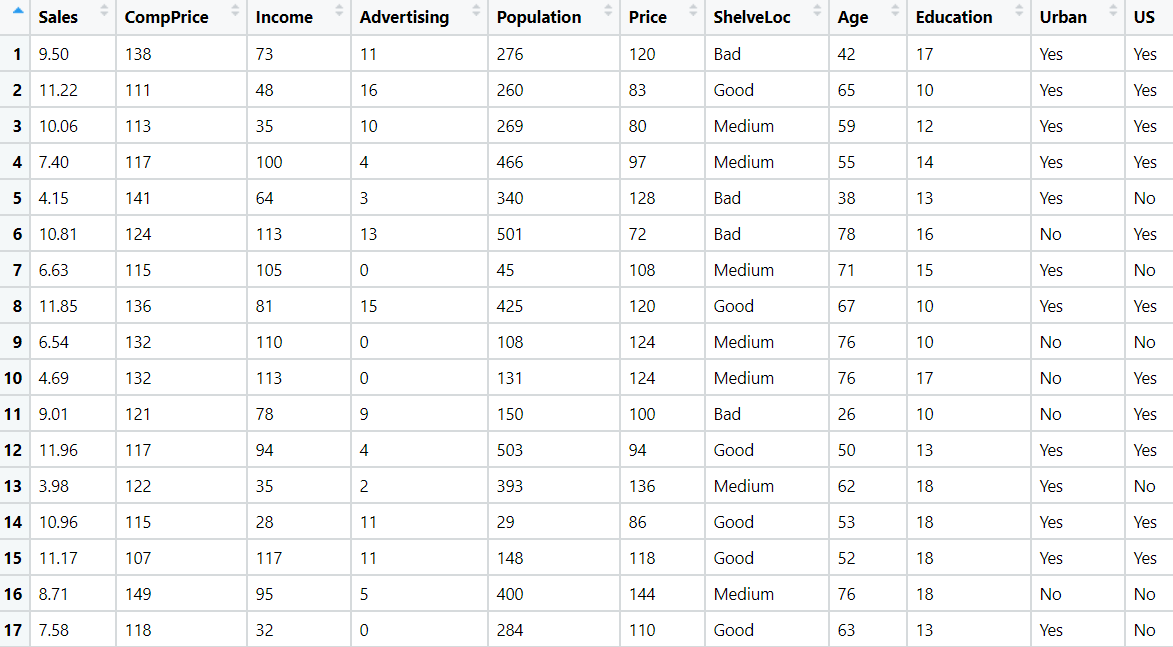
# Topic(s): Decision Tree & Random Forest

1.) A cloth manufacturing company is interested to know about the segment or attributes contributing to high sale. Approach - A decision tree & random forest model can be built with target variable 'Sale' (we will first convert it into categorical variable) & all other variables will be independent in the analysis.



**Ans:**

Analyzing the business problem

Target Variable = Sales

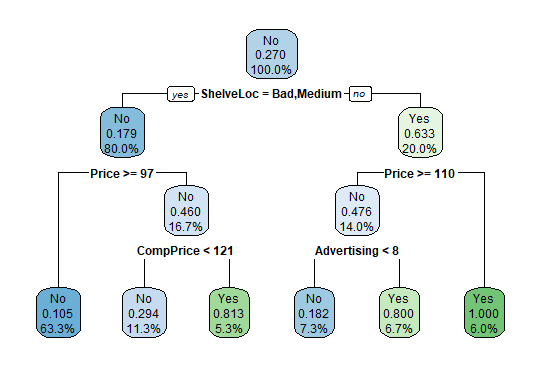
Independent Variables = Other Factors like Income, Adv, price, age

* As the sales is a continuous variable, so we need to discretization of the variable into 2 level factors as sales < 9 as one level factor and sales >= 9 another factor

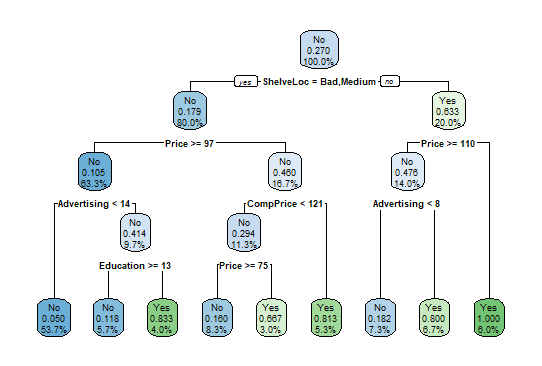
**Model Building:**

* Next step the data will be shuffled to make a division of test and train data randomly
* Need to check the proportions of the split data, so that the data split is done almost accurate or not
* Comp\_model <- C5.0(Comp\_train[, -11], Comp\_train$Salesdisc)
* Summary (Comp\_model) # shelve loc and Price has highest information gain
* The train and test accuracy are 0.95 and 0.71 which shows the model is overfit, no to get the right fit we use pruning technique.
* Model Building using C5.0, the above is the plot of decision tree which shows shelveloc has been chosen as root node, it might have highest information gain

**Applying Pruning Technique:**

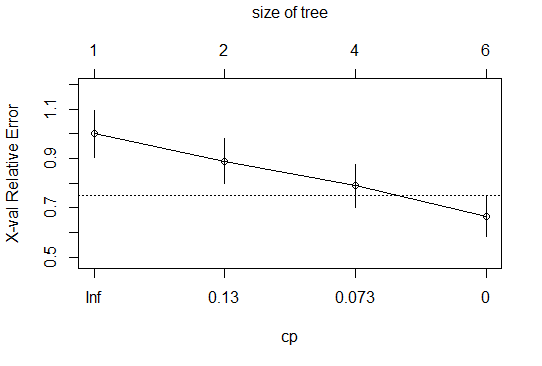


* After applying pruning technique, the test accuracy is 0.76

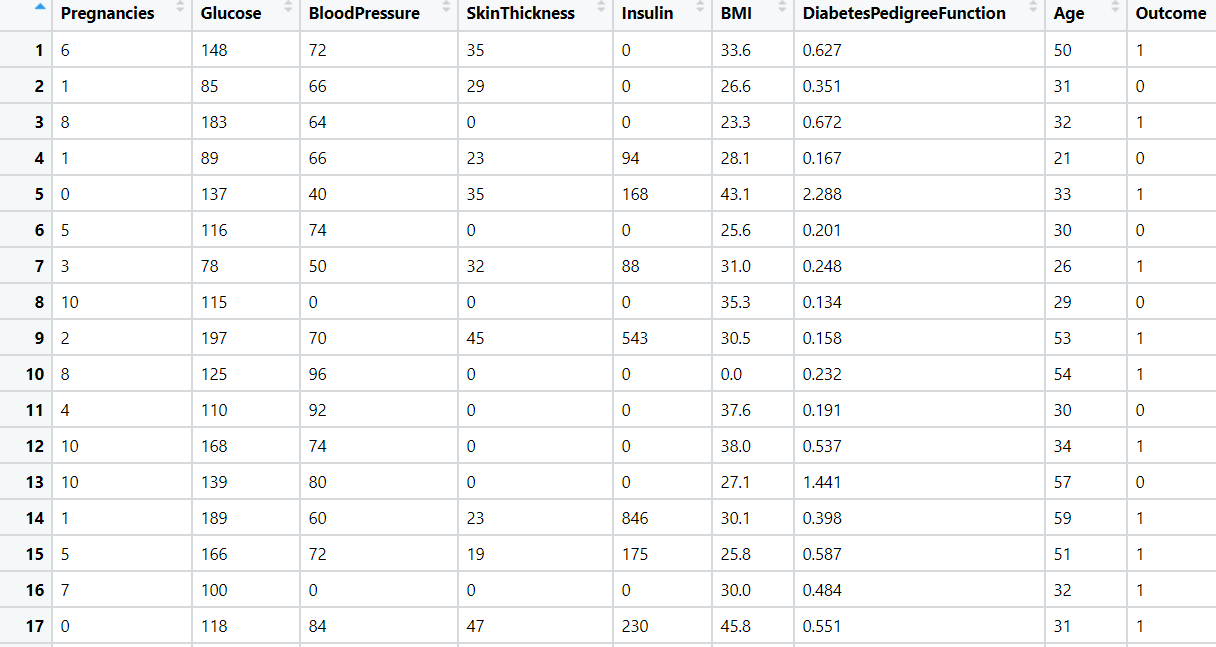


* The test accuracy has been decreased to 0.73 when cp = 0.02
* After applying all pruning techniques the test and train accuracy is 0.9 ,which is the right fit model

**Graph for Tree Size:**

****

2.) Divide the data (Diabetes) into training and test datasets and create a Random Forest Model to classify 'Class Variable'.



**Ans:**

Analyzing the business problem

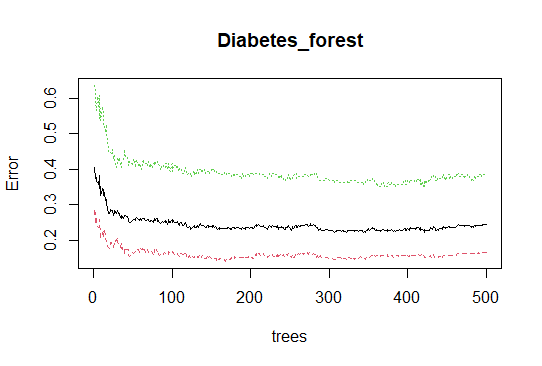
Target Variable = Class Variable

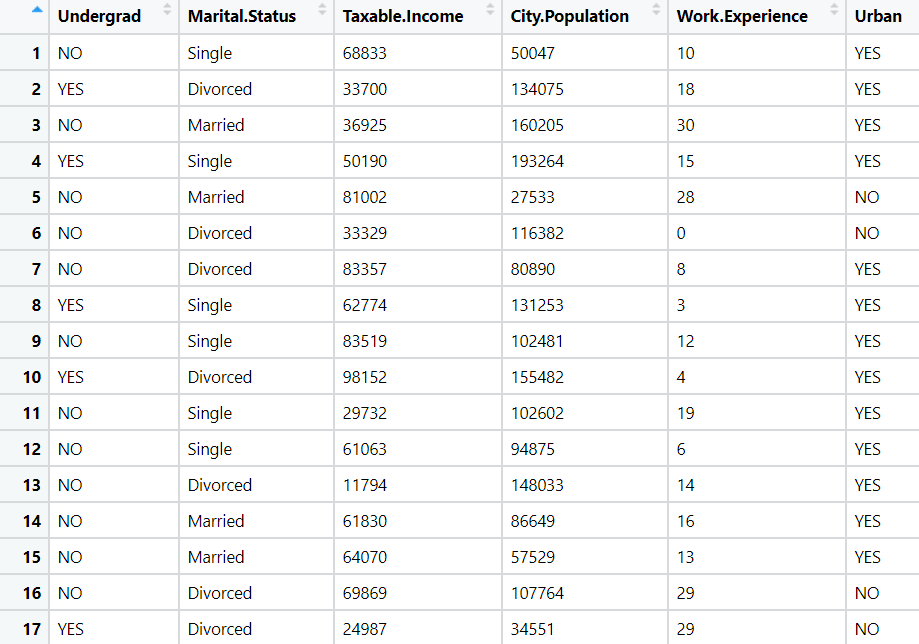
Independent Variables = Other Factors

* Normalizing data to make unit less and done Z scaling
* After Normalization divided the samples into train and test randomly

**Model Building:**

* Next step the data will be shuffled to make a division of test and train data randomly
* Need to check the proportions of the split data, so that the data split is done almost accurate or not
* Random Forest model is built to get the best accuracy
* The train and test accuracy are 0.75 and 0.78 which shows the model is right fit.



3.) Use decision trees & random forest algorithm to prepare a model on fraud data, treating those who have taxable\_income <= 30000 as "Risky" and others are "Good".

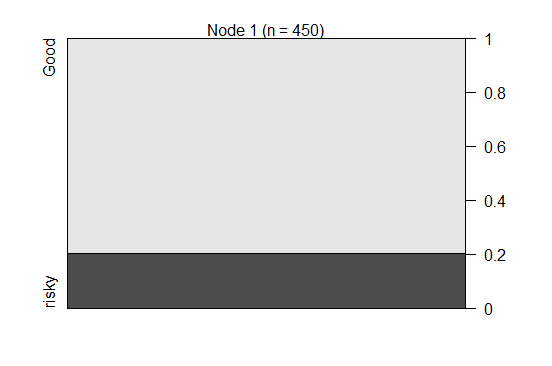
**Ans:**

Analyzing the business problem

Target Variable = Fraud check with the targeted variable income tax

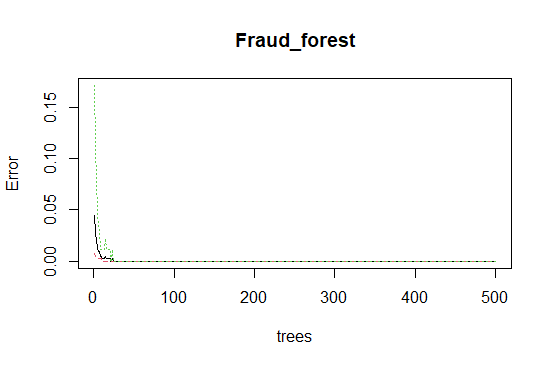
Independent Variables = Other Factors

* As the Income tax is a continuous variable, so we need to discretization of the variable into 2 level factors as tax <= 30000 as one level factor and tax > 30000 another factor
* Normalizing data to make unit less and done Z scaling
* After Normalization divided the samples into train and test randomly
* Next step the data will be shuffled to make a division of test and train data randomly
* Need to check the proportions of the split data, so that the data split is done almost accurate or not
* Build the model using c5.0 and calculate the test and train accuracy



* The train and test accuracy are 0.78 and 0.79 which shows the model is right fit.

**Building Random Forest Model:**



* The model gives the best accuracy

**Hints:**

1. Business Problem
   1. Objective
   2. Constraints (if any)
2. Data Pre-processing

2.1 Data cleaning, Feature Engineering, EDA etc.

1. Model Building
   1. Partition the dataset
   2. Model(s) - Reasons to choose any algorithm
   3. Model(s) Improvement steps
   4. Model Evaluation
   5. Python and R codes
2. Deployment

4.1 Deploy solutions using R shiny and Python Flask.

1. Result Share the benefits/impact of the solution - how or in what way the business (client) gets benefit from the solution provided.

**Note:**

1. For each assignment the solution should be submitted in the format
2. Research and Perform all possible steps for improving the model(s) accuracy Ex: Feature Engineering, Hyper Parameter tuning etc.
3. All the codes (executable programs) are running without errors
4. Documentation of the module should be submitted along with R & Python codes, elaborating on every step mentioned here