CHAPTER-19

Other Classification Methods

19. Other Classification Methods

19.1 Introduction

In this section, we give a brief description of a number of other classification methods. These methods include k-nearest neighbor classification, case-based reasoning, genetic algorithms, rough set,snd fuzzy set approaches. In general, these methods are less commonly used for classification in commercial data mining systems than the methods described earlier in this chapter. Nearest neighbor classification, for example, stores all training samples, which may present difficulties when learning from very large data sets. furthermore, many applications of case-based reasoning, genetic algorithms, and rough sets for classification are still in the prototype phase. These method , however, are enjoying increasing popularity, and hence we include them here.

19.2 K-Nearest Neighbor classifiers

Nearest neighbor classifiers are based on learning by analogy. The training sample are described by M-dimensional space; In this way, all of the training samples ai stored in an n-dimensional pattern space for the k training sample that are closest to the unknown sample. These k training samples are the k "nearest neighbors" of the unknown sample, "Closeness" is defined in terms of Eudidean distance, where the Euclidean distance between two points,

D(X,Y)=

The unknown sample is assigned the most common class among its k nearest neighbors when k=1, the unknown sample is assigned the class of the training sample that is closest to it in pattern space. Nearest neighbor classifiers are instance-based or lazy learners in that they store all of the training samples and do not build a classifier until a new (unlabeled) sample needs to be classified. This contrasts with eager learning methods, such as decision tree induction and backpropagation, which construct a generalization model before receiving new samples to classify. Lazy learners can incur expensive computational costs when the number of potential neighbors (i.e., stored training samples) with which to compare a given unlabeled sample is great. Therefore, they require efficient indexing techniques. As expected, lazy learning methods are faster at training-than eager methods, but slower at classification since all computation is delayed to that time. Unlike decision tree induction and backpropagation ,

nearest neighbor classifiers assign equal weight to each attribute. This may cause confusion when there are many irrelevant attributes in the data.

Nearest neighbor classifiers can also be used for prediction, that is, to return real-valued prediction for a given unknown sample. In this case, the classifier returns the average value of the real-valued labels associated with the k nearest neighbors of the unknown sample.

19.3 Case- Based Reasoning

Case-based reasoning (CBR) classifiers ate instanced-based. Unlike nearest neighbor classifiers, which store training samples as points in Euclidean space, the samples or “cases” stored by CBR are complex symbolic descriptions. Business applications of CBR include problem resolution for customer service help desks, for example, where cases describe product-related diagnostic problems. CBR has also been applied to areas such as engineering and law, where cases are either technical designs or legal rulings, respectively.

When given a new case to classify, a case-based reasoned will first check If an identical training case exists. If one is found, then the accompanying solution to that case is returned. If no identical case is found, then the case based reasoner will search for training cases having components that are similar to those of the new case. Conceptually, these training cases may be considered as neighbors of the new case. If cases are represented as graphs, this involves searching for subgraphs that ate similar to subgraphs within the new case. The case-based reasoner tries to combine the solutions of the neighboring training cases in order to propose a solution for the new case. If incompatibilities arise with the individual solutions, then backtracking to search for other solutions may be necessary. The case-based reasoned may employ background knowledge and problem-solving strategies in order to propose a feasible combined- solution.

Challenges in case-based reasoning include finding a good similarity metic (e.g., for matching subgraphs), developing efficient techniques for indexing training cases, and methods for combining solutions.

19.4 Genetic Algorithms

Genetic algorithms attempt to incorporate ideas of natural evolution. In general, genetic learning starts as follows. An initial population is created consisting of randomly generated rules. Each rule can be represented by a string of bits. As a simple example, suppose that samples in a given training set are described by two Boolean attributes, A1 and A2, and that there are two classes, C1 and C2. The rule "IF NOT A1 AND NOT A2 THEN C2" can be encoded as the bit string "100", where the two leftmost bits represent attributes A1 and A2, respectively, and the rightmost bit represents the class, similarly, the rule "IF NOT A1 AND NOT A2 THEN C1" can be encoded as "001". If an attribute has k values, where k>2, then k bits may be used to encode the attribute's values. Classes can be encoded in a similar fashion.

Based on the notion of survival of the fittest, a new population is formed to consist of the fittest rules in the current population, as well as offspring of these rules. Typically, the fitness of a rule is assessed by its classification accuracy on asset of training samples.

Applying genetic operators such as crossover and mutation created offspring, in crossover, substrings from pairs of rules are swapped to form new pairs of rules, in mutation, randomly selected bits in a rule's string are inverted.

The process of generating new populations based on prior populations of rules continues until a populations of rules continues until a population p "evolves" where each rule in P satisfies a prespecified fitness threshold.

Genetic algorithms are easily parallelizable and have been used for classification as well as other optimization problems, in data mining, they may be used to evaluate the fitness of other algorithms.

19.5 Rough Set Approach

Rough set theory can be used for classification to discover structural relationships within imprecise or noisy data. It applies to discrete-valued attributes. Continuous-valued attributes must therefore be discretized prior to its use.

Rough set theory is based on the establishment of equivalence classes within the given training data. All of the data samples forming an equivalence class are indiscernible, that is, the samples are identical with respect to the attributes describing the data. Given real-world data, it is common that some classes connot be distinguished in terms of the available attributes. Rough sets can be used to approximately or "roughly" define such classes. A rough set definition for a given class c is approximated by two sets-a lower approximation of C and an upper approximation of C. the lower approximation of C consists of all

of the data samples that, based on the knowledge of the attributes, ate certain to belong to C without ambiguity. The upper approximation of C consists of all of the samples that, based on the knowledge of the attributes, cannot be described as not belonging to C. decision rules can be generated for each class. Typically, a decision table is used to represent the rules.

Rough sets can also be used for feature reduction (where attributes that do not contribute towards the classification of the given training data can be identified and removed) and relevance analysis (where the contribution or significance of each attribute is assessed with respect to the classification task). The problem of finding the minimal subsets (reducts) of attributes that can describe all of the concepts in the given data set is NP-hard. However, algorithms to reduce the computation intensity have been proposed. In one method, for example, discernibility matrix is used that stores the differences between attribute values for each pair of data samples. Rather than searching on the entire training set, the matrix is instead searched to detect redundant attributes.

19.6 Fuzzy set Approaches

Rule-based systems for classification have the disadvantage that they involve sharp cutoffs for continuous attributes. For example, consider the following rule for customer credit application approval. The rule essentially says that applications for customers who have dad a job for two or more years and who have a high income (i.e., of at least $50k) ate approved:

IF (years_employed) >=2^(income>50k) THEN credit=”approved”

A customer who has had a job for at least two years will receive credit if her income is, say, $50k, but not id it is $49k. Such harsh thresholding may seem unfair. Instead, fuzzy logic can be introduced into the system to allow “fuzzy” thresholds or boundaries to be defined. Rather than having a precise cutoff between categories or sets, fuzzy logic uses truth-values between 0.0 and 1.0 to represent the degree of membership that a certain value has in a given category. Hence, with fuzzy logic, we can capture the notion that an income of $49k is, to some degree, high, although not as high as an income of $50k.

Fuzzy logic is useful for data mining systems performing classification. It provides the advantage of working at a high level of abstraction. In general, the use of fuzzy logic in rule-based systems involves the following:

- Attribute values are converted to fuzzy values. The fuzzy membership or truth values are calculated- Fuzzy logic systems typically provide graphical tools to assist users in this step.
- For a given new sample, more than one fuzzy rule may apply. Each applicable rule contributes a vote for membership in the categories. Typically, the truth-values for each predicted category are summed.
- The sums obtained above ate combined into a value that is returned by the system. Weighting each category by its truth sum and multiplying by the mean truth-value of each category may do this process, the calculations involved may be more complex, depending on the complexity of the fuzzy membership graphs.

Fuzzy logic systems have been used in numerous areas for classification, including health care and finance.

19.7 Prediction

The prediction of continuous values can be modeled by statistical techniques of regression. V or example, we may like to develop a model to predict the salary of college graduates with 10 years of work experience, or the potential sales of a new product given its price. Many problems can be solved by linear regression, and even more can be tackled by applying transformaions to the variables so that a nonlinear problem can be converted to a linear one. For reasons of space, we cannot give a fully detailed treatment of regression. Instead, this section provides an intuitive introduction to the topic. By the end of this section, you will be familiar with the ideas of linear, multiple, and nonlinear regression, as well as generalized linear models.

Several software packages exist to solve regression problems. Examples include SAS (http://www.sas.com), SPSS (http://www.spss.com), and S-plus (http://www.mathspfr.com).

15.8 Linear and Multiple Regression

In linear regression, data are modeled using a straight line. Linear regression is the simplest form of regression. Bivariate linear regression models a random variable, Y (called a response variable), as a linear function of another random variable, X (called a predictor variable), that is,

Y=

Where the variance of Y is assumed to be constant, and a and b are regression coefficients specifying the Y-intercept and slope of the line, respectively. These coefficients can be solved for by the method of least squares, which minimizes the error between the actual data and the estimate of the line.

19.9 Nonlinear Regression

"How can we model data that does not show a linear dependence? For example, what if a given response variable and predictor variables have a relationship that may be modeled by a polynomial function?" polynomial regression can be modeled by adding polynomial terms to the basic linear model. By applying transformations to the variables, we can convert the nonlinear model. By applying transformations to the variables, we can convert the nonlinear the model into a linear one that can then be solved by the method of least squares.

Some models are intractably nonlinear (such as the sum of exponential terms, for example) and cannot be converted to a linear model. For such cases, it may be possible to obtain least square estimates through extensive calculations on more complex formulae.

1910 Other Regression models

Linear regression is used to model continuous-valued functions. It is widely used, owing largely to its simplicity. "can if also be used to predict categorical labels?" generalized linear models represent the theoretical foundation on which linear regression can be applied to the modeling of categorical response variables. In generalized linear models, the variance of the response variable V is a function of the mean value of V, unlike in linear regression, where the variance of Y is constant. Common types of generalized linear models include logistic regression and Poisson linear function of a set of predictor variables. Count data frequently exhibit a Poisson distribution and are commonly modeled using Poisson regression.

Log-linear models approximate discrete multidimensional probability distributions. They may be used to estimate the probability value associated with data cube cells. For example, suppose we are given data for the attributes city, item, year, and sales, in the log-linear method, all attributes must be categorical; hence continuous-valued attributes (like sales) must first be discretized. The method can then be used to estimate the probability of each cell in the 4-D base cuboid for the given attributes, based on the 2-d cuboids for city and item, city and year, city and sales, and the 3-D cuboid for item, year, and sales. In this way, an iterative technique can be used to build higher- order data cubes from

lower-order ones. The technique scales up well to allow for many dimensions. Aside from prediction, the log-linear model is useful for data compression (since the smaller- order cuboids together typically occupy less space than the base cuboid) and data smoothing variations than cell estimates in the base cuboid).

19.11 Classifier Accuracy

Estimating classifier accuracy is important in that it allows one to evaluate how accurately a given classifier will label future data, that is, data on which the classifier has not been trained. For example, if data from previous sales arre used to train a classifier to predict customer purchasing behavior, we would like some estimate of how accurately the classifier can predict the purchasing behavior of future customers.

Estimating Classifier Accuracy

Using training data to derive a classifier and then to estimate the accuracy of the classifier can result in misleading overoptimistic estimates due to over- specialization of the learning algorithm(or model) to the data. Holdout and cross-validation are two common techniques for assessing classifier accuracy, based on randomly sampled partitions of the given data- in the holdout method, the given data are randomly partitioned into two independent sets, a training set and a test set. Typically, two thirds of the data are allocated to the training set, and the remaining one third is allocated to the test se. the training set is used to derive the classifier, whose accuracy is estimated with the test set. The estimate is pessimistic since only a portion of the initial data is used to derive the classifier. Random subsampling is a variation of the holdout method in which the holdout method is repeated k times. The overall accuracy estimate is taken as the average of the accuracies obtained from each iteration.

In k-fold cross-validation, the initial data are randomly partitioned into k mutually exclusive subsets or “folds,” S1,S2,…., Sk, each of approximately equal size. Training and testing is performed k times. In iteration I, the subset S, is reserved as the test se, and the remaining subsets are collectively used to train the classifier. That is, the classifier of the first iteration is trained on subsets S1,.. , Sk and tested on Si; the classifier of the second iteration is trained on subsets S2,.. ,Sk and tested on Si; and so on. The accuracy estimate is the overall number of correct classifications from the k iterations, divided by the total number of samples in the initial data. In stratified cross-validation, the folds are stratified so that the class distribution of the samples in each fold is approximately the same as that in the initial data.

Other methods of estimating classifier accuracy include bootstrapping, which samples the given training instances uniformly with replacement, and leave-one-out) which is k-fold cross-validation with k set to 5, the number of initial samples.

19.12 Review Questions

1 Expalinabout K-Nearest Neighbor classifiers

2 Exaplain about Case- Based Reasoning

3 Expalin about Genetic Algorithms

4 Expalian about Rough Set Approach

5 Expalin about Fuzzy set Approaches

6 Expalian about Linear and Multiple Regression

19.13 References

[1]. Data Mining Techniques, Arun k pujari 1st Edition

[2] .Data warehousung,Data Mining and OLAP, Alex Berson ,smith.j. Stephen

[3].Data Mining Concepts and Techniques ,Jiawei Han and MichelineKamber

[4]Data Mining Introductory and Advanced topics, Margaret H Dunham PEA

[5] The Data Warehouse lifecycle toolkit , Ralph Kimball Wiley student Edition