STAT 210
Applied Statistics and Data Analysis
Second Exam

November 26, 2022

This exam is open notes and open book but not open internet. You are not allowed to surf the internet or look for answers to the questions

You are reminded to adhere to the academic integrity code established at KAUST.

Show complete solutions to get full credit. Label your graphs appropriately

Please, do not submit zip files and identify the files you submit with your surname

---

For this exam, we will use the data in the file `dragons.txt`. Read the data onto a data frame. There are ten variables in the set:

- `height`, the height in m,
- `length`, the length from head to tail in m,
- `weight`, the weight in tons,
- `wing.ln`, the average length for the wings in m,
- `leg.ht`, the average length for the legs in m,
- `wing.span`, the distance between the tips of the outstretched wings,
- `sp`, the species with two values, `black` and `gold`,
- `age`, in years
- `strength`, the strength index for the dragon, and
- `firepwr`, a combined measure of the caloric power, size and duration of the fire breath.

Question 1 (30 points)

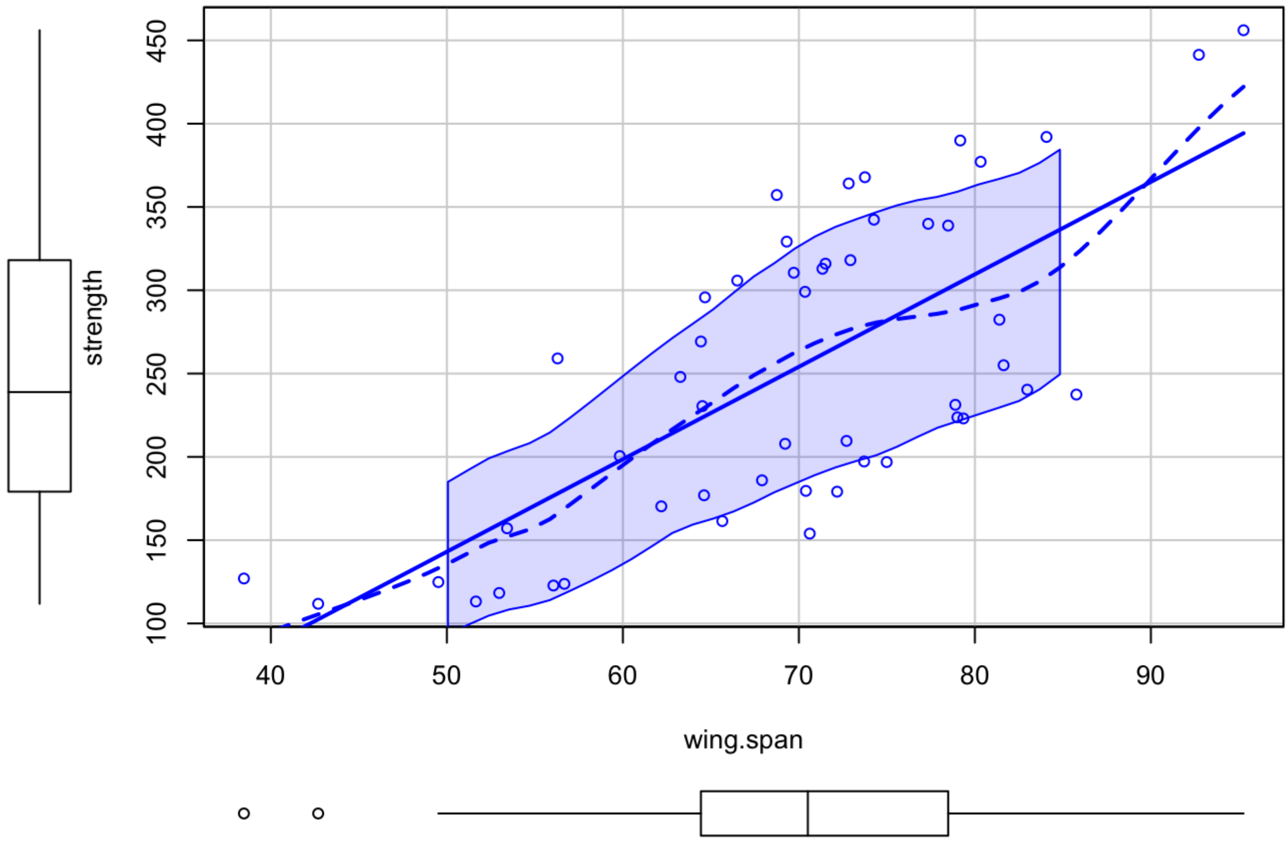
In this question, you have to explore the relationship between the variables `strength` and `wing.span`.

i. (2.5 pts) Graph a scatterplot of `strength` as a function of `wing.span`. Add the regression line for these variables and comment.

```
dragons = read.table('dragons.txt')
library(car)
```

```
## Loading required package: carData
```

```
scatterplot(strength ~ wing.span ,data= dragons)
```



This plot produces a local smoother curve (broken line) that can be compared with the regression line. Important discrepancies indicate that the linear regression model is not adequate.

ii. (5 pts) Fit a simple regression model for these variables and print the summary table. What is the R^2 for this model? Write down the equation for the model and give an interpretation of the parameters. Predict the strength of a dragon with a wingspan of 60 m. and include a prediction interval.

```
model1 = lm(strength ~ wing.span ,data = dragons)
summary(model1)
```

```
##
## Call:
## lm(formula = strength ~ wing.span, data = dragons)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -104.18  -58.77    4.84   59.41  110.09
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -134.445     56.695   -2.37    0.022 *
## wing.span      5.550      0.804    6.90    1e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65.7 on 48 degrees of freedom
## Multiple R-squared:  0.498,  Adjusted R-squared:  0.487
## F-statistic: 47.6 on 1 and 48 DF,  p-value: 1.05e-08
```

```
a = data.frame(wing.span = 60)
predict.lm(model1,a,interval = 'p')
```

```
##       fit   lwr    upr
## 1 198.57 64.18 332.96
```

The $R^2 = 0.4979$. Equation :

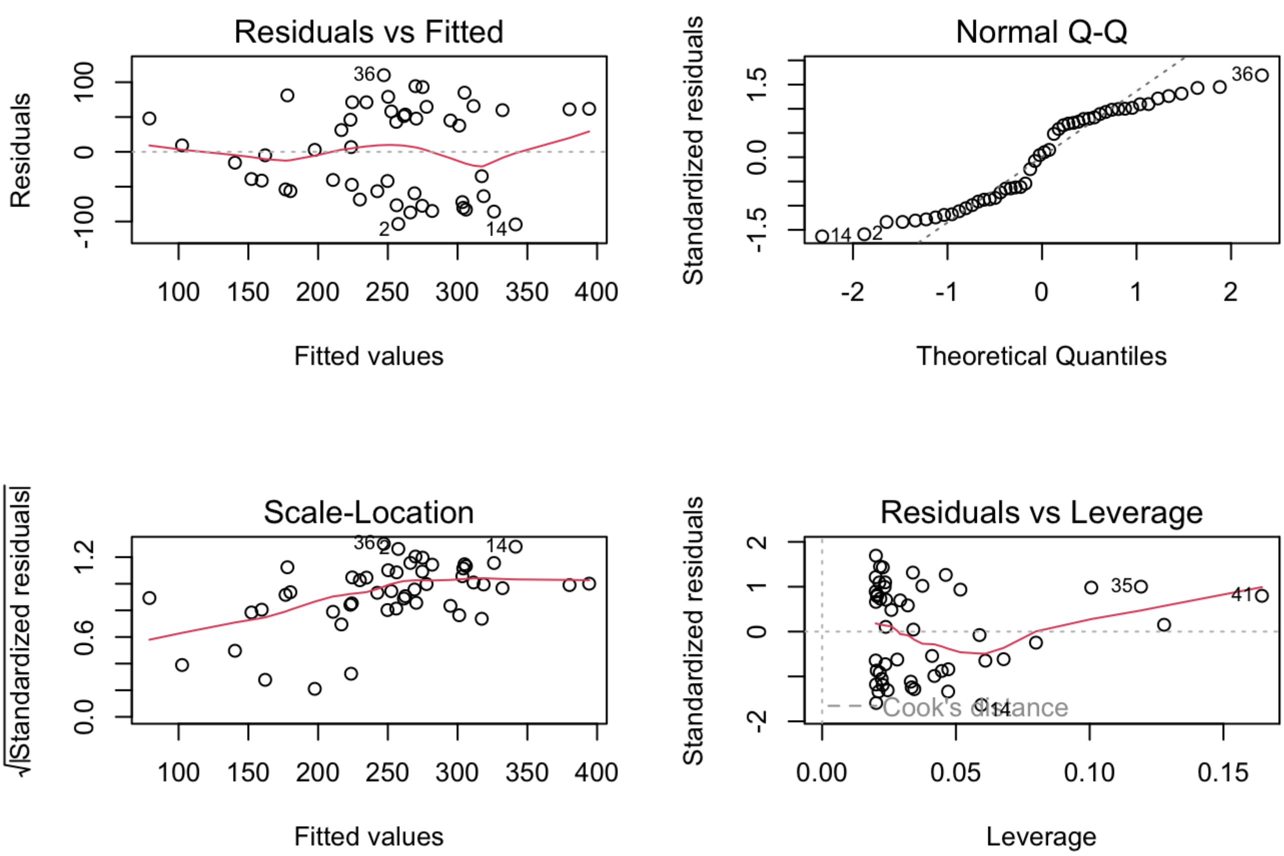
$$strength = -134.4447 + 5.5502 * wing.span$$

The Intercept -134.4447 is from Intercept estimated. The slope 5.5502 is from wing.span estimated, it means one unit increase in wing.span will lead to 5.55 unit increase in strength.

We can predict the strength of a dragon with a wingspan of 60 m is 198.5699. and include a prediction interval from 64.18 to 332.96.

iii. (5 pts) Use graphical methods and tests to check the assumptions on which the model is based. What are your conclusions?

```
par(mfrow = c(2,2))
plot(model1)
```




```
par(mfrow = c(1,1))
shapiro.test(rstandard(model1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(model1)
## W = 0.911, p-value = 0.0011
```

```
ncvTest(model1)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 2.1345, Df = 1, p = 0.144
```

Firstly , we plot diagnostics plots. All plots seem not good enough. The quantile plot is not match many points. We could check normality with Shapiro-Wilk test. In residuals against fitted values plot, the red line is not horizontal. The residuals vs leverage plot shows an increasing pattern.

In Shapiro-Wilk test, p is very small, so we can reject the hypothesis of normality.

In Non-constant Variance Score Test , p is larger than 0.05,so we cannot reject the hypothesis of homogeneous variance. Overall, The model is not adequate.

iv. (10 pts) There are two species of dragons in the file, black and gold, and this characteristic is available in the categorical variable `sp` . We want to add this variable to the regression model. If the variable was not read as a `factor` , transform it before you continue. Fit a model that includes the previous variable, the new variable, and the interaction between the two. Using a critical value for α of 0.05 and starting with the complete model, select a minimal adequate model.

```
dragons$sp <- factor(dragons$sp)
model2 = lm(strength ~ wing.span + sp ,data= dragons)
summary(model2)
```

```
##
## Call:
## lm(formula = strength ~ wing.span + sp, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.76 -17.11   0.14  13.30  64.19
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -168.804     23.066   -7.32  2.7e-09 ***
## wing.span      5.194      0.327   15.90  < 2e-16 ***
## spgold       118.322      7.549   15.67  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.6 on 47 degrees of freedom
## Multiple R-squared:  0.919,  Adjusted R-squared:  0.916
## F-statistic:  268 on 2 and 47 DF,  p-value: <2e-16
```

```
lm1 <- lm(strength ~ ., data = dragons)
summary(lm1)
```

```
##
## Call:
## lm(formula = strength ~ ., data = dragons)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.49 -13.05   0.06  17.77  48.55
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  125.429    220.050    0.57    0.572
## height         2.552      1.206    2.12    0.041 *
## length       -11.984      6.990   -1.71    0.094 .
## weight         4.432      3.112    1.42    0.162
## wing.l         0.959      1.662    0.58    0.567
## leg.ht         3.909      2.996    1.30    0.200
## wing.span      5.567      0.621    8.97    4e-11 ***
## spgold       114.236      7.495   15.24   <2e-16 ***
## age           -0.182      0.136   -1.33    0.190
## firepwr       -0.542      0.408   -1.33    0.191
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.6 on 40 degrees of freedom
## Multiple R-squared:  0.936,  Adjusted R-squared:  0.922
## F-statistic: 65.4 on 9 and 40 DF,  p-value: <2e-16
```

We choose a critical value of 0.05 for α. We remove wing.l which has the largest p-value.

```
lm2 <- update(lm1, ~. - wing.l)
summary(lm2)
```

```
##
## Call:
## lm(formula = strength ~ height + length + weight + leg.ht + wing.span +
##     sp + age + firepwr, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -51.62 -12.61   0.47  17.76  47.26
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  134.702    217.670    0.62    0.539
## height         2.588      1.195    2.17    0.036 *
## length       -12.021      6.933   -1.73    0.090 .
## weight         4.594      3.074    1.49    0.143
## leg.ht         3.985      2.969    1.34    0.187
## wing.span      5.789      0.482   12.02  5.2e-15 ***
## spgold       114.375      7.430   15.39  < 2e-16 ***
## age           -0.187      0.135   -1.39    0.173
## firepwr       -0.543      0.404   -1.34    0.187
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.4 on 41 degrees of freedom
## Multiple R-squared:  0.936,  Adjusted R-squared:  0.923
## F-statistic: 74.8 on 8 and 41 DF,  p-value: <2e-16
```

We now remove leg.ht

```
lm3 <- update(lm2, ~. - leg.ht)
summary(lm3)
```

```
##
## Call:
## lm(formula = strength ~ height + length + weight + wing.span +
##     sp + age + firepwr, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -55.21 -12.72  -2.85  15.39  47.79
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   79.021    215.709    0.37    0.716
## height         2.172      1.164    1.86    0.069 .
## length        -9.472      6.731   -1.41    0.167
## weight         3.377      2.965    1.14    0.261
## wing.span      5.918      0.477   12.41  1.2e-15 ***
## spgold       114.128      7.499   15.22  < 2e-16 ***
## age           -0.145      0.132   -1.10    0.279
## firepwr       -0.456      0.403   -1.13    0.264
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.7 on 42 degrees of freedom
## Multiple R-squared:  0.933,  Adjusted R-squared:  0.922
## F-statistic: 83.6 on 7 and 42 DF,  p-value: <2e-16
```

```
lm4 <- update(lm3, ~. - age)
summary(lm4)
```

```
##
## Call:
## lm(formula = strength ~ height + length + weight + wing.span +
##     sp + firepwr, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -58.43 -14.93  -1.33  16.17  51.21
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.5735   203.6209    0.00     1.00
## height        1.2916     0.8461    1.53     0.13
## length       -7.3971     6.4747   -1.14     0.26
## weight        2.5285     2.8686    0.88     0.38
## wing.span     5.8468     0.4734   12.35  9.8e-16 ***
## spgold      115.2949     7.4402   15.50  < 2e-16 ***
## firepwr      -0.0512     0.1621   -0.32     0.75
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.7 on 43 degrees of freedom
## Multiple R-squared:  0.931,  Adjusted R-squared:  0.921
## F-statistic: 96.8 on 6 and 43 DF,  p-value: <2e-16
```

```
lm5 <- update(lm4, ~. - firepwr)
summary(lm5)
```

```
##
## Call:
## lm(formula = strength ~ height + length + weight + wing.span +
##     sp, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -58.00 -14.79  -0.21  17.63  50.62
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   11.276    198.071    0.06     0.95
## height         1.222      0.808    1.51     0.14
## length        -7.777      6.296   -1.24     0.22
## weight         2.702      2.787    0.97     0.34
## wing.span      5.881      0.456   12.90   <2e-16 ***
## spgold       115.055      7.325   15.71   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.5 on 44 degrees of freedom
## Multiple R-squared:  0.931,  Adjusted R-squared:  0.923
## F-statistic:  119 on 5 and 44 DF,  p-value: <2e-16
```

```
lm6 <- update(lm5, ~. - weight )
summary(lm6)
```

```
##
## Call:
## lm(formula = strength ~ height + length + wing.span + sp, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.48 -14.33  -0.84  17.33  47.38
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -171.811     59.653   -2.88   0.0061 **
## height         1.311      0.802    1.63   0.1093
## length        -1.748      0.982   -1.78   0.0817 .
## wing.span      5.822      0.451   12.89   <2e-16 ***
## spgold       115.600      7.299   15.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.5 on 45 degrees of freedom
## Multiple R-squared:  0.929,  Adjusted R-squared:  0.923
## F-statistic:  148 on 4 and 45 DF,  p-value: <2e-16
```

```
lm7 <- update(lm6, ~. - height)
summary(lm7)
```

```
##
## Call:
## lm(formula = strength ~ length + wing.span + sp, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -60.09 -16.44  -1.54  16.91  55.22
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -101.272     41.903   -2.42    0.020 *
## length        -1.898      0.995   -1.91    0.063 .
## wing.span      5.827      0.460   12.68   <2e-16 ***
## spgold       116.451      7.411   15.71   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.9 on 46 degrees of freedom
## Multiple R-squared:  0.925,  Adjusted R-squared:  0.92
## F-statistic:  190 on 3 and 46 DF,  p-value: <2e-16
```

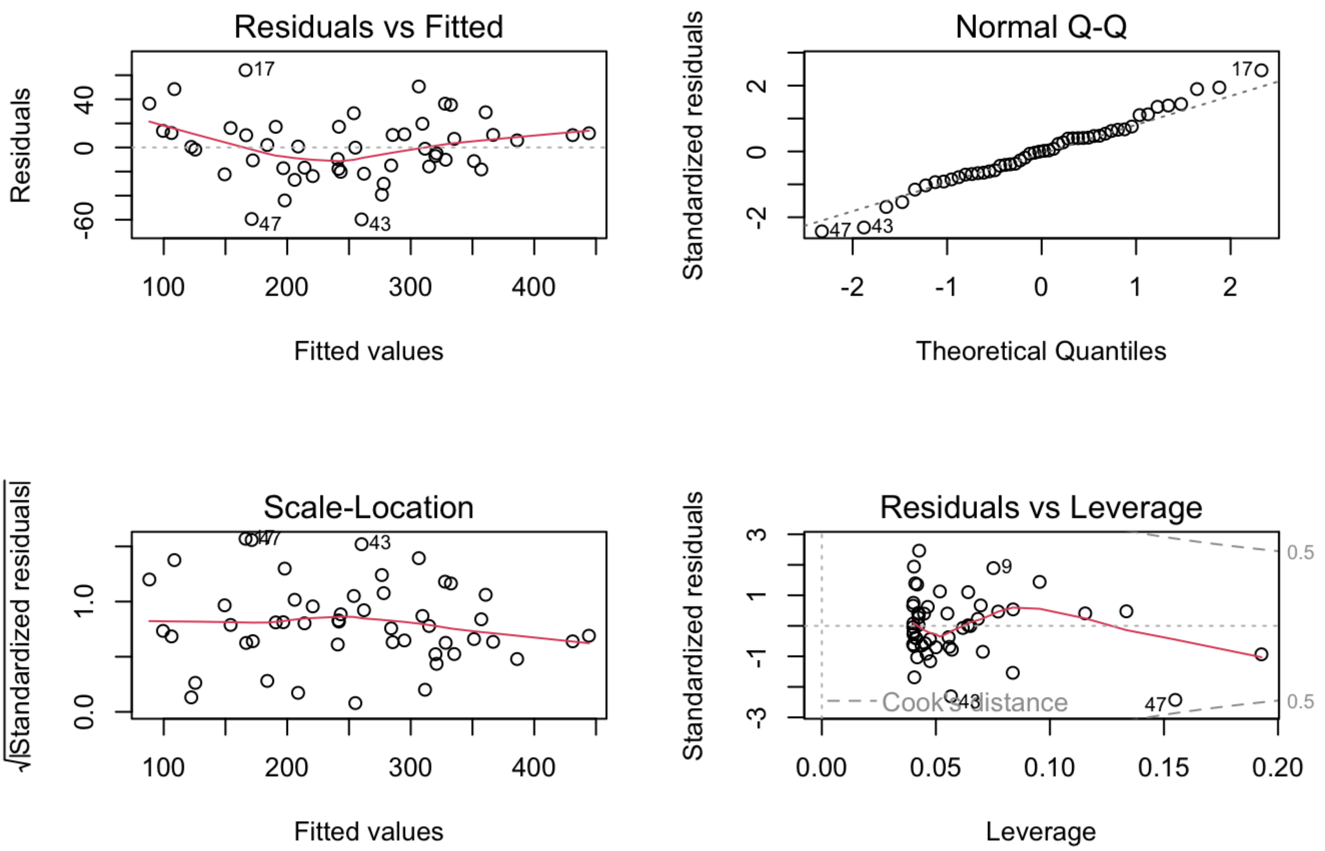
```
lm8 <- update(lm7, ~. - length)
summary(lm8)
```

```
##
## Call:
## lm(formula = strength ~ wing.span + sp, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.76 -17.11   0.14  13.30  64.19
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -168.804     23.066   -7.32  2.7e-09 ***
## wing.span      5.194      0.327   15.90  < 2e-16 ***
## spgold       118.322      7.549   15.67  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.6 on 47 degrees of freedom
## Multiple R-squared:  0.919,  Adjusted R-squared:  0.916
## F-statistic:  268 on 2 and 47 DF,  p-value: <2e-16
```

Finally we get a minimal adequate model. It is same with model2, we have wing.span and sp two variables.

v. (7.5 pts) Check the assumptions for the final model. Compare the adjusted R^2 with the previous model. Write down the equation for the regression model and predict the value of the strength for black and gold dragons of weight span 60 m, including prediction intervals. Compare with the previous prediction and comment.

```
par(mfrow = c(2,2))
plot(model2)
```




```
par(mfrow = c(1,1))
shapiro.test(rstandard(model2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(model2)
## W = 0.987, p-value = 0.85
```

```
ncvTest(model2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 1.987, Df = 1, p = 0.159
```

Firstly, we plot diagnostics plots. All plots seems improved. The quantile plot is match more points. We could check normality with Shapiro-Wilk test. In residuals against fitted values plot, the red line is closer to zero. The third plot also looks horizontal.

In Shapiro-Wilk test, p is larger than 0.05, so we cannot reject the hypothesis of normality.

In Non-constant Variance Score Test , p is larger than 0.05,so we cannot reject the hypothesis of homogeneous variance. Thus, the model is better.

```
a = data.frame(wing.span=60,sp='gold')
predict.lm(model2,a,interval = 'p')
```

```
##      fit    lwr    upr
## 1 261.13 206.08 316.18
```

```
a = data.frame(wing.span=60,sp='black')
predict.lm(model2,a,interval = 'p')
```

```
##      fit    lwr    upr
## 1 142.81 87.881 197.73
```

The adjusted R^2 is much larger than the previous model. So that this model is better.

Equation :

spgold =1 if sp is gold . spgold =0 if sp is black.

$$strength = -168.8036 + 118.3223 * spgold + 5.1935 * wing.span$$

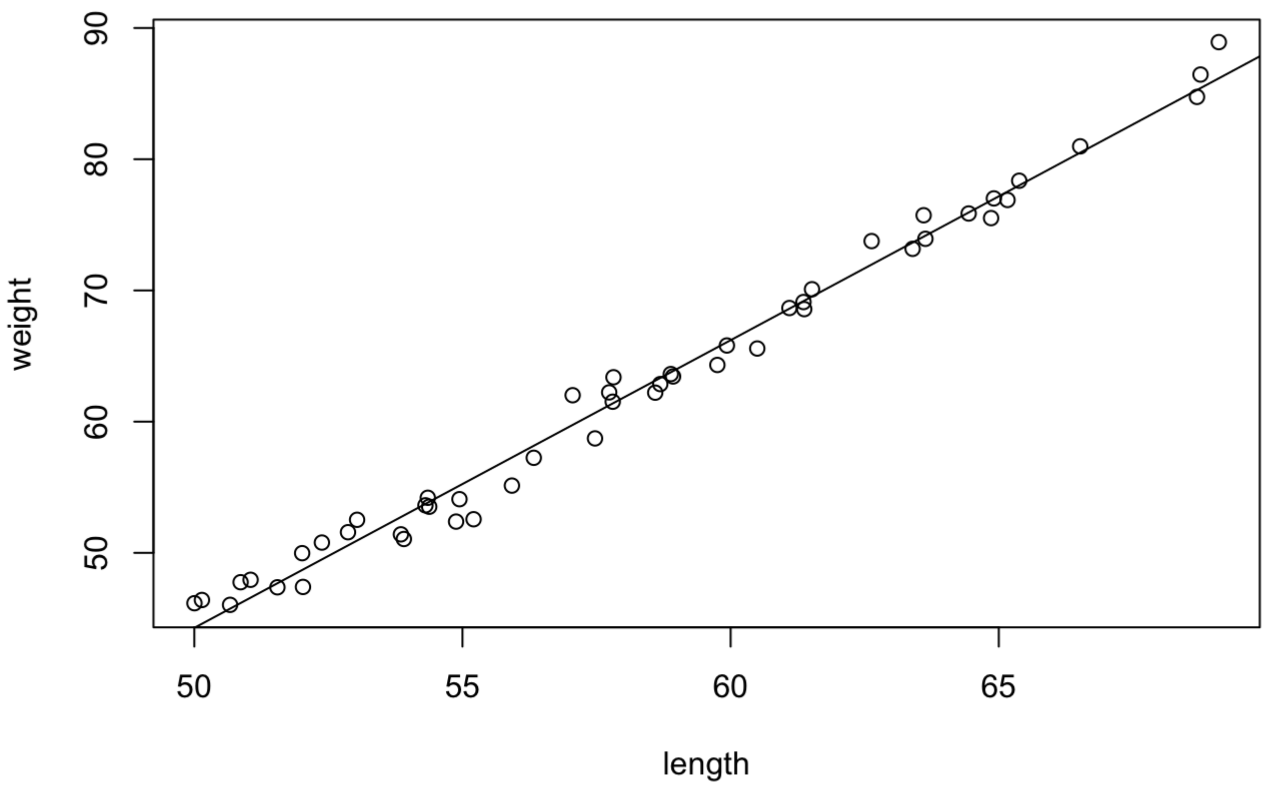
We can predict the strength of a dragon with a wingspan of 60 m, black dragons is 142.8068, and include a prediction interval from 87.88 to 197.73. We can predict the strength of a dragon with a wingspan of 60 m, gold dragons is 261.1291, and include a prediction interval from 206.08 to 316.17. Compare with the previous prediction, the interval are smaller.

Question 2 (30 points)

In this question, we want to explore the relation between the weight (`weight`) and the length (`length`) of dragons.

i. (15 pts) Start by plotting a graph of `weight` as a function of `length` . Fit a simple regression model and add a regression line to the plot. What is the R^2 for this model? Write down an equation for the model and give an interpretation of the parameters.

```
plot(weight ~ length, data = dragons)
model3 = lm(weight ~ length, data = dragons)
abline(model3)
```




```
summary(model3)
```

```
##
## Call:
## lm(formula = weight ~ length, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.142 -0.852 -0.099  1.019  2.747
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -65.3680     2.0841   -31.4   <2e-16 ***
## length        2.1930     0.0356    61.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.34 on 48 degrees of freedom
## Multiple R-squared:  0.988,  Adjusted R-squared:  0.987
## F-statistic: 3.79e+03 on 1 and 48 DF,  p-value: <2e-16
```

The R^2 for this model is 0.9875.

Equation : $weight = -65.368 + 2.19 * length$

The slope 2.19 is the rate of increase of the weight per meter increase in length. The intercept shows when length equal to zero, weight is -65.368(which means nothing).

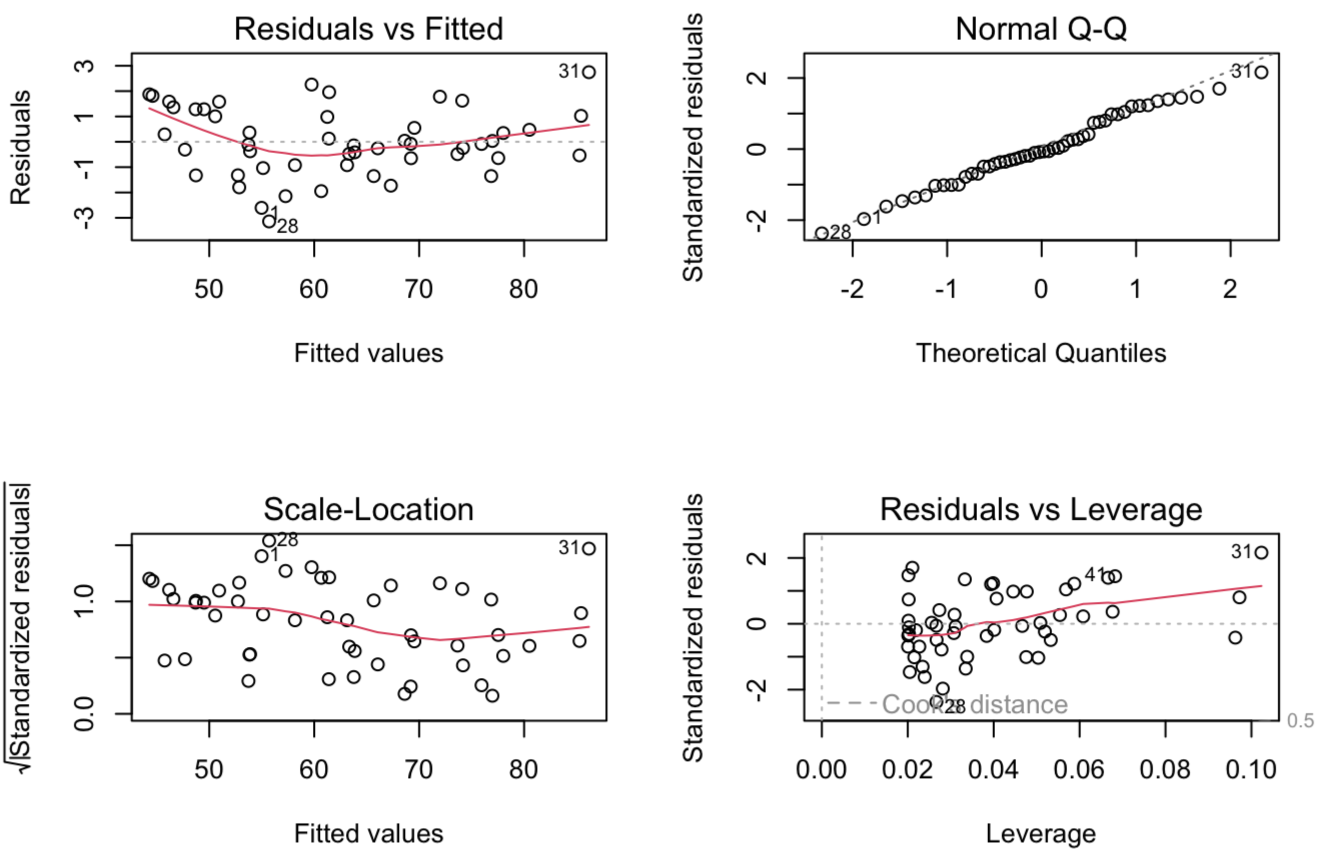
Give a prediction of the weight of a dragon with a length of 58 m, including a confidence interval. State explicitly the assumptions on which this model is based. Check whether these assumptions are satisfied. Use the function `residualPlots` in the `car` package and interpret the graphs and results of the hypotheses test. What do these results suggest?

```
a = data.frame(length= 58)
predict.lm(model3,a,interval = 'c')
```

```
##      fit    lwr   upr
## 1 61.828 61.445 62.21
```

We can predict the weight of a dragon with a length of 58 m is 61.83, including a confidence interval from 61.445 to 62.21.

```
par(mfrow = c(2,2))
plot(model3)
```



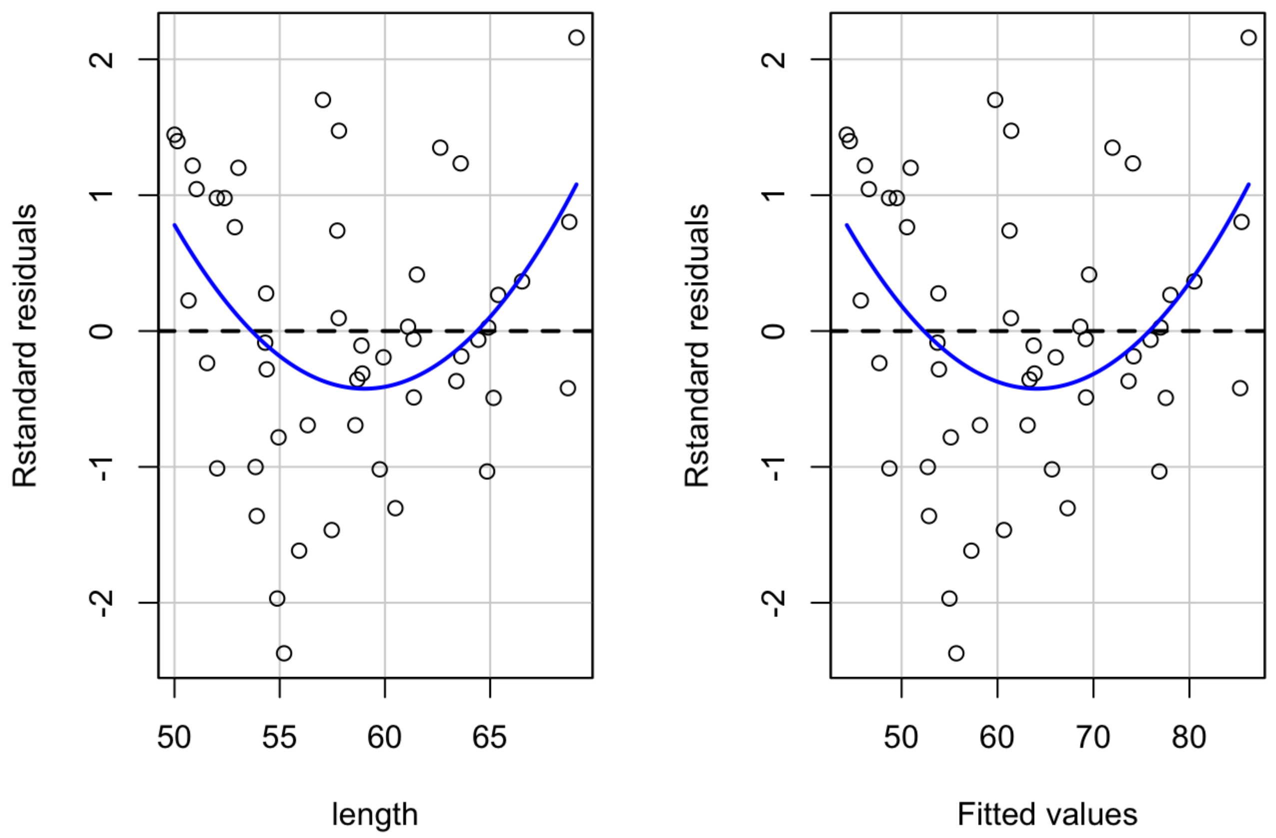

```
par(mfrow = c(1,1))
shapiro.test(rstandard(model3))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(model3)
## W = 0.986, p-value = 0.83
```

```
ncvTest(model3)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.91717, Df = 1, p = 0.338
```

```
residualPlots(model3, type = 'rstandard')
```




```
##            Test stat Pr(>|Test stat|)
## length         3.09           0.0033 **
## Tukey test     3.09           0.0020 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Firstly, we plot diagnostics plots. The quantile plot seems good. We could check normality with Shapiro-Wilk test. The residuals against fitted values plot is not horizontal. The third plot seems horizontal. The residuals vs leverage plot shows an increasing pattern.

In Shapiro-Wilk test, p is large, so we cannot reject the hypothesis of normality.

In Non-constant Variance Score Test, p is large,so we cannot reject the hypothesis of homogeneous variance.

`residualPlots` function plots residuals against length and also against fitted values, and adds a quadratic term. It also tests the significance of the added term and lists the p-values.In this case, the quadratic term for length has a small p-value. It suggests that we can add a quadratic term in model.

ii. (15 pts) Fit a new model, including the term(s) suggested by the tests in (i), if any. Look at the summary table. What is the adjusted R^2 for this model? Check whether the assumptions for linear regression are satisfied for the new model. Write an equation for the model. Give a prediction of the weight of a dragon with a length of 58 m, including a confidence interval, and compare it with the result in part (i).

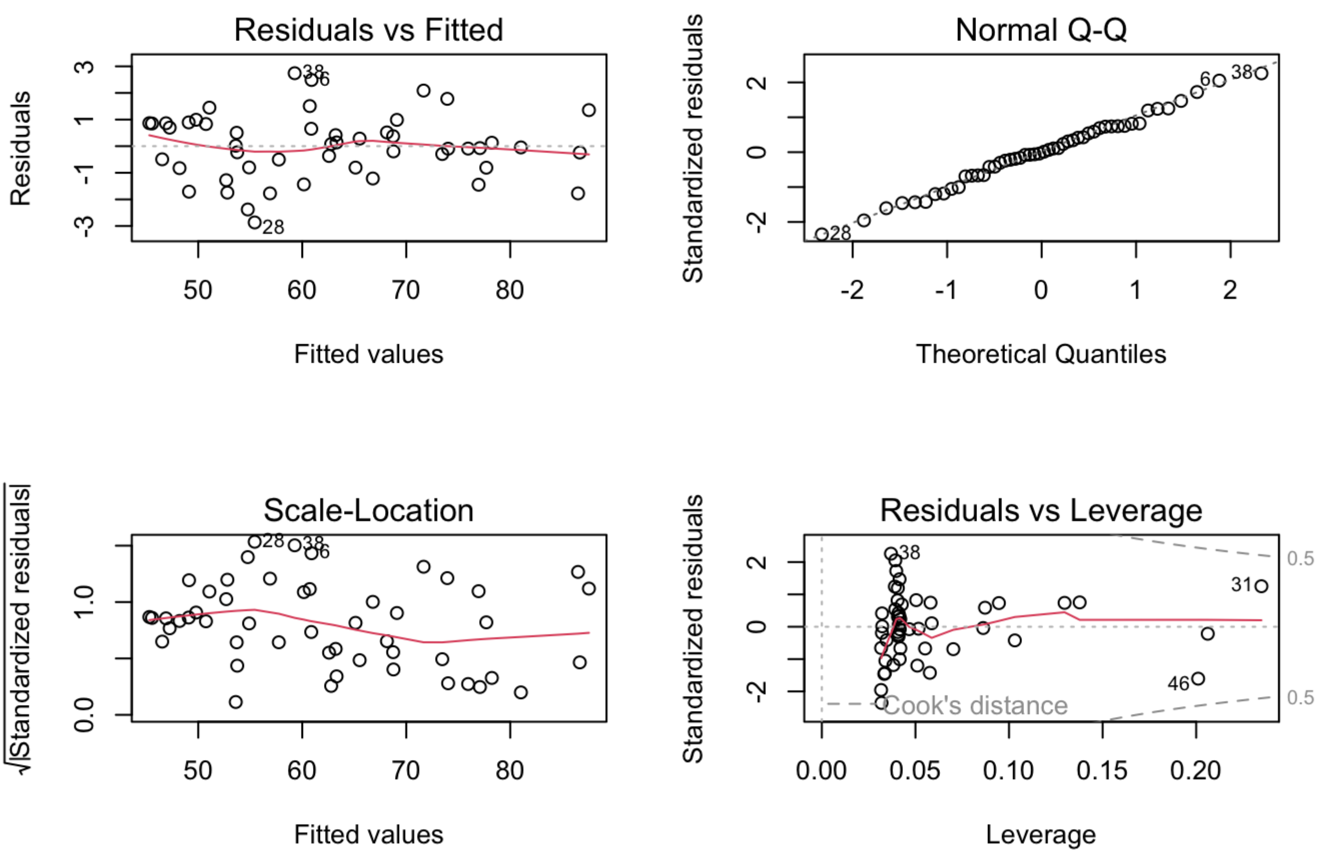
```
model4 <- update(model3, ~. + I(length^2))
summary(model4)
```

```
##
## Call:
## lm(formula = weight ~ length + I(length^2), data = dragons)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.8657 -0.8039 -0.0155  0.8472  2.7416
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.99611   21.54273    0.05   0.9633
## length      -0.07437    0.73383   -0.10   0.9197
## I(length^2)  0.01921    0.00621    3.09   0.0033 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.24 on 47 degrees of freedom
## Multiple R-squared:  0.99,   Adjusted R-squared:  0.989
## F-statistic: 2.24e+03 on 2 and 47 DF,  p-value: <2e-16
```

adjusted R^2 : 0.9892

Equation : $weight = 0.99611 - 0.074 * length + 0.01921 * length^2$

```
par(mfrow = c(2,2))
plot(model4)
```




```
par(mfrow = c(1,1))
shapiro.test(rstandard(model4))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(model4)
## W = 0.991, p-value = 0.97
```

```
ncvTest(model4)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.41007, Df = 1, p = 0.522
```

Firstly, we plot diagnostics plots. We can see plots have improved. The quantile plot seems good. We could check normality with Shapiro-Wilk test. The residuals against fitted values plot is relatively horizontal. The third plot seems horizontal. The residuals vs leverage plot doesn't have increasing pattern now.

In Shapiro-Wilk test, p is large, so we cannot reject the hypothesis of normality.

In Non-constant Variance Score Test , p is large,so we cannot reject the hypothesis of homogeneous variance.

```
a = data.frame(length= 58)
predict(model4,a,interval = 'c')
```

```
##      fit    lwr    upr
## 1 61.291 60.796 61.787
```

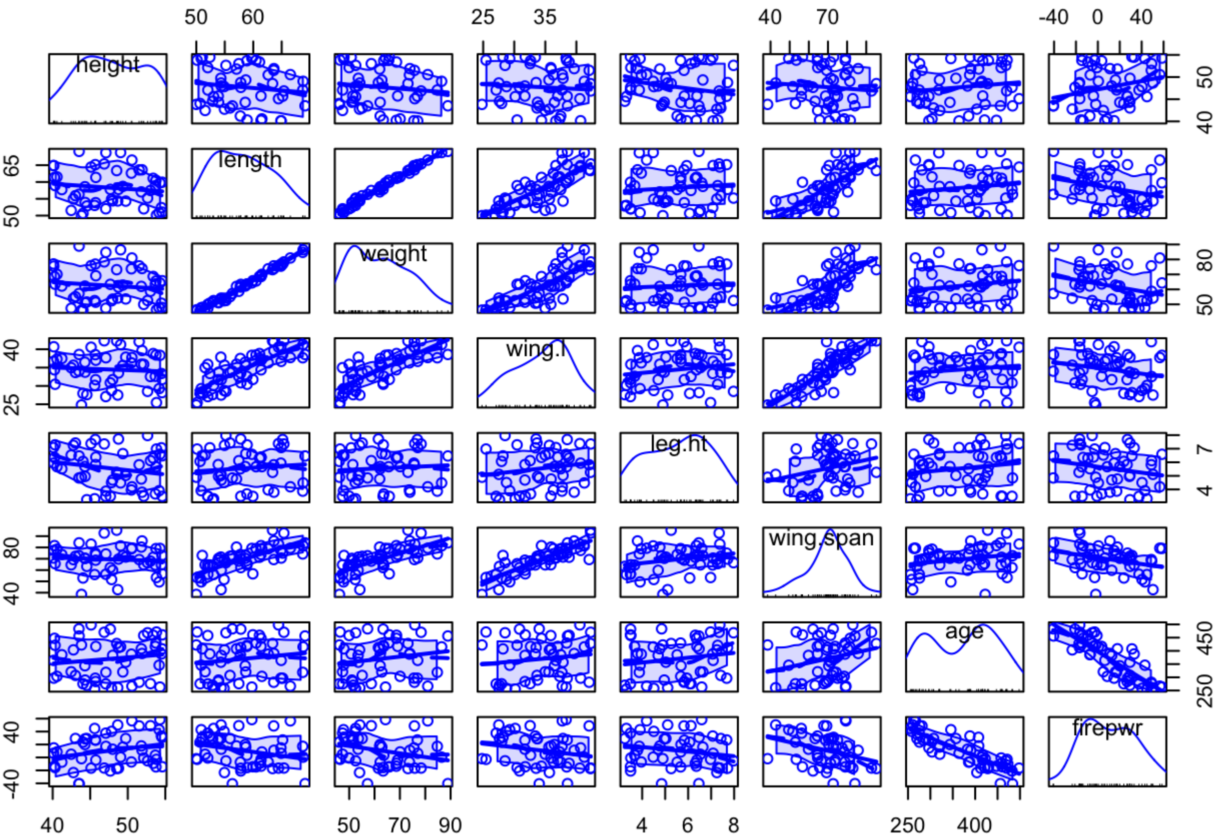
We can predict the weight of a dragon with a length of 58 m is 61.29, including a confidence interval from 60.7955 to 61.7871. It is similar with the result in part (i).(a little bit smaller)

Question 3 (40 points)

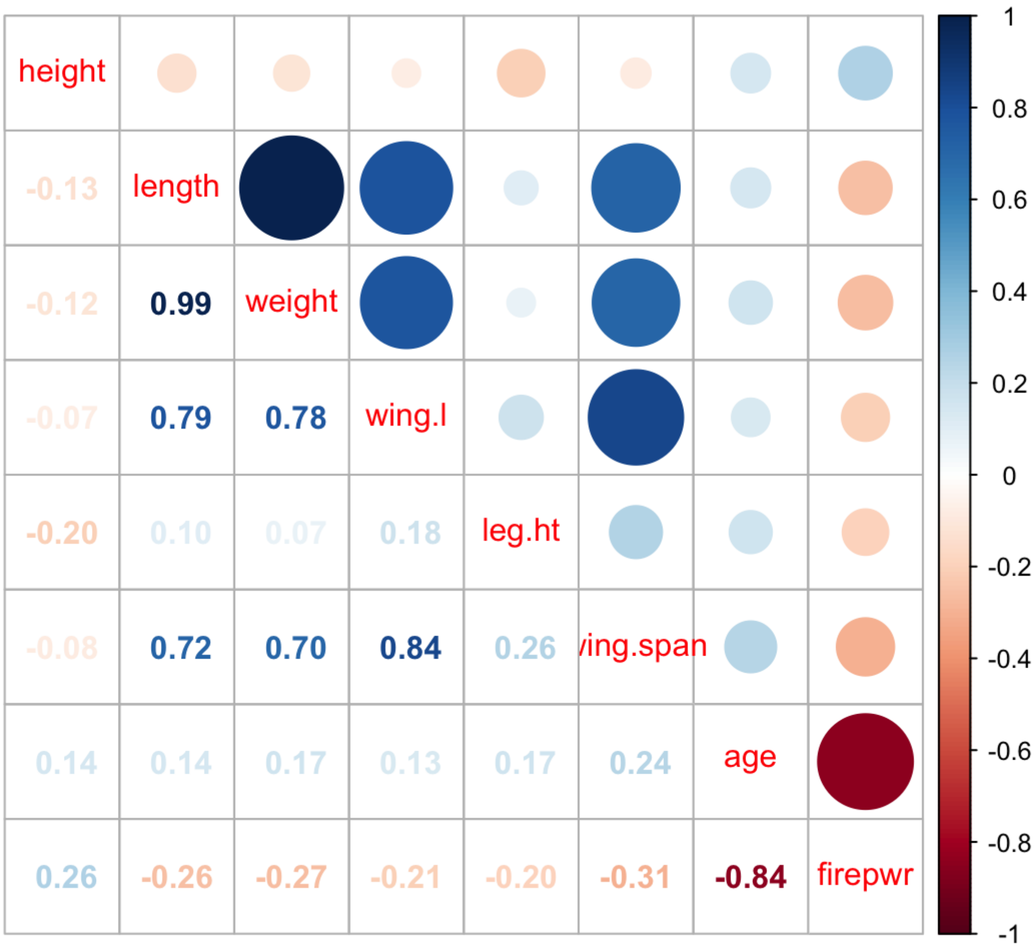
This question is about developing a model for `firepwr` as a function of the numerical variables in the set, excluding `strength` .

i. (5 pts) Do a scatterplot matrix for the numerical variables in the data set, excluding `strength` . Calculate and graph the correlation matrix for these variables. Comment on the results.

```
scatterplotMatrix(dragons[,c(1:6,8,10)])
```




```
 cor.dragon <- cor(dragons[,c(1:6,8,10)])
corrplot::corrplot.mixed(cor.dragon )
```



The highest correlation corresponds to weight and length, with a value of 0.99. The firepwr has strong negative correlation with age.

ii. (15 pts) Fit a regression model for `firepwr` as a function of the variables mentioned in (i). Using a critical α of 0.15 and a threshold for the variance inflation factor of 2, obtain a minimal adequate model that includes an intercept. Comment on the steps that you take.

```
lm1 = lm(firepwr ~ height+length+weight+ wing.l+ leg.ht+wing.span + age  ,data= dragons)
summary(lm1)
```

```
##
## Call:
## lm(formula = firepwr ~ height + length + weight + wing.l + leg.ht +
##     wing.span + age, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19.58  -6.00   2.00   5.53  17.71
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 135.6384    80.0995    1.69    0.098 .
## height        2.1025     0.3196    6.58  5.9e-08 ***
## length       -3.6115     2.5718   -1.40    0.168
## weight        1.4841     1.1522    1.29    0.205
## wing.l       -0.0267     0.6312   -0.04    0.966
## leg.ht        1.2048     1.1225    1.07    0.289
## wing.span    -0.0123     0.2339   -0.05    0.958
## age          -0.3039     0.0207  -14.69  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.74 on 42 degrees of freedom
## Multiple R-squared:  0.872,  Adjusted R-squared:  0.85
## F-statistic: 40.8 on 7 and 42 DF,  p-value: <2e-16
```

We choose a critical value of 0.15 for α. We remove wing.l which has the largest p-value.

```
lm2 <- update(lm1, ~. - wing.l)
summary(lm2)
```

```
##
## Call:
## lm(formula = firepwr ~ height + length + weight + leg.ht + wing.span +
##     age, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19.56  -5.91   1.98   5.51  17.67
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 135.3697    78.9159    1.72    0.093 .
## height        2.1015     0.3150    6.67  3.9e-08 ***
## length       -3.6101     2.5416   -1.42    0.163
## weight        1.4794     1.1335    1.31    0.199
## leg.ht        1.2028     1.1084    1.09    0.284
## wing.span    -0.0185     0.1792   -0.10    0.918
## age          -0.3037     0.0202  -15.05  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.63 on 43 degrees of freedom
## Multiple R-squared:  0.872,  Adjusted R-squared:  0.854
## F-statistic: 48.7 on 6 and 43 DF,  p-value: <2e-16
```

```
lm3 <- update(lm2, ~. - wing.span)
summary(lm3)
```

```
##
## Call:
## lm(formula = firepwr ~ height + length + weight + leg.ht + age,
##     data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19.51  -6.04   2.02   5.45  17.68
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 136.9122    76.6204    1.79    0.081 .
## height        2.1002     0.3112    6.75  2.7e-08 ***
## length       -3.6675     2.4524   -1.50    0.142
## weight        1.4930     1.1132    1.34    0.187
## leg.ht        1.1801     1.0742    1.10    0.278
## age          -0.3041     0.0196  -15.52  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.52 on 44 degrees of freedom
## Multiple R-squared:  0.872,  Adjusted R-squared:  0.857
## F-statistic: 59.8 on 5 and 44 DF,  p-value: <2e-16
```

```
lm4 <- update(lm3, ~. - leg.ht)
summary(lm4)
```

```
##
## Call:
## lm(formula = firepwr ~ height + length + weight + age, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -18.26  -6.53   1.47   5.82  18.80
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 119.4660    75.1284    1.59     0.12
## height        2.0283     0.3050    6.65  3.4e-08 ***
## length       -2.8295     2.3361   -1.21     0.23
## weight        1.1162     1.0615    1.05     0.30
## age          -0.2984     0.0189  -15.76  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.54 on 45 degrees of freedom
## Multiple R-squared:  0.868,  Adjusted R-squared:  0.856
## F-statistic: 74.1 on 4 and 45 DF,  p-value: <2e-16
```

```
lm5 <- update(lm4, ~. - weight)
summary(lm5)
```

```
##
## Call:
## lm(formula = firepwr ~ height + length + age, data = dragons)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19.29  -6.19   1.05   6.77  18.23
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  43.9488    22.1023    1.99    0.053 .
## height        2.0539     0.3043    6.75  2.2e-08 ***
## length       -0.3880     0.2590   -1.50    0.141
## age          -0.2938     0.0184  -15.94  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.55 on 46 degrees of freedom
## Multiple R-squared:  0.865,  Adjusted R-squared:  0.856
## F-statistic: 98.2 on 3 and 46 DF,  p-value: <2e-16
```

So now we obtain a model, satisfy the alpha requirement. Now take variance inflation factors test.

```
vif(lm5)
```

```
## height length    age
## 1.0449 1.0456 1.0486
```

And their vif all smaller than 2.

iii. (7.5 pts) Fit a model using the BIC criterion and compare it with the result of (ii).

```
library(MASS)
stepAIC(lm1, k = log(43))
```

```
## Start:  AIC=248.99
## firepwr ~ height + length + weight + wing.l + leg.ht + wing.span +
##     age
##
##             Df Sum of Sq   RSS AIC
## - wing.l     1         0  3984 245
## - wing.span  1         0  3984 245
## - leg.ht     1       109  4093 247
## - weight     1       157  4142 247
## - length     1       187  4171 248
## <none>                    3984 249
## - height     1      4105  8089 281
## - age        1     20475 24459 336
##
## Step:  AIC=245.23
## firepwr ~ height + length + weight + leg.ht + wing.span + age
##
##             Df Sum of Sq   RSS AIC
## - wing.span  1         1  3985 241
## - leg.ht     1       109  4093 243
## - weight     1       158  4142 243
## - length     1       187  4171 244
## <none>                    3984 245
## - height     1      4123  8107 277
## - age        1     20992 24977 333
##
## Step:  AIC=241.48
## firepwr ~ height + length + weight + leg.ht + age
##
##          Df Sum of Sq   RSS AIC
## - leg.ht  1       109  4095 239
## - weight  1       163  4148 240
## - length  1       203  4188 240
## <none>                 3985 241
## - height  1      4125  8110 273
## - age     1     21827 25812 331
##
## Step:  AIC=239.08
## firepwr ~ height + length + weight + age
##
##          Df Sum of Sq   RSS AIC
## - weight  1       101  4195 237
## - length  1       133  4228 237
## <none>                 4095 239
## - height  1      4025  8120 270
## - age     1     22614 26709 329
##
## Step:  AIC=236.53
## firepwr ~ height + length + age
##
##          Df Sum of Sq   RSS AIC
## - length  1       205  4400 235
## <none>                 4195 237
## - height  1      4154  8349 267
## - age     1     23164 27360 327
##
## Step:  AIC=235.15
```

```
## firepwr ~ height + age
##
##          Df Sum of Sq   RSS AIC
## <none>                 4400 235
## - height  1      4549  8949 267
## - age     1     24550 28950 326
```

```
##
## Call:
## lm(formula = firepwr ~ height + age, data = dragons)
##
## Coefficients:
## (Intercept)       height           age
##      19.669        2.124        -0.298
```

This procedures selects a model with only two regressors. The step is same with the result of (ii). Results is height + age. Before this final step, it choose firepwr ~ height + length + age. Notice that the length is biggest in final alpha comparison. Thus, two method reach same model.

iv. (7.5 pts) Write an equation for the final model and predict the `firepwr` for a dragon with the following covariates. Include confidence intervals at the 99% level.

```
model5 =lm(firepwr ~ height + age  ,data= dragons)
a = data.frame(height=50,age=350)
predict(model5,a, interval = 'c',confident = 0.99)
```

```
##      fit    lwr    upr
## 1 21.456 18.236 24.676
```

```
confint(model5, level = 0.99)
```

```
##                  0.5 %   99.5 %
## (Intercept) -21.20748 60.54629
## height        1.30598  2.94187
## age          -0.34777 -0.24886
```

Equation: $firepwr= 19.6694+ 2.1239*height -0.2983* age $

We can predict the `firepwr` for a dragon with the given covariates is 212.4557. and include a confidence interval from 18.23572 to 24.67568

v. (5 pts) Print an anova table for the final model and find the estimated variance of the errors. Describe explicitly the sampling distribution for the estimated parameters.

```
anova(model5)
```

```
## Analysis of Variance Table
##
## Response: firepwr
##           Df Sum Sq Mean Sq F value Pr(>F)
## height     1   2102    2102    22.5  2e-05 ***
## age        1  24550   24550   262.2 <2e-16 ***
## Residuals 47   4400      94
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

the estimated variance of the errors :93.6

The estimated parameters are β̂ = (β^0 , β^1, ,β^2), which have a normal distribution:

$\hat{\beta} = N((\beta 0, \beta 1, \beta 2)', \sigma^2 (X'X)^{-1}$

The matrix (X′X)^−1 is obtained in R with

```
(invXtX <- summary(model5)$cov.unscaled)
```

```
##              (Intercept)      height         age
## (Intercept)  2.47663253 -4.4201e-02 -9.3856e-04
## height      -0.04420075  9.9164e-04 -8.4382e-06
## age         -0.00093856 -8.4382e-06  3.6249e-06
```

The variance is unknown and is estimated by the mean square. The standard deviation is

```
summary(model5)$sigma
```

```
## [1] 9.6755
```

and the estimated variance is

```
summary(model5)$sigma^2
```

```
## [1] 93.616
```

The estimated covariance matrix for β̂ can be obtained with

```
vcov(model5)
```

```
##              (Intercept)      height         age
## (Intercept)  231.851977 -4.13788904 -0.08786406
## height        -4.137889  0.09283274 -0.00078995
## age           -0.087864 -0.00078995  0.00033935
```

or multiplying σ̂2 times (X′X)−1

```
(summary(model5)$sigma^2)*invXtX
```

```
##             (Intercept)      height         age
## (Intercept)  231.851977 -4.13788904 -0.08786406
## height        -4.137889  0.09283274 -0.00078995
## age           -0.087864 -0.00078995  0.00033935
```