181058_DSLab_Quiz

1. Loading dataset using read.csv

```
df= read.csv('LungCapData2.csv')
head(df)
```

```
##   Age LungCap Height Gender Smoke
## 1   9   3.124   57.0 female    no
## 2   8   3.172   67.5 female    no
## 3   7   3.160   54.5 female    no
## 4   9   2.674   53.0   male    no
## 5   9   3.685   57.0   male    no
## 6   8   5.008   61.0 female    no
```

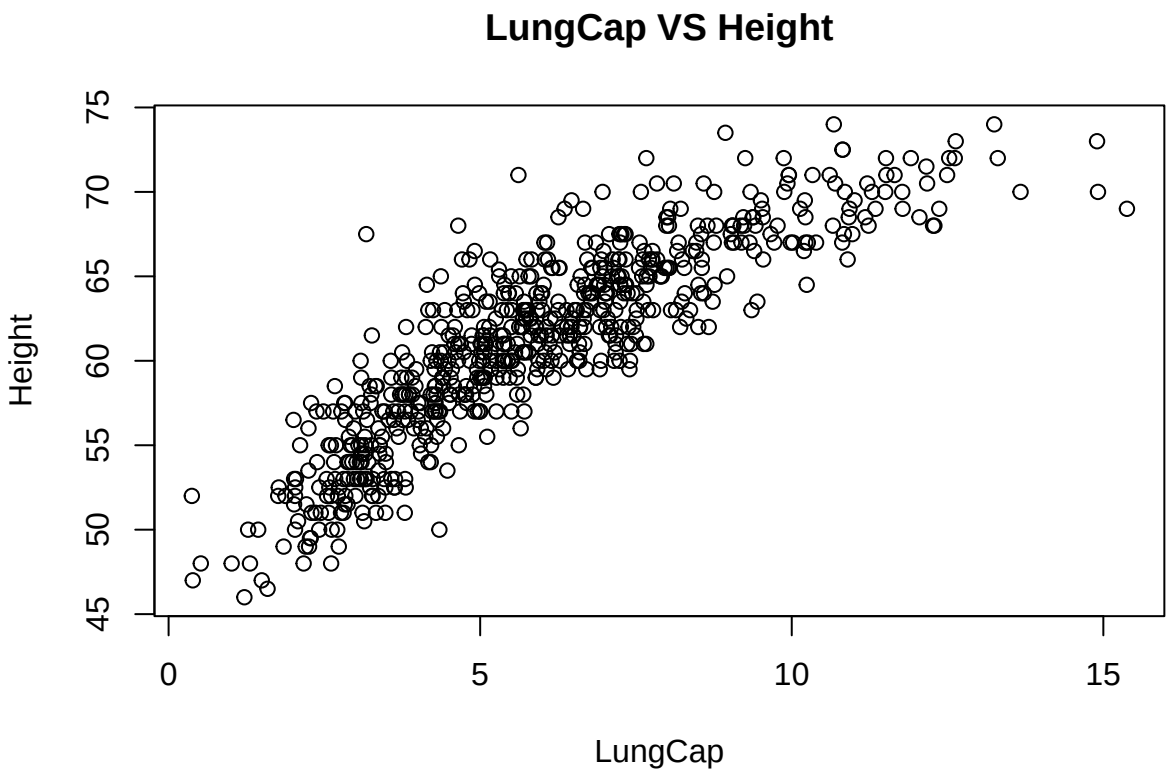
1.1. Printing the summary of df

```
summary(df)
```

```
##       Age            LungCap           Height         Gender
##  Min.   : 3.000   Min.   : 0.373   Min.   :46.00   Length:654
##  1st Qu.: 8.000   1st Qu.: 3.943   1st Qu.:57.00   Class :character
##  Median :10.000   Median : 5.643   Median :61.50   Mode  :character
##  Mean   : 9.931   Mean   : 5.910   Mean   :61.14
##  3rd Qu.:12.000   3rd Qu.: 7.356   3rd Qu.:65.50
##  Max.   :19.000   Max.   :15.379   Max.   :74.00
##    Smoke
##  Length:654
##  Class :character
##  Mode  :character
##
##
##
```

2. Making a plot

```
plot(df$LungCap, df$Height, main="LungCap VS Height", xlab="LungCap", ylab="Height")
```



> 3. Splitting the data in training and testing data

```
library(caTools)
set.seed(123)

split = sample.split(df, SplitRatio = 0.8)
trainingset <- subset(df, split == TRUE)
testset <- subset(df, split == FALSE)
paste("Training Set ->", dim(trainingset))
```

```
## [1] "Training Set -> 524" "Training Set -> 5"
```

```
paste("Test set ->", dim(testset))
```

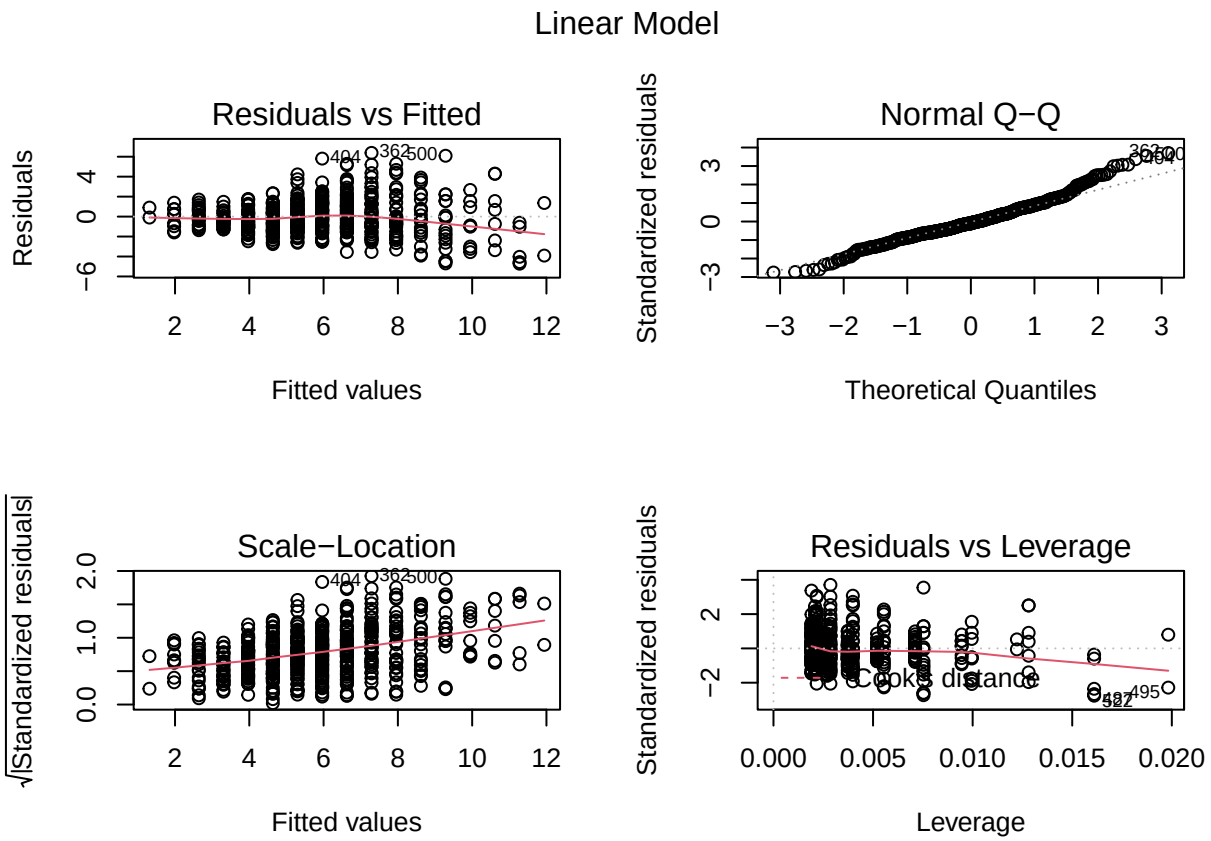
```
## [1] "Test set -> 130" "Test set -> 5"
```

4. Linear Regression on Lungcap vs Age on Training dataset

```
LungCapVSAgemodel = lm(trainingset$LungCap ~ trainingset$Age)
LungCapVSAgemodel
```

```
##
## Call:
## lm(formula = trainingset$LungCap ~ trainingset$Age)
##
## Coefficients:
##     (Intercept)  trainingset$Age
##         -0.6810           0.6643
```

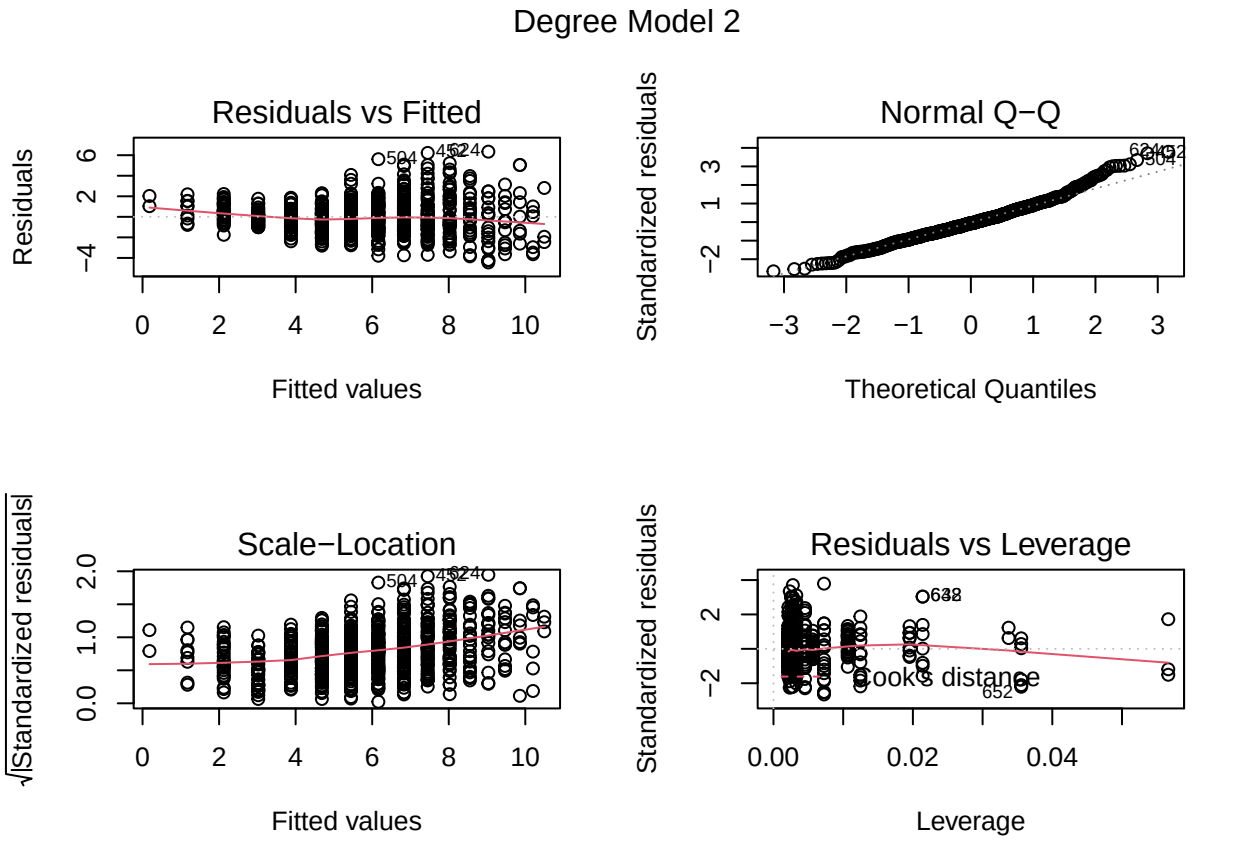
```
par(mfrow=c(2,2))
plot(LungCapVSAgemodel)
mtext("Linear Model",
      side = 3,
      line = - 1,
      outer = TRUE)
```




```
paste("Summary of fitted model ->", head(summary(LungCapVSAgemodel)))
```

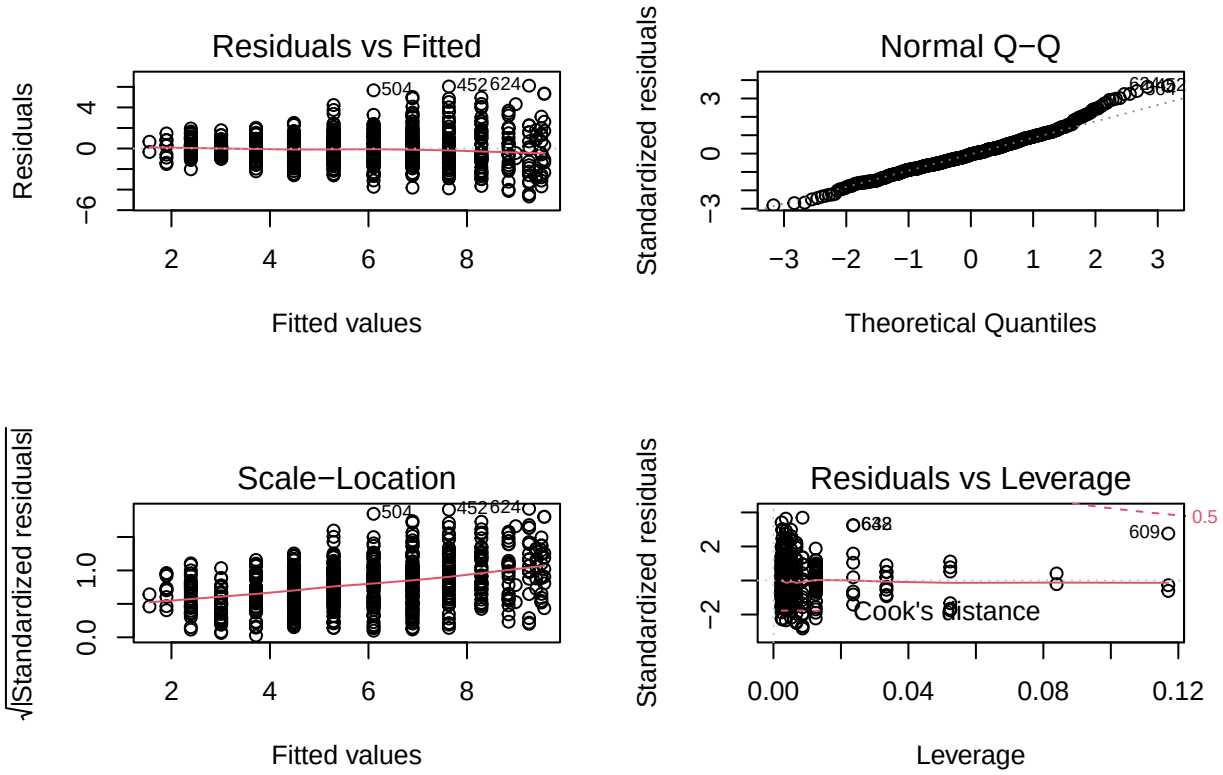
```
## [1] "Summary of fitted model -> lm(formula = trainingset$LungCap ~ trainingset$Age)"
## [2] "Summary of fitted model -> trainingset$LungCap ~ trainingset$Age"
## [3] "Summary of fitted model -> c(`1` = -2.17407594658005, `2` = -1.46173513475785, `3` = -0.8093943
```

```
DegreeModel2 <- lm(df$LungCap ~ poly(df$Age, 2), data = trainingset)
par(mfrow=c(2,2))
plot(DegreeModel2)
mtext("Degree Model 2",
      side = 3,
      line = - 1,
      outer = TRUE)
```



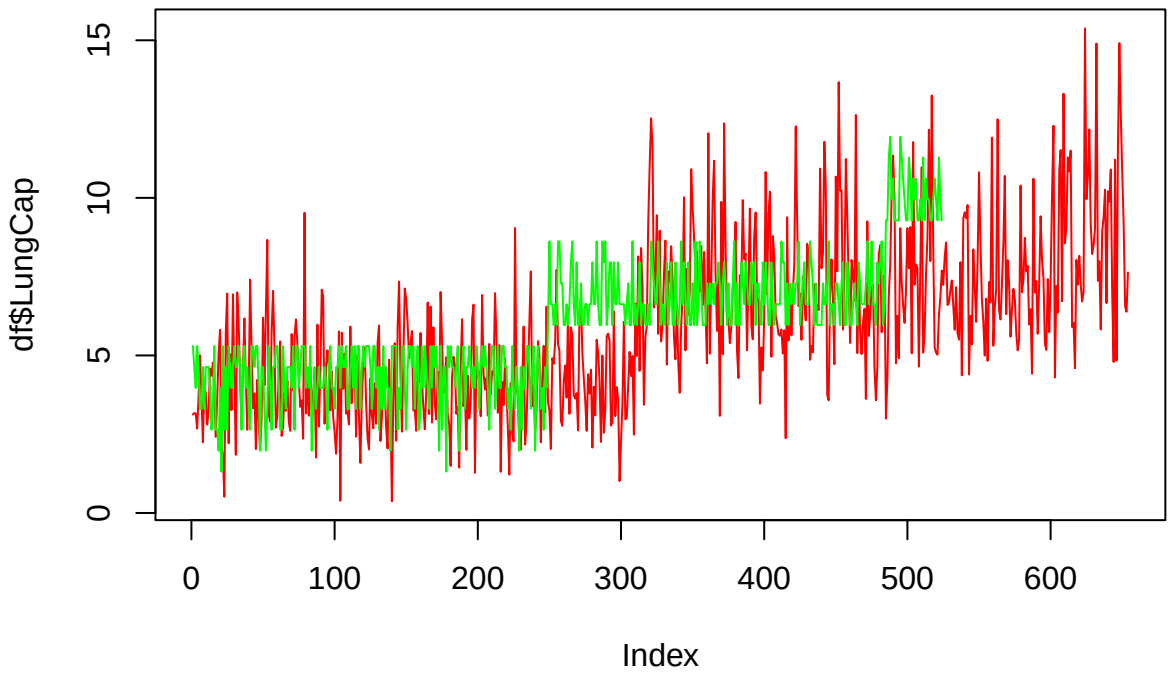

```
DegreeModel3 <- lm(df$LungCap ~ poly(df$Age, 3), data = trainingset)
par(mfrow=c(2,2))
plot(DegreeModel3)
mtext("Degree Model 3",
      side = 3,
      line = - 1,
      outer = TRUE)
```

Degree Model 3



Predicting the result using trained data

```
predicted_result_for_lungVsAge <- predict(LungCapVSAgemodel, testset)

res_LungvsAge <- residuals(LungCapVSAgemodel)
res_LungvsAge <- as.data.frame(res_LungvsAge)

final_res <- cbind(predicted_result_for_lungVsAge, df$LungCap)

colnames(final_res) <- c("Predicted","real")

final_res <- as.data.frame(final_res)

#plotting them
plot(df$LungCap, type = 'l', lty = 1.8, col = 'red')
lines(predicted_result_for_lungVsAge, col='green')
```




```
#checking accuracy
rmse <- sqrt(mean(predicted_result_for_lungVsAge - df$LungCap)^2)
paste("Accuracy is ", rmse)
```

```
## [1] "Accuracy is  0.340025983743368"
```