International Journal of Science and Engineering Applications
Volume 6 Issue 01, 2017, ISSN-2319-7560 (Online)

A Survey on Different Classification Techniques In Data Mining

Meghana.L[1], N.Deepika[2]
[1]New Horizon College of Engg. Bangalore, India
[2] New Horizon College of Engg. Bangalore, India

Abstract
Data mining is a procedure which finds valuable patterns from large amount of data. Data mining is the examination step of the "knowledge discovery in databases" process, or KDD. The general goal of the data mining process is to retrieve information from a data set and convert it into an explicable structure for further use. There are various data mining methods like classification, frequent patterns, association, clustering. The present paper provides a survey on different classification techniques involved in data mining. The idea of Classification analysis is the organization of data in given classes according to some constraints. The goal of this survey is to provide a comprehensive review on the advantages and disadvantages of the classification techniques.
Keywords: *data mining, data mining techniques, classification, Bayesian, Decision tree, Rule based.*

1. Introduction

Terabytes or petabytes of data are stored in various storage devices. This explosive growth of data in large volume has led to the introduction of data mining. The data mining technique is the process of retrieve the formerly unknown, useable patterns and interesting information from large data set. The extracted information is transformed into an understandable structure for further use. Data mining process use software techniques tools for finding patterns and regularities in sets of data.

2. Data mining techniques

Data mining is one of the tasks in the process of knowledge discovery in database (KDD). Data mining approaches are used to produce the kind of patterns to be created in data mining tasks. The different data mining techniques are as follows:

i) Classification:

Classification analysis is the forming of data in given classes. It is also known as supervised classification. The classification uses given class labels to direct the objects in the data collection. Classification procedure usually use a training set where all objects are already related with known class labels.

ii) Characterization:

Data characterization is a summarization of overall features of objects in a target class, and provides characteristic rules. The data applicable to a user-specified class are normally regained by a database query and run through a summarization module to extract the principle of the data at various levels of abstractions.

iii) Discrimination:

Data discrimination yields the discriminant rules and the technique is fundamentally the evaluation of the general structures of objects between two classes which are, the target class and the contrasting class. The techniques used for data discrimination are very similar to the techniques used for data characterization with the exception that data discrimination results include comparative measures.

iv) Association Analysis:

Association analysis is the detection of association rules. It studies the frequency of items taking place together in transactional databases, and identifies the frequent item sets.

v) Clustering:

Similar to classification, clustering is the formation of data in classes. In clustering, class labels are unidentified and it is a job of the clustering algorithm to discover suitable classes. Clustering technique can also be called as unsupervised classification.

vi) Outlier Analysis:

Outliers are some data rudiments that cannot be gathered in a given class or cluster. They are either known as exceptions or surprises. The outliers are considered noise and are rejected in some domains, but they can disclose important information in other domains. This method can be very substantial and their analysis can be valuable. [8]

3. Classification methods

Classification is a data mining method that allocates items to the target categories or classes in a given collection. The goal of classification is to precisely forecast the target class for each case in the data. Data classification is of two-step process, which involves a) learning step, where a classification model is constructed and b) a classification step, where the constructed model is used to guess the class labels for given data setThe different classification methods are as follows:

i. Decision tree induction
ii. Bayes classification method
iii. Rule-Based classification
iv. Classification by Backpropagation
v. Support Vector Machines
vi. k nearest neighbors method [3]

3.1 Decision tree induction

A decision tree is a tree in which each branch node signifies a choice between a number of substitutes, and each leaf node represents a particular decision. Decision tree are normally used for gaining the information with the purpose of making the decision.

Algorithm:

Generate a decision tree from the training tuples of data Partition D.

Input:

• Data partition, D, which is a set of training tuples and their associated class labels.

• Attribute list, the set of candidate attributes.

• Attribute selection method, a procedure to determine the splitting criterion that "best" partitions the data tuples into individual classes

Output: A decision tree.

Method:

1. Create a node N.
2. If tuples in D are all of the same class, C then
   a. Return N as a leaf node labelled with the class C.
3. If attribute list is empty then
4. Return N as a leaf node labelled with the majority class in D.
5. Apply Attribute selection method (D, attribute list) to find the "best" splitting criterion.
6. Label node N with splitting criterion.
7. If splitting attribute is discrete-valued and multi-way splits allowed then
8. Attribute list ß attribute list - splitting attribute.
9. For each outcome j of splitting criterion
   a. Let Dj be the set of data tuples in D satisfying outcome j;
   b. If Dj is empty then
   c. Attach a leaf labelled with the majority class in D to node N;
   d. Else attach the node returned by Generate decision tree (Dj, attribute list) to node N;
10. End for
11. Return N;

Attribute selection measure:

Three popular selection measure:

a. Information gain: The information gain measure is used to choose the test attribute at each node in the created tree. The attribute which has the highest information gain is selected as the test attribute for the current node.

$$\text{Info}(D) = -\sum_{i=1}^{n} p_i \log_2 (p_i) \quad (1)$$

Where, Pi is the probability that an indiscriminate tuple in D belongs to Ci and is valued by |Ci, D|/ |D|

$\text{Info}_A(D)$ is the anticipated information essential to organize a tuple from D base on partitioning by A.

$$\text{Info}_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \text{Info}(Dj) \quad (2)$$

Gain is given by $\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D)$

b) Gain ratio: This overwhelms the injustice of Information gain. It gains the information using a split information value by applying a normalization method. The split information value denotes the possible information that is created by splitting the training data set D into v partitions, equivalent to v outcomes on attribute A.

$$\text{SplitInfo}_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2 \frac{|D_j|}{|D|} \quad (3)$$

The gain ratio is defined as:

$$\text{GainRatio}(A) = \text{Gain}(A) / \text{SplitInfo}_A(D) \quad (4)$$

The attribute with the maximum gain ratio is selected as the splitting attribute.

c) Gini index: The Gini Index measures the contamination of a data partition D.

$$\text{Gini}(D) = 1 - \sum_{j=1}^{m} pi^2 \quad (5)$$

Where m is the number of classes, pi is the probability that a tuple in D belongs to class Ci.
This is a weighted sum of the impurity of each partition.

$$\text{Gini}_A(D) = \frac{|D1|}{|D|} \text{Gini}(D1) + \frac{|D2|}{|D|} \text{Gini}(D2) \quad (6)$$

The attribute that maximizes the reduction in impurity is chosen as the splitting attribute.[1]

3.2 Bayes classification method

A Bayesian classifier is a classification technique which is used to define or decide if the given tuple belongs to a particular class or not. The classification is based on Bayes' theorem.
Bayes theorem is given by the following:

$$P(H/D) = \frac{P(D/H)\,P(H)}{P(H)} \quad (7)$$

Where, P(H) : Prior probability of hypothesis h
P(D) : Prior probability of training data D
P(H/D) : Probability of H given D
P(D/H) : Probability of D given H

Algorithm:
Method:
1. Let D be set of training tuples
2. Each Tuple is an 'n' dimensional attribute vector like
X : (x1,x2,x3,…. xn) are attributes
3. Let there be 'm' Classes: C1,C2,C3…Cm
4. Naïve Bayes classifier predicts X belongs to Class Ci iff
P (Ci/X) > P(Cj/X) for $1 \leq j \leq m$, $j \neq i$
5. Compute Maximum Posteriori Hypothesis
P(Ci/X) = P(X/Ci) P(Ci) / P(X)
6. Maximize P(X/Ci) P(Ci) as P(X) is constant
7. With many attributes, it is computationally expensive to evaluate P(X/Ci).To reduce that expensive Naïve Assumption of "class conditional independence" is made.

$$P(X/Ci) = \prod_{k=1}^{n} P(x_k / Ci)$$

$$P(X/Ci) = P(x1/Ci) * P(x2/Ci) *\ldots * P(xn/ Ci)$$

We can easily estimate the probabilities $P(x1/Ci/)$, $P(x2/Ci)$, …. , $P(xn/Ci)$ from the training tuples. And here *xk* refers to the value of attribute *Ak* for tuple ***X***.
8. To predict the class label of ***X***, the equation $P(\boldsymbol{X}/Ci)P(Ci)$ is evaluated for each class *Ci* .
Therefore Naive Bayes goes out to be excellent in certain applications. Text classification is one area where it really works better. [4]

3.3 Rule Based Classification

A rule-based classifier makes usage of a set of IF-THEN rules for the purpose of classification. These rules are created either using a decision tree or directly from the training data which uses an algorithm called sequential covering algorithm.
An IF-THEN rule is an expression of the form:

IF *condition* THEN *conclusion*

Where, Condition (or LHS) IF part is rule antecedent/precondition and Conclusion (or RHS) THEN part is rule consequent.
Let's consider an example of rule *R*1,
*R*1: IF *age = youth* AND *student = yes* THEN *buys_computer = yes*.

*R*1 can also be written as
R1: ((age = youth) ∧ *(student = yes))* → *(buys_computer = yes)*

If the condition or all the attribute tests in a rule antecedent holds true for a given tuple, then we can say that the rule antecedent is satisfied and that the rule covers the tuple.

A rule *R* can be measured by coverage and accuracy.
Coverage of a rule: The percentage of instances that satisfy the antecedent of a Rule.

$$\text{Coverage}(R) = \frac{n_{covers}}{|D|} \quad (8)$$

Accuracy of a rule: The percentage of instances that satisfy both the antecedent and consequent of a rule.

$$\text{Accuracy}(R) = \frac{n_{correct}}{n_{covers}} \quad (9)$$

3.3.1 Rule Induction Using a Sequential Covering Algorithm

Algorithm for sequential covering

Input: *D*, a data set of class-labeled tuples;

Att_vals, the set of all attributes and their possible values.
Output**:** A set of IF-THEN rules.
Method:
1. Rule set = { } // an initial set of rules is empty
2. for each class *c* do
3. repeat
4. Rule = Learn_One_Rule(*D*, *Att_vals*, *c*);
5. remove tuples covered by Rule from *D*;
6. Rule_set = Rule_set + Rule; // add new rule to rule set
until terminating condition;
7. endfor
8. return Rule_set ;[7]

3.4 Classification by Backpropagation

Backpropagation is a neural network learning algorithm. Neural network learning is also referred to as connectionist learning due to the connections between units.

Neural Network

Backpropagation pick up by iteratively giving out a data set of training tuples, relating the network's expectation for each tuple with the actual known goal value. The goal value may be the identified class label of the training tuple (for classification problems) or a continuous value (for numeric prediction). For each training tuple, the heaviness are different so as to minimize the mean-squared error between the network's prediction and the actual goal value. These modifications are made in the "backwards" direction (i.e., from the output layer) through each hidden layer down to the first hidden layer.

Algorithm:Backpropagation Neural network learning for classification or numeric prediction, using the backpropagation algorithm.
Input:
D, a data set consisting of the training tuples and their associated target values;
l, the learning rate; *network*, a multilayer feed-forward network.
Output: A trained neural network.
Method:
1. Initialize all weights and biases in *network*;
2. while terminating condition is not satisfied {
3. for each training tuple *X* in *D* {
// Propagate the inputs forward:
4. for each input layer unit *j* {
5. $O_j = I_j$; // output of an input unit is its actual input value
6. for each hidden or output layer unit *j* {
7. $I_j = \sum_i w_{ij} O_i + \theta_j$; //compute the net input of unit *j* with respect to the previous layer, *i*
8. $O_j = 1/(1 + e^{-I_j})$; } // compute the output of each unit *j*
// Backpropagate the errors:
9. for each unit *j* in the output layer
10. $Err_j = O_j(1-O_j)(T_j - O_j)$; // compute the error
11. for each unit *j* in the hidden layers, from the last to the first hidden layer
12. $Err_j = O_j(1-O_j)\sum_k Err_k w_{jk}$; // compute the error with respect to the next higher layer, *k*
13. for each weight w_{ij} in *network* {
14. $\Delta w_{ij} = (l) Err_j O_i$; // weight increment
15. $w_{ij} = w_{ij} + \Delta w_{ij}$; } // weight update
16. for each bias θ_j in *network* {
17. $\Delta \theta_j = (l) Err_j$; // bias increment
18. $\theta_j = \theta_j + \Delta \theta_j$; } // bias update
19. }} [6]

3.5 Support Vector Machines

SVM was first introduced by Vapnik and has been very effective method for regression, classification and general pattern recognition. It is well thought-out a good classifier because of its high broad view performance without the need to add a priori knowledge, even when the aspect of the input space is very high.

Method:
1. In this algorithm we plot each data item as a point in n-dimensional space with the value of a particular coordinate.
2. Then we perform classification by finding the hyperplane that differentiate the two classes.
3. Maximize the distances between nearest data points and hyperplane. This distance is called margin.
4. Select a hyperplane that has high margin.

Geometrically, the margin match up to the shortest distance between the closest data points to a point on the hyperplane. Having this geometric definition agree us to travel around how to maximize the margin, so that even though there are an infinite

number of hyperplanes, only a a small number of qualify as the solution to SVM. To make certain that the maximum boundary hyperplanes are actually set up, an SVM classifier efforts to maximize the following function with respect to w and b

$$L_p = \frac{1}{2}\|\vec{W}\| - \sum_{i=1}^{t} \alpha_i y_i (\vec{W} \cdot \vec{X}_i + b) + \sum_{i=1}^{t} \alpha_i \qquad (10)$$

where t is the number of training examples, and αi , i = 1, . . . , t, are positive numbers such that the derivatives of LP with respect to αi are zero. αi are the Lagrange multipliers and LP is called the Lagrangian.

In this equation, the vectors and constant b define the hyperplane. A book learning machine, such as the SVM, can be displayed as a function class based on some parameters α. Different function classes can have various capacity in learning, which is represented by a parameter h known as the VC dimension.

3.6 k-Nearest Neighbor Algorithm for Classification

K-NN classifiers are lazy learners. It does not build models explicitly.

Assume each sample in our data set has n attributes which forms an n-dimensional vector:

x = (x1, x2, . . . , xn).

These n attributes are the liberated variables. Every single sample also has another attribute, denoted by y, whose value be governed by on the other n attributes x. We assume that y is a categoric variable, and there is a scalar function, f, which assigns a class, y = f(x) to every such vectors. Suppose the set of T is organized with its classes like,

x(i), y(i) for i = 1, 2, . . . , T

This set is the training set. The idea in k-Nearest Neighbor methods is to identify k samples in the training set whose independent variables x are similar to u, and to use these k samples to classify this new sample into a class, v.

Method:

1. Compute distance between two points using the below formula.s

Euclidean distance.

$$Dist(x_1, x_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \qquad (11)$$

2. Determine the class from nearest neighbor list

3. Take the majority vote of class labels among the k-nearest neighbors

4. Weigh the vote according to distance

weight factor, w = $1/Dist^2$

5. Choose the value of k with respect to the below two aspects.

a) If k is too small, sensitive to noise points

b) If k is too large, neighborhood may include points from other classes

6. Attributes are to be scaled to prevent distance measures from being dominated by one of the attributes. [8]

4. Advantages and Disadvantages of classification methods

Table 1: advantages and disadvantages

Methods	Advantages	Disadvantages
1. Decision tree induction	Rules can be generated and they are easy to interpret and understand. It is scalable for large database	It does not handle continuous data. Handling missing data is difficult.
2. Naive Bays Algorithm	It improves performance by removing the irrelevant Features. Good performance. It has short computational time	It requires a very large number of records to obtain good results.
3. Rule based classifiers	Easy to interpret. Can classify new instances rapidly Can easily handle missing values and numeric attributes	It is inefficiency Computational cost is high It involves complex domains
4.Backpropogation	Involves complex	Difficult to understand the

classification	relationship between input and output	model
5. Support vector machines	Popular in text classification problems	Computationally expensive, thus runs slow.
6. K-Nearest Neighbor Algorithm	Easy to understand and Training is very fast.	It has memory limitation

5. Conclusion

The data mining process provides the useful information from large data set by extraction. This process involves many techniques like classification, clustering, outlier analysis, frequent patterns, association. The present paper provides the explanation of different classification methods. There is a detailed description of each method. The goal of this paper to provide the advantages and disadvantages of each classification methods is also figured out in detail.

References

[1] Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, 2ndedition.
[2] Baik, S. Bala, J. (2004), A Decision Tree Algorithm For Distributed Data Mining.
[3] Classification Algorithms for Data Mining: A Survey: International Journal of Innovations in Engineering and Technology (IJIET) Vol. 1 Issue 2 August 2012, ISSN: 2319 – 1058:
[4] Survey on Classification Techniques Used in Data Mining and their Recent Advancements: International Journal of Science, Engineering and Technology Research, Volume 3, Issue 9, September 2014, ISSN: 2278 – 7798
[5] Classification algorithm in Data mining: An Overview: International Journal of P2P Network Trends and Technology (IJPTT) –
Volume 4 Issue 8- Sep 2013 ISSN:2249-2615 http://www.ijpttjournal.org
[6] Survey on Classification Techniques in Data Mining: International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169 Volume: 2 Issue: 7 1983 – 1986
[7]Ruled based classification method: https://www.tutorialspoint.com/data_mining/dm_rbc.html
[8] Data mining techniques and applications: Bharati M. Ramageri / Indian Journal of Computer Science and Engineering Vol. 1 No. 4 301-305 ISSN : 0976-5166
[9] Review on Classification Algorithms in Data Mining: International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 5, Issue 1, January 2015)

First Author
Meghana L received B.E degree in Computer science and Engineering from Vivekananda Institute of Technology in 2016. Currently pursuing M. Tech degree in Computer science and Engineering. Her area of interests includes Data Mining, Information and Network Security.

Second Author
N. Deepika Sr. Asst. Professor having 15 years of experience in academics as pursued her M.Tech from JNTU Hyderabad and B.Tech from SVU. She is currently working in NHCE, Dept. of CSE, Bangalore. She has guided many UG and PG students for their projects. Her research areas include clustering technique, data mining, Web mining, and Big Data analysis.