ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Classification with K-means Clustering and Decision Tree

Promila Devi[*1], Rajiv Kumar Ranjan[2]
[*1] M.Tech(CSE) Student, [2] Assistant Professor(CSE), Arni University, Indora, Kangra, India
promila.dakshu@gmail.com

Abstract

Classifiers can be either linear means Naive Bayes classifier or non-linear means decision trees.In this work we discusses with decision tree ,Naïve Bayes and k-means clustering .The Naive Bayes is based on conditional probabilities and affords fast, highly scalable model building and scoring. It scales linearly with the number of predictors and rows. And also build process is parallelized..Data Mining supports several algorithms that provide rules. Decision trees are among the best algorithms for data classification, providing good accuracy for many problems in relatively short time. Decision tree scoring is especially fast. The k-Means algorithm is a distance-based clustering algorithm that partitions the data into a predetermined number of clusters provided there are enough distinct cases.

Keywords: clustering, K-means,decision tree.

Introduction

Decision Tree

Classification with Decision Tree

The Decision Tree[1] algorithm, like Naive Bayes[2], is based on conditional probabilities. Unlike Naive Bayes, decision trees generate rules. A rule is a conditional statement that can easily be understood by humans and easily used within a database to identify a set of records. In some applications of data mining, the accuracy of a prediction is the only thing that really matters. It may not be important to know how the model works. For example, a Marketing professional would need complete descriptions of customer segments in order to launch a successful marketing campaign. The Decision Tree algorithm is ideal for this type of application.

Decision Tree Rules

Oracle Data Mining supports several algorithms that provide rules. In addition to decision trees, clustering algorithms provide rules that describe the conditions shared by the members of a cluster, and association rules that describe associations between attributes. Rules provide model transparency, a window on the inner workings of the model. Rules show the basis for the model's predictions. Oracle Data Mining supports a high level of model transparency. While some algorithms provide rules, all algorithms provide model details. You can examine model details to determine how the algorithm handles the attributes internally, including transformations and reverse transformations. Figure 1.1 shows a rule generated by a Decision Tree[3] model. This rule comes from a decision tree that predicts the probability that customers will increase spending if given a loyalty card. A target value of 0 means not likely to increase spending; 1 means likely to increase spending.

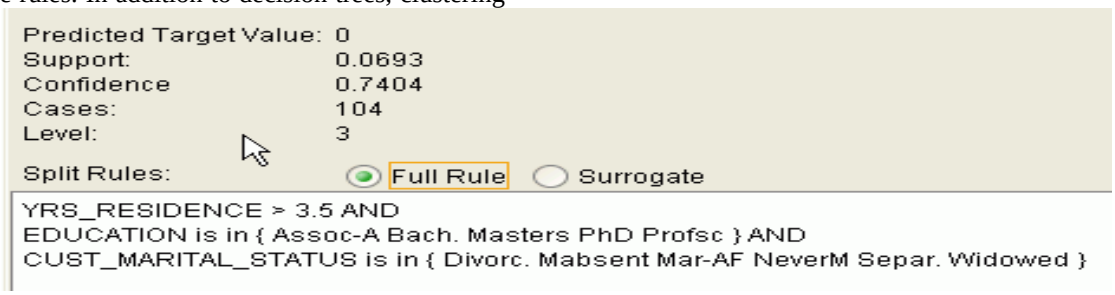

Figure 1.1 Sample Decision Tree Rule

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

Description of "figure 1.1 Sample decision tree rule"
The rule shown in Figure 1.1 represents the conditional statement:
IF (current residence > 3.5 and has college degree and is single)
THEN predicted target value = 0
This rule is a full rule. A surrogate rule is a related attribute that can be used at apply time if the attribute needed for the split is missing.

Advantages of Decision Trees

The Decision Tree [3]algorithm produces accurate and interpretable models with relatively little user intervention. The algorithm can be used for both binary and multiclass classification problems. The algorithm is fast, both at build time and apply time. The build process for Decision Tree is parallelized. (Scoring can be parallelized) irrespective of the algorithm. Decision tree scoring is especially fast.

Growing a Decision Tree

A decision tree [1]predicts a target value by asking a sequence of questions. At a given stage in the sequence, the question that is asked depends upon the answers to the previous questions. The goal is to ask questions that, taken together, uniquely identify specific target values. Graphically, this process forms a tree structure.

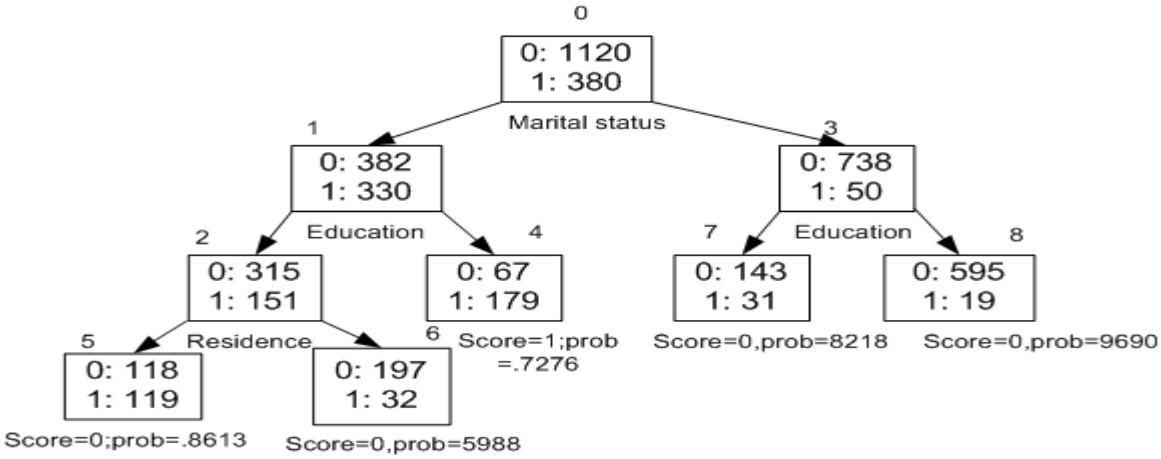

Figure 1.2 Sample Decision Tree

Description of "Figure 1.2 Sample Decision Tree"

Figure 1.2 is a decision tree with nine nodes (and nine corresponding rules). The target attribute is binary: 1 if the customer will increase spending, 0 if the customer will not increase spending. The first split in the tree is based on the CUST_MARITAL_STATUS attribute. The root of the tree (node 0) is split into nodes 1 and 3. Married customers are in node 1; single customers are in node 3.

The rule associated with node 1 is:

Node 1 record Count=712,0 Count=382, 1 Count=330 CUST_MARITAL_STATUS is IN "Married",surrogate:HOUSEHOLD_SIZE is In "3""4-5"

Node 1 has 712 records (cases). In all 712 cases, the CUST_MARITAL_STATUS attribute indicates that the customer is married. Of these, 382 have a target of 0 (not likely to increase spending), and 330 have a target of 1 (likely to increase spending)..

Strengths:

- can generate understandable rules
- perform classification without much computation
- can handle continuous and categorical variables
- provide a clear indication of which fields are most important for prediction or classification

Weakness:

- Not suitable for prediction of continuous attribute.
- Perform poorly with many class and small data.

Naive Bayes

Classification with Naive Bayes

The Naive Bayes[2] algorithm is based on conditional probabilities. It uses Bayes' Theorem, a formula that calculates a probability by counting the frequency of values and combinations of values in the historical

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

data Bayes' Theorem finds the probability of an event occurring given the probability of another event that has already occurred. If B represents the dependent event and A represents the prior event, Bayes' theorem can be stated as follows.

Bayes' Theorem:

Prob(B given A) = Prob(A and B)/Prob(A)

To calculate the probability of B given A, the algorithm counts the number of cases where A and B occur together and divides it by the number of cases where A occurs alone.

Example 2.1 Use Bayes' Theorem to Predict an Increase in Spending

Suppose you want to determine the likelihood that a customer under 21 will increase spending. In this case, the prior condition (A) would be Òunder 21,Ó and the dependent condition (B) would be Òincrease spending."

If there are 100 customers in the training data and 25 of them are customers under 21 who have increased spending, then:

Prob(A and B) = 25%

If 75 of the 100 customers are under 21, then:

Prob(A) = 75%

Bayes' Theorem would predict that 33% of customers under 21 are likely to increase spending (25/75).The cases where both conditions occur together are referred to as pairwise. In **Example 2.1,** 25% of all cases are pairwise.The cases where only the prior event occurs are referred to as singleton. In **Example 2.1**, 75% of all cases are singleton. A visual representation of the conditional relationships used in Bayes' Theorem is shown in **Figure 2.1**

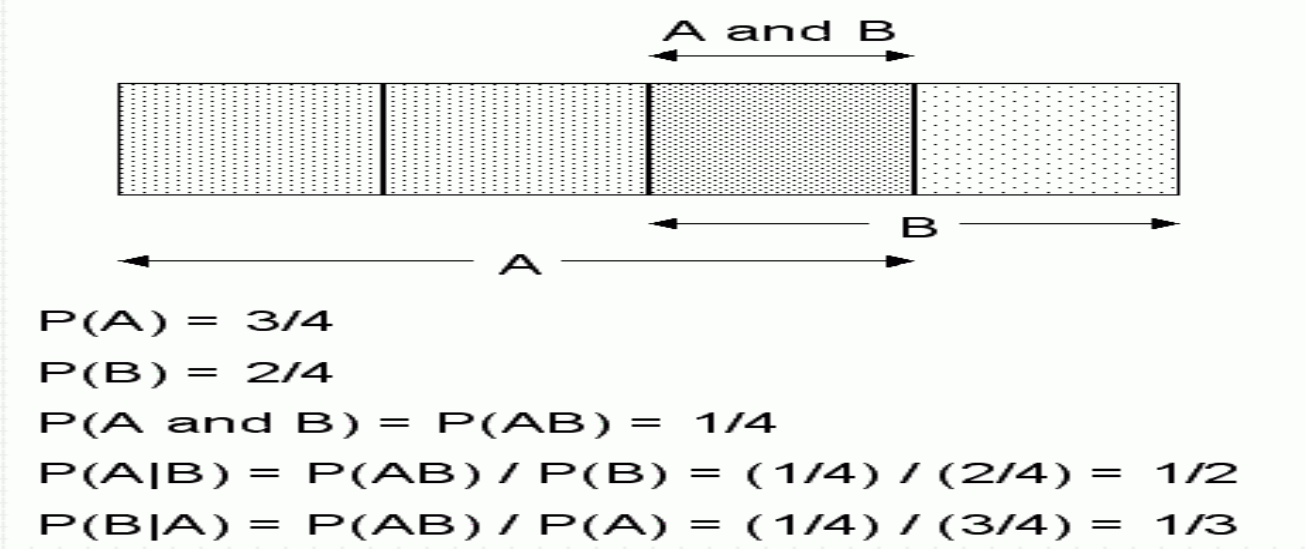

Figure 2.1 Conditional Probabilities in Bayes' Theorem

Description of "Figure 2.1 Conditional Probabilities in Bayes' Theorem"

For purposes of illustration,

Example 2.1 and **Figure** show a dependent event based on a single independent event. In reality, the Naive Bayes algorithm must usually take many independent events into account. In **Example 2.1,** factors such as income, education, gender, and store location might be considered in addition to age.Naive Bayes makes the assumption that each predictor is conditionally independent of the others. For a given target value, the distribution of each predictor is independent of the other predictors.

Advantages of Naive Bayes

The Naive Bayes[2] algorithm affords fast, highly scalable model building and scoring. It scales linearly with the number of predictors and rows. The build process for Naive Bayes is parallelized. (Scoring can be parallelized irrespective of the algorithm. Naive Bayes can be used for both binary and multiclass classification problems.

Data Preparation for Naive Bayes

Automatic Data Preparation performs supervised binning for Naive Bayes. Supervised binning uses decision trees to create the optimal bin boundaries. Both categorical and numerical attributes are binned.Naive Bayes handles missing values naturally as missing at random. The algorithm replaces sparse

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

numerical data with zeros and sparse categorical data with zero vectors. Missing values in nested columns are interpreted as sparse. Missing values in columns with simple data types are interpreted as missing at random.

K-Means Clustering

- Partition clustering approach
- Each cluster is associated with a centroid(center point)
- Each point is assigned to the cluster with the closet centroid.
- Number of clusters,K,must be specified(is predetermined)

Classification with k-Means: The k-Means [4]algorithm is a distance-based clustering algorithm that partitions the data into a predetermined number of clusters (provided there are enough distinct cases.
Distance-based algorithms rely on a distance metric (function) to measure the similarity between data points. The distance metric is either Euclidean, Cosine, or Fast Cosine distance. Data points are assigned to the nearest cluster according to the distance metric used..
This approach to k-means avoids the need for building multiple k-means models and provides clustering results that are consistently superior to the traditional k-means.The clusters discovered by enhanced k-Means are used to generate a Bayesian probability model that is then used during scoring (model apply) for assigning data points to clusters. The k-means[5] algorithm can be interpreted as a mixture model where the mixture components are spherical multivariate normal distributions with the same variance for all components.

Limitations of K-means

K-means has problems when clusters are of differing

a. Sizes
b. Densities

Data Preparation for k-Means

Automatic Data Preparation performs outlier-sensitive normalization for k-Means.When there are missing values in columns with simple data types (not nested), k-Means[6] interprets them as missing at random. The algorithm replaces missing categorical values with the mode and missing numerical values with the mean.When there are missing values in nested columns, k-Means interprets them as sparse. The algorithm replaces sparse numerical data with zeros and sparse categorical data with zero vectors.If you manage your own data preparation for k-Means[9], keep in mind that outliers with equi-width binning can prevent k-Means from creating clusters that are different in content. The clusters may have very similar centroids, histograms, and rules.

Conclusion

Almost every aspect of K-means has been modified Distance measures,Centroid and objective definitions, Overall process, Efficiency Enhancements, Initialization. In this paper we have presented a classification system that improves classification accuracy of any given decision tree algorithm by combining it with a clustering algorithm. The results exceeded our expectation, since clustering algorithms operate blindly (i.e. not taking the class into account) over the data, but yet manage to improve the accuracy of the system greatly, when compared to the basic system. In future research, more attention will be paid to studying cluster structure sensitivity that reflects real location of classes as well as to studying clustering quality. The K-means[7][8] algorithm is a popular data-clustering algorithm. However, one of its drawbacks is the requirement for the number of clusters, to be specified before the algorithm is applied. This paper first reviews existing methods for selecting the number of clusters for the algorithm. Factors that affect this selection are then discussed and a new measure to assist the selection is proposed. The paper concludes with an analysis of the results of using the proposed measure to determine the number of clusters for the K –means[9] algorithm for different data sets.

References

1. *[Ali et al. 2009] Ali, S. A., Sulaiman, N., Mustapha, A., & Mustapha, N. (2009). K-Meaiis Clustering to Improve the Accuracy of Decision Tree Response Classification. Information technology journal, 8(8), 1256-1262*
2. *[Bacon and van Dam 2010] Bacon, D. and van Dam, W. "Recent progress in quantum algorithms"; Commun. ACM53, 2 (2010), 84-93.*
3. *[Barak et al. 2011] Barak A., Gelbard R., "Classification by clustering decision tree-like classifier based on adjusted clusters"; Expert Systems with Applications, 38, 7, 2011, 8220- 8228.*

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

4. *[Bhattacharya et al. 2012] Bhattacharya, A., Chowdhury, N. and De Rajat, K. "Comparative Analysis of Clustering and Biclustering Algorithms for Grouping of Genes: Co-Function and Co-Regulation"; Current Bioinformatics, 7, 1 (2012), 63-76.*
5. *Pelleg, D.and Moore, A .Accelerating exact K-means algorithms with geometric reasoning. In Proceedings of the Conference on Knowledge Discovery in Databases (KDD 99) , San Diego, California, 1999, pp. 277 – 281.*
6. *Pelleg, D.and Moore, A. X -means: extending K –means with efficient estimation of the number of clusters. In Proceedings of the 17th International Conference on Machine Learning (ICML 2000) , Stanford, California, 2000, 727 – 734.*
7. *Pena, J. M., Lazano, J. A.,and Larranaga, P.An empiri- cal comparison of four initialization methods for the K -means algorithm. Pattern Recognition Lett., 1999,20, 1027 – 1040*
8. *The history of k-means type of algorithms (LBG Algorithm, 1980) R.M. Gray and D.L. Neuhoff, "Quantization," IEEE Transactions on Information Theory, Vol. 44, pp. 2325-2384, October 1998.(Commemorative Issue, 1948-1998)Kanungo, T., Mount, D. M., Netanyahu, N., Piatko, C.,Silverman, R.,and Wu, A.*
9. *The efficient K-means clus-tering algorithm: analysis and implementation.IEEETrans. Pattern Analysis Mach. Intell.2002,24(7),881 – 892*