Breast Cancer
PREDICTION SYSTEM

Introduction 1

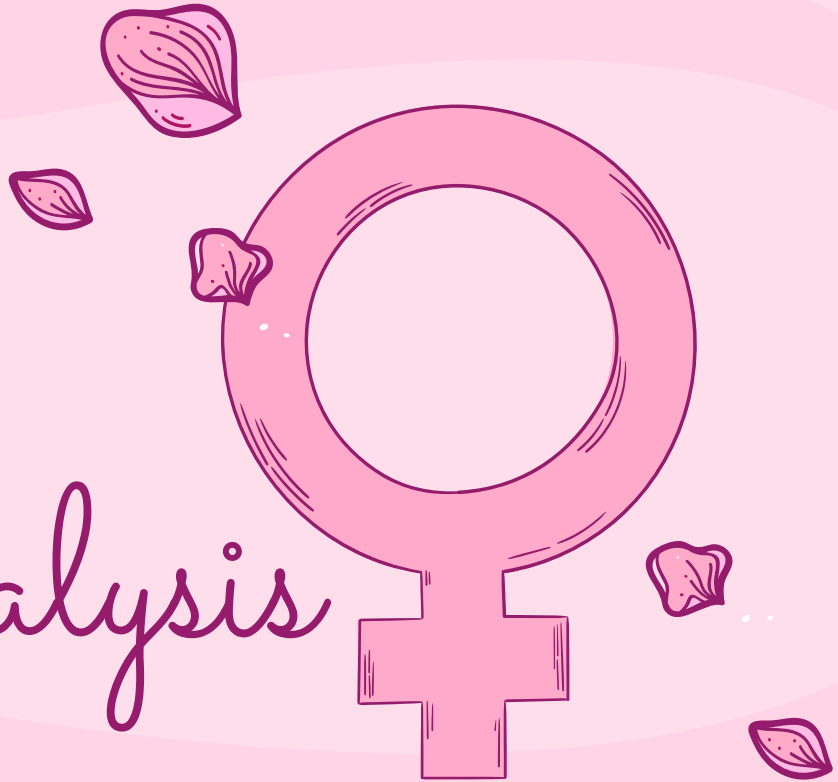
Introduction

Health is an essential aspect of everyone's life. Breast cancer is found in the body of male or female when the cells in the breast begin to grow out of control. These cells usually form a tumor and can be felt as a lump or could be seen on an x-ray. Cancer can be distinguished as benign, or either can be malignant (cancer).

To study the breast cancer, I have taken the dataset from the UCI repository. I will use Linear Model to help in to predict if patients have cancer or not.

2 Data Analysis

Exploratory Data Analysis



Data Loading

Breast Cancer excel file is loaded into the R Studio, and there weren't any null values in the dataset.

Some Basic Data Exploration :

- Dataset : Breast Cancer Dataset
- Source : UCI
- No. of columns : 31
- No .of Rows : 569

Reading the data

Code :

```
library("readxl")
breast_cancer_dataset <-read_excel("D:/breast cancer dataset.xlsx")
View(breast_cancer_dataset)
```

Output :

breast_cancer_dataset

Filter

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave.points_mean	symmetry_mean	fractal
1	17.990	10.38	122.80	1001.0	0.11840	0.27760	0.300100	0.147100	0.2419	
2	20.570	17.77	132.90	1326.0	0.08474	0.07864	0.086900	0.070170	0.1812	
3	19.690	21.25	130.00	1203.0	0.10960	0.15990	0.197400	0.127900	0.2069	
4	11.420	20.38	77.58	386.1	0.14250	0.28390	0.241400	0.105200	0.2597	
5	20.290	14.34	135.10	1297.0	0.10030	0.13280	0.198000	0.104300	0.1809	
6	12.450	15.70	82.57	477.1	0.12780	0.17000	0.157800	0.080890	0.2087	
7	18.250	19.98	119.60	1040.0	0.09463	0.10900	0.112700	0.074000	0.1794	
8	13.710	20.83	90.20	577.9	0.11890	0.16450	0.093660	0.059850	0.2196	
9	13.000	21.82	87.50	519.8	0.12730	0.19320	0.185900	0.093530	0.2350	
10	12.460	24.04	83.97	475.9	0.11860	0.23960	0.227300	0.085430	0.2030	

Supervised or Unsupervised:

It is Supervised dataset since all the columns are labeled.
This dataset is used for predictions using algorithms like linear models.

Checking for null values

Code :

```
sum(is.na(breast_cancer_dataset))
```

Output :

```
[1] 0
```

Code :

```
colnames(breast_cancer_dataset)[colSums(is.na(breast_cancer_dataset))>0]
```

Output :

```
character(0)
```

Count total missing values in each column of the data frame

Code :

```
sapply(breast_cancer_dataset,function(x) sum(is.na(x)))
```

Output :

```
            radius_mean            texture_mean          perimeter_mean
                      0                       0                       0
              area_mean         smoothness_mean        compactness_mean
                      0                       0                       0
         concavity_mean     concave.points_mean           symmetry_mean
                      0                       0                       0
 fractal_dimension_mean               radius_se              texture_se
                      0                       0                       0
           perimeter_se                 area_se           smoothness_se
                      0                       0                       0
         compactness_se            concavity_se       concave.points_se
                      0                       0                       0
            symmetry_se    fractal_dimension_se             radius_worst
                      0                       0                       0
          texture_worst         perimeter_worst               area_worst
                      0                       0                       0
       smoothness_worst        compactness_worst          concavity_worst
                      0                       0                       0
   concave.points_worst          symmetry_worst fractal_dimension_worst
                      0                       0                       0
              diagnosis
                      0
```

Prints the datatype of all the columns with column name

Code :

```
sapply(breast_cancer_dataset,function(x) typeof(x))
```

Output :

```
            radius_mean            texture_mean          perimeter_mean
               "double"                "double"                "double"
              area_mean         smoothness_mean        compactness_mean
               "double"                "double"                "double"
        concavity_mean     concave.points_mean           symmetry_mean
               "double"                "double"                "double"
fractal_dimension_mean               radius_se              texture_se
               "double"                "double"                "double"
          perimeter_se                 area_se           smoothness_se
               "double"                "double"                "double"
        compactness_se            concavity_se       concave.points_se
               "double"                "double"                "double"
           symmetry_se    fractal_dimension_se            radius_worst
               "double"                "double"                "double"
         texture_worst         perimeter_worst              area_worst
               "double"                "double"                "double"
      smoothness_worst       compactness_worst         concavity_worst
               "double"                "double"                "double"
  concave.points_worst          symmetry_worst fractal_dimension_worst
               "double"                "double"                "double"
              diagnosis
             "integer"
```

Statistics of the data

Code :

```
summary(breast_cancer_dataset)
```

Output :

```
> summary(breast_cancer_dataset)
  radius_mean      texture_mean   perimeter_mean
 Min.   : 6.981   Min.   : 9.71   Min.   : 43.79
 1st Qu.:11.700   1st Qu.:16.17   1st Qu.: 75.17
 Median :13.370   Median :18.84   Median : 86.24
 Mean   :14.127   Mean   :19.29   Mean   : 91.97
 3rd Qu.:15.780   3rd Qu.:21.80   3rd Qu.:104.10
 Max.   :28.110   Max.   :39.28   Max.   :188.50
   area_mean      smoothness_mean   compactness_mean
 Min.   : 143.5   Min.   :0.05263   Min.   :0.01938
 1st Qu.: 420.3   1st Qu.:0.08637   1st Qu.:0.06492
 Median : 551.1   Median :0.09587   Median :0.09263
 Mean   : 654.9   Mean   :0.09636   Mean   :0.10434
 3rd Qu.: 782.7   3rd Qu.:0.10530   3rd Qu.:0.13040
 Max.   :2501.0   Max.   :0.16340   Max.   :0.34540
 concavity_mean    concave.points_mean symmetry_mean
 Min.   :0.00000   Min.   :0.00000     Min.   :0.1060
 1st Qu.:0.02956   1st Qu.:0.02031     1st Qu.:0.1619
 Median :0.06154   Median :0.03350     Median :0.1792
 Mean   :0.08880   Mean   :0.04892     Mean   :0.1812
 3rd Qu.:0.13070   3rd Qu.:0.07400     3rd Qu.:0.1957
 Max.   :0.42680   Max.   :0.20120     Max.   :0.3040
 fractal_dimension_mean   radius_se        texture_se
 Min.   :0.04996        Min.   :0.1115   Min.   :0.3602
 1st Qu.:0.05770        1st Qu.:0.2324   1st Qu.:0.8339
 Median :0.06154        Median :0.3242   Median :1.1080
 Mean   :0.06280        Mean   :0.4052   Mean   :1.2169
 3rd Qu.:0.06612        3rd Qu.:0.4789   3rd Qu.:1.4740
```

Data Quality Assessment using Skimr

Code :

```
install.packages("skimr")
library("skimr")
breast_cancer_dataset %>% skim()
```

Output :

```
── Variable type: numeric ──────────────────────────────────────────────────────
  skim_variable           n_missing complete_rate      mean        sd
1 radius_mean                     0             1  14.1       3.52
2 texture_mean                    0             1  19.3       4.30
3 perimeter_mean                  0             1  92.0      24.3
4 area_mean                       0             1 655.      352.
5 smoothness_mean                 0             1   0.0964    0.0141
6 compactness_mean                0             1   0.104     0.0528
7 concavity_mean                  0             1   0.0888    0.0797
8 concave.points_mean             0             1   0.0489    0.0388
9 symmetry_mean                   0             1   0.181     0.0274
10 fractal_dimension_mean         0             1   0.0628    0.00706
11 radius_se                      0             1   0.405     0.277
12 texture_se                     0             1   1.22      0.552
13 perimeter_se                   0             1   2.87      2.02
14 area_se                        0             1  40.3      45.5
15 smoothness_se                  0             1   0.00704   0.00300
16 compactness_se                 0             1   0.0255    0.0179
17 concavity_se                   0             1   0.0319    0.0302
```

Output :

	p0	p25	p50	p75	p100	hist
1	6.98	11.7	13.4	15.8	28.1	
2	9.71	16.2	18.8	21.8	39.3	
3	43.8	75.2	86.2	104.	188.	
4	144.	420.	551.	783.	2501	
5	0.0526	0.0864	0.0959	0.105	0.163	
6	0.0194	0.0649	0.0926	0.130	0.345	
7	0	0.0296	0.0615	0.131	0.427	
8	0	0.0203	0.0335	0.074	0.201	
9	0.106	0.162	0.179	0.196	0.304	
10	0.0500	0.0577	0.0615	0.0661	0.0974	
11	0.112	0.232	0.324	0.479	2.87	
12	0.360	0.834	1.11	1.47	4.88	
13	0.757	1.61	2.29	3.36	22.0	
14	6.80	17.8	24.5	45.2	542.	
15	0.00171	0.00517	0.00638	0.00815	0.0311	
16	0.00225	0.0131	0.0204	0.0324	0.135	
17	0	0.0151	0.0259	0.0420	0.396	
18	0	0.00764	0.0109	0.0147	0.0528	

Statistics :

Code :

```
glimpse(breast_cancer_dataset)
```

Output :

```
> glimpse(breast_cancer_dataset)
Rows: 569
Columns: 31
$ radius_mean            <dbl> 17.990, 20.570, 19.690, 11.420, 20.290, 12.450, 18.2…
$ texture_mean           <dbl> 10.38, 17.77, 21.25, 20.38, 14.34, 15.70, 19.98, 20.…
$ perimeter_mean         <dbl> 122.80, 132.90, 130.00, 77.58, 135.10, 82.57, 119.60…
$ area_mean              <dbl> 1001.0, 1326.0, 1203.0, 386.1, 1297.0, 477.1, 1040.0…
$ smoothness_mean        <dbl> 0.11840, 0.08474, 0.10960, 0.14250, 0.10030, 0.12780…
$ compactness_mean       <dbl> 0.27760, 0.07864, 0.15990, 0.28390, 0.13280, 0.17000…
$ concavity_mean         <dbl> 0.30010, 0.08690, 0.19740, 0.24140, 0.19800, 0.15780…
$ concave.points_mean    <dbl> 0.14710, 0.07017, 0.12790, 0.10520, 0.10430, 0.08089…
$ symmetry_mean          <dbl> 0.2419, 0.1812, 0.2069, 0.2597, 0.1809, 0.2087, 0.17…
$ fractal_dimension_mean <dbl> 0.07871, 0.05667, 0.05999, 0.09744, 0.05883, 0.07613…
$ radius_se              <dbl> 1.0950, 0.5435, 0.7456, 0.4956, 0.7572, 0.3345, 0.44…
$ texture_se             <dbl> 0.9053, 0.7339, 0.7869, 1.1560, 0.7813, 0.8902, 0.77…
$ perimeter_se           <dbl> 8.589, 3.398, 4.585, 3.445, 5.438, 2.217, 3.180, 3.8…
$ area_se                <dbl> 153.40, 74.08, 94.03, 27.23, 94.44, 27.19, 53.91, 50…
$ smoothness_se          <dbl> 0.006399, 0.005225, 0.006150, 0.009110, 0.011490, 0.…
$ compactness_se         <dbl> 0.049040, 0.013080, 0.040060, 0.074580, 0.024610, 0.…
$ concavity_se           <dbl> 0.05373, 0.01860, 0.03832, 0.05661, 0.05688, 0.03672…
$ concave.points_se      <dbl> 0.015870, 0.013400, 0.020580, 0.018670, 0.018850, 0.…
$ symmetry_se            <dbl> 0.03003, 0.01389, 0.02250, 0.05963, 0.01756, 0.02165…
$ fractal_dimension_se   <dbl> 0.006193, 0.003532, 0.004571, 0.009208, 0.005115, 0.…
```

Print the no.of benign and no.of malignant cases

Code :

```
table(diagnosis)
```

Output :

```
diagnosis
  0   1
212 357
```

Finding mean of all the columns using group_by -> This helps us to identify the difference between malignant cases and
Benign cases -> here we observe that the mean values of all the columns for malignant are greater than benign
This helps us in the model differentiation

Code :

```
breast_cancer_dataset%>%group_by(diagnosis)%>%summarise_all("mean")
```

Output :

```
# A tibble: 2 × 31
  diagnosis radius_m…¹ textu…² perim…³ area_…⁴ smoot…⁵ compa…⁶ conca…⁷ conca…⁸ symme…⁹
      <int>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
1         0       17.5    21.6   115.     978.  0.103   0.145   0.161   0.0880   0.193
2         1       12.1    17.9    78.1    463.  0.0925  0.0801  0.0461  0.0257   0.174
# … with 21 more variables: fractal_dimension_mean <dbl>, radius_se <dbl>,
#   texture_se <dbl>, perimeter_se <dbl>, area_se <dbl>, smoothness_se <dbl>,
#   compactness_se <dbl>, concavity_se <dbl>, concave.points_se <dbl>,
#   symmetry_se <dbl>, fractal_dimension_se <dbl>, radius_worst <dbl>,
#   texture_worst <dbl>, perimeter_worst <dbl>, area_worst <dbl>,
#   smoothness_worst <dbl>, compactness_worst <dbl>, concavity_worst <dbl>,
#   concave.points_worst <dbl>, symmetry_worst <dbl>, …
# ℹ Use `colnames()` to see all variable names
```

T-test one sample

Code :

```
t.test(breast_cancer_dataset$radius_mean, mu=17)
```

Output :

```
> t.test(breast_cancer_dataset$radius_mean, mu=17)

        One Sample t-test

data:  breast_cancer_dataset$radius_mean
t = -19.445, df = 568, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 17
95 percent confidence interval:
 13.83712 14.41747
sample estimates:
mean of x
 14.12729
```

One way anova and Two way anova

Code :

```
a1 <- aov(radius_se~radius_mean,databreast_cancer_dataset)

summary(a1)

A2 <-aov(radius_se~radius_meanradius_worst,breast_cancer_dataset)
summary(a2)
```

Output :

```
> summary(a1)
             Df Sum Sq Mean Sq F value Pr(>F)
radius_mean   1  20.14  20.144   485.3 <2e-16 ***
Residuals   567  23.54   0.042
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> summary(a2)
              Df Sum Sq Mean Sq F value   Pr(>F)
radius_mean    1 20.144  20.144  537.82  < 2e-16 ***
radius_worst   1  2.337   2.337   62.41 1.46e-14 ***
Residuals    566 21.199   0.037
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Time series

Code :

```
time_series <- ts(breast_cancer_dataset$area_mean,start=1,end=12,frequency = 4)
time_series
```

Output :

```
     Qtr1   Qtr2   Qtr3   Qtr4
1  1001.0 1326.0 1203.0  386.1
2  1297.0  477.1 1040.0  577.9
3   519.8  475.9  797.8  781.0
4  1123.0  782.7  578.3  658.8
5   684.5  798.8 1260.0  566.3
6   520.0  273.9  704.4 1404.0
7   904.6  912.7  644.8 1094.0
8   732.4  955.1 1088.0  440.6
9   899.3 1162.0  807.2  869.5
10  633.0  523.8  698.8  559.2
11  563.0  371.1 1104.0  545.2
12  531.5
>
```

Correlation:

Code :

```
cor(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean,method='pearson')

cor(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean,method='spearman')

#measures the rank correlation between two variables
cor(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean,method='kendal')

#Correlation for all 4 variables using range
cor(breast_cancer_dataset[1:4],method='pearson')

c <- cor.test(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean)
c
```

Output :

```
> cor(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean,method='pearson')
[1] 0.9873572
> cor(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean,method='spearma
n')
[1] 0.999602
> #measures the rank correlation bt two variables
> cor(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean,method='kendal')
[1] 0.9855649
> #Correlation for all 4 variables using range
> cor(breast_cancer_dataset[1:4],method='pearson')
               radius_mean texture_mean perimeter_mean area_mean
radius_mean      1.0000000    0.3237819      0.9978553 0.9873572
texture_mean     0.3237819    1.0000000      0.3295331 0.3210857
perimeter_mean   0.9978553    0.3295331      1.0000000 0.9865068
area_mean        0.9873572    0.3210857      0.9865068 1.0000000
> c <- cor.test(breast_cancer_dataset$area_mean,breast_cancer_dataset$radius_mean)
> c

        Pearson's product-moment correlation

data:  breast_cancer_dataset$area_mean and breast_cancer_dataset$radius_mean
t = 148.32, df = 567, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.9851095 0.9892674
sample estimates:
      cor
0.9873572
```

Splitting train and test

Code :

```
Split_data <- sample.split(radius_mean,SplitRatio = 0.8)
train_data <- subset(breast_cancer_dataset , Split_data == TRUE)
test_data <- subset(breast_cancer_dataset ,Split_data == FALSE)

dim(train_data)
dim(test_data)
```

Output :

```
> dim(train_data)
[1] 455  31
> dim(test_data)
[1] 114  31
```

LINEAR REGRESSION

Code :

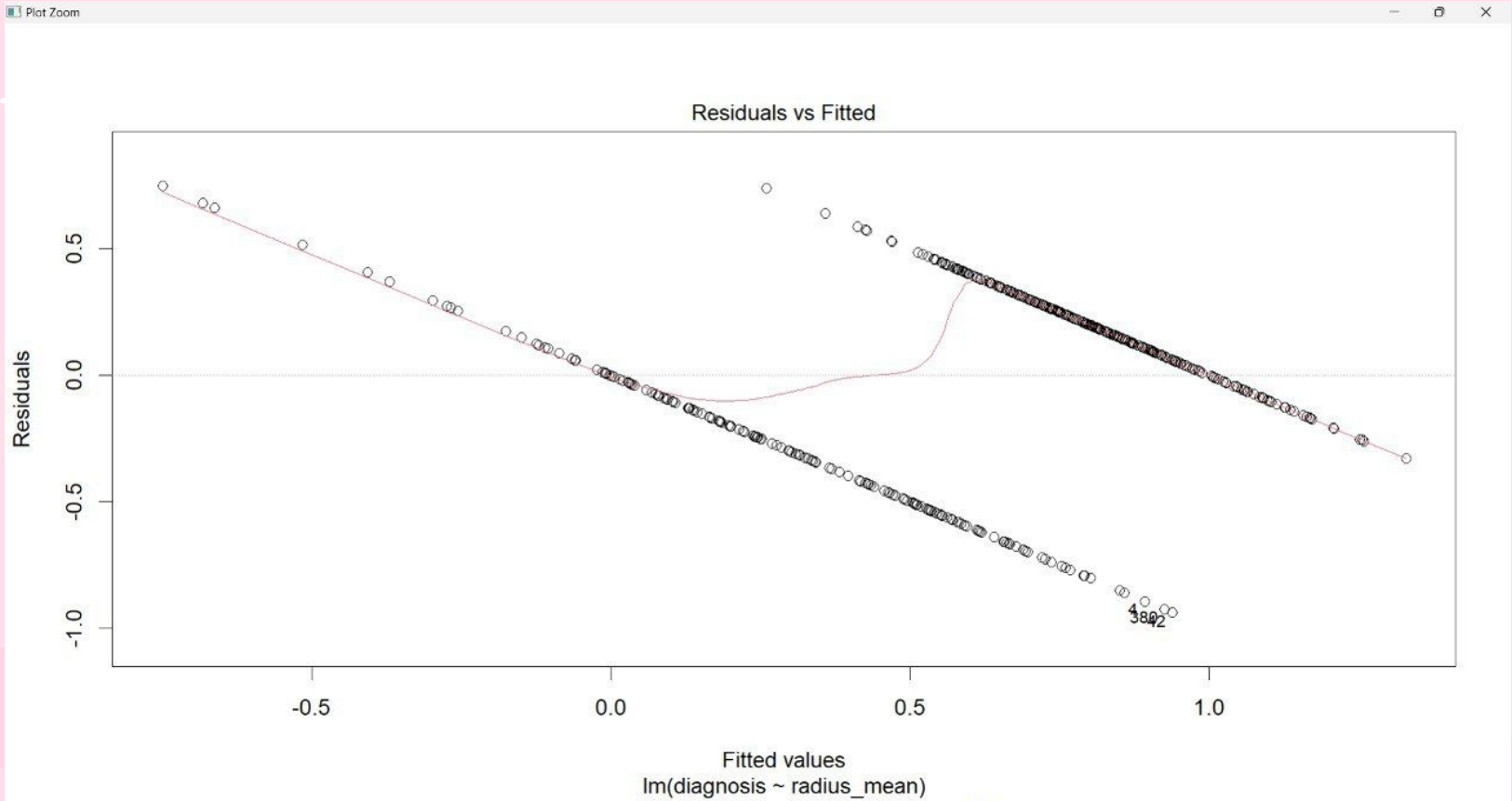
```
lm_mod1 <- lm(radius_mean~diagnosis,train_data)
lm_mod1
summary(lm_mod1)
plot(lm_mod1)
```

```
Call:
lm(formula = diagnosis ~ radius_mean, data = train_data)

Coefficients:
(Intercept)  radius_mean
     2.0129      -0.0984
```

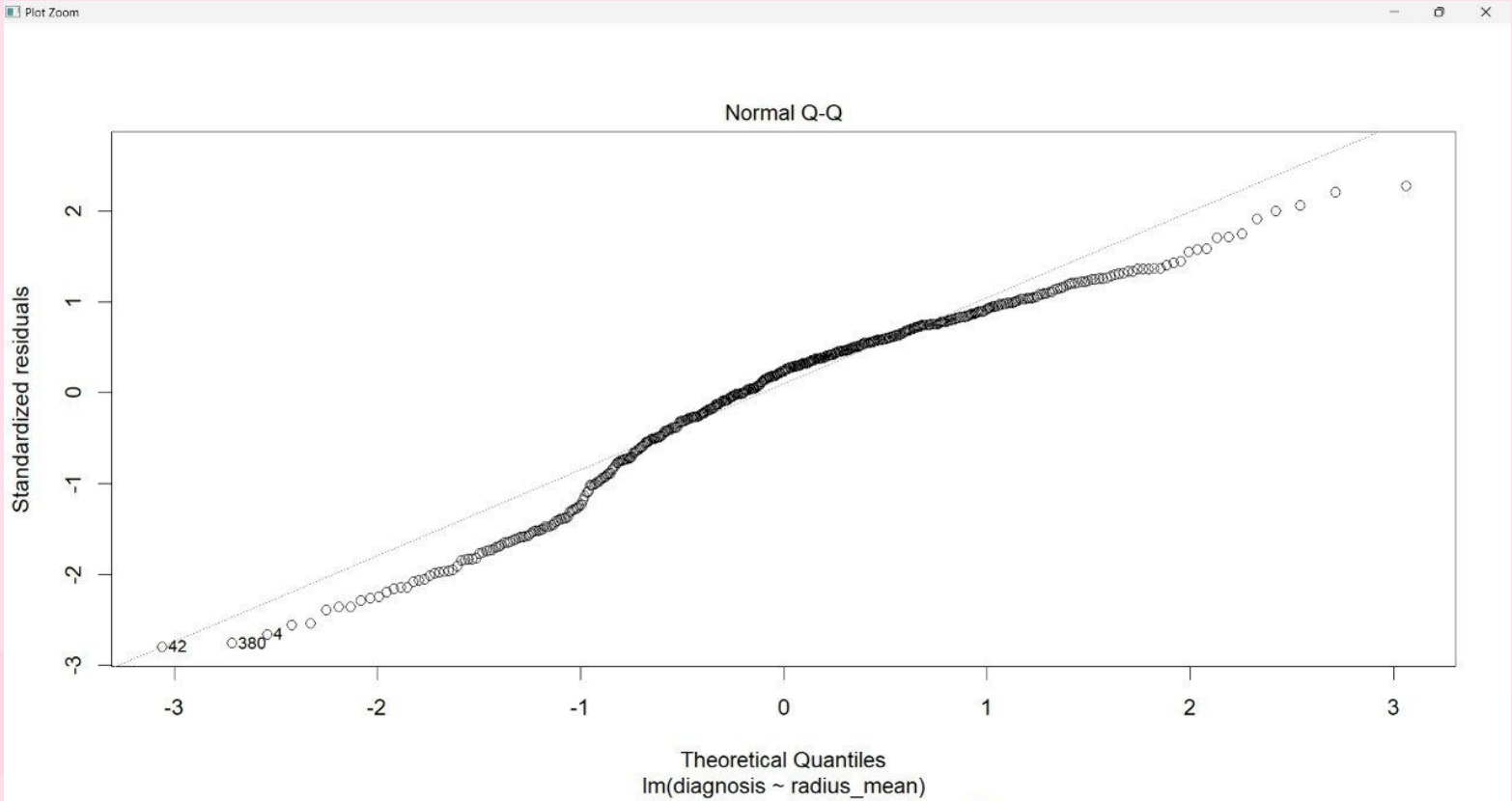
Output :

```
> summary(lm_mod1)

Call:
lm(formula = diagnosis ~ radius_mean, data = train_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.93541 -0.16639  0.07541  0.23876  0.75310

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.012870   0.062658   32.12   <2e-16 ***
radius_mean -0.098398   0.004316  -22.80   <2e-16 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3304 on 453 degrees of freedom
Multiple R-squared:  0.5344,    Adjusted R-squared:  0.5333
F-statistic: 519.9 on 1 and 453 DF,  p-value: < 2.2e-16
```

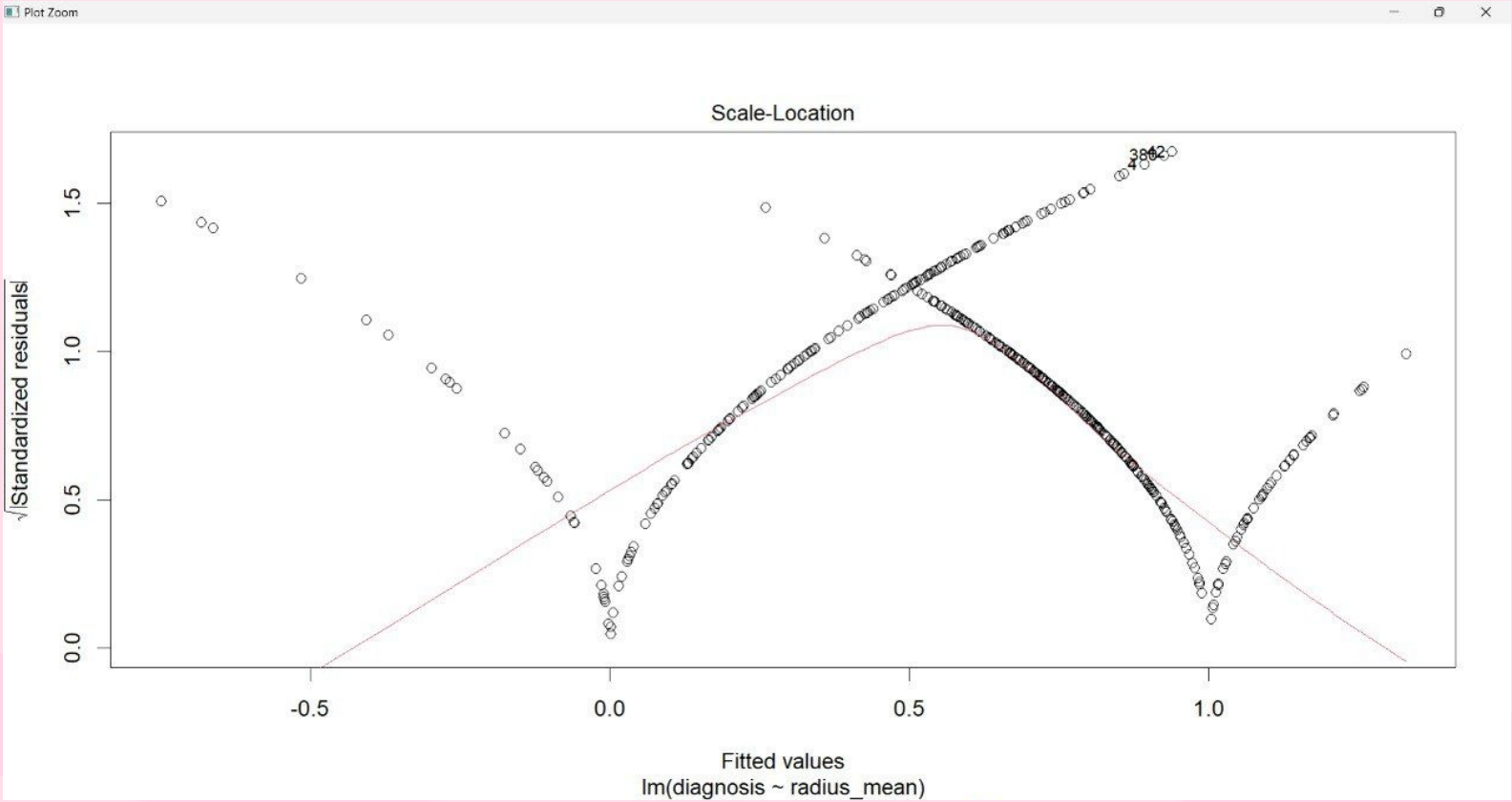
Output :


Plot Zoom
Residuals vs Fitted
Residuals
0.5
0.0
-0.5
-1.0
-0.5
0.0
0.5
1.0
Fitted values
lm(diagnosis ~ radius_mean)

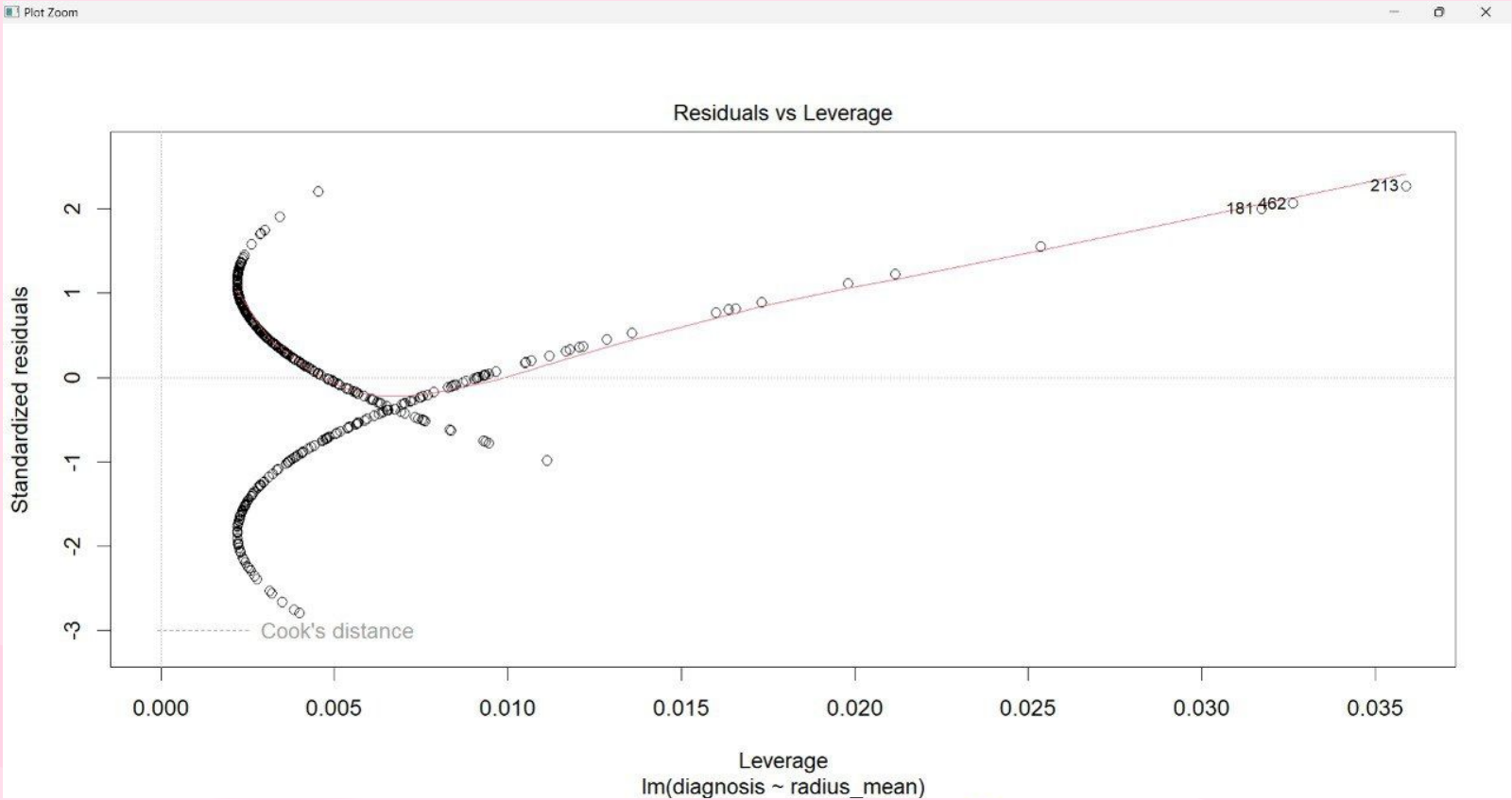
Output :


Plot Zoom
Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(diagnosis ~ radius_mean)
42
380
4
-3
-2
-1
0
1
2
3

Output :


Plot Zoom
Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
-0.5
0.0
0.5
1.0
Fitted values
lm(diagnosis ~ radius_mean)

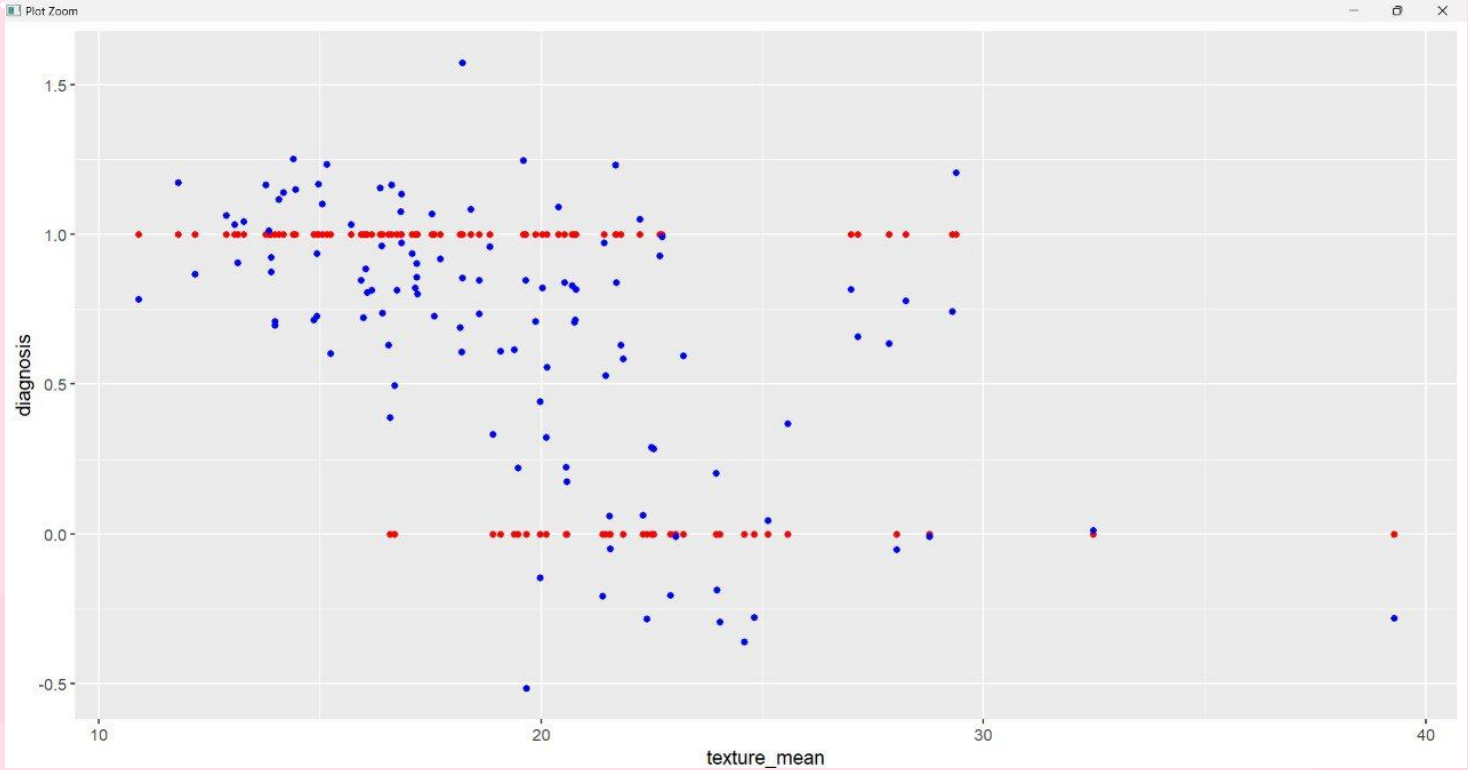
Output :


Plot Zoom
Residuals vs Leverage
Standardized residuals
2
1
0
-1
-2
-3
213
181
462
Cook's distance
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
Leverage
lm(diagnosis ~ radius_mean)

Code :

```
ggplot(test_data,aes(x = texture_mean)) +geom_point(aes(y = diagnosis),color = "red") +geom_point(aes(y = prediction),color = "blue")
```

Output :



MULTIPLE LINEAR REGRESSION

Code :

```
lm_mod <- lm(diagnosis~.,train_data)
lm_mod
summary(lm_mod)
plot(lm_mod)

prediction <- predict(lm_mod,test_data)
prediction

test_data$prediction = prediction
View(test_data)
```

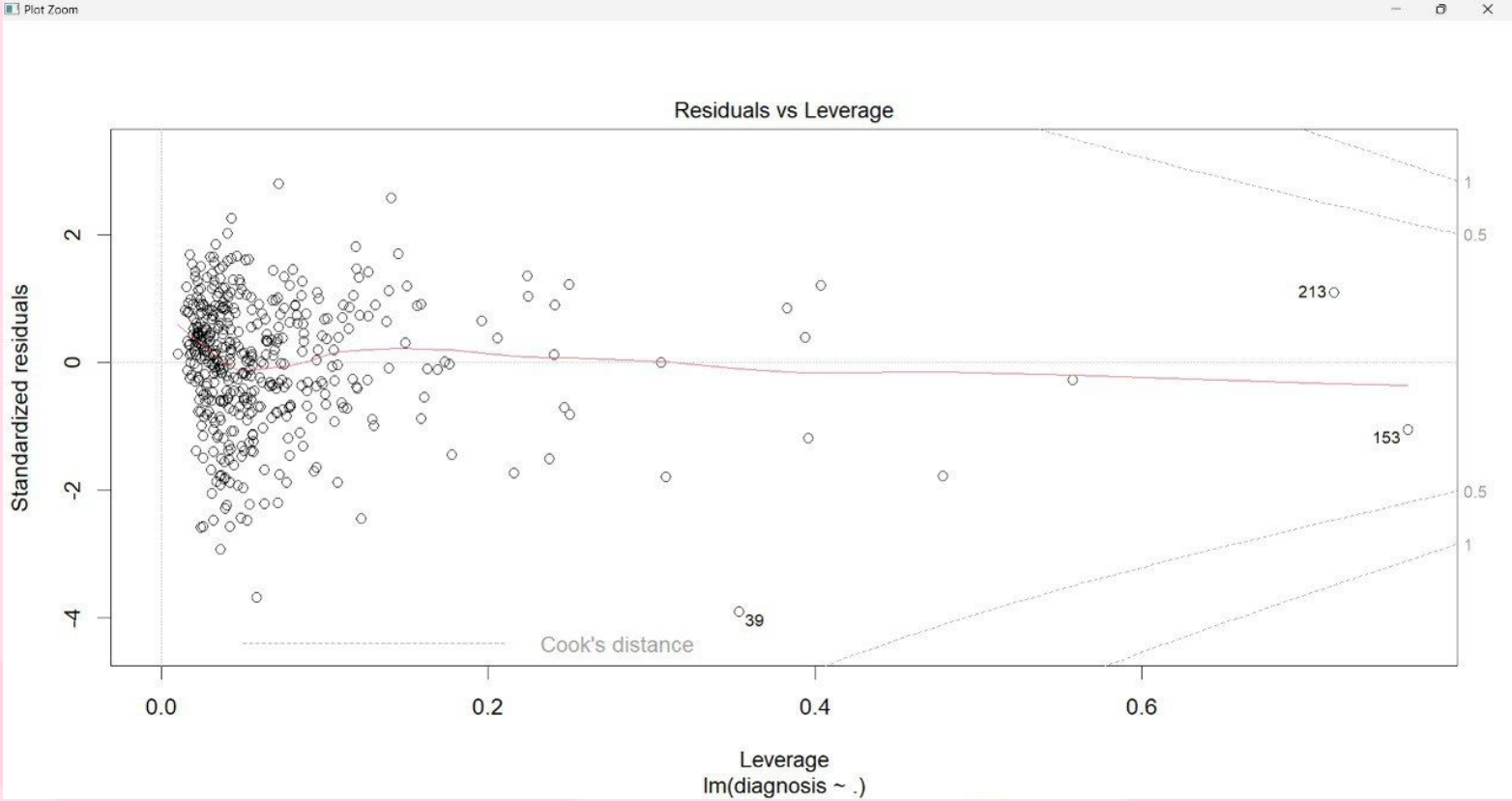
Output :

```
Coefficients:
            (Intercept)              radius_mean
              3.285e+00                5.675e-02
           texture_mean           perimeter_mean
             -7.893e-03               -9.429e-03
              area_mean          smoothness_mean
              1.555e-04               -2.511e-01
        compactness_mean           concavity_mean
              3.976e+00               -2.099e+00
     concave.points_mean             symmetry_mean
             -1.113e+00               -1.614e-01
 fractal_dimension_mean                radius_se
              2.853e-02               -3.616e-01
             texture_se             perimeter_se
              1.534e-02               -1.191e-02
                area_se            smoothness_se
              1.871e-03               -1.383e+01
          compactness_se             concavity_se
             -6.937e-01                3.484e+00
       concave.points_se              symmetry_se
             -8.426e+00               -2.168e+00
    fractal_dimension_se             radius_worst
              1.073e+01               -1.376e-01
           texture_worst          perimeter_worst
             -7.393e-03                2.160e-03
             area_worst         smoothness_worst
              6.558e-04               -1.282e+00
```

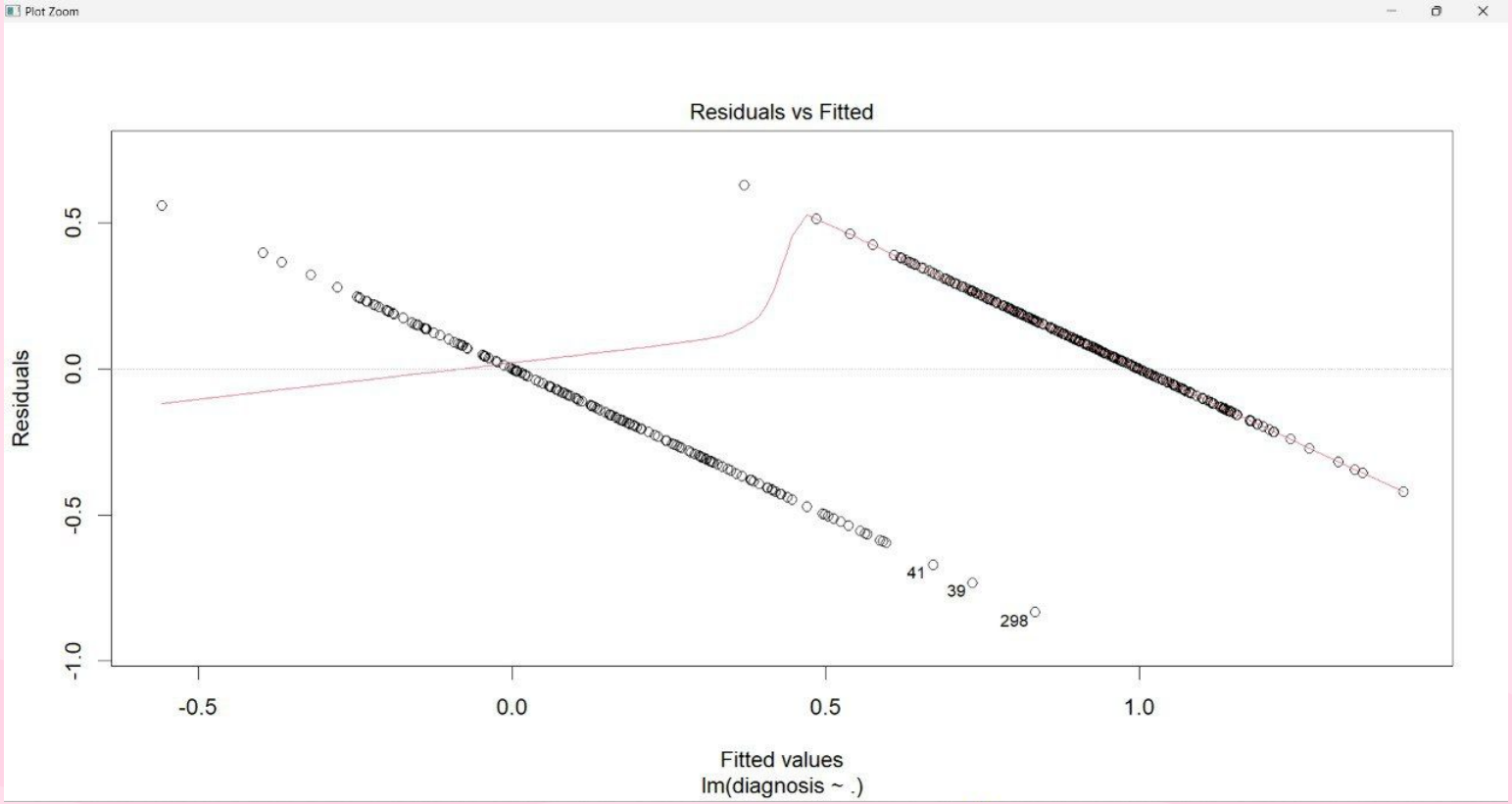
Output :

```
> summary(lm_mod)

Call:
lm(formula = diagnosis ~ ., data = train_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.82724 -0.13827  0.02421  0.15800  0.54413 

Coefficients:
                        Estimate Std. Error t value
(Intercept)             3.285e+00  4.607e-01   7.131
radius_mean             5.675e-02  1.991e-01   0.285
texture_mean           -7.893e-03  8.682e-03  -0.909
perimeter_mean         -9.428e-03  2.891e-02  -0.326
area_mean               1.555e-04  5.650e-04   0.275
smoothness_mean        -2.511e-01  2.165e+00  -0.116
compactness_mean        3.976e+00  1.422e+00   2.797
concavity_mean         -2.099e+00  1.142e+00  -1.838
concave.points_mean    -1.113e+00  2.125e+00  -0.524
symmetry_mean          -1.614e-01  8.297e-01  -0.195
fractal_dimension_mean  2.853e-02  6.082e+00   0.005
radius_se              -3.616e-01  3.331e-01  -1.086
texture_se              1.534e-02  4.143e-02   0.370
perimeter_se           -1.191e-02  4.473e-02  -0.266
area_se                 1.871e-03  1.506e-03   1.243
smoothness_se          -1.383e+01  6.945e+00  -1.992
compactness_se         -6.937e-01  2.303e+00  -0.301
```

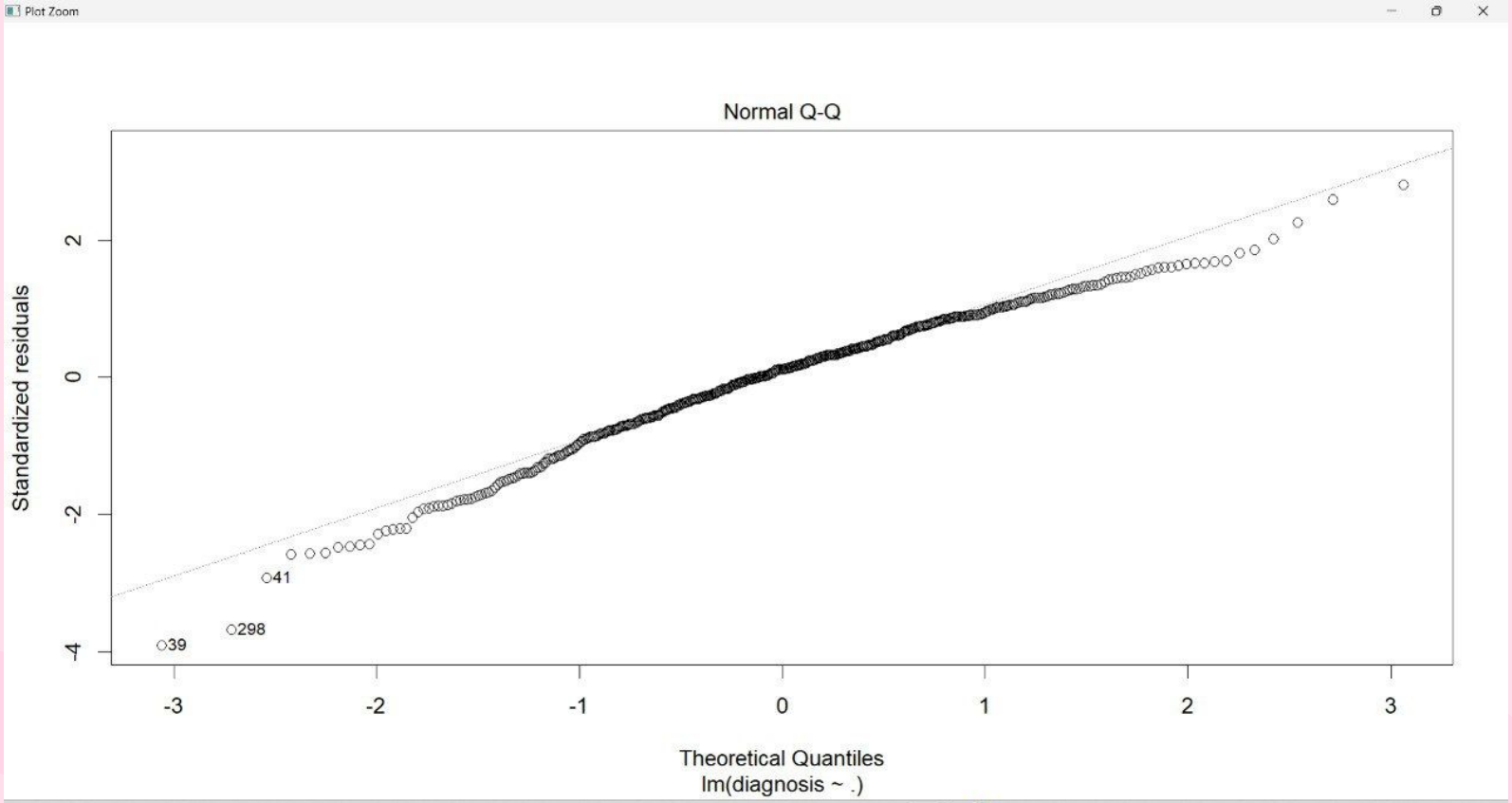
Output :


Plot Zoom
Residuals vs Leverage
Standardized residuals
2
0
-2
-4
0.0
0.2
0.4
0.6
Leverage
lm(diagnosis ~ .)
213
153
39
Cook's distance
1
0.5
0.5
1

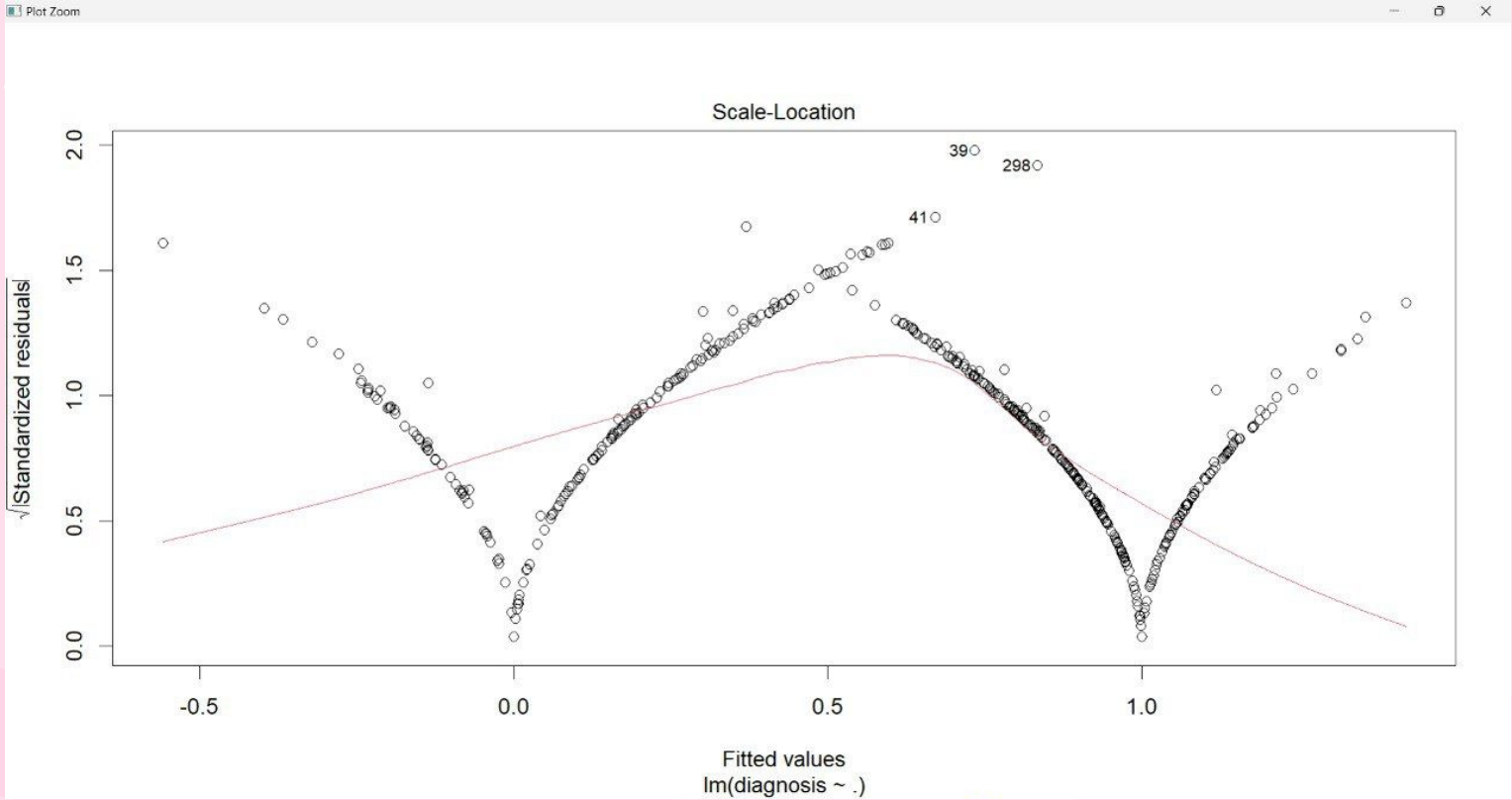
Output :


Plot Zoom
Residuals vs Fitted
Residuals
0.5
0.0
-0.5
-1.0
41
39
298
-0.5
0.0
0.5
1.0
Fitted values
lm(diagnosis ~ .)

Output :


Plot Zoom
Normal Q-Q
Standardized residuals
2
0
-2
-4
41
298
39
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(diagnosis ~ .)

Output :


Plot Zoom
Scale-Location
√|Standardized residuals|
2.0
1.5
1.0
0.5
0.0
39
298
41
-0.5
0.0
0.5
1.0
Fitted values
lm(diagnosis ~ .)

Thank You