Model reporting

The goal of this analysis is to predict the "Worldwide Gross" of movies using several machine learning models. The dataset used for this analysis is the IMDB_Movies_Dataset, which includes various features about the movies, such as their title, director, writer, cast, budget, rating, and runtime. The target variable for this prediction task is Worldwide Gross, and we aim to predict this value using four different algorithms:

- Linear Regression
- Decision Tree Regression
- Random Forest Regression
- K-Nearest Neighbors (KNN) Regression

Implementation part

The Budget and Runtime columns were cleaned, and missing values were appropriately handled. The Budget column was converted to numeric format, and missing values were filled with the mean. The Runtime column was converted into total minutes, and missing values were filled similarly.

In the dataset, categorical variables need to be converted into numeric representations to be used effectively in machine learning models. The LabelEncoder from scikit-learn was used to transform categorical columns into numerical values.

Missing values were handled by filling them with the most appropriate values based on the column type. This was done to ensure that the dataset is complete and ready for training models.

For columns with categorical data, missing values were replaced by the mode (the most frequent value). For columns with numeric data, missing values were replaced by the mean of the column.

Target Variable (y): The column 'Worldwide Gross' was chosen as the target variable since it represents the revenue that we are trying to predict using the other features in the dataset.

Feature Variables (X): All columns except the target variable 'Worldwide Gross' were selected as features. These include information like Budget, Runtime, Director, Cast, and other movie details. The dataset has been split into three parts for training, validation, and testing to evaluate model performance more effectively: 70, 15, 15.

The numerical features were standardized using StandardScaler to ensure all features have a mean of 0 and a standard deviation of 1. This scaling was applied to the training, validation, and test datasets to improve model performance and convergence. The scaler was fitted on the training data and then used to transform the validation and test data.

Linear Regression model didn't perform very well based on the evaluation metrics:

Mean Squared Error (MSE): 1,652,798.73. A high MSE indicates that the model's predictions are quite far from the actual values.

R-squared (R^2): -0.0037. This negative value suggests that the model performs worse than a simple mean-based model, which would predict the mean of the target variable for all instances. The model is not capturing any useful information from the features.

Decision Tree, Random Forest, and K-Nearest Neighbors (KNN). Here are the results of your evaluation:

Cross-Validation Results:

Decision Tree:Cross-validation scores: [0.0887, 0.0815, 0.0774, 0.0817, 0.0931]

Mean Cross-Validation Score: 0.0845

Random Forest: Cross-validation scores: [0.2146, 0.2060, 0.2106, 0.2120, 0.2049]

Mean Cross-Validation Score: 0.2096

KNN: Cross-validation scores: [0.1087, 0.1073, 0.1046, 0.1074, 0.1003]

Mean Cross-Validation Score: 0.1057

Random Forest seems to perform the best among the models in terms of cross-validation scores.

Decision Tree and KNN have lower performance, with KNN being the weakest.