**REPORT FOR TASK 3**

**Introduction**

The dataset used in this task, titled "nba\_rookie\_data.csv," was obtained from the assignment task on Canvas. This dataset entails in it a comprehensive collection of information about 1340 NBA stars 21 distinct columns among which are the number of Games Played, Minutes Played, Points Per Game, Rebounds etc, all of which are serving as a valuable attribute for analysis.

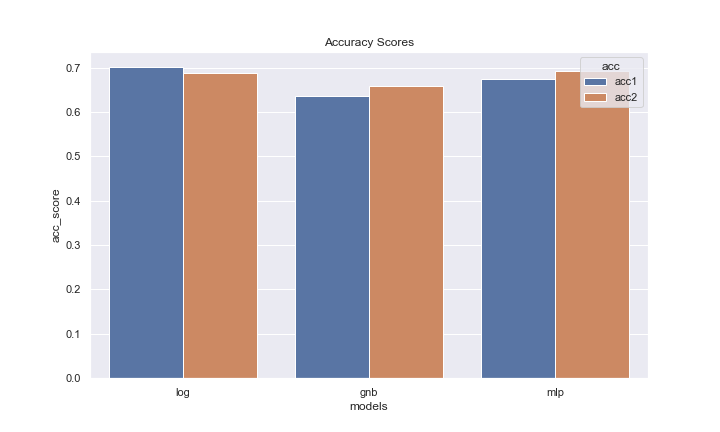
I began task 3 by first importing our necessary libraries among which are:

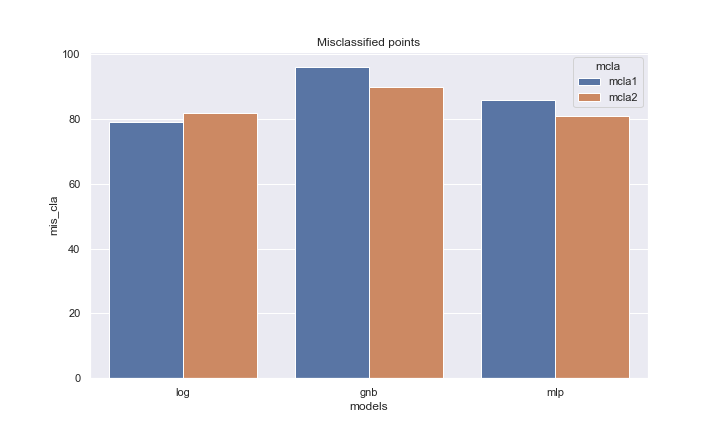
* Pandas for preprocessing, cleaning and loading of the data.
* Seaborn and Matplotlib were both used for the visualization.
* Sklearn modules in which we extracted the:
  + train\_test\_split to split the data into train and test data,
  + accuracy score for metrics, randomized CV for hyper tuning the data, and
  + LogisticRegression and GaussianNB were both used as the classsificiation model while MLPClassifier used to create neural networks. They all served as the selected algorithms used for the task.

I then moved on to the loading of the dataset into a data frame, then I also inspected the data and cleaned it. In inspecting, I found no null values but there were duplicates which was dealt with properly. I also check the summarized info of the data, statistical information as well as the correlation of each feature to another.

I delved into the machine learning part where I separated my features from the target “Target\_5Yrs”.

In trying to include as many features as I can I decide to use a method of iterating through all the models and features. I used all of the features to fit and predict and as the iteration proceeds, I ensured I kept track of the accuracy score and the misclassified point generated for each of the algorithms. I then Iterated again, this time around selecting only specific features, which from the correlation had relationship with the target variables. These features were Games Played, Minutes Played, Points Per Game, Rebounds I also kept track of the accuracy score and the misclassified points just as done earlier. Now I needed to visualize these two distinct metrics generated, then I went ahead an used the visualization modules i.e seaborn and matplotlib



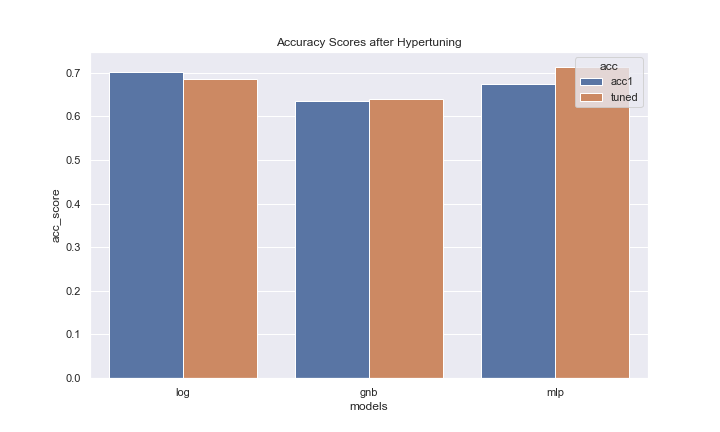


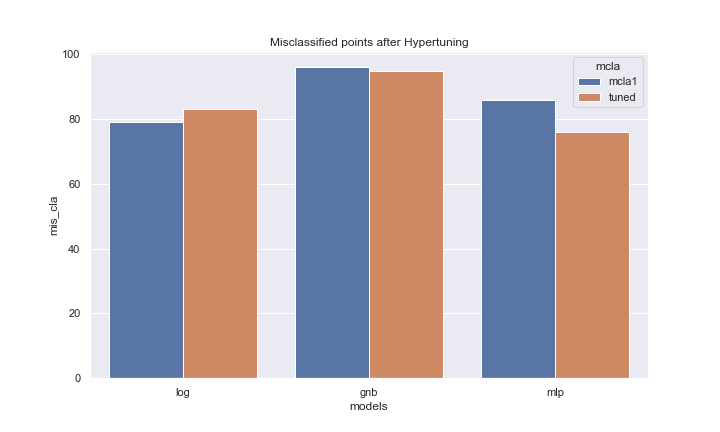
Tweaking the features by manipulating in terms of accuracy score, it is evident from the plot that the log reg performs better in the first instance than the others even when other accuracy scores have a high accuracy score in the second instance, the accuracy score of the log reg at the first instance supersedes others.

In the misclassification plots, which indicates the number of misclassified points made by our model, the log reg at the first instance is still the smallest with misclassifications of 79 which seems to be the best, I stuck with using the first instance, i.e all the features.

In making further improvements to the first instance, I used the randomized cross validation on the three models to hyper tune them by supplying their respective parameters to them.

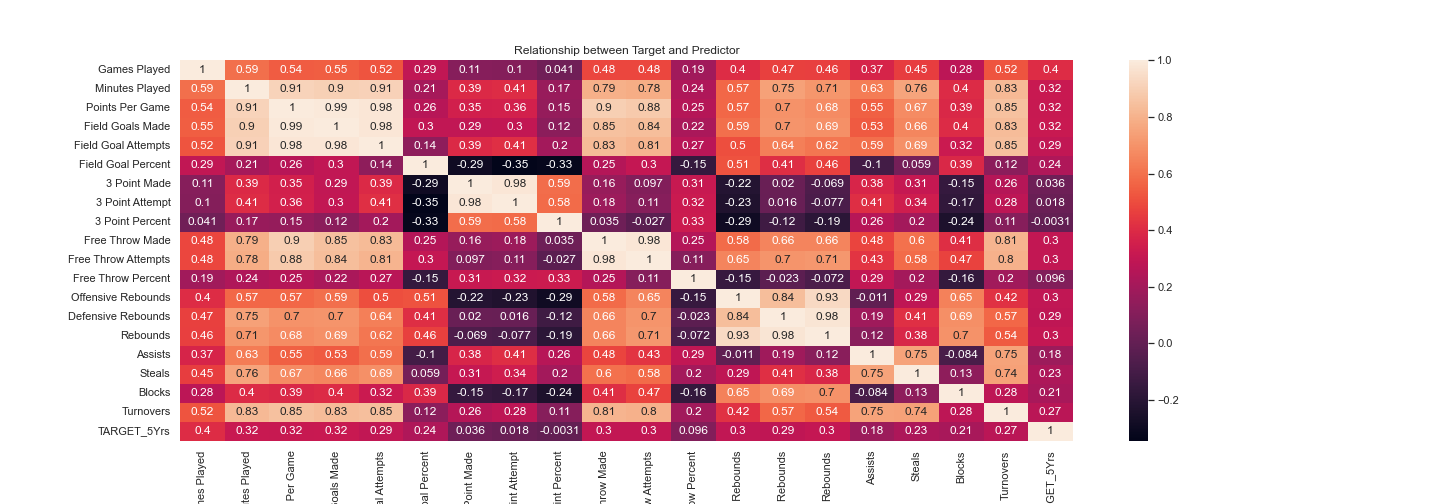
After which I fit them with the best params that was generate and made predictions again but this time and also kept track of their accuracy score and misclassified points that was generated.





Trying to improve our models worked we made use of Randomized cross validation which made positive impact on the gnb and mlp models, with a reduced misclassfied points and increase in its accuracy score. The Logisitic Regression worsen when we try to improve it, so we will stick with its initial instance.

In viewing the relationship between the target variable and the features we generated the correlation plot



From the plot, not all features are positively or negatively correlated to the Target variable, but there is a moderate relationship between the predictor and target variable.

In conclusion, all models were better when all the features where used. The models GaussianNB and MLPClassifier performed best when hypertuned but the Logistic Regression worsened when it was hypertuned.