

Machine Learning-03 Tree Based Models

Created by H. M. Samadhi Chathuranga Rathnayake

```
#Classification

#Classification trees
setwd("D:\\Workshops\\R Programming for Data Science Workshop\\Part 04 -
Machine Learning\\Datasets")
df=read.csv("loan.CSV")
head(df)

df=df[complete.cases(df),] #Removing missing values
summary(df)

nrow(df)

#Factorize the categorical variables
str(df)

df$Gender=factor(df$Gender)
df$Married=factor(df$Married)
df$Education=factor(df$Education)
df$Self_Employed=factor(df$Self_Employed)
df$Property_Area=factor(df$Property_Area)
df$Loan_Status=factor(df$Loan_Status)

str(df)

#Fitting the classification tree

set.seed(7777)
trainID=sample(1:nrow(df),0.8*nrow(df))
train=df[trainID,]
test=df[-trainID,]

library(rpart)
library(rpart.plot)

fit_tree=rpart(Loan_Status~.,data=train,method = "class")
summary(fit_tree)

rpart.plot(fit_tree)

y_pred=predict(fit_tree,test,type="class")
y_actual=test$Loan_Status

conf=table(y_actual,y_pred)
conf
```

```

accuracy=sum(diag(conf))/length(y_pred)*100
accuracy

#Pruning

fit_tree_prun=rpart(Loan_Status~.,data=train,method = "class",control =
rpart.control(maxdepth =3))
summary(fit_tree_prun)

rpart.plot(fit_tree_prun)

y_pred=predict(fit_tree_prun,test,type="class")
y_actual=test$Loan_Status

conf=table(y_actual,y_pred)
conf

accuracy=sum(diag(conf))/length(y_pred)*100
accuracy

#Random Forest Classification

library(randomForest)

rfmodel=randomForest(Loan_Status~.,data=train)

y_actual=test$Loan_Status
y_pred=predict(rfmodel,test,type = "class")

conf=table(y_actual,y_pred)
conf

accuracy=sum(diag(conf))/length(y_pred)*100
accuracy

#Regression

#Regression trees
setwd("D:\\Workshops\\R Programming for Data Science Workshop\\Part 04 -
Machine Learning\\Datasets")
df=read.csv("Boston.CSV")
head(df)

set.seed(7777)
trainID=sample(1:nrow(df),0.8*nrow(df))
train=df[trainID,]
test=df[-trainID,]

fit_tree=rpart(medv~.,data=train)
summary(fit_tree)

```

```
rpart.plot(fit_tree)

y_pred=predict(fit_tree,test)
y_actual=test$medv

MSE=mean((y_actual-y_pred)^2)
RMSE=sqrt(MSE)
RMSE

#Random Forest Regression

rfmodel=randomForest(medv~.,data=train)

y_actual=test$medv
y_pred=predict(rfmodel,test)

MSE=mean((y_actual-y_pred)^2)
RMSE=sqrt(MSE)
RMSE
```