Predictive Modeling I

Predictive Modeling:

Predictive modeling is the process of using known information to create, process, and validate a model that can be used to define the relationship between the various attributes and forecast future outcomes.

- Predictive modeling is one of the advanced techniques for handling missing data (lecture 2).
- The most widely used predictive modeling techniques are regression and neural networks.

The Process of Creating a Predictive Model:

1. Clean the data by removing/restimate outliers and treating missing data.
2. Identify the predictive modeling approach to use.
3. Specify a subset of the data to be used for training the model.
4. Train, or estimate, model parameters from the training data set.
5. Model Selection and develop Models.
6. Validate predictive modeling accuracy on data not used for calibrating the model (Testing).

1. Regression Model:

Regression is a data mining approach that predicts a value (number). Profit, sales, mortgage rates, house values, square footage, temperature, distance, etc.

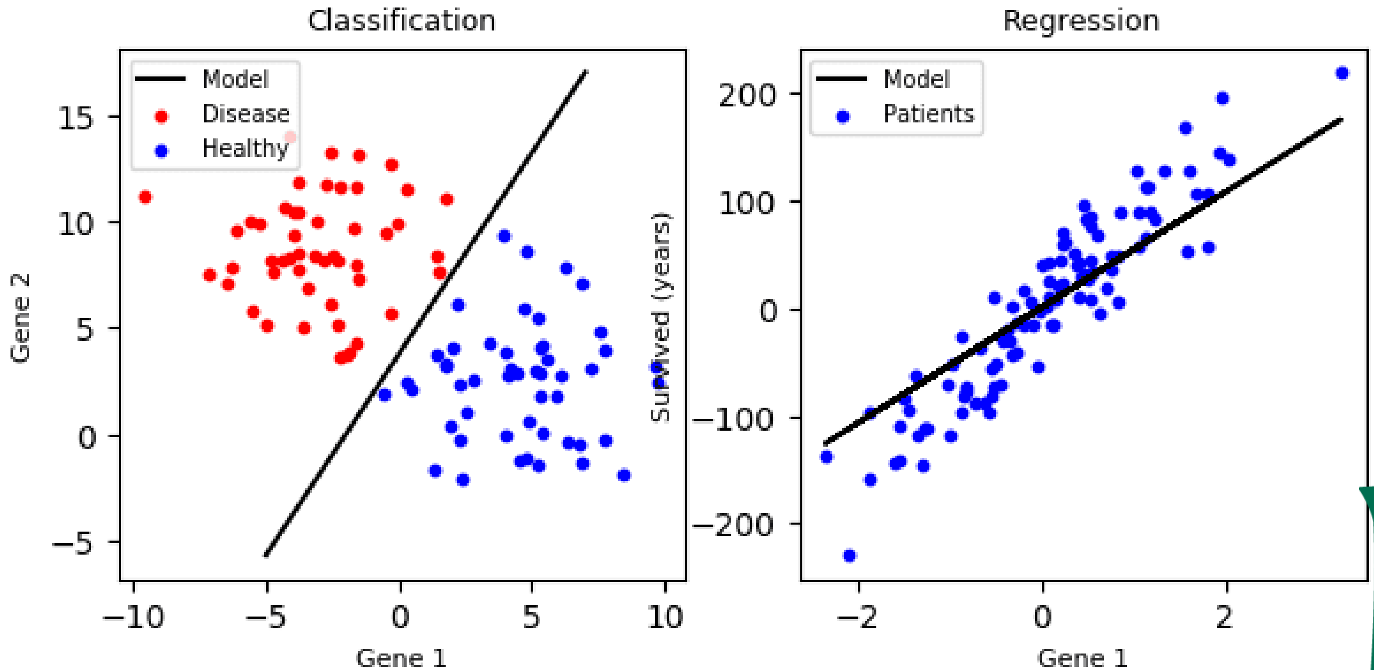
It is the process of estimating the values of a response (target) (y) as a function (F) of one or more predictors (x_1, x_2, ..., x_k), a set of parameters (β_1, β_2, ..., β_k), and a measure of error (e).

$$y = F(x, \beta) + e$$

The error, also called the residual, is the difference between the actual and predicted value of the dependent attribute. The regression parameters are also known as regression coefficients.

Regression vs. Classification:

Regression and classification are data mining techniques used to solve similar problems, but they are frequently confused. Both are used in prediction analysis, but regression is used to predict a numeric value while classification assigns data into discrete categories.



1.1 Linear Regression Model:

A linear regression technique can be used if the relationship between the predictors and the response can be approximated with a straight line.

- The response attribute is numerical (quantitative).
- The case of one predictor is called **simple linear regression**.
- The case of more than one predictor is called **multiple linear regression**.
- The predictors are numerical or categorical (**binary** which is coded as (0, 1)).

Linear Regression Model:

Simple Linear Regression

$$y = b_0 + b_1 x_1$$

Multiple Linear Regression

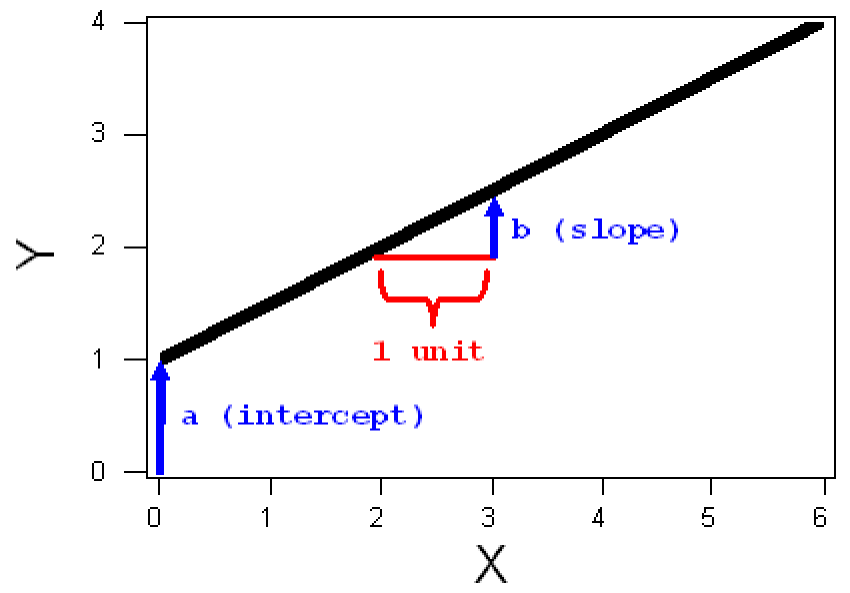
$$y = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_n x_n$$

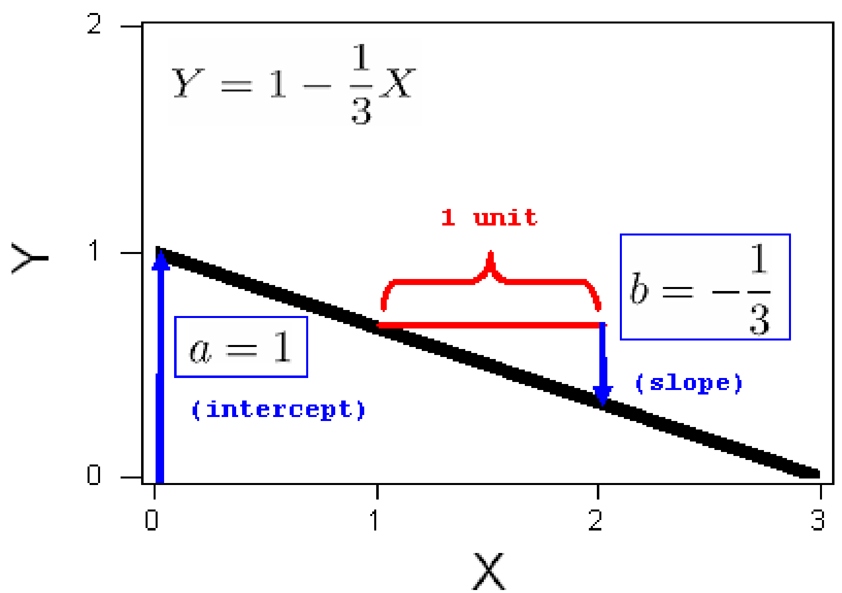
Polynomial Linear Regression

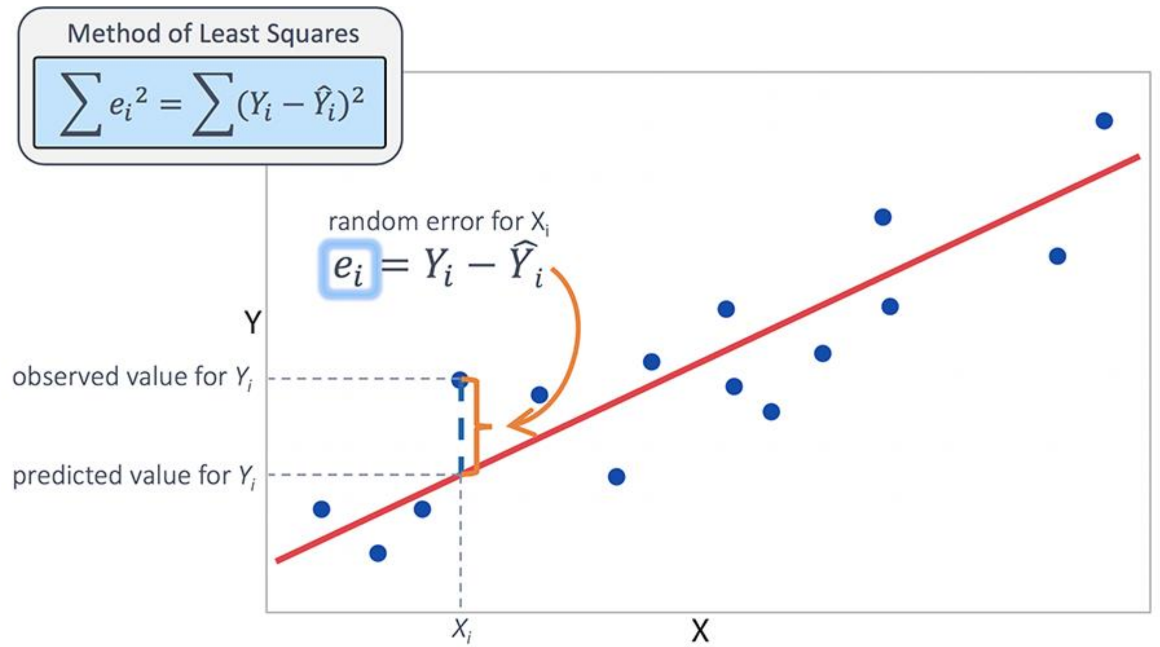
$$y = b_0 + b_1 x_1 + b_2 x_1^2 + \ldots + b_n x_1^n$$

- A linear regression means that a regression that linear in coefficients (β_1, β_2, ..., β_k).

Linear Regression Model:







Example 1: (Simple Linear Regression)

Consider the built-in data set *faithful*, which is a sample of the waiting time between eruptions (in mins) and the duration of the eruption (in mins) for the Old Faithful geyser in Yellowstone National Park, Wyoming.

1. Show the data structure.

```
> str(faithful)
'data.frame':   272 obs. of  2 variables:
 $ eruptions: num  3.6 1.8 3.33 2.28 4.53 ...
 $ waiting  : num  79 54 74 62 85 55 88 85 51 85 ...
```

Example 2: (Simple Linear Regression)

2. Identify the response and predictor.

Note: the predictor most be continuous numeric.

3. Present the summary statistics.

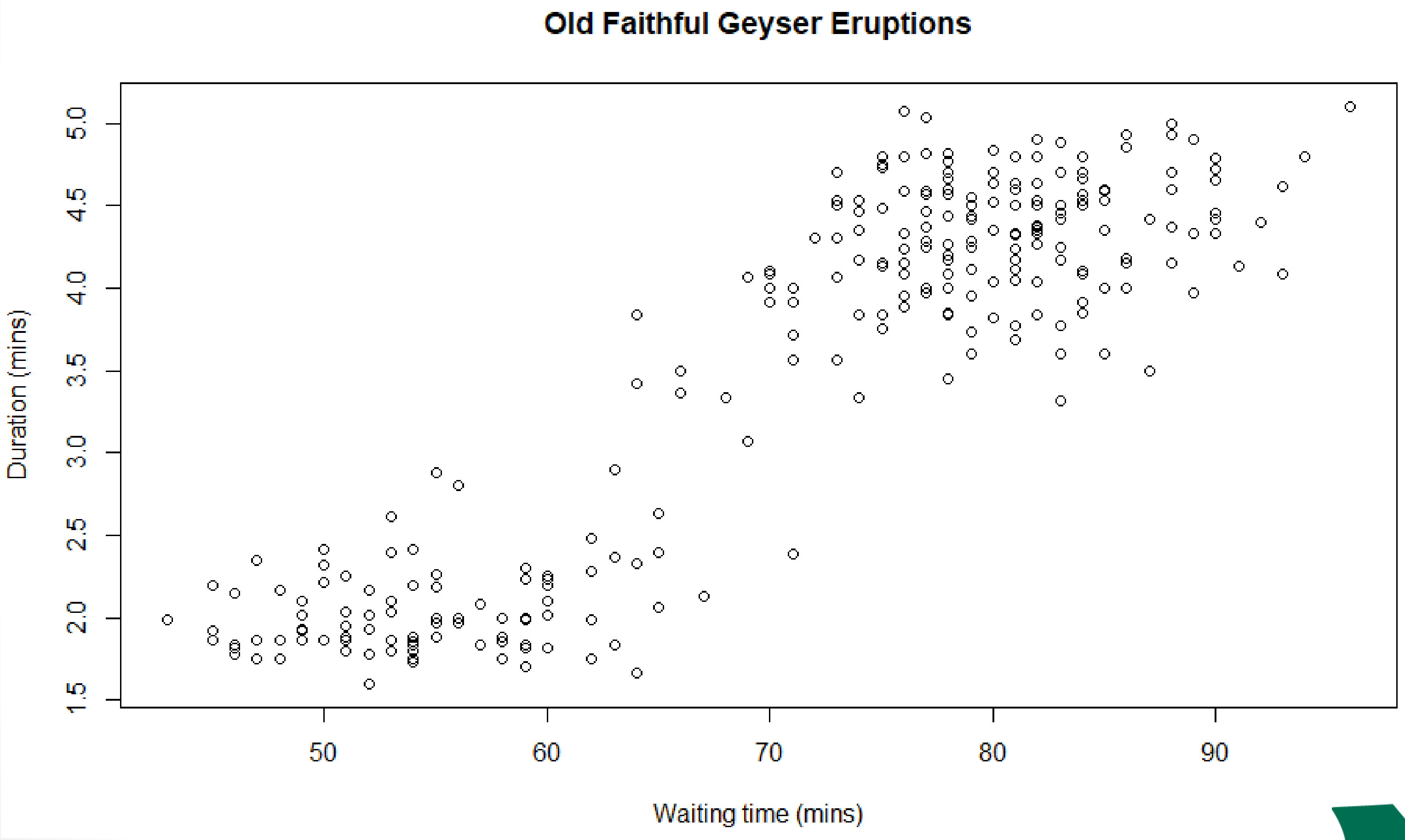
The response is waiting and the predictor is eruption.

```
> summary(faithful)
   eruptions        waiting
 Min.   :1.600   Min.   :43.0
 1st Qu.:2.163   1st Qu.:58.0
 Median :4.000   Median :76.0
 Mean   :3.488   Mean   :70.9
 3rd Qu.:4.454   3rd Qu.:82.0
 Max.   :5.100   Max.   :96.0
```

Example 2: (Simple Linear Regression)

4. Graph the scatterplot and explain the relationship.

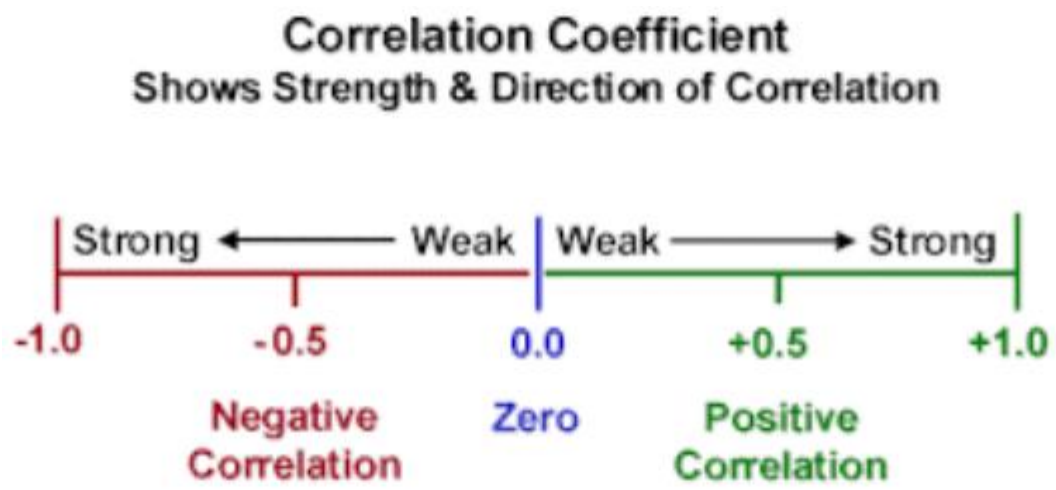
```
> plot(faithful$waiting, faithful$eruptions, main="Old Faithful Geyser Eruptions",
+      xlab="Waiting time (mins)", ylab="Duration (mins)")
```

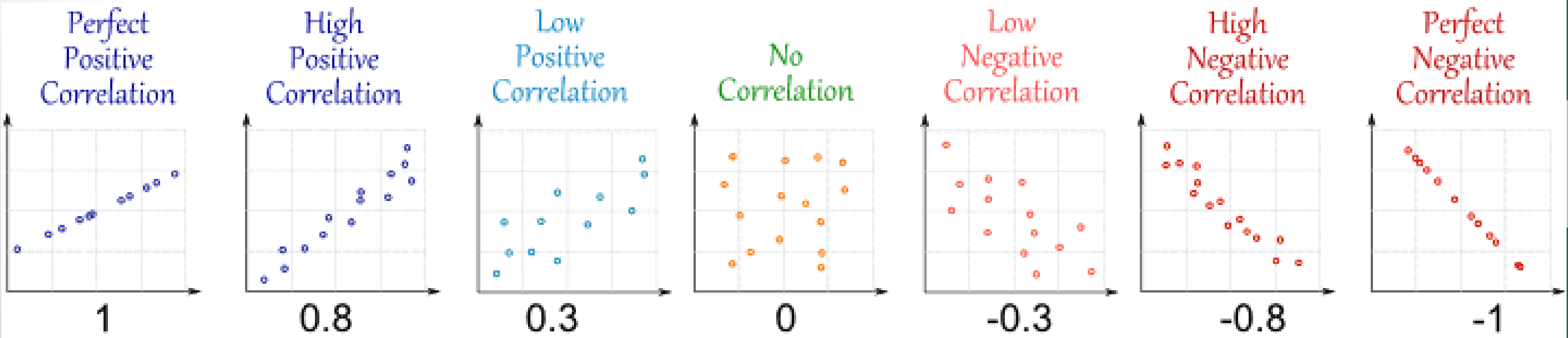


- ➢ There is a strong positive linear association between the waiting and eruption.

Example 2: (Simple Linear Regression)

Note: Pearson correlation coefficient (r) is used to test linear relationships between quantitative attributes.





Example 2: (Simple Linear Regression)

5. Calculate the correlation coefficient using cor() function, and interpret it.

```
> cor(faithful$waiting, faithful$eruptions)
[1] 0.9008112
```

$r = 0.90$, which means that there is strong positive linear association between the waiting and eruption.

- strong, because it close to 1 (> 0.70),
- positive, because it has r is positive.
- linear, because r is used to measure the linear association.

Example 2: (Simple Linear Regression)

6. Fit the linear regression (predictive) model using lm() function.

7. Find the intercept and slope, and interpret them.

```
> lm(eruptions~waiting, data = faithful)

Call:
lm(formula = eruptions ~ waiting, data = faithful)

Coefficients:
(Intercept)      waiting
   -1.87402      0.07563
```

The simple linear regression model is

$$\widehat{Eruptions} = -1.874 + 0.076\,Waiting$$

- ➢ For 0 minutes waiting time, the average eruption time is -1.87 minutes which is impossible (0 is far from the minimum waiting time).
- ➢ For every extra 1 minute of waiting time, the predicted eruption time increases by 0.076 minutes.

Example 2: (Simple Linear Regression)

7. Test whether the predictor is significant (important) or not.

Note: we use p-value, if it was small (usually < 5%), then the predictor is statistically significant.

```
> eruptions_model = lm(eruptions~waiting, data = faithful)
> summary(eruptions_model)

Call:
lm(formula = eruptions ~ waiting, data = faithful)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.29917 -0.37689  0.03508  0.34909  1.19329 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.874016   0.160143  -11.70   <2e-16 ***
waiting      0.075628   0.002219   34.09   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4965 on 270 degrees of freedom
Multiple R-squared:  0.8115,	Adjusted R-squared:  0.8108 
F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

Waiting time explain 81% of the variation in the eruptions duration

Since p-value of the waiting time is very small, so the predictor is statistically significant associated with the response.

Example 2: (Simple Linear Regression)

8. Use the fitted model to predict the duration of eruptions when the waiting time is 78 minutes.

Use predict() function.

```
> predict(eruptions_model, data.frame(waiting = 78))
       1
4.024964
```

For 78 minutes waiting time, the predict duration of eruptions is 4.03 minutes.

Example 2: (Simple Linear Regression)

9. Find the predicted (fitted) values for the first 6 data values.

10. Find the residuals for the first 6 data values.

residuals = actual − predicted

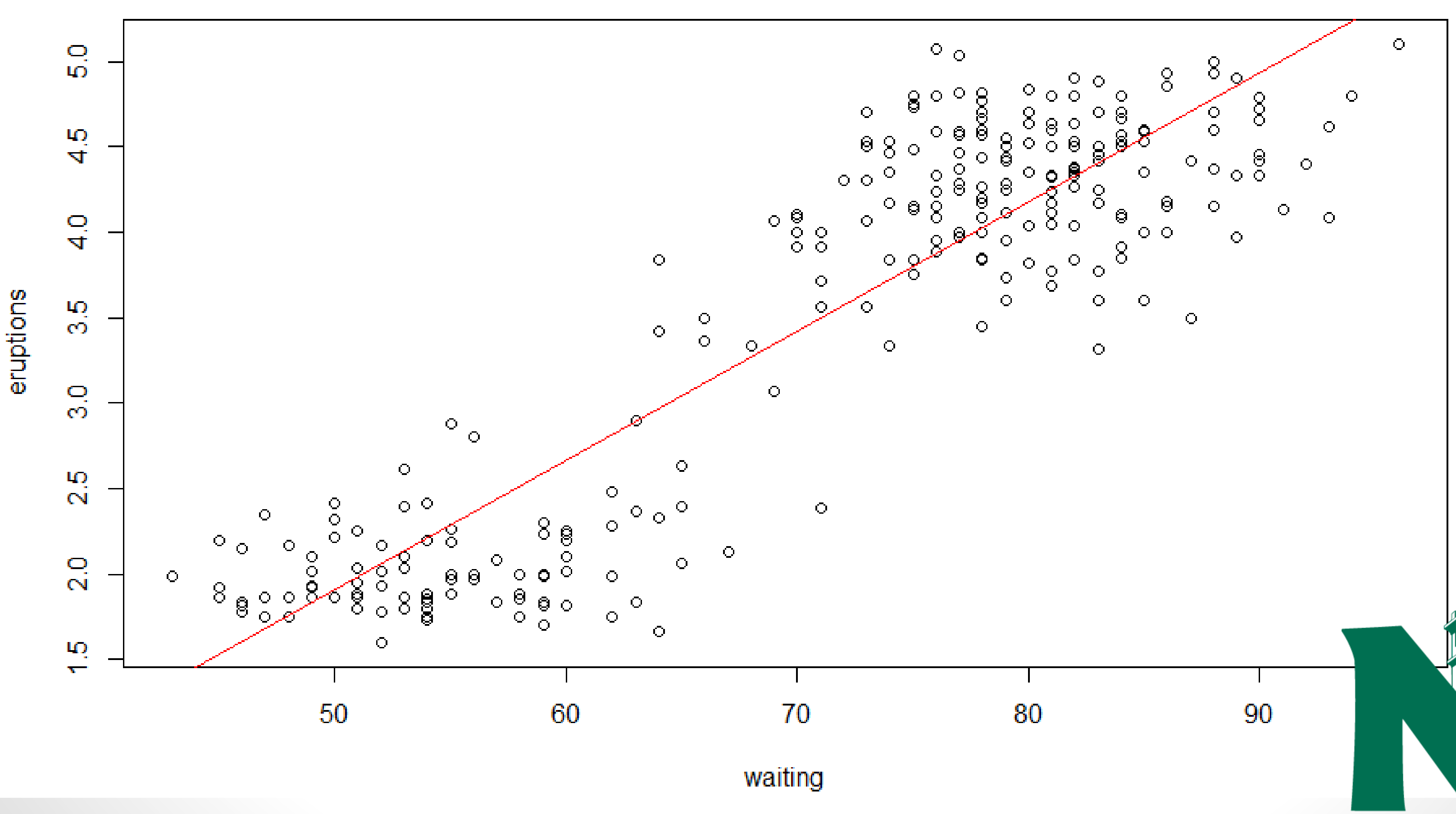
```
> head(eruptions_model$fitted.values)
       1        2        3        4        5        6 
4.100592 2.209893 3.722452 2.814917 4.554360 2.285521 
```

```
> head(eruptions_model$residuals)
          1           2           3           4           5           6 
-0.50059190 -0.40989320 -0.38945216 -0.53191679 -0.02135959  0.59747885 
```

Example 2: (Simple Linear Regression)

11. Graph the scatterplot with the best fit line (predicted values).

```
> plot(eruptions~waiting,data = faithful)
> abline(eruptions_model,col="red")
```



Example 2: (Simple Linear Regression)

12. Consider making predictions of eruption duration for waiting times of 70 and 150 minutes, which is more reliable?

70 is more reliable because it lies within the range of the waiting time

```
> range(faithful$waiting)
[1] 43 96
```

Example 2: (Simple Linear Regression)

Using regression to handle missing values:

13. Create a new data set and remove the 2nd value under eruptions. Create new variable and replace the missing values with the mean, median, and regression predicted values.

Note: we need to install dplyr package.

```
> faithful1 = faithful
> head(faithful1)
  eruptions waiting
1     3.600      79
2     1.800      54
3     3.333      74
4     2.283      62
5     4.533      85
6     2.883      55
```

```
> faithful1[2,1] = NA
> head(faithful1)
  eruptions waiting
1     3.600      79
2        NA      54
3     3.333      74
4     2.283      62
5     4.533      85
6     2.883      55
```

Example 2: (Simple Linear Regression)

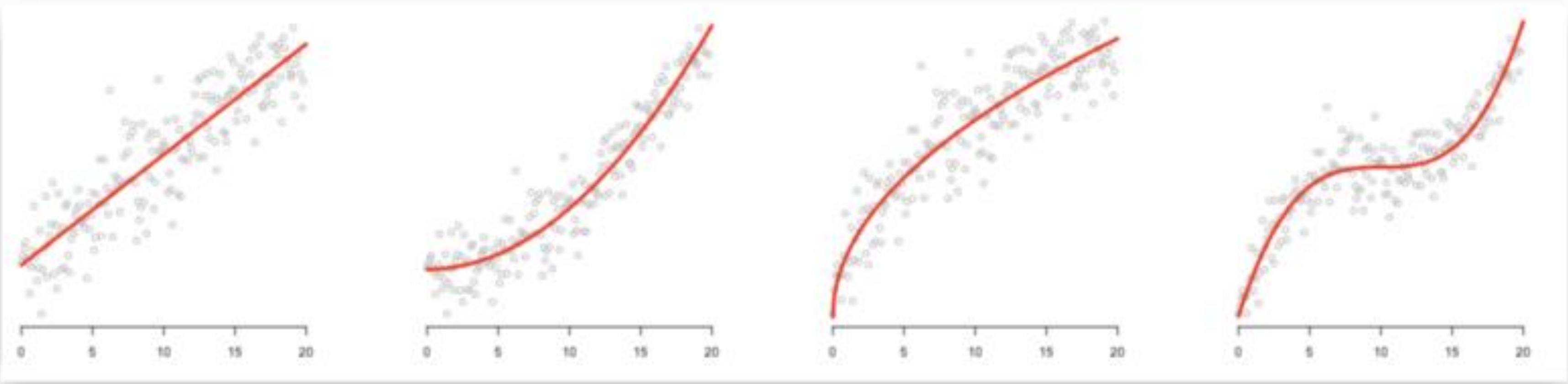
```
> faithful_replace <- faithful1 %>%
+   mutate(replace_mean_eruptions  = ifelse(is.na(eruptions), mean(faithful1$eruptions, na.rm=TRUE), eruptions),
+   replace_median_eruptions  = ifelse(is.na(eruptions), median(faithful1$eruptions, na.rm=TRUE), eruptions),
+   replace_regression_eruptions  = ifelse(is.na(eruptions), predict(eruptions_model1, data.frame(waiting = c(5
4))), eruptions))
>
> head(faithful_replace)
  eruptions waiting replace_mean_eruptions replace_median_eruptions replace_regression_eruptions
1     3.600      79               3.600000                    3.600                     3.600000
2        NA      54               3.494011                    4.000                     2.213773
3     3.333      74               3.333000                    3.333                     3.333000
4     2.283      62               2.283000                    2.283                     2.283000
5     4.533      85               4.533000                    4.533                     4.533000
6     2.883      55               2.883000                    2.883                     2.883000
```

- The actual value was 1.800, and we can observe that the regression predicted value is the closet one.
- Mean and median can work better if the missing value was close to the center of the data values, but since it is a missing value, so we don’t know its location.

1.2 Polynomial Regression Model:

The polynomial models can be used in case where the relationship between the response and predictor is curvilinear.

Note: the polynomial model is a linear model because it is linear in coefficients ($\beta_1, \beta_2, ..., \beta_k$)



$Y' = a + b_1 X_1$	Linear
$Y' = a + b_1 X_1 + b_2 X_1^2$	Quadratic
$Y' = a + b_1 X_1 + b_2 X_1^2 + b_3 X_1^3$	Cubic

Example 2: (Polynomial Regression)

ElectricityLoad.xlsx dataset (on canvas) contains the hourly electricity load and temperature in south New Jersey over one year.

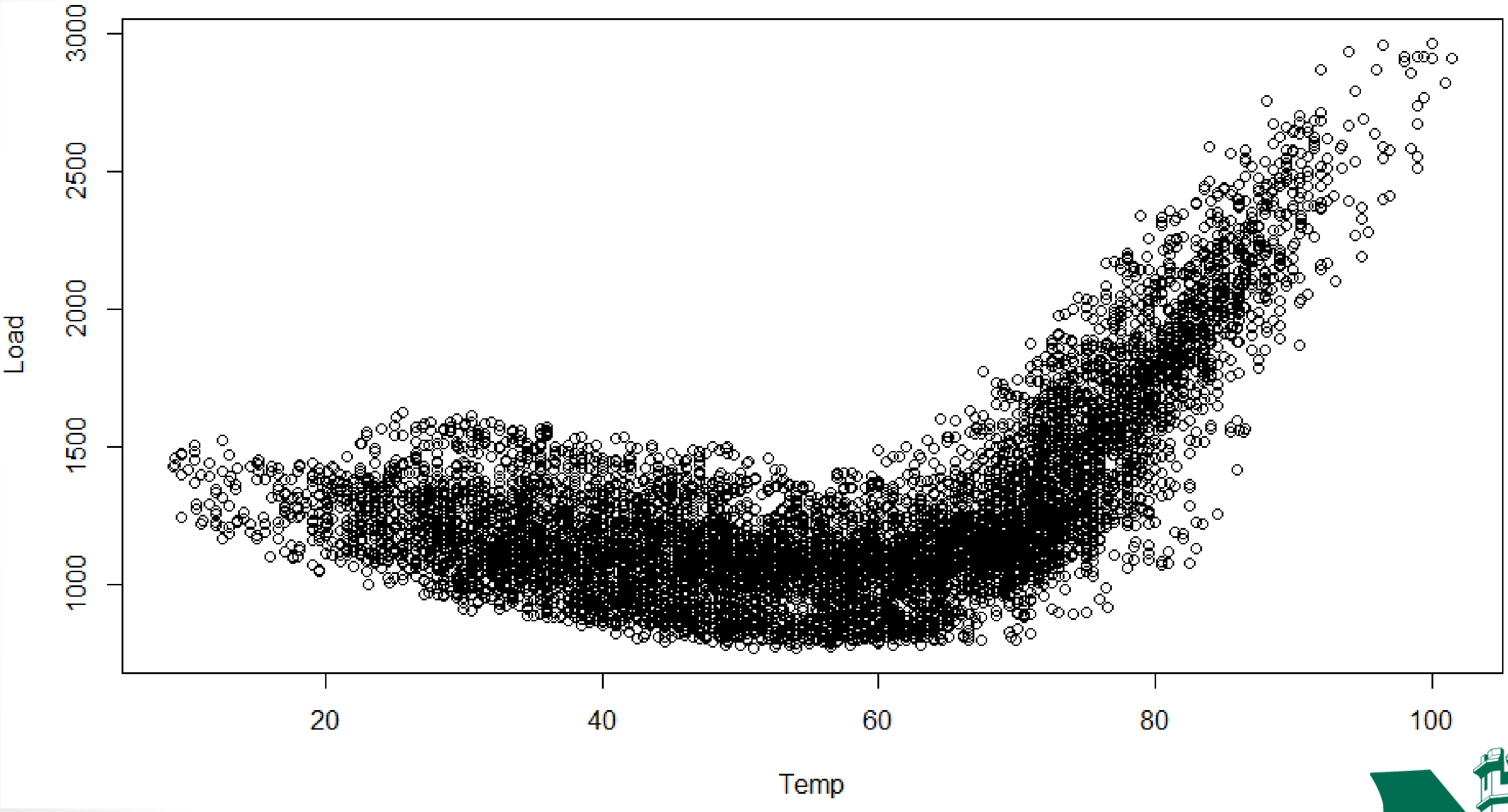
1. Identify the response and predictor.
2. Present the summary statistics.

```
> summary(ElectricityLoad)
      Temp              Load       
 Min.   :  8.97   Min.   : 768.2  
 1st Qu.: 42.50   1st Qu.:1050.5  
 Median : 58.01   Median :1175.1  
 Mean   : 56.60   Mean   :1271.2  
 3rd Qu.: 71.53   3rd Qu.:1383.4  
 Max.   :101.48   Max.   :2966.2  
```

Example 2: (Polynomial Regression)

3. Graph the scatterplot and interpret the association.

```
> plot(Load~Temp, data = ElectricityLoad)
```



- ➢ The relationship between the variables is nonlinear.

Example 2: (Polynomial Regression)

4. Fit the linear regression model and write down the model.

```
> Load_Lin = lm(Load~Temp, data = ElectricityLoad)
> summary(Load_Lin)

Call:
lm(formula = Load ~ Temp, data = ElectricityLoad)

Residuals:
    Min      1Q  Median      3Q     Max 
-613.35 -222.33  -50.98  186.15 1282.57 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  691.997     10.407   66.50   <2e-16 ***
Temp          10.234      0.175   58.48   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 298.9 on 8758 degrees of freedom
Multiple R-squared:  0.2808,	Adjusted R-squared:  0.2807 
F-statistic:  3420 on 1 and 8758 DF,  p-value: < 2.2e-16
```

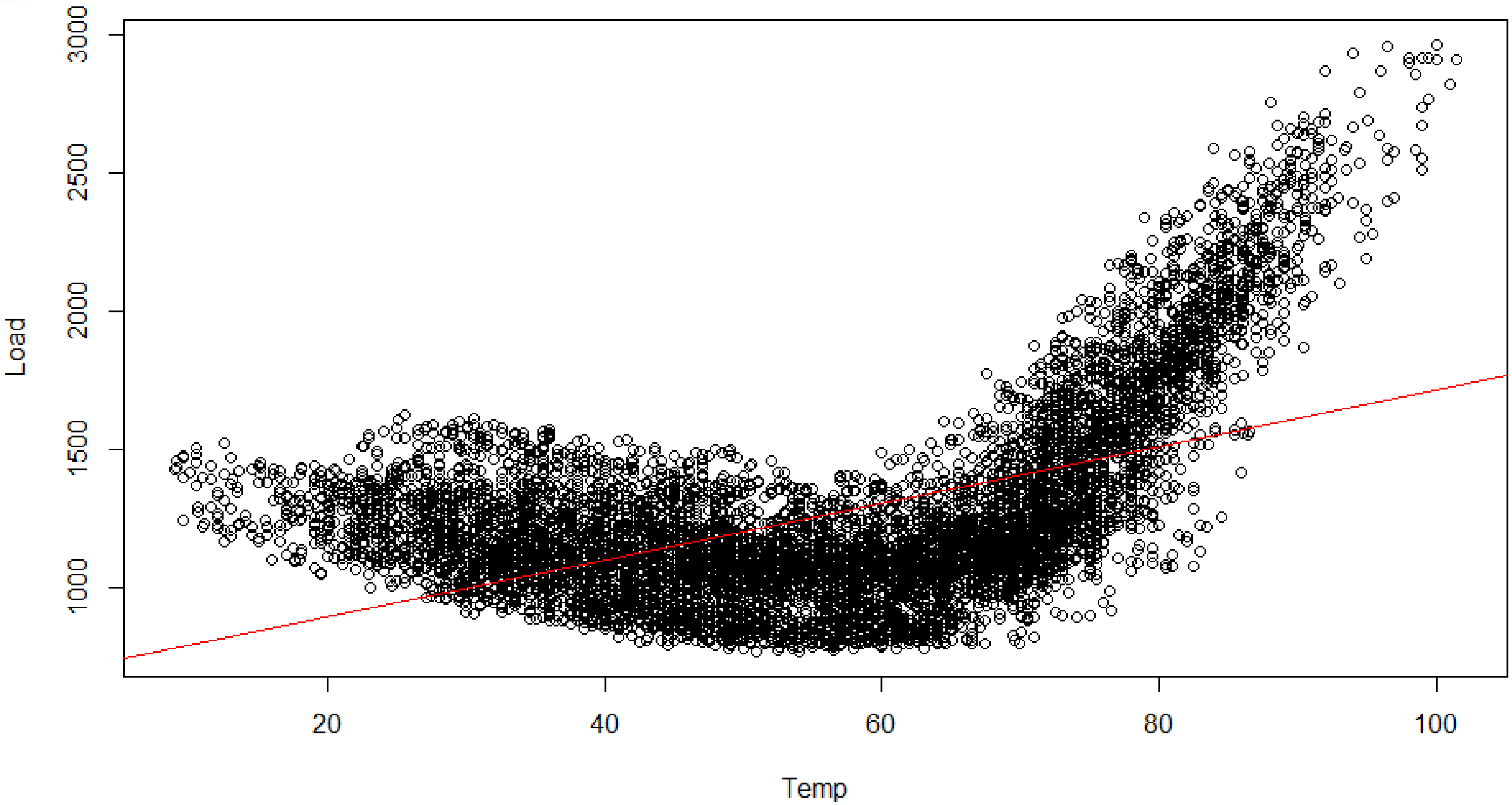
$$Load = 691.997 + 10.234\,Temp$$

Example 2: (Polynomial Regression)

5. Graph the actual and predicted values. Does the linear model fit the data well?

```
> plot(Load~Temp, data = ElectricityLoad)
> abline(Load_Lin, col="red")
```

```
> plot(ElectricityLoad$Temp, ElectricityLoad$Load)
> lines(ElectricityLoad$Temp, fitted(Load_Lin), col="red")
```

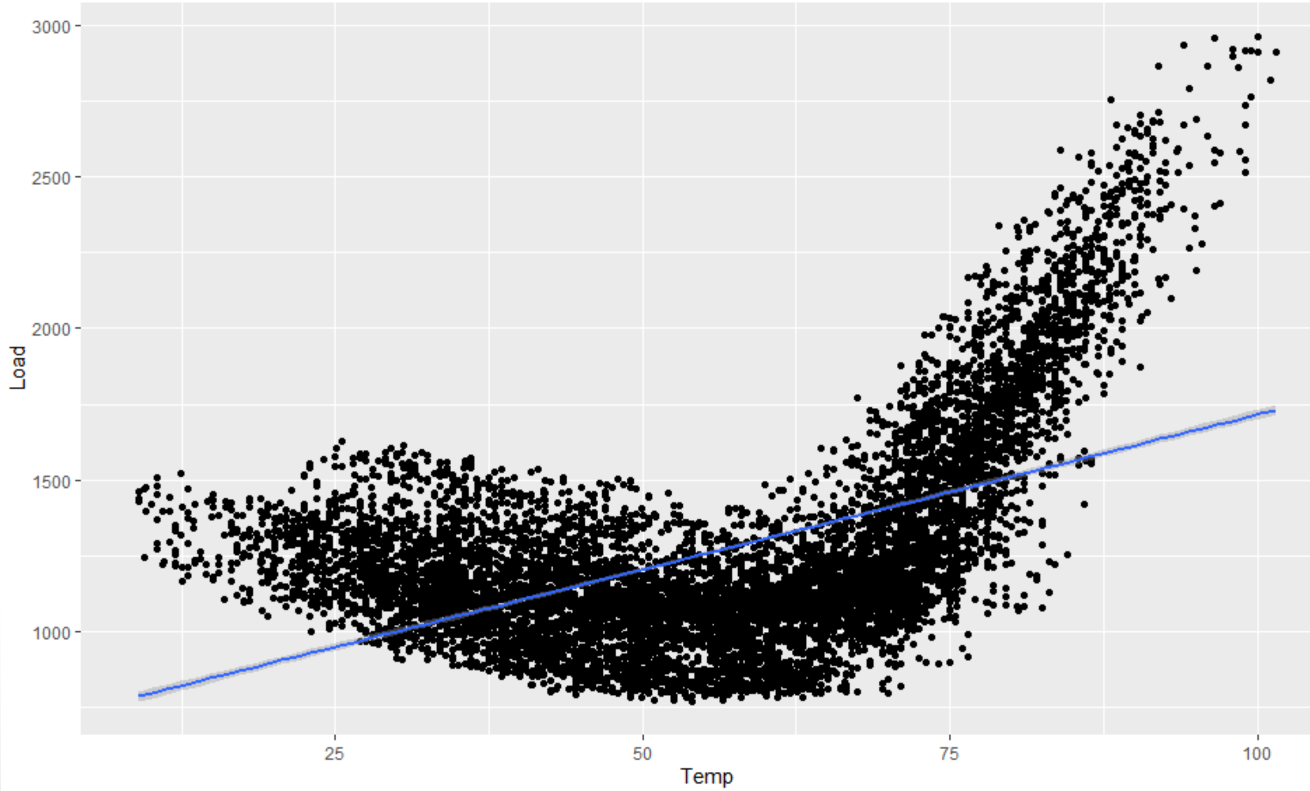


Example 2: (Polynomial Regression)

6. Graph the actual and predicted values. Does the linear model fit the data well? Use ggplot() function

```
install.packages("ggplot2")
library("ggplot2")
```

```
> ggplot(ElectricityLoad, aes(Temp, Load)) +
+   geom_point() +
+   geom_smooth(method = "lm", formula = y ~ x)
```



Example 2: (Polynomial Regression)

7. Fit the quadratic (2nd degree polynomial) regression model and write down the model.

```
> Load_Quad = lm(Load~poly(Temp,2), data = ElectricityLoad)
> summary(Load_Quad)

Call:
lm(formula = Load ~ poly(Temp, 2), data = ElectricityLoad)

Residuals:
    Min      1Q  Median      3Q     Max 
-731.45 -136.34   -6.99  119.68  711.71 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)     1271.200      2.103  604.60   <2e-16 ***
poly(Temp, 2)1 17477.065    196.789   88.81   <2e-16 ***
poly(Temp, 2)2 21051.457    196.789  106.97   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 196.8 on 8757 degrees of freedom
Multiple R-squared:  0.6882,	Adjusted R-squared:  0.6882 
F-statistic:  9665 on 2 and 8757 DF,  p-value: < 2.2e-16
```

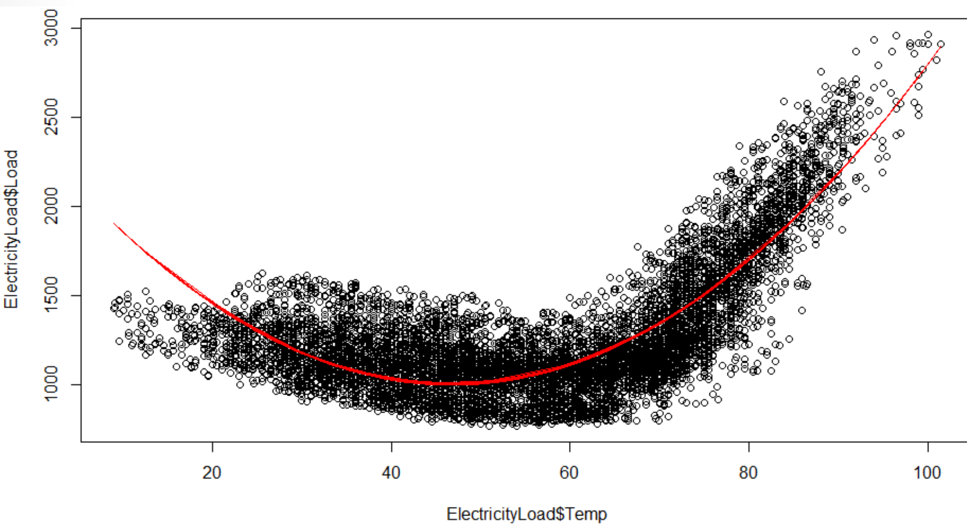
$$Load = 1271.2 + 17477.1\, Temp + 21051.5\, Temp^2$$

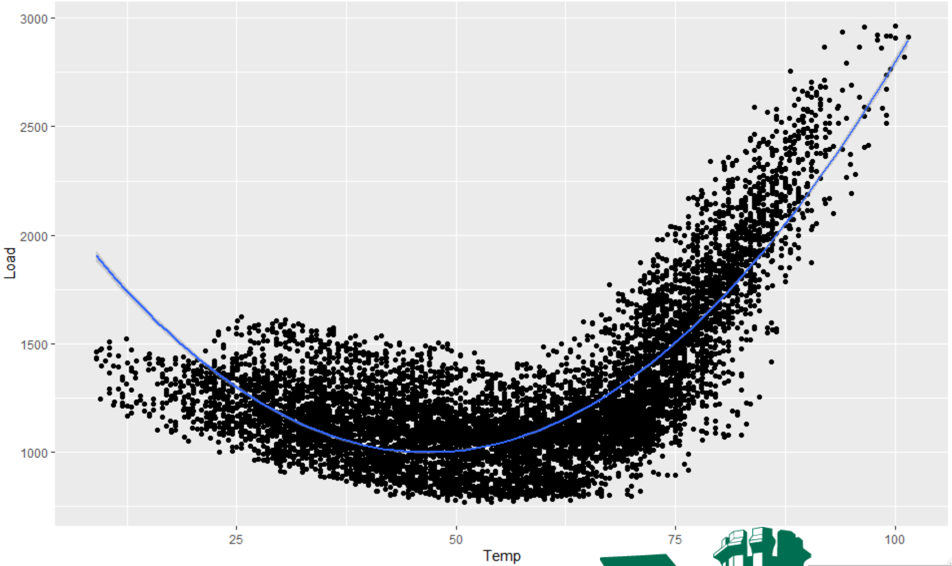
Example 2: (Polynomial Regression)

8. Graph the actual and predicted values. Does the quadratic regression model fit the data well?

```
> plot(ElectricityLoad$Temp, ElectricityLoad$Load)
> lines(ElectricityLoad$Temp, fitted(Load_Quad), col="red")
```

```
> ggplot(ElectricityLoad, aes(Temp, Load)) +
+   geom_point() +
+   geom_smooth(method = "lm", formula = y ~ x + I(x^2))
```





Example 2: (Polynomial Regression)

9. Fit the cubic (3rd degree polynomial) regression model and write down the model.

```
> Load_Cub = lm(Load~poly(Temp,3), data = ElectricityLoad)
> summary(Load_Cub)

Call:
lm(formula = Load ~ poly(Temp, 3), data = ElectricityLoad)

Residuals:
    Min      1Q  Median      3Q     Max 
-773.29 -127.63   -7.03  107.12  701.80 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)     1271.200      1.932  657.91   <2e-16 ***
poly(Temp, 3)1 17477.065    180.843   96.64   <2e-16 ***
poly(Temp, 3)2 21051.457    180.843  116.41   <2e-16 ***
poly(Temp, 3)3  7264.103    180.843   40.17   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 180.8 on 8756 degrees of freedom
Multiple R-squared:  0.7367,    Adjusted R-squared:  0.7367 
F-statistic:  8168 on 3 and 8756 DF,  p-value: < 2.2e-16
```

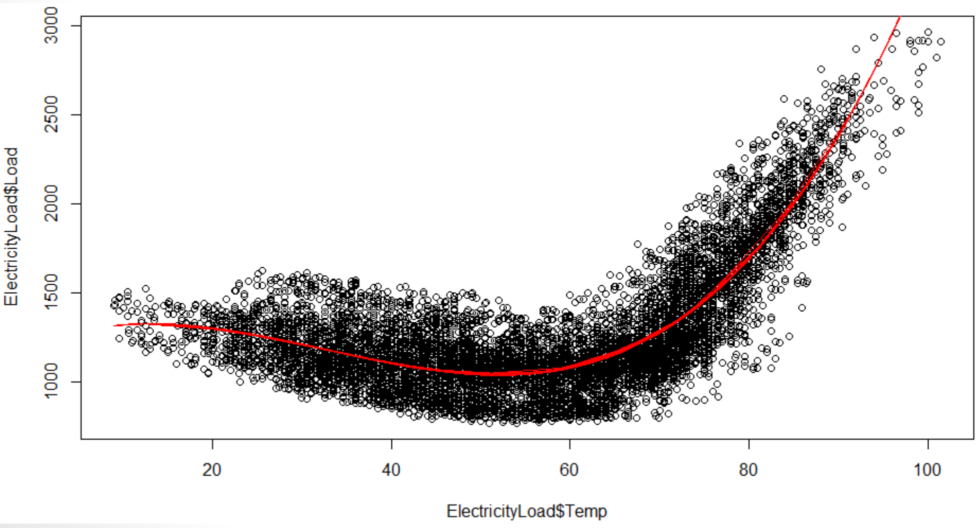
$$Load = 1271.2 + 17477.1\,Temp + 21051.5\,Temp^2 + 7264.1\,Temp^3$$

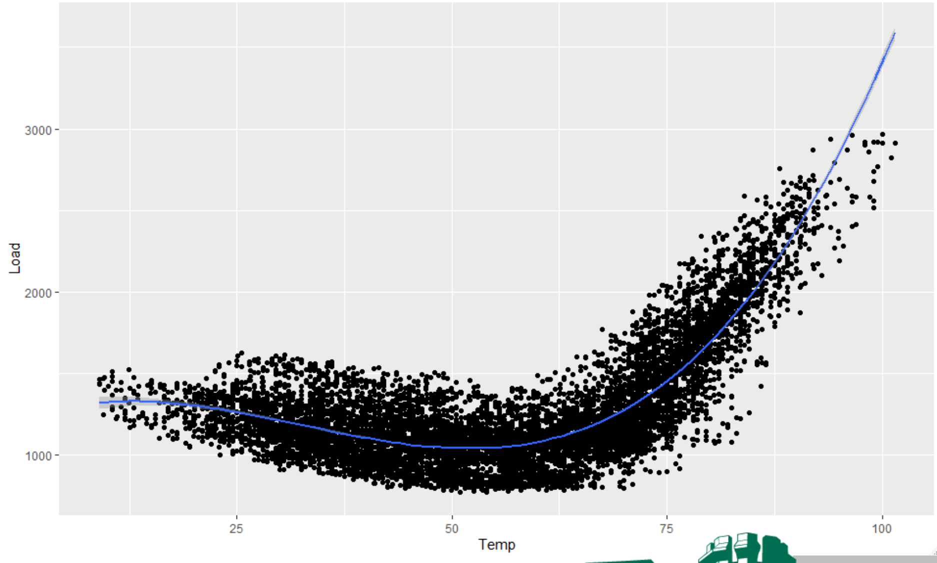
Example 2: (Polynomial Regression)

10. Graph the actual and predicted values. Does the cubic regression model fit the data well?

```
> plot(ElectricityLoad$Temp, ElectricityLoad$Load)
> lines(ElectricityLoad$Temp, fitted(Load_Cub), col="red")
```

```
> ggplot(ElectricityLoad, aes(Temp, Load)) +
+   geom_point() +
+   geom_smooth(method = "lm", formula = y ~ x + I(x^2) + I(x^3))
```





The Regression Assumptions (Conditions):

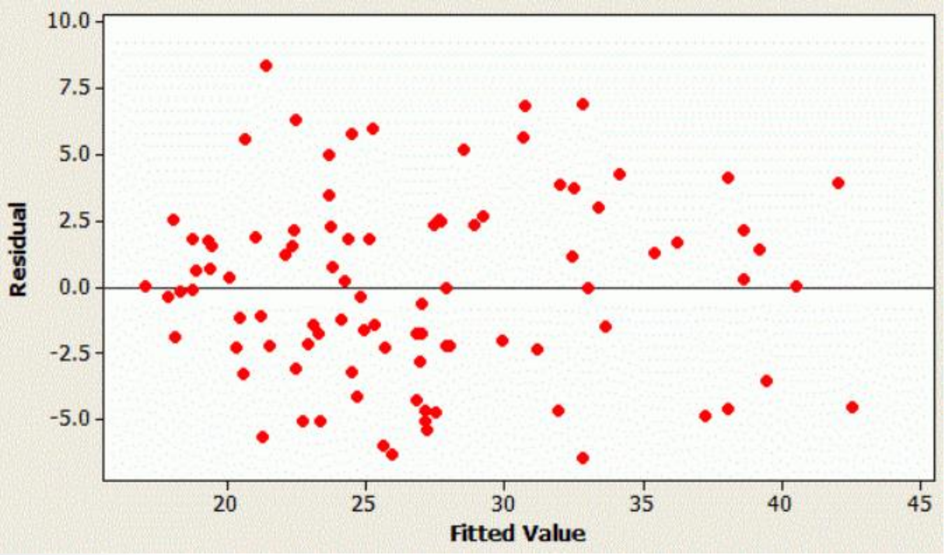
1. **Linearity:** The response attribute to have a roughly linear relationship with each of the predictors.
2. **Homoscedasticity:** The variance of the residuals should be the same at each level of the predictors.
3. **Independence:** This means that residuals should be uncorrelated.
4. **Normality:** The residuals should be normally distributed.
5. **Outliers/influential cases:** It is important to look out for cases which may have a disproportionate influence over your regression model.

Note: the residuals are the difference between the each actual response value and the predicted value using the model.

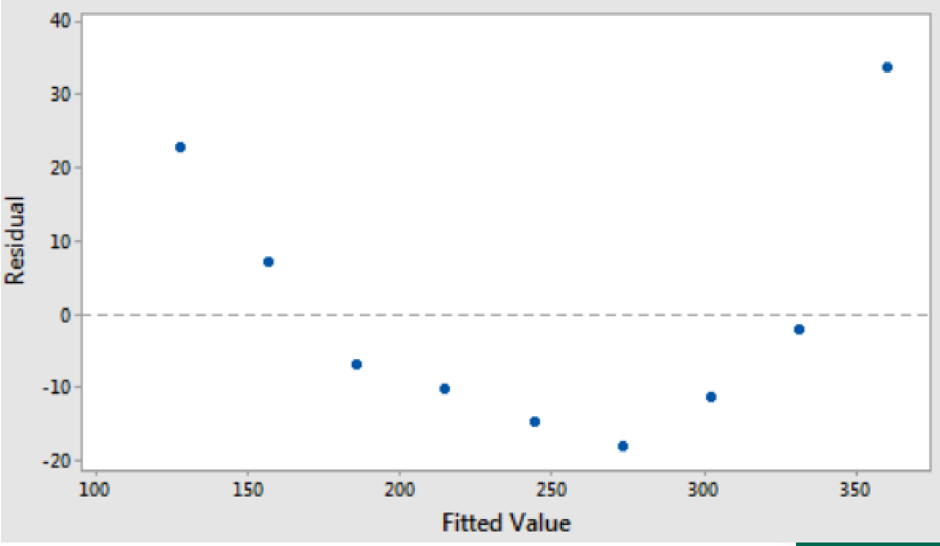
Assessing the Regression Assumptions:

Create a *scatterplot* with the residuals on the vertical axis and the fitted values $\hat{y}_i$ on the horizontal axis.

The residuals plot should look like this

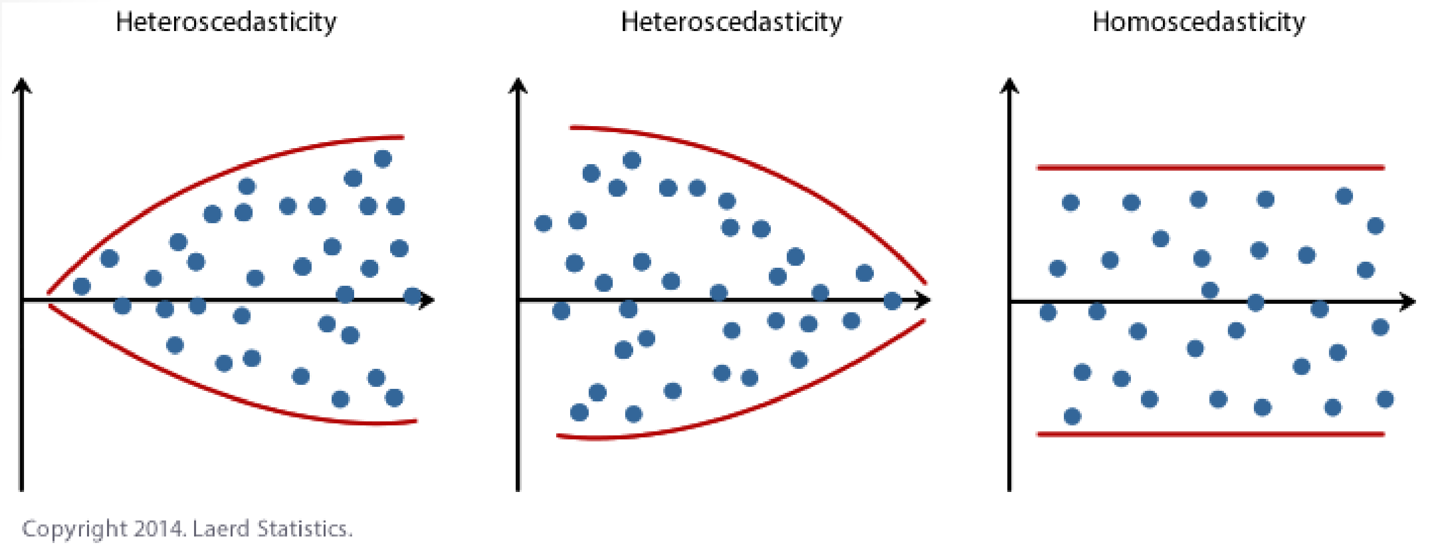


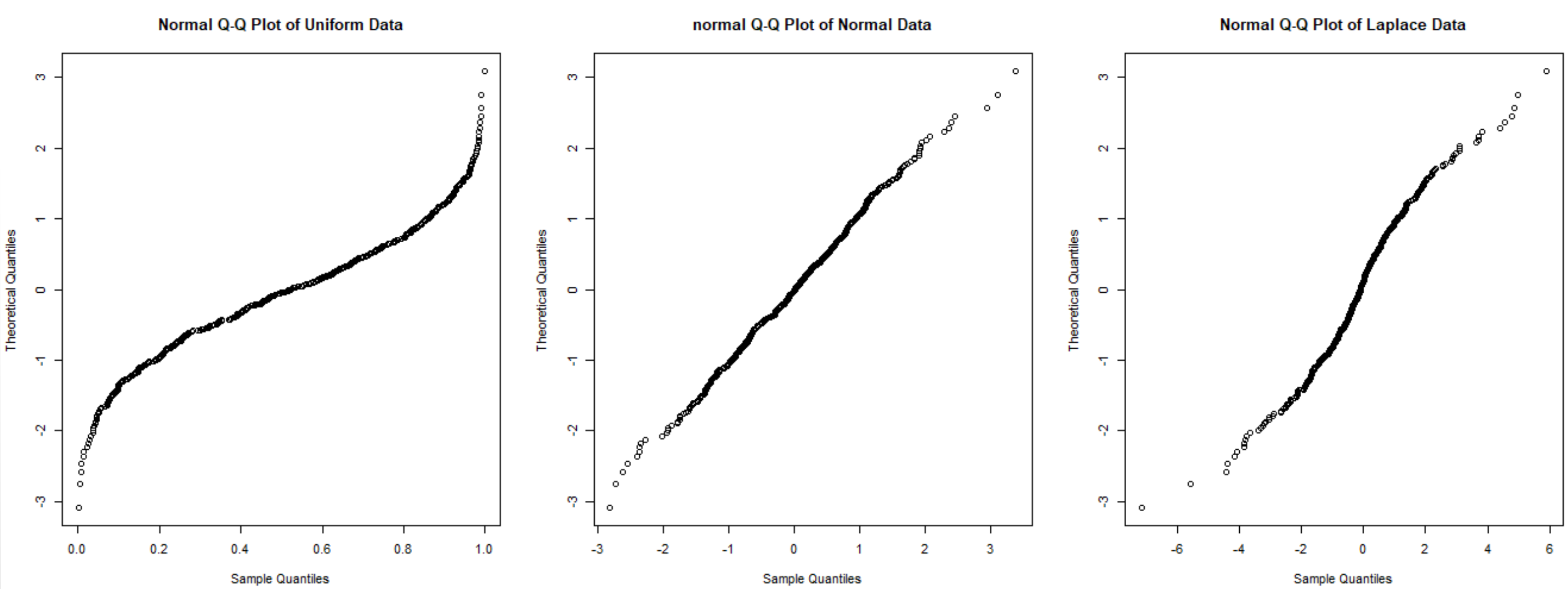
Linear



Non-Linear

Assessing the Regression Assumptions:



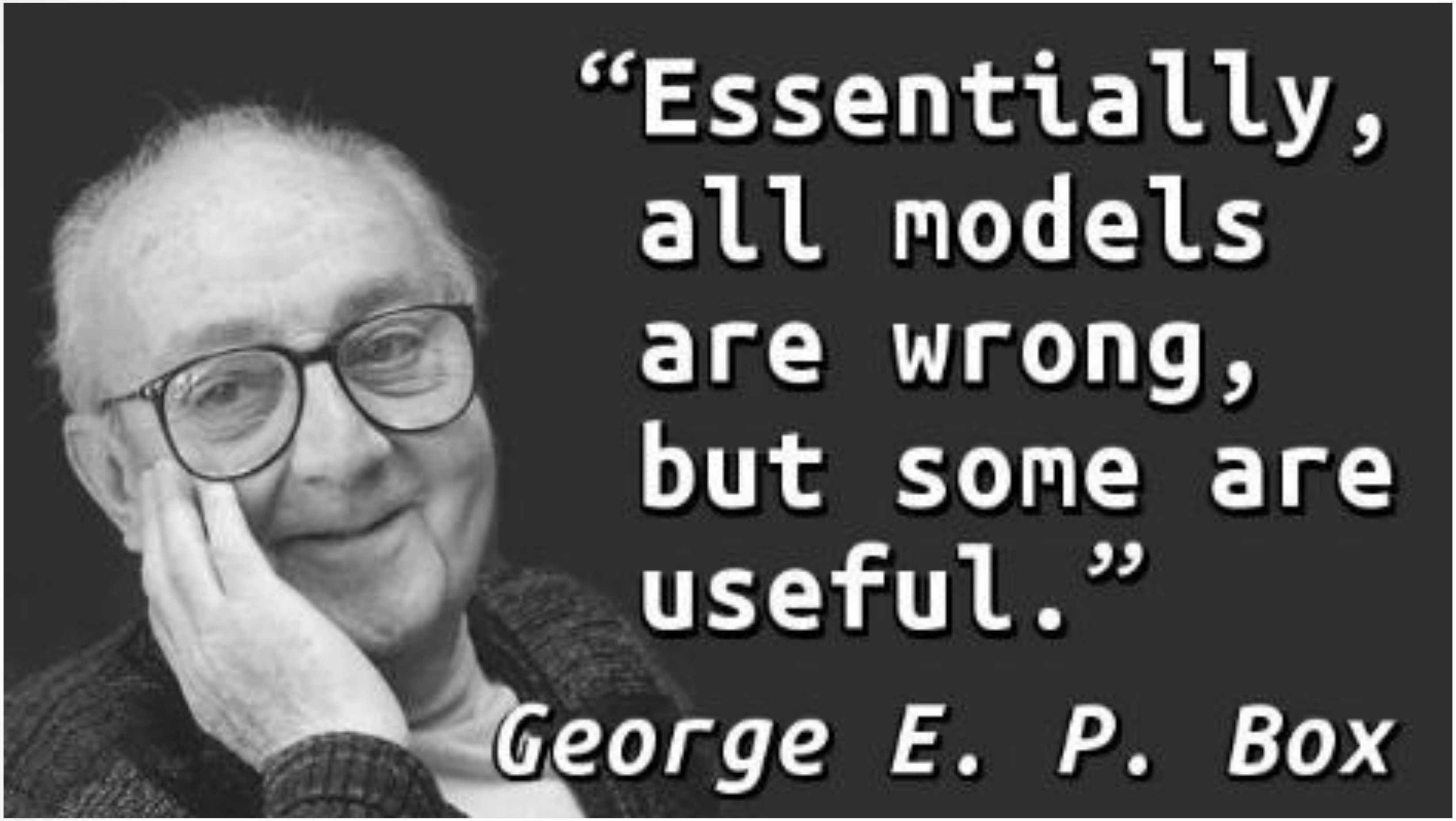


Non-normal Normal Non-normal

How to determine which model is best fit?

Various criteria can be used to judge the quality of a model. These include *adjusted* R^2 (higher the better), *Akaike information criterion* (AIC) (lower the better), *Bayesian information criterion* (BIC) (lower the better), and *Mallow's* C_p (= # of coefficients + 1).

Unfortunately, there are a total of 2^p models that contain subsets of p attributes.


"Essentially, all models are wrong, but some are useful."
George E. P. Box

Example 3: (Multiple Linear regression)

The dataset "*mtcars*" was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

1. Present the data structure.

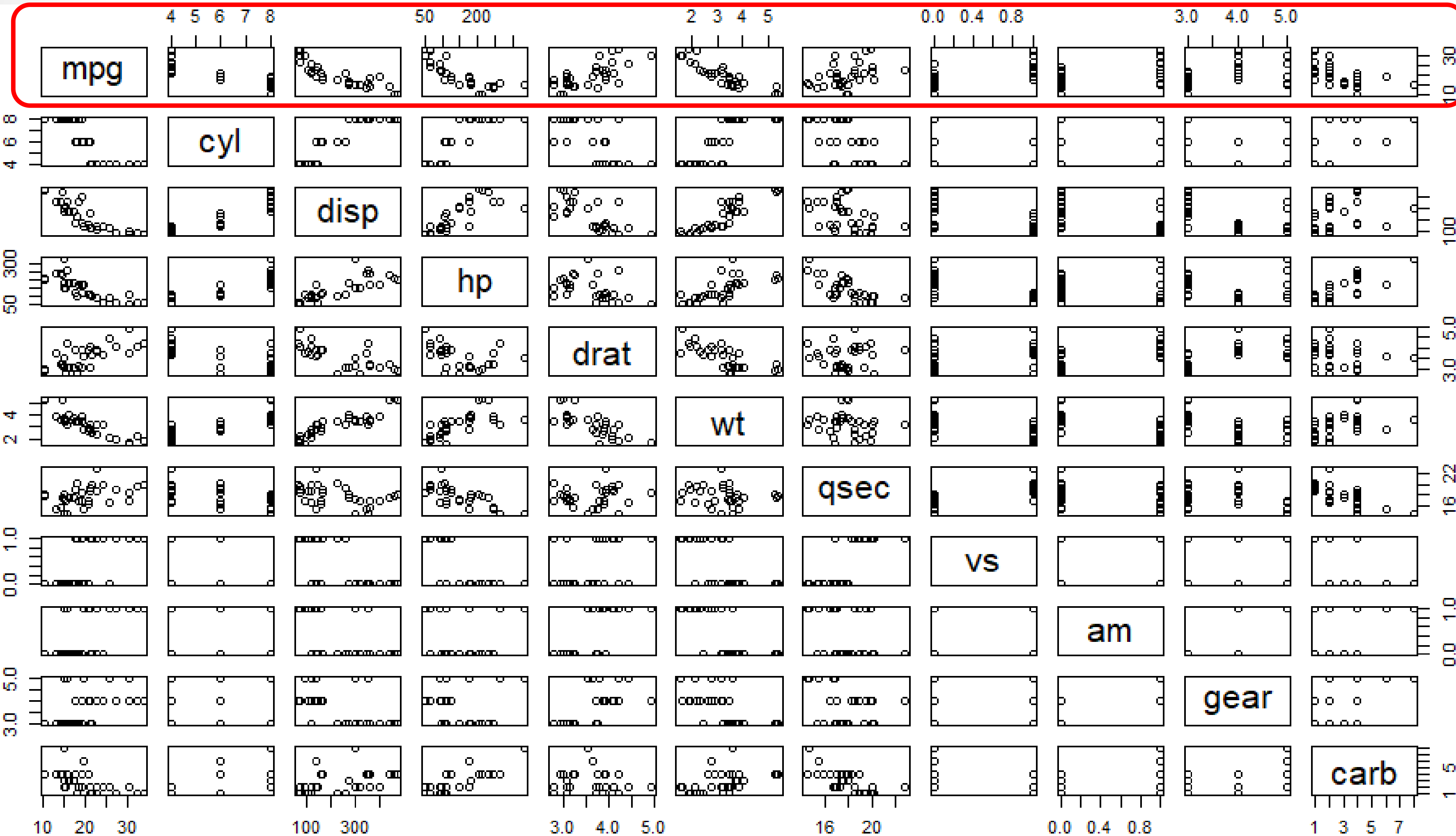
```
> str(mtcars)
'data.frame':   32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

Example 3: (Multiple Linear regression)

2. Display the scatterplot matrix.

(The response is mile per gallon - mpg)

```
> pairs(mtcars)
```



Example 3: (Multiple Linear regression)

3. Calculate the correlation matrix.

Note: you can use cor() function and round the results.

```
> round(cor(mtcars), digits = 2)
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1.00	-0.85	-0.85	-0.78	0.68	-0.87	0.42	0.66	0.60	0.48	-0.55
cyl	-0.85	1.00	0.90	0.83	-0.70	0.78	-0.59	-0.81	-0.52	-0.49	0.53
disp	-0.85	0.90	1.00	0.79	-0.71	0.89	-0.43	-0.71	-0.59	-0.56	0.39
hp	-0.78	0.83	0.79	1.00	-0.45	0.66	-0.71	-0.72	-0.24	-0.13	0.75
drat	0.68	-0.70	-0.71	-0.45	1.00	-0.71	0.09	0.44	0.71	0.70	-0.09
wt	-0.87	0.78	0.89	0.66	-0.71	1.00	-0.17	-0.55	-0.69	-0.58	0.43
qsec	0.42	-0.59	-0.43	-0.71	0.09	-0.17	1.00	0.74	-0.23	-0.21	-0.66
vs	0.66	-0.81	-0.71	-0.72	0.44	-0.55	0.74	1.00	0.17	0.21	-0.57
am	0.60	-0.52	-0.59	-0.24	0.71	-0.69	-0.23	0.17	1.00	0.79	0.06
gear	0.48	-0.49	-0.56	-0.13	0.70	-0.58	-0.21	0.21	0.79	1.00	0.27
carb	-0.55	0.53	0.39	0.75	-0.09	0.43	-0.66	-0.57	0.06	0.27	1.00

- The response (mpg) has a strong negative association with cyl, disp, and wt.

Example 3: (Multiple Linear regression)

4. Fit the multiple linear regression model.

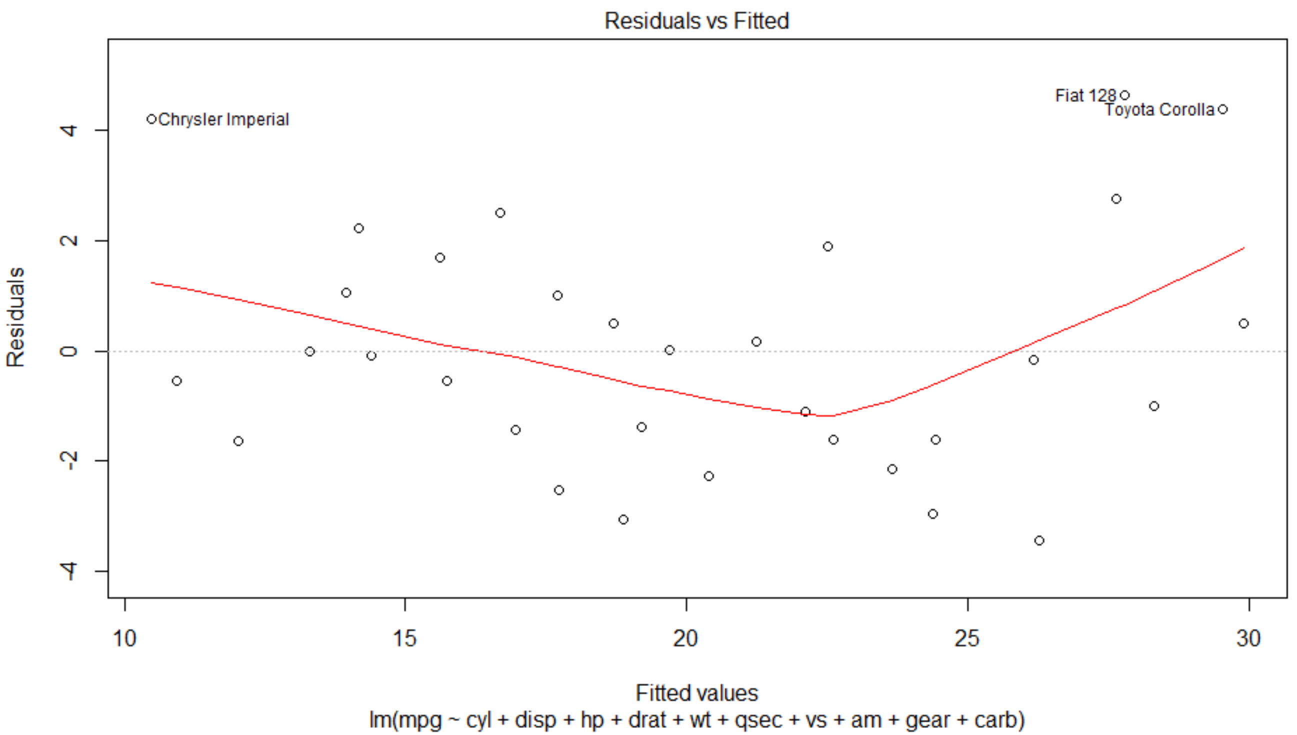
```
> mtcars_model = lm(mpg~cyl+disp+hp+drat+wt+qsec+vs+am+gear+carb, data = mtcars)
> summary(mtcars_model)

Call:
lm(formula = mpg ~ cyl + disp + hp + drat + wt + qsec + vs + 
    am + gear + carb, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.4506 -1.6044 -0.1196  1.2193  4.6271 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 12.30337   18.71788   0.657   0.5181  
cyl         -0.11144    1.04502  -0.107   0.9161  
disp         0.01334    0.01786   0.747   0.4635  
hp          -0.02148    0.02177  -0.987   0.3350  
drat         0.78711    1.63537   0.481   0.6353  
wt          -3.71530    1.89441  -1.961   0.0633 .
qsec         0.82104    0.73084   1.123   0.2739  
vs           0.31776    2.10451   0.151   0.8814  
am           2.52023    2.05665   1.225   0.2340  
gear         0.65541    1.49326   0.439   0.6652  
carb        -0.19942    0.82875  -0.241   0.8122  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.65 on 21 degrees of freedom
Multiple R-squared:  0.869,	Adjusted R-squared:  0.8066 
F-statistic: 13.93 on 10 and 21 DF,  p-value: 3.793e-07
```

- Only *wt* (car weight) has a significant association with the response at significance level 10%.

Example 3: (Multiple Linear regression)

5. Check the regression assumptions.

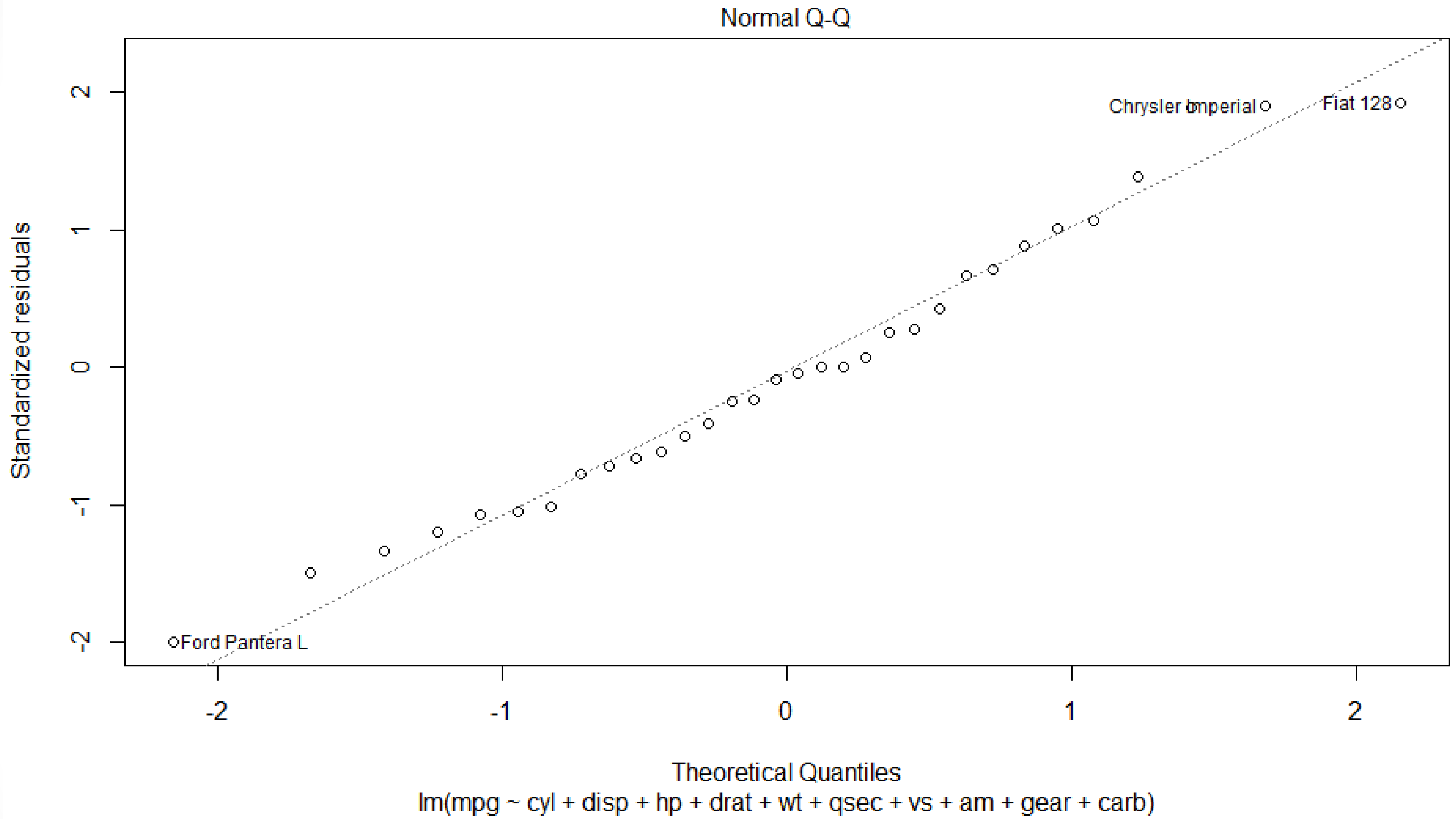
```
> plot(mtcars_model)
Hit <Return> to see next plot: qqnorm(mtcars_model$residuals)
Hit <Return> to see next plot: qqline(mtcars_model$residuals)
Hit <Return> to see next plot:
Hit <Return> to see next plot:
```



- The residuals are randomly distributed around zero with not pattern, so the linearity, Homoscedasticity (constant variance), and independence are met.

Example 3: (Multiple Linear regression)

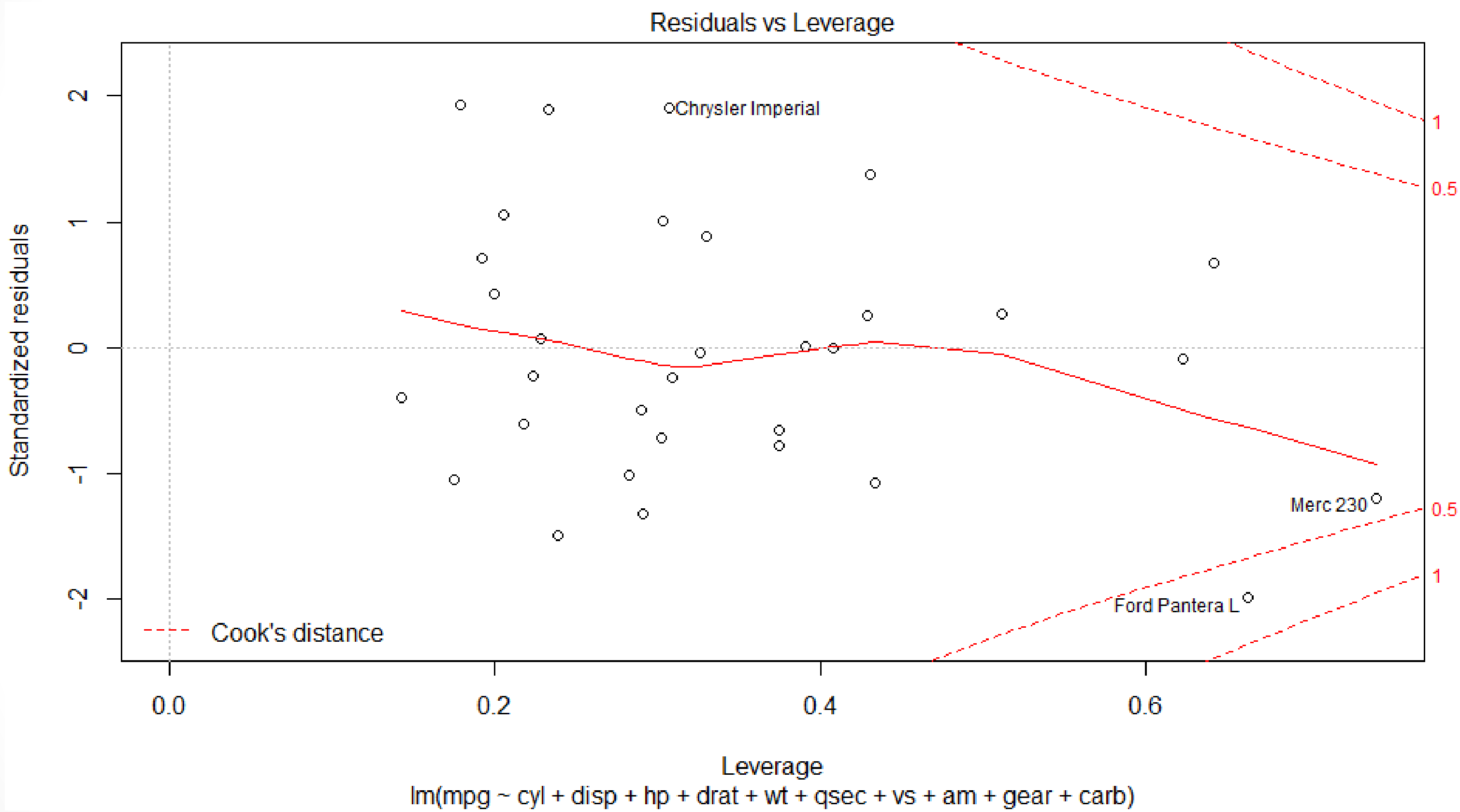
5. Check the regression assumptions. (continued)



➢ The residuals are normally distributed because all the residuals are approximately on a straight line.

Example 3: (Multiple Linear regression)

5. Check the regression assumptions.



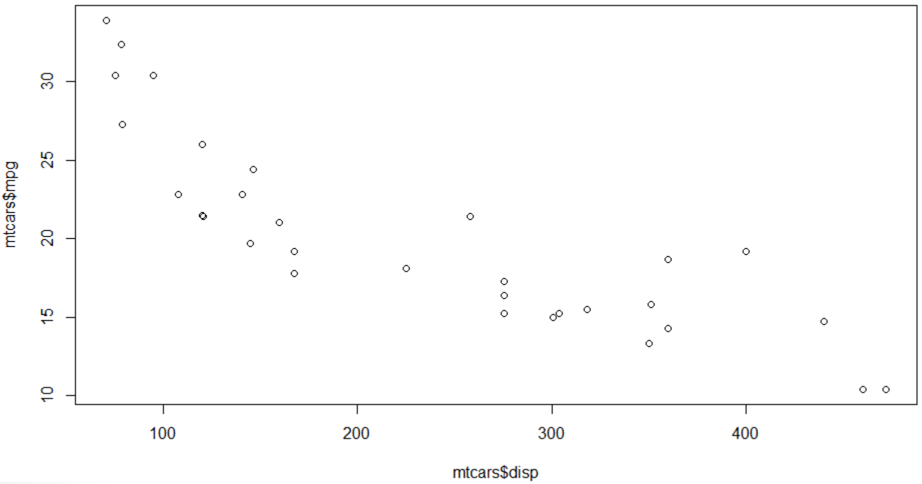
- All cases are well inside of the Cook's distance lines (a red dashed line), so there is no influential case (outlier).

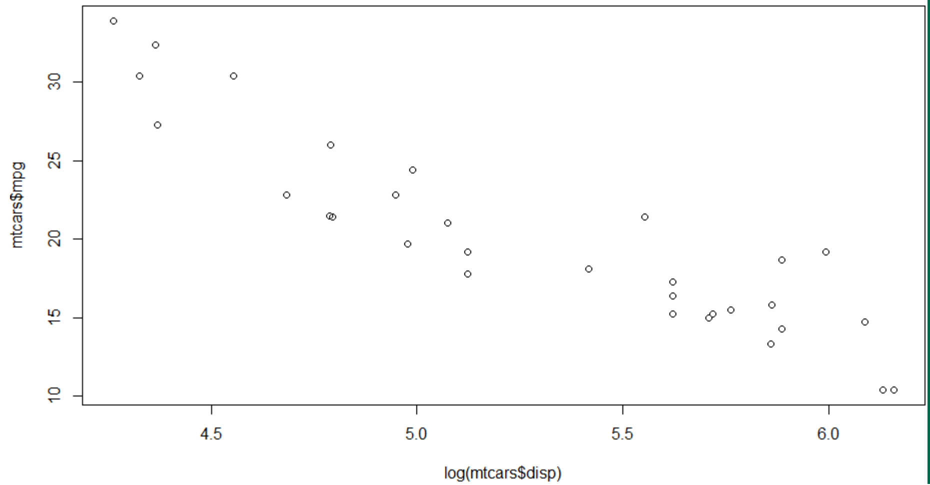
Example 3: (Multiple Linear regression)

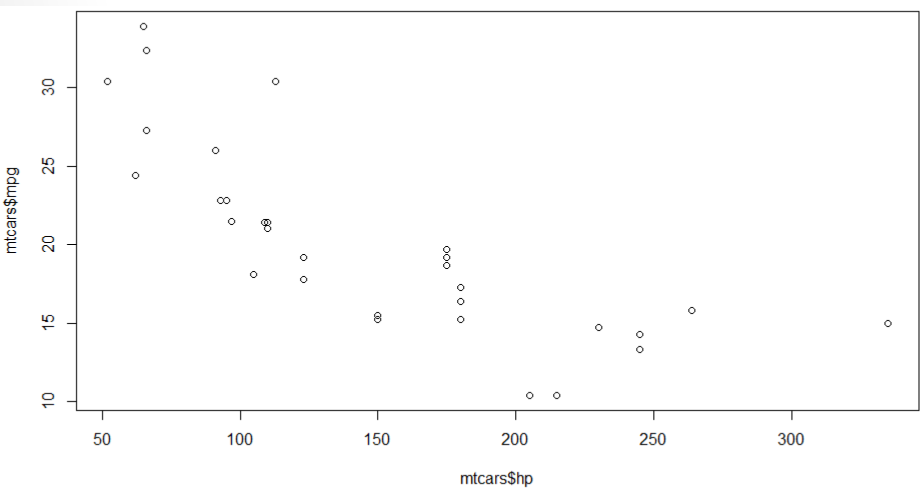
6. Graph the scatterplot of the response versus disp and hp. Is there a linear association? Transformation?

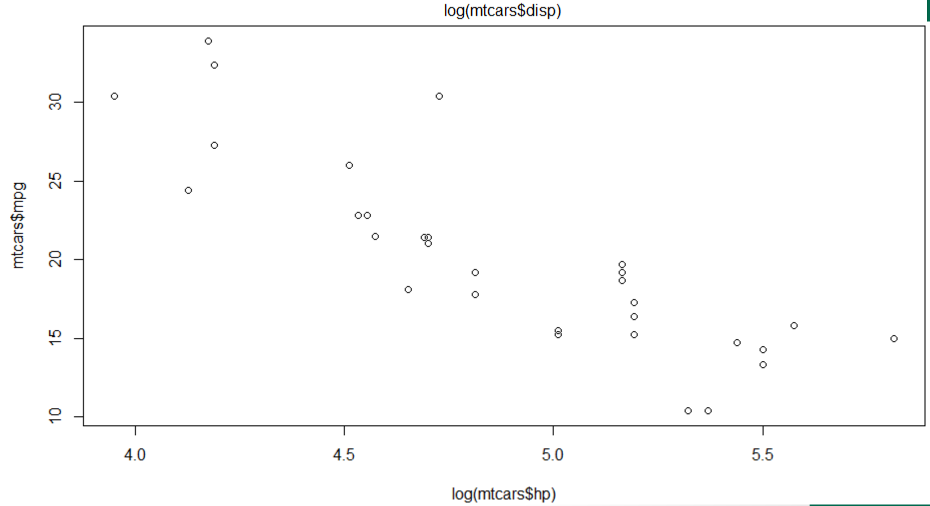
```
> plot(mtcars$disp, mtcars$mpg)
> plot(mtcars$hp, mtcars$mpg)
```

```
> plot(log(mtcars$disp), mtcars$mpg)
> plot(log(mtcars$hp), mtcars$mpg)
```









Example 3: (Multiple Linear regression)

7. Fit the multiple linear regression model with using transformations.

```
> mtcars_model1 = lm(mpg~cyl+log(disp)+log(hp)+drat+wt+qsec+vs+am+gear+carb, data = mtcars)
> summary(mtcars_model1)

Call:
lm(formula = mpg ~ cyl + log(disp) + log(hp) + drat + wt + qsec +
    vs + am + gear + carb, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-3.3225 -1.6278 -0.4725  1.1672  3.7616

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 52.749992  25.678622   2.054   0.0526 .
cyl          0.934117   1.010657   0.924   0.3658
log(disp)   -4.923860   3.852996  -1.278   0.2152
log(hp)     -3.406400   3.003988  -1.134   0.2696
drat         0.169684   1.549901   0.109   0.9139
wt          -0.975286   1.506152  -0.648   0.5243
qsec         0.156231   0.686120   0.228   0.8221
vs          -0.005731   1.904313  -0.003   0.9976
am           0.986507   2.084110   0.473   0.6408
gear         1.706226   1.463070   1.166   0.2566
carb        -0.973755   0.653397  -1.490   0.1510
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.452 on 21 degrees of freedom
Multiple R-squared:  0.8879,	Adjusted R-squared:  0.8345
F-statistic: 16.63 on 10 and 21 DF,  p-value: 8.023e-08
```

Non of the attributes is significant at 10% significance level.

Example 3: (Multiple Linear regression)

8. Check the model accuracy.

Note: we usually focus on adjusted r^2, RMSE (Root Mean Square Error - σ), AIC, and BIC.

```
> list(mtcars_model1 = broom::glance(mtcars_model1))
$`mtcars_model1`
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic      p.value    df logLik   AIC   BIC deviance df.residual
      <dbl>         <dbl> <dbl>     <dbl>        <dbl> <int>  <dbl> <dbl> <dbl>    <dbl>       <int>
1     0.888         0.834  2.45      16.6 0.0000000802    11  -67.4  159.  176.     126.          21
```

Example 3: (Multiple Linear regression)

9. Select the best model. Use stepwise regression.

```
> step(mtcars_model1, direction = "both")
Start:  AIC=65.93
mpg ~ cyl + log(disp) + log(hp) + drat + wt + qsec + vs + am +
    gear + carb
```

```
Call:
lm(formula = mpg ~ log(disp) + gear + carb, data = mtcars)

Coefficients:
(Intercept)    log(disp)         gear         carb
     51.789       -6.592        1.787       -1.227
```

➢ The last step is the best model.

Note: direction = “both” uses stepwise regression which combination of forward and backward elimination.

Example 3: (Multiple Linear regression)

10. Fit the multiple linear regression model for the best model.

```
> selection_model = lm(mpg~log(disp)+gear+carb, data = mtcars)
> summary(selection_model)

Call:
lm(formula = mpg ~ log(disp) + gear + carb, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.0461 -1.3931 -0.5111  1.8053  4.2983 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  51.7887     8.5069   6.088 1.45e-06 ***
log(disp)    -6.5917     1.2208  -5.399 9.31e-06 ***
gear          1.7869     0.9097   1.964  0.05950 .  
carb         -1.2271     0.3872  -3.170  0.00368 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.288 on 28 degrees of freedom
Multiple R-squared:  0.8699,	Adjusted R-squared:  0.8559 
F-statistic:  62.4 on 3 and 28 DF,  p-value: 1.62e-12
```

- All the attributes in this model are significant.

Example 3: (Multiple Linear regression)

11. Check the model accuracy, and compare it with the previous model.

```
> list(mtcars_model1 = broom::glance(mtcars_model1),
+      selection_model = broom::glance(selection_model))
$`mtcars_model1`
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic       p.value    df logLik   AIC   BIC deviance df.residual
      <dbl>         <dbl> <dbl>     <dbl>         <dbl> <int>  <dbl> <dbl> <dbl>    <dbl>       <int>
1     0.888         0.834  2.45      16.6 0.0000000802     11  -67.4  159.  176.     126.          21

$selection_model
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC deviance df.residual
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>    <dbl>       <int>
1     0.870         0.856  2.29      62.4 1.62e-12     4  -69.7  149.  157.     147.          28
```

- The final model is the best model since it has higher adjusted r^2 and lower RMSE (sigma), AIC, and BIC.