yacht_hydrodynamics

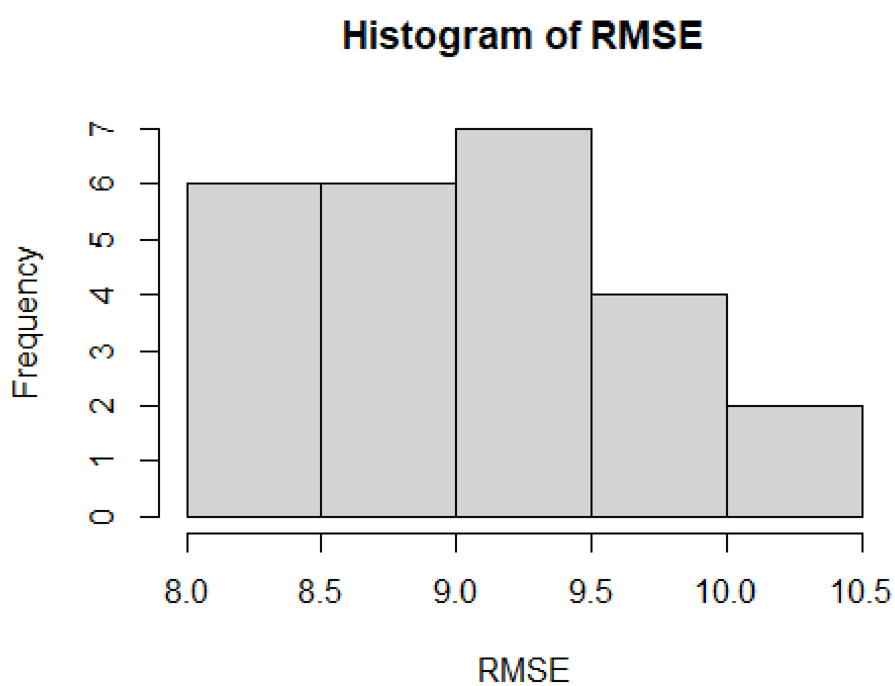
Shiva Sankar Modala

2023-02-10

```
## Installing the necessary packages for the problem ##

#install.packages('readr')     ## yacht_hydrodynamics.data is a large
dataset. So, I used readr package in handling the data
#install.packages('caret')     ## To use machine learning models, I used
caret to fit our model
#install.packages('ggplot2')   ## used ggplot2 for better visualizations of
data
#install.packages('lattice')   ## Lattice is used to implement the trellis
graphics for our data

# Loading the libraries
library(readr)
library(data.table)
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(ggplot2)
library(lattice)

# Reading the yacht_hydrodynamics.data as the table without the header
yacht_hydrodynamics = read.table("https://archive.ics.uci.edu/ml/machine-
learning-databases/00243/yacht_hydrodynamics.data", header = F)

# Assigning the column names for our dataset
names(yacht_hydrodynamics) = c("longitude","Prismatic","displacement","beam-
draught","beamlenght","fraude","residuary")
head(yacht_hydrodynamics)

##   longitude Prismatic displacement beam-draught beamlenght fraude
residuary
## 1      -2.3     0.568         4.78         3.99       3.17  0.125
0.11
## 2      -2.3     0.568         4.78         3.99       3.17  0.150
0.27
## 3      -2.3     0.568         4.78         3.99       3.17  0.175
0.47
## 4      -2.3     0.568         4.78         3.99       3.17  0.200
0.78
## 5      -2.3     0.568         4.78         3.99       3.17  0.225
```

```
1.18
## 6      -2.3     0.568         4.78         3.99       3.17  0.250
1.82

# Creating the data partition for our data having 80% our data for the
training. So the rest 20% is for testing.
#  I used the caret package to perform a 80/20 test-train split
cd = createDataPartition(y = yacht_hydrodynamics$residuary , p = 0.8, list =
FALSE)

# Separating the dataset for the train data
train_data = yacht_hydrodynamics[cd,]

# Separating the test data without the output label data.
test_data = yacht_hydrodynamics[-cd,]

# Applying the linear regression model for the dataset
# Applying the multiple linear regression
lm1 = lm(yacht_hydrodynamics$residuary~yacht_hydrodynamics$longitude +
yacht_hydrodynamics$Prismatic +
             yacht_hydrodynamics$displacement + yacht_hydrodynamics$`beam-
draught` + yacht_hydrodynamics$`beam-draught` +
             yacht_hydrodynamics$displacement + yacht_hydrodynamics$fraude,
              data = train_data)

# creating a function for the mean square error
mse = function(y, yt){
  return (mean((y - yt)^2))
}

# Applying the mean square error for the residuary and the fitted values for
the linear regression model.
msee = mse(yacht_hydrodynamics$residuary, lm1$fitted.values )
msee

## [1] 78.47651

cat("\n The MSE for the training data is = ", msee)

##
##  The MSE for the training data is =  78.47651

cat("\n The Root mean square error for the train data is = ", sqrt(msee))

##
##  The Root mean square error for the train data is =  8.858697

cat("\n The summary for the r-squared data for the linear model is  =
",summary(lm1)$r.sq)

##
##  The summary for the r-squared data for the linear model is  =  0.6574487
```

```
# train control specify the resampling scheme
# I used the caret package to perform a bootstrap from the full sample
dataset with N=1000 samples
train = trainControl(method = "boot", number = 1000)

lm2 = train(residuary~., data = train_data, method = "lm" )

# summary of the model
summary(lm2$resample$RMSE)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   8.243   8.679   9.306   9.109   9.478  10.337

summary(lm2$resample$Rsquared)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.5690  0.6301  0.6483  0.6457  0.6580  0.7025

# Plotting a histogram for the resampled data and the root mean square error
hist(lm2$resample$RMSE, xlab = "RMSE", main = "Histogram of RMSE")
```




```
# applying the mean for the resampled data as the mse2
mse2 = mean(lm2$resample$RMSE)^2
mse2

## [1] 82.96775

cat("\n Training MSE for the bootstrap model  is = ", mse2)
```

```
##
##  Training MSE for the bootstrap model  is =  82.96775

cat("\n Training RMSE for the bootstrap model is  ", mean(lm2$resample$RMSE))

##
##  Training RMSE for the bootstrap model is   9.108663

cat("\n Training Mean R-squared for the bootstrap model is
",mean(lm2$resample$Rsquared))

##
##  Training Mean R-squared for the bootstrap model is  0.6457281

predVals_boot = predict(lm2,test_data)

cat("\n From the above observations, there is no difference in performance
between the original and bootstrap models.")

##
##  From the above observations, there is no difference in performance
between the original and bootstrap models.
```