Prediction-Assignment-Writeup

SARVESH

16 AUG 2020

#I-) Essential Packages IscaretInstalled <- require(“caret”) if(!IscaretInstalled){ install.packages(“caret”) library(“caret”) }

IsrandomForestInstalled <- require(“randomForest”) if(!IsrandomForestInstalled){ install.packages(“randomForest”) library(“randomForest”) }

IsRpartInstalled <- require(“rpart”)
if(!IsRpartInstalled){ install.packages(“rpart”) library(“rpart”) }

Set seed for reproducability set.seed(20000)

#II-) Data Processing Load the data trainUrl <- “http://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv”
testUrl <- “http://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv”

#III-)Cleaning data Load data to memory. training <- read.csv(url(trainUrl), na.strings=c(“NA”,“#DIV/0!”,"“))
testing <- read.csv(url(testUrl), na.strings=c(”NA“,”#DIV/0¡‘,”"))

Remove variables with near zero variance training<-training[,colSums(is.na(training)) == 0] testing <-testing[,colSums(is.na(testing)) == 0]

Remove columns that are not predictors, which are the the seven first columns training <-training[,-c(1:7)]
testing <-testing[,-c(1:7)]

The data after cleaning

dim(training)

#IV-) Cross-validation In order to get out-of-sample errors, split the training data in training (75%) and testing (25%) data) subsets: inTrain <- createDataPartition(y=training$classe, p=0.75, list=FALSE)
NEOTraining <- training[inTrain,] NEOTesting <- training[-inTrain,]
dim(NEOTraining) dim(NEOTesting)

#V-) Prediction Models

DECISION TREE Fit model on NEOTraining data fitDT <- rpart(classe ~ ., data=NEOTraining, method=“class”)

Use model to predict class in validation set (NEOTesting) predictionDT <- predict(fitDT, NEOTesting, type = “class”)

Estimate the errors of the prediction algorithm in the Decision Tree model confusionMatrix(NEOTesting$classe, predictionDT)

RANDOM FOREST Fit model on NEOTraining data fitRF <- randomForest(classe ~ ., data=NEOTraining, method=“class”)

Use model to predict class in validation set (NEOTesting) predictionRF <- predict(fitRF, NEOTesting, type = “class”)

Estimate the errors of the prediction algorithm in the Random Forest confusionMatrix(NEOTesting$classe, predictionRF)

#VI-) TEST THE MODEL TO PREDICT 20 DIFFERENT TEST CASES # Perform prediction predictSubmission <- predict(fitRF, testing, type=“class”) predictSubmission