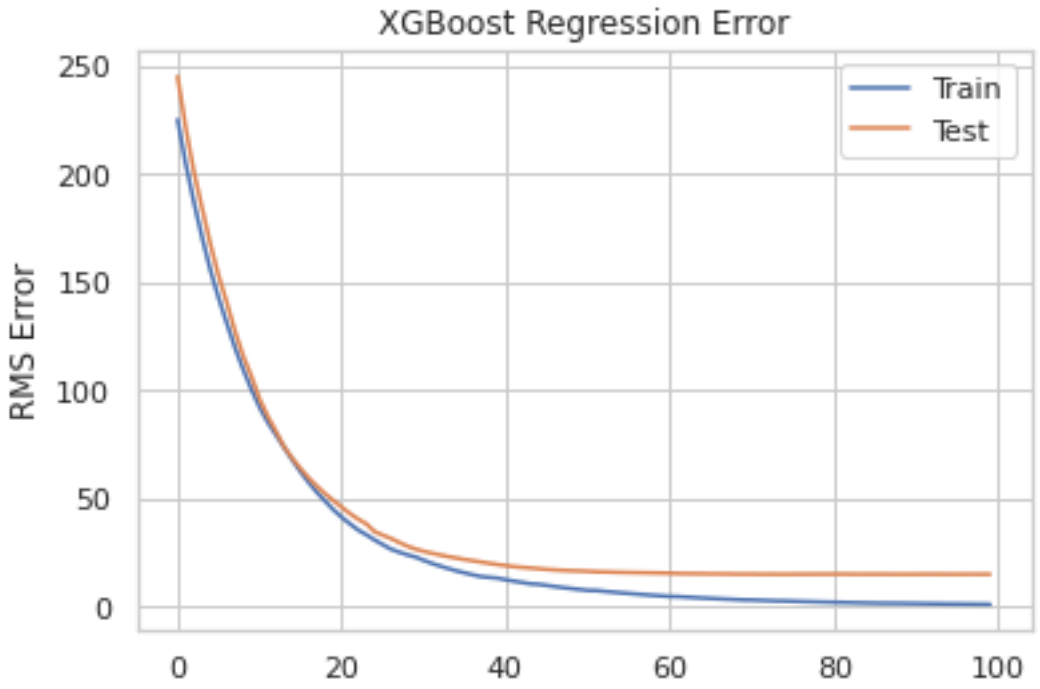
Solution Sheet

1. Which model have you used for Total IPL 2020 Runs prediction for each player? Explain your model.

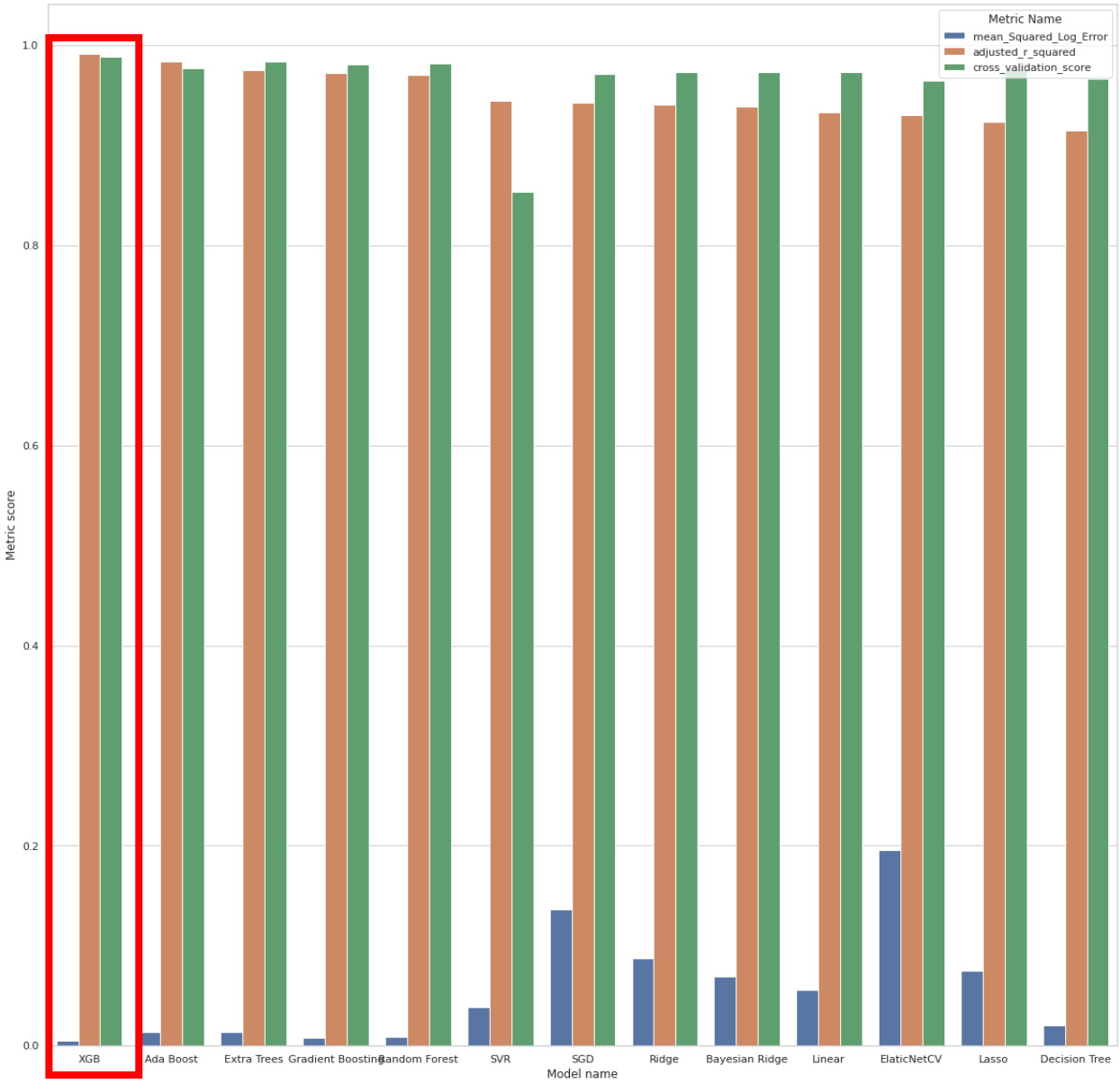
Ans: The problem statement of predicting IPL 2020 runs was a regression problem. The model which I have used to predict the total runs scored is XGBoost Regression Model. XGBoost is an advanced implementation of gradient boosting built for the purpose of improving the model performance and computational speed of boosting tree algorithms. It is used in Machine Learning today both for regression and classification. XGBoost stands for "Extreme Gradient Boosting", where the term "Gradient Boosting" originates from the paper *Greedy Function Approximation: A Gradient Boosting Machine*, by Friedman [1]. This model performed better as compared to the other models in terms of regression metrics.

The metrics used here are: Explained Variance Score, Mean Absolute Error, Mean Squared Error, Root Mean Squared Error (rmse), Mean Squared Log Error (msle), r-squared Score, Adjusted r squared Score and the cross validation score obtained from the Repeated K-Fold cross validation. The model of choice showed the highest value of adjusted r square score (approx. 0.991765) and the mean cross-

validation score (.99 +/- 0.2) and the lowest value of errors (rmse being approx. 8.93 and msle being approx. 0.0047) among others listed above.



The above plot shows the decrease of RMSE in XGBoost Regression over the various epochs.



This plot shows that XGB has the lowest value of errors (blue bar) and the highest value of adjusted r square score (brown bar) pointed out by the red box.

Among the other regression models build here for comparing the results are: Extra Trees Regression model, Random Forest, Gradient Boosting, Ada Boost which used Decision Trees as the base model, Support Vector Regressor, Stochastic Gradient Descent, Ridge Regression,

Bayesian Ridge Regression, Linear Regression, Elastic Net CV, Decision Tree, Lasso Regression and finally Artificial Neural Network (ANN). The order given here is sorted according to the decreasing values of adjusted r square score (though ANN was not compared here).

At first hypothesis for the problem was formed. Then the data pre-processing step was done. In the data exploration phase, data was checked for any null values. There were no null values present in the data explicitly. Data processing was done. Then it was separated into features and labels (X and y respectively).

After that feature scaling was done using standardization and normalization (later proceeded with the former because the data followed gaussian distribution). Then line, bar, density and pair plot were plotted. From the box plot it was observed that the data did not had many outliers. So, it was ignored because not much data was available to train the model. Then the train data was segregated into train and validation set (80:20 ratio split).

Thereafter the model building part was initiated. At first started with the simplest model of Linear Regression and then proceeded towards complex model of ANN. At first the models were trained using the training set and it was validated using the validation set. The results were compared with the metrics described above. Feature selection and hyperparameter tuning did not show much

improvement in performance of the model. Repeated K-fold cross validation was performed to check for any over-fitting in the data. The best result was given by XGBoost so the final prediction of IPL 2020 total runs using the test set was done using this model. Finally, the output file (Data_final_answer_flipr.xlsx) was generated.

All the code can be found in Flipr Hackathon 7.0 - Machine Learning Task - 16.10.2020.ipynb file.

Reference:

1. XGBoost Documentation