MARVEL MOVIES CLASSIFICATION

Objetives

- Predicting Success: Classification can help determine whether a movie will be a box office success based on various features such as budget, genre, director, and critic scores.
- Understanding Influential Factors: By identifying which features contribute most significantly to a movie's success or failure, stakeholders can make informed decisions about future projects.

Introduction

This analysis focuses on the Marvel Movies dataset, which includes information about films from the Marvel Cinematic Universe, such as titles, directors, budgets, and performance metrics like IMDb ratings and box office gross.
The primary goal is to conduct Exploratory Data Analysis (EDA) to uncover trends and patterns in the data, including visualizing rating distributions and the relationship between budgets and earnings.
Following the EDA, we will develop a classification model to predict the success of future Marvel movies based on historical data, categorizing them as "blockbuster" or "average performer."
Ultimately, this analysis aims to provide insights into the factors influencing movie performance, aiding strategic planning in the film industry.

EXPLORATORY DATA AND ANAYLISIS (EDA)

Data Info: This command prints a summary of the DataFrame, including the number of entries, column names, data types, and memory usage.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34 entries, 0 to 33
Data columns (total 14 columns):
 #   Column                                   Non-Null Count  Dtype  
---  ------                                   --------------  -----  
 0   Index                                    34 non-null     int64  
 1   Title                                    34 non-null     object 
 2   Director (1)                             34 non-null     object 
 3   Director (2)                             5 non-null      object 
 4   Release Date (DD-MM-YYYY)                34 non-null     object 
 5   IMDb (scored out of 10)                  34 non-null     float64
 6   IMDB Metascore (scored out of 100)       34 non-null     int64  
 7   Rotten Tomatoes - Critics (scored out of 100%)  34 non-null     int64  
 8   Rotten Tomatoes - Audience (scored out of 100%) 34 non-null     int64  
 9   Letterboxd (scored out of 5)             34 non-null     float64
 10  CinemaScore (grades A+ to F)             34 non-null     object 
 11  Budget (in million $)                    34 non-null     float64
 12  Domestic Gross (in million $)            34 non-null     float64
 13  Worldwide Gross (in million $)           34 non-null     float64
dtypes: float64(5), int64(4), object(5)
memory usage: 3.8+ KB
None
```

EXPLORATORY DATA AND ANAYLISIS (EDA)

Descriptive Statistics: This prints summary statistics for numerical columns, such as count, mean, standard deviation, min, and max values.

```
          Index  IMDb (scored out of 10)  IMDB Metascore (scored out of 10
count  34.000000                34.000000                              34.0000
mean   16.500000                 7.244118                              65.9705
std     9.958246                 0.692029                               8.8574
min     0.000000                 5.500000                              48.0000
25%     8.250000                 6.825000                              60.2500
50%    16.500000                 7.300000                              67.0000
75%    24.750000                 7.800000                              71.0000
max    33.000000                 8.400000                              88.0000

       Rotten Tomatoes - Critics (scored out of 100%)  \
count                                       34.000000
mean                                        80.882353
std                                         12.727362
min                                         46.000000
25%                                         76.250000
50%                                         83.500000
75%                                         91.000000
max                                         96.000000

       Rotten Tomatoes - Audience (scored out of 100%)  \
count                                        34.000000
mean                                         84.205882
std                                          10.446920
min                                          45.000000
25%                                          78.250000
50%                                          86.500000
75%                                          91.750000
max                                          98.000000

       Letterboxd (scored out of 5)  Budget (in million $)  \
count                     34.000000              34.000000
mean                       3.235294             229.523529
std                        0.519255              69.004243
```

MACHINE LEARNING MODEL

- Import Machine Learning Libraries: This imports necessary functions from scikit-learn for splitting the dataset, creating a linear regression model, and evaluating model performance.
- Data Cleaning: This line removes unnecessary columns that are not useful for modeling, such as the index, title of the movie, directors, and release date.
- Convert Categorical Variables: This converts categorical variables into dummy/indicator variables, which are necessary for modeling. The drop_first=True argument avoids multicollinearity.

```python
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score


df_cleaned = df.drop(columns=['Index', 'Title', 'Director (1)', 'Director (2)', 'Release Date (DD-MM-YYYY)'])

df_cleaned = pd.get_dummies(df_cleaned, drop_first=True)


X = df_cleaned.drop('Worldwide Gross (in million $)', axis=1)
y = df_cleaned['Worldwide Gross (in million $)']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


model = LinearRegression()
model.fit(X_train, y_train)


y_pred = model.predict(X_test)


print('Mean Squared Error:', mean_squared_error(y_test, y_pred))
print('R2 Score:', r2_score(y_test, y_pred))
```

MACHINE LEARNING MODEL

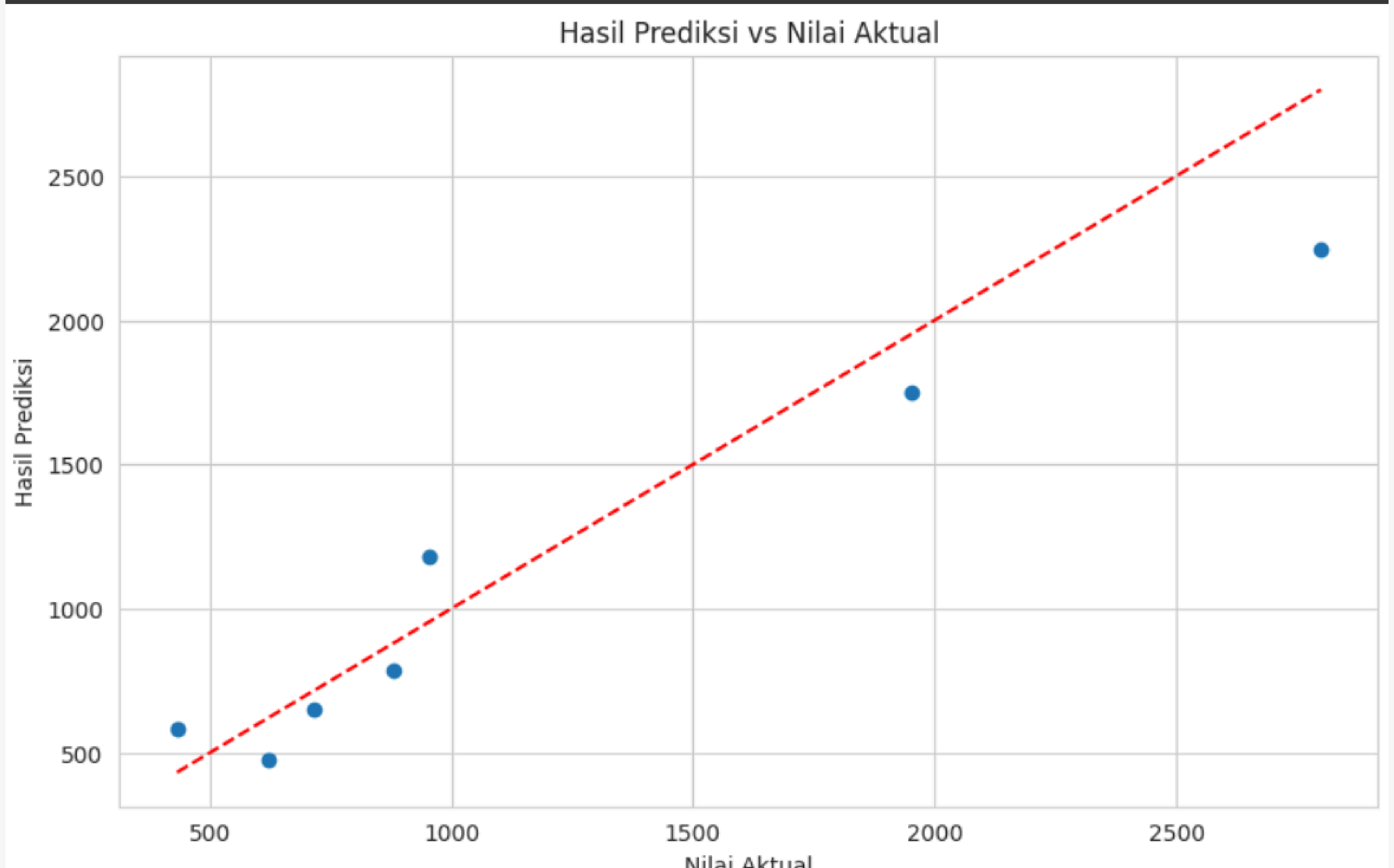
- Define Features and Target: Here, X contains all the feature columns while y is the target variable (Worldwide Gross).
- Split Dataset: This line splits the dataset into training and testing sets, with 20% of the data reserved for testing.
- Create and Train Model: This creates a linear regression model and fits it to the training data
- Make Predictions: This line uses the trained model to predict the target variable (Worldwide Gross) for the test set.
- Evaluate Model Performance: This prints the Mean Squared Error (MSE) and R^2 score to assess the model's accuracy. A lower MSE indicates better fit, and R^2 indicates the proportion of variance explained by the model.

```python
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score


df_cleaned = df.drop(columns=['Index', 'Title', 'Director (1)', 'Director (2)', 'Release Date (DD-MM-YYYY)'])

df_cleaned = pd.get_dummies(df_cleaned, drop_first=True)


X = df_cleaned.drop('Worldwide Gross (in million $)', axis=1)
y = df_cleaned['Worldwide Gross (in million $)']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


model = LinearRegression()
model.fit(X_train, y_train)


y_pred = model.predict(X_test)


print('Mean Squared Error:', mean_squared_error(y_test, y_pred))
print('R2 Score:', r2_score(y_test, y_pred))
```

DATA VISUALIZATION

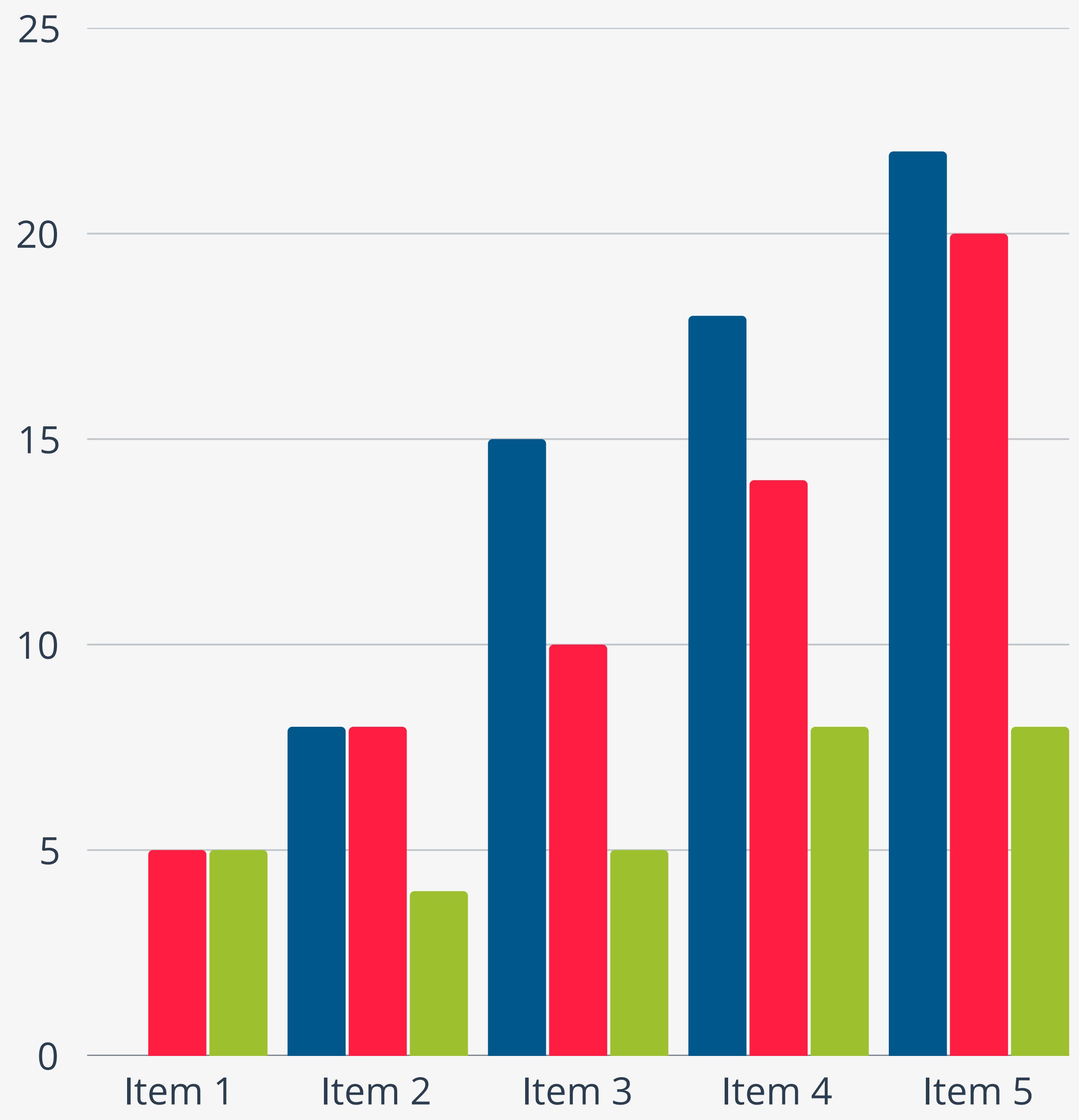
```
plt.figure(figsize=(10, 6))
plt.scatter(y_test, y_pred)
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], '--r')
plt.title('Hasil Prediksi vs Nilai Aktual')
plt.xlabel('Nilai Aktual')
plt.ylabel('Hasil Prediksi')
plt.show()
```

Predicted vs. Actual Values: This scatter plot compares the predicted values against the actual values of Worldwide Gross. The red dashed line represents the ideal scenario where predicted values match actual values perfectly.



CONCLUSION

These steps provide a comprehensive approach to analyzing the Marvel Movies dataset, building a predictive model, and visualizing results, helping derive insights from the data effectively.



CONTACT ME

FOR INQUIRIES

E-MAIL:
alifhynafis01@gmail.com

PHONE:
092265563872

Thank you!