University of Technology Sydney

36120 Advanced Machine Learning Applications

AT1 - NBA Draft

Archit Pradip Murgudkar
Student Id - 14190286

Business Objective:

The NBA draft, an annual event where teams pick players studying in American colleges and from international professional leagues to join the rosters. These picked players based on their statistics will be selected to play in the NBA league. The primary objective of the project is to build a binary classification model which in turn when trained will help predict whether a particular player based on his numbers will be drafted to get into the NBA.

The model built should first be used to train on the known data after which its performance can be tested on the validation and testing, i.e. on the unseen data. The models can be tuned on different hyperparameters based on the need. The metric we will use to assess the performance of the model is the roc_auc_score, which basically computes the area under the receiver operating characteristic curve from prediction scores.

An optimum model produced will provide accurate predictions of the possibility of the college player getting chosen for the NBA. The first stage in the project is data understanding, where we will examine the acquired data, describe the data, perform exploratory data analysis which plays a major role in making sure that data available is understood very well. Analysing the quality of data. The next phase comprises data preparation which is carried out before modelling machine learning algorithms. This phase may consist of cleaning raw data, executing feature engineering, feature selection, data transformations wherein all the features should strictly be in the numeric format before feeding them to the model. Modelling is the next phase wherein we split data into categories if required and build machine learning algorithms. The final stage of the project is analysis of built machine learning models, testing with what accuracy the algorithm makes correct classifications.

Data Understanding:

The dataset comprises 63 different independent features and one single target variable named ‘drafted’. Loading this information in the dataframe using Pandas library. Gaining basic information of features like column names, their data types and total count of all these features. In the dataset, the features are divided into three types of data types : ‘int64’, ‘float64’ and ‘object’, wherein 49 features are float, 8 are of int and 7 of object data type.

Data Preparation:

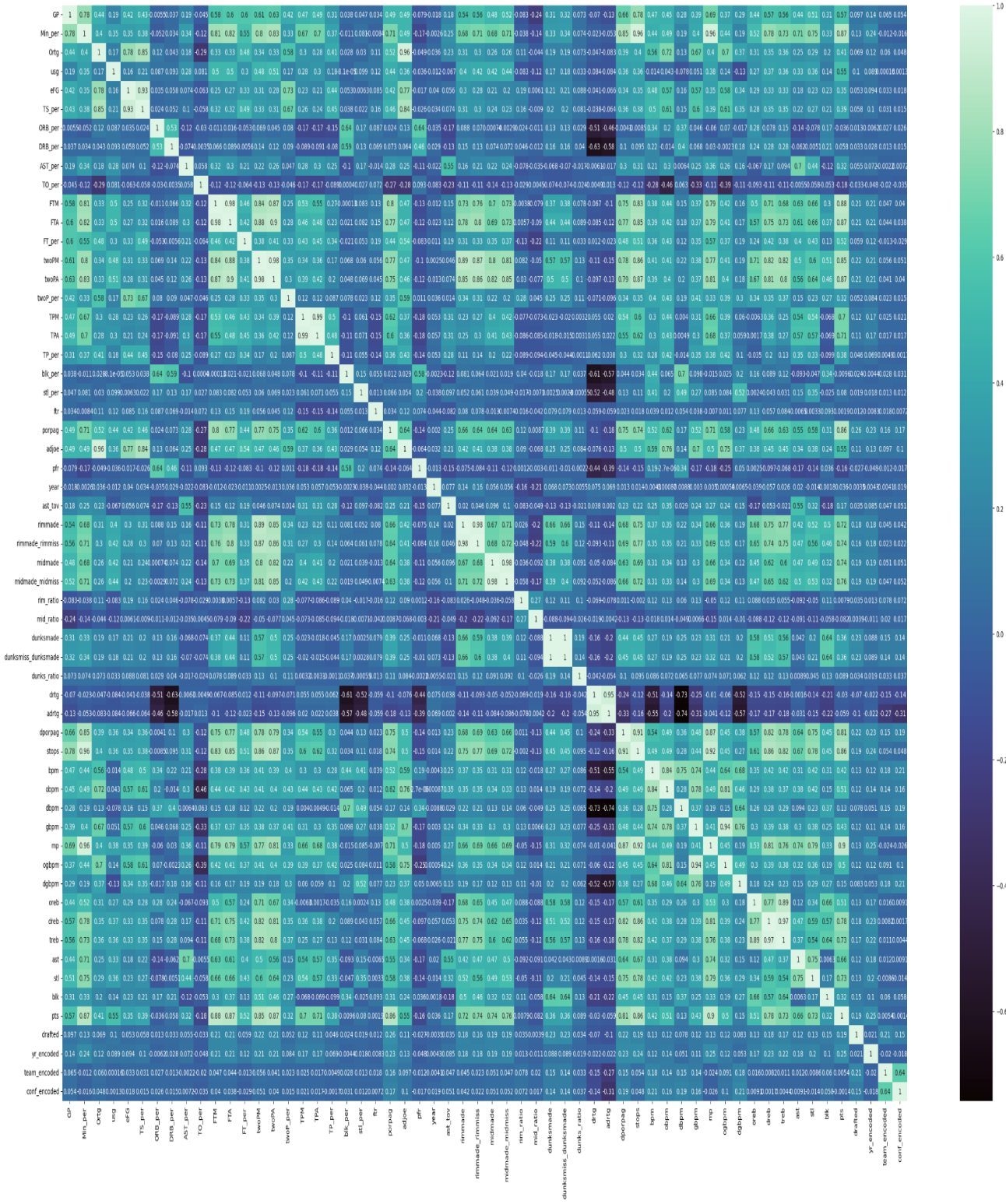
In this phase we need to make the necessary transformations to data, clean and process it before feeding it to the model. I made a copy of the dataset to perform data cleaning, so that we always have our original dataset if we want to refer and make the changes in the duplicate dataset. Further, I did a check on finding null/missing values in the records.The feature ‘rec_rank’ and ‘pick’ comprised a greater amount of null values, so I decided to totally drop off those columns. Adding any value like mean/median/0 etc. would create extreme bias in the predictions. After this, I deleted the following features due to valid reasons as follows:

1. num" feature as it is a player specific record
2. "ht" feature as it consists false information
3. "player_id" as it consists specific information

4. "type" as it consists of the same value in all records and would not help in making any decision.

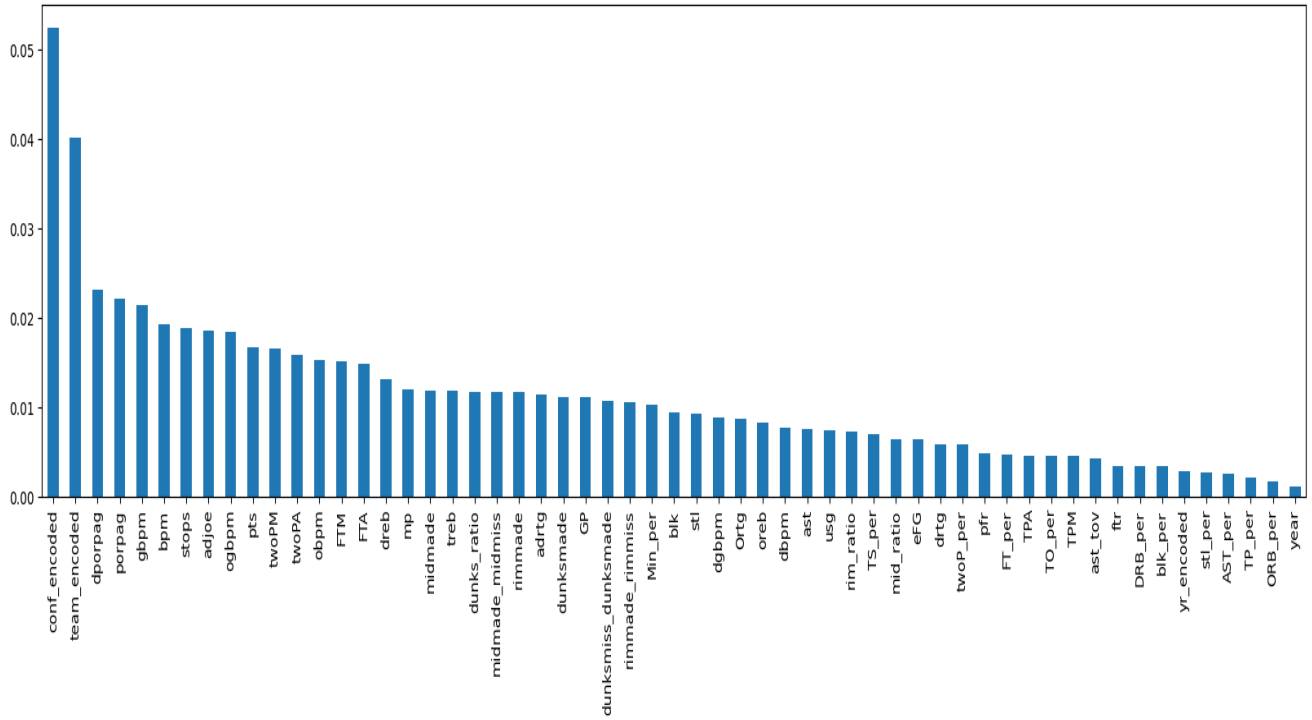
Further, in the missing values of other features, I imputed median, as median is robust on outliers. “Yr” had redundant values and nulls which I replaced with the mode of it. I also checked for duplicates and there were no duplicates present in the dataset.

I checked how all features are correlated with the target variable and also plotted a heatmap.



From that, I understood that the features did not strongly correlate much with the target feature, “porpag” with the correlation of 0.255was the highest among all, and “mid_ratio” ranked lowest.

I also checked the mutual information of dependent variables and target variables. Mutual information tells us about how strongly independent variables are related to the independent variable.



As our machine learning classifier requires data to be in the numeric format before feeding it, I decided to perform encoding on the features with object data type. Due to high cardinality in the “team” and “conf” variables, I performed encoding using leave one out encoding technique. I performed ordinal encoding on the “yr” feature and converted them into numeric data types. Another important factor is ensuring that we provide data to the model which is on the same scale, which will help our model make accurate predictions. For this, I performed data scaling by importing the ‘StandardScaler’ library from sklearn. After accomplishing all this, the data was finally suitable to be fed to the classifier algorithm and carry out modelling.

Modelling and Evaluation:

The first step in modelling the classifier model is dividing the data into training, validation, and test sets. I choose to divide the records in a 8:2 ratio, or 20% of the total data for the validation set and the remaining 70% of the data will then be used for the training set, used to train the algorithm. The dataset was highly imbalanced with just around 9 % output to be 1 and 99% to be 0. To tackle this, I made use of the SMOTE technique which generated 90% of the synthetic data of minority class present in the output variable. This helped to balance

the dataset. I then evaluated the baseline performance and imported accuracy measures from sklearn, including accuracy_score and roc_auc_score. In the experiments I performed, I have made use of classification algorithms namely, **Polynomial Logistic Regression, Random Forest and XgBoost**. I have developed these algorithms and tuned them with different hyperparameters and tested all on the unseen data (validation set). From this, I picked the best performing models in each category and tested on the test dataset.

Using the Random Forest model on the unseen data, i.e, the testing set, the score of the performance metric, roc_auc_score came out to be 0.7606 . So the predictions made by the model to classify if a player will get drafted is 76.06 %accurate.

I then modelled the XGBoost classifier to make predictions. I made a list of different hyperparameters namely 'n_estimators','max_depth', 'gamma', 'reg_alpha', 'colsample_bytree', 'min_child_weight', 'learning_rate' and performed a randomised search using RandomizedSearchCV on the XGBoost classifier. The best parameters obtained were as follows:

- 'reg_alpha' - 0.5
- ' n_estimators' - 100
- 'Min_child_weight' - 1
- ' max_depth' - 5
- ' leaninig_rate' - 0.3
- ' gamma' - 0.0
- ' colsample_bytree' - 0.3

I performed cross_val_score on the classifier and got a score of 0.9993. Then I fitted xg_classifier on the training set. The accuracy score using the 'roc_auc_score' metric came out to be 1 and on the validation set, it came out to be 0.9986. Using the XGBoost model on the unseen data, i.e, the testing set, the score of the performance metric, roc_auc_score came out to be 0.8917, which is a case of overfitting.

For logistic regression, I decided to perform a polynomial transformation on the original data with the order 2. I made a list of different hyperparameters namely 'penalty', 'max_iter', 'solver', 'C' and performed a grid search using GridSearchCV on the Logistic Regression classifier. The best parameters obtained were as follows:

- 'penalty' - l2
- 'max_iter' - 300
- 'solver' - saga
- 'max_iter' - 300

I fitted the logistic regression model on the training set. The accuracy score using the 'roc_auc_score' metric came out to be 0.9937 and on the validation set it came out to be 0.9886. Using the polynomial Logistic Regression classifier on the unseen data, i.e, the testing set, the score of the performance metric, roc_auc_score came out to be 0.97919. So the predictions made by the model to classify if a player will get drafted is 97.919% accurate.

Following are the models and accuracy score achieved when examined on the test data:

Model	roc_auc_score
Random Forest	0.7606
XgBoost	0.8917
Logistic Regression	0.97919

Recommendations:

Major issue I thought of was the features being weakly correlated with the target feature, the high imbalance present in the distribution of the target variable and high cardinality in the features. Also, alot of null and redundant values were present in the variables of the dataset.

There is a high need to collect more records which would contribute in balancing the dependent variable. Also, more features need to be collected which may have a stronger relation with the target variable which in turn will help the model predict with higher accuracy based on stronger evidence.