Data Mining Study Notes

ShengXiang Wang

June 2020

1 Introduction

Data mining is a theory that includes Machine-Learning and Statistics.

2 Attribute

There are nominal attributes and numeric attributes in Data Mining.Nominal attributes are divided into binary attributes and ordinal attributes.Binary attributes are attributes that contain only zero and one.A Binary attributes is symmetrical when the zero and one have the same weight. Ordinal attributes has more values and the difference in values is meaningless.Numeric Attributes are divided into interval-scaled attributes and ratio-scaled attributes.The first one does not make sense in proportion, but the other one does. They make sense in difference

3 Data visualization

We can use Space-Filling curve such as Hilbert curve, gray code curve and z-curve to realize data visualization.Circle segment technique is also a good way to do that. In order to express the dense nature of subspace we have to find another method.
Parallel coordinates is a coordinate system where many x-axis are put together.when the number of x-axis is small, the visual effect is very good, but with the increase of the number of x-axis , there will be too many clusters,resulting in unclear situations.Icon-base visualization can display high-dimensional data with a small number of lines.Just like Chernoff faces and stick figure.

4 Dissimilarity

For binary attribute we use formula $\frac{x}{y}$ where x is the number of different attributes and y is number of attributes. Some times our binary attribute is asymmetry, in this case we use Jaccard-formula $\frac{q}{q+d}$ where q is the number of

both two object are true and d is the number of different attributes. For numeric attributes we use Minkowski distance to describe the dissimilarity. For mixing attributes we can using average.

5 Data Pretreatment

Data Pretreatment is divided into four step which contains data cleaning,data integration, data reduction and data transformation.
The most important part of data cleaning is deal with the missing values and the noise. If the proportion of missing data is not large, we can just drop them.When the number is large, we can also using average or median number, we can even using regression or Bayes method to deal with it. For noise data, we can using regression or outlier analysis method to deal with.
Data integration is difficult, because we have to consider that the same attributes in different data set may correspond to different name , same attributes are related , duplicate or redundant and even conflict.We can use k-square detection, pearson product or covariance to solve it.
In general, we using DWT , PCA , greedy or sampling to reduce data.
data transformation generally refers to orthogonalization or normalization, we use $\frac{x-min}{max-min}, \frac{x-\overline{x}}{\sigma}$ or $\frac{x}{10^t}$ where t is an order of magnitude.

6 Classification

6.1 Decision tree

Decision tree is a tree and each node divided the data set of its sub-tree into two parts according to a feature. How to select the feature becomes an interesting thing. The ID3,C4.5,CART have chosen different ways. Sometimes our tree will be very big, this time you need to pruning, sometimes our data set is so large that it cannot be loaded into memory.

6.1.1 Feature selection

Information entropy ID3 uses information entropy to divide sub-trees. Information entropy has nothing to do with the amount of data, but related to the confounding of the classes in the data.It's expression is equation (1) where D is the data set and p_i is the probability that a piece of data in D belongs to class i.

$$Information(D) = \sum_i p_i * \log_2 p_i \tag{1}$$

if we choose the feature A to divide the sub-trees, the how to compute the remain information, then we get this equation (2)

$$Information'(D_A) = \sum_i \frac{|D_i|}{|D|} Information(D_i) \tag{2}$$

ID3 select the feature A to maximum $Gain(D_A)$ in (3) as much as possible.

$$Gain(D_A) = Information(D) - Information'(D_A) \tag{3}$$

gain radio The information entropy gain method is not suitable in some case,it is more inclined to divide the data set into many categories such as using feature id which is unique.The C4.5 select gain radio to divide data set. So we need to consider what kind of division is a good division. We can't divide too uneven or too many categories.Let's analyze this function $f(x) = x \log_2 x$ below at Figure 1.We can using this function to limit the number of categories. The

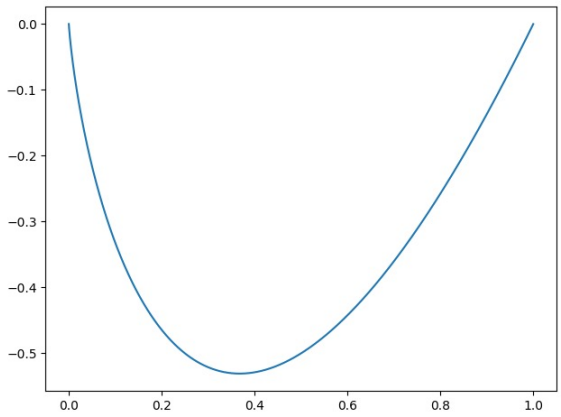

Figure 1: function: $y = x \log_2 x$

limit function is below,look at (4)

$$Radio(D_A) = \frac{Gain(D_A)}{-\sum_i \frac{|D_{Ai}|}{|D_A|} \log_2 \frac{|D_{Ai}|}{|D_A|}} \tag{4}$$

Gini Another idea is to use Gini index rather than equation (1) in CART,Just like equation (5)

$$Gini(D) = 1 - \sum_i p_i^2 \tag{5}$$

6.1.2 Pruning

Prepruning Using saliency analysis,information gain and gini index to evaluate whether a node is is worth dividing.

Postpruning After we get a decision tree,we can still postprune it, If cutting the sub-tree can improve accuracy or make decision tree simple.

6.1.3 Big Data Set

Rain forest using rain forest A Framework for Fast Decision Tree Construction of Large Dataset.

6.2 Naive Bayesian

Naive Bayesian is a simple and good classification and is most accurate of all method in theoretically if there is no assumption of conditional independence.Naive Bayesian using Bayesian theorem in equation (6) where X is the attributes and C is the class

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \tag{6}$$

sometime $P(X|C)$ is difficult to calculate we suppose that the attribute in X is conditional independence, then $P(X|C) = P(X_1|C)P(X_2|C)...P(X_n|C)$.

6.3 IF THEN

IF THEN is a classifier of logical reasoning type, and We can get this classifier from most of other classifier, sometime according to the IF THEN we can find that a X can be classified to multiple categories , We can give all rule a weight to solve it.Another way to get rule is by using sequential covering algorithm. This algorithm study a rule in each turns and then erase all data which is been covered by this rule until no data remain,but which rule we should choose in this turn? We can use information entropy or first order inductive learner.FOIL using the equation below.p and n is positive tuple and negative tuple in a rule and when we change to another rule, the gain is FOIL_Gain

$$FOIL\_Gain = p'(\log_2 \frac{p'}{p'+n'} - \log_2 \frac{p}{p+n}) \tag{7}$$

6.3.1 IF THEN pruning

For a rule if the value of equation below can be increase by delete a condition in the rule , the we delete it.

$$FOIL\_Pruning(R) = \frac{p-n}{p+n} \tag{8}$$

6.4 Model evaluation and selection

6.4.1 Evaluation Metric

metric	expression
accuracy	$\frac{TP+TN}{P+N}$
error rate	$\frac{FP+FN}{P+N}$
sensitivity	$\frac{TP}{P}$
specificity/recall	$\frac{TN}{N}$
precision	$\frac{TP}{TP+FP}$
F1	$\frac{2*precision*recall}{precision+recall}$
F_β	$\frac{(1+\beta^2)*precision*recall}{\beta^2*precision+recall}$

We also pay attention to calculation speed which is the cost of calculation, robustness which is the ability to deal with noise, scalability which is the performance on large data, and interpretability which is the reason of classification.

6.4.2 Selection

holdout we should divide our data into two parts,training set and validation set.

k-fold cross-validation we divide our data into k parts, and Perform training k times with each turn using a part as validation set and the remain parts as training set.

bootstrap Unlike the above division which is sampling without replacement, bootstrap will put back after each time sampling.

632Bootstrap We will get 63.2% of data set if we sample n times in a data set with n tuple.then the remain 36.8% will be the validation set.

ROC With the threshold of divide a tuple to positive classification from probability increasing, the number of $\frac{TP}{P}$ is increasing, and the number of $\frac{FP}{N}$ is decreasing. IF we draw them in the coordinate axis, the ROC curve appears.

6.4.3 Combination

bagging We can train may model, then we can choose the result which is the most number of model output when predicting a classification of a tuple.

boosting We can give weight to every model.

adaptive boosting The data on the training set often cannot be fitted by a classifier, So we give the error tuple more weight, so that the next classifier will pay more attention to them.At last we will get many classifier and we use their error rate to calculate their weight.

Random forest We can choose many subset of the feature, and then build many decision trees on them. Another Random forest is using random Linear combination of all feature.