# Homework 2

*Quantitative Analysis of Vertebrate Populations*

### 1. Using the `iris` data, make a new dataframe called `iris_petal` with just the petal length and width columns.

```
iris_petal <- iris[ , c("Petal.Length", "Petal.Width")]
str(iris_petal)
```

```
## 'data.frame':    150 obs. of  2 variables:
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
```

Note that depending on the order you add `Petal.Length` and `Petal.Width` will change the order of the columns. This can be useful but it also hints at a challenge or confusion that can arise from using column numbers to select columns (e.g. `iris[ , 3:4]`).

### 2. Now add a new column to `iris_petal` called `Random` filled with normally distributed random numbers.

```
iris_petal$Random <- rnorm(n = ncol(iris_petal), mean = 0, sd = 3)
```

### 3. Display the first 15 rows of the `iris_petal` dataframe.

```
iris_petal[1:15, ]
```

```
##    Petal.Length Petal.Width    Random
## 1           1.4         0.2 -1.466416
## 2           1.4         0.2 -3.823607
## 3           1.3         0.2 -1.466416
## 4           1.5         0.2 -3.823607
## 5           1.4         0.2 -1.466416
## 6           1.7         0.4 -3.823607
## 7           1.4         0.3 -1.466416
## 8           1.5         0.2 -3.823607
## 9           1.4         0.2 -1.466416
## 10          1.5         0.1 -3.823607
## 11          1.5         0.2 -1.466416
## 12          1.6         0.2 -3.823607
## 13          1.4         0.1 -1.466416
## 14          1.1         0.1 -3.823607
## 15          1.2         0.2 -1.466416
```

### 4. Print a summary of the `iris_petal` dataframe.

```
summary(iris_petal)
```

```
##   Petal.Length    Petal.Width        Random
##  Min.   :1.000   Min.   :0.100   Min.   :-3.824
##  1st Qu.:1.600   1st Qu.:0.300   1st Qu.:-3.824
##  Median :4.350   Median :1.300   Median :-2.645
##  Mean   :3.758   Mean   :1.199   Mean   :-2.645
##  3rd Qu.:5.100   3rd Qu.:1.800   3rd Qu.:-1.466
##  Max.   :6.900   Max.   :2.500   Max.   :-1.466
```

### 5. Select just the values of `Petal.Length` greater than the mean petal length. Then print the new summary.

```r
pl_mean <- mean(iris_petal$Petal.Length)

pl_big <- iris_petal[which(iris_petal$Petal.Length > pl_mean), ]

summary(pl_big)
```

```
##   Petal.Length    Petal.Width        Random
##  Min.   :3.800   Min.   :1.000   Min.   :-3.824
##  1st Qu.:4.500   1st Qu.:1.400   1st Qu.:-3.824
##  Median :4.900   Median :1.700   Median :-1.466
##  Mean   :5.018   Mean   :1.723   Mean   :-2.632
##  3rd Qu.:5.600   3rd Qu.:2.000   3rd Qu.:-1.466
##  Max.   :6.900   Max.   :2.500   Max.   :-1.466
```

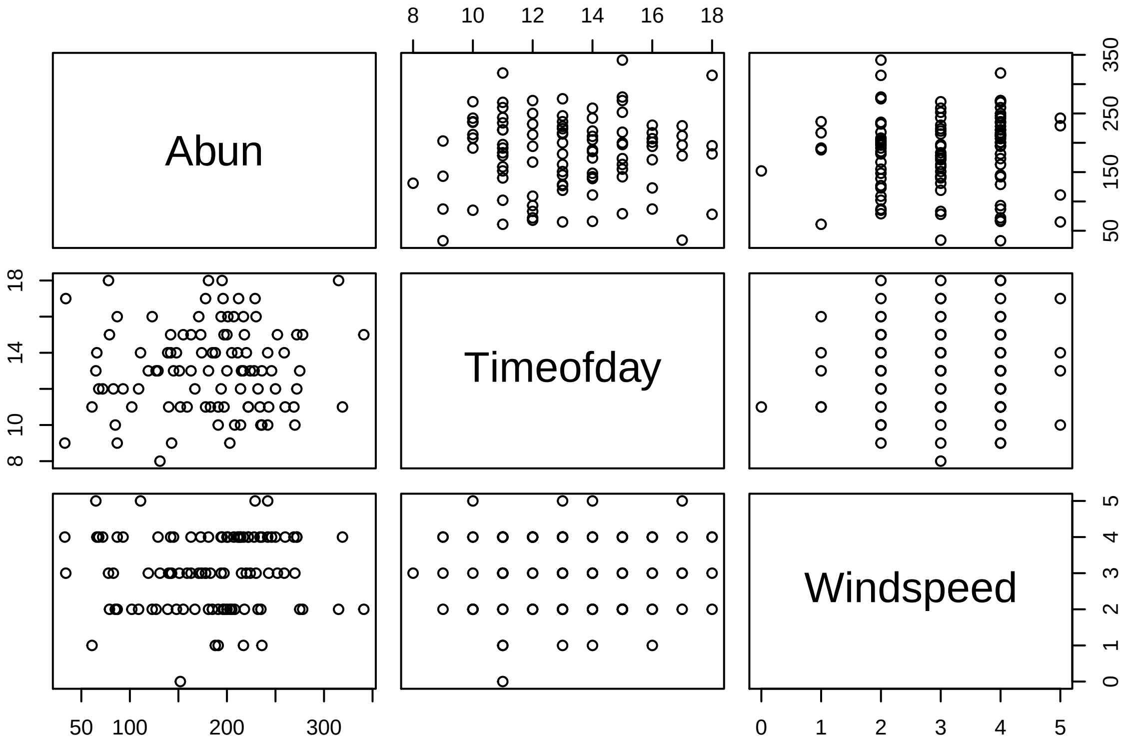
### 6. Read a file into R and save the dataframe as an object. Be sure to use `stringsAsFactors = FALSE`.

```r
seals <- read.table("Data/Seals.txt", header = TRUE, stringsAsFactors = FALSE)
str(seals)
```

```
## 'data.frame':    98 obs. of  10 variables:
##  $ Abun     : int  232 235 191 207 236 155 242 195 196 236 ...
##  $ Site     : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Month    : int  4 5 6 6 6 7 7 7 8 8 ...
##  $ Year     : int  2003 2003 2003 2003 2003 2003 2003 2003 2003 2003 ...
##  $ Week     : int  16 22 24 25 26 27 29 30 32 33 ...
##  $ WeekTime : num  2003 2003 2003 2003 2003 ...
##  $ Timeofday: int  12 10 10 16 10 15 14 18 17 13 ...
##  $ Winddir  : int  4 6 6 7 8 8 8 7 8 6 ...
##  $ Windspeed: int  2 2 2 4 4 2 4 4 2 1 ...
##  $ Weather  : int  3 3 3 3 2 3 2 1 3 3 ...
```

### 7. Make a scatterplot matrix of three variables from the new dataframe.

```r
pairs(seals[ , c("Abun", "Timeofday", "Windspeed")])
```



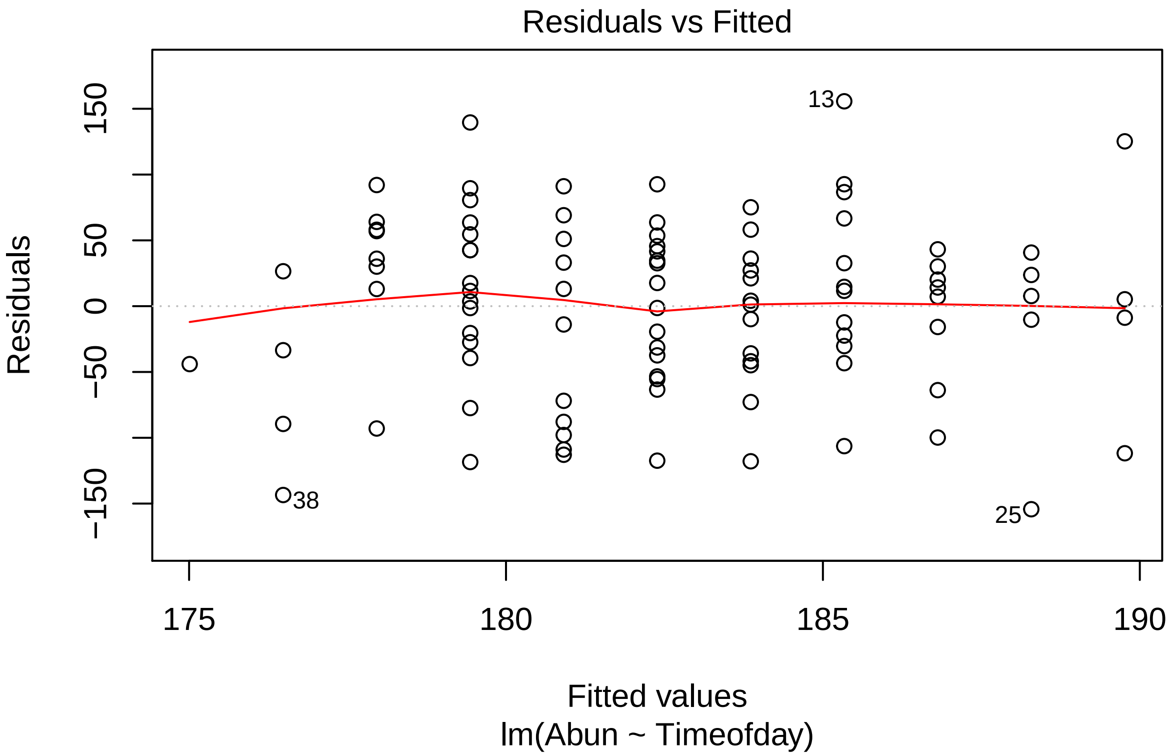

### 8. Run a linear regression and print the summary table.

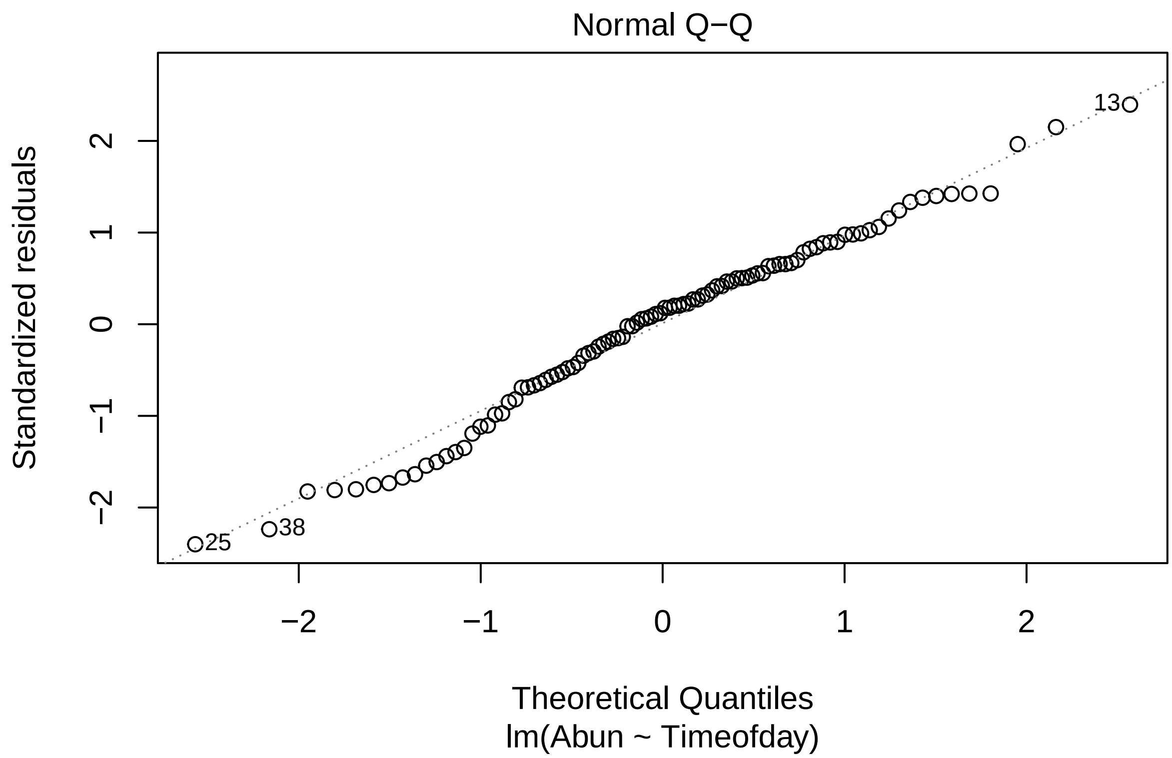
```
lm_seals <- lm(Abun ~ Timeofday, data = seals)
summary(lm_seals)
```

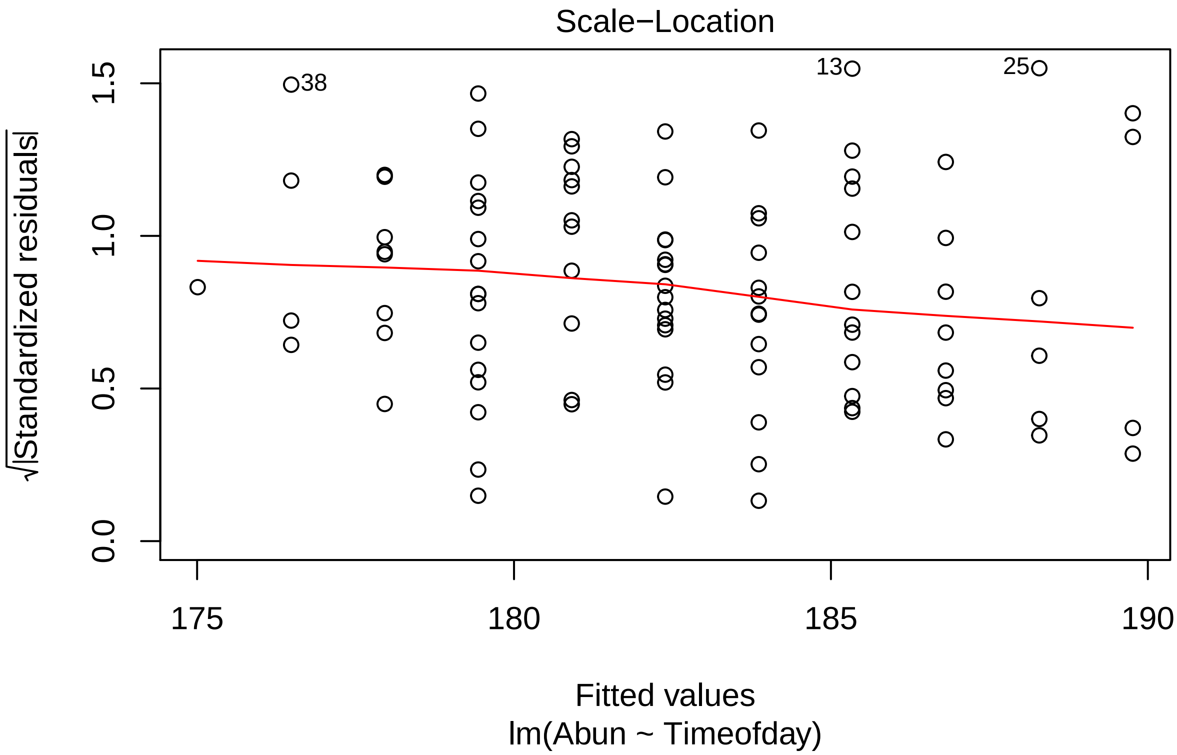
```
##
## Call:
## lm(formula = Abun ~ Timeofday, data = seals)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -154.287  -41.255    9.639   42.565  155.663
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  163.206     37.367   4.368 3.17e-05 ***
## Timeofday      1.475      2.800   0.527      0.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65.51 on 96 degrees of freedom
## Multiple R-squared:  0.002883,   Adjusted R-squared:  -0.007504
## F-statistic: 0.2776 on 1 and 96 DF,  p-value: 0.5995
```

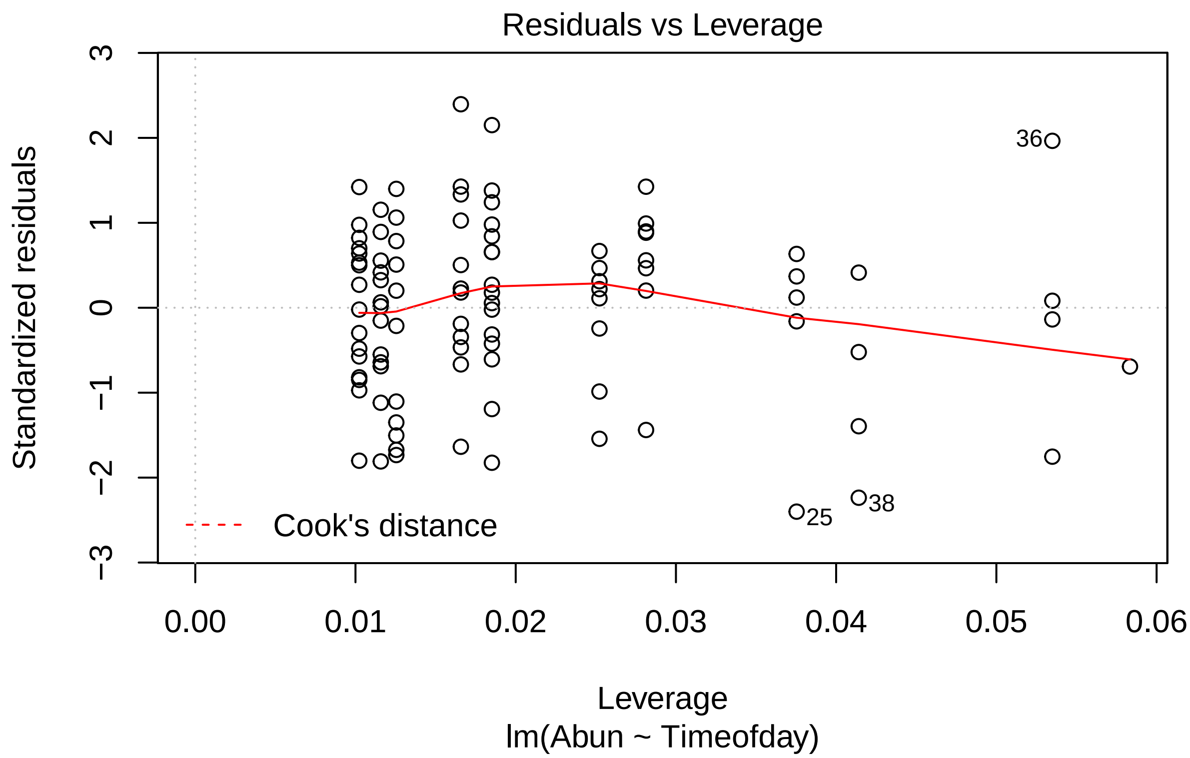
**9. Are the linear model assumptions met and how do you know (Hint: make diagnostic plots using `plot(Model_Name)` and interpret the output. You won't lose points if you correctly explain the plots even if I disagree with your judgement on whether things are normal or homogenous enough.**

```r
plot(lm_seals)
```




Normal Q−Q
Standardized residuals
−2
−1
0
1
2
25
38
13
Theoretical Quantiles
lm(Abun ~ Timeofday)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
38
13
25
175
180
185
190
Fitted values
lm(Abun ~ Timeofday)




There is no evidence of strong patterning in the residuals plot or in the standardized residuals plot. The QQ-plot shows evidence of normality and there are no clear outliers identified by Cook's distance. As such, there seems to be a good match between the data and the model (this is probably not true since there is likely some problems of independence in the experimental design that needs to be accounted for but it's not apparent from these plots).