1 Decision tree learning

Decision tree learning uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. It is one of the predictive modelling approaches used in statistics, data mining and machine learning. More descriptive names for such tree models are classification trees or regression trees. In these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels.

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

Classification tree analysis is when the predicted outcome is the class to which the data belongs. Regression tree analysis is when the predicted outcome can be considered a real number (e.g. the price of a house, or a patients length of stay in a hospital).

The term Classification And Regression Tree (CART) analysis is an umbrella term used to refer to both of the above procedures, first introduced by Breiman et al.[3] Trees used for regression and trees used for classification have some similarities - but also some differences, such as the procedure used to determine where to split.[3]

Some techniques, often called ensemble methods, construct more than one decision tree:

Bagging decision trees, an early ensemble method, builds multiple decision trees by repeatedly resampling training data with replacement, and voting the trees for a consensus prediction.[4] A Random Forest classifier uses a number of decision trees, in order to improve the classification rate. Boosted Trees can be used for regression-type and classification-type problems.[5][6] Rotation forest - in which every decision tree is trained by first applying principal component analysis (PCA) on a random subset of the input features.[7]

Classification and regression trees (CART) is a non-parametric decision tree learning technique that produces either classification or regression trees, depending on whether the dependent variable is categorical or numeric, respectively.

Decision trees are formed by a collection of rules based on variables in the modeling data set:

Rules based on variables' values are selected to get the best split to differentiate observations based on the dependent variable Once a rule is selected and splits a node into two, the same process is applied to each "child" node (i.e. it is a recursive procedure) Splitting stops when CART detects no further gain can be made, or some pre-set stopping rules are met. (Alternatively, the data are split as much as possible and then the tree is later pruned.) Each branch of the tree ends in a terminal node. Each observation falls into one and exactly one terminal node, and each terminal node is uniquely defined by a set of rules.

A very popular method for predictive analytics is Leo Breiman's Random forests or derived versions of this technique like Random multinomial logit.

The purpose of the analysis is to learn how we can discriminate between the three types of flowers, based on the four measures of width and length of petals and sepals. Discriminant function analysis will estimate several linear combinations of predictor variables for computing classification scores (or probabilities) that allow the user to determine the predicted classification for each observation. A classification tree will determine a set of logical if-then conditions (instead of linear equations) for predicting or classifying cases instead:

The interpretation of this tree is straightforward: If the petal width is less than or equal to 0.8, the respective flower would be classified as Setosa; if the petal width is greater than 0.8 and less than or equal to 1.75, then the respective flower would be classified as Virginic; else, it belongs to class Versicol.

```
boxplot(iris$Petal.Width~iris$Species,horizontal=T,col=c("lightblue","pink","yellow"),for
abline(v=0.80,col="red",lty=2)
```

```
boxplot(iris$Petal.Width~iris$Species,horizontal=T,col=c("lightblue","pink","yellow"),for
abline(v=0.80,col="red",lty=2)
abline(v=1.75,col="red",lty=2)

Class=numeric(150)
for (i in 1:150)
  {
  if (iris$Petal.Width[i]<0.80) {
     Class[i]=1
   } else if (iris$Petal.Width[i]>1.75) {
     Class[i]=3
   }else Class[i]=2
  }
```

1.1 Regression Trees

The general approach to derive predictions from few simple if-then conditions can be applied to regression problems as well. This example is based on the data file Poverty, which contains 1960 and 1970 Census figures for a random selection of 30 counties.

The research question (for that example) was to determine the correlates of poverty, that is, the variables that best predict the percent of families below the poverty line in a county. A reanalysis of those data, using the regression tree analysis [and v-fold cross-validation, yields the following results:

Again, the interpretation of these results is rather straightforward: Counties where the percent of households with a phone is greater than 72% have generally a lower poverty rate. The greatest poverty rate is evident in those counties that show less than (or equal to) 72% of households with a phone, and where the population change (from the 1960 census to the 170 census) is less than -8.3 (minus 8.3).

These results are straightforward, easily presented, and intuitively clear as well: There are some affluent counties (where most households have a telephone), and those generally have little poverty. Then there are counties that are generally less affluent, and among those the ones that shrunk most showed the greatest poverty rate.

A quick review of the scatterplot of observed vs. predicted values shows how the discrimination between the latter two groups is particularly well "explained" by the tree model.