Soft Computing Project Report

An Efficient Naïve Bayes Classifier with Negation Handling for Seismic Hazard Prediction

Submitted to:

Dr. Nagamma Patil

Assistant Professor

Department of Information Technology

NITK Surathkal

SACHIN – 15IT234
Ram Prakash –15IT115
Abhishek Kandukuri– 15IT214
Ketan Ramesh–15IT122

8th November 2017

Content:

List of Tables and figures

1. Introduction

Classification is the one of the most important techniques in Datamining for data analysis. In Datamining, different Classification Techniques are available to predict the outcome for a given dataset. There are many classification methods for predicting and estimating accuracy; one such famous method is Naïve Bayes Classifier. Naïve Bayes is very popular as it is easy to build, however to the assumption of conditional independence among predictor's results in loss of accuracy. In this project, we propose a technique to minimize loss of accuracy when predicting Seismic Hazard Activity. Hazard indicates a possible threat to life, health, property and environment. Mitigation of hazard when crossing stipulated level is paramount; otherwise, it may lead to an emergency.

Mineral, Diamonds/Gold and Coal exploration involves mining in a big way where hazard occurrence is quite common and addressing these mining hazards is a challenging task. An important threat of Mining Hazard is Seismic Hazard which is normal in underground mines. Thus Predicting Seismic Hazard is one of the most important aspect in countering Mining Hazards.

2. Literature Survey

2.1 Related Work

The classifier used in our model is the naïve Bayes classifier which assumes conditional independence between the attributes.
It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

There are three types of Naive Bayes model:

- **Gaussian:** It is used in classification and it assumes that features follow a normal distribution. It is used for continuous
- **Multinomial:** It is used for discrete counts. For example, let's say, we have a text classification problem. Here we can consider Bernoulli trials which is one step further and instead of "word occurring in the document", we have "count how often word occurs in the document",

you can think of it as "number of times outcome number x_i is observed over the n trials".

- **Bernoulli:** The binomial model is useful if your feature vectors are binary (i.e. zeros and ones). One application would be text classification with 'bag of words' model where the 1s & 0s are "word occurs in the document" and "word does not occur in the document" respectively.

The Bernoulli distribution could not be applied to the discrete data of our model since the discrete attributes have more than two discrete values hence the Gaussian and multinomial model were used for continuous and discrete data respectively.

2.2 Problem Statement

Use of negation handling technique to improve the accuracy of Naïve Bayes classifier.

2.3 Objectives

- To implement a working code of Naïve Bayes classifier that works on the given dataset.
- To implement negation handling to improve accuracy of the classifier.
- To analyze the results using the graphs.

3. Proposed Methodology

Implementation:

The stepwise implementation of the proposed model to minimize the loss of accuracy is as follows:

Step-1: Dataset in CSV is given as input and split into Training and Test datasets.
In this paper, a ratio of 60% and 40% is considered for Training and Test Sets.

Step-2: Differentiate the Training data set as per the Class values. i.e. 1, 2 & 3.
Step-3: Calculate the Mean and Standard Deviation for each data instance in the order of class values.
Step-4: Use the above values to calculate probabilities corresponding to class values using Gaussian Distribution Function.
Step-5: Generate probabilities for all attributes of a class belonging to Training set to the data instances of test dataset.
Step-6: If the resultant probability is '0' , after Step 5 for a particular data instance, which is not first in the list, then the mean value of preceding probabilities of the attributes is taken as the current probability. If the data instance is itself in the top of the list and probability is '0' then an equivalent value of '1' is added to attribute values of that particular data instance of the Training dataset.
Step-7: Generate Predictions by comparison between probabilities of data instances of each class values belonging to Training Dataset.
Step-8: Evaluate the accuracy of predictions by comparing with the class values of test dataset. The accuracy is computed regarding ratio between 0 to 100%.
The addition of Step-6 to the existing Naïve Bayes Classifier increases the accuracy by considering the probabilities of each data instance belonging to Training Set.

4. Result and Analysis:-

In this project, 60% of data for training and 40% of data for test is considered, which is normal for testing algorithm on a dataset.

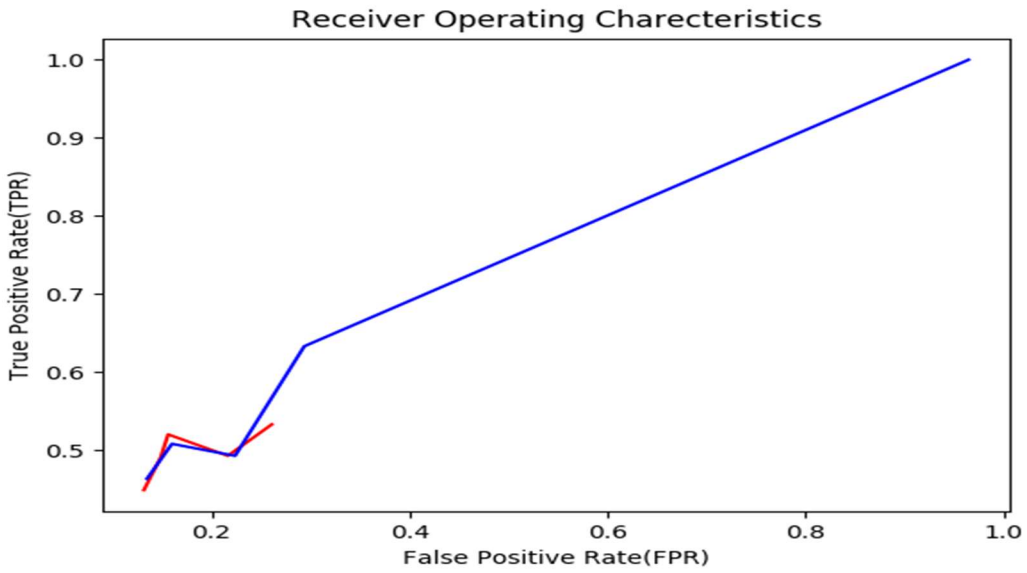

Figure 1

Blue line: Receiver operating characteristics without negation handling.
Red line: Receiver operating characteristics with negation handling.

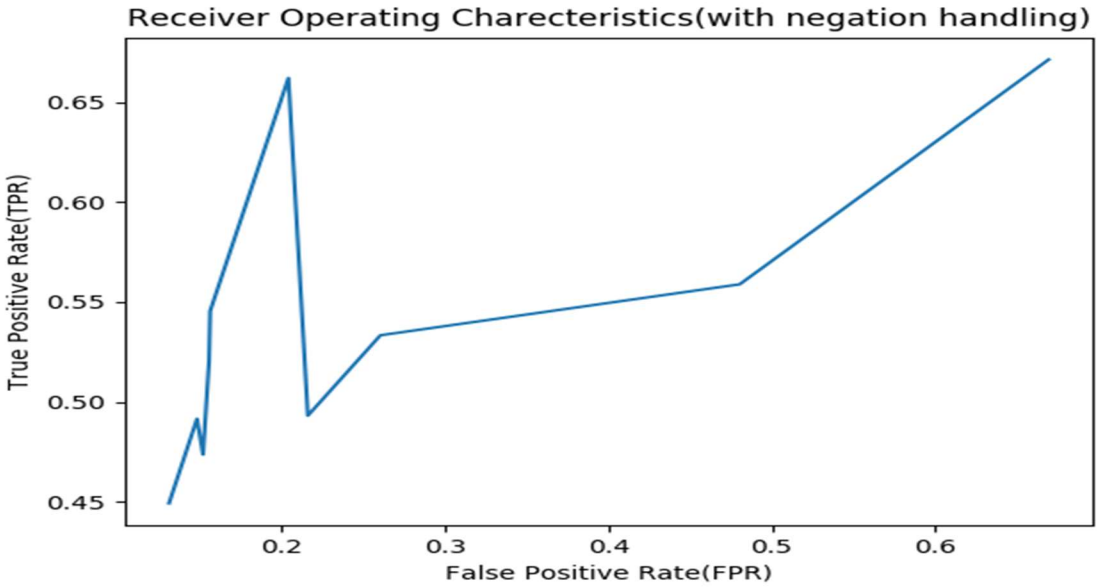

Figure 2

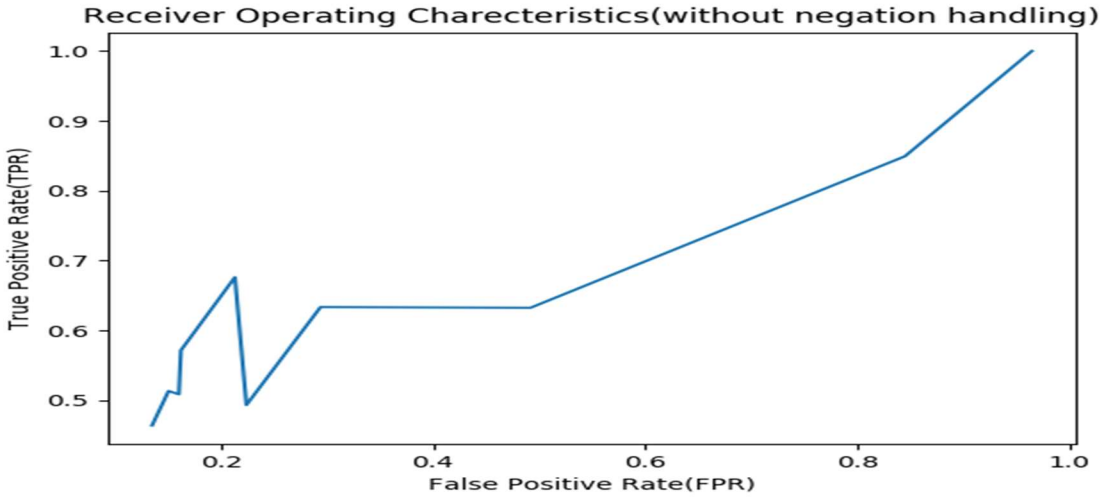

Figure 3

Confusion Matrix

Table 1

Predicted Class			
Actual Class		Hazard	Not-Hazard
	Hazard	35	37
	Not-Hazard	126	836

Accuracy without negation handling: 82.11%
Precision without negation handling: 0.16
Recall without negation handling: 0.51

Accuracy with negation handling: 83.075%
Precision with negation handling: 0.17
Recall with negation handling: 0.49

Analysis:

Applying negation handling increases the accuracy by

Table 2

Classifier	Training (60%)	Test (40%)	Accuracy
NBC with negation handling	1550	1034	83.075%
NBC without negation handling.	1550	1034	82.11%

Accuracy comparison chart with different classifiers with same ratios of Training and Test datasets

5. Conclusion and Future work

From the above results and graphs, the proposed Naïve Bayes Classifier algorithm used for predicting for Seismic Hazard, with 60%-40% ratio of Training and Test Datasets, yielded a good accuracy of 83.075% which is better when compared to the native Naïve Bayes Classifier. Experimental results indicate that using proposed NBC algorithm has significantly improved accuracy. Our future work will be exploring the feasibility of applying various distributions like Multinomial, Bernoulli, etc. and using other smoothing techniques to further improve the performance of Naïve Bayes Classifier.

6. References

1. https://www.irjet.net/archives/V3/i4/IRJET-V3I416.pdf
2. http://dataaspirant.com/2017/02/06/naive-bayes-classifier-machine-learning/
3. http://www.geeksforgeeks.org/naive-bayes-classifiers/