Capstone Proposal - Bharat Raman

Domain Background

My project will refer to a dataset of 5000 different movies from The Movie Database (TMDb). This dataset is widely popular in kaggle as a resource for projects, such as film-reccomendations and feature-predictions. Over one thousand user-made models (kernels) can be found.
The dataset can be found here:

```
https://www.kaggle.com/tmdb/tmdb-movie-metadata#tmdb_5000_movies.csv
```

Problem Statement, Datasets/Inputs, and Solution

The problem statement for this capstone is: How well is a film’s score correlated with various features of the film. And if there is a strong correlation, how accurately can I predict the score of the film using machine learning?

Dataset/inputs

As mentioned, the dataset consists of 5,000 different movies. There are two .csv files provided

- tmdb_5000_credits.csv
- tmdb_5000_movies.csv

The credits .csv file contains the following features

- cast
- crew

The movies .csv file contains a wide range of features, including:

- movie ID
- budget
- popularity
- genre(s)
- keyword(s)
- synopsis
- vote average
- vote count

Benchmark model

The model I will be referencing: What's my Score??? by Ashwini Swain

- Swain's model utilizes the following film features to make his predictions:
    - genre

    - actors
    - directors
    - keywords

Swain’s resultant predictions are as follows

- Godfather Part III
    - Prediction: 6.95
    - Actual: 7.10
    - Accuracy: 0.979
- Minions
    - Prediction: 6.50
    - Actual: 6.40
    - Accuracy: 0.984
- Rocky Balboa
    - Prediction: 6.56
    - Actual: 6.50
    - Accuracy: 0.991

As one can see, Swain’s predictions are quite close to the actual score, with an average accuracy of about 0.985

Solution and Evaluation Metrics

Since the class label is continuous, rather than discrete, I will

implement a regression-based model to make my predictions.
After training the model, I will evaluate my accuracy using r-squared score. I will also compare my predictions with those made by Swain, in his model. Should my predictions be considerably sub-par (accuracy <= 0.6), I'll re-tune my parameters, or try other regression algorithms to attain a better score.

General Outline

My project will proceed as follows:

1. Download and pre-process data
2. Select features to use for training
3. Split data into training and testing sets (70% trainng, 30% testing)
4. Train model using regression-based algorithm
5. Make predictions and evaluate using r-squared score
6. Compare resultant predictions with benchmark model