CHEMENG 787: Machine Learning: Classification Models

Comparing Classifiers

Instructor:

Dr. Jeff Fortuna

Submitted by:

Anirudh Ramesh Vijayameenakshi-400278584

Greeshma Gopal - 400245291

Objective

The expectation of this project is to implement classification models by choosing a dataset which has minimum of 1000 instances. 75% of the dataset were utilized to train the classifier and the remaining 25% was used for testing.

To compare the models, below parameters were calculated for the chosen dataset:-

- Computation times for test and train data
- Cross Validation accuracy scores
- Confusion Matrix of every model
- Receiver Operator Characteristics (ROC) Curve for every model

We have implemented the below models using the dataset to figure out the best one:-

- K nearest neighbors
- Naïve Bayes'
- Random forest
- Ada boost
- Decision Tree

To implement these models, we have also utilized the existing libraries numpy, sklearn and matplotlib in the python code.

Methodology

Dataset Description

The data set chosen is related to the detection of high energy seismic bumps in the coal mines. One of the major threats faced in these coaling mines are the seismic hazards. Seismic hazard is nothing but the probability of earthquake occurrence in a particular geographical area at a given time. Predicting the seismic hazard is highly significant to help make a decision in a given area of mine. If the high energy seismic bumps are expected to happen in a specific area its crucial to evacuate the crew from the endangered zones or stop the mining process. The data has been fetched from the UCI machine learning repository. The data acquisition is carried out at a geophysical station (Fig 1).

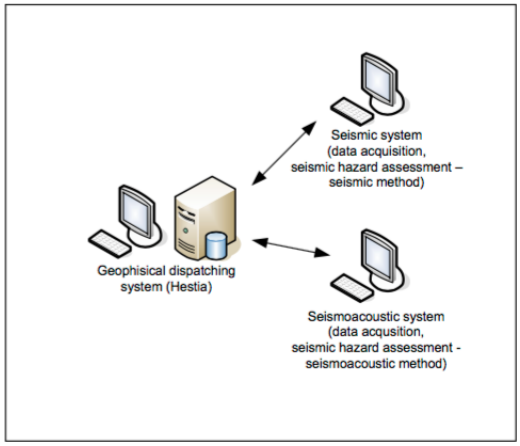


Fig 1: Data acquisition

There are 3604 data sets and 19 attributes associated with the same. The decision/target attribute is the 'CLASS' which is the supposed to be predicted. '1' means that high energy seismic bump occurred in the next shift ('hazardous state'), '0' means that no high energy seismic bumps occurred in the next shift ('non-hazardous state'). This data if predicted will aid in lowering the degree of hazard. The important factors affecting this hazard include the genergy(seismic energy recorded within previous shift by the most active geophone (GMax) out of geophones monitoring the longwall), gpuls(a number of pulses recorded within previous shift) and gdenergy(a deviation of energy recorded within previous shift by GMax from average energy recorded during eight previous shifts). Also the nbumps which is the number of the seismic bumps based on the energy range.

Data preprocessing was done by mapping some of the categorical data to corresponding numeric values before it was split to train and test data.

Results:

Confusion Matrix

Model	False positives	False Negatives	True Positives	True Negatives
KNN	0	82	292	527
Naïve Bayes	140	112	152	497
Random forest	0	26	292	583
Ada Boost	146	111	146	498
Decision Tree	0	53	292	556

Cross Validation Accuracy

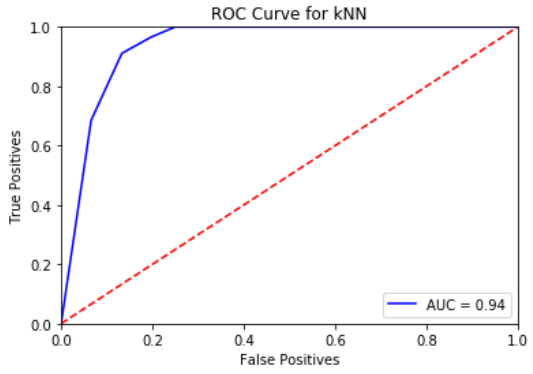
Model	Cross validation accuracies
KNN	95.2%
Naïve Bayes	67.9%
Random forest	99.9%
Ada Boost	86.9%
Decision Tree	95.6%

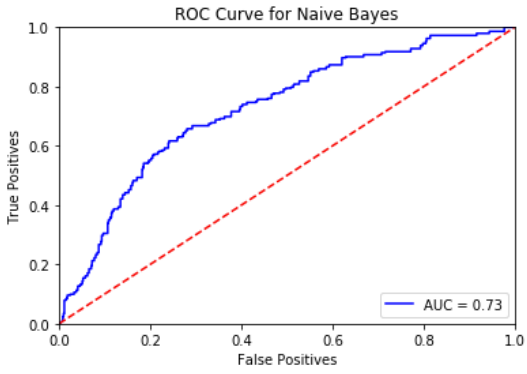
Computation time

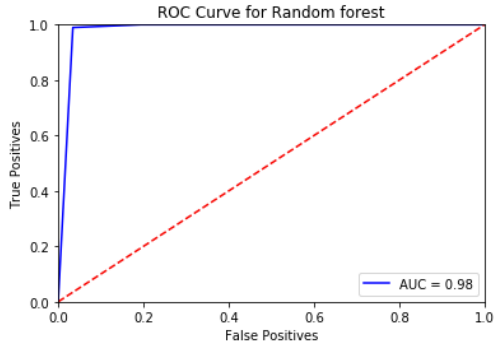
Model	Test time taken	Train time taken
KNN	0.071103683	0.006580947
Naïve Bayes	0.001271247	0.00682698
Random forest	0.006735898	0.036643038
Ada Boost	0.003385696	0.041445753
Decision Tree	0.00064987	0.01332349

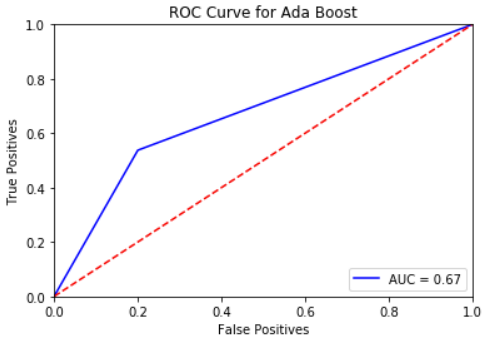
In general, the time taken for train data is more than the time taken for test data. However, when KNN was implemented, the time taken by train data is less than test data. This is because, the training phase of the algorithm consists only of storing vectors and class labels but in the test, a point is classified by assigning the label which are most frequent among the training samples.

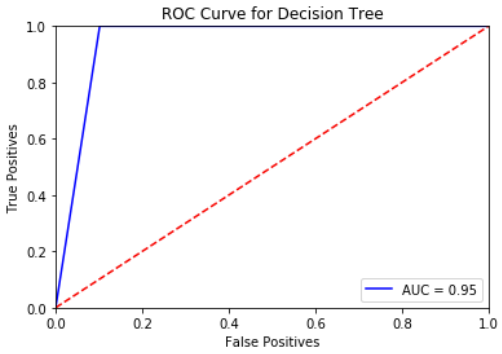
Receiver Operating Characteristics Curve (ROC):











Conclusion

We observed that KNN was giving a good score of 90%, however the cross validation score of KNN is 95%(mean of 5 cross validation scores). In the code, Random forest was also implemented(which was not asked but we tried implemented) and this gave the maximum scores. Apart from this, Decision tree can be considered as one of the best models for the given data set since the accuracy score is 94% and the cross validation accuracy is 95%. On the other hand, Naïve Bayes' gave the least accuracy 72%. With respect to Ada boost, the accuracy was 71% but with cross validation accuracy it was as high as 86%. Overall, it can be found that cross validation accuracies are better (for almost all classifiers) for this dataset than the actual accuracies.

Computational times were calculated for every model fit, and we found that the K nearest neighbors took the most time. The computational time was calculated for each model when fitting the test and train data. In KNN, it was observed that the test data computational time is higher than the train data. This is because train data is used to store data, while in test data it will classify the labels which are more frequent among training samples. However, in Random forest model fitting and Naïve Bayes' model fits the test time data is lesser than train data. The trend is the same for all the models during PCA and the backward selection. We can also observe that there is a slight increase in the computational time as the number of components increases in every classifier model fit. This is one of the major lessons learnt from this project.