EXPERIMENT REPORT

Student Name	Daniel Alexander
Project Name	Assignment_1_Part_B
Date	25/08/2023
Deliverables	Experiment_B.ipynb Random Forest Classifier ROC Score Github Repository: https://github.com/daniel-alexandr/adv_ml_application_assignment_1

1. EXPERIMENT BACKGROUND	
Provide information about the problem/project such as the scope, the overall objective, expectations. Lay down the goal of this experiment and what are the insights, answers you want to gain or level of performance you are expecting to reach.	
1.a. Business Objective	This project is aimed to help the company as one of the leaders in the sporting apparel industry to maximize potential investment return on young players. The result of this project will be used for determining potential sports models to be invested early and cheap before the young players are even drafted for the NBA. Inaccurate results will impact the scale of financial loss, for example offering contracts to players that is not drafted will not generate profit for the business.
1.b. Hypothesis	We can reduce the overfitting issue from the model in experiment A, while still achieving an accurate model by tweaking the hyperparameter on random forest classifier algorithm.
1.c. Experiment Objective	The objective of this experiment is to build a reliable predictive model that utilizes the statistics of college basketball players' current season performance to assess their likelihood of being selected in the NBA draft. If the experiment proves our hypothesis is correct, then the model should be pushed into production which then will be used as part of business as usual to help the player acquisition department to invest in the right people. If the model used in this experiment does not satisfy, then we would either pursue more statistics (features) of the players, or try other potential models.

If the results are way off, then we might also consider terminating the project.

2. EXPERIMENT DETAILS

Elaborate on the approach taken for this experiment. List the different steps/techniques used and explain the rationale for choosing them.

2.a. Data Preparation

1. Removing the missing observations from these following columns. From the EDA, we suspect those features play a significant role in determining the likelihood of being drafted, while the number of missing observations are insignificantly small. Therefore, we keep these features and remove missing observations.

   - Rimmade
   - Drtg
   - Mp

2. Performing resampling, specifically under sampling to mitigate the imbalance dataset. The minority class represented are less than 2% of the raw dataset. We resample the dataset with a 70:30 ratio, hoping that by capturing the scarcity of the minority will produce a better performing model.

3. Since the test data are separated from the training data, it requires different treatments for the missing values. We do not want to delete any of the missing observations on the test data.

Potential approaches to be used in future experiments are the resampling techniques, either changing the ratio or the methodologies, which might increase the performance of our model.

2.b. Feature Engineering

1. Removing these features as our EDA indicates that it is insignificant towards our objective, also to avoid overfitting caused by unique features such as ID

   - Team
   - Yr
   - Type
   - Num
   - Ftr (Definition missing. Need definition, before using this feature)
   - Pfr (Definition missing. Need definition, before using this feature)
   - year
   - player_id
   - ht (Values are absurd, need clarification)
   - pick (Will make the model only consider those who were given the draft order)

2. Fill in missing observations with 0 for these columns because it’s appropriate or to capture the missingness or inapplicableness.

- dunks_ratio || *caused by no succesful dunks were made*
- rim_ratio || *caused by no succesful rimmshot were made*
- mid_ratio || *caused by no midmade_midmiss were made*
- Rec_Rank || *to capture the state of unranked*

3. Fill in the missing value with its average, because these features are not supposed to be missing, significant for our purpose, and is appropriate to be filled with its average.

- ast_tov

4. Scaling the numerical features using standard scaler arguably to avoid scale sensitivity issues for some algorithms.

5. Applying one hot encoding for conf feature as the only categorical feature that we are using in this experiment.

Below are the data engineering treatments which we applied to the test data so it is compatible with the trained model:

1. Remove these following columns to match what were removed from the training data:

- Team
- Yr
- Type
- Num
- Ftr (Definition missing. Need definition, before using this feature)
- Pfr (Definition missing. Need definition, before using this feature)
- year
- player_id
- ht (Values are absurd, need clarification)
- pick (Will make the model only consider those who were given the draft order)

2. Since we do not want to remove any observations from our test set, therefore, we fill the missing observations for these following columns with their average value, because there are only 1 missing observation for each of these columns and filling them with the average value is insignificant.

- Drtg
- Adrtg
- Dporpag
- Stops
- Bpm
- Obpm
- Dbpm
- Gbpm
- Ogbpm
- Dgbpm
- ast_tov

	3. Filling missing observations for these columns with 0 because it is logical to use that value considering that the dependent columns (ratio’s relating to these columns) support our logic as no shots were made. - Rimmade - Rimmade_rimmiss - Midmade - Midmade_midmiss - Dunksmade - Dunksmiss_dunksmade - Rim_ratio - Dunks_ratio - Mid_ratio - Rec_Rank Potential approaches for future experiments, process feature ht as logically it’s a significant factor for our purpose. Get clarification on the definition of ftr and pfr in case we could use it to make our model better performing.
2.c. Modelling	In this experiment we are using a random forest classifier with hyperparameter: (n_estimators=700, max_depth=5, min_samples_split=50) The model selection used in this experiment is a result from our hypothesis as an improvement from experiment A. We want to reduce the overfitting issue introduced in experiment A by finding the best hyperparameter combination to produce an accurate model without overfitting. Other algorithms were not considered for this experiment because in experiment A, a random forest classifier was able to achieve a perfect recall score for the training set, however, it is overfitting. Therefore, the next best step would be to tweak the parameters on the random forest classifier algorithm to reduce overfitting. For our future experiment we might try AdaBoost, polynomial log regression and compare how they perform against the random forest classifier.

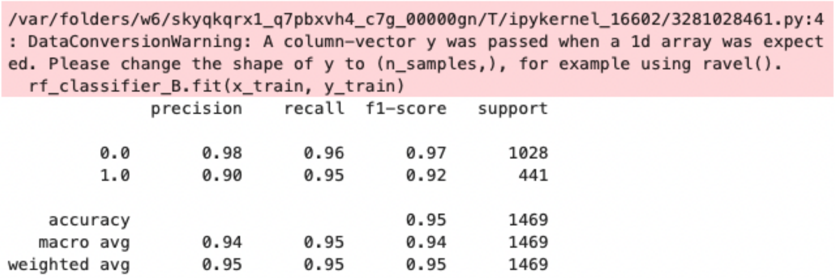
6. EXPERIMENT RESULTS

Analyse in detail the results achieved from this experiment from a technical and business perspective. Not only report performance metrics results but also any interpretation on model features, incorrect results, risks identified.

3.a. Technical Performance

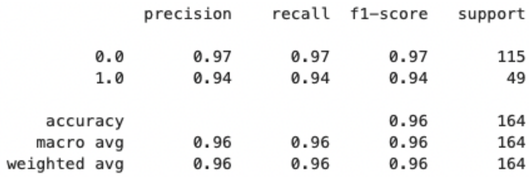
Naïve Model’s Recall score for positive class = 0

Experiment_B train dataset recall score= 0.95



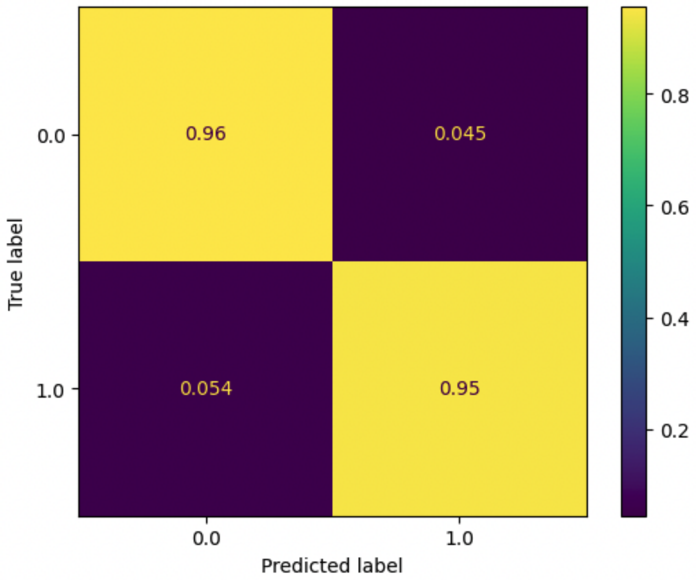
```
/var/folders/w6/skyqkqrx1_q7pbxvh4_c7g_00000gn/T/ipykernel_16602/3281028461.py:4
: DataConversionWarning: A column-vector y was passed when a 1d array was expect
ed. Please change the shape of y to (n_samples,), for example using ravel().
  rf_classifier_B.fit(x_train, y_train)
              precision    recall  f1-score   support

         0.0       0.98      0.96      0.97      1028
         1.0       0.90      0.95      0.92       441

    accuracy                           0.95      1469
   macro avg       0.94      0.95      0.94      1469
weighted avg       0.95      0.95      0.95      1469
```

Experiment_B validation dataset recall score= 0.94

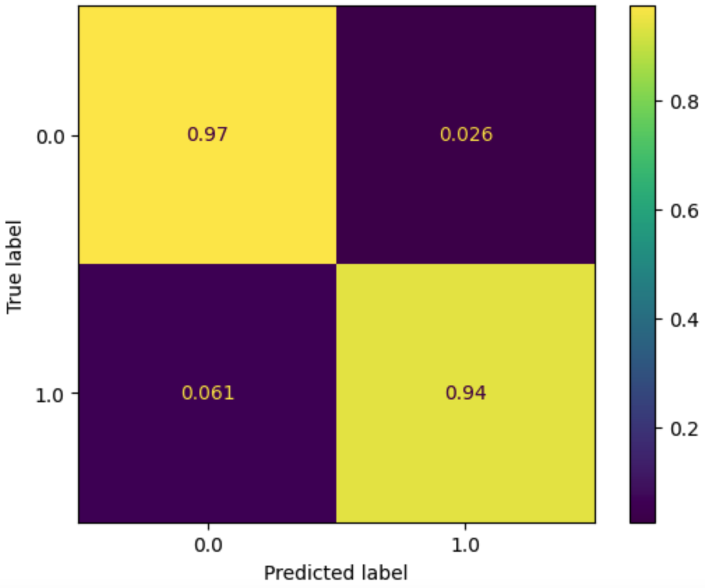


```
              precision    recall  f1-score   support

         0.0       0.97      0.97      0.97       115
         1.0       0.94      0.94      0.94        49

    accuracy                           0.96       164
   macro avg       0.96      0.96      0.96       164
weighted avg       0.96      0.96      0.96       164
```

Confusion Matrix for training (under sampled)



Confusion Matrix for validation (under sampled)



From the results, it looks like that we were able to reduce the overfitting, as the performance for training and validation are adequately consistent. Just by judging from these results, this model might be suitable to be pushed for production.

3.b. Business Impact	If this model is used for production, then supposedly we should be able to predict future NBA drafting with at > 90% chance of getting our prediction accurately. Ultimately, we should be able to increase our profit by investing early and cheaply on the potential young NBA players.
3.c. Encountered Issues	1. Missing data definition for 2 features which need to be clarified. Therefore, we removed them from our experiment. 2. Bad data quality in ht and the presence of missing data. We solve the issue by disregarding ht from our experiment and working around with the missing observations either with data engineering or removing the observations. 3. Imbalanced dataset, which causes more complications and needs extra effort to take care of the imbalanced dataset. 4. The test data came separated from the training data which require us to do different data preparation and engineering, also extra care should be applied to ensure that the test data set has the exact same features, format, and orders as the processed training dataset, i.e. we had to add dummy columns just to match the training, and reorder the features.

7. FUTURE EXPERIMENT

Reflect on the experiment and highlight the key information/insights you gained from it that are valuable for the overall project objectives from a technical and business perspective.

4.a. Key Learning	This model should perform well enough for our project objectives, at the least this model should be the benchmark for our future experiments. If other models could not give us better performance, then this model should be pushed for production. A final check should be done with the unseen data provided in kaggle competition and check the consistency of the performance from this model.
4.b. Suggestions / Recommendations	Potential next step: 1. Putting the model into test on unseen data on Kaggle, and if the performance is consistent, then push to production. 2. Compare different algorithms such as AdaBoost or polynomial LogRegression