```
# CART classification
library(rpart)
library(rpart.plot)
bank.df<-read.csv("UniversalBank.csv", header= T)
str(bank.df)
bank.df <- bank.df[ , -c(1, 5)]  # Drop ID and zip code columns.

# partition
set.seed(1)
train.index <- sample(c(1:dim(bank.df)[1]), dim(bank.df)[1]*0.6)
train.df <- bank.df[train.index, ]
valid.df <- bank.df[-train.index, ]

# classification tree
default.ct <- rpart(Personal.Loan ~ ., data = train.df, method = "class")
length(default.ct$frame$var[default.ct$frame$var == "<leaf>"])

# plot tree
prp(default.ct, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10)

library(rattle)
fancyRpartPlot(default.ct)
# predict
predict(default.ct,valid.df)

# set argument type = "class" in predict() to generate predicted class membership.

default.ct.point.pred.train <- predict(default.ct,train.df,type = "class")
# generate confusion matrix for training data
library(caret)
library(ggplot2)
confusionMatrix(default.ct.point.pred.train, as.factor(train.df$Personal.Loan))

# generate confusion matrix for validation data
default.ct.point.pred.valid <- predict(default.ct,valid.df,type = "class")
confusionMatrix(default.ct.point.pred.valid, as.factor(valid.df$Personal.Loan))

### Tree Pruning

deeper.ct <- rpart(Personal.Loan ~ ., data = train.df, method = "class",
                   cp = 0, minsplit = 1)
# count number of leaves
length(deeper.ct$frame$var[deeper.ct$frame$var == "<leaf>"])
# plot tree
prp(deeper.ct, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10,
    box.col=ifelse(deeper.ct$frame$var == "<leaf>", 'gray', 'white'))


# argument xval refers to the number of folds to use in rpart's built-in
# cross-validation procedure
# argument cp sets the smallest value for the complexity parameter.
cv.ct <- rpart(Personal.Loan ~ ., data = train.df, method = "class",
               cp = 0.00001, minsplit = 5, xval = 5)
# use printcp() to print the table.
printcp(cv.ct)

# prune by lower cp
pruned.ct <- prune(cv.ct,cp = 0.0169697)
length(pruned.ct$frame$var[pruned.ct$frame$var == "<leaf>"])
prp(pruned.ct, type = 1, extra = 1, split.font = 1, varlen = -10)

fancyRpartPlot(pruned.ct)
```

```
# CART regression
car.df <- read.csv("ToyotaCorolla.csv")
#car.df<-read.csv(file.choose(), header= T)
str(car.df)
# preprocess
set.seed(1)
train.index <- sample(c(1:dim(car.df)[1]), dim(car.df)[1]*0.6)
valid.index <- setdiff(c(1:dim(car.df)[1]), train.index)
train.df <- car.df[train.index, ]
valid.df <- car.df[valid.index, ]
# regression tree:
tr <- rpart(Price ~  Age_08_04 + KM + Fuel_Type +
              HP + Automatic + Doors + Quarterly_Tax +
              Mfr_Guarantee + Guarantee_Period + Airco +
              Automatic_airco + CD_Player + Powered_Windows +
              Sport_Model + Tow_Bar, data = train.df,
            method = "anova", minbucket = 1, maxdepth = 30, cp = 0.001)
prp(tr)
# errors
library(forecast)
library(ggplot2)
accuracy(predict(tr, train.df), train.df$Price)
accuracy(predict(tr, valid.df), valid.df$Price)
# shallower tree
tr.shallow <- rpart(Price ~  Age_08_04 + KM + Fuel_Type +
                      HP + Automatic + Doors + Quarterly_Tax +
                      Mfr_Guarantee + Guarantee_Period + Airco +
                      Automatic_airco + CD_Player + Powered_Windows +
                      Sport_Model + Tow_Bar, data = train.df,
                    method = "anova")
prp(tr.shallow)
accuracy(predict(tr.shallow, train.df), train.df$Price)
accuracy(predict(tr.shallow, valid.df), valid.df$Price)
#Classification Tree
#Model for categorical price
bins <- seq(min(car.df$Price),
            max(car.df$Price),
            (max(car.df$Price) - min(car.df$Price))/20)
Binned_Price <- .bincode(car.df$Price,
                         bins,
                         include.lowest = TRUE)
Binned_Price <- as.factor(Binned_Price)
train.df$Binned_Price <- Binned_Price[train.index]
valid.df$Binned_Price <- Binned_Price[valid.index]

tr.binned <- rpart(Binned_Price ~  Age_08_04 + KM + Fuel_Type +
                     HP + Automatic + Doors + Quarterly_Tax +
                     Mfr_Guarantee + Guarantee_Period + Airco +
                     Automatic_airco + CD_Player + Powered_Windows +
                     Sport_Model + Tow_Bar, data = train.df)
prp(tr.binned)

# predict price
new.record <- data.frame(Age_08_04 = 77, KM = 117000, Fuel_Type = "Petrol",HP =
110,Automatic = 0, Doors = 5,Quarterly_Tax = 100,Mfr_Guarantee = 0,Guarantee_Period =
3,Airco = 1,Automatic_airco = 0,CD_Player = 0,Powered_Windows = 0,Sport_Model =
0,Tow_Bar = 1)
# regression model
price.tr <- predict(tr, newdata = new.record)

# classification model
price.tr.bin <- bins[predict(tr.binned, newdata = new.record, type = "class")]
```