn, TensorFlow, or PyTorch will be used to develop, train, and fine-tune the predictive model. Regression or classification models can be considered, depending on the nature of the AQI prediction task.
- **Model Accuracy Evaluation**: After model training, accuracy and performance evaluation tools, such as Scikit-learn metrics or custom validation scripts, will assess

the model's predictive capabilities. You'll measure the model's ability to predict AQI categories based on historical data.

- **UI Based on Flask Environment**: Flask, a Python web framework, will be used to develop the user interface (UI) for the system. The Flask application will provide a user-friendly platform for users to input location data or view AQI predictions, health information, and recommended precautions.
- Google Colab will be the central hub for model development and training, while Flask will facilitate user interaction and data presentation. The dataset, along with data preprocessing, will ensure the quality of the training data, and feature selection will optimize the model. Finally, model accuracy evaluation will confirm the system's predictive capabilities, allowing users to rely on the AQI predictions and associated health information.

4.EXPERIMENTAL INVESTIGATION

ProjectWorkFlow:

- The user interacts with the UI (User Interface) to upload the input features.
- Uploaded features/input is analyzed by the model which is integrated.
- Once the model analyses the uploaded inputs, the prediction is showcased on the UI.

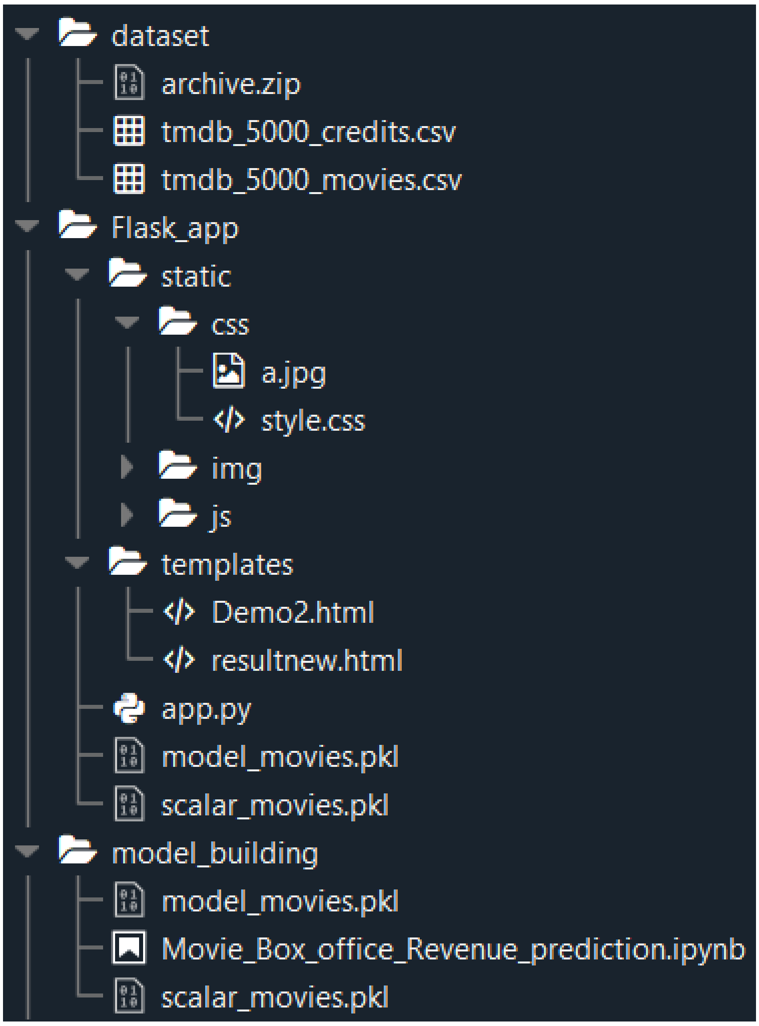
To accomplish this, we have to complete all the activities and tasks listed below Tasks:

- Data Collection.
  - Collect the dataset or Create the dataset
- Data Preprocessing.
  - Import the Libraries.
  - Reading the dataset.
  - Exploratory Data Analysis
  - Converting json objects to strings
  - Checking for Null Values.
  - Data Visualization.
  - Dropping the columns
  - Label Encoding

    - Splitting the Dataset into Dependent and Independent variable.
    - Feature scaling
    - Splitting Data into Train and Test.
- Model Building
    - Training and testing the model
    - Evaluation of Model
    - Save the model
    - Predicting the output using the model
- Application Building
    - Create an HTML file
    - Build a Python Code
    - Run the app

Project Structure

Your project folder looks like the below image. create all the files and assign them in the similar manner



```
dataset
    archive.zip
    tmdb_5000_credits.csv
    tmdb_5000_movies.csv
Flask_app
    static
        css
            a.jpg
            style.css
        img
        js
    templates
        Demo2.html
        resultnew.html
    app.py
    model_movies.pkl
    scalar_movies.pkl
model_building
    model_movies.pkl
    Movie_Box_office_Revenue_prediction.ipynb
    scalar_movies.pkl
```

- We have three folders dataset, Flask_app, and model_building
- A python file called app.py for server-side scripting.
- Templates folder which contains Demo2.HTML and resultnew.html files.
- The static folder which contains CSS folder which contains styles.css.

5.FLOWCHART

Creating a flowchart for predicting movie box office gross involves outlining the sequential steps and decisions involved in the process. Here’s a flowchart that illustrates the typical workflow.

Start

|

v

Data Collection

|

v

Data Preprocessing

|

v

Feature Selection

|

v

Model Selection

|

v

Training

|

v

Model Evaluation

|

v

Prediction

|

v

Output: Predicted Box Office Gross

|

v

End

Let's break down each step in detail:

1. Data Collection:

   - Gather historical data on movie box office gross.
   - Collect details about movies (genre, cast, director, budget, release date, etc.).
   - Include external factors (competition, holidays, economic conditions, etc.).

2. Data Preprocessing:

   - Clean the data (handle missing values, outliers, etc.).
   - Normalize or scale numerical features.
   - Perform feature engineering (create new features or transform existing ones).

3. Feature Selection:

   - Identify the most relevant features that influence box office gross.

4. Model Selection:

   - Choose a suitable prediction model (e.g., regression models like linear regression, decision trees, ensemble methods like random forests or gradient boosting, neural networks).

5. Training:

   - Split the data into training and validation sets.
   - Train the selected model using the training data.

6. Model Evaluation:

   - Evaluate model performance using metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), or others.
   - Validate the model's accuracy and reliability.

7. Prediction:

   - Input the features of a new movie into the trained model.
   - Obtain the predicted box office gross for the new movie.

8. Output:

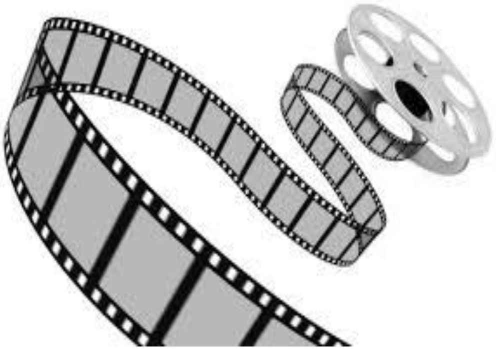
- Display or store the predicted box office gross.

9. End:

- End of the flowchart.

This flowchart outlines the sequential steps involved in predicting movie box office gross, from data collection to model evaluation and prediction. It provides a structured approach to understand the flow of the prediction process and the decision points along the way. Adjustments can be made based on specific modeling techniques or additional factors considered in the prediction model.

6.RESULT

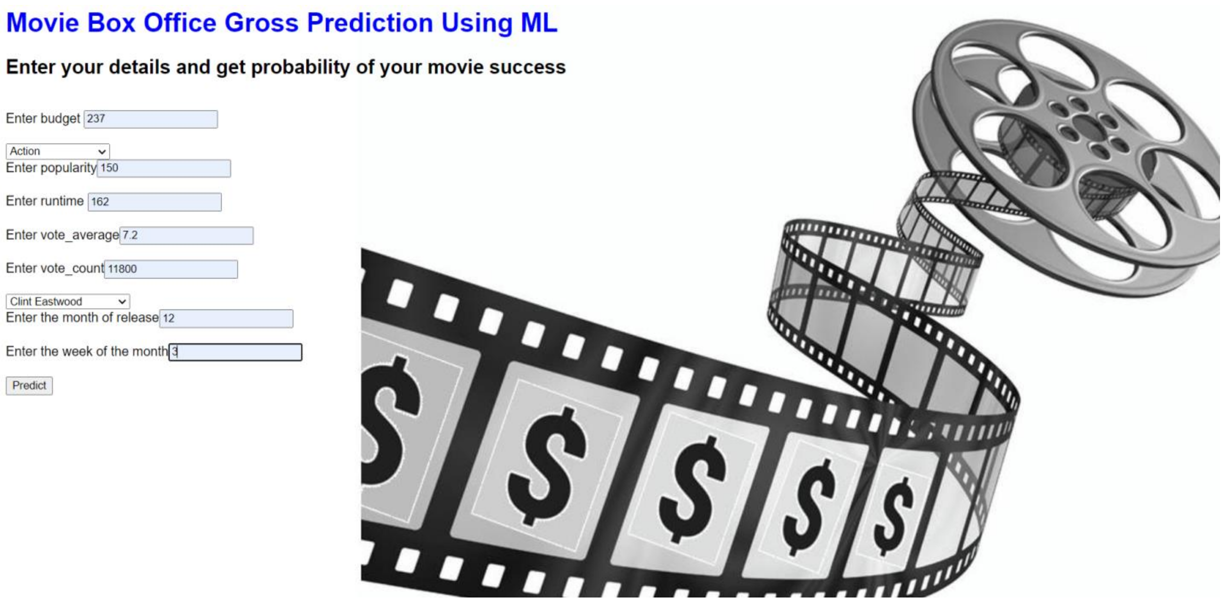


WHAT IS MOVIE BOX OFFICE?

It aims to predict movie box office gross using machine learning techniques. By leveraging historical data on various movie attributes such as genre, cast, budget, release date, and marketing strategies, we develop a predictive model to estimate a movie's potential box office performance. This predictive tool can assist movie production companies in making informed decisions regarding budget allocation, marketing strategies, and release timing to maximize revenue.

PREDICTIONS

- Showcasing the output on UI



Movie Box Office Gross Prediction Using ML

Enter your details and get probability of your movie success

Enter budget 237

Action

Enter popularity 150

Enter runtime 162

Enter vote_average 7.2

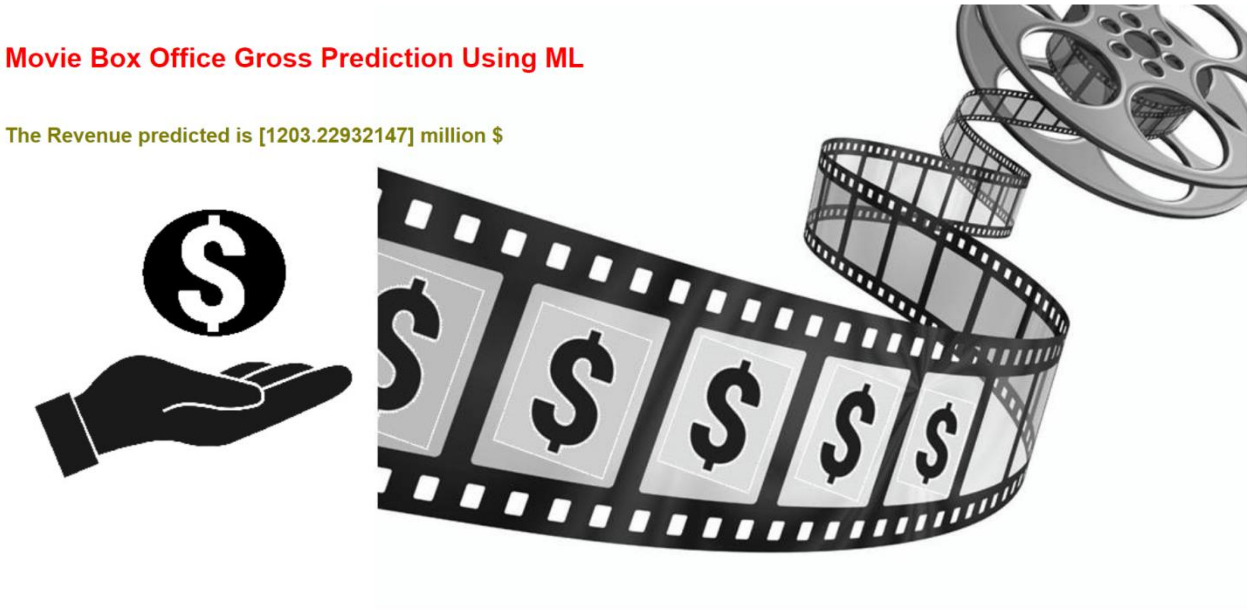
Enter vote_count 11800

Clint Eastwood

Enter the month of release 12

Enter the week of the month 3

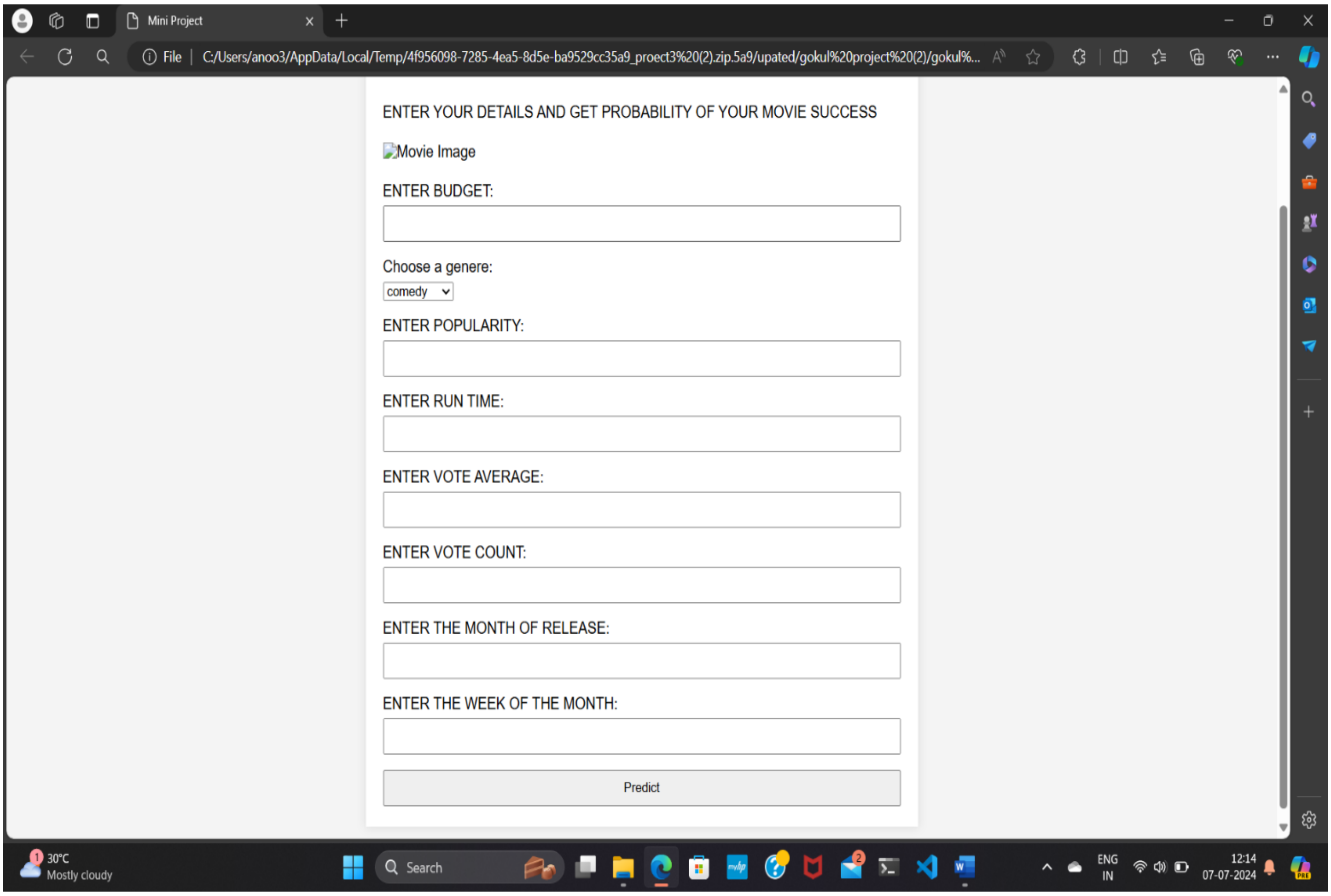
This is the prediction page where we get to choose the input from our local system and predict the output.



Movie Box Office Gross Prediction Using ML

The Revenue predicted is [1203.22932147] million $

Finally, the prediction for the given input features is shown.



Mini Project
File | C:/Users/anoo3/AppData/Local/Temp/4f956098-7285-4ea5-8d5e-ba9529cc35a9_proect3%20(2).zip.5a9/upated/gokul%20project%20(2)/gokul%...
Search
30°C
Mostly cloudy
ENG
IN
12:14
07-07-2024

ENTER YOUR DETAILS AND GET PROBABILITY OF YOUR MOVIE SUCCESS

Movie Image

ENTER BUDGET:

Choose a genere:
comedy

ENTER POPULARITY:

ENTER RUN TIME:

ENTER VOTE AVERAGE:

ENTER VOTE COUNT:

ENTER THE MONTH OF RELEASE:

ENTER THE WEEK OF THE MONTH:

7.ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

Predicting movie box office gross can provide several advantages for various stakeholders in the film industry:

1. Strategic Planning:

- Studios and distributors can use predictions to strategically plan release dates, marketing budgets, and distribution strategies based on expected revenue potential. This helps in optimizing resource allocation and maximizing profitability.

2. Risk Management:

- Predictions help mitigate financial risks associated with movie production and distribution by providing insights into potential revenue streams. This allows studios to make informed decisions about investment and budgeting.

3. Marketing Optimization:

- Understanding the factors influencing box office performance helps in tailoring marketing campaigns more effectively. Studios can target specific audience segments and optimize marketing spend to increase awareness and drive ticket sales.

4. Performance Evaluation:

- Predictions allow for benchmarking and evaluating the success of movies relative to expectations. This helps in assessing the performance of different genres, actors, directors, and production teams, leading to better decision-making in future projects.

5. Competitive Advantage:

- Studios that can accurately predict box office performance gain a competitive edge by releasing movies at optimal times and maximizing revenue potential. This can lead to increased market share and brand reputation within the industry.

6. Investor Confidence:

- Predicting box office gross provides stakeholders, including investors and financiers, with a clearer picture of potential returns on investment. This increases confidence and attracts investment in movie projects.

7. Audience Insights:

- Analyzing historical data and predicting box office gross can reveal trends and preferences among moviegoers. This insight helps studios in developing content that resonates with audiences, leading to higher ticket sales and viewer satisfaction.

8. Forecasting Revenue Streams:

- Predictions enable accurate forecasting of revenue streams from theatrical releases, which is crucial for financial planning and forecasting overall profitability of a movie.

Overall, movie box office gross prediction serves as a valuable tool for decision-makers in the film industry, facilitating better planning, risk management, marketing effectiveness, and overall business strategy.

DISADVANTAGES:

While predicting movie box office gross offers several advantages, there are also some potential disadvantages and challenges associated with this practice:

1. Uncertainty and Variability:

- Movie box office performance can be highly unpredictable due to various factors such as audience preferences, competition from other releases, and external events (e.g., economic downturns, unexpected world events). Predictions may not always accurately reflect these uncertainties.

2. Limited Data Availability and Quality:

- Historical box office data and other relevant information may not always be comprehensive, accurate, or readily available. This can impact the reliability and accuracy of predictions, especially for new genres, emerging markets, or unique film concepts.

3. Complexity of Factors:

- Predicting box office gross involves considering a wide range of factors (e.g., cast popularity, director reputation, marketing efforts, release timing, critical reception). Managing and incorporating these complex interactions into predictive models can be challenging and prone to errors.

4. Over-Reliance on Historical Patterns:

- Predictive models often rely on historical data patterns to forecast future box office performance. However, the film industry is dynamic, and past trends may not always predict future outcomes accurately, especially with changing audience preferences and consumption habits.

5. Influence of Non-Quantifiable Factors:

- Some critical factors influencing box office success (e.g., word-of-mouth, critical reviews, social media buzz) are difficult to quantify and incorporate into predictive models. These qualitative aspects can significantly impact a movie's performance but may not be captured effectively in data-driven predictions.

6. Impact of External Factors:

- External factors such as unexpected events (e.g., natural disasters, political unrest) or global trends (e.g., shifts in consumer behavior, technological advancements) can disrupt box office predictions. These factors are often beyond the control of studios and prediction models.

7. Ethical and Cultural Considerations:

- Predictive models may prioritize commercial success over artistic merit or cultural significance, potentially influencing film content and diversity in the industry. This can limit creativity and innovation in filmmaking if decisions are solely driven by financial forecasts.

8. Risk of Bias and Over-Optimization:

- Predictive models can be prone to biases based on historical data trends or assumptions about audience behavior. Over-optimization of models to fit past data patterns may overlook emerging trends or unconventional successes in the market.

9. Complexity in Model Interpretation:

- Understanding and interpreting predictive models requires expertise in data science and domain knowledge of the film industry. Misinterpretation or misuse of predictions can lead to suboptimal decision-making and resource allocation.

In summary, while movie box office gross prediction offers valuable insights and benefits for stakeholders, it also presents challenges related to data availability, accuracy, model complexity, and the dynamic nature of the film industry. Awareness of these disadvantages helps in adopting a balanced approach to using predictive analytics in film distribution and marketing strategies.

8.APPLICATIONS

Movie box office gross prediction finds several practical applications across various stakeholders in the film industry and related sectors:

1. **Studios and Distributors:**
   - **Release Strategy:** Predicting box office gross helps studios decide on the optimal release dates and distribution strategies. This includes selecting the right timing to maximize ticket sales and minimize competition.
   - **Budget Allocation:** Studios can allocate marketing budgets more effectively by predicting potential revenue streams. This ensures resources are utilized efficiently to promote movies with higher expected returns.
   - **Risk Management:** Predictions assist in mitigating financial risks associated with movie production and distribution, guiding studios in making informed investment decisions.
2. **Marketing and Promotion:**
   - **Targeted Marketing:** Predictions enable targeted marketing campaigns based on audience demographics and preferences. This includes tailoring promotional efforts to reach specific audience segments likely to contribute to box office success.
   - **Campaign Optimization:** Insights from box office predictions help optimize marketing strategies across various channels, including digital advertising, social media, and traditional media platforms.
3. **Investors and Financiers:**

  - **Financial Planning:** Predictions provide investors with insights into potential returns on investment (ROI) from financing movie projects. This enhances decision-making by evaluating the financial viability and profitability of film investments.

4. **Film Critics and Analysts:**
   - **Performance Evaluation:** Box office predictions serve as benchmarks for evaluating the success and performance of movies. Analysts and critics can use predicted and actual box office results to assess industry trends, audience preferences, and film quality.
5. **Streaming Platforms:**
   - **Content Acquisition:** Streaming platforms use box office predictions to assess the commercial viability of acquiring rights to movies for digital distribution. This includes licensing decisions based on expected viewer demand and profitability.
6. **Audience Insights:**
   - **Consumer Behavior:** Predictions offer insights into audience behavior and preferences, guiding content creation and acquisition decisions. This includes identifying trends in genre popularity, actor appeal, and viewing habits across different demographics.
7. **Academic and Research Purposes:**
   - **Industry Studies:** Researchers and academics analyze box office predictions to study market dynamics, economic trends, and consumer behavior in the film industry. This contributes to the development of theories and models in entertainment economics and business strategy.

Overall, movie box office gross prediction plays a crucial role in enhancing decision-making, optimizing resource allocation, and understanding audience dynamics in the dynamic and competitive film industry.

9.CONCLUSION

Concluding a movie box office gross prediction typically involves summarizing the results of the analysis, reflecting on the accuracy of the model, and discussing any factors that might influence the final outcomes. Here's a sample conclusion:

Summary of Prediction Results

Based on our predictive model, [Movie Title] is estimated to gross approximately $[Predicted Gross] at the box office. This prediction is based on various factors, including historical box office data, genre trends, star power, marketing efforts, and initial audience reception.

Model Accuracy and Limitations

Our model has shown a reasonable degree of accuracy when validated against past movie releases, with a margin of error of approximately [X%]. However, it is important to note that predicting box office performance is inherently uncertain due to the dynamic nature of the film industry and external influences that can significantly impact results.

Influencing Factors

Several factors could affect the actual box office gross of [Movie Title]:

1. **Competition**: The release schedule and performance of competing films can divert potential audiences.
2. **Marketing and Promotions**: The effectiveness and reach of marketing campaigns can significantly influence audience turnout.
3. **Critical Reception**: Early reviews and word-of-mouth can sway audience decisions.
4. **Current Events**: Socioeconomic factors, global events, or unexpected occurrences can impact audience attendance.

10.FUTURE SCOPE

The future scope of movie box office gross prediction involves exploring advanced methodologies, integrating new data sources, and adapting to the evolving entertainment landscape. Here are some key areas to consider:

1. Advanced Predictive Models

- **Machine Learning and AI**: Implementing more sophisticated machine learning algorithms and artificial intelligence techniques can improve the accuracy of predictions. Deep learning models, for instance, can capture complex patterns and relationships in the data.
- **Hybrid Models**: Combining different modeling approaches (e.g., statistical models with machine learning techniques) can enhance prediction robustness.

2. Expanded Data Sources

- **Social Media and Online Sentiment**: Analyzing social media platforms, online reviews, and forums can provide real-time insights into audience sentiment and anticipation, helping to refine predictions.
- **Search Trends**: Monitoring search engine trends and keywords related to the movie can offer additional indicators of public interest and potential box office performance.

3. Real-time and Dynamic Predictions

- **Continuous Monitoring**: Developing systems that continuously update predictions based on new data, such as early box office returns, marketing campaign effectiveness, and audience feedback.
- **Adaptive Models**: Creating models that adapt to changing trends and factors in real-time, providing more accurate and timely forecasts.

4. Global Box Office Predictions

- **Regional Analysis**: Expanding predictions to include international markets, taking into account regional preferences, local marketing efforts, and competition.
- **Cultural Factors**: Integrating cultural and socio-economic factors unique to different regions can improve the accuracy of global predictions.

5. Cross-Platform Performance

- **Streaming and Digital Platforms**: As the industry shifts towards digital consumption, predicting revenue from streaming services, video-on-demand, and digital downloads becomes increasingly important.
- **Hybrid Release Models**: Analyzing the impact of hybrid release models (simultaneous theater and digital releases) on overall revenue.

6. Audience Demographics and Preferences

- **Personalized Predictions**: Leveraging demographic data and viewing preferences to tailor predictions for specific audience segments.
- **Behavioral Analytics**: Using behavioral analytics to understand how different audience segments respond to various types of content and marketing strategies.

7. Collaborations and Data Sharing

- **Industry Partnerships**: Collaborating with studios, distributors, and marketing firms to access proprietary data and insights.
- **Open Data Initiatives**: Encouraging data sharing and collaboration within the industry to enhance predictive models.

8. Ethical and Privacy Considerations

- **Data Privacy**: Ensuring that predictive models comply with data privacy regulations and ethical standards, especially when using personal data from online platforms.
- **Transparency**: Maintaining transparency in how predictions are made and the factors considered, fostering trust among stakeholders.

Conclusion

The future of movie box office gross prediction lies in embracing advanced technologies, leveraging diverse data sources, and adapting to the ever-changing entertainment landscape. By continuously improving predictive models and incorporating real-time data, the industry can achieve more accurate and insightful forecasts, ultimately benefiting filmmakers, distributors, and audiences alike.

11.BIBILOGRAPHY

A comprehensive bibliography for movie box office gross prediction might include academic articles, books, and industry reports that cover various aspects of predictive modeling, data analytics, and the film industry. Here are some key references:

Books

1. **Eliashberg, J., Elberse, A., & Leenders, M. A. A. M. (2006).** "The Motion Picture Industry: Critical Issues in Practice, Current Research, and New Research Directions." *Marketing Science*, 25(6), 638-661.
2. **Moul, C. C. (Ed.). (2005).** *A Concise Handbook of Movie Industry Economics*. Cambridge University Press.
3. **Neelamegham, R., & Chintagunta, P. K. (1999).** "A Bayesian Model to Forecast New Product Performance in Domestic and International Markets." *Marketing Science*, 18(2), 115-136.
4. **Vogel, H. L. (2011).** *Entertainment Industry Economics: A Guide for Financial Analysis*. Cambridge University Press.

Academic Articles

1. **Einav, L. (2007).** "Seasonality in the U.S. Motion Picture Industry." *RAND Journal of Economics*, 38(1), 127-145.
2. **Hennig-Thurau, T., Houston, M. B., & Heitjans, T. (2009).** "Conceptualizing and Measuring the Monetary Value of Brand Extensions: The Case of Motion Pictures." *Journal of Marketing*, 73(6), 167-183.
3. **Sawhney, M. S., & Eliashberg, J. (1996).** "A Parsimonious Model for Forecasting Gross Box-Office Revenues of Motion Pictures." *Marketing Science*, 15(2), 113-131.
4. **Simonoff, J. S., & Sparrow, I. R. (2000).** "Predicting Movie Grosses: Winners and Losers, Blockbusters and Sleepers." *Chance*, 13(3), 15-24.

Industry Reports

1. **PwC. (2020).** *Global Entertainment & Media Outlook 2020-2024*. PricewaterhouseCoopers.
2. **MPAA. (2021).** *Theatrical Market Statistics*. Motion Picture Association of America.
3. **Comscore. (2021).** *State of the Box Office*. Comscore.

Conference Papers

1. **Rui, H., Liu, Y., & Whinston, A. B. (2010).** "Whose and What Chatter Matters? The Effect of Tweets on Movie Sales." *Proceedings of the 2010 International Conference on Information Systems*.
2. **Duan, W., Gu, B., & Whinston, A. B. (2008).** "Do Online Reviews Matter? An Empirical Investigation of Panel Data." *Journal of Business Research*, 61(8), 866-874.

Online Resources

1. **Box Office Mojo.** "Box Office Data and Analysis." Available at: boxofficemojo.com
2. **IMDb.** "Movie Information and Data." Available at: imdb.com
3. **The Numbers.** "Movie Financial Analysis." Available at: the-numbers.com

These references provide a solid foundation for understanding the various aspects of predicting movie box office gross, from theoretical models and empirical studies to industry reports and data sources.

12.APPENDIX

Model building :

1)Dataset

2)Google colab and VS code Application Building

   1. HTML file (Index file, Predict file)
   1. CSS file
   2. Models in pickle format

SOURCE CODE:

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <title>Mini Project</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         margin: 0;
         padding: 20px;
         background-color: #f4f4f4;
      }
      .container {
            max-width: 600px;
            margin: auto;
            background: white;
            padding: 20px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        }
        h1 {
            color: blue;
            text-align: center;
        }
        form div {
            margin-bottom: 15px;
        }
        form label {
            display: block;
            margin-bottom: 5px;
        }
        form input[type="text"], form input[type="submit"] {
            width: 100%;
            padding: 8px;
            box-sizing: border-box;
        }
        form input[type="radio"] {
            margin-right: 10px;
        }
        img {
            display: block;
            max-width: 100%;
            margin: 20px auto;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>MOVIE BOX OFFICE GROSS PREDICTION USING ML</h1>
        <p>ENTER YOUR DETAILS AND GET PROBABILITY OF YOUR MOVIE SUCCESS</p>
        <img src="static/1.jpg" alt="Movie Image">

        <form action="{{ url_for('predict') }}" method="post">
            <div>
                <label for="budget">ENTER BUDGET:</label>
                <input type="text" id="budget" name="budget">
            </div>
            <div>
                <form action="/action_page.php">
                    <label>genres</label><br />
                    <input type="text" name="genres" id="genres" list="genres" /><br />
                    <datalist id="companies">
                        <option data-value="1">Comedy</option>
                        <option data-value="2">Action</option>
                        <option data-value="3">Adventure</option>
                        <option data-value="4">Documentary</option>
                    </datalist>
            </div>

            <div>
                <label for="popularity">ENTER POPULARITY:</label>
                <input type="text" id="popularity" name="popularity">
            </div>
            <div>
                <label for="runtime">ENTER RUN TIME:</label>
                <input type="text" id="runtime" name="runtime">
            </div>
            <div>
                <label for="vote_average">ENTER VOTE AVERAGE:</label>
                <input type="text" id="vote_average" name="vote_average">
            </div>
            <div>
                <label for="vote_count">ENTER VOTE COUNT:</label>
                <input type="text" id="vote_count" name="vote_count">
            </div>
            <div>
                <label for="release_month">ENTER THE MONTH OF RELEASE:</label>
                <input type="text" id="release_month" name="release_month">
            </div>
            <div>
                <label for="release_week">ENTER THE WEEK OF THE MONTH:</label>
                <input type="text" id="release_week" name="release_DOW">
            </div>
            <input type="submit" value="Predict">
        </form>
    </div>
</body>
</html>
```

PREDICT.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Prediction Result</title>
</head>
<body>
    <h1>Prediction Result</h1>
    <img src="static/1.jpg" alt="Movie Image" style="align-items: center;">
    <p>{{ prediction_text }} million $</p>

</body>
```

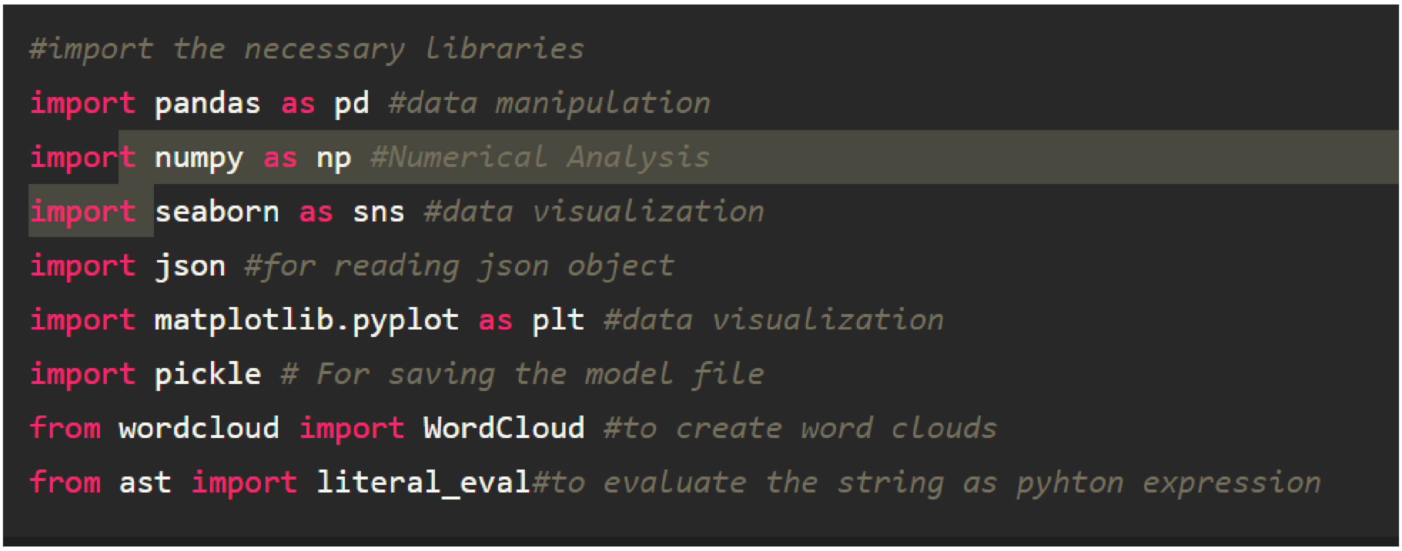
```
</html>
```

APP.PY

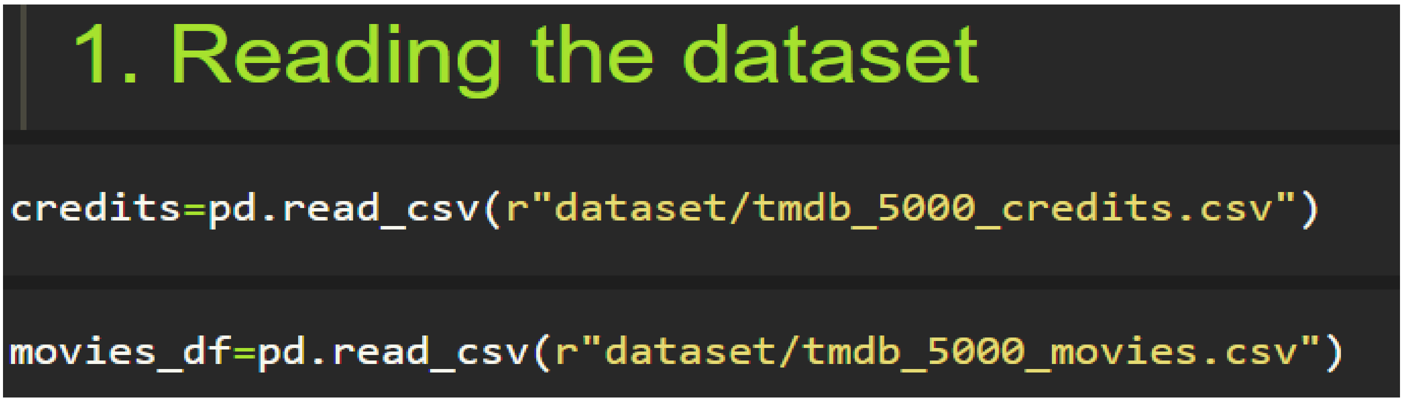
```
from flask import Flask,render_template,request
import joblib
import numpy as np
import pandas as pd
import pickle
app=Flask(_name_)
#model=joblib.load('random_forest_model.pkl')
model=pickle.load(open('model.pkl','rb'))
scalar=pickle.load(open('scalar.pkl','rb'))
label=pickle.load(open('label.pkl','rb'))
app=Flask(_name_,template_folder='template')
@app.route('/')
def home():
    return render_template('Untitled-1.html')
@app.route('/predict', methods=['POST'])
def predict():
   input_feature=[x for x in request.form.values()]
   input_feature=np.transpose(input_feature)
   input_feature=[np.array(input_feature)]
   print(input_feature)
   names=['budget','genres','popularity','runtime','vote_average','vote_count','release_month','release_DOW']
   data=pd.DataFrame(input_feature,columns=names)
   x=scalar.transform(data)
   prediction=model.predict(data)
   result=int(prediction[0])
   print(result)
   return render_template('result.html', prediction_text='The Revenue Predicted is: {}'.format(result))
if _name=='main_':
  app.run(debug=True)
```

CODE SNIPPETS

MODEL BUILDING



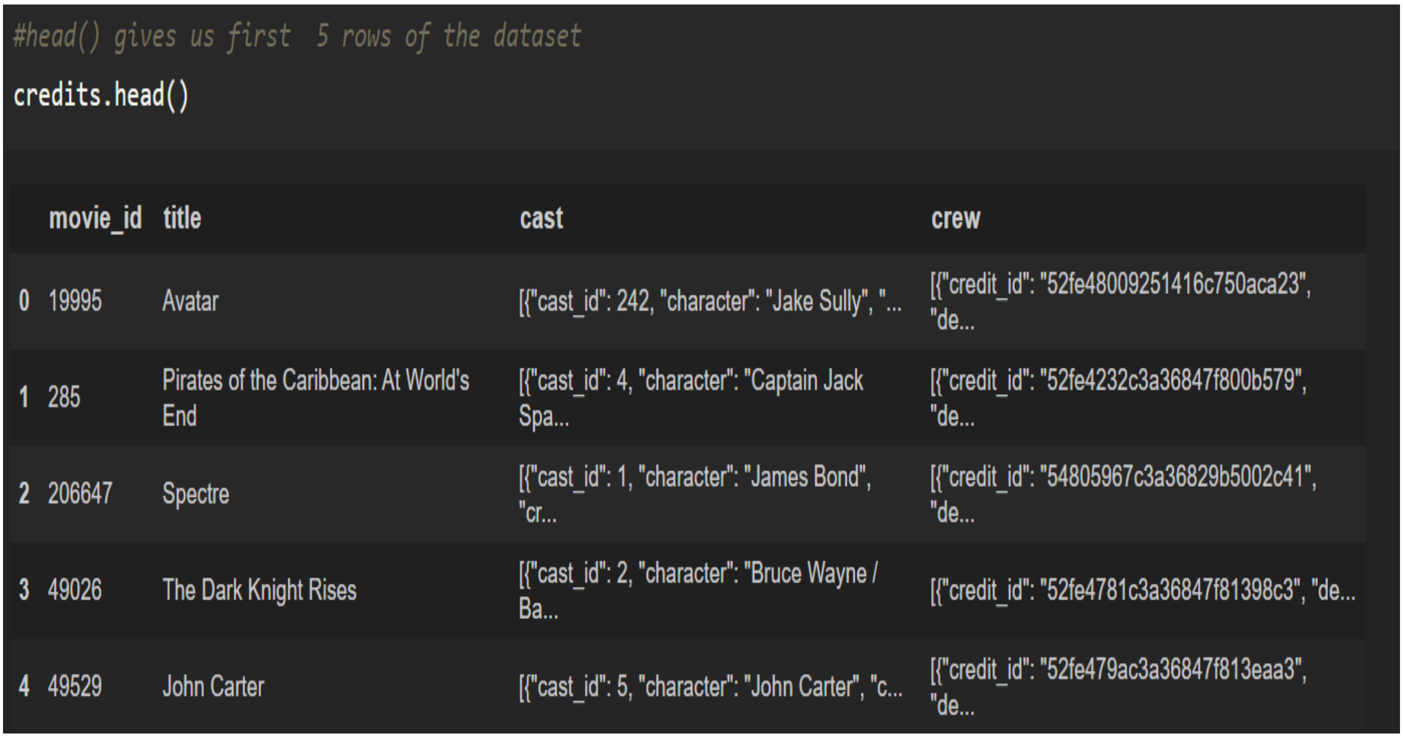
```
#import the necessary libraries
import pandas as pd #data manipulation
import numpy as np #Numerical Analysis
import seaborn as sns #data visualization
import json #for reading json object
import matplotlib.pyplot as plt #data visualization
import pickle # For saving the model file
from wordcloud import WordCloud #to create word clouds
from ast import literal_eval#to evaluate the string as pyhton expression
```



1. Reading the dataset

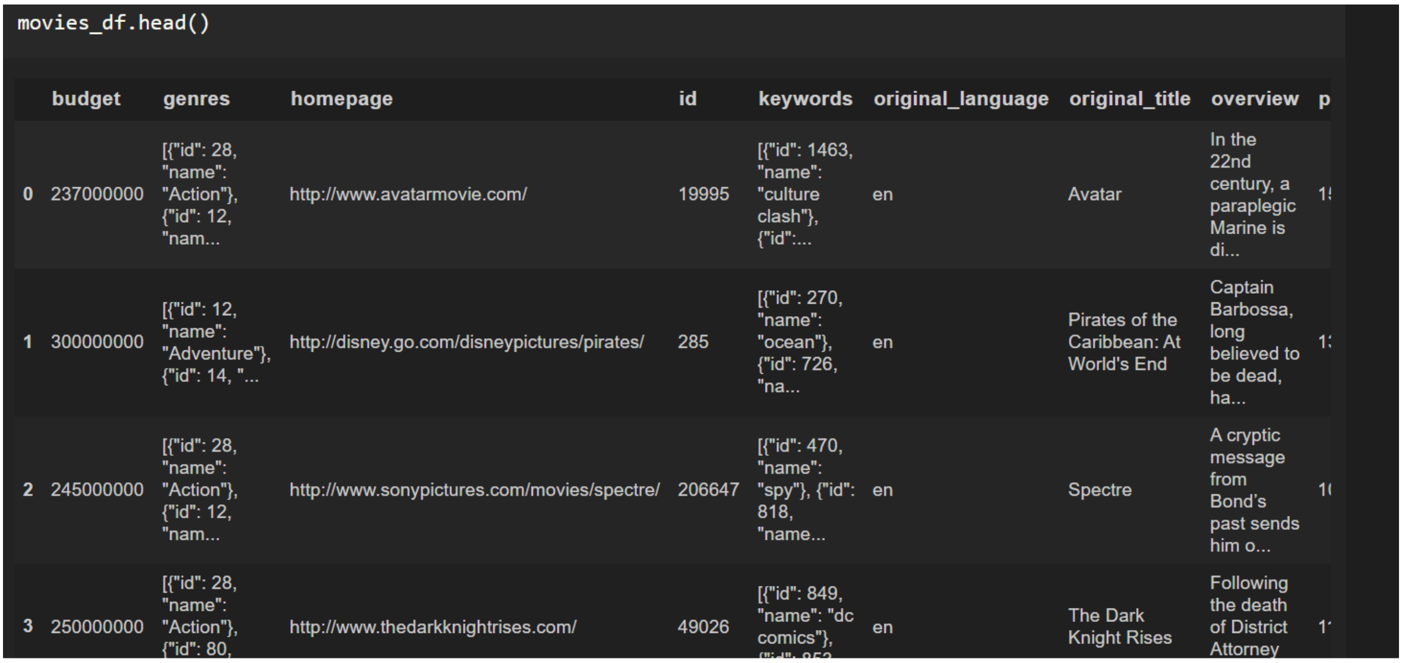
```
credits=pd.read_csv(r"dataset/tmdb_5000_credits.csv")
```

```
movies_df=pd.read_csv(r"dataset/tmdb_5000_movies.csv")
```



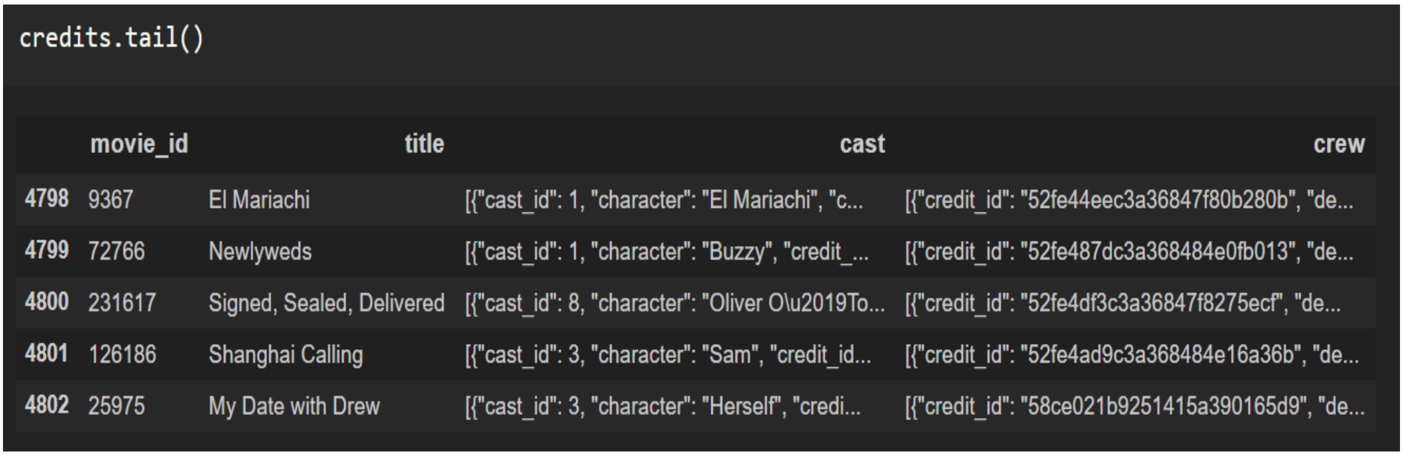
```
#head() gives us first  5 rows of the dataset
credits.head()
```

	movie_id	title	cast	crew
0	19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...
1	285	Pirates of the Caribbean: At World's End	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de...
2	206647	Spectre	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de...
3	49026	The Dark Knight Rises	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de...
4	49529	John Carter	[{"cast_id": 5, "character": "John Carter", "c...	[{"credit_id": "52fe479ac3a36847f813eaa3", "de...



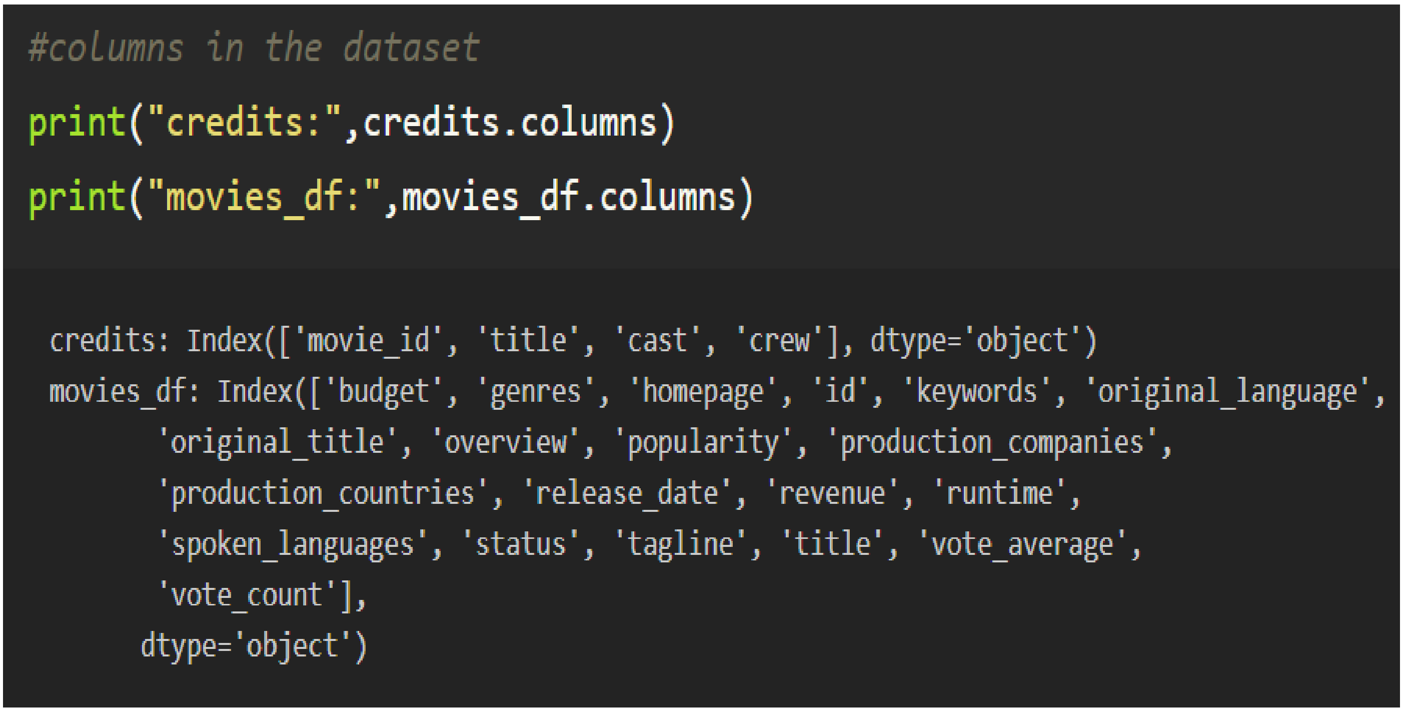
```python
movies_df.head()
```

	budget	genres	homepage	id	keywords	original_language	original_title	overview	p
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	1
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	1
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cryptic message from Bond's past sends him o...	1
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80,	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"},	en	The Dark Knight Rises	Following the death of District Attorney	1



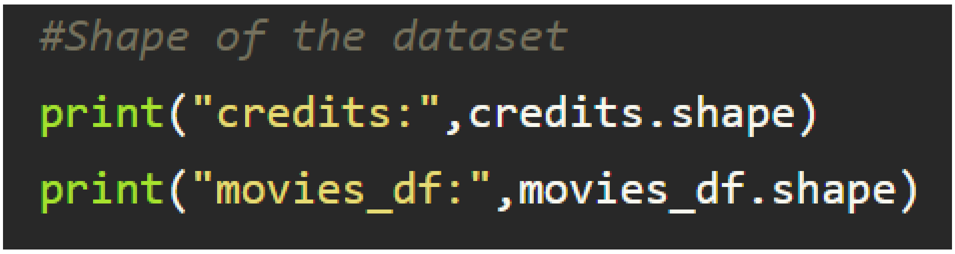
```python
credits.tail()
```

	movie_id	title	cast	crew
4798	9367	El Mariachi	[{"cast_id": 1, "character": "El Mariachi", "c...	[{"credit_id": "52fe44eec3a36847f80b280b", "de...
4799	72766	Newlyweds	[{"cast_id": 1, "character": "Buzzy", "credit_...	[{"credit_id": "52fe487dc3a368484e0fb013", "de...
4800	231617	Signed, Sealed, Delivered	[{"cast_id": 8, "character": "Oliver O\u2019To...	[{"credit_id": "52fe4df3c3a36847f8275ecf", "de...
4801	126186	Shanghai Calling	[{"cast_id": 3, "character": "Sam", "credit_id...	[{"credit_id": "52fe4ad9c3a368484e16a36b", "de...
4802	25975	My Date with Drew	[{"cast_id": 3, "character": "Herself", "credi...	[{"credit_id": "58ce021b9251415a390165d9", "de...

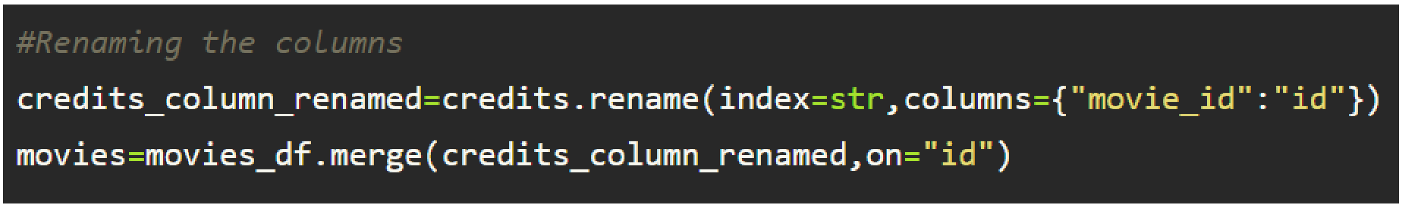


```python
#columns in the dataset
print("credits:",credits.columns)
print("movies_df:",movies_df.columns)
```

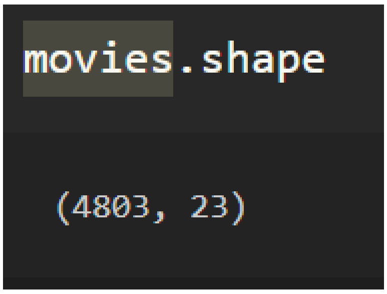
```
credits: Index(['movie_id', 'title', 'cast', 'crew'], dtype='object')
movies_df: Index(['budget', 'genres', 'homepage', 'id', 'keywords', 'original_language',
       'original_title', 'overview', 'popularity', 'production_companies',
       'production_countries', 'release_date', 'revenue', 'runtime',
       'spoken_languages', 'status', 'tagline', 'title', 'vote_average',
       'vote_count'],
      dtype='object')
```



```python
#Shape of the dataset
print("credits:",credits.shape)
print("movies_df:",movies_df.shape)
```

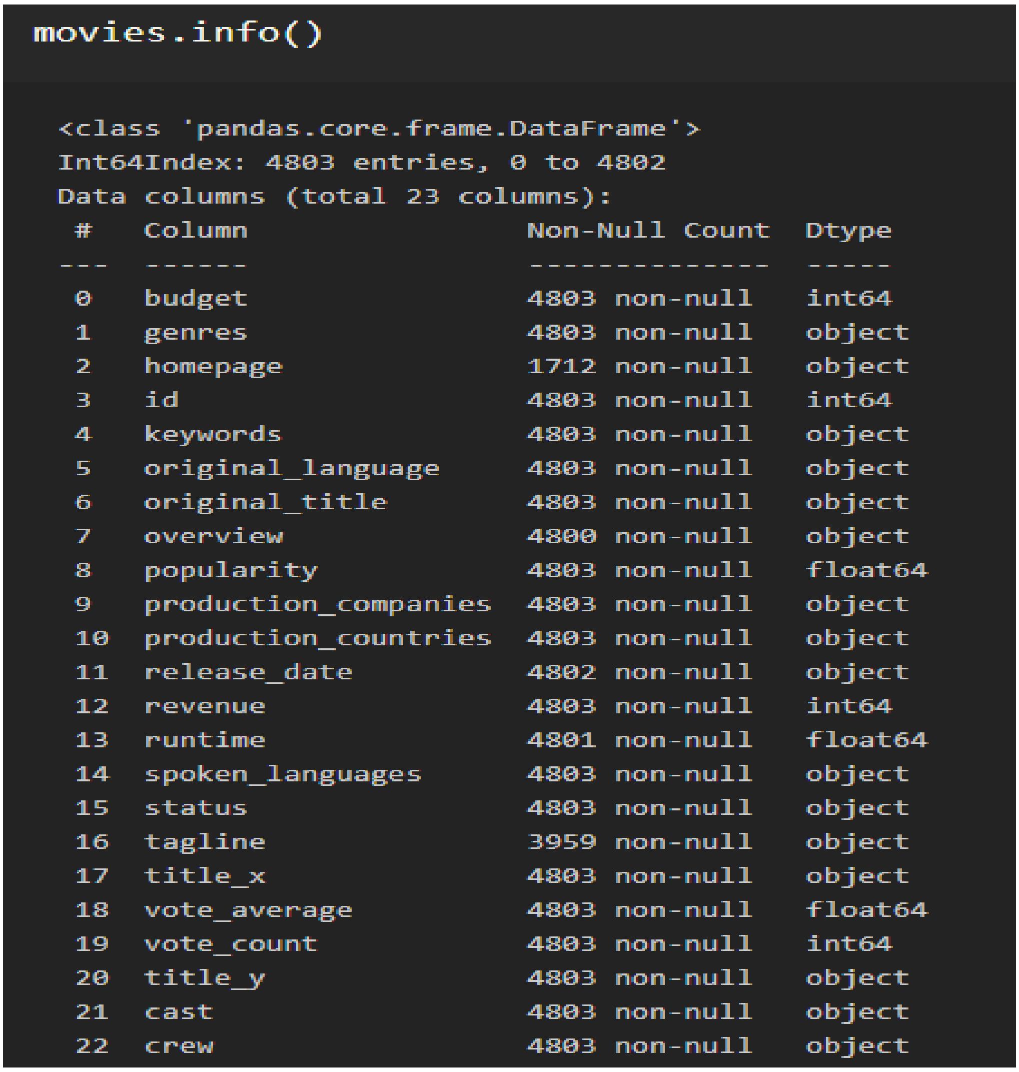


```python
#Renaming the columns
credits_column_renamed=credits.rename(index=str,columns={"movie_id":"id"})
movies=movies_df.merge(credits_column_renamed,on="id")
```



```python
movies.shape
```

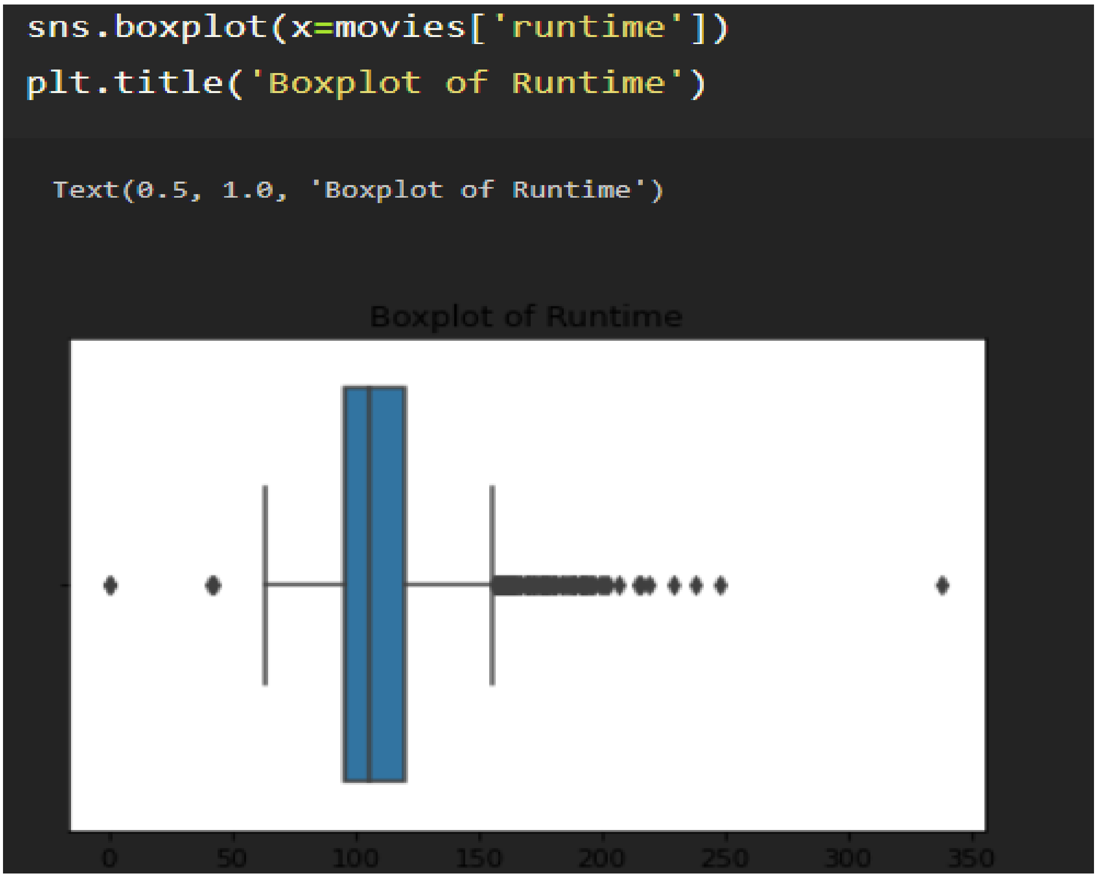
```
(4803, 23)
```



```python
movies.info()
```

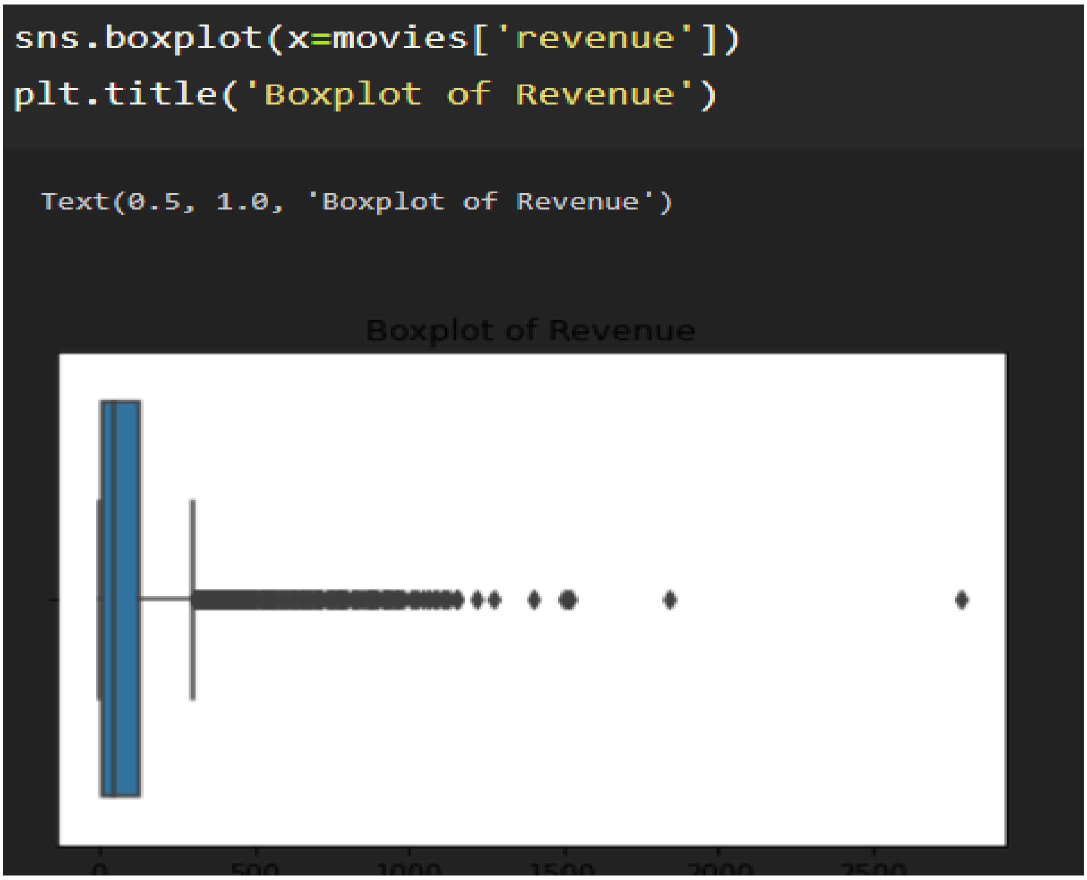
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4803 entries, 0 to 4802
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   budget                4803 non-null   int64  
 1   genres                4803 non-null   object 
 2   homepage              1712 non-null   object 
 3   id                    4803 non-null   int64  
 4   keywords              4803 non-null   object 
 5   original_language     4803 non-null   object 
 6   original_title        4803 non-null   object 
 7   overview              4800 non-null   object 
 8   popularity            4803 non-null   float64
 9   production_companies  4803 non-null   object 
 10  production_countries  4803 non-null   object 
 11  release_date          4802 non-null   object 
 12  revenue               4803 non-null   int64  
 13  runtime               4801 non-null   float64
 14  spoken_languages      4803 non-null   object 
 15  status                4803 non-null   object 
 16  tagline               3959 non-null   object 
 17  title_x               4803 non-null   object 
 18  vote_average          4803 non-null   float64
 19  vote_count            4803 non-null   int64  
 20  title_y               4803 non-null   object 
 21  cast                  4803 non-null   object 
 22  crew                  4803 non-null   object 
```

BOX PLOT



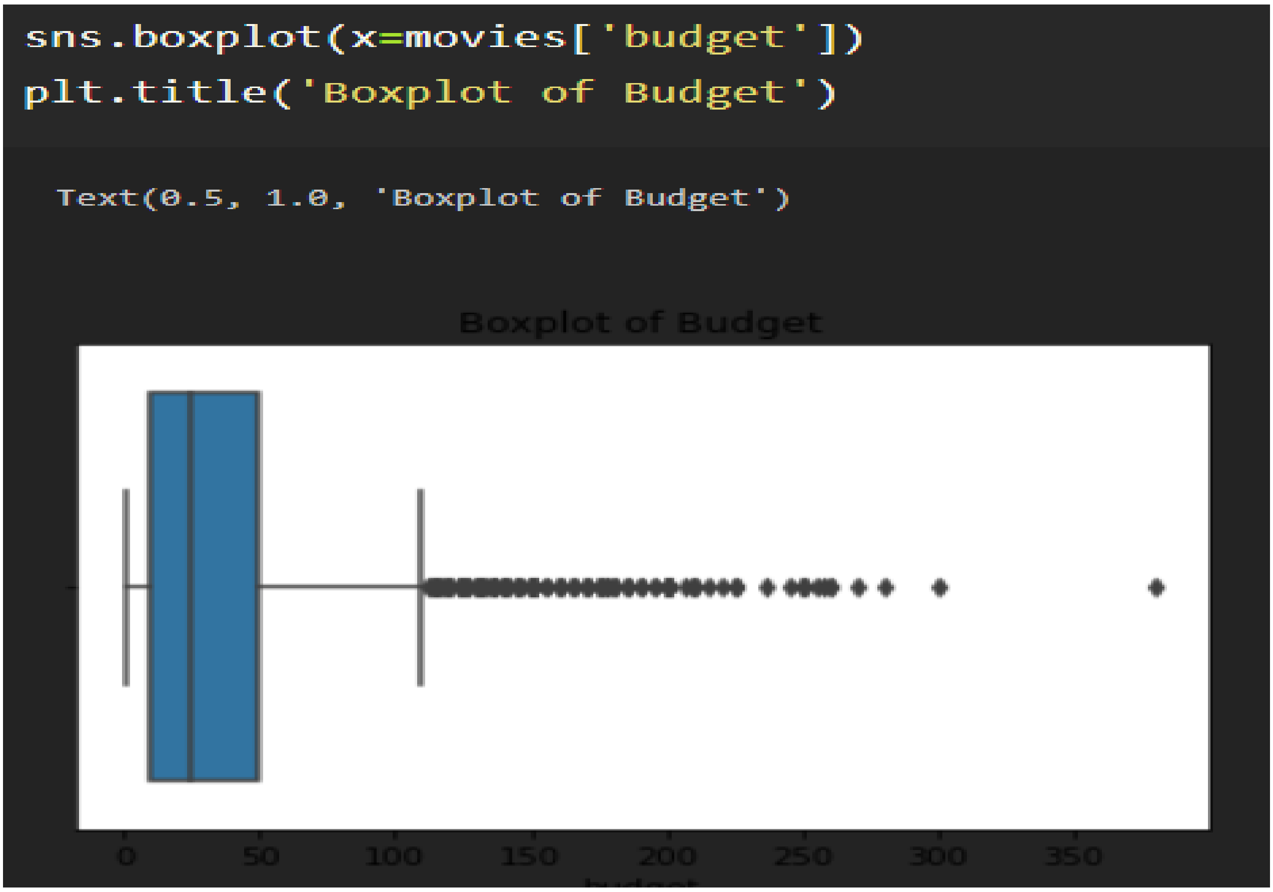
```
sns.boxplot(x=movies['runtime'])
plt.title('Boxplot of Runtime')
```

```
Text(0.5, 1.0, 'Boxplot of Runtime')
```



```
sns.boxplot(x=movies['revenue'])
plt.title('Boxplot of Revenue')
```

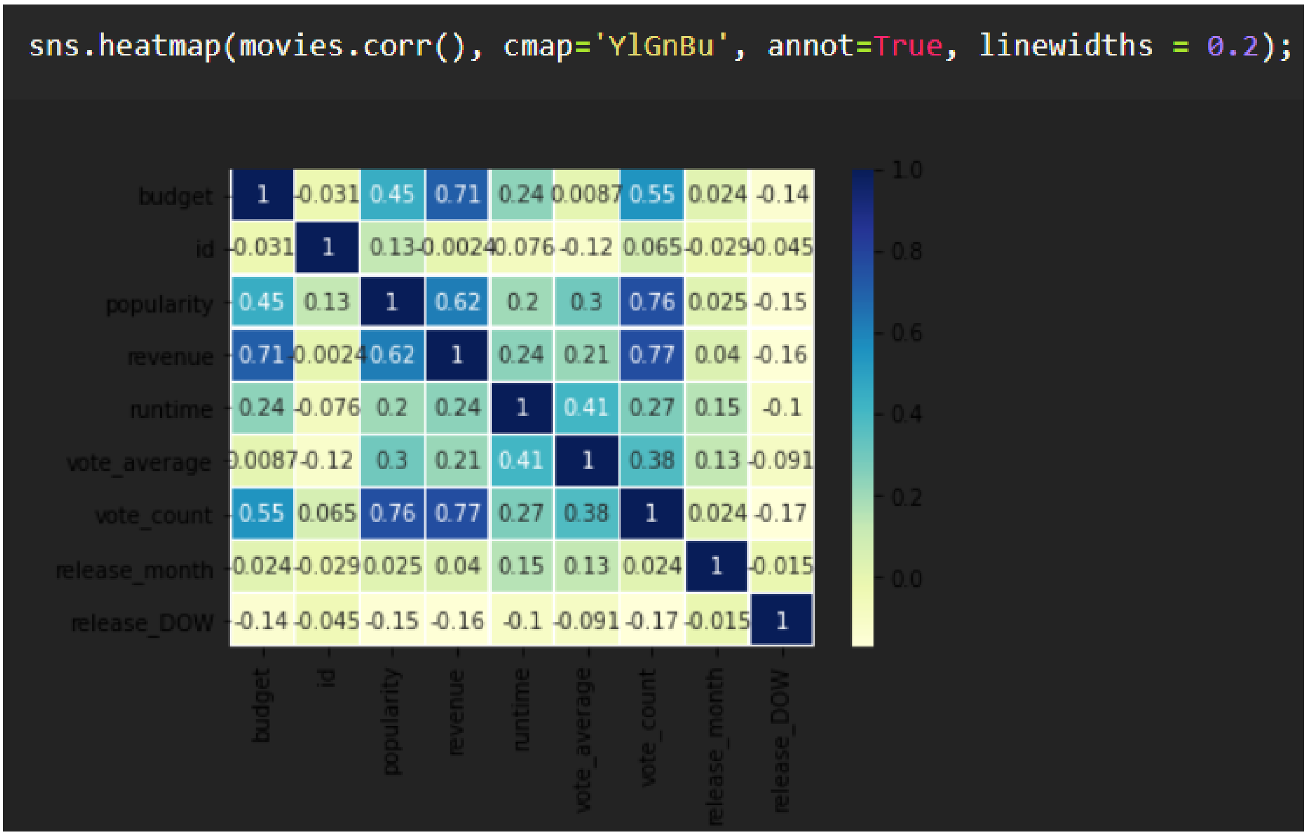
```
Text(0.5, 1.0, 'Boxplot of Revenue')
```



```
sns.boxplot(x=movies['budget'])
plt.title('Boxplot of Budget')
```

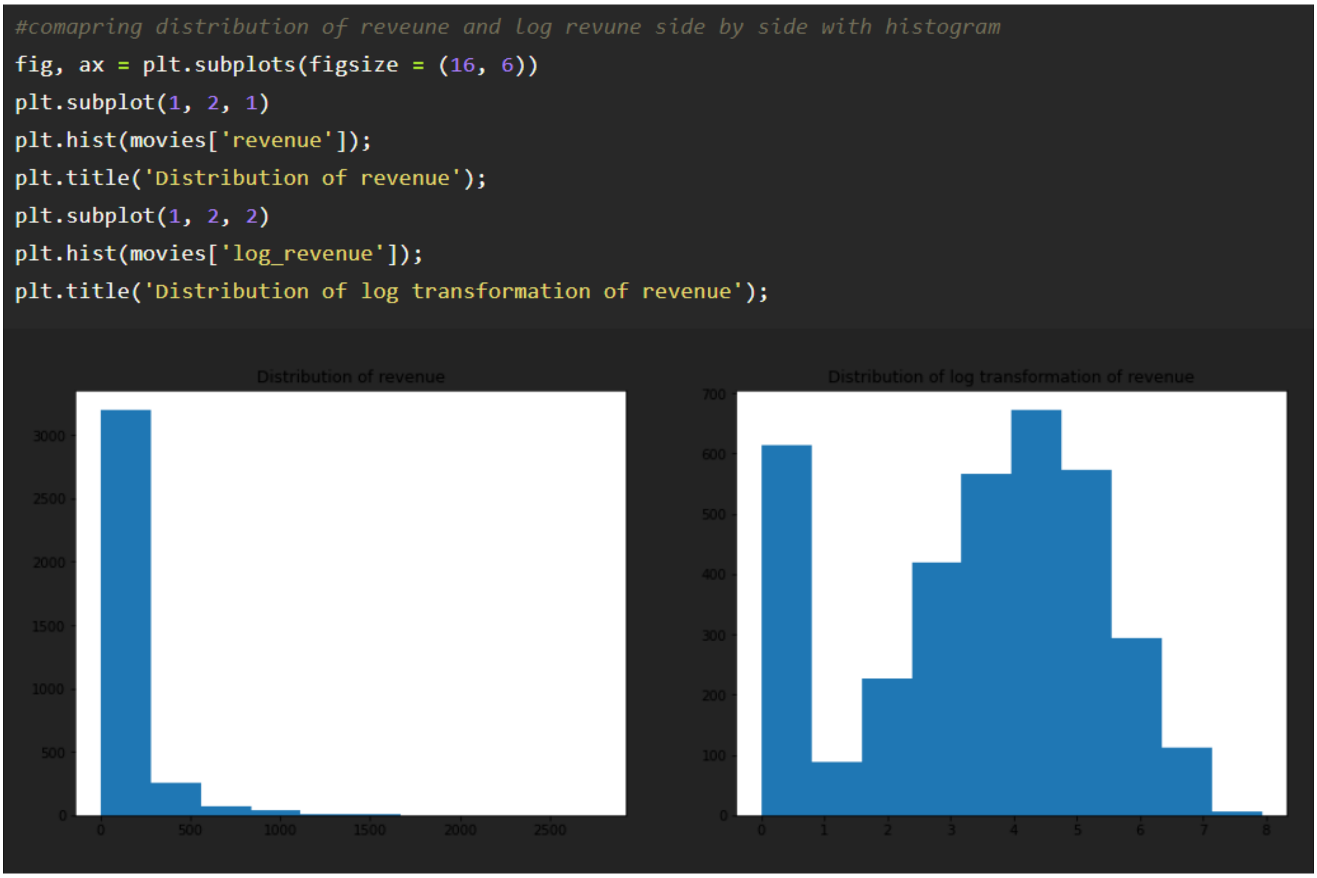
```
Text(0.5, 1.0, 'Boxplot of Budget')
```

HEAT MAP

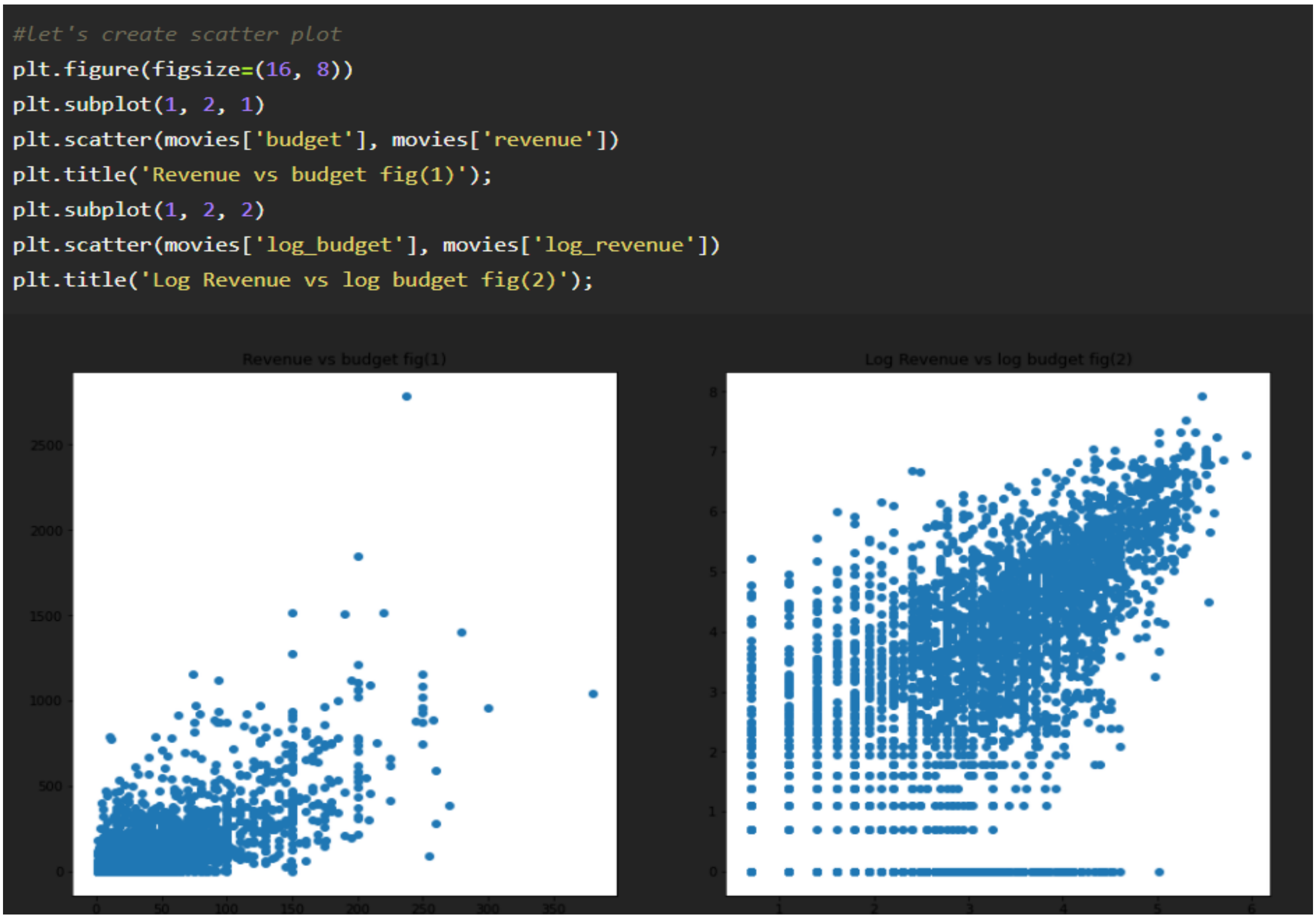


```
sns.heatmap(movies.corr(), cmap='YlGnBu', annot=True, linewidths = 0.2);
```

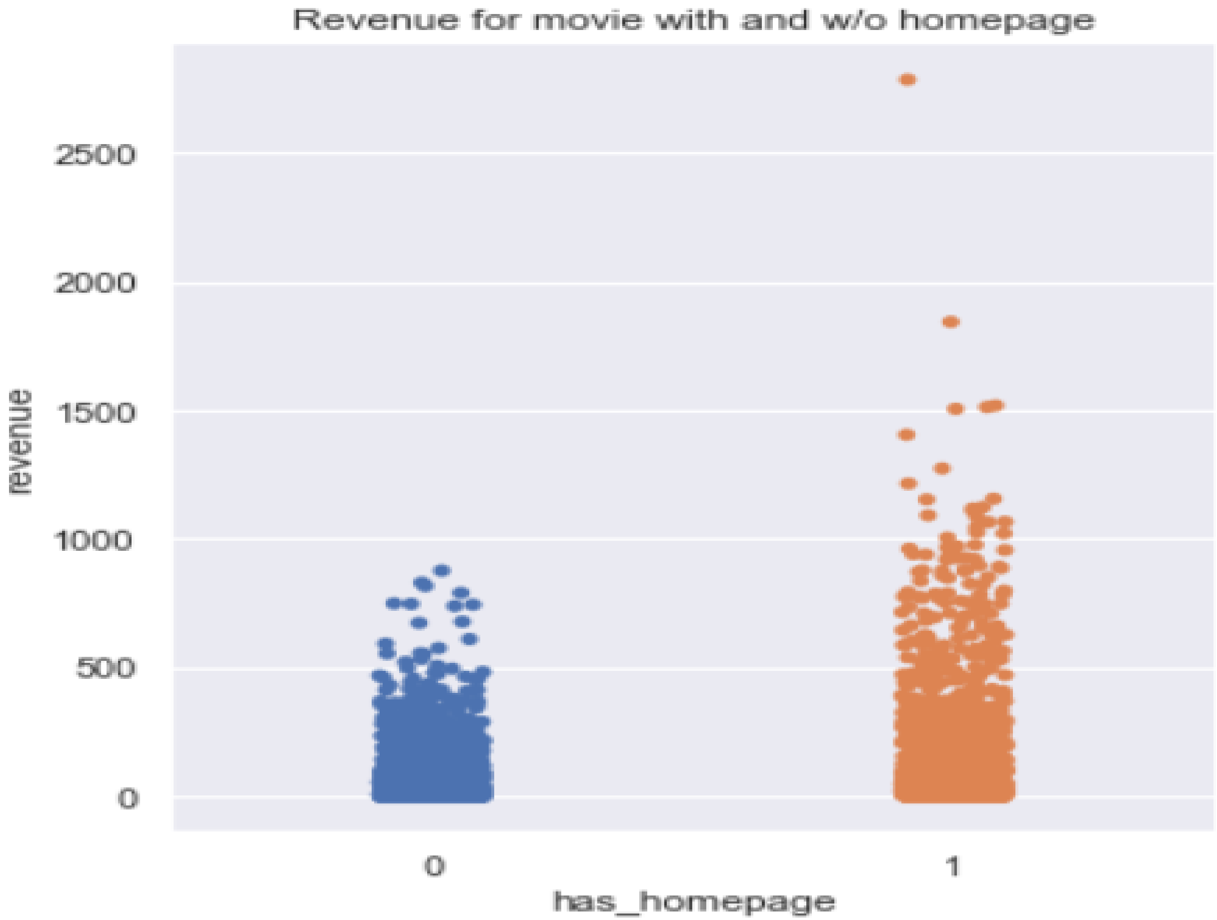
budget 1 -0.031 0.45 0.71 0.24 0.0087 0.55 0.024 -0.14
id -0.031 1 0.13 -0.0024 -0.076 -0.12 0.065 -0.029 -0.045
popularity 0.45 0.13 1 0.62 0.2 0.3 0.76 0.025 -0.15
revenue 0.71 -0.0024 0.62 1 0.24 0.21 0.77 0.04 -0.16
runtime 0.24 -0.076 0.2 0.24 1 0.41 0.27 0.15 -0.1
vote_average 0.0087 -0.12 0.3 0.21 0.41 1 0.38 0.13 -0.091
vote_count 0.55 0.065 0.76 0.77 0.27 0.38 1 0.024 -0.17
release_month 0.024 -0.029 0.025 0.04 0.15 0.13 0.024 1 -0.015
release_DOW -0.14 -0.045 -0.15 -0.16 -0.1 -0.091 -0.17 -0.015 1
budget
id
popularity
revenue
runtime
vote_average
vote_count
release_month
release_DOW
1.0
0.8
0.6
0.4
0.2
0.0

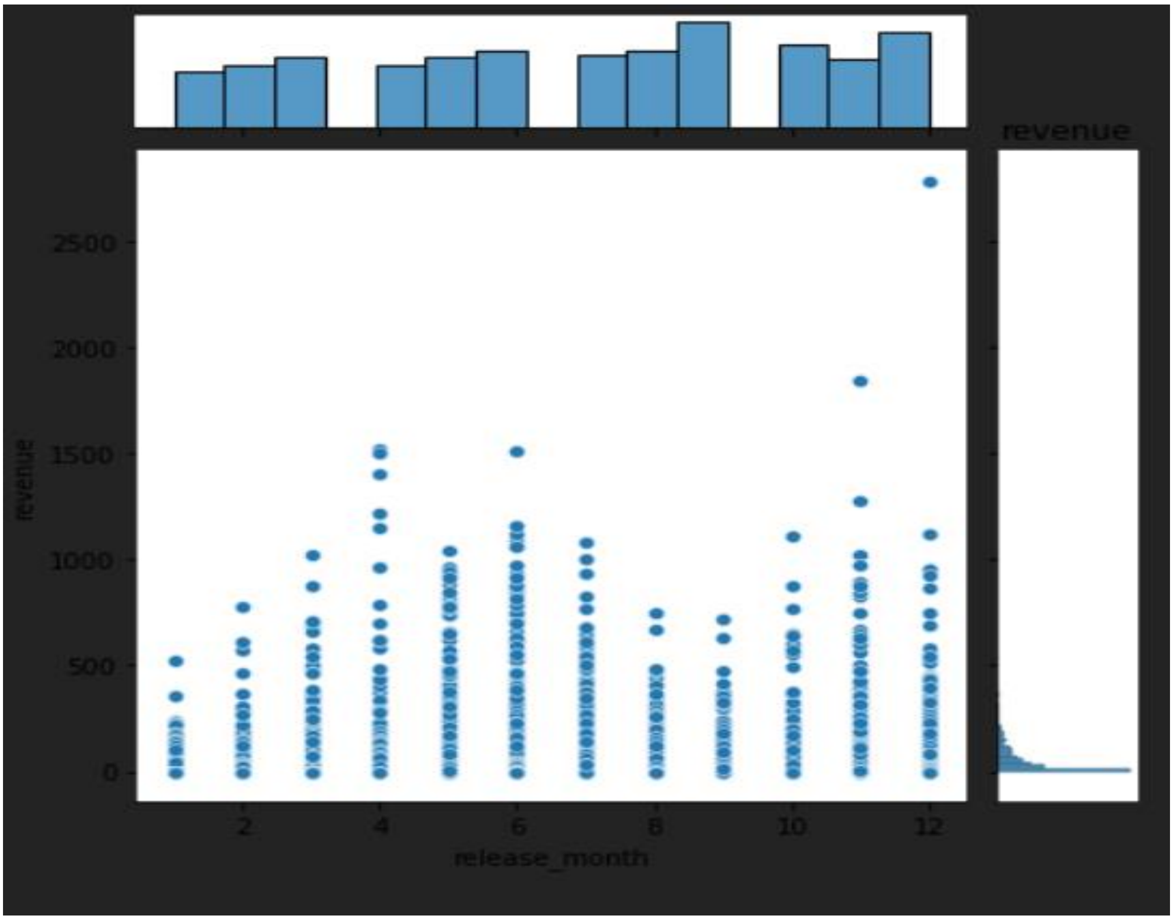


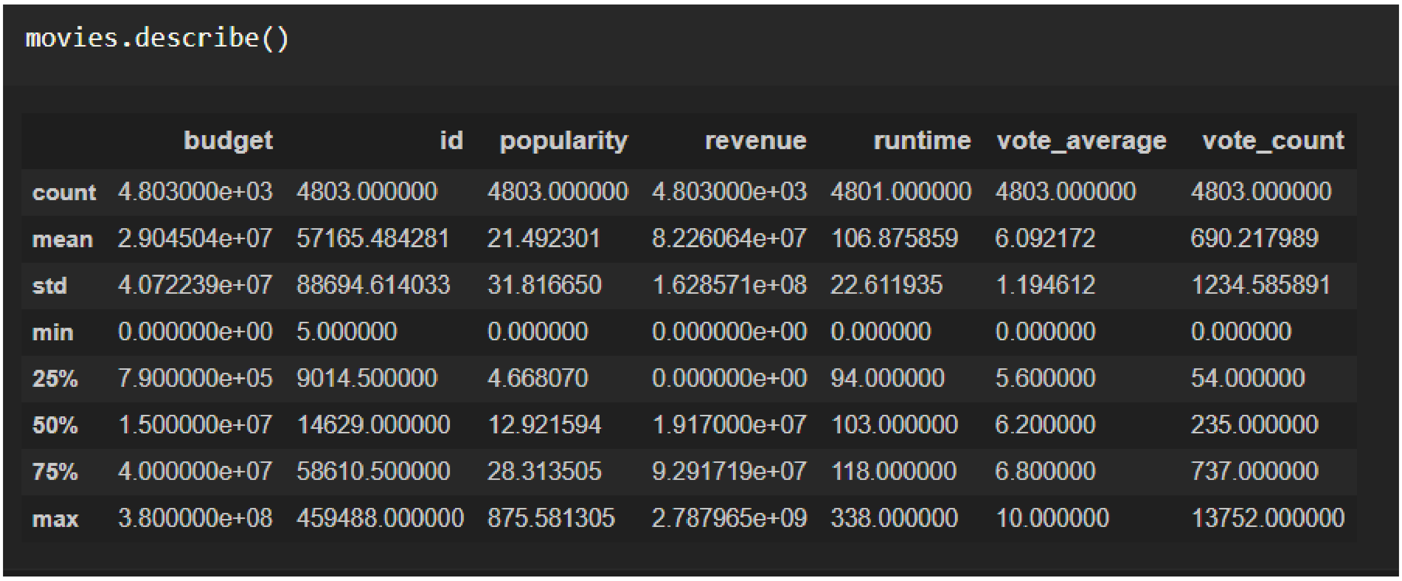
```
#comapring distribution of reveune and log revune side by side with histogram
fig, ax = plt.subplots(figsize = (16, 6))
plt.subplot(1, 2, 1)
plt.hist(movies['revenue']);
plt.title('Distribution of revenue');
plt.subplot(1, 2, 2)
plt.hist(movies['log_revenue']);
plt.title('Distribution of log transformation of revenue');
```



```
#let's create scatter plot
plt.figure(figsize=(16, 8))
plt.subplot(1, 2, 1)
plt.scatter(movies['budget'], movies['revenue'])
plt.title('Revenue vs budget fig(1)');
plt.subplot(1, 2, 2)
plt.scatter(movies['log_budget'], movies['log_revenue'])
plt.title('Log Revenue vs log budget fig(2)');
```

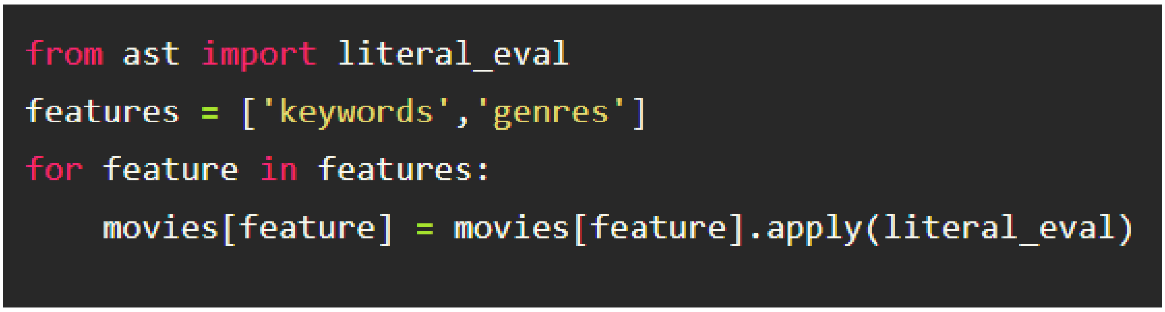

Revenue for movie with and w/o homepage
revenue
2500
2000
1500
1000
500
0
0
1
has_homepage


revenue
2500
2000
1500
1000
500
0
revenue
2
4
6
8
10
12
release_month

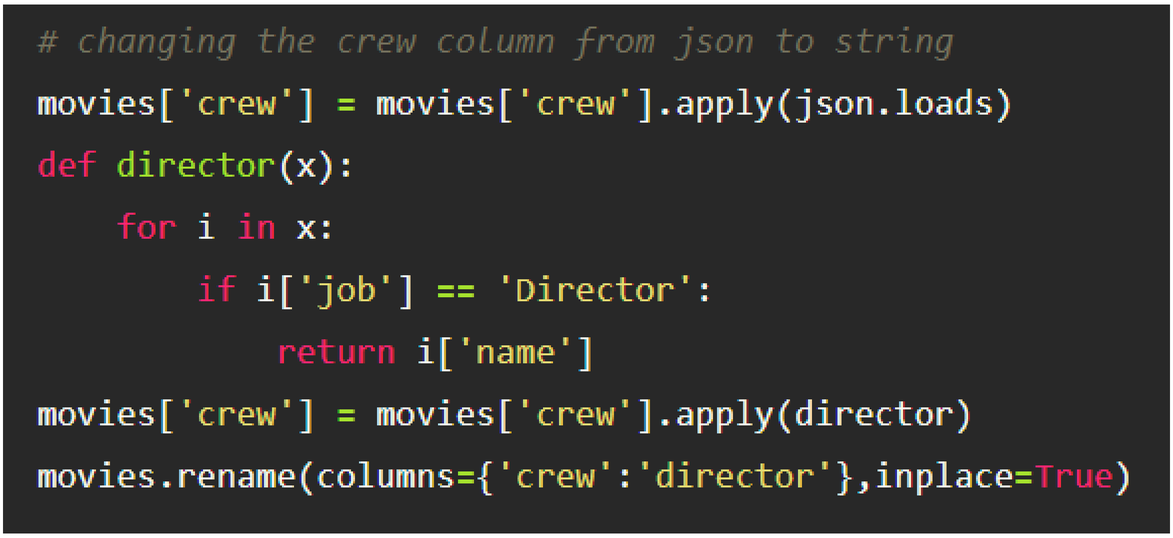


```
movies.describe()
```

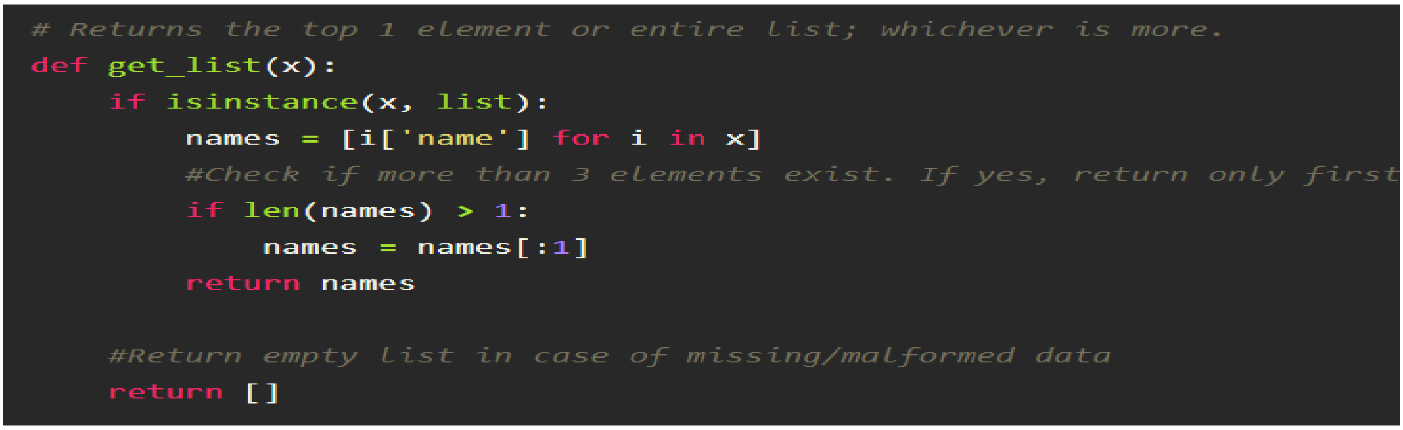
	budget	id	popularity	revenue	runtime	vote_average	vote_count
count	4.803000e+03	4803.000000	4803.000000	4.803000e+03	4801.000000	4803.000000	4803.000000
mean	2.904504e+07	57165.484281	21.492301	8.226064e+07	106.875859	6.092172	690.217989
std	4.072239e+07	88694.614033	31.816650	1.628571e+08	22.611935	1.194612	1234.585891
min	0.000000e+00	5.000000	0.000000	0.000000e+00	0.000000	0.000000	0.000000
25%	7.900000e+05	9014.500000	4.668070	0.000000e+00	94.000000	5.600000	54.000000
50%	1.500000e+07	14629.000000	12.921594	1.917000e+07	103.000000	6.200000	235.000000
75%	4.000000e+07	58610.500000	28.313505	9.291719e+07	118.000000	6.800000	737.000000
max	3.800000e+08	459488.000000	875.581305	2.787965e+09	338.000000	10.000000	13752.000000



```
from ast import literal_eval
features = ['keywords','genres']
for feature in features:
    movies[feature] = movies[feature].apply(literal_eval)
```

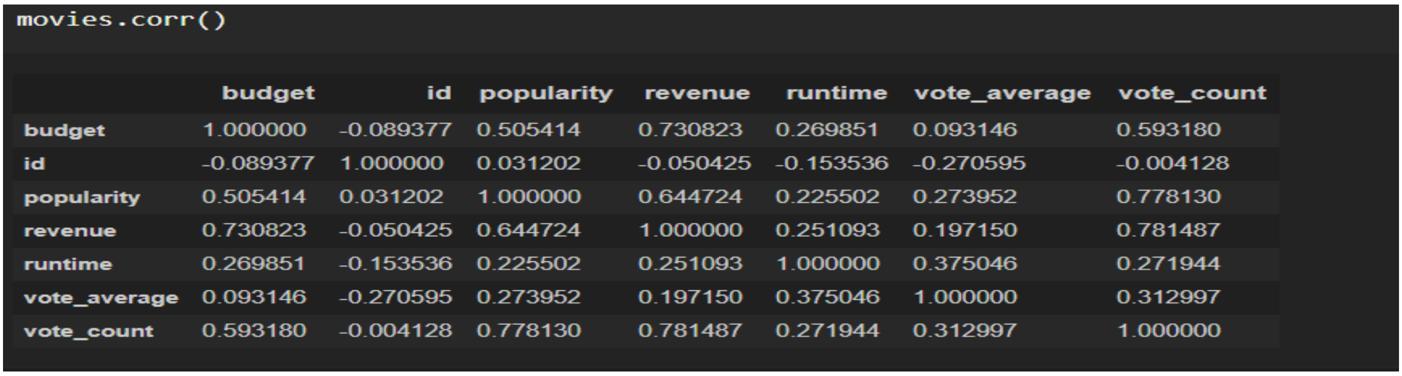


```
# changing the crew column from json to string
movies['crew'] = movies['crew'].apply(json.loads)
def director(x):
    for i in x:
        if i['job'] == 'Director':
            return i['name']
movies['crew'] = movies['crew'].apply(director)
movies.rename(columns={'crew':'director'},inplace=True)
```



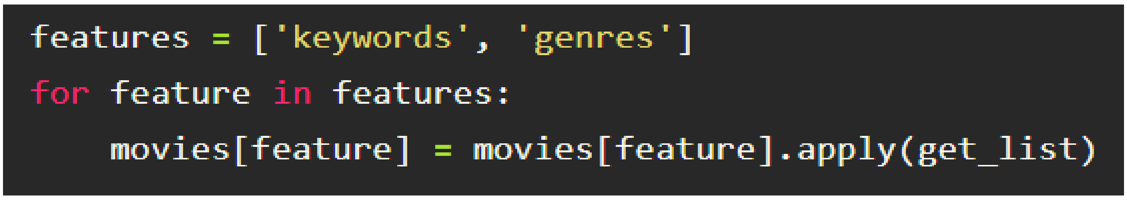
```
# Returns the top 1 element or entire list; whichever is more.
def get_list(x):
    if isinstance(x, list):
        names = [i['name'] for i in x]
        #Check if more than 3 elements exist. If yes, return only first
        if len(names) > 1:
            names = names[:1]
        return names

    #Return empty list in case of missing/malformed data
    return []
```

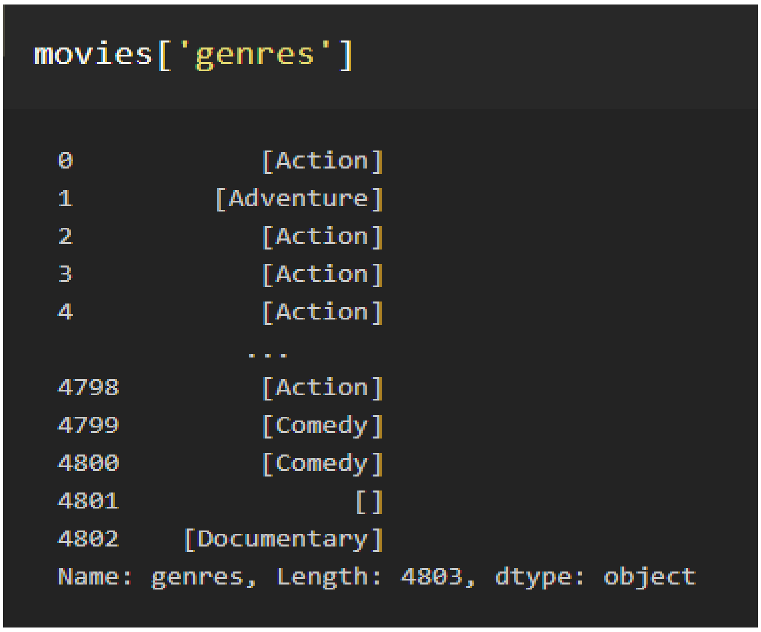


```
movies.corr()
```

	budget	id	popularity	revenue	runtime	vote_average	vote_count
budget	1.000000	-0.089377	0.505414	0.730823	0.269851	0.093146	0.593180
id	-0.089377	1.000000	0.031202	-0.050425	-0.153536	-0.270595	-0.004128
popularity	0.505414	0.031202	1.000000	0.644724	0.225502	0.273952	0.778130
revenue	0.730823	-0.050425	0.644724	1.000000	0.251093	0.197150	0.781487
runtime	0.269851	-0.153536	0.225502	0.251093	1.000000	0.375046	0.271944
vote_average	0.093146	-0.270595	0.273952	0.197150	0.375046	1.000000	0.312997
vote_count	0.593180	-0.004128	0.778130	0.781487	0.271944	0.312997	1.000000

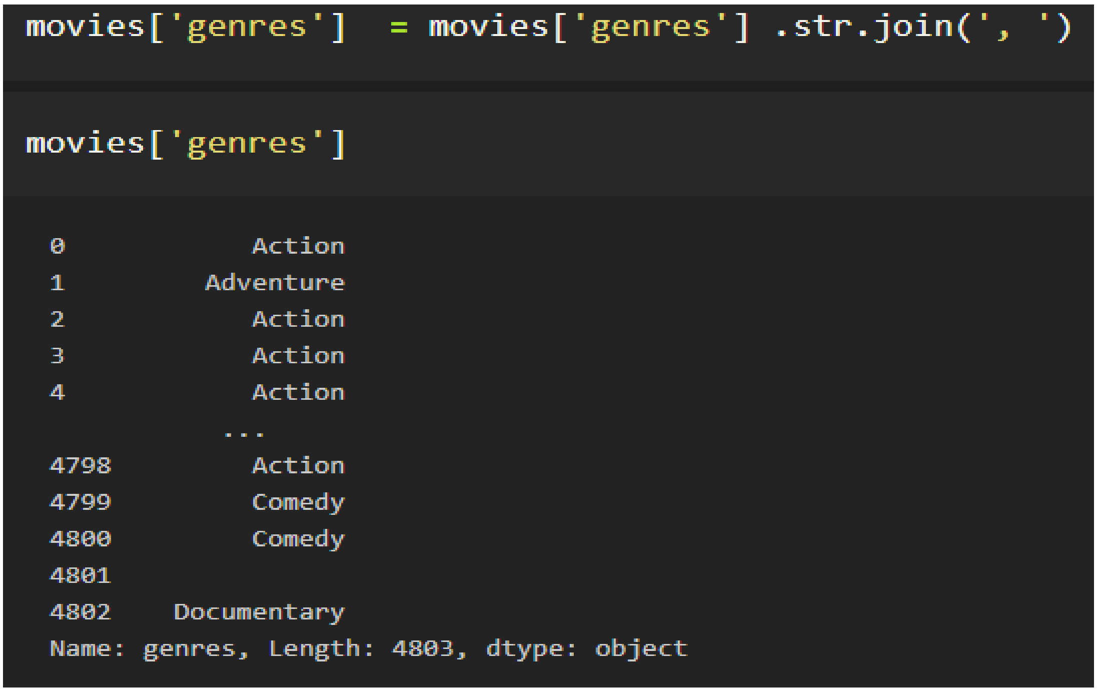


```
features = ['keywords', 'genres']
for feature in features:
    movies[feature] = movies[feature].apply(get_list)
```



```
movies['genres']
```

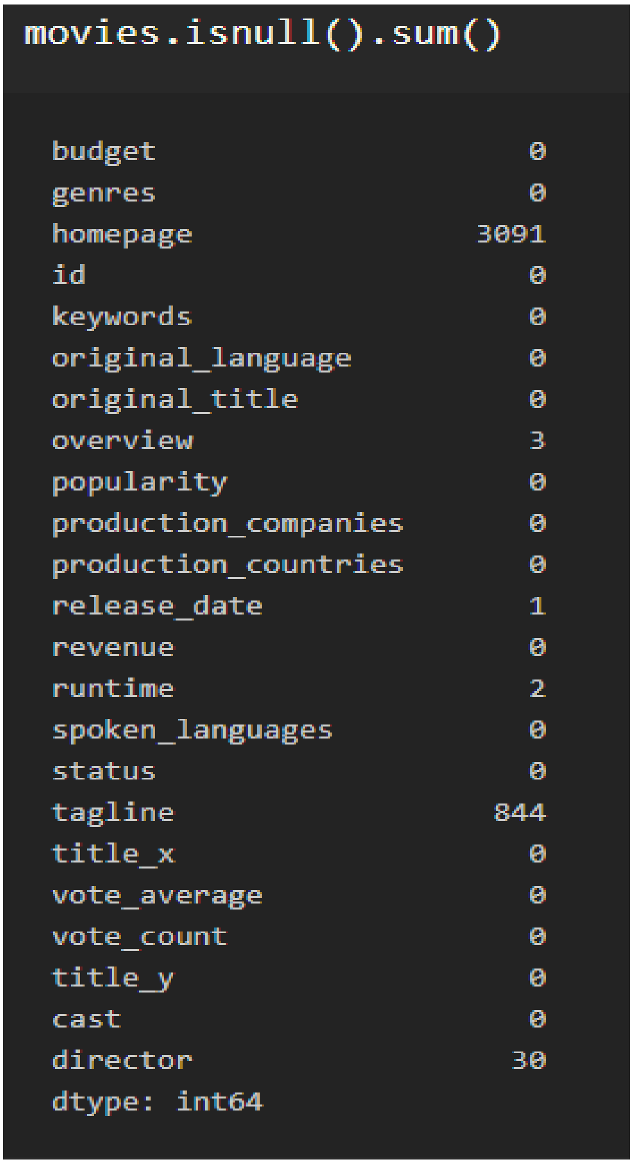
```
0              [Action]
1           [Adventure]
2              [Action]
3              [Action]
4              [Action]
              ...
4798           [Action]
4799           [Comedy]
4800           [Comedy]
4801                 []
4802      [Documentary]
Name: genres, Length: 4803, dtype: object
```



```
movies['genres']  = movies['genres'] .str.join(', ')
```

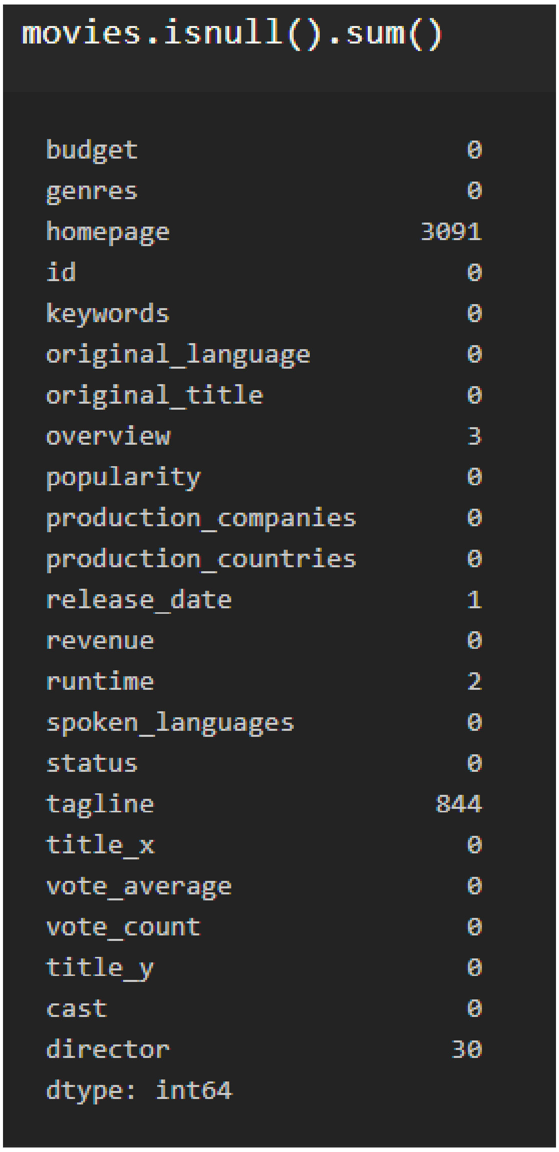
```
movies['genres']
```

```
0              Action
1           Adventure
2              Action
3              Action
4              Action
             ...
4798           Action
4799           Comedy
4800           Comedy
4801
4802      Documentary
Name: genres, Length: 4803, dtype: object
```



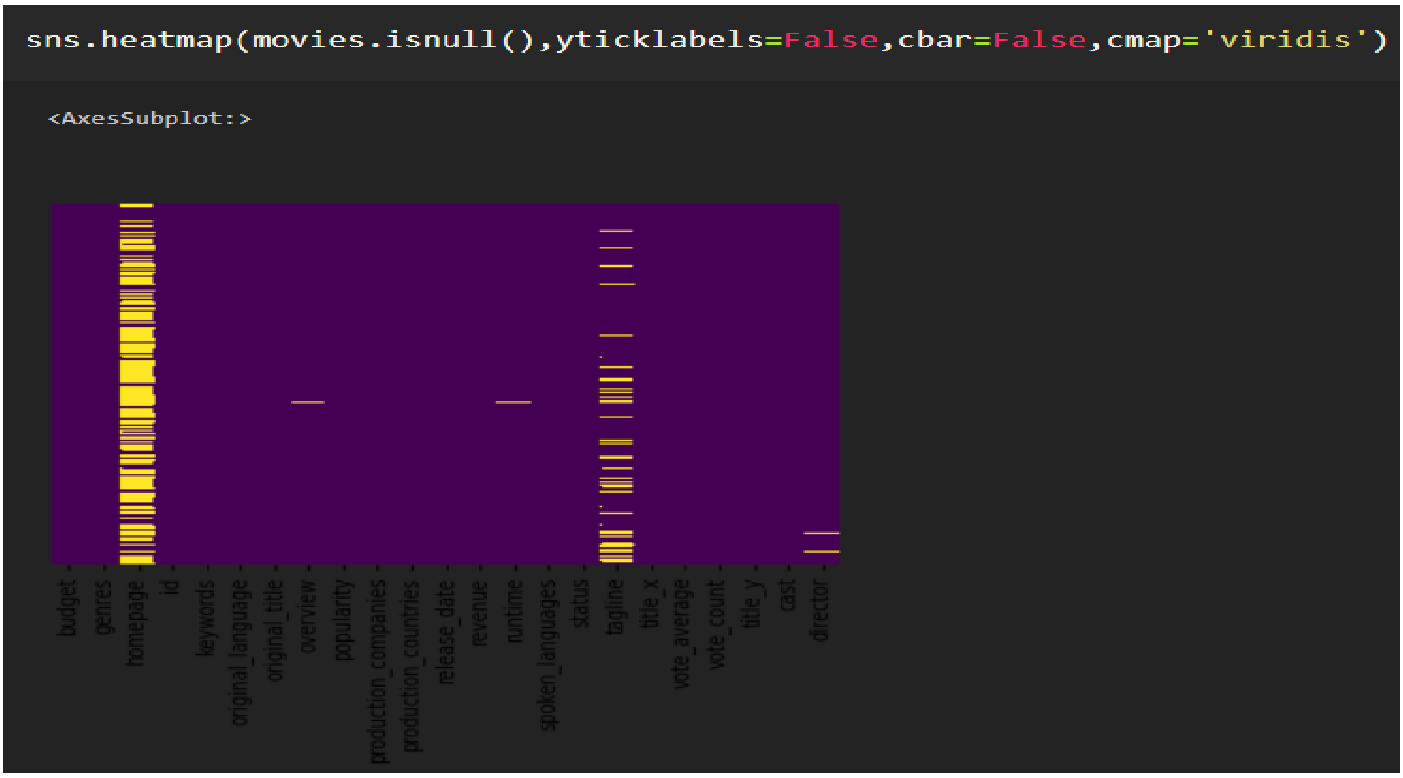
```
movies.isnull().sum()

budget                     0
genres                     0
homepage                3091
id                         0
keywords                   0
original_language          0
original_title             0
overview                   3
popularity                 0
production_companies       0
production_countries       0
release_date               1
revenue                    0
runtime                    2
spoken_languages           0
status                     0
tagline                  844
title_x                    0
vote_average               0
vote_count                 0
title_y                    0
cast                       0
director                  30
dtype: int64
```



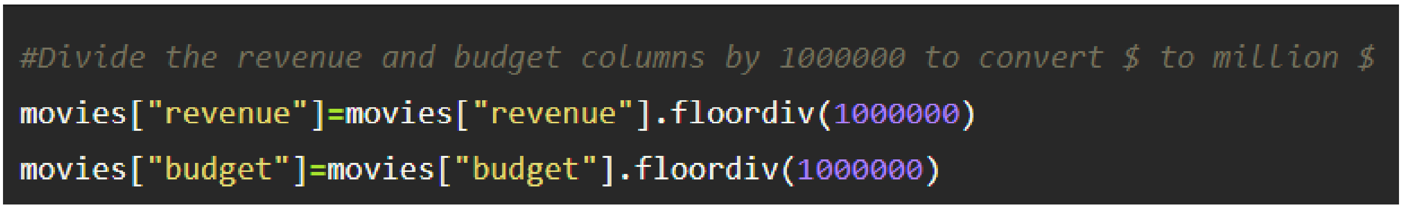
```
movies.isnull().sum()

budget                     0
genres                     0
homepage                3091
id                         0
keywords                   0
original_language          0
original_title             0
overview                   3
popularity                 0
production_companies       0
production_countries       0
release_date               1
revenue                    0
runtime                    2
spoken_languages           0
status                     0
tagline                  844
title_x                    0
vote_average               0
vote_count                 0
title_y                    0
cast                       0
director                  30
dtype: int64
```

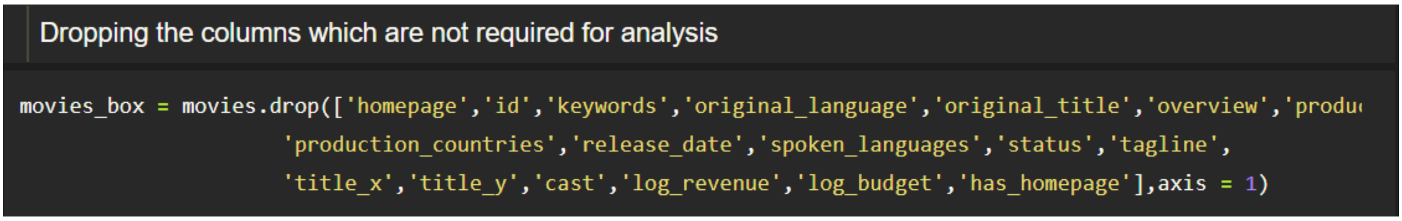


```
sns.heatmap(movies.isnull(),yticklabels=False,cbar=False,cmap='viridis')

<AxesSubplot:>
```

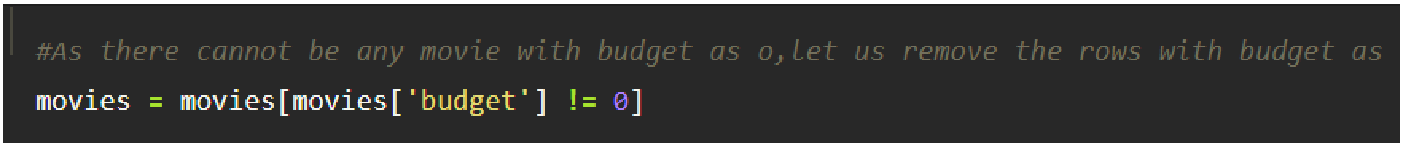


```
#Divide the revenue and budget columns by 1000000 to convert $ to million $
movies["revenue"]=movies["revenue"].floordiv(1000000)
movies["budget"]=movies["budget"].floordiv(1000000)
```

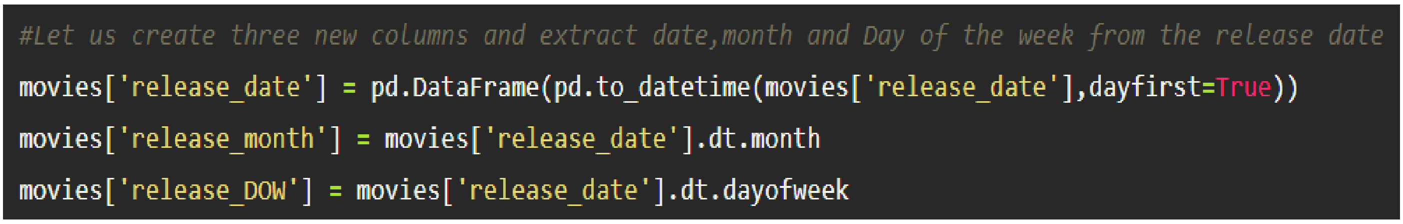


Dropping the columns which are not required for analysis

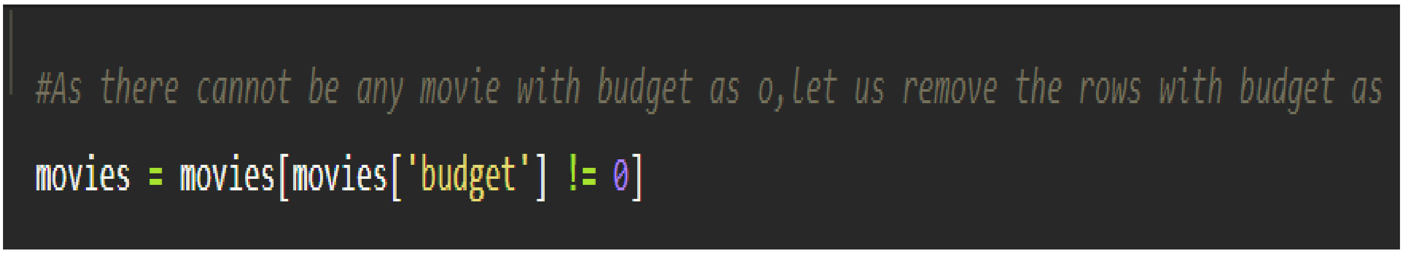
```
movies_box = movies.drop(['homepage','id','keywords','original_language','original_title','overview','produ
                  'production_countries','release_date','spoken_languages','status','tagline',
                  'title_x','title_y','cast','log_revenue','log_budget','has_homepage'],axis = 1)
```



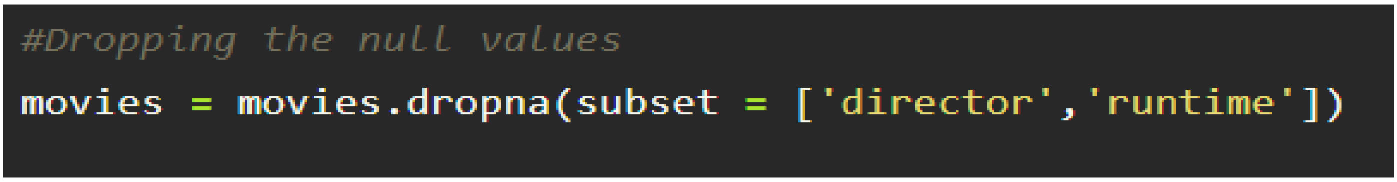
```
#As there cannot be any movie with budget as 0,let us remove the rows with budget as
movies = movies[movies['budget'] != 0]
```



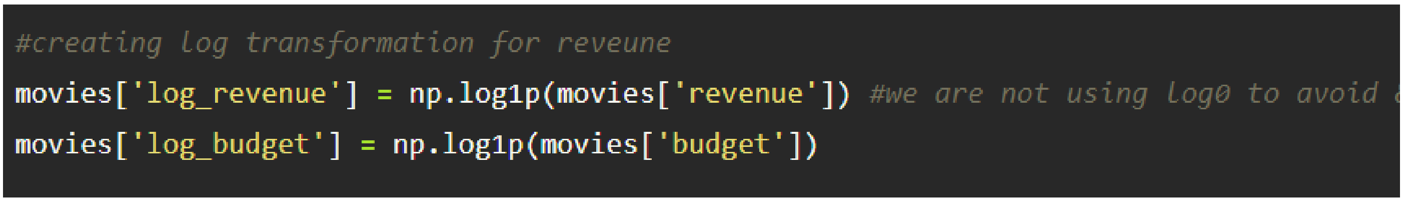
```
#Let us create three new columns and extract date,month and Day of the week from the release date
movies['release_date'] = pd.DataFrame(pd.to_datetime(movies['release_date'],dayfirst=True))
movies['release_month'] = movies['release_date'].dt.month
movies['release_DOW'] = movies['release_date'].dt.dayofweek
```



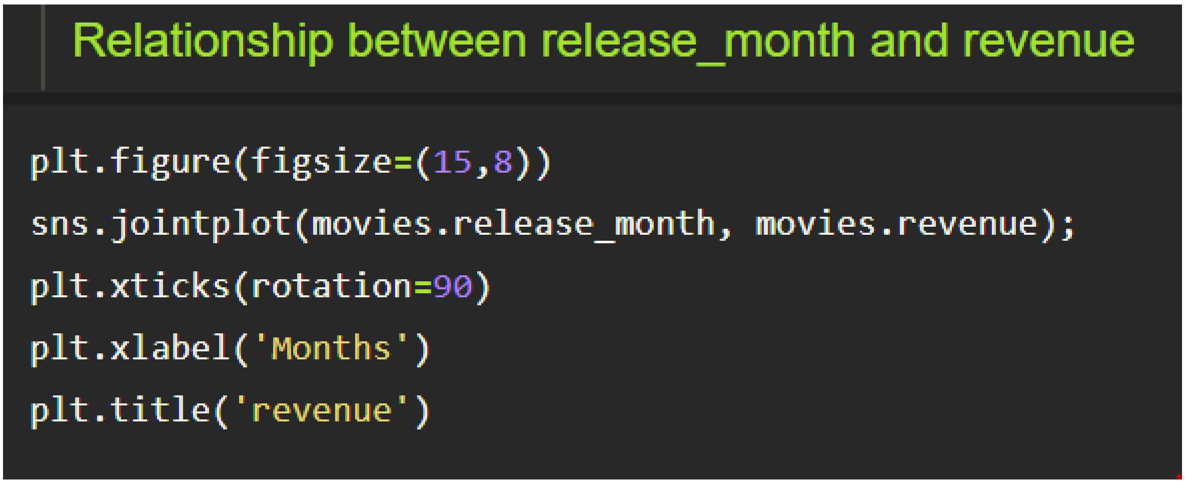
```
#As there cannot be any movie with budget as 0,let us remove the rows with budget as
movies = movies[movies['budget'] != 0]
```



```
#Dropping the null values
movies = movies.dropna(subset = ['director','runtime'])
```

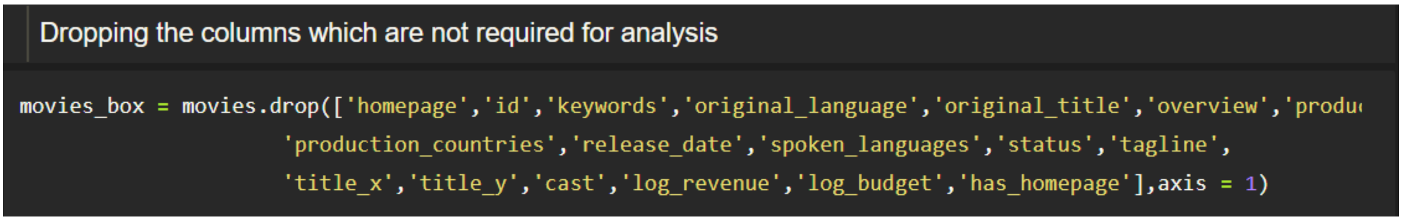


```
#creating log transformation for reveune
movies['log_revenue'] = np.log1p(movies['revenue']) #we are not using log0 to avoid
movies['log_budget'] = np.log1p(movies['budget'])
```



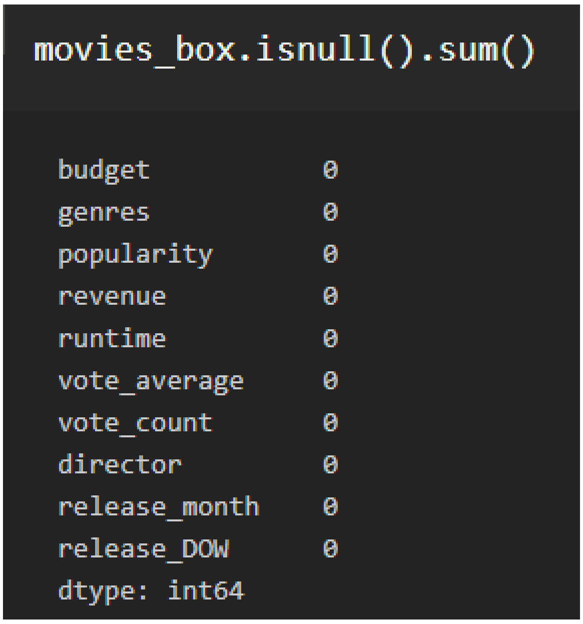
Relationship between release_month and revenue

```
plt.figure(figsize=(15,8))
sns.jointplot(movies.release_month, movies.revenue);
plt.xticks(rotation=90)
plt.xlabel('Months')
plt.title('revenue')
```



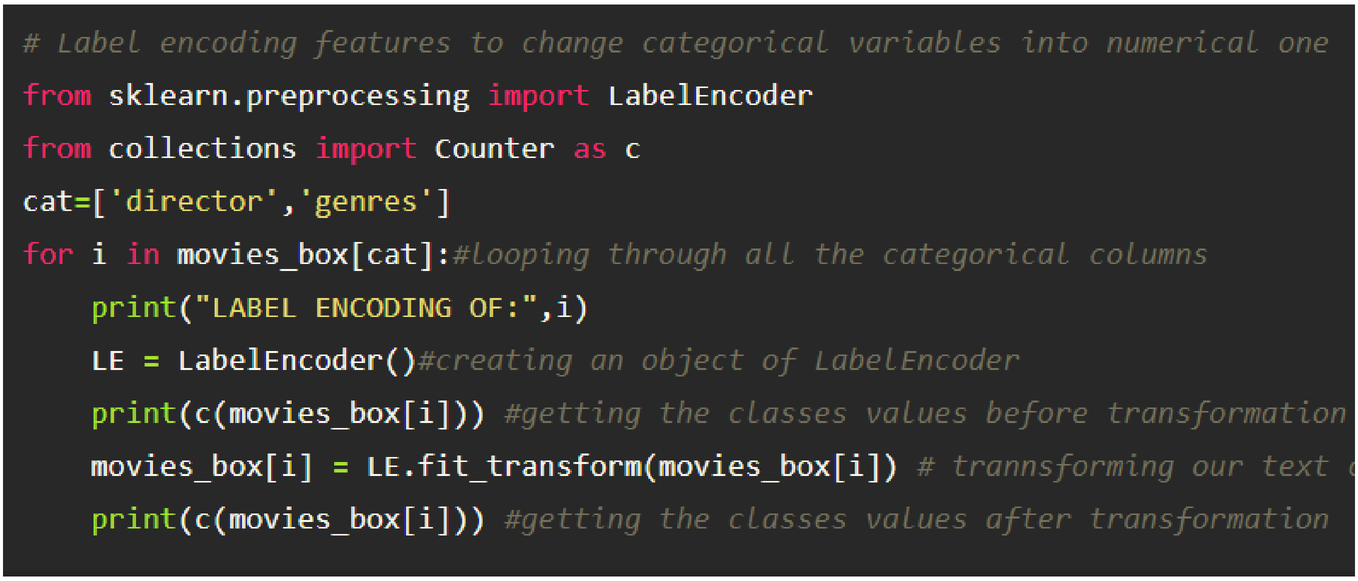
Dropping the columns which are not required for analysis

```
movies_box = movies.drop(['homepage','id','keywords','original_language','original_title','overview','produ
                        'production_countries','release_date','spoken_languages','status','tagline',
                        'title_x','title_y','cast','log_revenue','log_budget','has_homepage'],axis = 1)
```

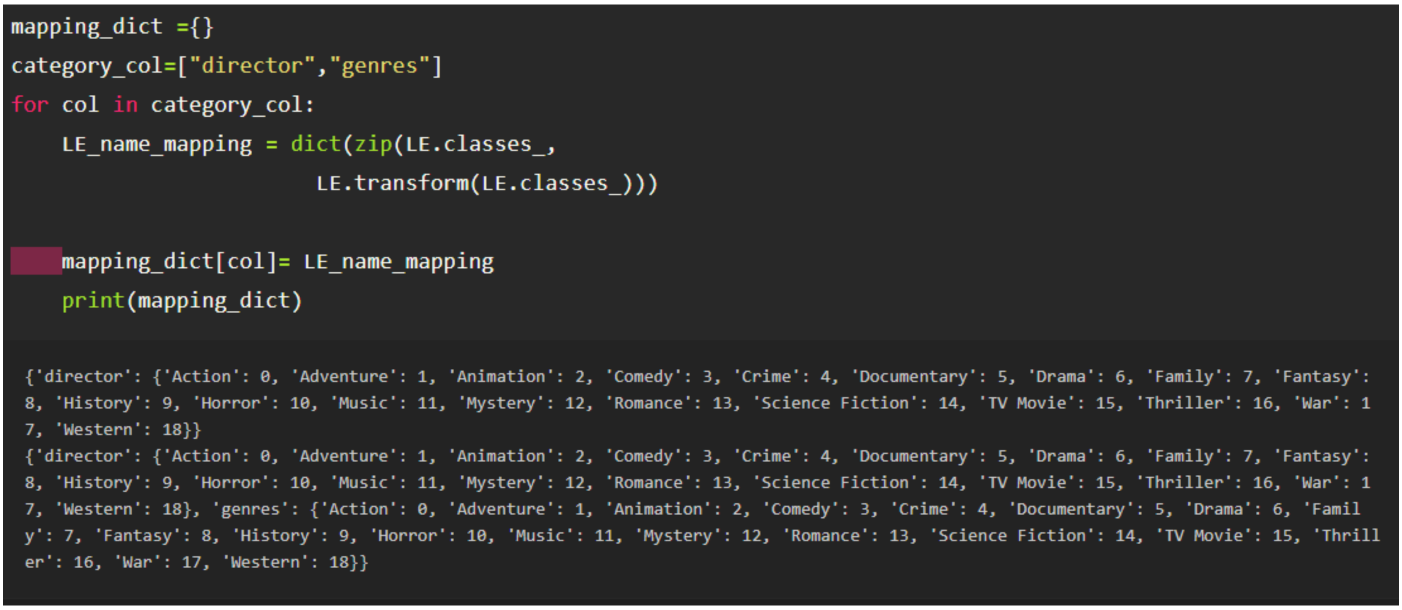


```
movies_box.isnull().sum()
```

```
budget           0
genres           0
popularity       0
revenue          0
runtime          0
vote_average     0
vote_count       0
director         0
release_month    0
release_DOW      0
dtype: int64
```



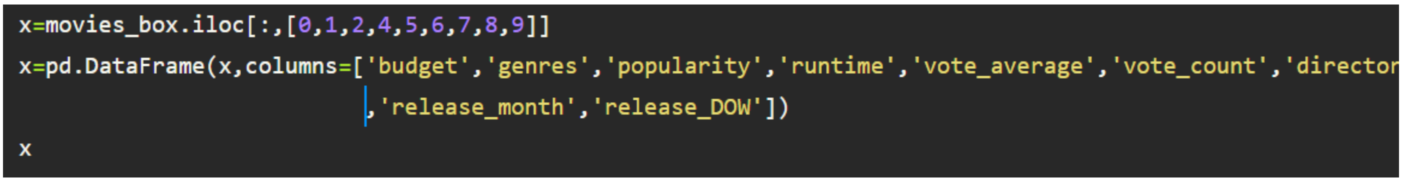
```
# Label encoding features to change categorical variables into numerical one
from sklearn.preprocessing import LabelEncoder
from collections import Counter as c
cat=['director','genres']
for i in movies_box[cat]:#looping through all the categorical columns
    print("LABEL ENCODING OF:",i)
    LE = LabelEncoder()#creating an object of LabelEncoder
    print(c(movies_box[i])) #getting the classes values before transformation
    movies_box[i] = LE.fit_transform(movies_box[i]) # trannsforming our text c
    print(c(movies_box[i])) #getting the classes values after transformation
```



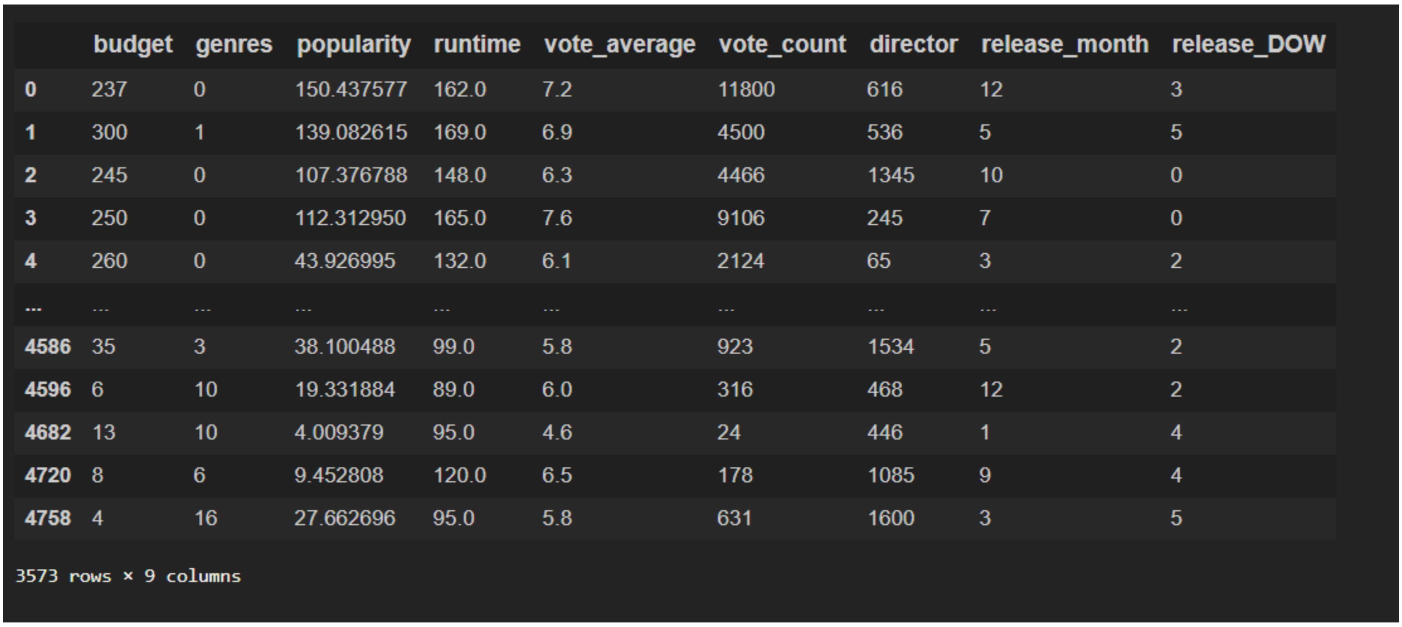
```
mapping_dict ={}
category_col=["director","genres"]
for col in category_col:
    LE_name_mapping = dict(zip(LE.classes_,
                        LE.transform(LE.classes_)))

    mapping_dict[col]= LE_name_mapping
    print(mapping_dict)
```

```
{'director': {'Action': 0, 'Adventure': 1, 'Animation': 2, 'Comedy': 3, 'Crime': 4, 'Documentary': 5, 'Drama': 6, 'Family': 7, 'Fantasy':
8, 'History': 9, 'Horror': 10, 'Music': 11, 'Mystery': 12, 'Romance': 13, 'Science Fiction': 14, 'TV Movie': 15, 'Thriller': 16, 'War': 1
7, 'Western': 18}}
{'director': {'Action': 0, 'Adventure': 1, 'Animation': 2, 'Comedy': 3, 'Crime': 4, 'Documentary': 5, 'Drama': 6, 'Family': 7, 'Fantasy':
8, 'History': 9, 'Horror': 10, 'Music': 11, 'Mystery': 12, 'Romance': 13, 'Science Fiction': 14, 'TV Movie': 15, 'Thriller': 16, 'War': 1
7, 'Western': 18}, 'genres': {'Action': 0, 'Adventure': 1, 'Animation': 2, 'Comedy': 3, 'Crime': 4, 'Documentary': 5, 'Drama': 6, 'Famil
y': 7, 'Fantasy': 8, 'History': 9, 'Horror': 10, 'Music': 11, 'Mystery': 12, 'Romance': 13, 'Science Fiction': 14, 'TV Movie': 15, 'Thrill
er': 16, 'War': 17, 'Western': 18}}
```

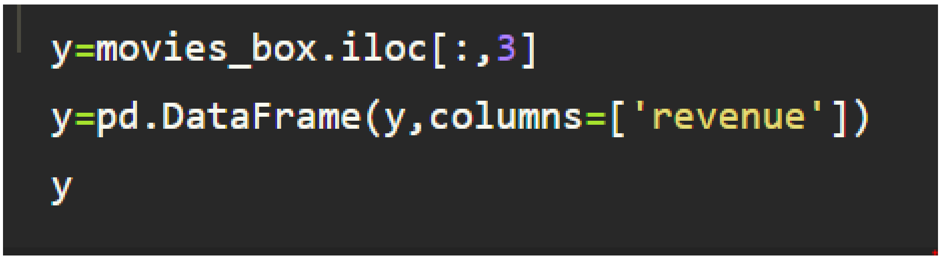


```
x=movies_box.iloc[:,[0,1,2,4,5,6,7,8,9]]
x=pd.DataFrame(x,columns=['budget','genres','popularity','runtime','vote_average','vote_count','director
                        ,'release_month','release_DOW'])
x
```

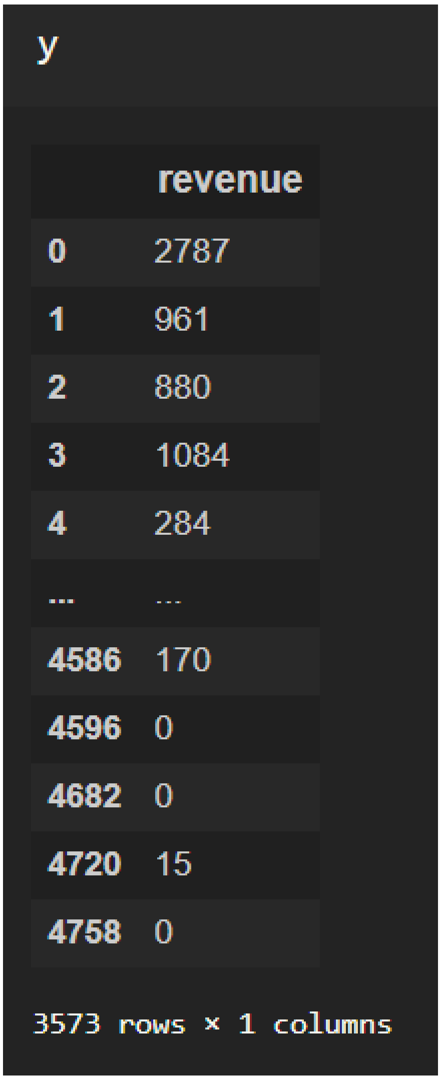


	budget	genres	popularity	runtime	vote_average	vote_count	director	release_month	release_DOW
0	237	0	150.437577	162.0	7.2	11800	616	12	3
1	300	1	139.082615	169.0	6.9	4500	536	5	5
2	245	0	107.376788	148.0	6.3	4466	1345	10	0
3	250	0	112.312950	165.0	7.6	9106	245	7	0
4	260	0	43.926995	132.0	6.1	2124	65	3	2
...	...	...	...	...	...	...	...	...	...
4586	35	3	38.100488	99.0	5.8	923	1534	5	2
4596	6	10	19.331884	89.0	6.0	316	468	12	2
4682	13	10	4.009379	95.0	4.6	24	446	1	4
4720	8	6	9.452808	120.0	6.5	178	1085	9	4
4758	4	16	27.662696	95.0	5.8	631	1600	3	5

3573 rows × 9 columns



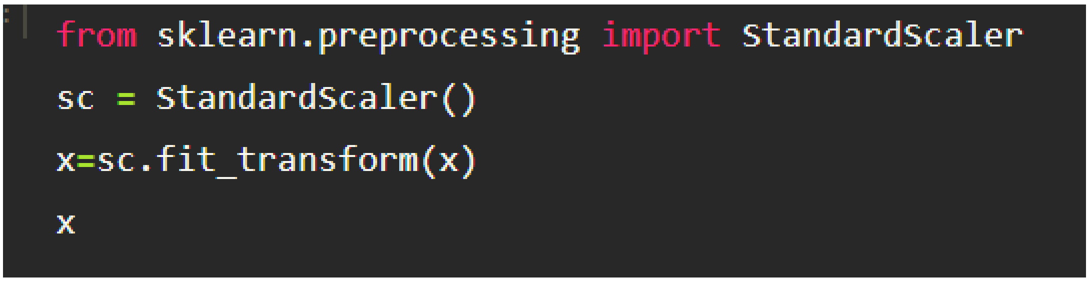
```
y=movies_box.iloc[:,3]
y=pd.DataFrame(y,columns=['revenue'])
y
```



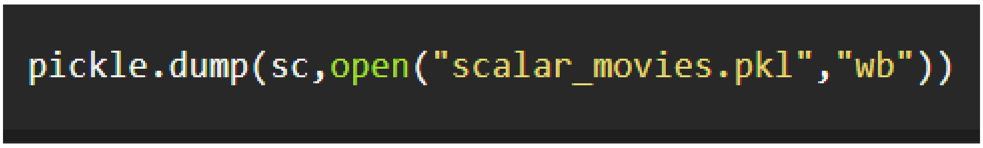
```
y
```

	revenue
0	2787
1	961
2	880
3	1084
4	284
...	...
4586	170
4596	0
4682	0
4720	15
4758	0

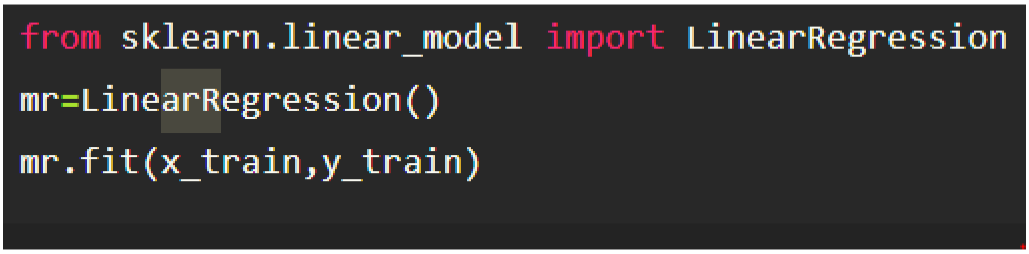
3573 rows × 1 columns



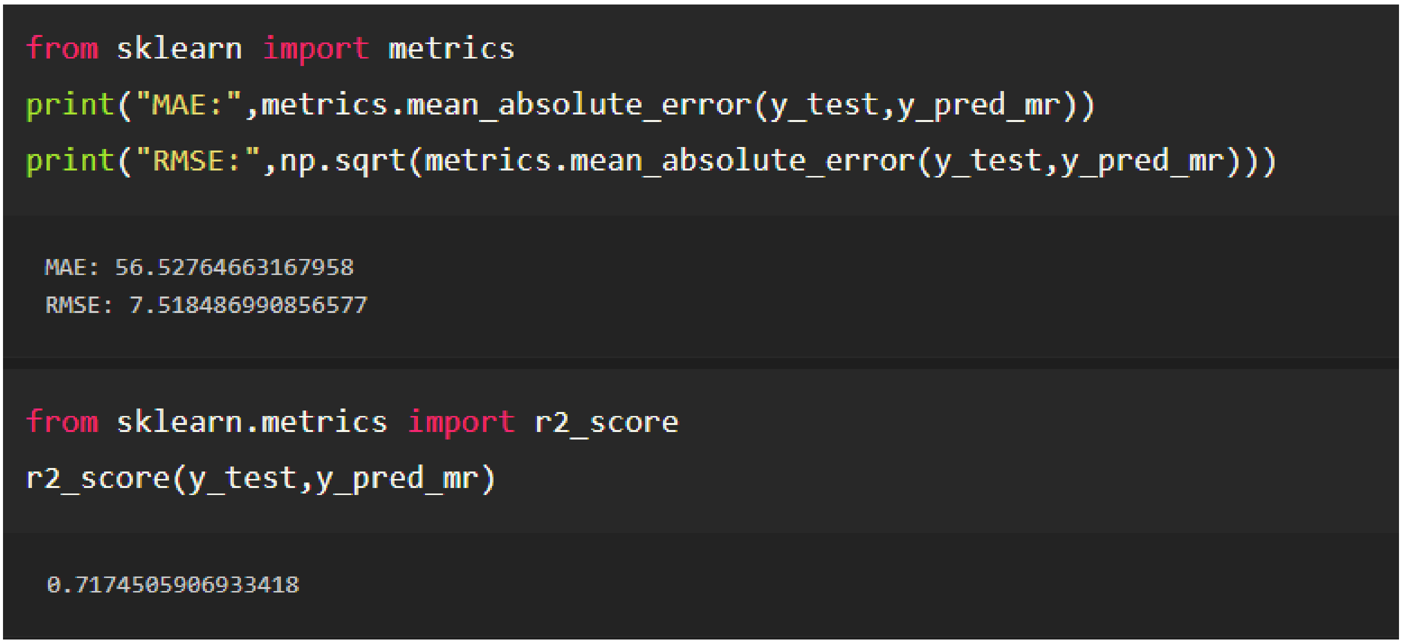
```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x=sc.fit_transform(x)
x
```



```
pickle.dump(sc,open("scalar_movies.pkl","wb"))
```



```
from sklearn.linear_model import LinearRegression
mr=LinearRegression()
mr.fit(x_train,y_train)
```

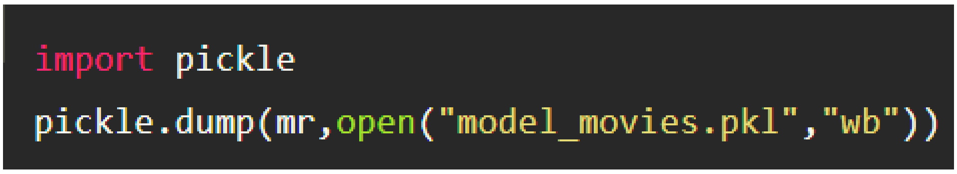


```
from sklearn import metrics
print("MAE:",metrics.mean_absolute_error(y_test,y_pred_mr))
print("RMSE:",np.sqrt(metrics.mean_absolute_error(y_test,y_pred_mr)))
```

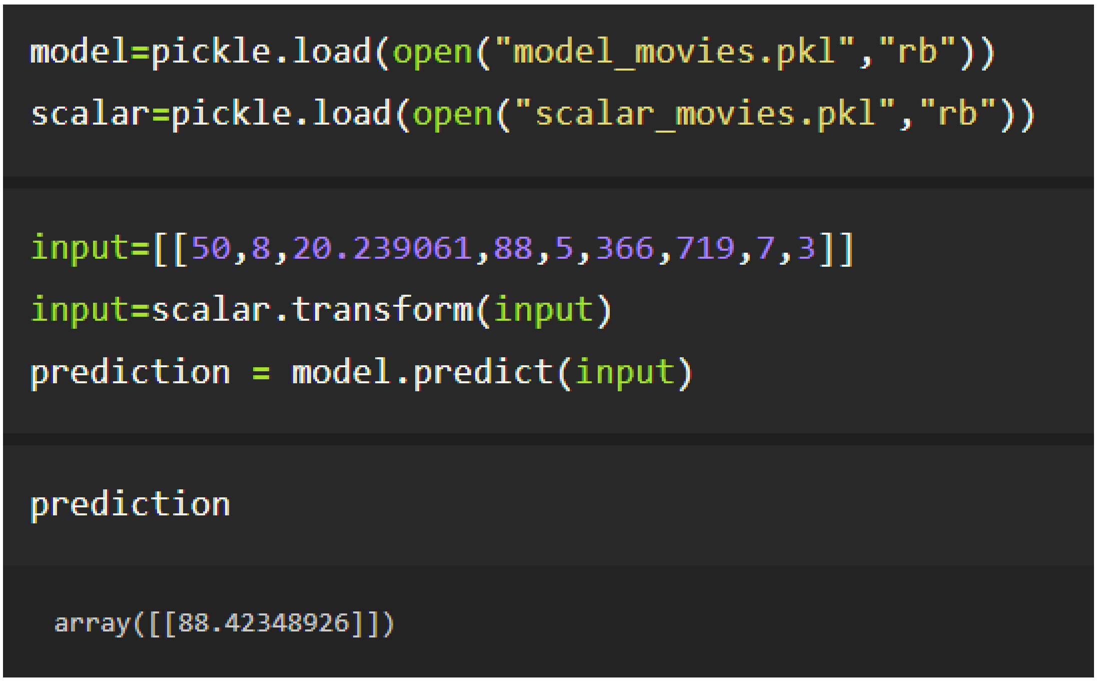
```
MAE: 56.52764663167958
RMSE: 7.518486990856577
```

```
from sklearn.metrics import r2_score
r2_score(y_test,y_pred_mr)
```

```
0.7174505906933418
```



```
import pickle
pickle.dump(mr,open("model_movies.pkl","wb"))
```

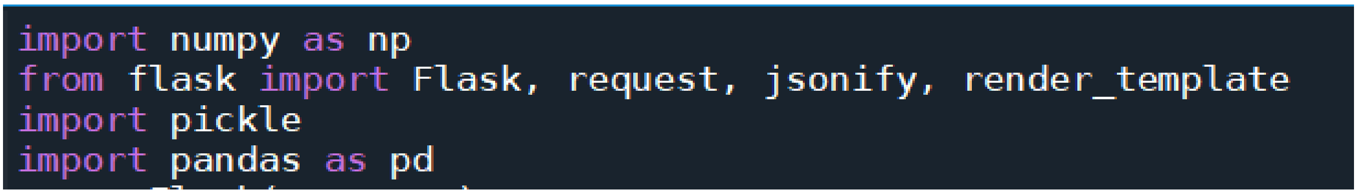


```
model=pickle.load(open("model_movies.pkl","rb"))
scalar=pickle.load(open("scalar_movies.pkl","rb"))
```

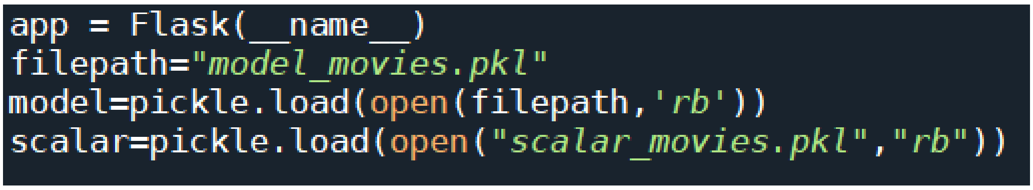
```
input=[[50,8,20.239061,88,5,366,719,7,3]]
input=scalar.transform(input)
prediction = model.predict(input)
```

```
prediction
```

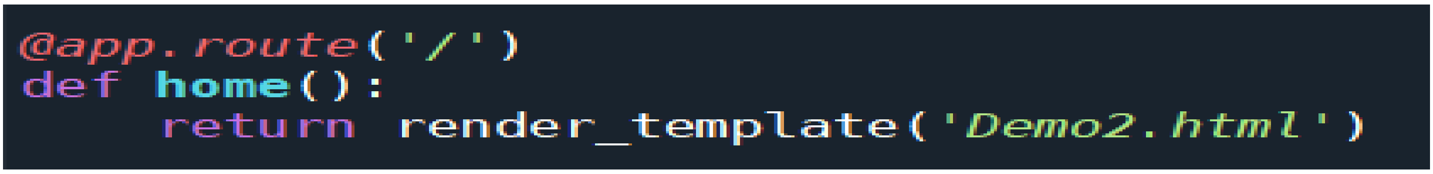
```
array([[88.42348926]])
```



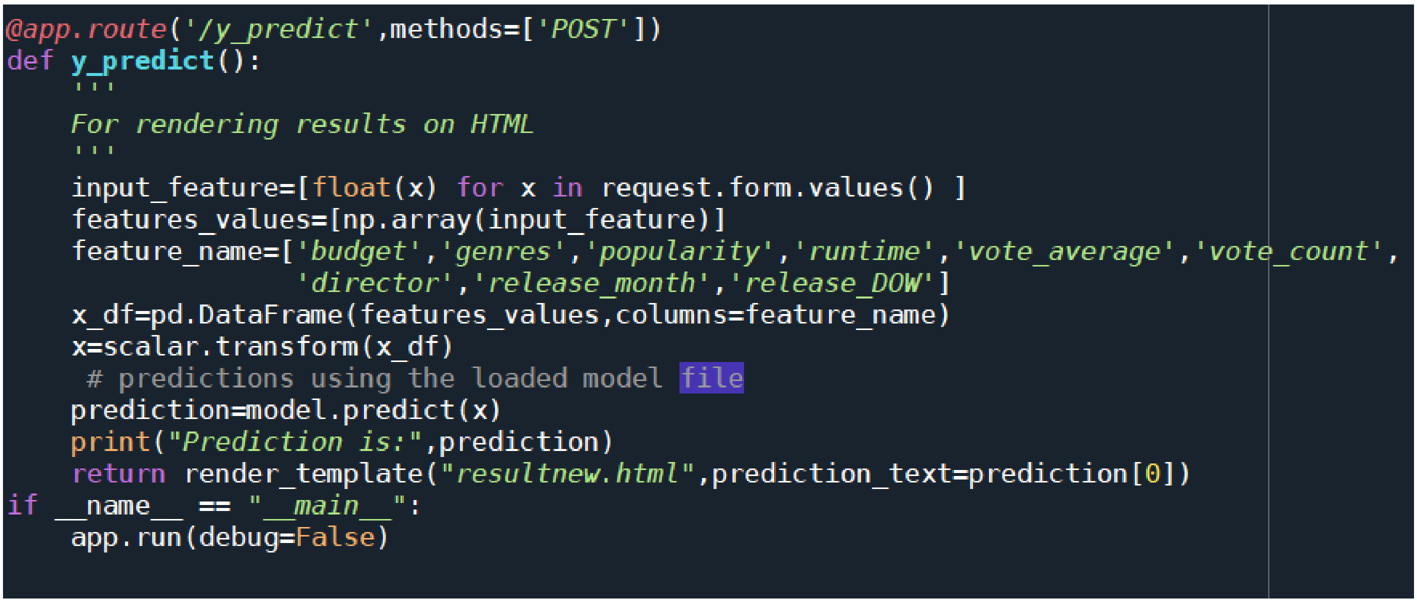
```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle
import pandas as pd
```



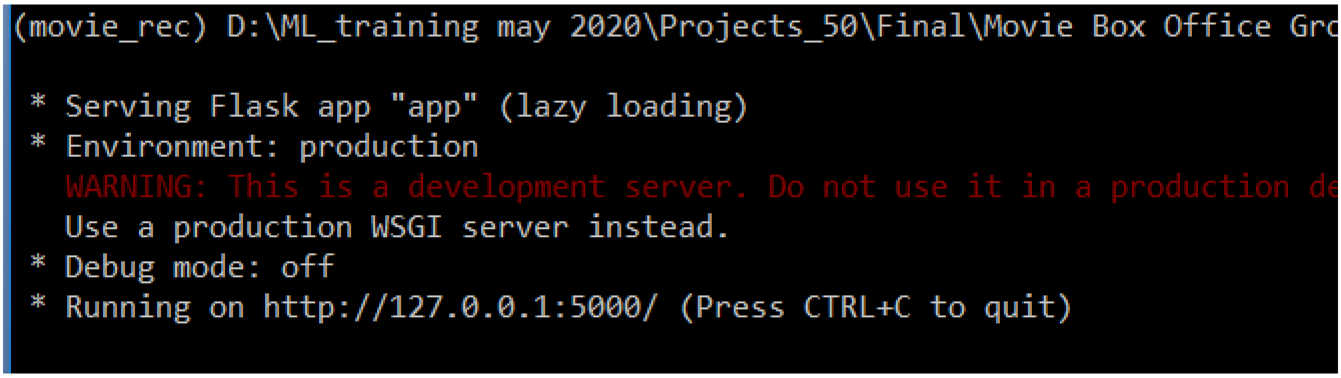
```
app = Flask(__name__)
filepath="model_movies.pkl"
model=pickle.load(open(filepath,'rb'))
scalar=pickle.load(open("scalar_movies.pkl","rb"))
```



```
@app.route('/')
def home():
    return render_template('Demo2.html')
```



```
@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML
    '''
    input_feature=[float(x) for x in request.form.values() ]
    features_values=[np.array(input_feature)]
    feature_name=['budget','genres','popularity','runtime','vote_average','vote_count',
                  'director','release_month','release_DOW']
    x_df=pd.DataFrame(features_values,columns=feature_name)
    x=scalar.transform(x_df)
     # predictions using the loaded model file
    prediction=model.predict(x)
    print("Prediction is:",prediction)
    return render_template("resultnew.html",prediction_text=prediction[0])
if __name__ == "__main__":
    app.run(debug=False)
```



```
(movie_rec) D:\ML_training may 2020\Projects_50\Final\Movie Box Office Gro

 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production de
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```