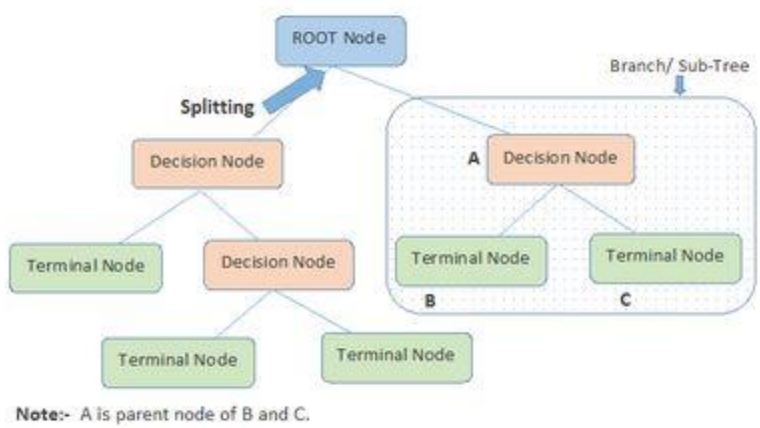
Foundations of Algorithm and Data Structure Presentation
Decision Tree

I. Introduction:

Tree based learning algorithm is one of the best supervised learning methods for the classification. Decision tree can handle both continuous and categorical variables as well as linear and non linear data relationships. The output is relatively easier and more intuitive for the general audience than the other mathematically complex models.

Decision tree model split the population or sample into two or more sets based on most significant splitter or differentiator in input variables in association with the target variable.



Basic Structure of Decision Tree:

-Root Node: It represents entire population or sample and this further gets divided into two or more subsets

-Splitting: It is a process of dividing a node into two or more sub-nodes

-Decision Node: When a sub-node splits into further sub-nodes, then it is called decision node

-Leaf/Terminal Node: Nodes do not split is called Leaf or Terminal node

-Pruning: When we remove sub-nodes of a decision node. this process is called pruning. It is opposite process of splitting.

-Branch/Sub-Tree: A sub section of entire tree is called branch or sub-tree

-Parent and Child Node: A node, which is divided into sub-nodes is called parent node of sub-nodes where as sub-nodes are child of parent node.

II. Statistical Analysis Case Study (Insurance Customer Data):

Here is a dataset of the customers who bought (or didn't buy) the insurance product. We want to know what attributes are likely to influence their customers' purchase decision. Decision Tree classification is one of the simpliest ways to identify the key features in association with the target variable, which is buy decision in this scenario.

For the simplicity of the presentation (and for the sake of time), I use the statistical analysis language called "R" and its decision tree package of "rpart", instead of Python that I have used through this course.

II-1. Descriptive Statistics (as a part of EDA):

Training data: Insurance product customer data

Target: Buy_Insurance variable (YES or NO)

Feature: 30 predictor variables except Target

```
##Descriptive
customers = read.csv("C:/Users/hirotak/Desktop/R/sample_customers.csv")
#dim(customers)
#head(customers)
summary(customers)

##   CUSTOMER_ID        LAST          FIRST          STATE           REGION
##  CU100   :    1   JUDE    :  4   BRYSON :  4   NY      :343   Midwest  :220
##  CU10006:    1   VAL     :  4   COYLE  :  4   CA      :235   NorthEast:375
##  CU10011:    1   ALVA    :  3   HOGUE  :  4   MI      :168   South    : 69
##  CU10012:    1   BOYCE   :  3   BRANCH :  3   FL      : 36   Southwest: 57
##  CU10020:    1   CALEB   :  3   CASH   :  3   DC      : 32   West     :294
##  CU10025:    1   CAMERON:  3   DICKENS:  3   MN      : 26
##  (Other):1009   (Other):995   (Other):994   (Other):175
##  SEX                PROFESSION  BUY_INSURANCE      AGE
##  F:344   Programmer/Developer:137   No :742       Min.   : 0.00
##  M:671   IT Staff            : 89   Yes:273       1st Qu.:27.00
##          Nurse               : 54                 Median :36.00
##          Clerical            : 35                 Mean   :38.19
##          Not specified       : 34                 3rd Qu.:48.00
##          Cashier             : 32                 Max.   :84.00
##          (Other)             :634
##   HAS_CHILDREN        SALARY       N_OF_DEPENDENTS CAR_OWNERSHIP
##  Min.   :0.0000   Min.   : 37572   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.: 60804   1st Qu.:1.000   1st Qu.:1.0000
##  Median :1.0000   Median : 64173   Median :1.000   Median :1.0000
##  Mean   :0.5113   Mean   : 65103   Mean   :1.993   Mean   :0.9468
##  3rd Qu.:1.0000   3rd Qu.: 68392   3rd Qu.:3.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :109943   Max.   :6.000   Max.   :1.0000
##
##  HOUSE_OWNERSHIP  TIME_AS_CUSTOMER  MARITAL_STATUS CREDIT_BALANCE
##  Min.   :0.0000   Min.   :1.000    DIVORCED:286    Min.   :     0
```

```
##  1st Qu.:1.0000   1st Qu.:1.000    MARRIED :327    1st Qu.:      0
##  Median :1.0000   Median :2.000    OTHER   : 11    Median :      0
##  Mean   :0.8049   Mean   :2.429    SINGLE  :347    Mean   :   2234
##  3rd Qu.:1.0000   3rd Qu.:3.000    WIDOWED : 44    3rd Qu.:      0
##  Max.   :2.0000   Max.   :5.000                    Max.   :170498
##
##    BANK_FUNDS     CHECKING_AMOUNT   MONEY_MONTLY_OVERDRAWN
##  Min.   :    0   Min.   :   25.0   Min.   :32.16
##  1st Qu.:    0   1st Qu.:   25.0   1st Qu.:53.06
##  Median :  500   Median :   25.0   Median :53.24
##  Mean   : 2640   Mean   : 1055.8   Mean   :53.71
##  3rd Qu.: 2900   3rd Qu.:  228.5   3rd Qu.:53.81
##  Max.   :36000   Max.   :23476.0   Max.   :73.61
##
##  T_AMOUNT_AUTOM_PAYMENTS MONTHLY_CHECKS_WRITTEN MORTGAGE_AMOUNT
##  Min.   :     0.0        Min.   : 0.000         Min.   :    0
##  1st Qu.:   191.5        1st Qu.: 1.000         1st Qu.:  176
##  Median :   623.0        Median : 3.000         Median : 1100
##  Mean   :  4980.3        Mean   : 4.311         Mean   : 2066
##  3rd Qu.:  2322.5        3rd Qu.: 5.000         3rd Qu.: 3000
##  Max.   :499362.0        Max.   :18.000         Max.   :45000
##
##   N_TRANS_ATM      N_MORTGAGES      N_TRANS_TELLER  CREDIT_CARD_LIMITS
##  Min.   :0.000   Min.   :0.0000   Min.   :0.000   Min.   : 500
##  1st Qu.:1.000   1st Qu.:1.0000   1st Qu.:1.000   1st Qu.: 800
##  Median :3.000   Median :1.0000   Median :1.000   Median :1000
##  Mean   :2.827   Mean   :0.8049   Mean   :1.731   Mean   :1286
##  3rd Qu.:4.000   3rd Qu.:1.0000   3rd Qu.:3.000   3rd Qu.:1500
##  Max.   :8.000   Max.   :2.0000   Max.   :9.000   Max.   :5000
##
##  N_TRANS_KIOSK    N_TRANS_WEB_BANK       LTV              LTV_BIN
##  Min.   : 0.000   Min.   :    0    Min.   :    0   HIGH     :483
##  1st Qu.: 1.000   1st Qu.:  250    1st Qu.:18930   LOW      : 89
##  Median : 1.000   Median :  800    Median :23132   MEDIUM   :334
##  Mean   : 1.864   Mean   : 1450    Mean   :22452   VERY HIGH:109
##  3rd Qu.: 3.000   3rd Qu.: 1990    3rd Qu.:26335
##  Max.   :10.000   Max.   :45000    Max.   :43101
##
```

II-2. Decision Tree Classification Modeling:

R's rpart package runs the decision tree classification model to identify the key features that influence the target variables (the customers' buy decision).

Here is the model output:

```
##Decision Tree Classification Model
#install.packages("rpart")
library(rpart)
#model = rpart(BUY_INSURANCE ~ ., data = customers); model #raw model
```

```
model = rpart(BUY_INSURANCE ~ ., data = customers[,-1:-7], control = rpart.co
ntrol(maxdepth = 4)); model #cleaner model
```

```
## n= 1015
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 1015 273 No (0.73103448 0.26896552)
##    2) BANK_FUNDS< 270.5 429   7 No (0.98368298 0.01631702) *
##    3) BANK_FUNDS>=270.5 586 266 No (0.54607509 0.45392491)
##      6) CHECKING_AMOUNT>=158 235  46 No (0.80425532 0.19574468)
##       12) MONEY_MONTLY_OVERDRAWN< 54.26 184  21 No (0.88586957 0.11413043)
*
##       13) MONEY_MONTLY_OVERDRAWN>=54.26 51  25 No (0.50980392 0.49019608)
##         26) CHECKING_AMOUNT>=1991 28   5 No (0.82142857 0.17857143) *
##         27) CHECKING_AMOUNT< 1991 23   3 Yes (0.13043478 0.86956522) *
##      7) CHECKING_AMOUNT< 158 351 131 Yes (0.37321937 0.62678063)
##       14) CREDIT_BALANCE>=999 29   3 No (0.89655172 0.10344828) *
##       15) CREDIT_BALANCE< 999 322 105 Yes (0.32608696 0.67391304) *
```

The decision tree model identified the four key features associated with the target variable:

Bank_FUNDS

CHECKING_AMOUNT

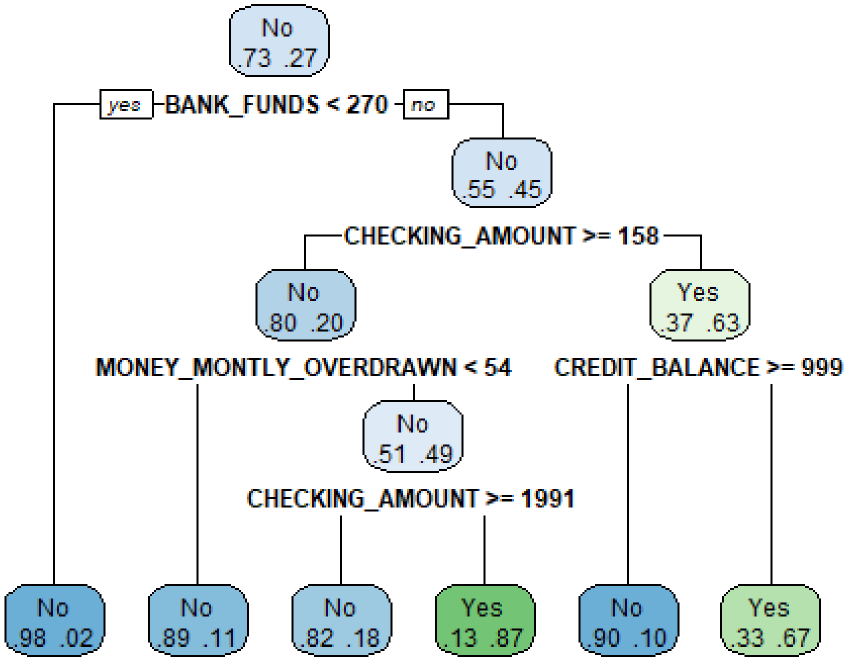
MONEY_MONTHLY_OVERDRAWN

CREDIT_BALANCE

II-3. Visualization:

Decision Tree visualization by rpart.plot package shows the logic to select the four key features clearly. One of the benefits to use decision tree for the classification modeling is easier and more intuitive to comprehend the output than other mathematically complex models.

```
##Decision Tree Visualization
#install.packages("rpart.plot")
library(rpart.plot)
rpart.plot(model, extra = 4)
```



III. Conclusion:

Pros:

-Easier and more intuitive to comprehend outputs than other models. (Writing code from scratch is hard.)

-Easy to implement due to the availability of library packages

-It can handle non-linear relationship well unlike regression model

-It can be used for the data imputation

-It can be used for both categorical and continuous variables

Cons:

-It is hard to comprehend the output as the tree grows

-Overfitting issue

IV. Reference:

1. https://www.analyticsvidhya.com/blog/2016/04/complete-tutorial-tree-based-modeling-scratch-in-python/

2. http://qiita.com/nkjm/items/e751e49c7d2c619cbeab