1B EXPERIMENT REPORT

Student Name	AJ Duncanson 12823819
Project Name	[UTS AdvDSI] NBA Career Prediction
Date	14 February 2021
Deliverables	/notebooks/ Duncanson_aj-12823819-week2_xgb02g.ipynb /models/ aj_xgb02g_best /src/ All modelling is via Notebooks at this stage, although subfolders do contain function modules. Github repo https://github.com/adv-dsi-group4/nba-career-prediction

---

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	*Explain clearly what is the goal of this project for the business. How will the results be used? What will be the impact of accurate or incorrect results?* The aim is to predict which players are likely to still be playing after 5 years. The team will be able to focus our investment on those players who can be expected to last the distance. If our model results in a false positive, then we risk investing in players who will not be a good long term return. If our model results in a false negative, then we risk under-investing in (or passing on) players with potential. False positives will lead to worse financial outcomes than false negatives.

1.b. Hypothesis	*Present the hypothesis you want to test, the question you want to answer or the insight you are seeking. Explain the reasons why you think it is worthwhile considering it.* Having achieved a working model last week, using a logistic regression and a polynomial transformation, the team was keen to examine whether a different modelling algorithm might result in an even better fit. XGBoost is widely considered to be a powerful model for a range of different problems and often outperforms other models. We want to test this hypothesis on our problem.
1.c. Experiment Objective	*Detail what will be the expected outcome of the experiment. If possible, estimate the goal you are expecting. List the possible scenarios resulting from this experiment.* The aim of the experiment is to fit and tune an XGBoost model to the data in an attempt to outperform the Area Under the ROC Curve of 0.71133 that was achieved on the test data with our best model in the prior week. We were unsure how much improvement could be expected, but because XGBoost gets talked up quite a bit we'd hope to see a clear jump in our metric, at least above 0.72.

---

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	*Describe the steps taken for preparing the data (if any). Explain the rationale why you had to perform these steps. List also the steps you decided to not execute and the reasoning behind it. Highlight any step that may potentially be important for future experiments* As in the previous week's experiments: ● Data was examined to ensure training & test sets contained the same features to ensure that there's been no error in data loading. ● It was verified that the target variable is a binary classifier, to inform the types of models we might employ. ● We confirmed no null values, and hence no need to exclude samples or impute values. ● Some variables contain negative entries. It is unclear what this can mean, since the values are supposed to represent game statistics. This week, mini-experiments were performed to assess the value of different approaches to the problem of negative data values. Using both last week's logistic regression model and this week's XGBoost model, 3 different treatments were applied to any negative values in the features of the training, validation and testing datasets to see which resulted in the better fit.

	• Retain negatives • Turn negatives into absolute values • Replace negatives with null values. In both cases, retaining the negative values resulted in a better-fitting model. We will assume that the negative values are intended. If we had access to subject matter experts, we would confirm this with them.
2.b. Feature Engineering	*Describe the steps taken for generating features (if any). Explain the rationale why you had to perform these steps. List also the feature you decided to remove and the reasoning behind it. Highlight any feature that may potentially be important for future experiments* After last week's experiments determined that there was no benefit to be gained from dropping potential features from the training process, all modelling this week was applied to the full set of data features.

2.c. Modelling	*Describe the model(s) trained for this experiment and why you choose them. List the hyperparameter tuned and the values tested and also the rationale why you choose them. List also the models you decided to not train and the reasoning behind it. Highlight any model or hyperparameter that may potentially be important for future experiments.*
	
	Base xgb
	• As a baseline, an XGBoost model was fitted to the data with default parameters and the existing imbalance classes.
	• Results were AUC-ROC of 1.00 on the training set and 0.64 on validation, suggesting significant overfit.
	• The model was also overly concentrating on the positive class, getting 95% of them right but only 9% of the negative class.
	
	Tuned xgb
	• Next step was to use the HyperOpt package to tune the hyperparameters.
	• An optimisation space was created with the key hyperparameters that can help deal with overfitting, and using values indicated as being typically found in practice. In each tuning case, we checked that the tuned parameters were not at the edge of these ranges:
	    ○ Max_depth from 5 to 20
	    ○ Learning rates between 0.01 and 0.5
	    ○ Mi_chld_weights from 1 to 10
	    ○ Gamma from 0 to 0.1
	    ○ Subsample between 0.1 and 1
	    ○ Colsample_by_tree from 0.1 to 1.0
	• Using a 10-fold Stratified KFold cross-validation, the optimum values of each of the parameters were found, and the resulting 'best' model was again fitted to the training data and used to predict on the training data and the validation

	data. ● Results were essentially the same as the Base xgb. This was surprising, and suggested that the class imbalance aspect needed to be dealt with in order to improve results. Class weights - The XGBClassifier class that we are using has a hyperparameter 'scale_pos_weight' which can be used to scale up the minot class in an imbalance classification problem. - I included this parameter in the HyperOpt space, with values from 1 (no scaling) to 5 (close to full balance with the majority class). - Results were AUC-ROC of 0.89 on the training set and 0.65 on validation. An improvement on the base case in that it seems less overfitted, but not in terms of practical predictive power. Pipeline of SMOTE and undersampling - I also applied another strategy to deal with the imbalance classes. In last week's experiments I had some success with a modelling pipeline that included a SMOTE treatment to create additional synthetic minority-class observations, followed by an undersampling of the majority class. - After contructuring this pipeline and applying it to the tuning process of the XGBoost, it was found to give very similar results to the in-built scaling parameter in the XGBoost class itself.

---

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	*Score of the relevant performance metric(s). Provide analysis on the main underperforming cases/observations and potential root causes.* All of the XGBoost experiments this week have been disappointing. While an excellent exercise in the tuning of the algorithm, it has not come close to the predictive performance of the team's best model from last week, nor even to my own best model which was essentially a well-tuned but very straightforward logistic regression. The best XGBoost results on validation data was AUC-ROC of 0.68, lower than the 0.71 achieved last week. I have not delved into individual cases or feature importance, since the performance is so far off the mark.

3.b. Business Impact	*Interpret the results of the experiments related to the business objective set earlier. Estimate the impacts of the incorrect results for the business (some results may have more impact compared to others)* Compared to Week 1, the Week 2 model has delivered lower Precision: 86% down from 90%. In business terms, this means that if we invest in players according to this model then 86% of the players chosen will actually be likely to last 5 years, down from 90% if we use the previous model.
3.c. Encountered Issues	*List all the issues you faced during the experiments (solved and unsolved). Present solutions or workarounds for overcoming them. Highlight also the issues that may have to be dealt with in future experiments.* There were no significant new issues encountered in this experiment.

---

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	*Reflect on the outcome of the experiment and list the new insights you gained from it. Provide rationale for pursuing more experimentation with the current approach or call out if you think it is a dead end.* My key learnings from this experiment have been about how to practically tune an XGBoost model, and how to inject a data transformation pipeline into the process. And in terms of the business problem, my conclusion is that the XGBoost magic box is not always a silver bullet. Having tried to tune all the key hyperparameters and without improving on our predictive performance, I think the XGBoost experiments are concluded.

4.b. Suggestions / Recommendations	*Given the results achieved and the overall objective of the project, list the potential next steps and experiments. For each of them assess the expected uplift or gains and rank them accordingly. If the experiment achieved the required outcome for the business, recommend the steps to deploy this solution into production.* There are still other algorithm approaches to try, including SVM, KNN and Neural nets. I've had a preliminary run at both SVM and KNN, but armed with a more robust pipeline and a more efficient tuning space, I'll take another run at them. But my instinct tells me they won't do much better than we've done already so far. I am keen to investigate whether a neural network algorithm might be a step up from where we are now. And after that: an ensemble approach to see if more than 1 modelling approach can deliver the extra power we are hoping to find.