Breast Cancer Classification using Ensemble Methods and Clustering

Gaurav Naresh
Department of Computer Science
University of Texas at Dallas
Texas, United States
gaurav.naresh@utdallas.edu

Vignesh Thiagarajan
Department of Computer Science
University of Texas at Dallas
Texas, United States
vignesh@utdallas.edu

***Abstract*—In the modern world, one of the biggest challenges we are facing as a society is breast cancer. We need to detect it early so as to provide the best treatment possible. The use of machine learning in the medical field has revolutionized the way diagnosis is done. Here, we have discussed how an ensemble method like random forests and also k-NN have been used for calculating the accuracy to which the cancer is benign or malignant based on data from a well-renowned repository. The advantage here is that accuracy is way better than a single decision tree in case of a random forest.**

Keywords—ensemble, nearest-neighbors, prediction, random-forests.

I. INTRODUCTION

Breast cancer detection is one of the most intriguing field of research today. There are so many women in the world who are diagnosed with this. Breast cancer is generally because of the change in the hormones in the female body and risk percentage calculated is different in different stages of life. Machine learning techniques has become very common nowadays in the healthcare industry especially for the detection and diagnosis of this cancer. ML can be used for analysis of large datasets of different patients to provide information accurately such as if a person is likely to get a chronic disease, if a person requires to be readmitted etc. Using ML and deep learning algorithms have proven to reduce the mortality of a patient by one year. This is quite a significant figure in terms of how a person's wellbeing and health are taken care off. With the help of these algorithms the speed of diagnosis also increases multiple folds which saves lot of time and money.

We have various classification techniques in machine learning out of which we have implemented random forests and k-nearest neighbors for our analysis.

Ensemble methods in machine learning combine many processes into a single one. By doing this, the prediction and accuracy is much higher compared to each of the techniques used individually. These methods reduce bias and variance and increase predicting accuracy greatly. When we compare we will see that decision tree ensembles give lesser error compared to ensemble methods of k-NN.

II. MACHINE LEARNING -ALGORITHMS

A. Random Forest

Random forest is a group of decision trees put together in order to make the prediction and solve problems of both classification and regression. The basic idea is to return the depth of individual decision trees and the leaf nodes so that this can be used by the random forest to optimize the output. One of the best ensemble methods in ML is the random forest method. It is one of the best classification methods for both minor and major datasets.

We need to combine various eigen values to get optimum results. It was brought into theory by a mathematician named Ho. He observed that these can keep increasing accuracy without the fear of overtraining, with a condition that it keeps training on a selected dimension of features. There cannot be huge dimensions involved as independence/freedom of each decision tree will decrease drastically. The maximum dimensions of the features can be the square root of maximum dimensions or logN+1.

It is also called a combined classifier because it is a combination of K decision trees. It does this by a method called as the bagging-sampling. This typically means we need to have that many training set which is a fraction approximately around 0.66 of the original set used for training. The process behind this really amazing. If we closely observe, after every step of the so called bagging sampling only around 60% is collected as a training set and the remaining is not. So, this gives us a basis to form something called as a test data. This test data has a specific name called Out Of Bag or OOB.

The description of this algorithm is as follows: the decision tree obtained after random sampling is now eligible to be trained with data. The idea to be kept in mind is that each and every decision tree have no dependence on another thereby have no dependence on the results as well. Another very important concept is of voting. This method involves choosing the best or the optimum solution from the results.

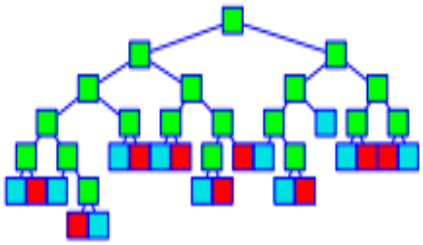


Figure 1: Basic Topology

It can be clearly seen from the above description that the random selection of samples and attributes can avoid overfitting.

The decision trees and the random forests are based on ID3 algorithm. This algorithm travels through the entire tree starting from the top node or root. From there it goes to each

branch and calculates entropy. It also is used for determining the information gain of the attributes. This algorithm works on the principle of recursion and it recurses again and again until it reaches no more unreached or unselected attribute.

There are three possibilities because of which this algorithm can terminate. First, if all data points belong to a specific class, it is classified as a node with no children or leaf and hence terminated. Second, if there are no more attributes or if there are no more examples itself then the recursion comes to a halt. A major drawback with this algorithm is that it can lead to overfitting. So as a preventive measure we generally use minute decision trees compared to larger ones. Another major advantage of this is that this does not give the best time complexity if data happens to be continuous. The reason is there will be multiple places where splitting can take place and deciding among them is costly.

B. K Nearest Neighbors

In K Nearest Neighbors algorithm predictions are made directly using the training data set. We search the entire dataset for k identical instances so that we can make a good prediction for a new instance. Breast cancer prediction is a classification problem and for such problems we need to find the mode of the data instead of mean which is used in regression examples.

This algorithm is also defined as a lazy algorithm which is capable of good learning. It gives us the flexibility that the data can be either a scalar or a vector which is multidimensional. In early stages of development of the algorithm there were two classes used which were "positive" and "negative". But now, it can work with any number of classes. This algorithm just expects the value of K as its input. If the number of classes is an even number, then basically the value of K is an odd number.

KNN involves using k, and labels for prediction. The way we choose our k becomes vital in how our output is obtained. This algorithm is a trial and error algorithm where we have to keep choosing different values of k to find an optimum value. If by chance we choose a bigger value, cost of calculation is enormous and also suppresses the structure of data. Similarly, if we choose a K which is very very small we see that the effect of noise on the output will be very high.

$$\text{Euclidean Distance}(x, xi) = \text{sqrt}(\text{sum}((xj - xij)^2))$$

The following are the steps involved in the k-NN process:

- At first, we initialize the value of K.
- After getting this value from user, we need to measure and evaluate the distance between input sample and training sample space.
- After we get this distance, we need to use any of the sorting algorithms and sort these distances.
- Now, we take the k nearest neighbors.
- The nearest neighbors are placed in a buffer and a simple majority is taken.

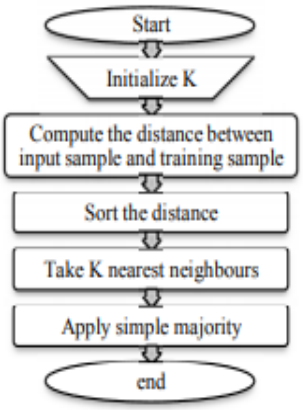


Figure 2: K-Nearest Neighbor Algorithm

III. Literature Survey

Li Deng and Dong Yu mentioned about various supervised and unsupervised techniques and also their application in various fields like autonomous cars, satellite imaging etc. Mellisa used a different kind of preprocessing like the morphing preprocessing to identify all lumps from the mammogram. Ahmeet Mertet explains in depth about principal component analysis on this cancer dataset. She studied results of various classifiers such as k-nearest neighbours, neural networks etc. S. Kharya in his paper mentioned about the advantages and disadvantages of various classification problems she identified when she ran it on a breast cancer dataset. The algorithms she used were Support vector machines and Bayesian classifier.

Dheeba et.al used a support vector machine on this dataset and just reported an error of 13%. Suki et.al used a similar strategy by using Support vector machines and a radial basis kernel to give an error of just 3%.

IV. Random forest Implementation Without scikit learn

Random Forest is implemented without Sklearn's inbuilt model. The following steps are involved in the Implementation of the algorithm.

A. Preprocessing of Breast Cancer Dataset

The Breast cancer dataset is preprocessed by dropping missing values and unwanted columns. The ID's column of the dataset is of no significance and hence dropped. The last column is called Unnamed 32, and consists of NaN values. This column seems to be an innate error of the dataset and also is dropped as it is unnecessary. Also, the Class/diagnosis column is labelled into classes 0 and 1 from Benign and Malignant.

B. Exploratory Analysis

The dataset is explored to discover its various features, and the way they affect the distribuion. First of all, the split up between Benign and Malignant counts were observed. The histograms were also observed for each feature and this further plays a part in the Feature Selection process.

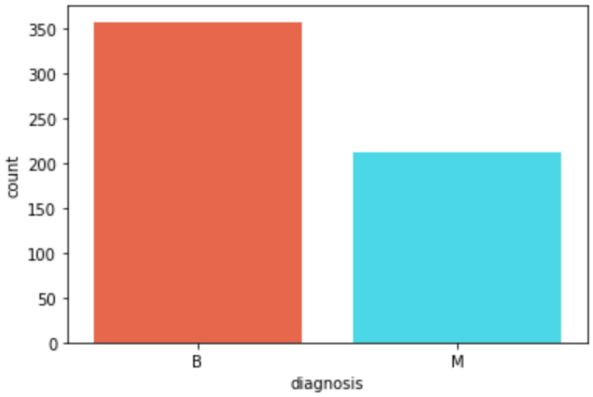


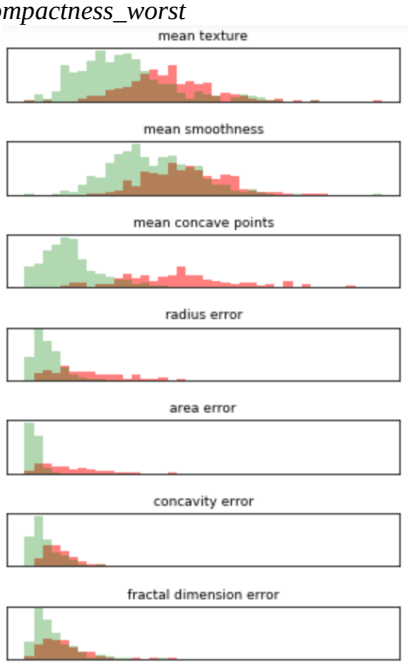
Figure 3: Distribution into the two classes

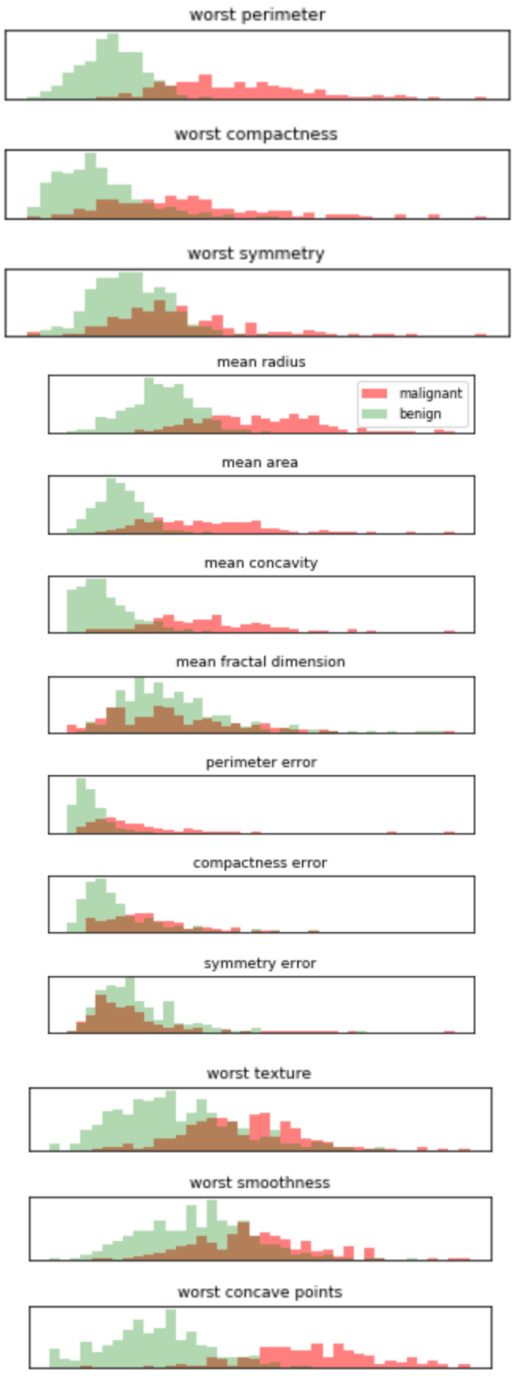
C. *Feature Selection*

Further exploration helps us to analyze the dataset, and the features that can best split the dataset into the two classes. This helps play a part in deciding the most important features that explains most of the variance in the dataset.

The diagrams below show the distribution between the two classes and we can visually decide on which features best splits this dataset into its two classes: “Benign” and “Malignant”. From the figure below, the following features/columns were selected to move forward with:

1) *radius_worst*
2) *concave points_mean*
3) *area_worst*
4) *area_mean*
5) *concave points_worst*
6) *perimeter_mean*
7) *area_se*
8) *concavity_worst*
9) *radius_se*
10) *compactness_worst*





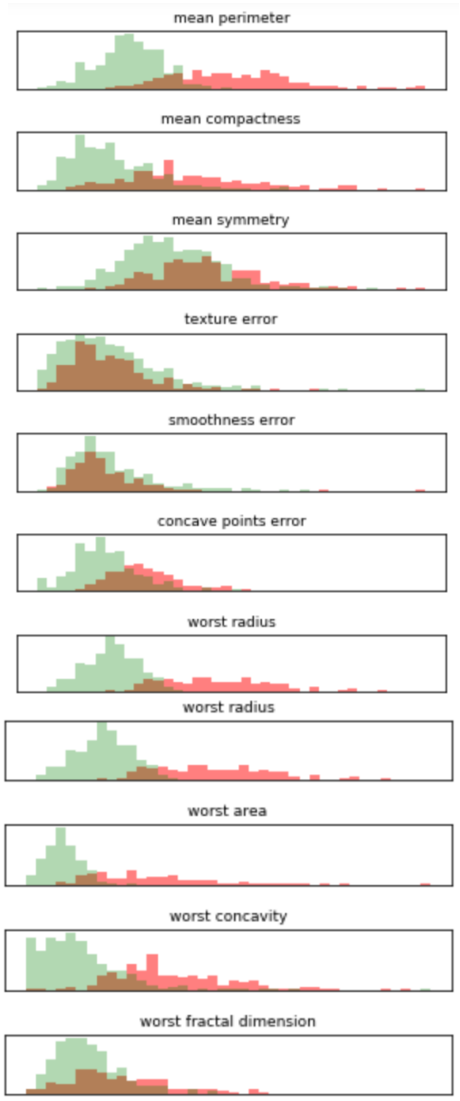


Figure 4: Finding the best features to divide the dataset into the two classes by maximizing distance between the 'red' and 'green' distribution of the two classes shown above.

D. Implementation of the Random forests algorithm:

ID3 algorithm is used to build each decision tree in the Random Forest algorithm. Variation of the trees is introduced by using bootstrapped samples of the training data to train each of the trees separately. The training data is then passed through each of the trees in the random forest and each votes on a particular class. The majority vote of all the trees in the forest is taken as the final decision for the predicted class. Various number of trees are used to construct various random forests and the percentage accuracy is shown below.

Number of Trees	Accuracy
N=5	85.97%
N=10	86.61%
N=20	93.66%
N=25	93.73%
N=50	94.36%

Table 1. Results of the Random Forest for different number of trees.

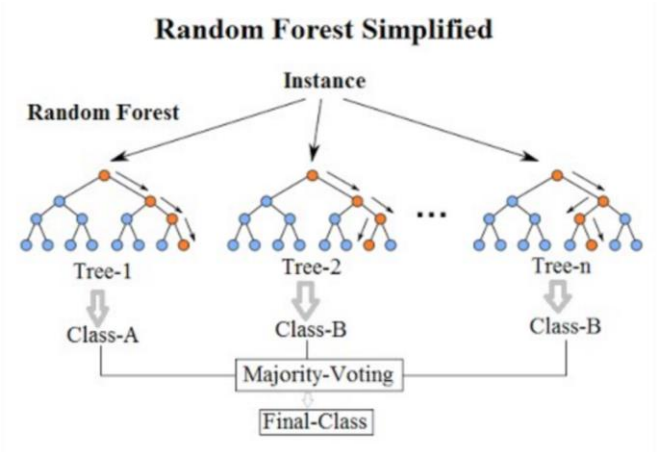


Figure 5. Structure of Random Forest

V. Random Forest Implementation With Scikit Learn

Random Forest is also implemented with the Scikit learn library for comparison. GridSearch is applied on this to find the set of parameters which produces the best accuracy.

The best features were found to be 'criterion': 'gini', 'max_depth': 20, 'max_features': 3, 'min_samples_leaf': 1, 'min_samples_split': 6, 'n_estimators': 5

The best accuracy obtained is 0.94 for 5 estimators.

VI. K Nearest Neighbours Implementation With Scikit Learn

K Nearest neighbours (KNN) is also implemented with the Scikit learn library for comparison. GridSearch is applied on this to find the set of parameters which produces the best accuracy.

'algorithm': 'auto', 'n_neighbors': 7, 'p': 1, 'weights': 'distance'

The best accuracy obtained is 0.94 for for number of neighbors = 7.

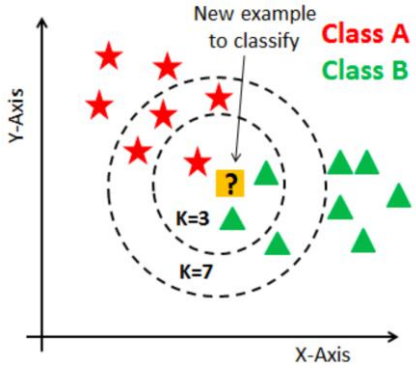


Figure 6. KNN segregation

RESULTS AND ANALYSIS

The following observations were made from the tables and implementations explained above:

1) The accuracy of the random forest generally increases with increase in the number of trees in the forest. However the increase in accuracy is small when compared to the total amount of time taken train the larger random forest. Highest results are consistent when number of trees is 50.

2) SciKits inbuilt models are more optimized and produce slightly better results at every stage. Hence the overall accuracy from Scikits models are usually better.

3) SciKits inbuilt models are more optimized and produce slightly better results at every stage. Hence the overall accuracy from Scikits models are usually better.

4) KNN and Random Forests both perform well for this dataset with a similar accuracy of 0.94.

REFERENCES

[1] https://medium.com/@madanflies/k-nearest-neighbour-for-classification-on-breast-cancer-data-results-with-preprocessing-and-w-o-e21b0cc98a2f

[2] https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d

[3] L. Breiman, L. Random Forests, Machine Learning, vol. 45 pp. 5–32, 2001.

[4] Robnik-Sikonja, M. Improving to Machine Learning, MIT Press, Cambridge, Massachusetts, 2nd Edition

[5] Breiman, Leo. Bagging predictors. Machine Learning. 1996, 24 (2): 123–140.