SMOTE: Synthetic Minority Over-Sampling Technique

Kruthika Surineni
Masters of Computer Science
Arizona State University
Tempe, USA
ksurinen@asu.edu

Sagar Navgire
Masters of Computer Science
Arizona State University
Tempe, USA
snavgire@asu.edu

***Abstract*— Oftentimes it is important to build a classifier that is sensitive to the minority class samples within a dataset. Datasets with imbalanced class representations can lead to improper classifiers with high accuracy yet incorrect predictions. The cost of incorrectly classifying a minority sample could be much larger as compared to that of misclassifying a majority sample thus leading to an ineffective classifier. To tackle this problem, techniques such as SMOTE have been researched and developed to improve classifier performance and make the classifier more sensitive to the minority class. Over-sampling the minority class involves creating synthetic minority class examples rather than a strategy involving replication with replacement of minority class samples. Strategies such as the Area under the Receiver Operating Characteristics (ROC) curve and ROC Convex Hull are used to evaluate this technique.**

Keywords*— AUC, Classifier, Convex Hull, Minority, Over-sampling, Under-sampling, ROC

I INTRODUCTION

Datasets used for machine learning predominantly have balanced class representations. The problem arises when there are minority class samples within a dataset which are "interesting". For example, consider the task of predicting massive cell growth in a person's body as cancerous or not. The number of cases where the cell growth is attributed to cancer would be very less in number, almost of the order of 1 in every 100,000. Few samples and the cost of misclassifying cancerous tumor as non-cancerous makes building an accurate classifier vital and complex at the same time.

The motive of this paper is thus to highlight the importance and effectiveness of Synthetic Minority Over-Sampling Technique to make a classifier more sensitive to minority class samples.

II DISCUSSION

A. *Imbalanced Datasets*

The two broad ways of tackling the class imbalance issue are: Assignment of distinct costs to training examples and resampling the original dataset by oversampling the minority class and/ or under-sampling the majority class.

Previous work in the area included Kubat and Matwin's work [1] by selectively under-sampling the majority class while keeping the original population of the minority class [1]. Japkowicz also discussed the effect of imbalance in a dataset [2]. Three strategies were evaluated: under-sampling, resampling and a recognition-based induction scheme. Her work on sampling was particularly important which included random and focused variants of sampling. "*Random resampling"* involved resampling the minority class samples at random until it was approximately equal in number to the majority class samples. The "*focused resampling*" consisted of resampling only those minority examples that occurred on the boundary between the minority and majority classes. Under-sampling included similar work in the area. Random under-sampling consisted of under-sampling the majority class samples at random until the proportion of minority and majority class samples were similar. Focused under-sampling involved under-sampling the majority class samples lying further away. It was noted that both the sampling approaches were effective but did not give any clear advantage in the domain considered.

Ling and Li's work [3] was particularly influential in the idea behind the SMOTE technique. Over-sampling of the minority class was combined with under-sampling of the majority class.

Oversampling was mainly done by replicating examples from the minority class with replacement [4], in the original dataset until the minority and majority classes had equal representation. Such replication lead to more specific decision regions and more terminal nodes in the decision trees leading to overfitting as the classifier tries to learn the minority class exactly. The under-sampling technique involves under-sampling the majority class to match the number of samples in the minority class. Both these techniques suffer from flaws and fail to support a classifier that can accurately predict minority classes. Hence, the authors of the paper on the SMOTE technique [5], describe an over-sampling methodology in which synthetic examples are created rather than replicated with replacement.

Upon reading these papers we understood the relevance of this topic and the amount of research that has gone into building an algorithm that might tackle resultant issues by the efforts to improve classifier performance. Simple oversampling by

replication and under-sampling don't seem as effective as a method that would shift the bias from the majority class towards the minority class. This project seeks to utilize a combination of the Minority Over-Sampling technique either using SMOTE or random sampling with majority class Under-Sampling in combination with C4.5, Ripper and Naïve Bayes Classifiers. To summarize, the synthetically generated minority class samples cause the classifier to build larger decision regions that contain nearby minority class points allowing a learner to carve broader decision regions leading to more coverage of the minority class.

B. *Performance Measures*

The performance of classifiers for a 2-class problem, is generally measured using a confusion matrix (as shown in figure 1). This is a comparison between the actual classes and the predicted classes of the samples giving us True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN) values.

	Predicted Negative	**Predicted Positive**
Actual Negative	TN	FP
Actual Positive	FN	TP

Table 1: Confusion Matrix

Accuracy of classifier predictions is the performance measure generally associated with machine learning algorithms and is defined as:

$$Accuracy = (TP + TN) \div (TP + FP + TN + FN)$$

But, this does not consider imbalanced datasets and thus, returns a high accuracy rate by simply classifying all the samples as the majority class. The *error rate* also cannot be simply calculated as: $(1 - Accuracy)$

On account for this, the Receiver Operating Characteristic better explains such imbalanced datasets with unequal error costs. The ROC curve explains the relative costs of the True Positive and False Positive classified samples. Area under the ROC curve (AUC) is also effective in choosing an optimal classifier from multiple classifiers. AUC is particularly useful as the class priors does not affect the measurement of classifier performance.

III ALGORITHM DETAILS

Our work starts with the implementation of the SMOTE algorithm. This algorithm works on the minority class samples within a dataset. The paper [5] introduces an innovative approach of over-sampling the minority class by generating synthetic samples by operating over the data in feature space. The algorithm which was inspired from work on handwriting recognition [6] generates a synthetic sample for every minority sample along the line segments joining any or all its selected k nearest neighbors. Fig. 1 is the pseudocode for the SMOTE algorithm as taken from the original paper [5] that we have implemented.

Based on the amount of oversampling specified, the algorithm randomly selects 1 to k of the k nearest neighbors and creates new synthetic sample in the direction of each one of them. While generating a synthetic sample, for each of the attributes, the difference between the minority sample and its nearest neighbor is calculated. This difference is multiplied by a random number between 0 and 1, and added to that feature vector. This process picks a random point on the line segment joining the two features under consideration. The SMOTE algorithm pushes the decision regions to become more general for the minority class as compared to over-sampling with replacement which results in more specific decision regions and thus over-fitting.

```
SMOTE(T, N, k):
1. if (N<100)
2.       Then Randomize T minority class samples
3.              T = (N/100) * T
4.              N = 100
5. endif

6. N = (int)(N/100)
7. k: Number of nearest neighbors
8. numAttrs: Number of attributes (features)
9. minoritySamples: List of minority class samples
10. newindex: number of syntehtic samples generated
11. Synthetic: List of Synthetic samples

12. for i <- T
13.       nnarray = Compute k nearest nighbors for i
14.       Populate(N, i, nnarray)
15. endfor

Populate(N, i, nnarray):
16. while N > 0
17.       nn = select the random nearest neigbor from nnarray
18.       for attr <- 1 to numattrs
19.              Compute: dif = Sample[i][attr] - Sample[i][attr]
20.              Compute: gap = random number between 0 and 1
21.              Synthetic[newindex][attr] = Sample[i][attr] + gap * dif
22.       endfor
23.       newindex++
24.       N = N - 1
25. endwhile
26. return Sythetic
```

Figure 1: SMOTE Algorithm

Section 4.3 of the paper [5] explains the combination of Under-sampling and SMOTE. Under-sampling is performed by randomly eliminating the majority class samples until the minority class becomes the specified percentage of the majority class. For example, under-sampling the majority class at 100% would result in the number of majority samples equal to those of the minority class samples. Such under-sampling enables the minority class to dominate at higher percentages. This combination of under-sampling and SMOTE reverses the initial bias of majority class towards minority class.

IV DESIGN OF EXPERIMENTS

We used two classifiers to obtain the necessary evaluation metrics for the synthetic datasets generated:

1. **C4.5 Decision Tree Classifier**: Decision tree classifier

provided by http://scikit-learn.org. [6] This classifier uses CART technique which is analogous to C4.5.

2. **Naive Bayes Classifier**: We used Gaussian Naïve Bayes classifier from http://scikit-learn.org was used to compare the ROC curves obtained from the C4.5 classifier. As mentioned in the paper, the minority class priors were varied from 1 to 50 times the majority class to plot the ROC curve.

Fig.3 provides an overview of the experiments conducted. We used the best percentages (N) mentioned in the paper [4] for each dataset for the SMOTE algorithm. We also varied the amount of under-sampling from 10% to 2000% depending on the dataset and its distribution. Two datasets were generated based on amount of under-sampling, one consisting the original minority samples and other consisting the SMOTEd samples. These datasets were fed into C4.5 classifier to generate the respective ROC curves. Further the Naïve Bayes classifier was used to plot another ROC curve using the raw dataset. All the curves were plotted on a single graph and compared based on the Convex hull and AUCs obtained.

The tasks of plotting the ROC curves and Convex hull, AUCs calculation were performed using the functions provided by http://scikit-learn.org.[11][12][13]

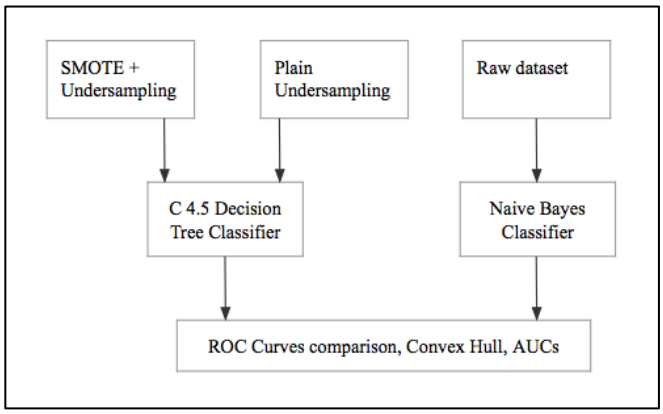

Figure 2: Overview of Experiments.

We used three datasets for verifying the results for the proposed technique. These datasets largely vary in size and distributions. The datasets used were:

1. **The PIMA Indian Diabetes**: This multivariate bi-nary class dataset consists of a total of 768 samples with 286 minority class samples. [8] It is used to identify positive diabetes cases.

2. **The Phoneme Dataset**: This dataset is obtained from the ELENA project. [9] It is a binary class data where class 0 signifies a nasal sound whereas class 1 signifies an oral sound. It has five features and the minority class samples sums to 1586 out of the total 3818 samples.

3. **The Satimage Dataset**: This multi-spectral dataset has 6 classes. [10] As mentioned in the paper the class with the smallest number of samples was chosen as minority class and rest all the 5 classes were merged to form the majority class. The modified dataset consisted of 626 minority class samples and 5809 majority class samples.

V RESULTS

The experiments carried out allowed the study of the performance of different classifiers on various combinations of the data. The Naïve Bayes implementation in which the weights for each class can be varied was compared with the results of a decision tree classifier that learnt and generalized decision regions based on the datasets.

%FP and %TP were calculated over the 10-fold cross validation done in which in each iteration, 1 fold of the uniformly distributed data into folds, was used as test data with the remaining 9 folds being training data. This was done to build a thorough classification model. The Naïve Bayes classifier used the dataset with the original class distributions while varying the class priors, i.e., minority class distribution increased from 1 to 50 times the majority proportion. The prior combinations were varied in the following way for all the datasets: [[Majority, Minority]]: [[0.5, 0.5], [0.333, 0.667], [0.167, 0.833], [0.091, 0.909], [0.062, 0.938], [0.048, 0.952], [0.038, 0.962], [0.032, 0.968], [0.028, 0.972], [0.024, 0.976], [0.022, 0.978], [0.02, 0.98]]. So, for each class prior combination, we obtained the mean %TP and mean %FP over the 10-fold cross validation performed. Similarly, the C4.5 Decision Tree classifier model returned mean %TP and mean %FP values of the 10-fold cross validation done for a fixed rate of over-sampling and varied rates of under-sampling of the majority class in the SMOTE technique. This was comparable with the results for the simple under-sampling technique where the minority class samples were constant and only the majority class was under-sampled.

As sampling is random, we noticed a lot of variations in the results obtained. The classification accuracy of the models built varied based on the class distributions and the structure of the data. But it was noticed that as the under-sampling factor was increased we obtained better %TP. This also include an increase in the %FP with respect to the minority class but in most cases, a higher %TP is what is desired.

Fig. 3 below shows the results obtained based on the original PIMA dataset and different prior combinations for the Naïve Bayes Classifier. This is a blue line on the graph. The C4.5 Classifier on the plain under-sampled data at different rates of under-sampling is shown with a green line. The C4.5 Classifier with 100% SMOTE over-sampling and under-sampling done at different rates is the red line. We can all see the convex hull of the best points obtained from all the classifiers and data combinations. We can see the red line and the convex hull almost overlap which shows that the ROC Curve obtained for SMOTE + Under-sampling by C4.5 performs the best. A comparison with the graphs from the original paper (*Fig. 4 below*) shows that the curves obtained are somewhat similar without interpolation being done. Their Naïve Bayes classifier seems to operate better in this case.

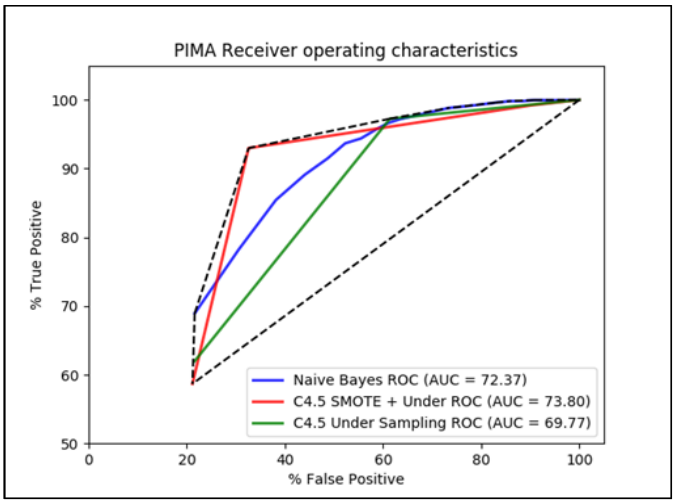

Figure 3: Obtained ROC Curves and Convex Hull plotted for the PIMA Indian Diabetes Dataset. SMOTE + Under-sampling curve drawn for a 100% over-sampling rate.

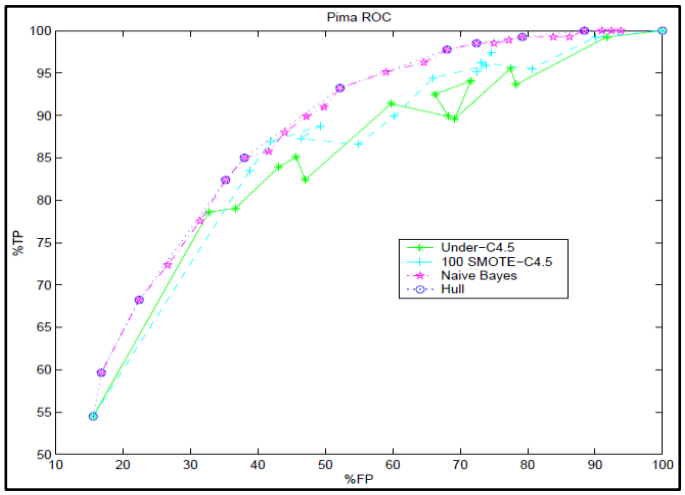

Figure 4: PIMA. Comparison of SMOTE-C4.5, Under-C4.5, and Naïve Bayes. Naïve Bayes dominates over SMOTE-C4.5 in the ROC space from the original paper [5].

Fig. 5 shows the results obtained based on the original Phoneme dataset and different prior combinations for the Naïve Bayes Classifier with a blue line on the graph. The C4.5 Classifier on the plain under-sampled data at different rates of under-sampling is a red line. The C4.5 Classifier with 200% SMOTE over-sampling and under-sampling done at different rates is the green line. We can see that that the ROC Curve obtained for SMOTE + Under-sampling by C4.5 seems to achieve high %TP for a very low %FP indication good classifier performance Fig. 6 shows the ROC curves obtained in the original paper [5]. The graph shows similar order of performance of classifiers based on the data and techniques as obtained in fig. 5.

Fig. 7 and fig. 8 are the ROC curves obtained for the Satimage dataset. Again, the SMOTE curve represented by the green line seems to obtain high %TP for low %FP and is close to the convex hull. We see similar curves in fig. 8 which is from the original paper [5].

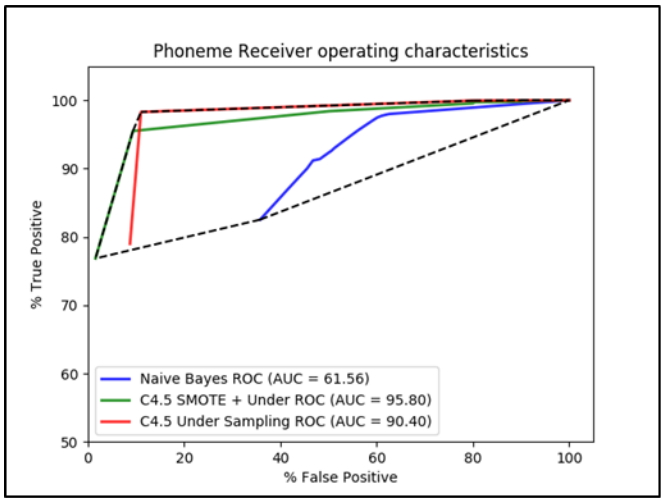

Figure 5: Obtained ROC Curves and Convex Hull plotted for the Phoneme Dataset. SMOTE + Under-sampling curve drawn for a 200% over-sampling rate.

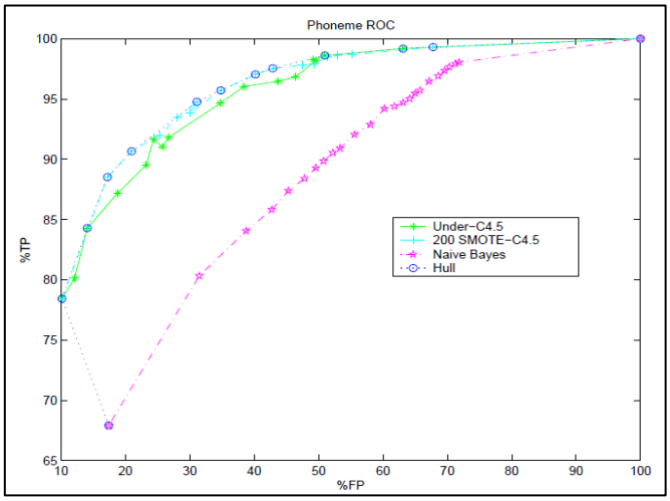

Figure 6: Phoneme. Comparison of SMOTE-C4.5, Under-C4.5, and Naive Bayes. SMOTEC4.5 dominates over Naive Bayes and Under-C4.5 in the ROC space. SMOTEC4.5 classifiers are potentially optimal classifiers.

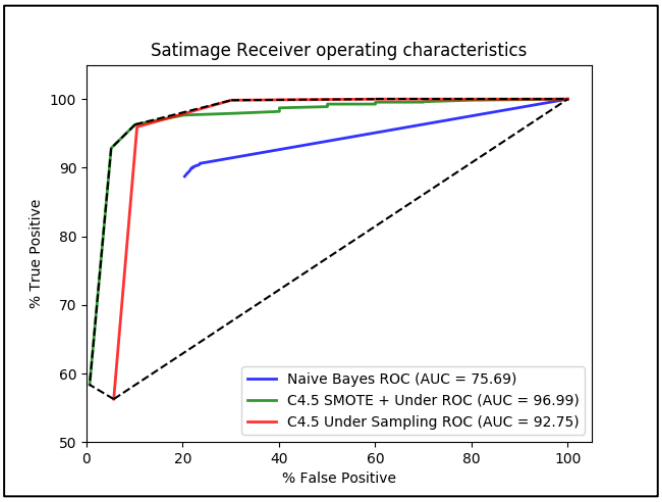

Figure 7: Obtained ROC Curves and Convex Hull plotted for the Satimage Dataset. SMOTE + Under-sampling curve drawn for a 100% over-sampling rate.

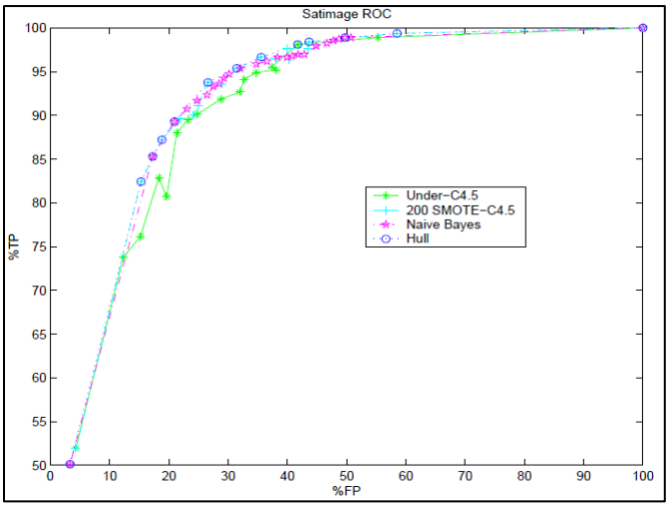

Figure 8: Comparison of SMOTE-C4.5, Under-C4.5, and Naive Bayes. The ROC curves of Naive Bayes and SMOTE-C4.5 show an overlap; however, at higher TP's more points from SMOTE-C4.5 lie on the ROC convex hull as the original paper [5].

Data Set	Plain Under sampling	SMOTE + Under sampling
PIMA	73.80	69.77 (100 SMOTE)
Phoneme	90.40	95.80 (200 SMOTE)
Satimage	96.99	92.75 (200 SMOTE)

Table 2: AUCs obtained for the 3 datasets for the ROCs obtained in the previously described graphs. This shows the comparison between AUC obtained for plain under-sampling and the SMOTE + under-sampling combination.

VI ASSUMPTIONS AND PROBLEMS ENCOUNTERED

It was a challenge to first identify the order of under-sampling and over-sampling done on the dataset. The resultant proportions would be drastically different if under-sampling was done based on the new count of minority class samples. But, in order to match the number of majority class samples in the SMOTE implementation and the simple under-sampling approach we needed to perform under-sampling before SMOTE was done. Also, the prior used for the Naïve Bayes Classifier was not mentioned in the original paper. We assumed multiple prior combinations ranging from 1 to 50. The paper [5] was also unclear about the under-sampling done and the values of points on the curve. We had to run multiple experiments and identify the best combinations of under-sampling and over-sampling of the samples in the dataset to obtain the ROC curves needed. We have plotted curves without interpolation done and hence have crude plots.

VII FUTURE SCOPE

We can work on other classifiers to see how this algorithm performs. A more diverse range of datasets will give us an idea of what kind of datasets this algorithm works best for. Upon doing that, we can then further analyze other approaches to classify imbalanced datasets. Also, we can run more experiments to identify factors in order to set the over-sampling rate and identify the optimal number of nearest neighbors to be identified so as to save time and resources as the data scales.

VIII POTENTIAL AREAS OF STUDY

Imbalanced datasets are quite common now. Study can be performed on the structure of the data to link it to the rate of under-sampling and over-sampling required. Classification algorithms that perform well on such datasets need to be identified. If various kinds of models can be trained and tested on such data, we would know what classifiers work best for the particular dataset and what kind of sampling needs to be done. This would save time and ensure reasonably good accuracy with respect to ROC curves.

IX CONCLUSION

The research and work done as a part of this project shows that the SMOTE approach can improve the accuracy of classifiers for a minority class. SMOTE provides an innovative approach to over-sampling. SMOTE and under-sampling combined performs better than plain under-sampling. SMOTE was tested on a variety of datasets, with varying degrees of imbalance and varying amounts of data in the training set. The combination of SMOTE and under-sampling also performs better, based on domination in the ROC space, than varying the class priors in Naive Bayes Classifier. SMOTE forces focused learning and introduces a bias towards the minority class.

X REFERENCES

1. Kubat, M., Holte, R., & Matwin, S. (1998). Machine Learning for the Detection of Oil Spills in Satellite Radar Images. Machine Learning, 30, 195–215.
2. Japkowicz, N. (2000). The Class Imbalance Problem: Significance and Strategies. In Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000): Special Track on Inductive Learning Las Vegas, Nevada.
3. Ling, C., & Li, C. (1998). Data Mining for Direct Marketing Problems and Solutions. In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98) New York, NY. AAAI Press.
4. Japonica, N. (2000). The Class Imbalance Problem: Significance and Strategies. In Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000): Special Track on Inductive Learning Las Vegas, Nevada.
5. Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegel Meyer. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16:321–357, 2002.
6. http://scikit-learn.org/stable/modules/tree.html
7. http://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html#sklearn.naive_bayes.GaussianNB
8. https://archive.ics.uci.edu/ml/datasets/pima+indians+diabetes
9. https://www.elen.ucl.ac.be/neural-nets/Research/Projects/ELENA/databases/REAL/phoneme/
10. https://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite)
11. http://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html
12. http://scikit-learn.org/stable/modules/generated/sklearn.metrics.auc.html
13. https://docs.scipy.org/doc/scipy-0.19.0/reference/generated/scipy.spatial.ConvexHull.html