# **EXPERIMENT REPORT**

|  |  |
| --- | --- |
| **Student Name** | Peter Brotherhood |
| **Project Name** | Adv\_dsi\_at\_1 |
| **Date** | 19/11/2022 |
| **Deliverables** | brotherhood\_peter-12875237-week3\_Experiment\_1 LazyPredict  brotherhood\_peter-12875237-week3\_Experiment\_2 RandomForestUpsData  rf\_ups.joblib  rf2\_ups.joblib  rf3\_ups.joblib  rf4\_ups.joblib  Git address: https://github.com/PeRoBr/adv\_dsi\_at\_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** | 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 objective of this ML study is to identify rookie basketballers, at the completion of their first season in the NBA, who are most likely to have successful (5 year or greater) careers. Currently only the most experienced NBA pandits in America are able to assess a rookie’s game in their first year and predict careers prospects. These predictions result in multimillion dollar contracts and great competition between teams to sign the most promising rookies. Accurate prediction of quality players with potential for long careers will allow our client NBA teams to best price, risk assess and identify the best player value propositions in the NBA draft. |
| **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,  Our hypothesis is that performance metrics from a player’s first year in the league will be predictive of longevity in the league. We have detailed historical rookie performance data together with a 5 year outcome variable with which we believe we a train machine learning algorithms to out perform the pandits and deliver better player value to our client NBA teams. |
| **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.  Week 1:  The current experiment objective for this week is to understand the available data and run some initial, very simple predictions to see if, with very little experimental effort, predictions that are better than random guessing can be achieved.  Week 2:  This week we seek better performance with XGBoost than was achieved with Logistic Regression, and also examine if upsampling improves model performance in identifying those players who do not make it to 5 years in the NBA.  Week 3:  This week I have used the python package [Lazy Predict](https://pypi.org/project/lazypredict/) to quickly assess 27 different classifiers for their efficacy in this classification problem. Random forest classifier came out very ear the top in terms of auroc on test data, and this algorithm was selected for further experimentation with hyperparameter optimisation. |

|  |  |
| --- | --- |
| 1. **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  Week 1:  Data has been prepared via the following basic steps:  Drop player identifier column  Scale predictive variables mean 0 and variance 1 using [StandardScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html)  Week 2:  In addition to these steps, above, the training data has been balanced by upsampling using the steps described here: <https://wellsr.com/python/upsampling-and-downsampling-imbalanced-data-in-python/>. Upsampling was chosen as the available data was small and system resources could cope easily with the expanded dataset. The immediate alternative, downsampling, results in loss of a significant about of data and thus was decided against.  Week 3:  Again this week models were trained on upsampled data, as described for week 2. A 80-20 test train split was used on the upsampled data. |
| **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  none |
| **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  Week 1:  This early experiment was conducted with Logistic Regression from [Python package](https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html) SciKitLearn.  Week 2:  Again, I used Logistic Regression trained with balanced data as described above. Also, I trained (on raw imbalanced data and a balanced upsampled dataset) and tested GradientBoostingClassifier using default hyperparameters.  Week 3:  When the sklearn.ensemble.RandomForestClassifier was trained with default hyperparameters and predictions submitted to kaggle the results showed it had clearly overfit to the training data. On examination of the sklearn default hyperparameters it is not surprising that they tend to overfit. With max\_depth = None and min\_samples\_leaf = 1, impure leaves will continue to split until only a single record remains in the final leaf, ie overfitting. Also, with the current data set, models are training very quickly with 100 estimators. This first result prompted the following experimentation with hyperparameters:   * Sklearn default parameters: n\_estimators = 100, min\_samples\_leaf = 1 * n\_estimators = 1000, min\_samples\_leaf = 10 * n\_estimators = 1000, min\_samples\_leaf = 20   With each of these experiments the performance decreased slightly on the test data but improved on the kaggle data set. |

|  |  |
| --- | --- |
| 1. **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.  Week 1:  This early experiment promising results that machine learning can outperform a random guess. ~80% of rookie players do continue to a 5 year career, thus a random guess that ALL rookies make it to 5 years yields an accuracy in the order of ~80%. Of course, this prediction is of no value.  The performance of the logistic regression was determined via AUROC with a score of ~0.7. When the average probability (~80%) was applied to all players the AUROC score was 0.5.  Week 2:  This week I conducted 3 additional experiments on the previous week. We now have 4 experiments comparing 2 parameters: choice of predictive algorithm LogisticRegression or GradientBoostingClassifier (default hyperparameters), and training data imbalanced or training data balanced by upsampling.  Very interesting to note that LogisticRegression, to date, has performed better than the GradientBoostingClassifier with default hyperparameters.  Training algorithms on the balanced data yielded a small improvement in AUROC for both but still ~0.70.  Week 3:  Lazy Predict is an extraordinary tool for quickly assessing the relative performance of ML algorithms without the need for selecting hyperparameters. With only two lines of code and 60 sec computation time 27 algorithms were compared. As stated, randomforestclassifier was near the top in performance and was selected for hyperparameter tuning. With the incremental parameter changes described above the test AUROC score declined from 0.9, to 0.85 to 0.79. When submitted to kaggle predictions on the test set scored 0.68, 0.69 and 0.70, clear evidence that the overfitting was reduced by increasing estimators and min\_samples\_leaf. It is again interesting to note that that to date the most performant algorithm on kaggle has been the humble LogisticRegression trained on the upsampled dataset. In the direct comparison of LogisticRegression and RandomForestClassifier performed with LazyPredict, the AUROC for each algorithm was 0.64 and 0.97 respectively. |
| **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)  Week 1:  Very little business impact at this stage except to say that results to date support the idea that machine learning can be used with this data to predict player longevity.  Week 2:  We continue to accrue evidence that this problem can be solved by machine learning, and we have seen that balancing the training dataset makes a small improvement to model predictions.  Week 3:  We have yet to achieve a particularly useful prediction on the kaggle dataset but we have seen the effects of overfitting on the training data and begun to understand how to control this in the random forest classifier. |
| **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.  none |

|  |  |
| --- | --- |
| 1. **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.  Week 1:  As this is an unbalanced dataset, with ~80% of players making it to 5 years our next experiments will be to provide the algorithm greater relative experience with data from players who have not made it to 5 years. It is hypothesised that with this greater training experience the algorithm will better identify these players.  The techniques used to produce a balanced data set will be upsampling, down sampling and synthetic sampling.  Week 2:  This week we have some fairly good evidence that addressing imbalance in the training data set by upsampling improves prediction by both LogisticRegression and GradientBoostingClassifier.  It is very interesting to note that LogisticRegression, to date, has performed better than the GradientBoostingClassifier with default hyperparameters, given that the default parameters are a very reasonable starting place and GradientBoostingClassifier is a modern and high performing algorithm. Perhaps this suggests that tree based algorithms are not very well suited to this particular problem, however, some hyperparameter optimisation is worth to investigate.  Week 3:  To date the best performance achieved has been with a relatively simple algorithm Logistic regression. Despite fair to excellent performance on the labelled training data with the algorithms trialled so far, the perfomance on the kaggle data set has not exceeded an auroc of 0.71. Together with overfitting on the training data that we are beginning to understand how to control, this shows there is bias in the kaggle dataset that we have not yet overcome with our trained models. |
| **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.  None at this early stage. |