EXPERIMENT REPORT

Student Name	Mark Brackenrig (12964298)
Project Name	Assignment 1B
Date	14/02/2021
Deliverables	notebooks/brackenrig_mark-12964298-week2-SMOTE.ipynb models/brackenrig_mark_12964298_week2_SMOTE.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	The previous experiment highlighted that using an ensemble model of two different classifiers (random forest and logistic regression) saw a slight improvement in AUC. The main weakness of this model was that it struggled to predict the minority class. Based on the results of the previous experiment, I think that using an up sampling method would improve results by allowing the model to better learn patterns in the minority class.

1.c. Experiment Objective

The objective is to test whether training the model on a dataset with an up-sampled minority class will improve the AUC on the test dataset.

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 transformed using SMOTE (Synthetic Minority Oversampling Technique). SMOTE randomly selects one of the K nearest neighbours to synthetically generate a data point between the two data points in the feature space. SMOTE will create new data points until the classes are represented evenly in the sample.

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.

Unlike experiment 1, the models were trained on the entire training dataset rather than 80% of the training data. This decision was made as cross validation was used, so generating a new validation set was unnecessary.

2.b. Feature Engineering	Feature engineering was the same as experiment 1. The baseline model was used to conduct the test – which was a VotingClassifier model consisting of a linear regression model and a random forest model. The logistic regression model applied the StandardScaler method for normal standardization, and Principal Components Analysis was used to create orthogonal independent variables. This was used due to the high levels of collinearity identified in the dataset (see appendix for correlation matrix). No feature engineering was used for the random forest model.
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 Hyperparameter tuning was kept the same as in experiment one. In this experiment, testing the original model was tested against the original model trained on the oversampled training set. CalibrationClassifierCV was used to apply probability calibration on training of the new sample. Since the model included synthetic data in the training, it was important to calibrate the probabilities generated. CalibrationClassifierCV was also used to train the model without calibrating against the original training data, but using cross-validation using the SMOTE dataset.

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	BASE MODEL AUC (Training): 0.713 AUC (Test – Kaggle): 0.709 (On raw training data) Precision (Negative Class / Positive Class): 0.54/0.84

	Recall (Negative Class / Positive Class): 0.07/0.99 F1 (Negative Class / Positive Class): 0.13/0.91 SMOTE AUC (Training): 0.720 AUC (Test – Kaggle): 0.711 (On raw training data) Precision (Negative Class / Positive Class): 0.3/0.9 Recall (Negative Class / Positive Class): 0.59/0.72 F1 (Negative Class / Positive Class): 0.39/0.80 SMOTE – Calibrated on Original Data AUC (Training): 0.720 AUC (Test – Kaggle): 0.711 (On raw training data) Precision (Negative Class / Positive Class): 0.65/0.83 Recall (Negative Class / Positive Class): 0.01/1 F1 (Negative Class / Positive Class): 0.02/0.91 SMOTE – Calibrated using CV on Synthetic Data AUC (Training): 0.723 AUC (Test – Kaggle): 0.712 (On raw training data) Precision (Negative Class / Positive Class): 0.3/0.9 Recall (Negative Class / Positive Class): 0.59/0.72 F1 (Negative Class / Positive Class): 0.4/0.80
3.b. Business Impact	The best performing model was the SMOTE model using cross validation calibration on the synthetic data. The model has improved its ability to predict the minority class marginally. Unfortunately, the minor improvement is not large enough to realistically improve the business applications of the model. In saying this, small systematic improvements to the model will eventually create business value by improving predictions. By using synthetic data, we were able to improve on the AUC of the model marginally. To a certain extent, this has overcome the issue encountered in the first experiment where there wasn't enough data in the minority class.

3.c. Encountered Issues	While the SMOTE technique improved predictions, using synthetic data in model training needs to be used with caution. By using synthetic data, you are essentially ‘guessing’ what the classes should look like, and allow your model to train itself on this guess. A way around this could be using a bootstrap resampling method to under sample the majority class. While this would mean the model is trained on less data, it also avoids the issue of generating synthetic data. While this experiment focused on resampling the data, feature engineering, and base classifiers have not been explored in great detail. It is entirely possible that we are not using the most appropriate features or base classifiers for this task. More exploration is needed in this area.

4. FUTURE EXPERIMENT

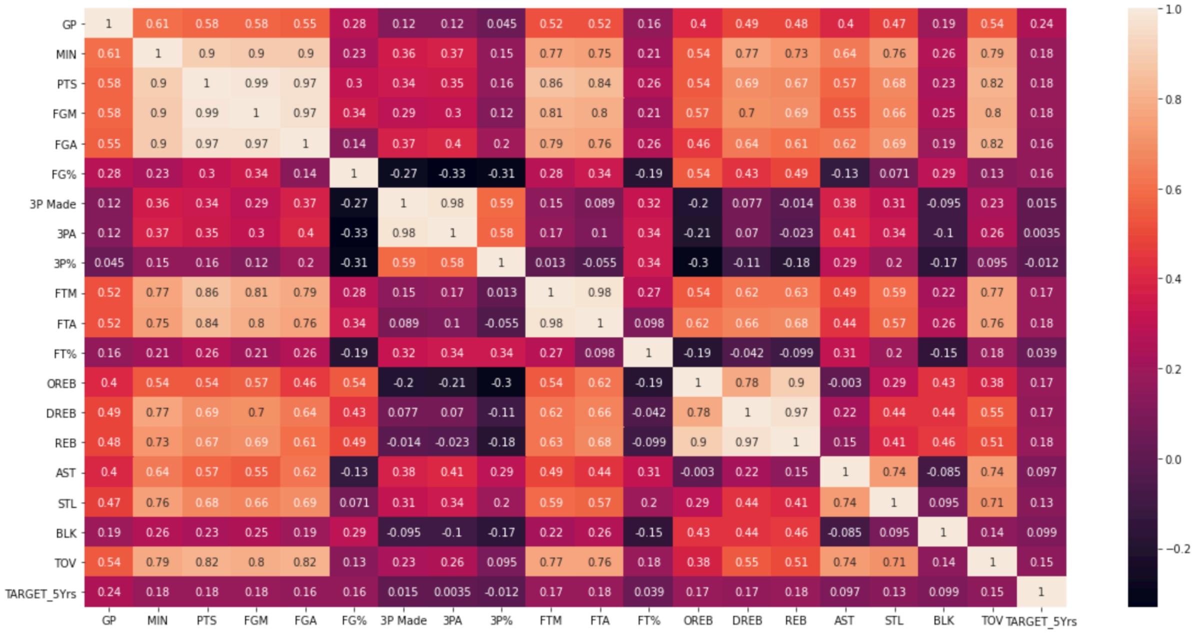
Reflect on the experiment and highlight the key information/insights you gained from it that are valuable for the overall project objectives from a technical and business perspective.

4.a. Key Learning	Our hypothesis was correct in assuming that the original model struggled to predict the minority class, and therefore an oversampling method improved the model. Our learnings thus far have shown that ensembling different classifiers and reweighting the training data has resulted in improving the prediction on the test set. We still need to complete more exploration into base classifiers and feature engineering as these have not been explored in great detail.
4.b. Suggestions / Recommendations	Future experiments should revisit comparing base classifiers. There are potentially many more complex classifiers not yet considered in our modelling process which could improve on the final model we use. The only feature engineering used to date has been PCA. Various other techniques could be employed (such as Non-negative matrix factorization or t-SNE) to account for non-linearity and improved information preservation. However, regardless of the modelling or feature engineering methods employed, there is an inherit limitation based on the

	dataset. Since we are trying to predict whether an NBA rookie will have a 5 year long career, it is entirely possible that there a level of randomness in the data. For example, career ending injuries to a star prospect would be a significant outlier in our dataset. While this is not possible in this context, in a professional context finding more data (such as reason for career ending) could be used to improve the model. In this context, we could potentially look at excluding or downvoting outliers to improve the model.

Appendix

Correlation Matrix



	GP	MIN	PTS	FGM	FGA	FG%	3P Made	3PA	3P%	FTM	FTA	FT%	OREB	DREB	REB	AST	STL	BLK	TOV	TARGET_5Yrs
GP	1	0.61	0.58	0.58	0.55	0.28	0.12	0.12	0.045	0.52	0.52	0.16	0.4	0.49	0.48	0.4	0.47	0.19	0.54	0.24
MIN	0.61	1	0.9	0.9	0.9	0.23	0.36	0.37	0.15	0.77	0.75	0.21	0.54	0.77	0.73	0.64	0.76	0.26	0.79	0.18
PTS	0.58	0.9	1	0.99	0.97	0.3	0.34	0.35	0.16	0.86	0.84	0.26	0.54	0.69	0.67	0.57	0.68	0.23	0.82	0.18
FGM	0.58	0.9	0.99	1	0.97	0.34	0.29	0.3	0.12	0.81	0.8	0.21	0.57	0.7	0.69	0.55	0.66	0.25	0.8	0.18
FGA	0.55	0.9	0.97	0.97	1	0.14	0.37	0.4	0.2	0.79	0.76	0.26	0.46	0.64	0.61	0.62	0.69	0.19	0.82	0.16
FG%	0.28	0.23	0.3	0.34	0.14	1	-0.27	-0.33	-0.31	0.28	0.34	-0.19	0.54	0.43	0.49	-0.13	0.071	0.29	0.13	0.16
3P Made	0.12	0.36	0.34	0.29	0.37	-0.27	1	0.98	0.59	0.15	0.089	0.32	-0.2	0.077	-0.014	0.38	0.31	-0.095	0.23	0.015
3PA	0.12	0.37	0.35	0.3	0.4	-0.33	0.98	1	0.58	0.17	0.1	0.34	-0.21	0.07	-0.023	0.41	0.34	-0.1	0.26	0.0035
3P%	0.045	0.15	0.16	0.12	0.2	-0.31	0.59	0.58	1	0.013	-0.055	0.34	-0.3	-0.11	-0.18	0.29	0.2	-0.17	0.095	-0.012
FTM	0.52	0.77	0.86	0.81	0.79	0.28	0.15	0.17	0.013	1	0.98	0.27	0.54	0.62	0.63	0.49	0.59	0.22	0.77	0.17
FTA	0.52	0.75	0.84	0.8	0.76	0.34	0.089	0.1	-0.055	0.98	1	0.098	0.62	0.66	0.68	0.44	0.57	0.26	0.76	0.18
FT%	0.16	0.21	0.26	0.21	0.26	-0.19	0.32	0.34	0.34	0.27	0.098	1	-0.19	-0.042	-0.099	0.31	0.2	-0.15	0.18	0.039
OREB	0.4	0.54	0.54	0.57	0.46	0.54	-0.2	-0.21	-0.3	0.54	0.62	-0.19	1	0.78	0.9	-0.003	0.29	0.43	0.38	0.17
DREB	0.49	0.77	0.69	0.7	0.64	0.43	0.077	0.07	-0.11	0.62	0.66	-0.042	0.78	1	0.97	0.22	0.44	0.44	0.55	0.17
REB	0.48	0.73	0.67	0.69	0.61	0.49	-0.014	-0.023	-0.18	0.63	0.68	-0.099	0.9	0.97	1	0.15	0.41	0.46	0.51	0.18
AST	0.4	0.64	0.57	0.55	0.62	-0.13	0.38	0.41	0.29	0.49	0.44	0.31	-0.003	0.22	0.15	1	0.74	-0.085	0.74	0.097
STL	0.47	0.76	0.68	0.66	0.69	0.071	0.31	0.34	0.2	0.59	0.57	0.2	0.29	0.44	0.41	0.74	1	0.095	0.71	0.13
BLK	0.19	0.26	0.23	0.25	0.19	0.29	-0.095	-0.1	-0.17	0.22	0.26	-0.15	0.43	0.44	0.46	-0.085	0.095	1	0.14	0.099
TOV	0.54	0.79	0.82	0.8	0.82	0.13	0.23	0.26	0.095	0.77	0.76	0.18	0.38	0.55	0.51	0.74	0.71	0.14	1	0.15
TARGET_5Yrs	0.24	0.18	0.18	0.18	0.16	0.16	0.015	0.0035	-0.012	0.17	0.18	0.039	0.17	0.17	0.18	0.097	0.13	0.099	0.15	1

Figure 1: Correlation Matrix Heatmap