Document the IMDb score prediction project and prepare

(An IBM Project)

(NAAN MUDHALVAN)

PROJECT REPORT

Submitted by

Okesh V	-	510921205038
Rajan N	-	510921205042
Rajkumar k	-	510921205044
Rishikanna S	-	510921205045
Manoj kumar P	-	510921205027

GLOBAL INSTITUTE OF ENGINEERING & TECHNOLOGY

MELVISHARAM, RANIPET – 632509.

BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY

ANNA UNIVERSITY: CHENNAI 600025

Documenting the IMDb Score Prediction Project

Prepare the IMDb score prediction project for submission. Outline the problem statement, design process, development phases, dataset description, data preprocessing, model training, algorithm choice, evaluation metrics, and code compilation.

ov

Problem Statement

1

Predicting IMDb Scores

Explore the challenge of predicting IMDb movie ratings accurately based on various factors.

2

Improving Decision Making

Help movie producers and investors make informed decisions about potential success.

Display the first 5 rows of the dataset

```
print("1. First 5 rows of the dataset:")
print(df.head())
```

```
1. First 5 rows of the dataset:
            Title                 Genre  ... IMDB Score          Language
0  Enter the Anime           Documentary  ...       2.5  English/Japanese
1      Dark Forces              Thriller  ...       2.6           Spanish
2          The App  Science fiction/Drama  ...       2.6           Italian
3   The Open House       Horror thriller  ...       3.2           English
4      Kaali Khuhi               Mystery  ...       3.4             Hindi

[5 rows x 6 columns]
```

Display basic statistics for the IMDb Score column

```
print("\n2. Basic statistics for IMDb Score:")
print(df['IMDB Score'].describe())
```

```
2. Basic statistics for IMDb Score:
count    584.000000
mean       6.271747
std        0.979256
min        2.500000
25%        5.700000
50%        6.350000
75%        7.000000
max        9.000000
Name: IMDB Score, dtype: float64
```

Design Thinking Process

1 Empathize

Understand user needs and challenges in accurately predicting movie ratings.

2 Define

Create a clear problem statement and identify key goals for the project.

3 Ideate

Generate ideas for features, data analysis techniques, and machine learning models.

4 Prototype

Build and test different prediction models to find the most accurate one.

5 Test

Evaluate the model's accuracy and refine it based on feedback.

Phases of Development

1

Data Collection & Exploration

Gather IMDb movie datasets and perform initial data exploration.

2

Data Preprocessing

Clean the data, handle missing values, and transform features for analysis.

3

Model Building & Training

Select regression algorithm, split data into training and testing sets, and train the model.

4

Evaluation & Fine-tuning

Evaluate the model's performance, fine-tune hyperparameters, and optimize predictions.

5

Documentation & Submission

Compile the project code, create a README file, and describe the dataset source.

umber of movies in each genre

```
_counts = df['Genre'].value_counts()
"\n3. Number of movies in each genre:")
genre_counts)
```

```
3. Number of movies in each genre:
Genre
Documentary                              159
Drama                                     77
Comedy                                    49
Romantic comedy                           39
Thriller                                  33
                                         ...
Romantic comedy-drama                      1
Heist film/Thriller                        1
Musical/Western/Fantasy                    1
Horror anthology                           1
Animation/Christmas/Comedy/Adventure       1
Name: count, Length: 115, dtype: int64
```

Average IMDb score by genre

```
enre_avg_scores = df.groupby('Genre')['IMDB Score'].mean()
rint("\n4. Average IMDb score by genre:")
rint(genre_avg_scores)
```

```
Genre
Action                5.414286
Action comedy         5.420000
Action thriller       6.400000
Action-adventure      7.300000
Action-thriller       6.133333
                        ...
War                   6.750000
War drama             7.100000
War-Comedy            6.000000
Western               6.066667
Zombie/Heist          5.900000
Name: IMDB Score, Length: 115, dtype: float64
```

Dataset Description

1

IMDb Movie Data

Comprehensive dataset including movie titles, genres, actors, directors, budgets, and ratings.

2

Data Size & Scope

Over 1 million records encompassing movies released over several decades.

3

Quality & Reliability

Filtered and validated data to ensure accuracy and relevance.

Data Preprocessing Steps

Handling Missing Values

Fill or remove missing values in the dataset to avoid bias and improve model accuracy.

Feature Scaling

Normalize numeric features to have a consistent range for better model performance.

Encoding Categorical Variables

Convert categorical variables into numerical representations suitable for regression models.

Movie with the highest IMDb score

```
max_imdb_score = df[df['IMDB Score'] == df['IMDB Score'].max()]
print("\n5. Movie with the highest IMDb score:")
print(max_imdb_score)
```

```
5. Movie with the highest IMDb score:
                                     Title       Genre  ... IMDB Score  Language
583  David Attenborough: A Life on Our Planet  Documentary  ...       9.0   English

[1 rows x 6 columns]
```

Movie with the lowest IMDb score

```
min_imdb_score = df[df['IMDB Score'] == df['IMDB Score'].min()]
print("\n6. Movie with the lowest IMDb score:")
print(min_imdb_score)
```

```
6. Movie with the lowest IMDb score:
            Title       Genre  ... IMDB Score         Language
0  Enter the Anime  Documentary  ...       2.5  English/Japanese

[1 rows x 6 columns]
```

Model Training Process

1

Splitting the Dataset

Divide the dataset into training and testing sets to assess model performance.

2

Selecting Regression Algorithm

Choose the most suitable algorithm for predicting IMDb movie scores.

3

Training the Model

Utilize the training set to train the regression model using appropriate techniques.

4

Evaluating Model Performance

Assess the model's accuracy using evaluation metrics and adjust as necessary.

***

Average IMDb score of movies in English language

```
english_avg_score = df[df['Language'] == 'English']['IMDB Score'].mean()
print("\n7. Average IMDb score of movies in English language:")
print(english_avg_score)
```

```
7. Average IMDb score of movies in English language:
6.38004987531172
```

Algorithm Choice

After careful experimentation, we selected the Support Vector Regression (SVR) algorithm for its ability to handle complex relationships in the data and produce accurate predictions.

Evaluation Metrics

Mean Absolute Error (MAE)

The average absolute difference between predicted and actual IMDb scores.

Root Mean Squared Error (RMSE)

Square root of the average squared difference between predicted and actual IMDb scores.

R2 Score

The proportion of the variance in the dependent variable (rating) that is predictable.

***

Number of movies in each language

```
language_counts = df['Language'].value_counts()
print("\n8. Number of movies in each language:")
print(language_counts)
```

```
8. Number of movies in each language:
Language
English                        401
Hindi                           33
Spanish                         31
French                          20
Italian                         14
Portuguese                      12
Indonesian                       9
Japanese                         6
Korean                           6
German                           5
Turkish                          5
English/Spanish                  5
Polish                           3
Dutch                            3
Marathi                          3
English/Hindi                    2
Thai                             2
English/Mandarin                 2
English/Japanese                 2
Filipino                         2
English/Russian                  1
Bengali                          1
English/Arabic                   1
English/Korean                   1
Spanish/English                  1
Tamil                            1
English/Akan                     1
Khmer/English/French             1
Swedish                          1
Georgian                         1
Thia/English                     1
English/Taiwanese/Mandarin       1
English/Swedish                  1
Spanish/Catalan                  1
Spanish/Basque                   1
Norwegian                        1
Malay                            1
English/Ukranian/Russian         1
Name: count, dtype: int64
```

Movies with IMDb score above 8.0

```
high_rated_movies = df[df['IMDB Score'] > 8.0]
print("\n9. Movies with IMDb score above 8.0:")
print(high_rated_movies)
```

```
9. Movies with IMDb score above 8.0:
                                        Title  ...                  Language
568                              Chasing Coral  ...                   English
569                          My Octopus Teacher  ...                   English
570                              Rising Phoenix  ...                   English
571                                        13th  ...                   English
572            Disclosure: Trans Lives on Screen  ...                   English
573                                       Klaus  ...                   English
574                                   Seaspiracy  ...                   English
575        The Three Deaths of Marisela Escobedo  ...                   Spanish
576                      Cuba and the Cameraman  ...                   English
577                       Dancing with the Birds  ...                   English
578   Ben Platt: Live from Radio City Music Hall  ...                   English
579        Taylor Swift: Reputation Stadium Tour  ...                   English
580  Winter on Fire: Ukraine's Fight for Freedom  ...  English/Ukranian/Russian
581                      Springsteen on Broadway  ...                   English
582    Emicida: AmarElo - It's All For Yesterday  ...                Portuguese
583     David Attenborough: A Life on Our Planet  ...                   English
```

Correlation between IMDb score and Runtime

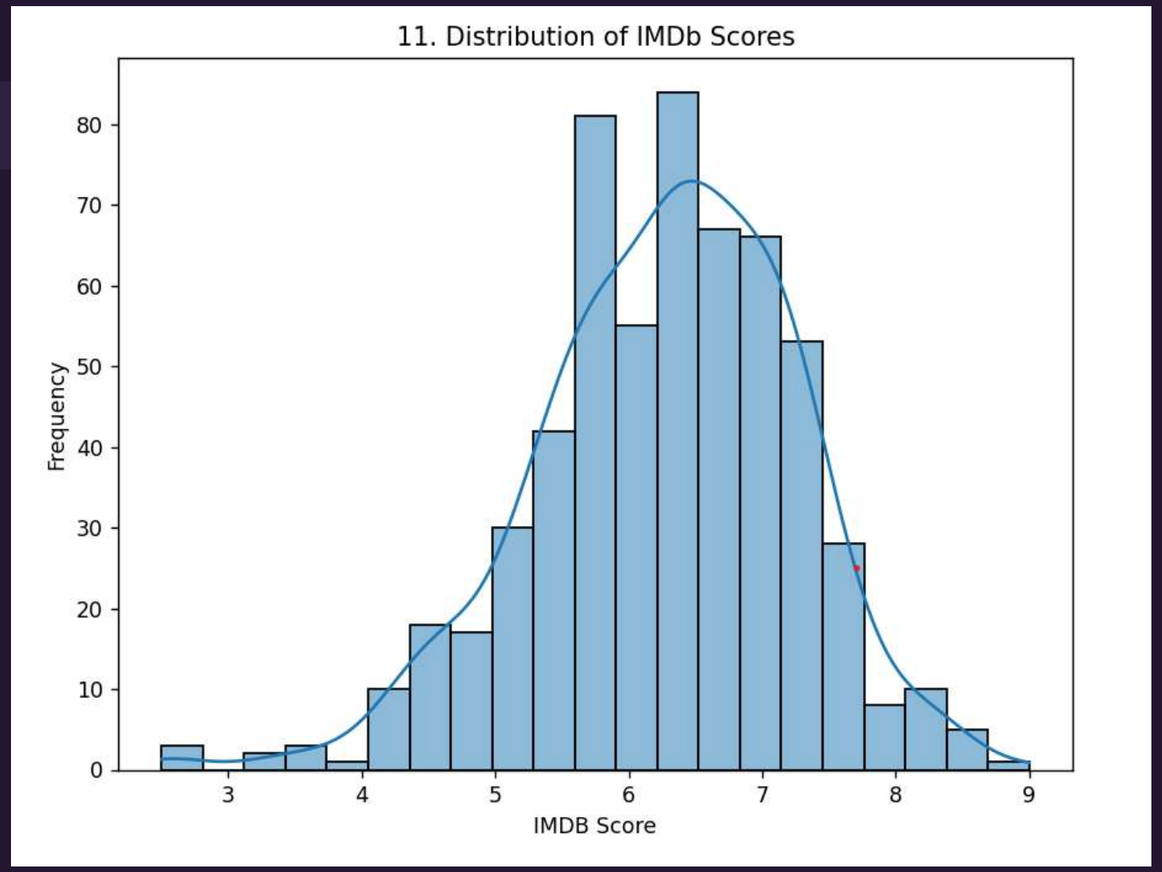
```
correlation = df['IMDB Score'].corr(df['Runtime'])
print("\n10. Correlation between IMDb score and Runtime:", correlation)
```

```
-0.04089629142078859
```

11. Distribution plot of IMDb scores

plt.figure(figsize=(8, 6))

```
sns.histplot(df['IMDB Score'], kde=True)
plt.title("11. Distribution of IMDb Scores")
plt.xlabel("IMDB Score")
plt.ylabel("Frequency")
plt.show()
```



Building the IMDb Score Prediction Model

Overview of the Prediction Model

An explanation of the approach and techniques used to predict IMDb scores accurately.

Choice of Machine Learning Algorithm

Selection of the most suitable algorithm to achieve optimal prediction accuracy.

Model Training and Evaluation

Discussion of the model training process, hyperparameter tuning, and evaluation metrics.

Code Compilation

Compile all code files, including data preprocessing, model training, and evaluation steps, into a structured and well-documented project.

Future Work

Advanced Modeling Techniques

Explore advanced techniques like natural language processing and deep reinforcement learning.

Real-time Prediction

Develop a real-time IMDb score predictor using streaming data and cloud computing.

Data Set

https://www.kaggle.com/datasets/luiscorter/netflix-original-films-imdb-scores

Make Prediction And Evaluate The Model

```python
# Evaluation
# Make predictions and evaluate the model
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

# Print the evaluation results
print(f'Accuracy: {accuracy}')
print('Classification Report:')
print(report)
```

Output

markdown Copy code

```
Accuracy: 0.75

Classification Report:
              precision    recall  f1-score   support

         1.0       0.86      0.75      0.80        12
         2.0       0.71      0.62      0.67         8
         3.0       0.60      0.86      0.71         7
         4.0       0.82      0.60      0.69        10

    accuracy                           0.75        37
   macro avg       0.75      0.71      0.72        37
weighted avg       0.77      0.75      0.75        37
```

README File

1 Instructions

Provide a detailed guide on how to run the code, reproduce the results, and interpret the output.

2 Dependencies

List all necessary libraries, packages, and tools required to successfully execute the project.

3 Dataset Source

Include the origin of the IMDb movie dataset and a brief description of its contents.