EXPERIMENT REPORT

Student Name	Mark Brackenrig (12964298)
Project Name	Assignment 1A
Date	07/02/2021
Deliverables	notebooks/brackenrig_mark-12964298-week1_ensemble.ipynb models/brackenrig_mark_12964298_week1_votingclassifier.sav github: https://github.com/roger-yu-ds/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	The goal of this project is to predict whether an NBA 'rookie' will have a career that spans at least 5 years. A rookie is an NBA player who is in the first year of their career. This model has numerous applications – for example, a collectibles investor could use this prediction to inform purchases of 'rookie' basketball cards while they are still cheap.
1.b. Hypothesis	Using an ensemble model of different model variations will improve upon a 'base' random forest model. Currently, the team's two best performing models are a random forest and a linear regression model. Theoretically, averaging the results of the two models could improve the predictions.
1.c. Experiment Objective	I expect that using an ensemble method with different models will improve the results. Different models typically perform better in different circumstances. For example, a random forest will outperform a linear model in scenarios where the relationship between the independent variables and dependent variable are non-linear.

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	A shared function `download_data` was used to retrieve data from the Kaggle API, unzips the files and saves them into the raw data directory. This data contains two CSV files – 'train' and 'test'. The training dataset is cleaned by removing the ID columns 'Id' and 'Id_old'. The test dataset is split into 'X_test' (without the id column) and 'test_id' – a pandas series of the Ids. Next, the training set is split into the independent and dependent variables using the shared function `separate_target`. Lastly, the training data is then split into the training and validation datasets using the 'train_test_split' function. I elected to use a test_size of 20%. This was an arbitrary value as this aspect was not the main focus of this experiment. There was no notable data cleaning performed on the dataset as it was a generally clean dataset with no missing values, or obvious erroneous entries.
2.b. Feature Engineering	Since this model is using the `VotingClassifier` ensemble method, different feature engineering methods were used for the two input classifiers. The input classifiers were structured as sklearn pipelines to allow for different feature inputs to the classifiers. Both classifiers used the `StandardScaler` - which performs normal standardization: $$z = \frac{x - \bar{x}}{s}$$ The logistic regression model uses Principal Components Analysis to convert the independent variables to Principal Components. I did this to convert the independent variables to orthogonal variables.

2.c. Modelling	The model uses a voting classifier with a random forest classifier and a logistic regression classifier as inputs. The voting classifier uses the 'soft' voting method which classifies the result based on the highest average probability of the underlying classifiers. The model was tuned on the voting weights, the max_depth and max_features of the random forest classifier. Other hyperparameter settings were chosen from previous experiments on individual models. To tune the model, randomised search was used to search the feature space as grid search would be computationally expensive. 12 hyperparameter sets were used, as well as a 7-fold cross validation.

---

3. 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	AUC (Validation): 0.719 AUC (Test – Kaggle): 0.709 Precision (Negative Class / Positive Class): 0.5/0.84 Recall (Negative Class / Positive Class): 0.07/0.99 F1 (Negative Class / Positive Class): 0.12/0.91
3.b. Business Impact	The model is reasonably poor at predicting the minority class (that the rookie does not have a 5 year career) as the F1 score is only 0.12. This suggests that the model would have limited utility in being able to use this model for investment/prediction purposes. However, the model has reasonable AUC, suggesting that some pattern has been identified. This could be used by NBA teams to identify 'at risk' players.
3.c. Encountered Issues	The main issue encountered in modelling was the unbalanced classes. The minority class accounts for approximately 16% of the test set. The models typically biased the majority class.

4. 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	By using the ensemble method, you can improve on the results of individual models marginally. This can be leveraged in future experiments by incorporating new models into the 'VotingClassifier'. Also, not dealing with the unbalanced classes is having a negative effect on the strength of the model. Overfitting was not an issue with this model. This could potentially indicate that the model has sacrificed too much predictive power to avoid overfitting. Adding more complexity into the modelling could improve results.
4.b. Suggestions / Recommendations	Another team member used SMOTE, an oversampling method to reweight the sample. This proved to be successful in providing our best Kaggle score. As my model performed poorly on the minority class, incorporating this into my model could improve my results. Further to this, using calibration, such as 'CalibratedClassifierCV' could further improve the model trained on the reweighted sample, which was not done on the teams best performing model. As the voting classifier method was successful, experimenting with other models to include in the ensemble could further improve results. In particular, using models that are different from logistic regression and random forests (i.e. do not require linearity and are not based on decision trees) is an area of interest. Using models such as SVM and KNN could pick up on information not identified in the current model.