IJCSI International Journal of Computer Science Issues, Volume 13, Issue 5, September 2016
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org https://doi.org/10.20943/01201605.6975

Application of k-NN and Naïve Bayes Algorithm in Banking and Insurance Domain

Gourav Rahangdale[1], Mr. Manish Ahirwar[2] and Dr. Mahesh Motwani[3]

[1] Department of Computer Science & Engineering, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, Madhya Pradesh, India

[2] Department of Computer Science & Engineering, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, Madhya Pradesh, India

[3] Department of Computer Science & Engineering, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, Madhya Pradesh, India

Abstract

In today's globalized world, Business Intelligence software would form part of many firm's business oriented information technology strategies. Before such strategy could be planned, these firms would require in-depth data analysis on the products that they would be planning to sell such as sales purchases, staff costs and other items that could potentially impact services or goods. We have applied data mining classification technique in Banking and Insurance domain to provide solution and prediction model. In classification machine learning technique, we will use k-NN and Naïve Bayes algorithm on Portuguese bank dataset and Dutch insurance company dataset and shall also compare accuracy of their classification.

Keywords: *Data Mining, k-Nearest Neighbor, Naïve Bayes and Classification Algorithm*

1. Introduction

The volume of data stored in databases is expanding exponentially day by day very rapidly. This necessitates devising brand new strategies and tools that could help people and intellectual elite to naturally dissect large volume of data sets with useful and accumulate information. This would lead to conception of another examination field known as Data Mining or Knowledge Discovery in Databases (KDD), which is pulling considerable research largely in various fields including Database Design, Pattern Recognition, Statistics, Machine Learning and Information Visualization. In this section we intend to provide an introduction of Data Mining while at the same time explaining its methods, applications and more provisions and its tasks. Our inspiration in this investigation will be picking up the best systems for extracting suitable and useful information from a dataset. The objectives of this study are:

a) To provide scholars, developers and researchers an ideal predictive model using machine learning techniques inside such systems.

b) To assist organization for optimum utilization the resources.

2. Classification Techniques

Classification is a data mining task of predicting the value of a categorical variable (target or class) by building a model based on one or more numerical and/or categorical variables (predictors or attributes) [1]. In another words classification could be a data processing operation that assigns things during an assortment to focus on classes or categories. The main objective of classification is to correctly predict the target class for every situation within the knowledge. For example, in our case a classification model may be to verify that the candidates who want to subscribe to a bank debit scheme i.e. 'Y Label' in 'Yes' or 'No' in a bank dataset [2] and 'Number of mobile home policies' in '0s' and '1s' for an insurance company in an insurance dataset [3].

Classification could be a form of a supervised machine learning within which an algorithmic program "learns" to classify new observations with the help of their trained model from samples of a labelled knowledge. For more flexibility, we will be able to pass predictor or feature data with corresponding responses or labels to an algorithm fitting operate [4]. Classification is to isolate things into many predefined categories or classes provided to form an accumulation from claiming training samples. This sort of task is designed to seek out a model for class attributes as an operate of the values of different attributes.

In classification learning process are mainly divided into two parts which are as follows:

- Classifier Building or Training Model.
- Using Classifier Model for Classification or Test Classification.

Classifier Building or Training Model

Classifier Building or Training Model is the learning phase of the model where the classification algorithm is used for training the classifier and building it for further classification on test dataset. To build the classification model, training dataset is used where their attributes and labelled classes are pre-defined. Further the training model is ready using this training dataset and algorithm of classification is provided thereof.

Using Classifier Model for Classification or Test Classification

In 'Using Classifier Model for Classification or Test Classification' step, the trained model is used to classify the new data set which is called 'Test Dataset'. In this test data is used to determine the accuracy of classification. The accuracy of the algorithm which we have used in our classification technique will also be checked using this formula:

$$Accuracy = \frac{Ncc}{Tc} \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where,

Ncc = Number of Correct Classification,

Tc = Total Number of Test Cases

Classification Algorithms or Rules

There are lot of classification algorithms available for training model of which few are very popular and some are traditionally used which are explained below:

2.1 k-Nearest Neighbor (k-NN) Classifier

k- Nearest Neighbor may be a straightforward algorithmic rule that stores all offered cases and classifies new cases supporting similar measures (for e.g., distance functions). k-NN has been employed in applied statistical estimation and pattern recognition already since the beginning of 1970's as a non-parametric technique. In pattern recognition techniques, the k-Nearest Neighbor algorithm (or k-NN for short) is a non-parametric system utilized for classification and Regression [5]. In each cases, the input consists of the k nearest training examples within the feature area. The output depends on whether or not k-NN is employed for classification or Regression [6]:

- In k-NN classification, the output could be a category membership. Associate object is decided by a majority vote of its neighbors, with the neighbor being assigned to the class most typical among its k nearest neighbor (k could be a positive whole number, usually small). If k = 1, then also its object is just assigned the category of that single nearest neighbor.
- In k-NN regression, the output is that the property worth for the items. This worth is that the average of the values of its k nearest neighbor.

k- Nearest Neighbor is also called a lazy learner because it is an instance based learning type. In this kind of learning, whatever they perform is simply approximated regionally and every computation is postponed till classification. The k-NN algorithm rule is among the best of all machine learning algorithms.

The k-nearest neighbor algorithmic rule makes a classification for a given sample while not creating any assumptions regarding the distribution of the training and testing of datasets. Every testing sample should be compared to all the existing samples within the training model set so as to classify the sample. So whenever there is a choice to be made for utilizing this algorithm, first of all, the distances between the testing dataset and everyone in the samples within the training model set should be calculated. During this proposal, the Euclidean distance is calculated which is a default distance measurement technique, but in general, any distance measurement technique may be used as per choice. The Euclidean distance metric needs standardization of all options into identical range. At this time, the k nearest neighbor of the given sample are determined wherever k represents associate whole number is 1 between one and therefore the total number of samples. The k-nearest neighbor classifier could be a statistic classifier that's aforementioned to yield associate economic performance for optimum values of k.

2.2 Naïve Bayes Classifier

Naive Bayes is a classification technique which supported Bayes' Theorem with a presumption of independence among predictors. Simply we can say that, a Naive Bayes classifier considers the presence of a specific feature in a specific class, unrelated to the presence of the other feature [7]. Parenthetically, a fruit could also be thought of to be an Orange if it's color is orange (between red and yellow), shape is circular, and size being approximately three inches in diameter. Although these properties rely upon one another or upon the existence of the opposite options, all of these properties severally contribute to the likelihood that this fruit is an orange which is why it's called 'Naive' which means simple or unwary.

Naïve Bayes classification is a family of easy probabilistic classifier in machine learning technique which is based on Bayes theorem. From early 1960s studies on naive Bayes have intensified. The naive Bayes model is quite easy to prepare and is significantly useful for huge number of datasets.

The naive Bayes classifier is meant to be used once. Predictors used to measure must be free from one another among every class, however it seems to figure well in follow even once that independence assumption isn't valid. Naive Bayes classifier classifies data in 2 steps: one is training step and second is prediction step [4].

Bayes theorem provides the way of calculative posterior probability P(c|e) from P(c), P(e) and P(e|c). examine the equation below:

$$\mathbf{P(c|e)} = \frac{P(e \mid c) * P(c)}{P(e)} \quad \ldots\ldots\ldots\ldots\ldots\ldots (3)$$

$$\mathbf{P(c|E) = P(e_1|c) * P(e_2|c) *\ldots..* P(e_n|c) * P(c)} \quad \ldots\ldots\ldots\ldots\ldots\ldots (4)$$

Where,

P(c|e) = is the posterior probability of class (c, target or class) given (e, attributes or events).

P(c) = is the prior probability of class.

P (e|c) = is likelihood which is the probability of predictor given class.

P(e) = is the prior probability of predictor (event).

3. Proposed Work

In our proposed work, we will apply machine learning techniques in Portuguese bank datasets and Insurance dataset. The work flow chart is mentioned in Fig 1. which can show overview of steps taking place in our work.

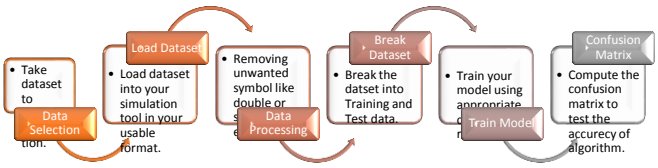


Figure 1. Steps of Proposed Work

a) Data Selection: Data selection is the first step where selection of the data from data warehouse is done for applying the process of data mining on it. We can select the Portuguese bank dataset and insurance dataset for data mining.

b) Loading Data: In data loading process we need to load the dataset into our data processing tool so that further operation on this dataset can be applied. We load the bank and insurance dataset into Matlab workspace.

c) Data Processing: Data processing is the very important phase, in this we can process the data and remove unnecessary things from data. In other words, we clean our data in this phase. In our case, we can remove the double and single quotes from bank and insurance dataset.

d) Variable Selection: In variable selection phase we can select our useful variables from existing dataset. From bank and insurance dataset we can select the variable for training model.

e) Model Selection: Model selection is the important phase in which we can select the appropriate algorithms or methods for training our model. We have used k-NN and Naïve Bayes algorithm.

f) Test Model: When our model is trained, then we can test that model in test or new data so that we can analyze the result. In our code we can test 8000 new records in our model for bank and 4000 for insurance dataset.

g) Diagnostic: This is the accuracy checking phase, in this part we can examine the predicted result of our model. We can use cross matrix for bank dataset predicted result.

3.1 Methodology

By using the machine learning technique, we can perform data mining in this dissertation on bank dataset for predicting labels or class of customers. In this we can use the two classification algorithm of machine learning. One is k- Nearest Neighbor and other is Naïve Bayes classifier. We will also compare the results of both the algorithm.

First of all, we need to load the dataset in Matlab workspace, then we need to pre-process the dataset like, removing unnecessary double quotes from certain attributes and convert all the categorical variables into nominal arrays., followed by breaking our dataset into two part, firstly for Training Data and secondly for Test data. We train our training model by using 32000 records for training dataset and we have used 8000 records dataset for testing purpose. The next step is use the classification algorithm for training the model. When the model is trained, we can use this model in testing our dataset after which we can compute the confusion matrix for checking accuracy of the algorithm. The final step will be to check accuracy of the predicted label.

3.1.1 K- Nearest Neighbor Classification Technique:

Nearest neighbor search locates the k closest observations to the specified data points, based on your chosen distance measurement. Available distance measures include Euclidean, Hamming, Mahalanobis, and more [4]. We have used the 'seucleadin' distance which is Standard Euclidean distance for K- Nearest Neighbor classifier in Matlab.

Steps: k-NN classification steps for bank dataset [8]

1) INPUT: bank and insurance dataset
2) OUTPUT: predictor

---

1) Load the dataset into workspace.
2) Pre-process the dataset.
   a) Remove unnecessary double quotes from certain attributes.
   b) Convert all the categorical variables into nominal arrays.
3) Break dataset into two parts.
   a) For Training Data.
   b) For Test Data.
4) Train the classifier using 'ClassificationKNN.fit' Matlab function.
5) Test the model using test dataset.
6) Compute the Confusion Matrix for examine.
7) Check Accuracy of Predicted Label/Class.

Note: Repeat steps 5 to 7 for test new dataset prediction.

3.1.2 Naïve Bayes Classification Technique:

Naive Bayes is a classification technique which supports Bayes' Theorem with a presumption of independence among predictors. Simply put, a Naive Bayes classifier considers that the presence of a specific feature during a class is unrelated to the presence of the other feature. To illustrate, a fruit is also thought of to be an apple if it's red, round, and measuring three inches in diameter. Although these properties depend upon one another or upon the existence of the other property, all of these properties severally contribute to the likelihood that this fruit is an apple which is why it's referred to as 'Naive'.

Naive mathematician model is straightforward to create and is significantly helpful for dissecting terribly big data sets. Together with simplicity, Naive Bayes is thought to outmatch even extremely subtle classification strategies. Naive Bayes is also Called a weak learner.

Bayes theorem provides the simplest way of shrewd posterior probability P(c|e) from P(c), P(e) and P(e|c).

Algorithm: Naïve Bayes classification algorithm [9]

INPUT: bank and insurance dataset
OUTPUT: predictor

```
    case 'normal'
1)  for i=1:num_of_class
        class_prob(i)=sum(double(Ytrain==Ytra
in_unique(i)))/length(Ytrain);
    end

2)  for i=1:num_of_class

        x=Xtrain((Ytrain==Ytrain_unique(i)),:
        );
        mu(i,:) = mean(x,1);
        sigma(i,:) = std(x,1);
    end
    case 'kernel'
3)  for j=1:len_test
        test_prob=normcdf(ones(num_of_class,1
)*Xtest(j,:),mu,sigma);

        Prob(j,:)=class_prob.*prod(test_prob,
        2)';
    end

4)  [pred_Ytest0,id] = max(Prob,[],2);
        for i=1:length(id)
        pred_Ytes(i,1) = Ytrain_unique(id(i));
        end

    'Confusion Matrix & Accuracy'
5)  confMat=myconfusionmat(Ytest,pred_Ytest);
    conf=sum(pred_Ytest==Ytest)/length(pred_
Ytest);
    (['accuracy = ',num2str(conf*100),'%'])
```

We have applied both k-NN and Naïve Bayes classification algorithm in both bank and insurance dataset and also compare both of them.

4. Implementation and Results

In our work, we have used Matlab simulation tool which is a strong numeric engine providing programming atmosphere with interactive tools for applied mathematics analysis, image process, signal process, data mining and alternative domains for implementing machine learning classification techniques, wherein we will use k-NN and Naïve Bayes algorithm for class perdition.

4.1 Dataset Description

In this work, two datasets have been used out of which one is bank [2] and another is insurance [3]. The brief description of datasets is given below.

a) Bank Dataset: The bank dataset which we have used for classification have following attributes: Age, Job, Marital Status, Education, Default, Balance, Housing, Loan, Contact, Day, Month, Duration, Campaign, Pdays, Previous, Poutcomes and Y labels. More details of dataset are mentioned in the table.
b) Insurance Dataset: In insurance dataset, there are total 86 attributes in all. In this dataset, all the customers are living same zip code area with similar socio-demographic attributes. Our label is attribute 86th, 'Number of mobile home policies' is the target variable.

4.1.1 KNN and Naïve Bayes Classifier Outcomes for Bank Dataset

k-NN Classifier

In k-NN classification for bank dataset, we have trained our model using 32000 training records and applying k-NN rule in this training set, following which the classification model becomes ready to predict the test dataset. In this dataset, we have selected a total of 8000 records for test dataset in which total actual 'no' label counts are 7000 (i.e. 87.50%) and total 'yes' label counts

are 1000 (i.e. 12.5%). Predicted counts of 'no' and 'yes' label is 7260 (i.e. 90.75%) and 740 (i.e. 9.25%).

The test dataset which we have used to classify into 'no' and 'yes' class is 8000 and our training model accuracy to classify these 8000 records is 83.9% for correct classification and 16.1% for misclassification.

And out of total 1000 'Yes' (in actual) it correctly classifies 226 (out of 1000 of 'Yes') into 'No' and misclassifies 774 (out of 7000 of 'Yes').

Naïve Bayes Classifier

Similarly, k-NN classifier, in Naïve Bayes classification we will train our model using 32000 training records and apply Naïve Bayes rule in this training set, following which our classification model would be ready to predict the test dataset. In this dataset, we have selected a total 8000 records in test dataset in which total actual 'no' label counts are 7000 (i.e. 87.50%) and total 'yes' label counts are 1000 (i.e. 12.5%). Predicted counts of 'no' and 'yes' label is 7260 (i.e. 90.75%) and 740 (i.e. 9.25%).

The test dataset which can be used to classify into 'No' and 'Yes' class is 8000 and our training model accuracy to classify this 8000 records is 88.4% for correct classification and 11.6% for misclassification.

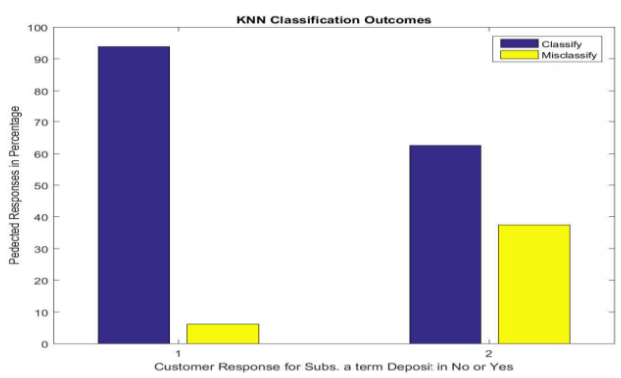
And out of total 1000 'Yes' (in actual) it correctly classifies 235 (out of 1000 of 'Yes') into 'No' and misclassifies 765 (out of 7000 of 'Yes').

Table 1. shows the number of predicted and actual class in 'no' and 'yes' for both k-nearest neighbor and naïve Bayes classifier for bank dataset.

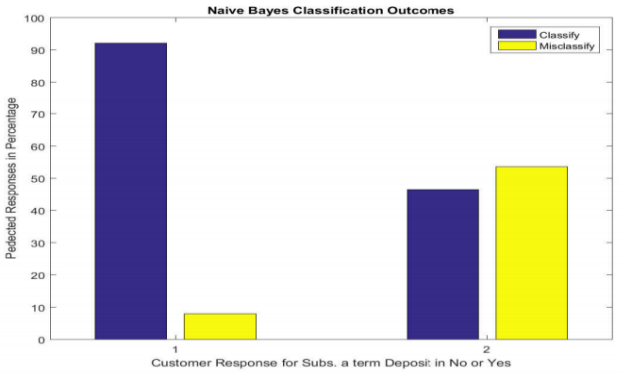
k-NN Classifier Prediction		
	Predicted 'NO'	**Predicted 'Yes'**
Actual 'NO'	6486	514
Actual 'YES'	774	226
Naïve Bayes Classifier Prediction		
	Predicted 'NO'	**Predicted 'Yes'**
Actual 'NO'	6837	163
Actual 'YES'	765	235

Table 1. k-NN and Naïve Bayes Classifier Prediction for Bank Dataset

Figure 2. shows, in (a.) k-NN model out of total 7000 'No' (in actual) it correctly classifies 6486 (out of 7000 of 'No') into 'No' and misclassifies 514 (out of 7000 of 'No') and in (b.) Naïve Bayes model out of total 7000 'No' (in actual) it correctly classifies 6837 (out of 7000 of 'No') into 'No' and misclassifies 163 (out of 7000 of 'No').



a) Bar Graph for k-NN Classifier Result



b) Bar Graph for Naïve Bayes Classifier Result

Fig. 2. Result of k-NN and Naïve Bayes Classifier for Bank Dataset

So, in this case according to result of bank dataset Naïve Bayes classifier gives us more accuracy as compared to k-NN classifier as k-NN provides 83.9 % accuracy and Naïve Bayes provides 88.4 % accuracy.

4.1.2 KNN and Naïve Bayes Classifier Outcomes for Insurance Dataset

k-NN Classifier

In k-NN classification for insurance dataset, we have trained our model using 5822 customer training records and applying k-NN rule in this training set, following which the classification model becomes ready to predict the test dataset. In this dataset we have selected total 4000 records for test dataset in which total actual '0s' label counts are 3743 (i.e. 93.58%) and total '1s' label counts are 257 (6.42%). Predicted counts for 'Number of mobile home policies' in '0s' and '1s'.

The test dataset which we have used to classify into '0s' and '1s' class is 4000 and our training model accuracy to classify this 4000 records is 89.47% for correct classification and 10.53% for misclassification.

And out of total 257 '1s' (in actual) it correctly classifies 37 (out of 257 of '1s') into '1s' and misclassifies 201 (out of 257 of '1s').

Naïve Bayes Classifier

Similarly, k-NN classifier, in Naïve Bayes classification on insurance dataset we have trained our model using 5822 customer training records and applying Naïve Bayes rule in this training set, following which classification model becomes ready to predict the test dataset. In this dataset we have selected a total of 4000 records for test dataset in which total actual '0s' label counts are 3743 (i.e. 93.58%) and total '1s' label counts are 257 (i.e. 6.42%). Predicted counts for 'Number of mobile home policies' in '0s' and '1s'.

The test dataset which we have used to classify into '0s' and '1s' class is 4000 and our training model accuracy to classify this 4000 records is 90.62% for correct classification and 9.38% for misclassification.

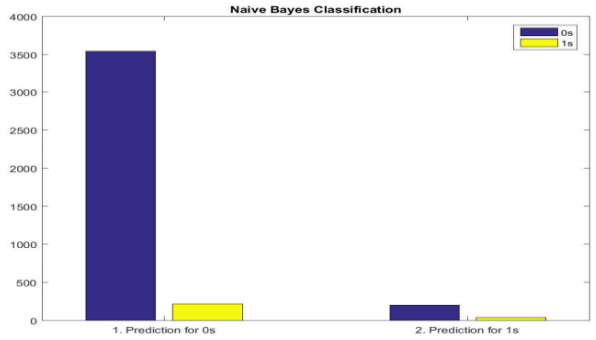
And out of total 257 '1s' (in actual) it correctly classifies 54 (out of 257 of '1s') into '1s' and misclassifies 184 (out of 257 of '1s').

Table 2. shows the number of predicted and actual class in '0s' and '1s' for both k-nearest neighbor and naïve Bayes classifier for Insurance dataset.

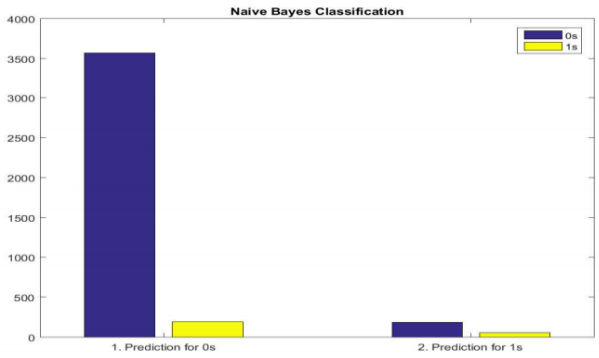
k-NN Classifier Prediction		
	Predicted '0s'	**Predicted '1s'**
Actual '0s'	3542	220
Actual '1s'	201	37
Naïve Bayes Classifier Prediction		
	Predicted '0s'	**Predicted '1s'**
Actual '0s'	3571	191
Actual '1s'	184	54

Table 2. k-NN and Naïve Bayes Prediction of Class for Insurance Dataset

Figure 3. shows, in (a.) k-NN model out of total 3743 '0s' (in actual) it correct classifies 3542 (out of 3743 of '0s') into
'0s' and misclassify 220 (out of 3743 of '0s') and in (b.) Naïve Bayes model out of total 3743 '0s' (in actual) it correct classifies 3571 (out of 3743 of '0s') into '0s' and misclassify 191 (out of 3743 of '0s').



a) Bar Graph for k-NN Classifier Result



b) Bar Graph for Naïve Bayes Classifier Result

Fig. 3. Result of k-NN and Naïve Bayes Classifier for Insurance Dataset

So, similarly as in case of bank dataset, Naïve Bayes classifier provides more accuracy as compared to k-NN classifier in insurance dataset also as k-NN provides 89.47% accuracy and Naïve Bayes provides 90.62% accuracy.

5. Conclusion

Organizations like Banks, Insurance and Finance service providers have large records of their business history in the form of their own database which are stored in their data warehouses and the size of this data are continuously increasing with every passing day. Therefore, it is imperative for an organization to adopt data mining because through this, data can be collected & stored at enormously high speeds (GBs/hour) from different sources and traditional techniques are infeasible for processing such large amount of raw data. So, by adopting proper data mining techniques, they can identify the patterns, trends and relationship from that data which can help them to oversee, forecast and analyze their business strategies. In my dissertation work, we have classified the bank dataset using machine learning techniques (K-Nearest Neighbor and Naïve Bayes algorithm) for data mining and in both datasets - Naïve Bayes provides more accurate results as compared to k-Nearest Neighbor algorithm.

6. References

[1] S. Sayad, "Data Mining," June 2010. [Online]. Available: http://www.saedsayad.com. [Accessed Feb 2016].

[2] S. Moro, P. Cortez and P. Rita, "UCI Machine Leraning Repository: Bank Marketing Data Set," Feb 2012. [Online]. Available: http://mlr.cs.umass.edu/ml/datasets/Bank+Marketing. [Accessed May 2016].

[3] P. V. D. Putten, "UCI Machine Learning Respository: Insurance Company Benchmark (COIL 2000) Data Set," Sentient Machine Research, October 2012. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Insurance+Company+Benchmark+(COIL+2000). [Accessed June 2016].

[4] Matlab, "Matlab Documentation - MathWorks India," MathWorks, [Online]. Available: http://in.mathworks.com/help/matlab/. [Accessed April 2016].

[5] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," The American Statistician, vol. XLVI, no. 3, pp. 175-185, 1992.

[6] T. F. E. Wikipedia, "k-Nearest Neighbors Algorithm," [Online]. Available: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm. [Accessed June 2016].

[7] S. Ray, "6 Easy Steps to Learn Naive Bayes Algorithm," 13 September 2015. [Online]. Available: https://www.analyticsvidhya.com/blog/2015/09/naive-bayes-explained/. [Accessed April 2016].

[8] Sonots, "Matlab CVPR toolbox / Code / [r297]: cvKnn.m," Sourceforge, [Online]. Available: https://sourceforge.net/p/cvprtoolbox/code/HEAD/tree/cvKnn.m. [Accessed May 2016].

[9] I. Biswas, "File Exchange - Matlab Central," MathWorks, August 2012. [Online]. Available: http://in.mathworks.com/matlabcentral/fileexchange/37737-naive-bayes-classifier/content/NaiveBayesClassifier.m. [Accessed February 2016].