Data Mining Liberty dataset

using R Language

Data Analysis Report

Author: Janardhan Bonu

Date: May-13-2015 .

Submitted to Professor

Anil K. Maheshwari

Maharishi University of Management

Table of Contents

1. Introduction 3

2. Decision Tree: 3

1.1. Algorithm 3

1.2. Training Data 3

1.3. Testing Data 3

1.4. Steps 4

1.5. R Code 4

1.5.1. Create Decision Tree 4

1.5.2. Prediction using C50 tree 4

1.6. Decision Tree generated by C50 algorithm 4

1.7. C50 Imp 5

1.8. Summary 5

2. Regression 6

2.1. Compute the correlation 6

Regression Model that best predicts the revenue 8

Prediction: 8

Prediction 1: 8

3. Clustering 10

# Introduction

This document scope is to discuss below data mining techniques using R language.

|  |  |  |
| --- | --- | --- |
| **Learning Type** | **Technique type** | **Technique** |
| Supervised learning: Classification | Machine learning | Decision Trees |
| Supervised learning: Classification | Statistical learning | Regression |
| Unsupervised learning: Exploration | Machine learning | Cluster Analysis |

# Decision Tree:

A decision tree is a hierarchically structured and branched according to the classification that help to make decision logically.

## Algorithm

The C5.0 is an extension of C4.5 classification algorithm.

## Training Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CitySize | AvgIncome | LocalInvestors | LOHASAwareness | Decision |
| big | high | yes | high | yes |
| med | med | no | med | no |
| small | low | yes | low | no |
| big | high | no | high | yes |
| small | med | yes | high | no |
| med | high | yes | med | yes |
| med | med | yes | med | no |
| big | med | no | med | no |
| med | high | yes | low | no |
| small | high | no | high | yes |
| small | med | no | high | no |
| med | high | no | med | no |

## Testing Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CitySize | AvgIncome | LocalInvestors | LOHASAwareness | Decision |
| med | med | no | med | ? |

## Steps

1. Import training data set.
2. Invoke C50 algorithm on data set.
3. Plot tree
4. Import test data for prediction
5. Summary

## R Code

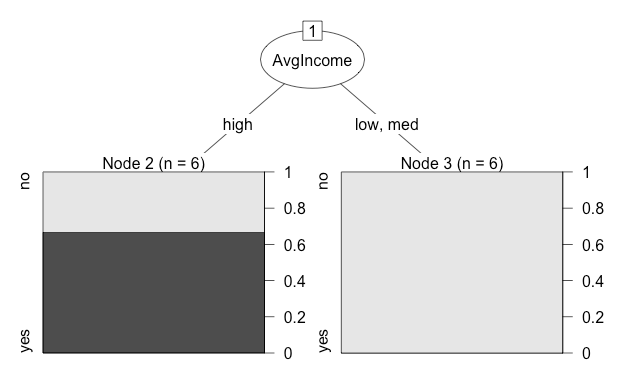
### Create Decision Tree

|  |
| --- |
| install.packages("C50")  library(C50)  libertydata <- read.csv("/Users/MUM/datamining/LibertyStoreDecisionTree.csv")  libertyc50tree <- C5.0(Decision~CitySize+AvgIncome+LocalInvestors+LOHASAwareness,data=libertydata)  plot(libertyc50tree) |

### Prediction using C50 tree

|  |
| --- |
| libertyPredict <- read.csv("/Users/janardhanbonu/OneDrive/MUM/datamining/LibertyStorePredict.csv")  predict(libertyc50tree, libertyPredict)  Result:  [1] **no**  Levels: no yes |

## Decision Tree generated by C50 algorithm



## C50 Imp

C5imp(libertyc50tree)

Overall

AvgIncome 100

CitySize 0

LocalInvestors 0

LOHASAwareness 0

## Summary

summary(libertyc50tree)

Call:

C5.0.formula(formula = Decision ~ CitySize + AvgIncome + LocalInvestors + LOHASAwareness,

data = libertydata)

C5.0 [Release 2.07 GPL Edition] Wed May 13 17:05:05 2015

-------------------------------

Class specified by attribute `outcome'

Read 12 cases (5 attributes) from undefined.data

Decision tree:

AvgIncome = high: yes (6/2)

AvgIncome in {low,med}: no (6)

Evaluation on training data (12 cases):

Decision Tree

----------------

Size Errors

2 2(16.7%) <<

(a) (b) <-classified as

---- ----

6 2 (a): class no

4 (b): class yes

Attribute usage:

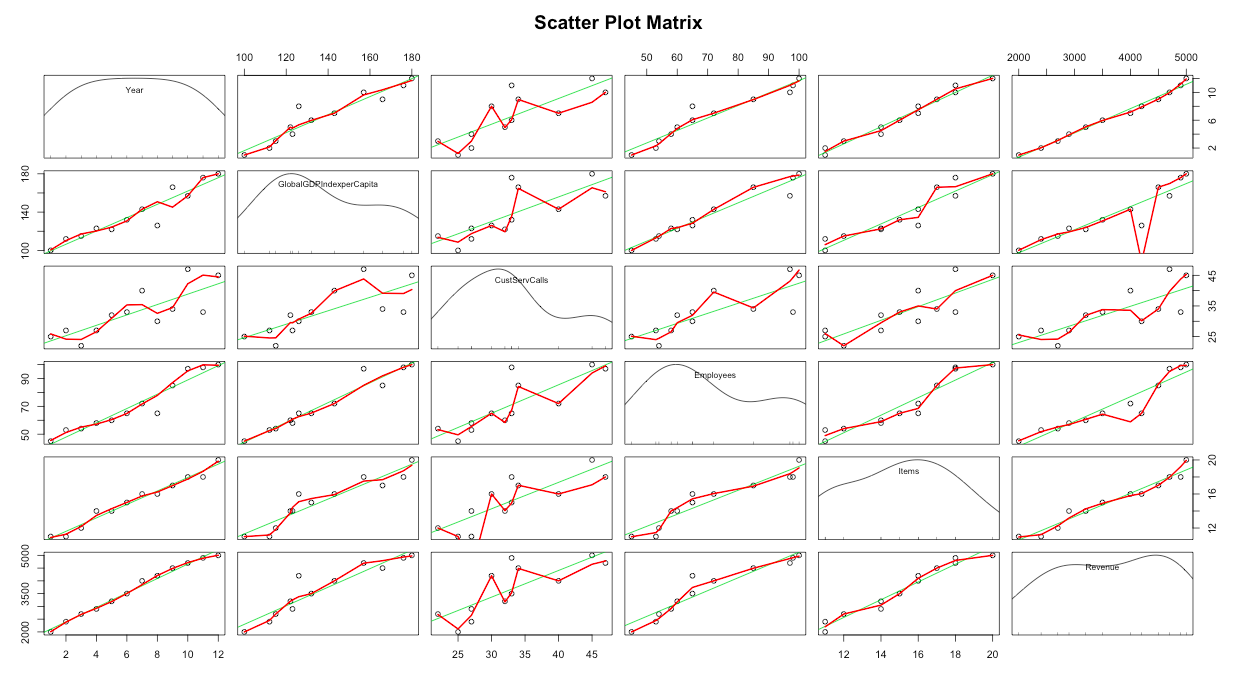
100.00% AvgIncome

## Regression

It is a statistical data mining technique. The goal is to fit a smooth well-defined curve to the data. The quality of fit of the curve to the data can be measured by a coefficient of correlation(r).

## Compute the correlation

1. Create the scatter plot for all variables.



2. Create the correlation for all variables

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | cor(libertyRegressionData) | | Year\*\* GlobalGDPIndexperCapita CustServCalls Employees Items Revenue | | Year 1.0000000 0.9480895 0.7912042 0.9592993 0.9869717 0.9941073 | | GlobalGDP  IndexperCapita 0.9480895 1.0000000 0.7510390 0.9734849 0.9371649 0.9359715 | | CustServCalls 0.7912042 0.7510390 1.0000000 0.8189303 0.8320012 0.7863883 | | Employees 0.9592993 **0.9734849**  0.8189303 1.0000000 0.9435137 0.9463931 | | Items 0.9869717 **0.9371649** 0.8320012 **0.9435137** 1.0000000 0.9769609 | | Revenue 0.9941073 **0.9359715** 0.7863883 **0.9463931 0.9769609** 1.0000000 | |

\*\*Year is not considered in the calculation since the calculation is to forecast the revenue for an year.

Observe the correlation

Strongly correlated variables

|  |  |  |
| --- | --- | --- |
|  | Variables | Correlation |
| 1 | Revenue Vs Items | **0.9769609** |
| 2 | Employees Vs GlobalGDP | **0.9734849** |
| 3 | Employees Vs Items | **0.9435137** |
| 4 | GlobalGDP Vs Items | **0.9371649** |

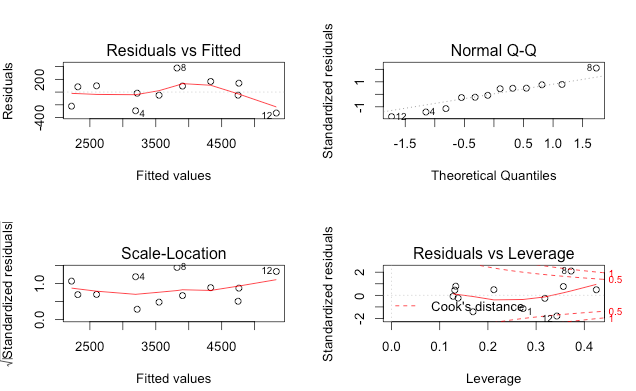
4. From the above correlation table, it is understood that Revenue Vs Items has the highest correlation.

5. Analyze the Multiple R-Square and Adjusted R-Squared

* Multiple R-squared: 0.9545, Adjusted R-squared: 0.9499– Revenue Vs Items
* Multiple R-squared: **0.96**, Adjusted R-squared: **0.951196** – Revenue Vs Items + Employees
* Multiple R-squared: 0.96, Adjusted R-squared: 0.945 - Revenue Vs Items + Employees+ GlobalGDP

From the above 3 summaries it is understandable that ‘Revenue Vs Items + Employees’ is providing the right one.

## Regression Model that best predicts the revenue



### Prediction:

### Prediction 1:

predict(model, list(Year=13,GlobalGDPIndexperCapita=190, CustServCalls=50,Employees=100,Items=25))

6691.044

(OR)

predict(model, list(Employees=100,Items=25))

6691.044

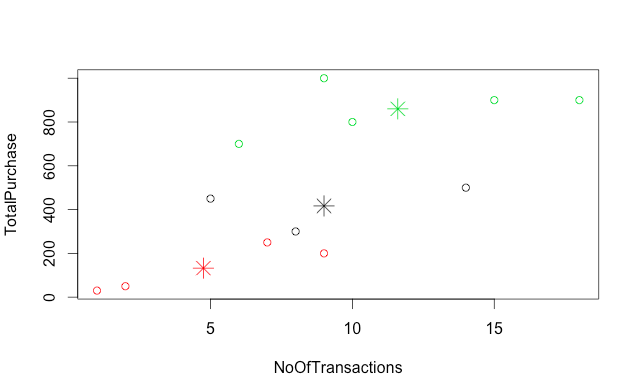
R Code

|  |
| --- |
| # Install car package  install.packages(c("car", "gvlma", "MASS", "leaps"))  library(car)  # Import liberty data  libertyRegressionData <- read.csv("/Users/janardhanbonu/OneDrive/MUM/datamining/LibertyRegressionData.csv")  names(libertyRegressionData)  # Scatter plot  scatterplotMatrix(libertyRegressionData, spread=FALSE, lty.smooth=2,main="Scatter Plot Matrix")  # Correlation  cor(libertyRegressionData)  fit <- lm(Revenue~Items, data=libertyRegressionData)  plot(fit)  summary(fit)  fit <- lm(Revenue~Items+Employees, data=libertyRegressionData)  plot(fit)  summary(fit)  fit <- lm(Revenue~Items+Employees+GlobalGDPIndexperCapita, data=libertyRegressionData)  plot(fit)  summary(fit)  # Best fit model  model <- lm(Revenue~Items+Employees, data=libertyRegressionData)  abline(model)  plot(model)  # Prediction  predict(model, list(Year=13,GlobalGDPIndexperCapita=190, CustServCalls=50, Employees=100,Items=25))  predict(model, list(Employees=100,Items=25)) |

# Clustering

|  |  |  |  |
| --- | --- | --- | --- |
| **CustId** | **NoOfTransactions** | **TotalPurchase** | **Income** |
| 1 | 5 | 450 | 90 |
| 2 | 10 | 800 | 82 |
| 3 | 15 | 900 | 77 |
| 4 | 2 | 50 | 30 |
| 5 | 18 | 900 | 60 |
| 6 | 9 | 200 | 45 |
| 7 | 14 | 500 | 82 |
| 8 | 8 | 300 | 22 |
| 9 | 7 | 250 | 90 |
| 10 | 9 | 1000 | 80 |
| 11 | 1 | 30 | 60 |
| 12 | 6 | 700 | 80 |

1. What is the right number of customer segments for Liberty ?
2. What are the Centroids ?



R Code

|  |
| --- |
| \*\*\*Note: The below code cleaning  libertyClusteringData <- read.csv("/Users/janardhanbonu/OneDrive/MUM/datamining/libertystore/LibertyClustering.csv")  km <- kmeans(libertyClusteringData,3)  km  km <- kmeans(libertyClusteringData)  km <- kmeans(libertyClusteringData,1)  plot(km)  km <- kmeans(libertyClusteringData,3)  km  km <- kmeans(libertyClusteringData,1)  km1 <- kmeans(libertyClusteringData,1)  km2 <- kmeans(libertyClusteringData,2)  km3 <- kmeans(libertyClusteringData,3)  summary(km1)  summary(km2)  summary(km3)  km1  km2  km3  km3 <- kmeans(libertyClusteringData[-1],3)  km3  plot(km3)  table(libertyClusteringData$Income, km3$cluster)  plot(libertyClusteringData[c("NoOfTransactions", "TotalPurchase")], col = km2$cluster)  par(mfrow = c(1,1))  plot(libertyClusteringData[c("NoOfTransactions", "TotalPurchase")], col = km2$cluster)  plot(libertyClusteringData[c("NoOfTransactions", "TotalPurchase", "Income")], col = km2$cluster)  plot(libertyClusteringData[c("NoOfTransactions", "TotalPurchase")], col = km2$cluster)  points(km3$centers[, c("NoOfTransactions", "TotalPurchase")],col = 1:3, pch = 8, cex = 2)  plot(libertyClusteringData[c("NoOfTransactions", "TotalPurchase")], col = km3$cluster)  points(km3$centers[, c("NoOfTransactions", "TotalPurchase")],col = 1:3, pch = 8, cex = 2) |