ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

Survey on Classification Techniques in Data mining

M.Soundarya[1], R.Balakrishnan[2]

Research Scholar, Department of IT, Dr. N.G.P. Arts and Science College, Coimbatore, India[1]

Assistant Professor, Department of IT, Dr. N.G.P. Arts and Science College, Coimbatore, India[2]

Abstract: Data mining is the analysis step of the "Knowledge Discovery in database" process or KDD. It is an interdisciplinary subfield of computer science and the computational process of discovering patterns in large data sets involving methods at the intersection of artificial brainpower, machine learning, figures and relevant data and database systems. Classification is a data mining (machine learning) technique used to predict group membership for data instance. In this paper, it deals about the survey of the several classification techniques. Examples are several ways of classification method such as decision tree induction, Bayesian networks, k-nearest neighbor classifier and fuzzy logic techniques.

Keywords: Bayesian networks, k-nearest neighbor classifier, Fuzzy logic and decision tree induction.

I. INTRODUCTION

Data mining can be chance to be altered and it can generate results which come out to be significant and which cannot actually predict future behavior and cannot be reproduced on a new sample of data and allow small use. Often this results from investigating too many hypotheses and not performing proper statistical hypothesis testing. A simple report of this problem in machine learning is known as over fitting but the same problem can arise at different phases of the process and thus a train/test split when applicable at all it may not be sufficient to prevent this from happening. The final step of knowledge discovery from data is to verify that the patterns produced by the data mining algorithms occur in the wider data set.

The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.

The Knowledge Discovery in Databases (KDD) process is commonly defined with the stages:

(1) Selection
(2)Preprocessing
(3)Transformation
(4) Data Mining
(5) Interpretation/Evaluation.

Classification is the task of generalizing known structure to relate to new data. It is the problem of identifying to which of a set of categories (sub-populations) a new examination belongs on the source of a preparation set of data contain notes (or instances) whose category membership is known.

II. RELATED WORK

1. Decision tree induction:

The decision tree learning uses decision tree as a predictive model which maps observations about an item to conclusions about the item's end value. It is one of the analytical modeling approach used in statistics, data mining and machine learning. More explanatory names for such tree models are classification trees .In tree structures, leaves represent class labels and branches represent conjunctions of features that direct to those class labels.

Decision trees are trees that classify instances by sorting them based on feature values. Each node in a decision tree represents as the node can assume. Instances are classified starting at the root node and sorted based on their feature values.

In decision analysis, a decision tree can be used to visually and clearly represent decisions and decision making. In data mining, a decision tree describes data but not decisions; rather the resultant classification tree can be an input for decision making.

In data mining, decision trees can be described also as the combination of mathematical and computational techniques to aid the description, categorization and generalization of a given set of data.

Data comes in records of the form:

$$(\mathbf{x}, Y) = (x_1, x_2, x_3, ..., x_k, Y)$$

The reliant variable Y, is the target variable that we are trying to understand, classify or generalize. The vector **x** is composed of the input variables, x_1, x_2, x_3 etc., that are used for that task.

Decision trees used in data mining are of two main types:

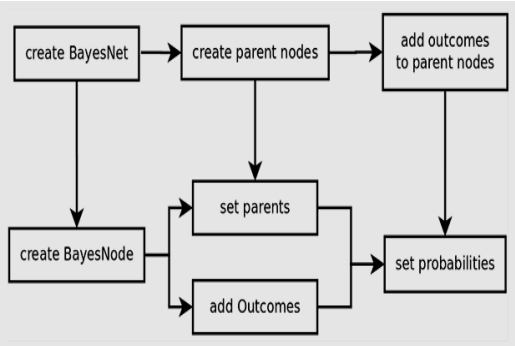
- **Classification tree** analysis is used to predict outcome of the class to which the data belongs.
- **Regression tree** analysis is predicted when the outcome is considered as a real number. (e.g. the price of a house, or a patient's length of stay in a hospital).

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

2. Bayesian Networks:

A Bayesian network, Bayes network, belief network, Bayesian model or probabilistic directed acyclic graphical model. It is a probabilistic graphical model, type of statistical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG). For example, a Bayesian network could represent the probabilistic relationships between diseases and symptoms.



Creating parent nodes using Bayesian network

The Bayes Network is a container for Bayes Nodes, which represent the random variables of the probability distribution of modeling.

A Bayes Node has outcomes, parents, and a conditional probability. It is important to set the probabilities last, after setting the outcomes of the parent nodes.

The methods from Bayes Node that perform some steps they are

- BayesNode.addOutcomes (String...)
- BayesNode.setParents (List<BayesNode>)
- BayesNode.setProbabilities (double...)

Formally, Bayesian networks are DAGs whose nodes represent random variables in the Bayesian sense: they may be observable quantities, latent variables, unknown parameters or hypotheses. Edges represent conditional dependencies; nodes that are not connected represent variables that are conditionally independent of each other. Each node is associated with a probability function that takes as input a particular set of values for the node's parent variables and gives the probability of the variable represented by the node. For example, if the parents are m Boolean variables then the probability function could be represented by a table of 2^m entries, one entry for each of the 2^m possible combinations of its parents being true or false. Similar ideas may be applied to undirected, and possibly cyclic graphs are called Markov networks.

Bayesian network can take the following forms:
1.Declaring that a node is a root node, i.e., it has no parents.
2.Declaring that a node is a leaf node, i.e., it has no children.
3.Declaring that a node is a direct cause or direct effect of another node.
4.Declaring that a node is not directly connected to another node.
5.Declaring that two nodes are independent, given a condition-set.
6. Providing partial nodes ordering, that is, declare that a node appears earlier than another node in the ordering.
7. Providing a complete node ordering.

3.k-nearest neighbor's algorithm:

Nearest neighbor classifiers are based on learning by analogy. The training samples are described by n dimensional numeric attributes. Each sample represents a point in an n-dimensional space. In this way, all of the training samples are stored in an n-dimensional pattern space. When given an unknown sample, a k-nearest neighbor classifier searches the pattern space for the k training samples that are closest to the unknown sample. "Closeness" is defined in terms of Euclidean distance, where the Euclidean distance, where the Euclidean distance between two points,
X=(x1,x2,……,xn) and Y=(y1,y2,….,yn)

The k-nearest neighbors' algorithm is among the simplest of all machine learning algorithms. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its *k* nearest neighbors. *k* is a positive integer, typically small. If *k* = 1, then the object is simply assigned to the class of its nearest neighbor. In binary (two class) classification problems, it is helpful to choose *k* to be an odd number as this avoids tied votes.

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.

In the classification phase, *k* is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the *k* training samples nearest to that query point.

A commonly used distance metric for continuous variables is Euclidean distance. For discrete variables, such as for text classification, another metric can be used, such as the overlap metric (or Hamming distance). Often, the classification accuracy of *k*-NN can be improved significantly if the distance metric is learned with specialized algorithms such as Large Margin Nearest Neighbor or Neighborhood components analysis.

A drawback of the basic "majority voting" classification occurs when the class distribution is skewed. That is, examples of a more frequent class tend to dominate the

prediction of the new example, because they tend to be common among the *k* nearest neighbors due to their large number. One way to overcome this problem is to weight the classification, taking into account the distance from the test point to each of its *k* nearest neighbors. The class (or value, in regression problems) of each of the *k* nearest points is multiplied by a weight proportional to the inverse of the distance from that point to the test point. Another way to overcome skew is by abstraction in data representation. For example in a self-organizing map (SOM), each node is a representative (a center) of a cluster of similar points, regardless of their density in the original training data. *K*-NN can then be applied to the SOM.
Shalev-Shwartz et al [6] proposed an online learning algorithm for learning a Mahalanobis distance metric. The metric is trained with the goal that all similarly labeled inputs have small pair wise distances (bounded from above), while all differently labeled inputs have large pair wise distances (bounded from below). A margin is defined by the difference of these thresholds and induced by a hinge loss function. Our work has a similar basis in its appeal to margins and hinge loss functions, but again differs in its focus on local neighborhoods for KNN classification. In particular, we do not seek to minimize the distance
between all similarly labeled inputs, only those that are specified as neighbors.

Goldberger et al [7] proposed neighborhood component analysis (NCA), a distance metric learning algorithm especially designed to improve kNN classification. The algorithm minimizes the probability of error under stochastic neighborhood assignments using gradient descent. Our work shares essentially the same goals as NCA, but differs in its construction
of a convex objective function.

4. Fuzzy logic:

Fuzzy logic is a form of many-valued logic; it deals with reasoning that is approximate rather than fixed and exact. Compared to traditional binary sets (where variables may take on true or false values), fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false.

In mathematical logic, there are several formal systems of "fuzzy logic"; most of them belong among so-called t-norm fuzzy logics.

Propositional fuzzy logics:

The most important propositional fuzzy logics are:

- Monoidal t-norm-based propositional fuzzy logic MTL is an axiomatization of logic where conjunction is defined by a left continuous t-norm, and implication is defined as the residuum of the t-norm. Its models correspond to MTL-algebras that are prelinear commutative bounded integral residuated lattices.
- Basic propositional fuzzy logic BL is an extension of MTL logic where conjunction is defined by a continuous t-norm, and implication is also defined as the residuum of the t-norm. Its models correspond to BL-algebras.
- Łukasiewicz fuzzy logic is the extension of basic fuzzy logic BL where standard conjunction is the Łukasiewicz t-norm. It has the axioms of basic fuzzy logic plus an axiom of double negation, and its models correspond to MV-algebras.
- Gödel fuzzy logic is the extension of basic fuzzy logic BL where conjunction is Gödel t-norm. It has the axioms of BL plus an axiom of idempotence of conjunction, and its models are called G-algebras.
- Product fuzzy logic is the extension of basic fuzzy logic BL where conjunction is product t-norm. It has the axioms of BL plus another axiom for cancel activity of conjunction, and its models are called product algebras.
- Fuzzy logic with evaluated syntax (sometimes also called Pavelka's logic), denoted by EVŁ, is a further generalization of mathematical fuzzy logic. While the above kinds of fuzzy logic have traditional syntax and many-valued semantics, in EVŁ is evaluated also syntax. This means that each formula has an evaluation. Axiomatization of EVŁ stems from Łukasziewicz fuzzy logic. A generalization of classical Gödel completeness theorem is provable in EVŁ.

III. CONCLUSION

Compare to k-nearest neighbors, Decision trees and Bayesian Network (BN) generally have different operational profiles, when one is very accurate the other is not and vice versa.The role of classification is to generate more precise and accurate system results.Many techniques were used In this paper,decision trees provides accurate results with Bayesian networks.

REFERENCES

[1] Baik, S. Bala, J., A Decision Tree Algorithm for Distributed Data Mining: Towards Network Intrusion Detection, Lecture Notes in Computer Science, Volume 3046,Pages 206 – 212,2004.
[2] Bouckaert, R., Naive Bayes Classifiers That Perform Well with Continuous Variables, Lecture Notes in Computer Science, Volume 3339, Pages 1089 – 1094,2004.
[3] Breslow, L. A. & Aha, D. W, Simplifying decision trees: A survey. Knowledge Engineering Review 12: 1–4,1997..
[4] Brighton, H. & Mellish, C., Advances in Instance Selection for Instance-Based Learning Algorithms. Data Mining and Knowledge Discovery 6: 153–17,2002.
[5] Cheng, J. & Greiner, R., Learning Bayesian BeliefNetwork Classifiers: Algorithms and System, In Stroulia, E.& Matwin, S. (ed.), AI 2001, 141-151, LNAI 2056,2001.
[6] S. Shalev-Shwartz, Y. Singer, and A. Y. Ng. Online and batch learning of pseudo-metrics. InProceedings of the 21st International Conference on Machine Learning, Banff, Canada, 2004.
[7] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis.In L. K. Saul, Y. Weiss, and L. Bottou, editors, Advances in Neural Information Processing Systems 17, pages 513–520, Cambridge, MA, 2005, MIT Press.