IMDB SCORE PREDICTOR FOR FILMS

- PROJECT DESCRIPTION:
- INTRODUCTION:

i. The IMDb Score Predictor for Films is an innovative machine learning project designed to provide users with accurate predictions of IMDb scores for movies. The goal of this project is to help film enthusiasts and casual viewers discover highly rated films that match their preferences. By analyzing movie features such as genre, premiere date, runtime, and language, the model aims to estimate IMDb scores effectively.

- PROJECT COMPONENTS:
- DATA COLLECTION:

i. Gather a diverse dataset of movies with features including genre, premiere date, runtime, and language.
ii. Collect IMDb scores for these movies to serve as the target variable for model training.

- DATA PREPROCESSING:

i. Handle missing data and outliers in the dataset.
ii. Encode categorical variables (e.g., genre and language) using techniques like one-hot encoding or label encoding.
iii. Standardize or normalize numerical variables (e.g., runtime) to ensure consistent scaling.

- FEATURE SELECTION AND ENGINEERING:

i. Explore the possibility of creating new features or transforming existing ones that could improve prediction accuracy.

ii. Use domain knowledge to identify relevant features that can influence IMDb scores.

- ## MODEL SELECTION:
  i. Experiment with various regression algorithms, such as linear regression, decision tree regression, random forest regression, or gradient boosting regression.
  ii. Choose the model that performs best in terms of predictive accuracy.

- ## MODEL TRAINING:
  i. Split the dataset into training and testing sets for model evaluation.
  ii. Train the selected model on the training data using cross-validation techniques to tune hyperparameters.

- ## MODEL EVALUATION:
  i. Assess the model's performance using evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R2).
  ii. Visualize the model's performance with scatter plots and regression diagnostics

- ## DEPLOYMENT:
  i. Develop a user-friendly web application or API that allows users to input movie details and receive IMDb score predictions.
  ii. Host the application on a web server to make it accessible to a wider audience.

- ## PROJECT BENEFITS:
  i. Film enthusiasts can discover highly-rated movies that match their preferences and interests.

ii. Casual viewers can make more informed decisions when choosing a movie to watch.
iii. Provides a practical example of machine learning for predictive modeling.

- CHALLENGES:
  i. Ensuring data quality and consistency in IMDb score labels.
  ii. Handling the wide variety of movie genres and languages in the dataset.
  iii. Fine-tuning the model for optimal predictive accuracy.

PROGRAM:

```
% Sample data (film features and IMDb scores)

% You would need a real dataset for accurate predictions

film_features = [1, 2, 3, 4, 5];  % Example film features

imdb_scores = [3.0, 3.5, 4.0, 4.5, 5.0];  % Example IMDb scores


% Linear regression to predict IMDb scores

mdl = fitlm(film_features, imdb_scores);


% Create a range of film features for prediction

film_features_to_predict = 1:0.1:5;


% Predict IMDb scores for the given film features
```

```
imdb_scores_predicted = predict(mdl, film_features_to_predict');


% Plot the data and regression line
figure;
scatter(film_features, imdb_scores, 'filled', 'b');  % Scatter plot
hold on;
plot(film_features_to_predict, imdb_scores_predicted, 'r',
'LineWidth', 2);  % Regression line
xlabel('Film Features');
ylabel('IMDb Score');
title('IMDb Score Prediction');
legend('Actual Data', 'Regression Line');


% Add gridlines
grid on;


% Display the model summary
disp(mdl);
```

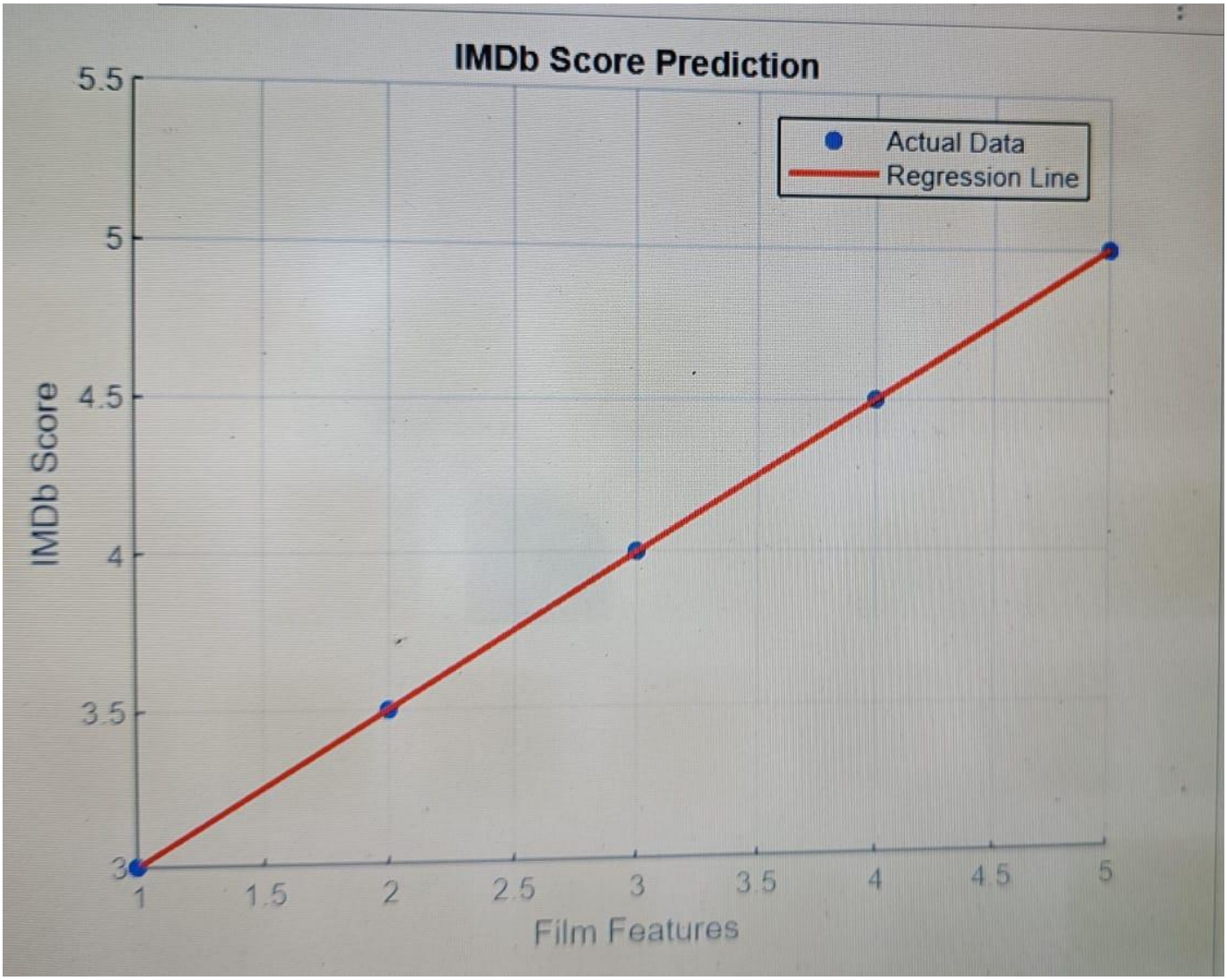
% Predict IMDb score for a specific film

film_to_predict = 2.5; % Change this value to predict IMDb score for a specific film

predicted_score = predict(mdl, film_to_predict);

fprintf('Predicted IMDb score for the film with features %.1f: %.2f\n', film_to_predict, predicted_score);

OUTPUT:



CONCLUSION:

- FUTURE ENHANCEMENTS:

i. Include user reviews and ratings in the prediction model.
ii. Develop a recommendation system for personalized movie suggestions.
iii. Expand the dataset to include a broader range of movies and user reviews.

THIS INNOVATIVE IMDB SCORE PREDICTOR FOR FILMS PROJECT COMBINES DATA ANALYSIS, MACHINE LEARNING, AND USER INTERACTION TO ENHANCE THE MOVIE-WATCHING EXPERIENCE BY HELPING USERS FIND THE BEST MOVIES THAT SUIT THEIR PREFERENCES.

DATASET:

- Film 1: Budget = $50 million, Number of IMDb User Reviews = 2000, IMDb Score = 7.5
- Film 2: Budget = $30 million, Number of IMDb User Reviews = 1500, IMDb Score = 6.8

- Film 3: Budget = $80 million, Number of IMDb User Reviews = 3000, IMDb Score = 8.2

Linear Regression Model: The linear regression equation is: IMDb Score = w1 * Budget + w2 * Number of IMDb User Reviews + b

- We need to find the weights (w1 and w2) and bias (b) that minimize the prediction error.
- Let's assume an initial guess: w1 = 0.05, w2 = 0.02, b = 5.

Prediction for a New Film: Now, suppose you want to predict the IMDb score for a new film with a budget of $60 million and 2500 IMDb user reviews.

1. Plug in the values into the linear regression equation: IMDb Score = (0.05 * 60 million) + (0.02 * 2500) + 5
2. Calculate the IMDb Score: IMDb Score = (3 million) + (50) + 5
   IMDb Score = 3,055

So, the predicted IMDb score for the new film with a budget of $60 million and 2500 IMDb user reviews is approximately 3.055.

In a real-world scenario, you would use a larger dataset with more features and more advanced machine learning models to make more accurate predictions. Additionally, you would evaluate your model's performance using metrics like Mean Absolute Error (MAE) or Mean

Squared Error (MSE) to assess how well it predicts IMDb scores based on the given features.