Performance Evaluation and Interpretation

Since this is a prediction problem where we need to predict the rating of an application based on various factors (or predictors), a regression model would be an ideal approach.

We trained our data using the following models:

- Linear Regression
- KNN Regressor
- SVM Regressor
- Random Forest Regressor
- Neural Nets Predictor

Although Neural Nets was not initially included, we wanted to assess its impact on our scores and performance. It's worth noting that the implementation of the other models was submitted in previous submissions (P2 and P3). Before comparing all the models, let's delve into the process of building the Neural Nets model.

Neural Nets Model Implementation:

MLP (Multi-layer Perceptron) regression is a versatile and widely used technique for solving regression problems in NLP, healthcare industry, and retail. MLP neural net problems predict continuous numerical values based on input parameters. However, like other neural network models, MLP regression requires careful hyperparameter tuning, regularization techniques, and validation to ensure optimal performance and prevent overfitting.

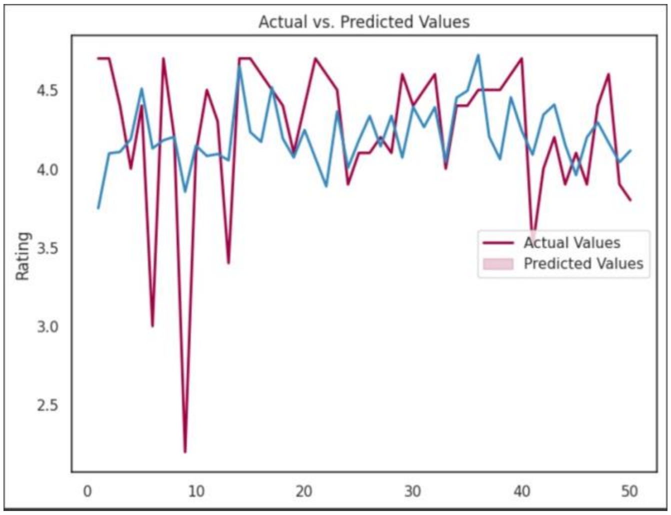
Scaling data is recommended while training the model in MLP. We ran a custom function to fine tune the hyperparameters for our data. We found that the best parameters for our data are:

- activation = tanh
- hidden_layer_sizes = (50, 100, 50)
- max_iter = 5000
- n_iter_no_change = 200

Scores:

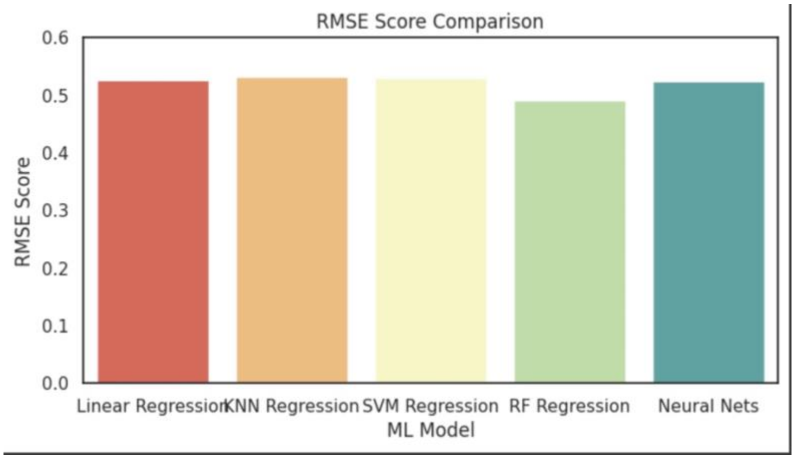
- RMSE – 0.52
- SSE – 0.27

Actual vs Predicted values graph:



Overall Performance Comparison between all models:

Model	RMSE	MSE
Linear Regressor	0.53	0.28
KNN Regressor	0.53	0.28
SVM Regressor	0.53	0.28
Random Forest Regressor	0.49	0.24
Neural Networks MLP	0.52	0.27



Interpretation:

The objective of this project was to assess the rating of applications on the Google Play Store based on various predictors such as Category, Price, Content Rating, etc. Identifying the factors that influence app ratings is crucial for app developers to enhance the incentives for highly rated apps and avoid penalties associated with low-rated apps. This knowledge can help developers focus on improving user engagement and ratings. Among the models evaluated, it was found that the ***Random Forest Regressor*** exhibited the lowest RMSE (Root Mean Squared Error) compared to other models, indicating its superior performance. Therefore, the Random Forest Regressor is recommended as the ideal model for detecting the rating of applications on the Google Play Store.

Future Scope:

Random Forest has been identified as the ideal model for this prediction task due to its low RMSE value. However, there is still scope for further improving the model's performance by experimenting with the inclusion or exclusion of certain predictors, based on their relevance and correlation with the output variable. Additionally, tweaking parameters such as maximum depth, number of estimators, etc., could also potentially enhance the model's efficiency and accuracy in predicting the ratings of applications.