International Journal of Mathematics and Statistics Invention (IJMSI)
E-ISSN: 2321 – 4767 P-ISSN: 2321 - 4759
www.ijmsi.org Volume 11 Issue 1 || Jan. - Feb., 2023 || PP 50-54

Comparison of Classification Techniques

SripriyaMuppidi and Bhatracharyulu N.Ch.
muppidisripriya@gmail.com and dwarakbhat@osmania.ac.in
Department of Statistics, University College of Science,
Osmania University, Hyderabad-7, TS, India.

Abstract
In this paper an attempt is made to compare the classification techniques, linear Discriminant, K-nearest Neighbourhood, Perceptron learning, Naïve Bayes Classifier, Logistic regression. The comparison made is with respect to their methods, merits, and demerits. The methods were implemented for a credit card bank data set and evaluated their accuracy and found that perceptron learning neural network algorithm have more accuracy among them.

Key words*: Machine learning techniques, Credit Card, supervised learning.*

Date of Submission: 13-02-2023 Date of Acceptance: 26-02-2023

I. INTRODUCTION

A classifier partitions the feature space into K disjoint subspaces, C_1, C_2, ... , C_K, such that for a sample with expression profile $\mathbf{X} = (x_1, x_2, \ldots, x_p) \in C_k$ the predicted class is k. Classifiers are built from a learning or training set: L = ($\mathbf{X}_1$, Y_1), ..., ($\mathbf{X}_n$,Y_n); where $Y_i \in \{C_1, C_2, \ldots, C_K\}$ Classifier D built from a learning set L: D(. , L): $\mathbf{X} \rightarrow \{C_1, C_2, \ldots, C_K\}$. Predicted class for observation $\mathbf{X}$: D ($\mathbf{X}$, L) = C_K if $\mathbf{X}$ is in C_K.

II. CLASSIFICATION TECHNIQUES

Several researchers contributed in developing different classification algorithms.In this section, the methods of popular classification techniques were presented.

METHOD-1: Fisher (1936) developed a discriminant function for classifying the different species of Iris setosa and Iris versicolor having more than two features. It is a multivariate statistical technique used to separate the two or more populations by constructing a boundary function between the populations.

Step-1: Let T = $\{(X_i, Y_i), i = 1, 2, \ldots, n\}$ be the sample of training data set, measured on feature space X in 'p' dimensions. Let X_0 be the sample observation to be classified to one of the class. Let D = b'X be the discrimination function that classifies the population into two or more classes (where b′ be the vector of discriminant score).

Step-2: The vector b′ of discriminant scores can be obtained by maximizing λ = b'Bb / b'Wb where B = $\Sigma\Sigma$ $(X_i - X)$ $(X_i - X)'$ and W = $\Sigma\Sigma$ $(X_{ij} - X_i)$ $(X_{ij} - X_i)'$..

Step-3: Obtain eigen values λ using the characteristic equation $|W^{-1}B - \lambda I| = 0$. and b is the eigen vector corresponding to λ. The resulting discriminant function is D= b'X.

Step-4**:** Let X_0 be an observation to be allocated to one of the populations. Allocate X_0 to i^{th} population if $|b'X_0 - b'X_i| < |b'X_0 - b'X_l|$.

METHOD-2:Cover and Hart(1967)developed Nearest Neighbours algorithm that searches the pattern for the k-training tuples that are closest to the unknown tuple. The closeness is defined in terms of distance metric. The detailed method is presented below.

Step-1: Let T = $\{(X_i, Y_i), i = 1, 2, \ldots, n\}$ be the trained data set measured on feature space in 'p' dimensions and 'm' be the number of labeled classes that sample data set T is classified. Let X_0 be the sample observation to be classified to one of the class. Let *k* is a parameter, the value of which will be determined by minimizing the cross-validation error

Step-2: Find the shortest distance between the sample observation X_0 to the 'k' closest neighbor observations in the data set T.

Step-3:Choose the class that has majority voting among those 'k' neighbors, Let'k' is a parameter, the value of which will be determined by minimizing the cross-validation error later.

METHOD-3:Rosenblatt (1957) developed an artificial neuron model used to classifythe sample observation, called perceptron learning and explained the concept behind it.
Step-1: Let T = $\{(X_i, Y_i), i = 1, 2, \ldots, n\}$ be the training sample data set measured on feature space in 'p' dimensions. Assume Y be the vector of desired binary output response.
Step-2: Initially choose a random vector of weights $W^{(1)}$ in p-dimension and choose a learning rate constant η ($0 \leq \eta \leq 1$).
Step-3: For each input vector Xi, evaluate $O_i = \text{Sign}\,[W^{(i)}X_i]$.
Step-4: Update the weight vector using the relation

$$W^{(i+1)} = W^{(i)} + \eta*(Y_i - O_i).\ Xi$$

Step-5: Repeat the steps 3 and 4 'n' times (until all training samples are over)

METHOD-4:Cortes andVapnik (1995) presented a training algorithm that maximizes the margin between the training patterns and the decision boundary in support vector machines, which is used to classify the given observation to one of the label classlike as follows. Let x be the feature space. Let $y_i \in \{-1,1\}$ denote its class label for binary response.
Step 1: Let $X = (X_1, X_2, \ldots, X_p)'$ be the feature vector and Y be the binary response variable. Let $T = \{(X_i, Y_i), i = 1, 2, \ldots, n\}$ be the trained data set measured on feature space in 'p' dimensions and 'm' be the number of labeled classes that sample data set T is classified. Let X_0 be the sample observation to be classified to one of the classes.
Step-2:The decision boundary of a linear classifier can be expressed as W.X+b=0,where W and b are parameters of the model.The margin is given by two parallel hyperplanes that are separated by the maximum possible distance 2/||w|| with no data points inside the margin. Then the hyperplanes based on training data set can be constructed as $W.X_i + b \geq 1$ for X_i in the +1classand $W.X_i + b \leq -1$ for X_i in the -1 class.
Step-3: Consider the optimization model, Minimize: $Z = \frac{\|W\|^2}{2}$Subject to the constraints $y_i(w.\phi) + b) \geq 1$. The Lagrangian function for the optimization problem is

$$L = \frac{1}{2}W.W - \sum_{i=1}^{N} \alpha_i[y_i(W.X_i + b) - 1]$$

Where the parameters $\lambda_i$are called the Lagrange multipliers. To minimize the Langrangian function,we can obtain the Karush-Kuhn-Tucker conditions as

$$\frac{\partial L}{\partial W} = W - \sum_{i=1}^{N} \propto_i y_i x_i = 0 \text{ and } \frac{\partial L}{\partial b} = \sum_{i=1}^{N} \propto_i y_i = 0$$

$$\lambda_i \geq 0, \text{ and } \lambda_i[y_i(W.X_i + b) - 1] = 0$$

Step 4:The dual formulation is maximizing over the Karush- Kuhn-Tucker multipliers $\lambda_i = (\lambda_1, \lambda_2, \ldots, \lambda_N)$ the function:

$$L_D = \sum_{i=1}^{l} \lambda_i - \frac{1}{2}\sum_{i,j} \lambda_i \lambda_j y_i y_j x_i x_j$$

Step 5: The decision boundary can be expressed as follows

$$\sum_{i=1}^{N} \lambda_i y_i x_i x + b = 0$$

Step-6:$f(X_0)$= Evaluate $\text{sign}(W.X_0+b)=\text{sign}(\sum_{i=1}^{N} \propto_i y_i x_i.x_0 + b)$.$X_0$can be classified aspositive class if $f(x_0)$= +1, and negative class if $f(x_0)$= -1,

METHOD-5:Naive Bayes classifier algorithm is used to classify the given sample observation based on data set is presented below.
Step-1:Let $X = (X_1, X_2, \ldots, X_p)'$ be the feature vector and Y be the binary response variable.Let $T = \{(X_i, Y_i), i = 1, 2, \ldots, n\}$ be the trained data set measured on feature space in 'p' dimensions and 'm' be the number of labeled classes $C_1, C_2, \ldots, C_m$ that sample data set T is classified. Let X_0 be the sample observation to be classified to one of the class.
Step-2: evaluate the Prior probabilities $P(C_i)$ and likelihood probabilities $P(C_i/X)$ based on the training data set.Evaluate the posterior probability

$$P(X/C_i) = \frac{P(C_i/X).P(C_i)}{P(C_i)} \qquad \text{for each class } C_i.$$

Step-3:The object or instance 'x' is allocated to class $C_i$iff

$$P(X/C_i).P(C_i) > P(X/C_j).P(C_j) \text{ for } 1 \leq j \leq m,\ j \neq i.$$

the predicted class label is the class C_i for which $P(X/C_i).P(C_i)$ is the maximum.

METHOD-6: The Logistic regression model can be used as a classifier is presented below.
Step-1: Let X = (X_1, X_2, …, X_p)' be the feature vector and Y be the binary response variable, Let T = {(X_i, Y_i), i = 1, 2, …, n} be the trained data set measured on feature space in 'p' dimensions and 'm' be the number of labeled classes that sample data set T is classified. Let X_0 be the sample observation to be classified to one of the classes.
Step-2: Assume the functional relationship between the variables is

$$Y = \frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}$$

Estimate the vector of parameters $\beta = (\beta_0\ \beta_1\ \beta_2 \ldots \beta_p)$ using least square method.
Step 3: Now optimize the β values by using $\beta_j = \beta_j + \alpha * (y - p) * x_j$ where α is the learning rate, p is the prediction of the model and x_j is the input value
Step 4:X_0 can be classified based on the decision rule: 0 if p<0.5; and 1 if p>0.5

III. COMPARISON OF VARIOUS CLASSIFICATION TECHNIQUES

In this section an attempt is made to compare the various classification methods considered in section 2.

Table 3.1: Merits and demerits of Classification methods

Method	Merits	Demerits
Discriminant Model	**1.** It is a parametric supervised learning classifier, finds the vector which maximizes the separation between classes and builds a model. 2. It uses the mean values of the classes and maximizes the distance between them. 3. It minimizes the variance in the dataset by reducing the number of features. 4. It can be used for both binary and multi class problems.	1. It makes assumption about feature vector follows normal. 2. It is based on the variance-covariance matrix of the data set.
K-Nearest Neighbor	1. It is a non-parametric supervised learning, butdoesn't build any mathematical model but classifies the given sample point to one of the labeled classes based on 'k' nearest neighbors with majority voting. 2. It uses the distance metric, Euclidean, Manhattan, Minkowski etc. 3. It is a lazy learning and the accuracy is used to find the number of nearest neighbors. 4. Root means square error is used to find the number of nearest neighbors.	1. Choosing the value of k is difficult task. 2. Computational cost is high for high dimensional data. 3. It doesn't perform well on imbalanced dataset.
Perceptron	1. A Perceptron maps an m-dimensional input vector, onto a n-dimensional output vector. 2. A distinct feature of a Perceptron is that the weights are not pre-calculated as in a McCulloch-Pitts model of neuron but are adjusted by an iterative process. 3. Activation function is used to convert the input vector into a useful output. 4. It doesn't make assumptions about input feature vector. 5. A Multilayer perceptron can be used to solve complex non-linear problems and as the number of hidden layers increased the accuracy also increased.	1. Training a multilayer perceptron is usually time consuming. 2. It is difficult to predict how much the dependent variable affects each independent variable.
Support Vector Machine	1. It is a supervised optimization learning technique with a quadratic objective function and linear constraints. 2. It uses nonlinear programming approach for its solvation. (ThroughLagrange function) 3. It constructs the optimal hyper-plane that classifies the original feature space into two sets by minimizing the squared norm of the separating hyper-plane. 4. Number of support vectors will be identified. The hyperplane is chosen in such a way that it maximizes the margin, which is the distance between the hyperplane and support vectors.	1. It is computationally expensive for large datasets due to quadratic optimization problem. 2. Choice of kernel greatly affect the performance of SVM. 3. Tuning the hyper parameters is difficult.
Naïve bayes Classifier	1. It is a supervised nonparametric probabilistic classifierworks based on Bayes theorem. 2. It calculates posterior probabilities based on the prior information. Itassumes the independence of feature vector. 3. It doesn't build any mathematical model rather than posterior probabilities.	1. It assumes all the features are independent but it fails sometimes. 2. Not suitable for continuous feature vectors.
Logistic	1. It is a non-parametric supervised learning classification technique. 2. It constructs the linear decision boundary. 3. Sigmoid function $f(x) = \frac{1}{1+e^{-x}}$ is used to convert the	1. It assumes the linearity between dependent and independent variables. 2. Due to linear decision boundarynon-linear problems cannot be solved.

	predictions into probabilities. 4. It provides the difference in the percentage of dependent variable and provides the rank of individual variable according to its importance. 5. It is used to classify the low dimensional data having nonlinear boundaries,	3. Number of observations should be greater than the number of feature vectors otherwise it leads to overfitting.

EXAMPLE 3.1: A data set of 13,444 customers of a national bank with afeature vector **X**= (X_1, X_2, … X_{12}) of 12 attributes under study with a categorical response variable Y of the status of credit-card is considered. For the classification, 80% (10,755) of the data is used for training the model and 20% (2,689) of the data set is used for testing the model and its accuracy.

Table 3.2: Description of variables under study

Variable	Variable description	Label
Y	Credit card status	Credit card status
X_1	Age	Age
X_2	Months living at current address	ACADMOS
X_3	1+No of dependents	ADEPCNT
X_4	No. of Major Derogatory reports	MAJORDRG
X_5	No. of Minor Derogatory reports	MINORDRG
X_6	Own-rent	Own-Rent
X_7	Income	Income
X_8	Self-employed	Self-employed
X_9	Income divided by number of dependents	INCPER
X_{10}	Ratio of monthly credit card expenditure to yearly income	EXP_INC
X_{11}	Average monthly credit card expenditure	SPEND
X_{12}	Log of Spending	LOGSPEND

Sample data Set:

Table 3.3 Sample data set of Response (Y) & Feature vector (X) with data values

Y	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}
…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…
1	27.66667	16	1	0	0	0	1650	0	23900	0.0953847	157.384808	5.0586938
0	33.75	18	0	0	0	1	1833.333	1	31000	3.87E-04	17.5041667	3.7810954
1	25.91667	54	0	1	1	1	1918	0	23016	0.169264	324.64833	5.7827425
1	38.58333	24	1	0	0	1	5000	0	30000	0.0202707	101.353331	4.6186127
0	42.41667	2	3	0	0	0	2916.667	0	8750	3.43E-04	69.3808347	4.6803215
1	43.25	118	4	0	1	1	3333.333	0	9000	2.67E-04	0.8888889	-0.117783
0	42	36	2	0	0	1	1250	0	6600	6.06E-04	8	0.7731899
1	40.58333	36	0	0	0	1	3000	0	42000	0.042216	126.647852	4.8414104
1	28.83333	26	0	0	0	1	2000	0	24500	0.0387955	77.5910243	4.3514518
1	26.33333	24	0	0	0	1	866.6667	0	18720	0.1779535	154.22639	5.0384216
…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…

Comparison with respect to the existence of the model, confusion matrix, accuracy, mis-classification rate evaluated using R- programming are summarized in the following table.

Table 3.4: A comparison w.r.t Model, Confusion matrix, Accuracy, Mis-classification rate

Classification Method	Classification Model	Confusion matrix	Accuracy	Rate of Mis-classification
Linear Discriminant	Linear Model	$\begin{pmatrix} 287 & 103 \\ 291 & 2008 \end{pmatrix}$	0.8535	0.1465
K-Nearest Neighbour	No model	$\begin{pmatrix} 348 & 90 \\ 242 & 2009 \end{pmatrix}$	0.8765	0.1235
Perceptron Learning	No model	$\begin{pmatrix} 562 & 6 \\ 8 & 2113 \end{pmatrix}$	0.9948	0.0052
Support Vector Machine	Linear model	$\begin{pmatrix} 576 & 18 \\ 18 & 2077 \end{pmatrix}$	0.9866	0.0134
Naïve Bayes Classifier	Probabilities	$\begin{pmatrix} 572 & 108 \\ 6 & 2003 \end{pmatrix}$	0.9576	0.0424
Logistic Regression	Logistic model	$\begin{pmatrix} 580 & 12 \\ 11 & 2086 \end{pmatrix}$	0.9914	0.0086

IV. CONCLUSION

It can observe that some classification methods provide mathematical linear functions to classify the given sample observations. It can be observed that for the chosen data set perceptron neural network learning technique provides 99.48% accuracy for the testing data set and logistic regression provides 99.14% accuracy.

REFERENCES

[1]. Cortes, C. and Vapnik, V. (1995): "Support vector networks", Machine Learning, Vol 20, pp 273-297.

[2]. Cover, T.M.and Hart, P. E. (1967): "Nearest neighbor pattern classification", IEEE Transactions on Information Theory, Vol 13 *(*1): pp 21–27.

[3]. Fisher R.A. (1936): "The Use of Multiple Measurements in Taxonomic Problems" (PDF). *Annals of Eugenics*. Vol **7**(2), pp 179–188.

[4]. Rosenblatt, F. (1957): The Perceptron, a Perceiving and Recognizing Automaton Project Para. Cornell Aeronautical Laboratory. Technical report.

[5]. Zurada J.M. (1994): Introduction to Artificial Neural Systems, Jaico Publishing house.

[6]. Johnson R. A. and Wichern D.W. (2012):Applied Multivariate Statistical Analysis, PHI, Eastern Economic Edition, 6th edition.

[7]. Draper N.R and Smith H. (1998): Applied Regression Analysis, 3rdEdition, John Wiley Publications.