

COMPARATIVE STUDIES OF VARIOUS CLASSIFICATION TECHNIQUES AND ITS CHARACTERISTICS

¹M.Manoj priya

¹Research scholar, PG & Research Department of Computer Science, Raja Doraisingham Govt. Arts College, Sivagangai.

²Dr.N.Sujatha

²Assistant Professor, PG & Research Department of Computer Science, Raja Doraisingham Govt. Arts College, Sivagangai.

ABSTRACT: Classification is a model pronouncement process that is used for portioning the data into different classes according to some constraints. In other words, we can say that classification is a procedure of generalizing the data according to different instances. Several major kinds of classification algorithms classifier including Naïve Bayes, SVM, Back propagation, Decision Tree and Neural Network. This paper provides an inclusive survey of the different classification algorithm.

Keywords: Classification techniques, SVM, Neural network, Decision tree, Naïve Bayes classification.

1. INTRODUCTION

Data mining is an investigation and analysis of data from different perspectives, in order to discover meaningful prototype and rules[1]. Data mining extract, transform and load transaction data onto the data warehouse system, it supplies and manages the data in a multidimensional database system. it also analyzes the data by application software and presents the data in a helpful format, such as a graph or table. The aim of data mining is to plan and work efficiently with large data sets. Data mining provides different techniques for discovering hidden patterns from large data set[1][18]. Data mining is a multistep process which requires accessing and analyzing results and taking appropriate action. The data to be accessed can be stored in one or more equipped databases. In data mining, the data is mined using two learning approaches i.e supervised learning or unsupervised learning.

Supervised Learning:

It is also known as intended for data mining. In this, the variables under observation are split into expounding variables and dependent variables. The main objective is to determine a relationship between these two variables. In data mining techniques, the values of the dependent variable must be well defined for an adequately large part of the data set. Supervised models are neural network and decision trees.

Unsupervised Learning:

In this, there is no peculiarity between dependent variables and clarifying variables, both are treated same. The main difference among supervised learning and unsupervised learning is that supervised learning requires that the value of the target variable should be known and well defined while in unsupervised learning either target variable is unknown or its value is known for a small number of cases.

2. TECHNIQUES OF DATA MINING:

The different data mining techniques used for the precise classes of six activities are as follows:

1. Classification
2. Prediction
3. Estimation
4. Clustering
5. Association rules
6. Description and visualization

3. CLASSIFICATION:

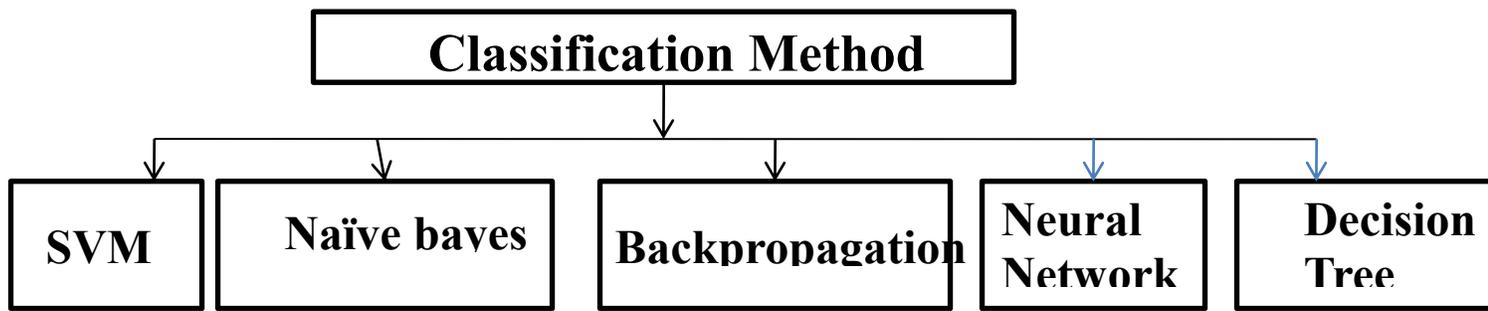
There are two forms of data analysis: classification and prediction. Classification is a machine learning method which is used to allocate each dataset to predefined groups while prediction is used to predict continuous valued function. The main objective of classification is to accurately envisage the target category for each item in a given data set[17].

The classification is done in two steps:

- construct the model
- use the classifier for classification

The accuracy of classification rule is projected and if it is established acceptable then applied to other data sets. The simplest classification problem is binary classification which has only two potential values low and high. The other classification problem has additional two values. There are different techniques used for data classification to find out relationships between the values of the predictors and the target value.

The frequently used methods for data mining classification tasks can be classified into the following groups:



3.1 Decision Tree Induction:

A decision tree is a method frequently used in data mining. It consists of a decision tree generated on the source of instances. The decision tree is a directed tree with a node called “root” that has no incoming edges as all other nodes have precisely one incoming edge[16]. A node with leaving edge is called an internal node. All other nodes are called leaves.

Each internal node denotes a test on an attribute and each division denotes the outcome of a test, and the leaf node holds a class label. Decision tree induction algorithms function recursively[6]. In this, first, an attribute must be selected as the root node. In order to generate the most well-organized decision tree, the root node must effectively divide the data. Each internal node splits the instance space into two or more subspaces until they all have the same classification. The best split depends on the numerical property information gain. The central algorithm used for decision tree induction is a greedy algorithm that constructs decision tree in a recursive, top-down, divide-and-conquer manner.

Algorithm: ID3 (Examples, Target_Attribute, Attributes)

Create a root node for the tree

If every example is positive, Return the single-node tree Root, with label = +.

If every example is negative, Return the single-node tree Root, with label = -.

If number of predicting attribute is empty, then Return the solitary node tree Root,

With label = most common value of the target attribute in the examples.

Otherwise Begin

A ← The Attribute that most excellent classifieds examples.

Decision Tree attribute for Root = A.

For each possible value, v_i , of A,

Add a new tree branch under Root, corresponding to the test $A = v_i$.

Let Examples(v_i) be the subset of examples that contain the value v_i for A

If Examples(v_i) is empty

Then below this new branch add a leaf node with label = most frequent target value in the examples

Else below this new branch add the subtree ID3 (Examples(v_i), Target_Attribute, Attributes – {A})

End

Return Root

Advantage:

1. The evaluation decisive factor in the decision tree algorithm is the solution of an attribute to test at each decision node in the tree.
2. Decision trees can handle both insignificant and numerical attributes.
3. Decision trees representation is rich enough to represent any discrete-value classifier.
4. Decision trees are able to handle datasets that may have errors.
5. Decision trees are able to handle datasets that may have missing values.
6. Decision trees are measured to be a nonparametric method. This means that decision trees have no assumptions about the space allocation and the classifier structure.

Disadvantage:

1. A decision tree is that trees use up data very quickly in the training procedure.
2. They should never be used with small data sets.
3. They are also vastly sensitive to noise in the data and they try to fit the data precisely.

3.2 Baye’s Theorem

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification[4]. It can solve diagnostic and predictive problems.

Bayesian classifiers are numerical classifiers. They can predict class membership probabilities [15]. Naïve Bayes (NB) probabilistic classifiers are generally studied in machine learning. The basic idea in NB approaches is to utilize the joint probabilities of words and categories to estimate the probabilities of categories specified a document.

The naïve part of NB methods is the assumption of word independence, i.e. the conditional probability of a word given a category is implicit in being independent of the conditional probabilities of other words given that category. This assumption makes the computation of the NB classifiers for more efficient than the exponential complexity of non-naïve Bayes approaches because it does not use word combinations as predictors [7].

For $y \in C$, let $P(y)$ be the prior probability of every class. For x_{ij} (feature j conditionally to y). Then, given a test point x whose feature values are (x_1, \dots, x_d) , the naïve Bayes classification function is expressed by,

$$\hat{\Phi}(x') = \underset{y' \in C}{\operatorname{argmax}} P(y') \prod_{j=1}^d P(x'_j | y')$$

Algorithm:

- Let T be a training set of samples, each with their class labels. There are k classes, C_1, C_2, \dots, C_k . Every sample is represented by an n -dimensional vector, $X = \{x_1, x_2, \dots, x_n\}$, depicting n calculated values of the n attributes, A_1, A_2, \dots, A_n , respectively.
- Given a sample X , the classifier will calculate that X belongs to the class having the highest a posteriori probability, conditioned on X . That is X is predicted to fit into the class C_i if and only if, $P(C_i | X) > P(C_j | X)$ for $1 \leq j \leq m, j \neq i$.
Thus we discover the class that maximizes $P(C_i | X)$. The class C_i for which $P(C_i | X)$ is maximized is called the maximum posterior hypothesis.
- As $P(X)$ is similar for all classes, only $P(X|C_i)P(C_i)$ need be maximized. If the class a priori probabilities, $P(C_i)$, are not known, then it is normally understood that the classes are equally likely, that is, $P(C_1) = P(C_2) = \dots = P(C_k)$, and we maximize $P(X|C_i)$. Otherwise, we maximize $P(X|C_i)P(C_i)$. Note that the class prior probabilities may be projected by $P(C_i) = \text{freq}(C_i, T)/|T|$.
- Given data sets with a lot of attributes, it would be computationally expensive to work out $P(X|C_i)$. In order to reduce computation in evaluating $P(X|C_i)P(C_i)$, the naïve hypothesis of class conditional independence is made. This presumes that the value of the attributes is conditionally self-sufficient of one another, given the class label of the sample.
- In order to calculate the class label of X , $P(X|C_i)P(C_i)$ is evaluated for each class C_i . The classifier predicts with the intention of the class label of X is C_i if and only if it is the class that maximizes $P(X|C_i)P(C_i)$.

Advantages:

- Training is very easy and fast.
- Naive Bayes is that it only requires a small amount of training data to approximate the parameters necessary for classification.
- Naive Bayes classifiers worked quite well in many complex real-world problems.
- Naive Bayes algorithm affords fast highly scalable model building and scoring.
- Naive Bayes can be used for both binary and multiclass classification problems.

Disadvantage:

- Violation of independence assumption.
- Zero conditional probability problem.

3.3 Support Vector Machines:

The main aim of support vector machine is to find the accurate classification technique to differentiate between members of the two classes in the training data. In support vector machine technique the optimal boundary is known as the hyperplane. The vectors that are placed near the hyperplane are called supporting vectors. If space is not linearly separable there may be no separating hyperplane.

SVMs have been mostly used in semantic image analysis tasks due to their reported generalization ability. Under the proposed approach, SVMs are engaged for performing the association of the computed image regions to one of the distinct high-level semantic concepts based on the estimated region feature vector. Each SVM at the evaluation stage returns for all segment a numerical value in the range $[0, 1]$ denoting the degree of confidence.

Algorithm:

```

candidateSV = { closest pair from opposite classes }
while there be violating points do
    Find a violator
    candidateSV = candidateSV S violator
if any  $\alpha p < 0$  due to addition of  $c$  to  $S$  then
    candidateSV = candidateSV \ p
    repeat till all such points are pruned
end if
end while

```

Advantages:

- Support vector machine can be easily extended to perform numerical calculations.
- Support vector machine is very useful for general pattern recognition, regression, and classification.
- Support vector machine can be used for pattern classification which has multilayer perceptrons and radial basis function network.

Disadvantages:

- Support vector machine is computationally inefficient.
- SVM is a binary classifier. To do a multi-class classification, couple-wise classifications can be used (one class besides all others, for every class).

Algorithm:

function Perception-Learner(*problem*) **returns** a set of (learned) weights

inputs: *examples*: Training examples

targets: label value

rate: learning rate

local variables: *weights*: a node

weights ← Initialize-Weights(*weights*)

while not termination condition satisfied **do**

for *i* ← 1 to a number of examples **do**

for *j* ← 0 to a number of features **do**

weights[*j*] ← *weights*[*j*] + *rate* * (*Target-Sum*[*i*] - *Weighted-Sum*[*i*] * *examples*[*i*][*j*])

end

if termination condition is satisfied **then return** *weights*

end

Advantages:

1. Adapt to unknown situations.
2. Powerful, it can model complex functions.
3. Ease of use learns by example, and very little user domain-specific expertise needed.

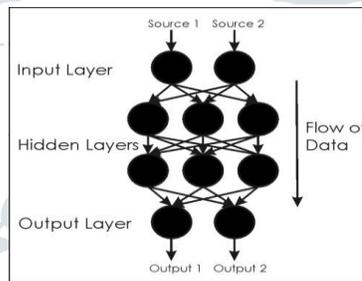


Fig: 1 Neural Network with hidden layers.

Disadvantages:

1. Time to train NN is probably identified as biggest disadvantage.
2. The large complexity of the network structure.

Classification Techniques	Descriptions	Features	Limitations
Decision Tree	A decision tree is a support tool that uses a tree-like graph.	<ul style="list-style-type: none"> • Decision trees are simple to understand and interpret. • It can be combined with other decision techniques. 	<ul style="list-style-type: none"> • Low-Performance • Poor Resolution on Data With Complex Relationships Among the Variables.
Naïve Bayes	In machine learning, naïve Bayes classifiers are a relations of simple probabilistic classifiers based on applying Bayes theorem with strong (naïve) independence assumptions between the features.	<ul style="list-style-type: none"> • Simple to implement. • Great computational efficiency and classification rate. • It predicts accurate results for most of the classification and prediction problems. 	<ul style="list-style-type: none"> • The precision of algorithm de-creases if the amount of data is the loss. • For obtaining good results it requires a very large number of records.
Support Vector Machine	A support vector Machine is a discriminative classifier formally defined by a separating hyper-plane.	<ul style="list-style-type: none"> • High accuracy. • Work well even if data is not linearly separable in the base feature space 	<ul style="list-style-type: none"> • Speed and size requirements are both in training and testing is more. • High complexity and extensive memory require-ments for classification in many cases.
Backpropagati on algorithm	The backward propagation of errors is a common method of training artificial neural networks and used in concurrence with an optimization method such as gradient descent.	<ul style="list-style-type: none"> • It is very flexible. • It tends to be significantly faster for training recurring neural networks than general-purpose optimization techniques such as evolutionary optimization. 	<ul style="list-style-type: none"> • Gradient descent with backpropa-gation is not guaranteed to find the global mini-mum of the error function, but only a local minimum. • It does not require normalization of input vectors; however, norma-lization could develop perfor-mance.

Neural network algorithm	A neural network is a system of hardware and software patterned after the operation of neurons in the human brain.	<ul style="list-style-type: none"> • It is easy to use, with few parameters to adjust. • A neural network learns and reprogramming is not needed. 	<ul style="list-style-type: none"> • Requires high processing time if the neural network is large. • Difficult to know how many neurons and layers are necessary.
--------------------------	--	---	---

4. CONCLUSION

Classification methods are naturally strong in modeling interactions. Several of the classification methods produce a set of interacting loci that best predict the phenotype. However, a straightforward application of classification methods to large numbers of markers has a probable risk picking up randomly associated markers.

This paper covers with various classification techniques used in data mining. Each technique has got its own pros and cons as given this paper. The study analyzed algorithms such as Naive Bayes, Decision Tree, Support Vector Machine, Back Propagation and Neural Network Algorithms.

REFERENCES

- [1] G.Kesavaraj, Dr.S.Sukumaran "A Study on Classification Techniques in Data Mining" IEEE – 31661.
- [2] R. O. Duda and P. E. Hart, Pattern classification, and scene analysis, John Wiley and Sons, 1973.
- [3] J. R. Quinlan. Induction of decision trees. Machine Learning, 1:81–106, 1986.
- [4] P. Langley, W. Iba and K. Thompson, "An analysis of Bayesian Classifiers.," in Proceedings of the Tenth National Conference on Artificial Intelligence, San Jose, CA, 1992.
- [5] N. Friedman, D. Geiger and M. Goldszmidt, "Bayesian Network Classifiers.," Machine Learning, vol. 29, pp. 131-163, 1997.
- [6] Micheline Kamber, Lara Winstone, Wan Gong, Shang Cheng, Jiawei Han, "Generalization and Decision Tree Induction: Efficient Classification in Data Mining., Canada V5A IS6, 1996.
- [7] Y. Yang, X. Liu, "A re-examination of text categorization methods," Annual ACM Conference on Research and Development in Information Retrieval, USA, 1999, pp. 42-49.
- [8] S.S.Keerthi, S.K. Shevade, C. Bhattacharyya, and K. R. K. Murthy. A fast iterative nearest point algorithm for support vector machine classifier design. IEEE Transactions on Neural Networks, 11(1):124–136, 2000.
- [9] Fabricio Voznika Leonardo Viana "DATA MINING CLASSIFICATION" Springer, 2001.
- [10] Danny Robert. DirectSVM: A simple support vector machine perception. Journal of VLSI Signal Processing Systems, 2001.
- [11] I. Rish, "An empirical study of the naive Bayes classifier," IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, vol. 22, pp. 41-46, 2001
- [12] S. V. N. Vishwanathan and M. Narasimha Murty. Geometric SVM: A fast and intuitive SVM algorithm. Technical Report IISC-CSA-2001-14, Dept. of CSA, Indian Institute of Science, Bangalore, India, November 2001. Submitted to ICPR 2002.
- [13] K. I. Kim, K. Jung, S. H. Park, and H. J. Kim. Support vector machines for texture classification. IEEE Trans. Pattern Anal. Mach. Intell., 24(11):1542–1550, 2002.
- [14] Chien-cheng Yu, and Bin-Da Liu, "A backpropagation algorithm with adaptive learning rate and momentum coefficient," IEEE, 2002.
- [15] J. Han, M. Kamber, "Data Mining: Concepts and Techniques," Elsevier, Second Edition, 2006.
- [16] Tomoki Watauga, Tomonobu Ozaki, and Takenao Ohkawa. "Decision Tree Construction from Multidimensional Structured Data..Sixth IEEE International Conference on Data Mining – Workshops, 2006.
- [17] S.Archana, Dr. K.Elangovan "Survey of Classification Techniques in Data Mining" International Journal of Computer Science and Mobile Applications, Vol.2 Issue. 2, February- 2014.
- [18] Sagar S. Nikam "A Comparative Study of Classification Techniques in Data Mining Algorithms" ISSN: 0974-6471 April 2015, Vol. 8, No. (1).