Simple Linear Regression:Diagnostics and Assumptions Test

EXST 7014 - Lab 2

January 22, 2020

Table of Contents

Objectives

Simple Linear Regression (SLR) is a common analysis procedure, used to describe the significant relationship between two variables: the dependent (or response) variable, and the independent (or explanatory) variable. In lab 1, SLR was performed to fit a straight line model relating two variables. We learned how to interpret parameter estimates and R^2, and understood the hypothesis test of SLR.

You might notice that a single observation that is substantially different from all other observations can make a large difference in the results of your regression analysis. If a single observation (or small group of observations) substantially changes your results, you would want to know about this and investigate further. In this lab exercise, we will conduct appropriate regression diagnostics to detect outliers (or unusual observations) as well as evaluate some model assumptions.

In this lab exercise, you will get familiar with and understand as listed:

1. Conduct appropriate regression diagnostics to detect outliers (or unusual observations)

2. Evaluate the assumptions of SLR using Residual Plots and the Normality Test.

Part I

Lab Setup

Run the following code to both install and load the required packages.

```
install.packages('olsrr')          # install the package that runs residual
plots and check assumptions
library(olsrr)                     # Load the package
```

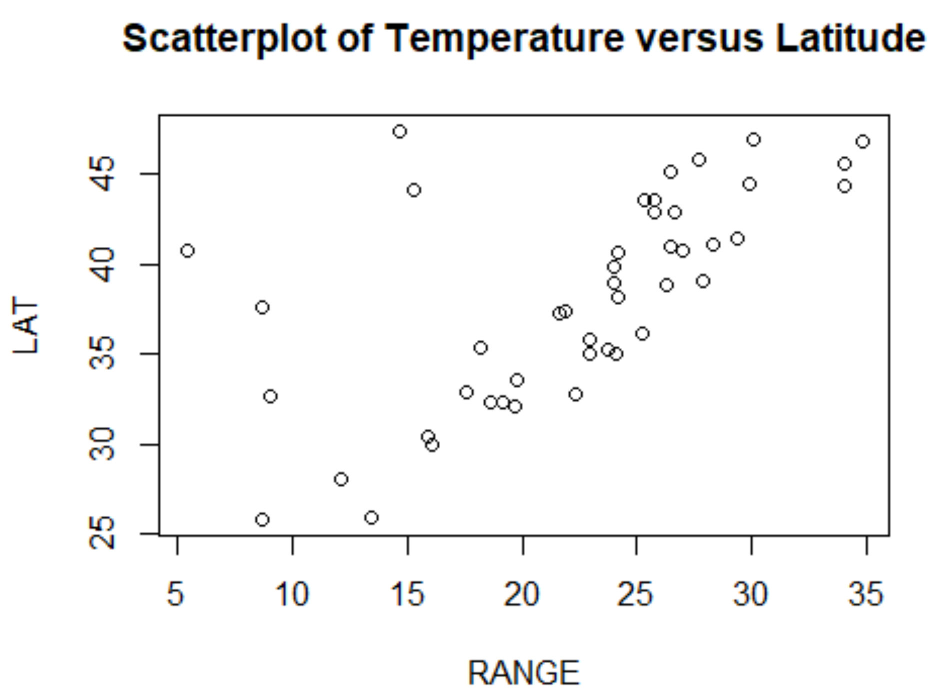
The Dataset

The data is from the textbook, Chapter 7, problem 6 and can be obtained from the url: http://stat.lsu.edu/exstweb/statlab/datasets/fwdata97/FW07P06.txt

The latitude (LAT) and the mean monthly range (RANGE), which is the difference between mean monthly maximum and minimum temperatures, are given for a selected set of US cities. The following program performs a SLR using RANGE as the dependent variable and LAT as the independent variable.

```
# Create an object called 'theData' to store the data

theData <- read.table(header=T, stringsAsFactors = TRUE,text='
       CITY STATE LAT RANGE
       Montgome     AL      32.3     18.6
       Tuscon       AZ      32.1     19.7
       Bishop       CA      37.4     21.9
       Eureka       CA      40.8      5.4
       San_Dieg     CA      32.7      9.0
       San_Fran     CA      37.6      8.7
       Denver       CO      39.8     24.0
       Washingt     DC      39.0     24.0
       Miami        FL      25.8      8.7
       Talahass     FL      30.4     15.9
       Tampa        FL      28.0     12.1
       Atlanta      GA      33.6     19.8
       Boise        ID      43.6     25.3
       Moline       IL      41.4     29.4
       Ft_wayne     IN      41.0     26.5
       Topeka       KS      39.1     27.9
       Louisv       KY      38.2     24.2
       New_Orl      LA      30.0     16.1
       Caribou      ME      46.9     30.1
       Portland     ME      43.6     25.8
       Alpena       MI      45.1     26.5
       St_cloud     MN      45.6     34.0
       Jackson      MS      32.3     19.2
       St_Louis     MO      38.8     26.3
       Billings     MT      45.8     27.7
       N_PLatte     NB      41.1     28.3
```

```
        L_Vegas       NV       36.1      25.2
        Albuquer      NM       35.0      24.1
        Buffalo       NY       42.9      25.8
        NYC           NY       40.6      24.2
        C_Hatter      NC       35.3      18.2
        Bismark       ND       46.8      34.8
        Eugene        OR       44.1      15.3
        Charestn      SC       32.9      17.6
        Huron         SD       44.4      34.0
        Knoxvlle      TN       35.8      22.9
        Memphis       TN       35.0      22.9
        Amarillo      TX       35.2      23.7
        Brownsvl      TX       25.9      13.4
        Dallas        TX       32.8      22.3
        SLCity        UT       40.8      27.0
        Roanoke       VA       37.3      21.6
        Seattle       WA       47.4      14.7
        Grn_bay       WI       44.5      29.9
        Casper        WY       42.9      26.6
        ')


# Scatterplot of Temperature versus Latitude
with(theData, plot(RANGE, LAT, main = 'Scatterplot of Temperature versus
Latitude'))
```



Part II

Fitting the SLR model

Based on the scatterplot produced above, we assume that an appriopriate regression model relating RANGE and LAT is the linear model given by

$$y = \beta_0 + \beta_1 X + \epsilon$$

where Y is the RANGE, X is the LAT, and ϵ is a random error term that is normally distributed with the mean 0 and the unknown variance σ^2.

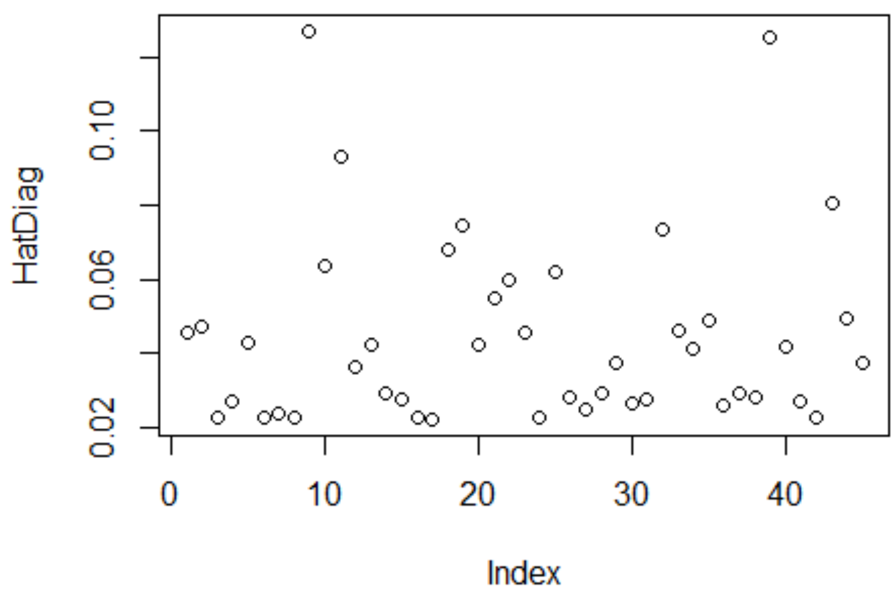
β_0 is the estimate of the Y-intercept. and β_1 is the estimate of the slope coefficient.

```r
# Fit the model

SLR_model <- lm(RANGE ~ LAT, data = theData)
summary(SLR_model)

##
## Call:
## lm(formula = RANGE ~ LAT, data = theData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.7823  -0.4865   0.8395   3.0765   7.1123
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -6.4793     5.5481  -1.168    0.249
## LAT           0.7515     0.1438   5.228 4.79e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.498 on 43 degrees of freedom
## Multiple R-squared:  0.3886, Adjusted R-squared:  0.3744
## F-statistic: 27.33 on 1 and 43 DF,  p-value: 4.786e-06

## R Student
rStudent <- rstudent(SLR_model)    # get r-student values  N/B call rStudent
to print the Rstudent scores

# Install.packages('car')
library(car)
outlierTest(SLR_model)  # run test to get possible outliers

##    rstudent unadjusted p-value Bonferonni p
## 4 -4.031139          0.00022872     0.010292

## Hat diagonal values
HatDiag <- lm.influence(SLR_model)$hat        # get Hat Diag values
```

```r
cutoff <- 2*(length(coef(SLR_model))/length(HatDiag))     # cu-off at 2*p/n
a <- theData[which(lm.influence(SLR_model)$hat > cutoff),]   # Print obs with
HAT > 2*p/n
cbind(a, HatDiag[HatDiag > cutoff])
```

```
##         CITY STATE  LAT RANGE HatDiag[HatDiag > cutoff]
## 9      Miami    FL 25.8   8.7                0.12686850
## 11     Tampa    FL 28.0  12.1                0.09295866
## 39 Brownsvl     TX 25.9  13.4                0.12518355
```

```r
plot(HatDiag, ylab="HatDiag")
```

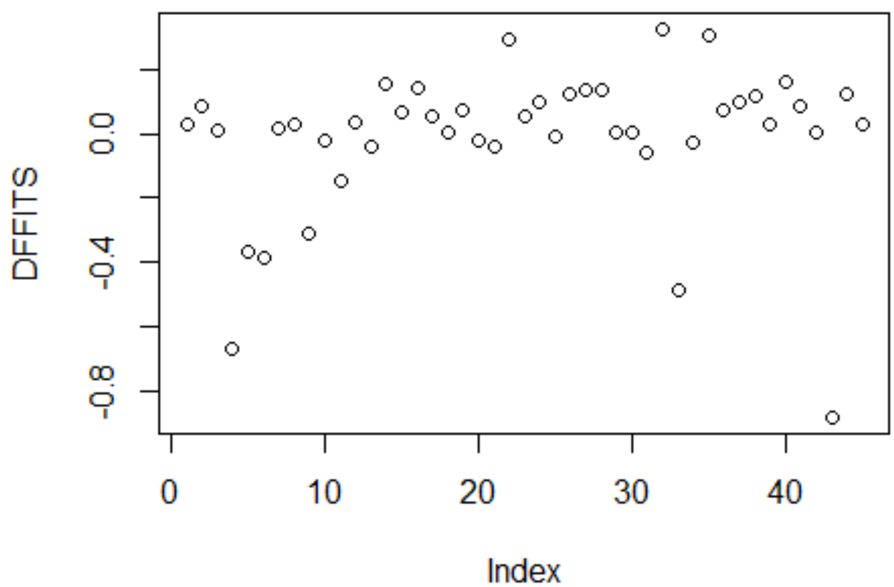



```r
# Get ALL INFLUENCE MEASURES DISCUSSED
influence.measures(SLR_model)
```

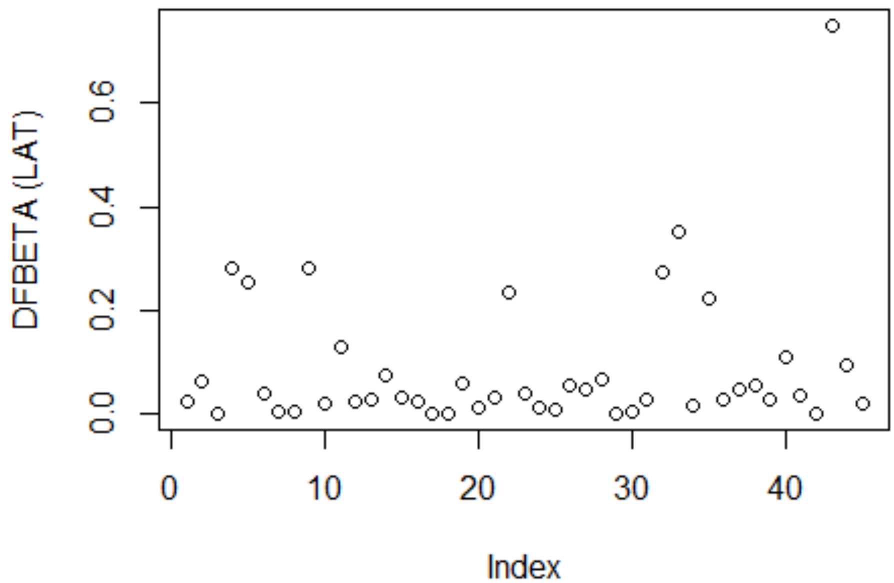
```
## Influence measures of
##   lm(formula = RANGE ~ LAT, data = theData) :
##
##        dfb.1_   dfb.LAT    dffit cov.r   cook.d    hat inf
## 1   0.026392 -0.023305  0.03248 1.097 5.40e-04 0.0458
## 2   0.069580 -0.061699  0.08464 1.093 3.65e-03 0.0474
## 3   0.002106 -0.001012  0.00755 1.072 2.92e-05 0.0226
## 4   0.187855 -0.280922 -0.67084 0.560 1.66e-01 0.0269   *
## 5  -0.289049  0.252898 -0.36524 0.954 6.37e-02 0.0427
## 6  -0.095110  0.038608 -0.38731 0.803 6.65e-02 0.0224   *
## 7  -0.002108  0.004466  0.01626 1.073 1.35e-04 0.0240
## 8   0.000126  0.004669  0.03245 1.070 5.38e-04 0.0227
```

```
## 9  -0.298734  0.282598 -0.31116 1.163 4.88e-02 0.1269   *
## 10 -0.019976  0.018203 -0.02258 1.119 2.61e-04 0.0635
## 11 -0.139515  0.130166 -0.14922 1.144 1.13e-02 0.0930   *
## 12  0.026925 -0.022949  0.03669 1.086 6.88e-04 0.0365
## 13  0.021933 -0.026301 -0.03814 1.093 7.44e-04 0.0424
## 14 -0.054779  0.075221  0.15263 1.041 1.17e-02 0.0294
## 15 -0.020532  0.029699  0.06681 1.070 2.28e-03 0.0277
## 16 -0.001859  0.022544  0.14018 1.031 9.86e-03 0.0228
## 17  0.007723  0.000274  0.05412 1.065 1.49e-03 0.0222
## 18  0.001531 -0.001402  0.00171 1.125 1.49e-06 0.0679
## 19 -0.052798  0.059154  0.07065 1.129 2.55e-03 0.0743
## 20  0.010814 -0.012967 -0.01880 1.094 1.81e-04 0.0424
## 21  0.027334 -0.031509 -0.04080 1.108 8.52e-04 0.0550
## 22 -0.205299  0.234446  0.29552 1.046 4.33e-02 0.0600
## 23  0.046075 -0.040686  0.05671 1.095 1.64e-03 0.0458
## 24  0.003852  0.011007  0.10039 1.050 5.11e-03 0.0225
## 25  0.008058 -0.009171 -0.01145 1.117 6.71e-05 0.0620
## 26 -0.038915  0.055472  0.12139 1.053 7.45e-03 0.0281
## 27  0.063941 -0.045816  0.13417 1.040 9.07e-03 0.0252
## 28  0.082978 -0.066140  0.13606 1.049 9.34e-03 0.0291
## 29 -0.000740  0.000913  0.00143 1.089 1.05e-06 0.0375
## 30 -0.001265  0.001970  0.00503 1.076 1.29e-05 0.0263
## 31 -0.032962  0.025705 -0.05715 1.072 1.67e-03 0.0279
## 32 -0.242094  0.271576  0.32549 1.062 5.26e-02 0.0731
## 33  0.297134 -0.350808 -0.48667 0.882 1.09e-01 0.0463
## 34 -0.019164  0.016682 -0.02457 1.093 3.09e-04 0.0412
## 35 -0.190875  0.223549  0.30304 1.014 4.51e-02 0.0488
## 36  0.038170 -0.028397  0.07394 1.066 2.79e-03 0.0261
## 37  0.059484 -0.047413  0.09754 1.063 4.83e-03 0.0291
## 38  0.068525 -0.053853  0.11652 1.055 6.87e-03 0.0283
## 39  0.028959 -0.027380  0.03019 1.198 4.67e-04 0.1252   *
## 40  0.125570 -0.109591  0.15981 1.064 1.29e-02 0.0419
## 41 -0.024006  0.035898  0.08573 1.064 3.74e-03 0.0269
## 42  0.000393 -0.000201  0.00133 1.073 9.07e-07 0.0227
## 43  0.673312 -0.749969 -0.88152 0.777 3.28e-01 0.0805   *
## 44 -0.078978  0.092266  0.12418 1.088 7.84e-03 0.0496
## 45 -0.015729  0.019395  0.03038 1.088 4.72e-04 0.0375

## To get individual outliers run the codes below.

## DFFITS
DFFITS_model <- abs(dffits(SLR_model))          # get absolute values of DFFITS
b <- theData[which(DFFITS_model > 1),]     # get DFFITS > 1
cbind(b, DFFITS = DFFITS_model[DFFITS_model > 1])   # Print obs with DFFITS > 1

## [1] CITY   STATE  LAT    RANGE  DFFITS
## <0 rows> (or 0-length row.names)
```

```r
plot(dffits(SLR_model), ylab="DFFITS")      # Plot all DFFITS
```




```r
## DFBETAS
DFBETAS_model <- abs(dfbetas(SLR_model)[, 'LAT'])    # get absolute values of DFBETAS for LAT
c <- theData[which(DFBETAS_model > 1),]           # get LAT DFBETAS > 1
cbind(c, DFBETA_LAT = DFBETAS_model[DFBETAS_model > 1])     # Print obs with LAT DFBETAS > 1

## [1] CITY       STATE      LAT        RANGE      DFBETA_LAT
## <0 rows> (or 0-length row.names)

plot(DFBETAS_model, ylab="DFBETA (LAT)")    # Plot LAT DFBETAS
```




```
## COOK's Distance
COOKS_mod <- cooks.distance(SLR_model)  # get cook's D for all observations
cutoff <- 4/length(COOKS_mod)    # cut off at 4/n
d <- theData[which(COOKS_mod > cutoff),]
cbind(d, COOKsD = COOKS_mod[COOKS_mod > cutoff])

##        CITY STATE  LAT RANGE    COOKsD
## 4   Eureka     CA 40.8   5.4 0.1661055
## 33  Eugene     OR 44.1  15.3 0.1086150
## 43 Seattle     WA 47.4  14.7 0.3283570
```

Part III

Evaluate Assumptions - Residual Analysis

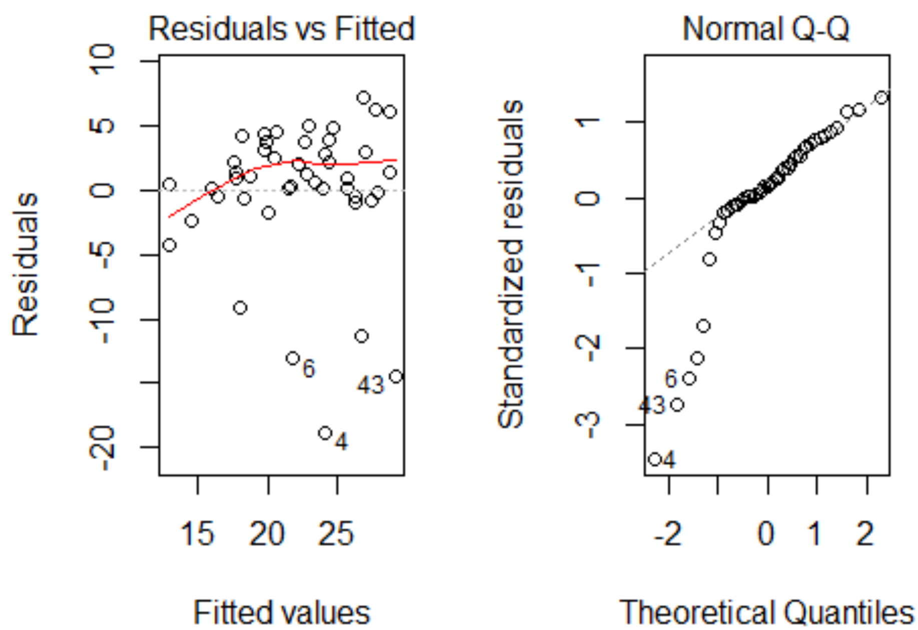
Residual Plot can be used to detect various problems such as non-linear pattern, non-homogeneous variances and outliers.

- If the data is of homogeneity, most of residual points of data scatter around zero.

- If problesm such as courvature or non-homogenous variance are detected in residual plot, we may need to consider fitting a more complicated model.

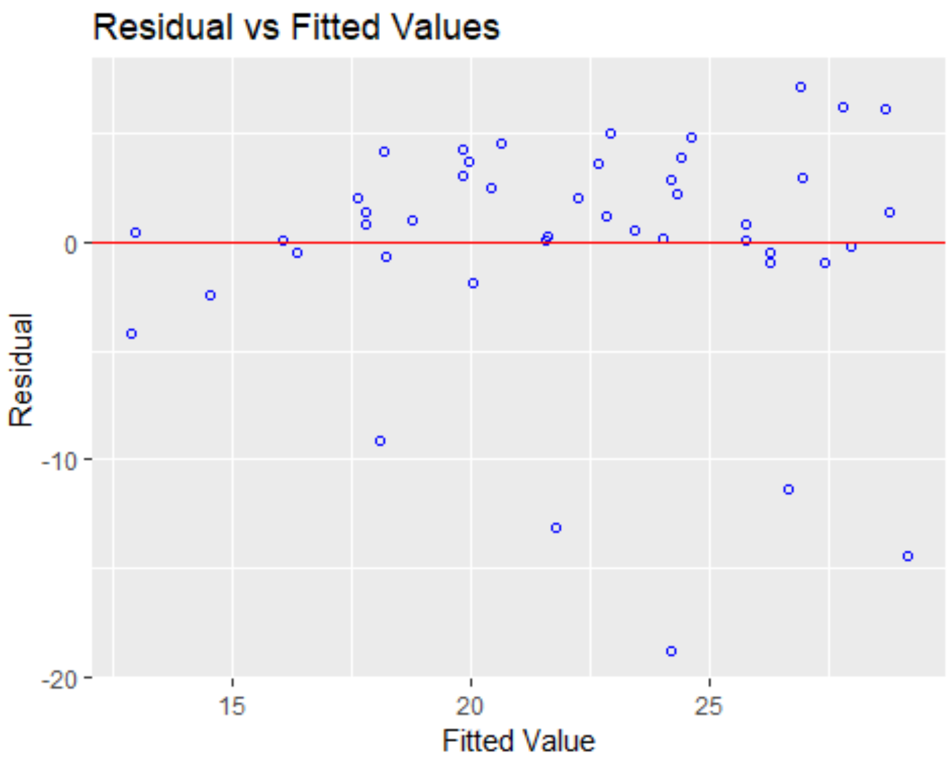
Shapiro-Wilk Test is conducted on the **RESIDUALS of the fitted model** to check for normiality. If the p-value of this test is less than the signifcant level of 0.05, the null hypothesis is rejected and we conclude that the data was not sampled from a normally distributed population. Otherwise, we fail to reject to rejct the null and conclude that the data is normally distributed.

```
par(mfrow = c(1,2))
plot(SLR_model, which = 1:2)
```

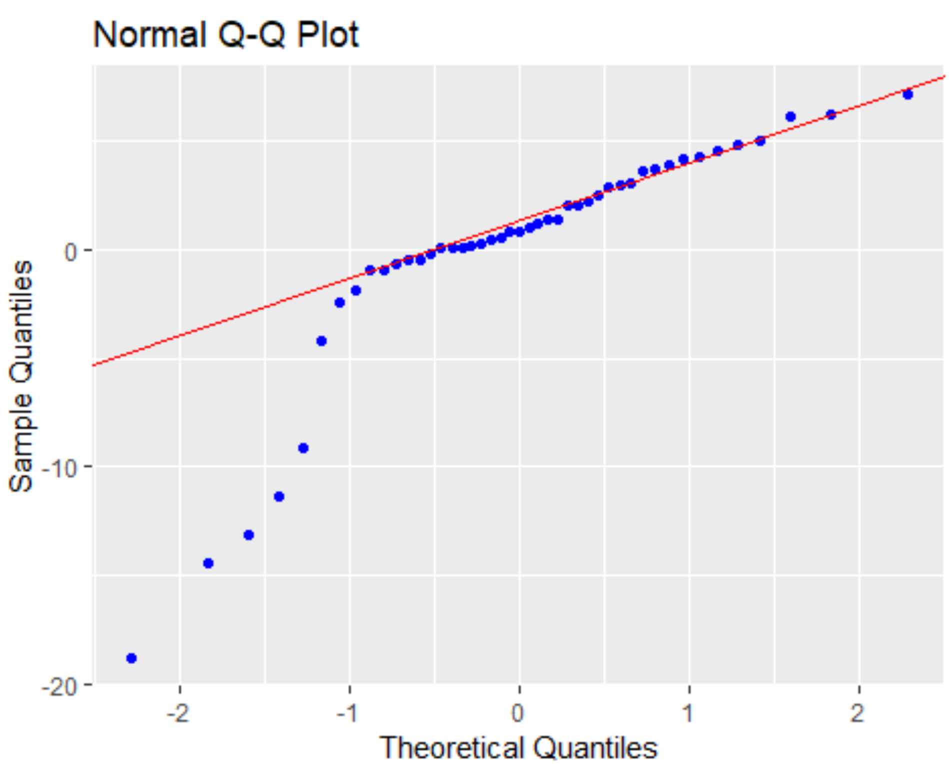


Alternative plotting using the olsrr package

```
ols_plot_resid_fit(SLR_model)
```




```r
ols_plot_resid_qq(SLR_model)
```




```r
ols_test_normality(SLR_model)  ## Test normality
```

```
## -----------------------------------------------
##        Test             Statistic       pvalue
## -----------------------------------------------
## Shapiro-Wilk              0.7976         0.0000
## Kolmogorov-Smirnov        0.2502         0.0057
## Cramer-von Mises          2.7863         0.0000
## Anderson-Darling          3.1553         0.0000
## -----------------------------------------------
```

Lab Assignment

Your assignment is to perform necessary analysis using R to answer the following questions.

1. Use **Residual Plot** to check the assumption of homogeneity of variance. Does the data set appear to be homogenous?

2. Use the **olsrr package** or any function you deem appropriate. Does RANGE appear to be normally distributed? Why? Is this relevant to the normality assumption? Why?

3. Using the **lm** function fit the regression model. Write down the regression equation and answer: Does the model fit the data well? Why? Is this relevant to the normality assumption? Why?

4. What is the predicted value of RANGE at LAT=42 ? (Hint use the **predict** function. See example below)

```
predict(SLR_model, newdata=data.frame(LAT=29))  # remember to change to the required valude of LAT for this question
```

5. Does there appear to be any possible outlier(s)? State the name and value of the statistics that you use to reach your conclusion.