**Module 17 Challenge Report**

**(Francis Odo)**

**Background**

There are several statistical algorithms used to perform predictive analysis and decision-making tasks in supervised machine learning models. This exercise is centered around techniques of classification and logistic regression. In all cases, the model is trained and evaluated. Then, we make predictions and generate confusion matrix.

**Objective**

The objective is to build a number of models, train and evaluate to assess credit risk based on the supplied data in LoanStats\_2019Q1.csv file.

**Development Environment** Python Pandas Jupyter Notebook

**Code Plan** The code plan follows similar, and in some cases same steps in for all models. 1. Read or load the csv data file (LoanStats\_2019Q1.csv) 2. Split data into features and targets 3. Split the data into Training and Test sets 4. Establish or fit the decision tree model 5. Make predictions 6. Evaluate the model 7. Calculate the balanced accuracy score 8. Calculate the confusion matrix and print classification report

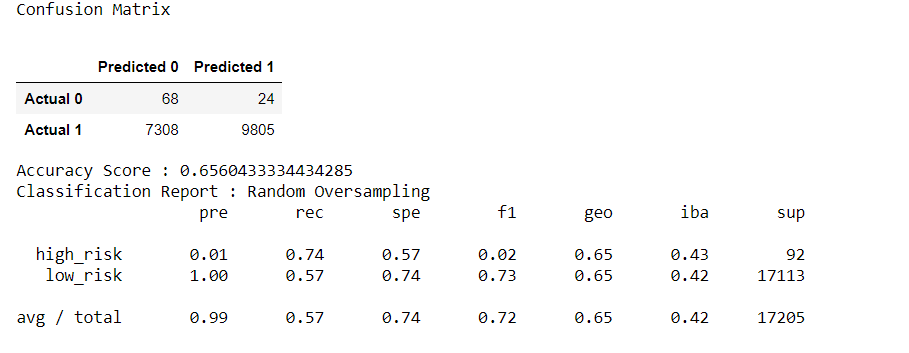
**Approach** Fit and resample the training data- instantiate the resampling module Train a logistic regression model Create predictions and generate the metrics Assess the results.

We performed all of the above for Random Oversampling, SMOTE Oversampling, Undersampling, Combination (Over and Under) SMOTE and The Easy Ensemble AdaBoost Classifier**.**

**Summary**

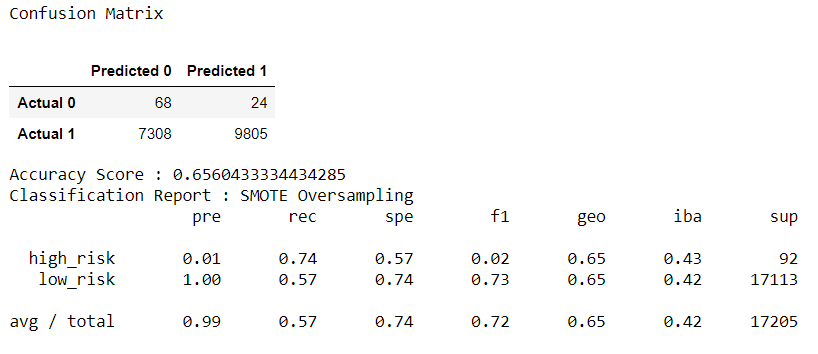
**a) Random Oversampling** – Initially, the balance of our target value was low\_risk = 68,470 high\_risk = 347. Counter(y\_resampled) showed low\_risk = 51357 high\_risk = 51357 (balanced).

The Confusion Matrix revealed [68, 24], [7308, 9805], which suggests that out of a total sample of 92 good loan applications (68 predicted low\_risk(0) while 24 predicted high\_risk (1) for Actual (0)). Also, out of 17,113 loan applications (7308 predicted low\_risk(0) while 9805 predicted high\_risk(1) for Actual(1). The calculated balanced accuracy score is 0.65, which indicates that the model is accurate 65% of the



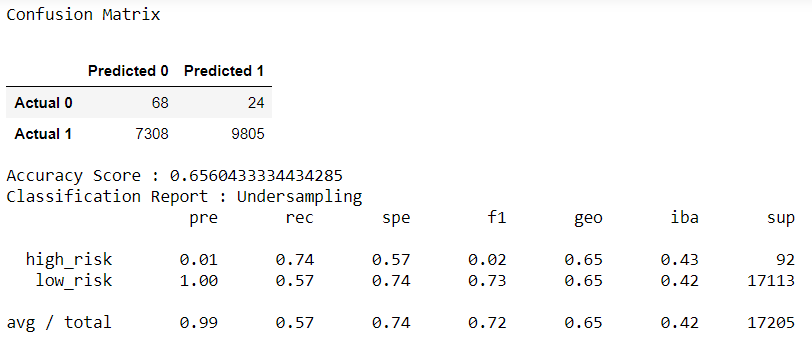
time. However, the Imbalanced accuracy score is 0.61 equating to 61% accuracy. Calculated F1 Score is at 0.72, while the Recall showed a calculated value of 0.57 slightly above average. F1 Score and Recall needs to be higher for a more reliable outcome. Calculated Precision is at 0.99, which indicates acceptable reliable classification.

**b) SMOTE Oversampling** – Counter(y\_resampled) is 51357 for low\_risk and 51357 for high\_risk. The balanced accuracy score is 0.612, with a confusion matrix of [53, 39], [5994, 11119]. This means that the model was correct 61.2% of the time.



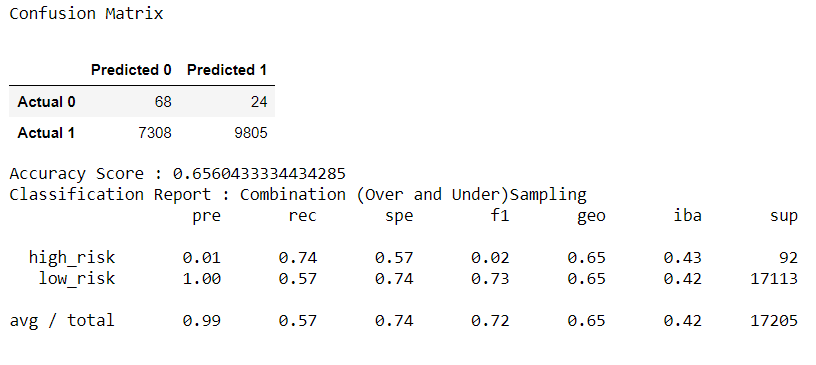
In this method of resampling, this model increases the minority class size to equal the majority class’s size. However, there is no noticeable significant difference in the overall outcome. Precision, Recall and F1 Score are unchanged.

**c) Undersampling** - This model applies the Clustercentroid method. It undersamples the majority class down to the size of the minority. The Counter(y\_resampled) shows low\_risk = 255 and high\_risk = 255.



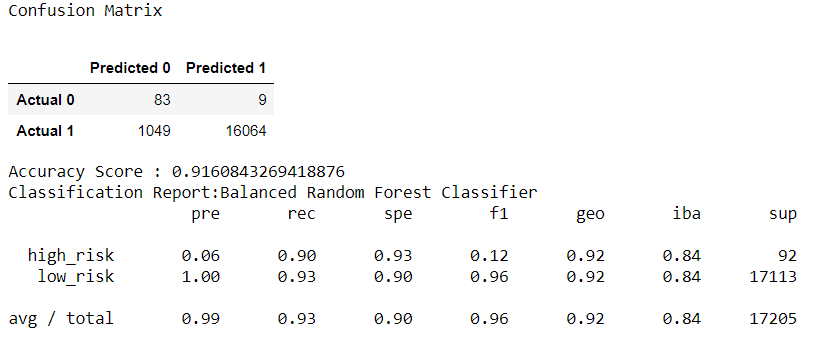
While we may have good classification with the Accuracy Score of 0.65, the matrix did not produce any better result for Precision, Recall and F1 Score.

**d) Combination (Over and Under) Sampling** – SMOTTEEN oversamples the minority class with SMOTE. This technique does not see the overall data spread. Instead, it focuses on the neighboring data points, but with the risk of losing data from outliers.



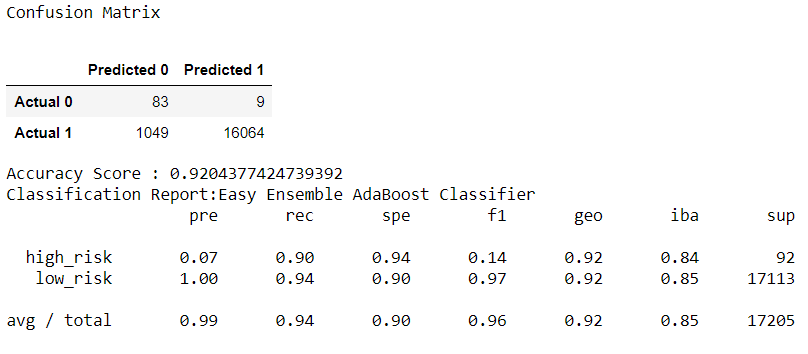
The outcome in this case does not demonstrate any significant advantage over other techniques so far.

e) Balanced Random Test Classifier – This model applies the technique that samples the data and build several smaller decision trees from a random subset of features. Decision trees are created for only that smaller piece of data, which makes them weak learners. In this model, many of these weak learners are combined to create a stronger learner with better decision-making performance. Accuracy score is at 91% with a Precision score of 0.99 and Recall Score at 0.93.



**The Easy Ensemble AdaBoost Classifier** – The Ensemble learning algorithm combines multiple models, which improves the accuracy through accumulations of predictions from a collection of models. Also, it decreases the variance or the spread of the distribution, thereby allowing for a more reliable performance.

The result shows a significant improvement across the matrix in all areas with consistency. Precision is at 0.99, Recall at 0.94 and F1 Score at 0.96. The prediction is significantly improved compared to all other models in this exercise. Highest accuracy score of 92% among all.



The Easy Ensemble AdaBoost Classifier and Balanced Random Forest Classifier models performed best considering the accuracy and consistency of the predictions across metrics. I will recommend either one of the two for use in the analysis because of the strong performance shown by the accuracy score and the precision score.

Among the advantages of Balanced Random Forest Classifier include 1) Robust against overfitting 2) Capability of ranking the importance of input variables 3) High efficiency running on large datasets.