

Oversampling : oversampling duplicates examples from the minority class in the training dataset.  
UnderSampling : underSampling deletes examples from the majority class

### Performance Comparisons For All Models That We Used

1. Logistic Classifier (Original Sample) : After using logistic classifier we got a balanced accuracy of around 66 % and a recall of about 33.33% which was not good as it is was not predicting the minority classes properly as we only got 31 right identification of negative class (0).
2. Logistic Classifier (Over Sample) :After Performing Over Sampling we got a balanced accuracy of around 76% which was very good and a recall of about 76.5% which was good as it is predicted minority classes efficiently as we got 71 right identification of negative class
3. Logistic Classifier (Under Sample) : After Performing Under Sampling we got a balanced accuracy of around 65% which was good and a recall of about 73% which was good as it is predicted minority classes effectively as we got 68 right identification of negative class.
4. Naive Bayes (Original Sample) : After using Naive Bayes classifier we got a balanced accuracy of around 73 % recall of about 66 % which was good as it is was predicting the minority class effectively but not better than Logistic (Over Sample) as we only got 61 right identification while we got 71 in case of that of Logistic (Over Sample) of negative class (0).
5. Naive Bayes (Over Sample): After Performing Over Sampling we got a balanced accuracy of around 68% which was good and a recall of about 74.19% which was good as it is predicted minority classes efficiently as we got 69 right identification of negative class but it reduced our balanced accuracy.
6. Naive Bayes (Under Sample):After Performing Under Sampling we got a balanced accuracy of around 44.14% which was poor and a recall of about 54% which was also not good as it predicted only 50 minority classes right.
7. Decision Tree Classifier (Original Sample):After using Decision Tree we got a balanced accuracy of around 59 % and a recall of about 31.11% which was not good as it is was not predicting the minority classes properly as we only got 29 right identification of negative class (0) but after parameter tuning our accuracy rose to 63.1 % recall rose to 37.6 % but was still not good.
8. Decision Tree Classifier (Over Sample): After Performing oversampling we got a balanced accuracy of around 61.33 %an recall of about 40.8% which was not good as it is was not predicting the minority classes properly as we only got 38 right identification.
9. Decision Tree Classifier (Under Sample):After Performing UnderSampling we got a balanced accuracy of around 54.5 %an recall of about 65.5% which was good as it was predicting the minority classes as we got 61 right identification but after parameter tuning our accuracy rose to 59.5 % and recall rose to 73.11 % but was still not good compared to Logistic (OverSample).
10. Random Forest (Original Sample):  
After using Random Forest classifier we got a balanced accuracy of around 58. and a recall of about 17.22 % which was not good as it was not predicting the minority classes properly as we only got 16 right identification of negative class (0).
11. Random Forest (Over Sample):After Performing Over Sampling we got a balanced accuracy of around 63.2 % and a recall of about 29 % which was not good as it was not predicting the minority classes properly as we only got 27 right identification of negative class (0).
12. Random Forest (Under Sample):After Performing UnderSampling we got a balanced accuracy of around 58.9 % and a recall of about 77.42 % which was good as it is was predicting the minority classes properly as we got 72 right identification of negative class (0) but overall accuracy reduced to 58.9 % which was not good.

13. K-NN (Original Sample): After K-NN we got a balanced accuracy of around 54.33 % and a recall of about 09.67 % which was not good as it was not predicting the minority classes properly as we only got 9 right identification of negative class (0).

14. K-NN (Over Sample): After Performing Over Sampling we got a balanced accuracy of around 64.8 % and a recall of about 65.59 % which was not good as it was not predicting the minority classes properly as we only got 61 right identification of negative class (0).

15. K-NN (Under Sample): After Performing UnderSampling we got a balanced accuracy of around 57.5 % and a recall of about 34.4 % which was not good as it was not predicting the minority classes properly as we only got 32 right identification of negative class (0).

16. SVM (Original Sample): After SVM we got a balanced accuracy of around 57.2 % and a recall of about 15.05 % which was not good as it was not predicting the minority classes properly as we only got 14 right identification of negative class (0) but after tuning parameters accuracy rose to 67.8 % and recall rose to 47.3 % and we identified 44 right negative class (0) same that for Over Sample.

17. SVM (Under Sampling): After Applying Undersampling we got a balanced accuracy 59 % and recall of 78. which was good as it was predicting the minority classes properly as we got 7 right identification of negative class (0) but our accuracy was very low so it was not a good model and when tuning the parameters the accuracy reduced to 50. While recall increased to 95.69 % because of which we were able to identify 89 minority class but precision reduced to 16 % and positive class that were identified were 29 which is not good.

#### # Final Comments

After Performing parameter tuning, over-sampling and under-sampling we came to for choosing a model with good recall\_score and a good balanced\_accuracy score. We are choosing balanced accuracy score because it is a measure of recall of positive class + recall of negative class and it outperforms f1\_score when positives >> negative. We according to the results got, the best model is Logistic Classifier with over Cause we are getting a good balanced accuracy around 76% and a recall of about 76.5%