# MATH 4432

# Result Report of Mini Project 1, Topic 2

LAU, Wing Shing - 20342662

## I. Introduction

This report is written to summarize the results in the discovery of the factors that affect the need of sleep of different animals. This research is based on the data from 63 distinct species with the following data fields.

| Data Field | Description |
|---|---|
| species | Name of the species |
| slowWaveSleep | Time of slow-wave sleeping of the species in hour(s) |
| dreamSleep | Time of sleep with dreams of the species in hour(s) |
| sleep | Overall time of sleep of the species in hour(s) |
| body | Average weight of the species in kilogram(s) |
| brain | Average weight of brain of species in gram(s) |
| life | Maximum lifespan of the species in year(s) |
| gestation | Period of pregnancy of the species in day(s) |
| predation | Predation index of the species, ranked from 1 to 5, where 1 is the least possible to be preyed and 5 is the most possible to be preyed |

| | |
|---|---|
| sleepExposure | Exposure index of the species during its sleep, ranked from 1 to 5, where 1 is the least exposed and 5 is the most exposed |
| danger | The dangerousness in the living environment of the species, ranked from 1 to 5, where 1 is the least in-danger and 5 is the most in-danger |

## II. Methodology

In this research, the response variable $Y$ would be sleep, and the predictor variables $X_i \; for \; i = 1, 2, \dots, 7$ include body, brain, life, gestation, predation, sleepExposure and danger. Body, brain, life and gestation are quantitative variables. Predation, sleepExposure and danger are qualitative variables.

As there are missing data(NA) in the dataset, k-nearest neighbours method with $k = 5$ will be used to estimate the missing values in the responses and predictors. For slowWaveSleep and dreamSleep, if both are missing, and variable sleep is presented, slowWaveSleep will be estimated based on k-NN with $k = 5$. Then, dreamSleep is estimated as $sleep - slowWaveSleep$ as $slowWaveSleep + dreamSleep = sleep$.

The regression model used for the response variable and the predictor variables will be **multiple linear regression method**.

The test error will be determined by the K-fold Cross Validation with K = 5, it will be calculated based on the following equation:

$$CV(K = 5) = \frac{1}{K}\sum_{i=1}^{K} MSE_i \; where \; MSE_i = \frac{1}{n}\sum_{j=1}^{n}\left(y_j - \hat{y_j}\right)^2$$

$$and \; n = size \; of \; validation \; dataset \; i$$

Bootstrapping is chosen as the model which determines the uncertainty of estimating the sleeping hours in the regression.

The data will be resampled for $n = 30$ and $m = 1000$, i.e. 100 times with 5 samples each time. By considering this dataset as the sample from the population, the estimation error $\bar{x} - \mu$ where $\bar{x}$ is the sample mean of sleeping hours and $\mu$ is the population mean of sleeping hours, is estimated by the distribution of the estimator named by resampling mean error which is $\overline{x_i^*} - \bar{x}$ where $\overline{x_i^*}$ is the mean of the resample $i$. If the estimator follows the normal distribution, estimation error of the population mean can be estimated by the 95% confidence interval which the start of interval is the 2.5% quantile of the distribution plot of estimator, and the end of interval is the 97.5% quantile of the plot.

Remarks: R language is applied in this research to observe the characteristics of the data and the correlation of the data fields, and make predictions upon the observation of the data.

## III. Code and Analysis

```
library(data.table)
data = fread("https://raw.githubusercontent.com/yuany-pku/data/master/sleep1.csv")
data = as.data.frame(data)
summary(data)
```

```
##    species          slowWaveSleep      dreamSleep        sleep
##  Length:62          Min.   : 2.100   Min.   :0.000   Min.   : 2.60
##  Class :character   1st Qu.: 6.250   1st Qu.:0.900   1st Qu.: 8.05
##  Mode  :character   Median : 8.350   Median :1.800   Median :10.45
##                     Mean   : 8.673   Mean   :1.972   Mean   :10.53
##                     3rd Qu.:11.000   3rd Qu.:2.550   3rd Qu.:13.20
##                     Max.   :17.900   Max.   :6.600   Max.   :19.90
##                     NA's   :14       NA's   :12      NA's   :4
##       body              brain              life           gestation

##  Min.   :   0.005   Min.   :   0.14   Min.   :  2.000   Min.   : 12.
00
##  1st Qu.:   0.600   1st Qu.:   4.25   1st Qu.:  6.625   1st Qu.: 35.
75
```

```
##  Median :   3.342   Median :  17.25   Median : 15.100   Median : 79.
00
##  Mean   : 198.790   Mean   : 283.13   Mean   : 19.878   Mean   :142.
35
##  3rd Qu.:  48.203   3rd Qu.: 166.00   3rd Qu.: 27.750   3rd Qu.:207.
50
##  Max.   :6654.000   Max.   :5712.00   Max.   :100.000   Max.   :645.
00
##                                       NA's   :4         NA's   :4

##    predation     sleepExposure       danger
##  Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000
##  Median :3.000   Median :2.000   Median :2.000
##  Mean   :2.871   Mean   :2.419   Mean   :2.613
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.000
##
```

As we can see, there are values with “NA” in the columns of “slowWaveSleep”, “dreamSleep”, “sleep”, “life” and “gestation”. We need to first fill in the missing values for responses and quantitative predictors by kNN with k = 5.

```
library(DMwR)

## Loading required package: lattice

## Loading required package: grid

dataWithValues = knnImputation(data[, 4:8], k = 5)
data = cbind(data[, 1:3], dataWithValues[, 1:5], data[, 9:11])
```

After filling the missing values, we need to determine slowWaveSleep or dreamSleep if either one of them and sleep variable are represents by simple subtraction.

```
for(i in 1:nrow(data)){
  slow = data[i, 2]
  dream = data[i, 3]
  sleep = data[i, 4]
  if(is.na(slow) && !is.na(dream)){
    data[i, 2] = sleep - dream
  }
  else if(!is.na(slow)&&is.na(dream)){
    data[i, 3] = sleep - slow
  }
}
```

In case the species has missing values for both slowWaveSleep or dreamSleep, we have to determine the value for the slowWaveSleep by kNN and subtract it to sleep to get dreamSleep. If slowWaveSleep is greater than sleep, then it would be set to the value of sleep.

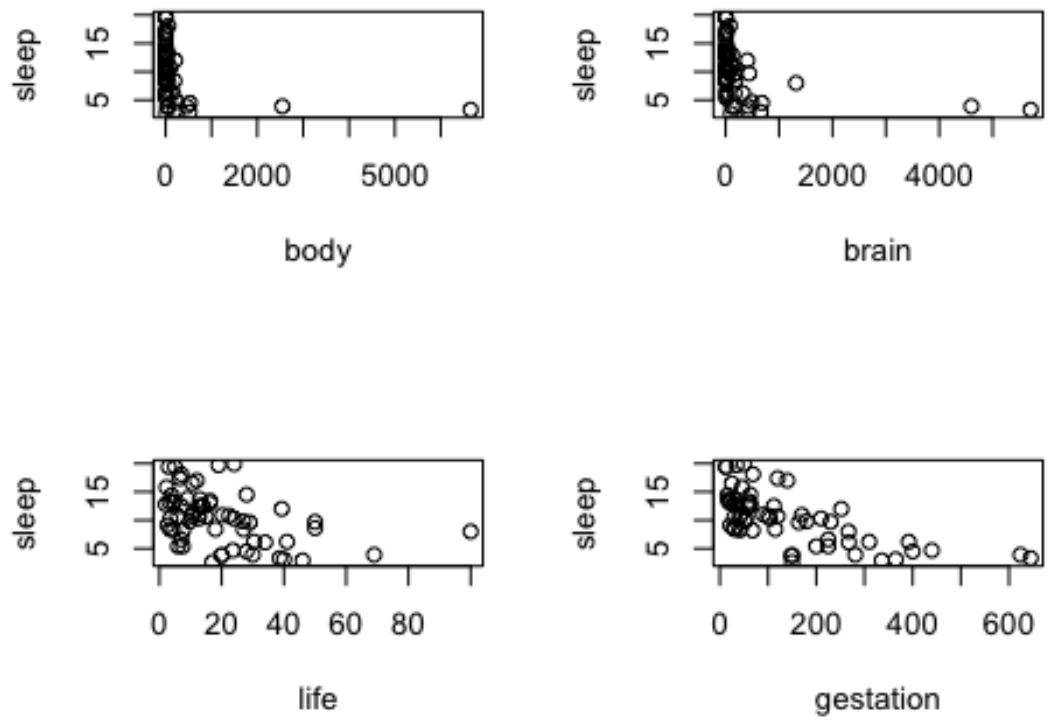
```
sleepRef = cbind(data[, 2], data[, 4:11])
col = colnames(data)
slowData = knnImputation(sleepRef[, 1:9], k = 5)
data = cbind(data[, 1], slowData[, 1], data[, 3:11])
for(i in 1:nrow(data)){
  slow = data[i, 2]
  dream = data[i, 3]
  sleep = data[i, 4]
  if(slow > sleep){
    data[i, 2] = sleep
  }
  if(is.na(dream)){
    data[i, 3] = sleep - data[i, 2]
  }
}
colnames(data) = col
summary(data)
```

```
##                        species   slowWaveSleep      dreamSleep
##  African_elephant          : 1   Min.   : 2.100   Min.   :0.000
##  African_giant_pouched_rat: 1   1st Qu.: 5.800   1st Qu.:0.925
##  Arctic_Fox                : 1   Median : 8.350   Median :1.900
##  Arctic_ground_squirrel    : 1   Mean   : 8.292   Mean   :2.127
##  Asian_elephant            : 1   3rd Qu.:10.750   3rd Qu.:2.776
##  Baboon                    : 1   Max.   :17.900   Max.   :6.600
##  (Other)                   :56
##      sleep            body              brain              life

##  Min.   : 2.60   Min.   :   0.005   Min.   :   0.14   Min.   :  2.00

##  1st Qu.: 6.95   1st Qu.:   0.600   1st Qu.:   4.25   1st Qu.:  7.00

##  Median :10.45   Median :   3.342   Median :  17.25   Median : 13.85

##  Mean   :10.42   Mean   : 198.790   Mean   : 283.13   Mean   : 19.39

##  3rd Qu.:13.20   3rd Qu.:  48.203   3rd Qu.: 166.00   3rd Qu.: 27.00

##  Max.   :19.90   Max.   :6654.000   Max.   :5712.00   Max.   :100.00

##
```

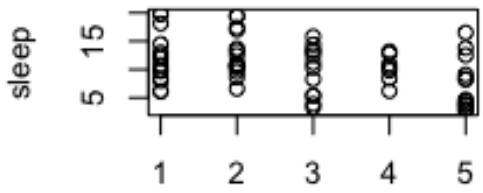
```
##    gestation         predation     sleepExposure       danger
##  Min.   : 12.00   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.: 39.00   1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000
##  Median : 79.35   Median :3.000   Median :2.000   Median :2.000
##  Mean   :141.14   Mean   :2.871   Mean   :2.419   Mean   :2.613
##  3rd Qu.:207.50   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :645.00   Max.   :5.000   Max.   :5.000   Max.   :5.000
##
```

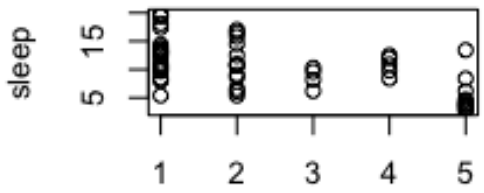
The estimated data are all filled up at this point. We need to next consider the relation between response and predictors.

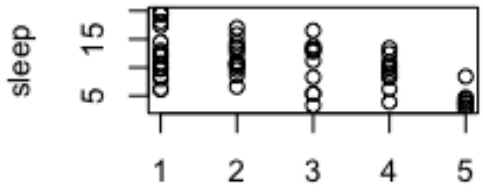
```
par(mfrow=c(2,2))
plot(sleep ~ body, data = data)
plot(sleep ~ brain, data = data)
plot(sleep ~ life, data = data)
plot(sleep ~ gestation, data = data)
```




```
plot(sleep ~ predation, data = data)
plot(sleep ~ sleepExposure, data = data)
plot(sleep ~ danger, data = data)
```

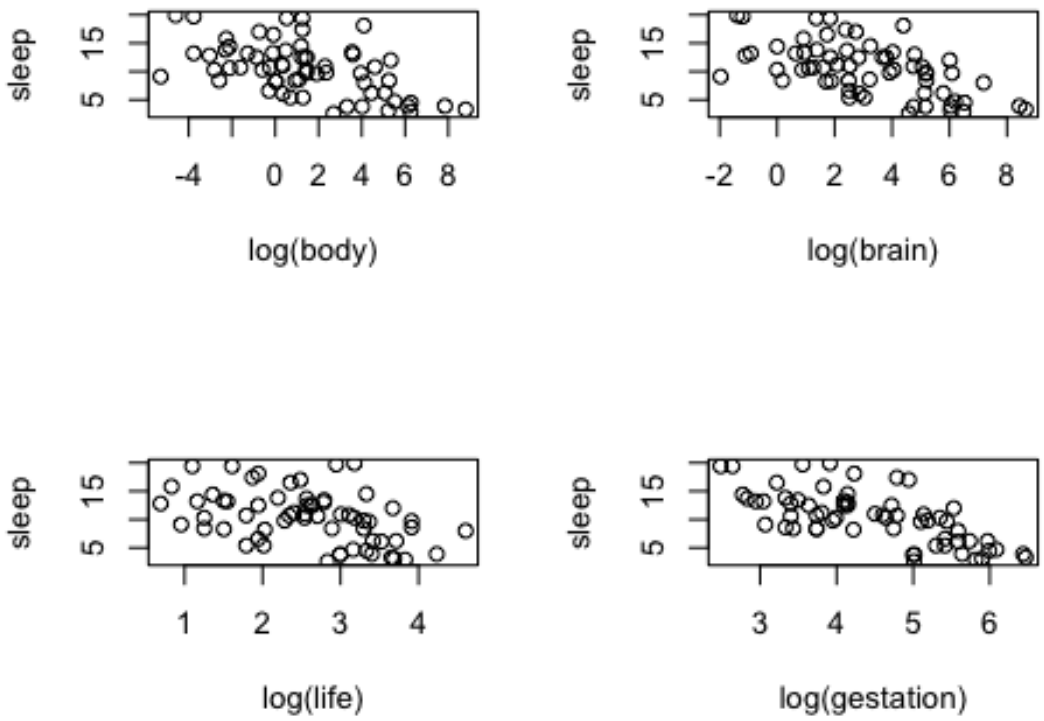



It is obviously that the predicting variables are not in a linear relation with quantitative respondent variable.

```
par(mfrow=c(2,2))
plot(sleep ~ log(body), data = data)
plot(sleep ~ log(brain), data = data)
plot(sleep ~ log(life), data = data)
plot(sleep ~ log(gestation), data = data)
```




From the result, taking ln function for the predictors will improve the linearity between predictors and response.

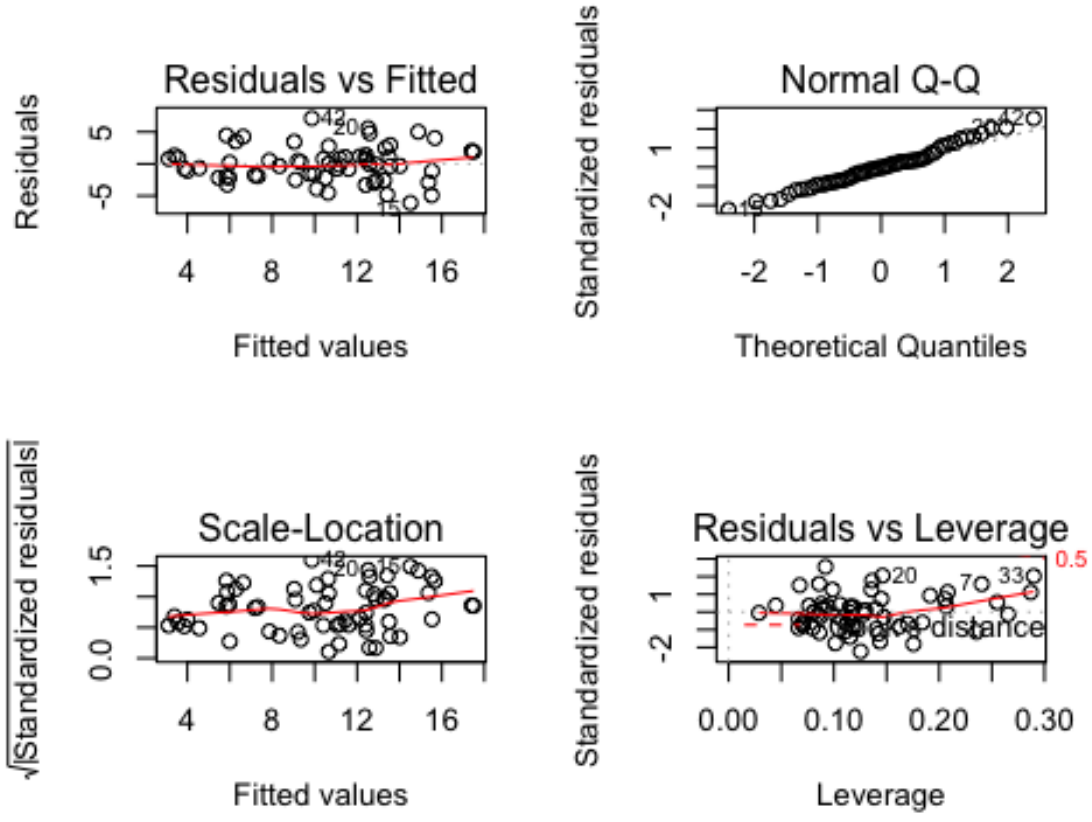
Next, we are trying to fit the responses with ln(quantitative predictors) and qualitative predictors, using the multiple linear regression.

```
fit = lm(sleep ~ log(body) + log(brain) + log(life) + log(gestation) +
(predation) + (sleepExposure) + (danger), data = data)
summary(fit)
```

```
##
## Call:
## lm(formula = sleep ~ log(body) + log(brain) + log(life) + log(gestat
ion) +
##     (predation) + (sleepExposure) + (danger), data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1150 -2.0027 -0.0788  1.2667  7.1298
##
## Coefficients:
```

```
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     23.8460     2.5931   9.196 1.22e-12 ***
## log(body)        0.2329     0.4700   0.496  0.62221
## log(brain)      -0.6237     0.7028  -0.888  0.37870
## log(life)        0.1127     0.7335   0.154  0.87851
## log(gestation)  -2.0267     0.6132  -3.305  0.00169 **
## predation        0.6133     0.7549   0.812  0.42008
## sleepExposure    0.6064     0.5584   1.086  0.28233
## danger          -2.3998     0.9217  -2.604  0.01189 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.934 on 54 degrees of freedom
## Multiple R-squared:  0.6401, Adjusted R-squared:  0.5935
## F-statistic: 13.72 on 7 and 54 DF,  p-value: 4.762e-10
```

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




It is acceptable for a regression model with an adjusted r-squared value > 0.5.

In the next part, K-fold cross validation with K = 5 will be performed to estimate the training error in the regression model.

```
data = data[sample(nrow(data)),]
numFolds = 5
num = 0
folds = cut(seq(1,nrow(data)),breaks = numFolds,labels = FALSE)
sum_mse = 0
for(i in 1:numFolds){
  testIndexes = which(folds == i,arr.ind = TRUE)
  testData = data[testIndexes, ]
  trainData = data[-testIndexes, ]
  train.fit = lm(sleep ~ log(body) + log(brain) + log(life) + log(gesta
tion) + (predation) + (sleepExposure) + (danger), data = trainData)
  sse = 0
  for(j in 1:nrow(testData)){
    yhat = coef(train.fit)[1] + coef(train.fit)[2] * log(testData[j,5])
 + coef(train.fit)[3] * log(testData[j,6]) + coef(train.fit)[4] * log(t
estData[j,7]) + coef(train.fit)[5] * log(testData[j,8]) + coef(train.fi
t)[6] * testData[j,9] + coef(train.fit)[7] * testData[j,10] + coef(trai
n.fit)[8] * testData[j,11]
    se = (testData[j,4] - yhat)^2
    sse = sse + se
  }
  mse = sse/nrow(testData)
  sum_mse = sum_mse + mse
}
CV_K = sum_mse/numFolds
print(CV_K)
```

```
## (Intercept)
##    10.58154
```

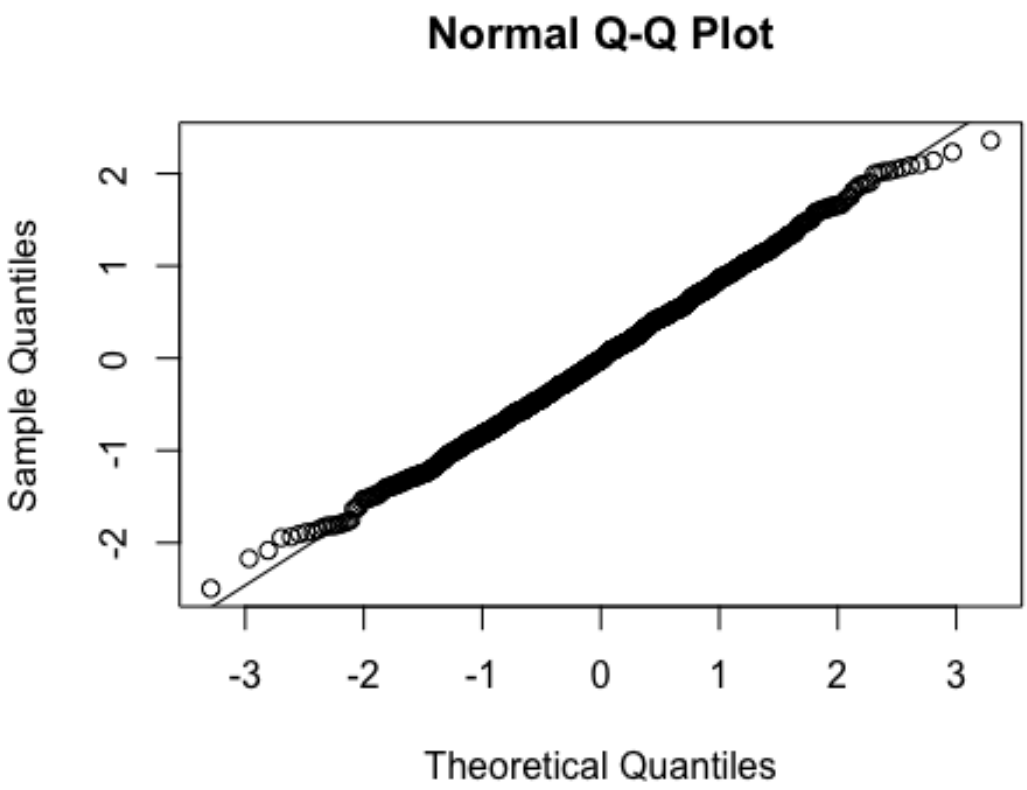
This is the value for CV(K) after K-fold cross validation with K = 5 is performed.

In the next part, bootstrap is performed to quantify the uncertainty for the estimation, which is the estimation error of sleep hours in the regression model, with n = 30, m = 1000.

```
sampleSleepHours = data[, 4]
n = 30
m = 1000
resamplingMeans = rep(0, m)
meanSampleSleep = mean(sampleSleepHours)
for(i in 1:m){
  resamples = sample(sampleSleepHours, n, replace = TRUE)
  resamplingMeans[i] = mean(resamples)
```

```
}
fullResampleErrors = sort(resamplingMeans) - meanSampleSleep
hist(fullResampleErrors)
```


```
qqnorm(fullResampleErrors)
abline(a = mean(fullResampleErrors), b = sd(fullResampleErrors))
```




The resampling mean errors is likely following the normal distribution. By choosing the estimation error between resample mean and sample mean as the estimator of the actual estimation error between sample mean and population mean, we can conclude that the 95% confidence interval for actual estimation error is approximately equal to:

```
ci = quantile(fullResampleErrors, c(0.025,0.975))
print(paste("[", ci[1], ", ", ci[2], "]"))
```

```
## [1] "[ -1.51903010381943 ,  1.64181620488814 ]"
```

## IV. Summary

From the result of regression above, after the response variable is fitted with predictor variables, there are three variables with negative intercepts, which are log(brain), log(gestation) and danger. While other variables are having positive intercepts, which include log(body), log(life), predation, and sleepExposure. This fact indicates that the increase in brain weight, gestation period and in-dangerousness would decrease the sleeping time, while the increase in body weight, lifespan, ease of predation and sleep exposure would increase the sleep time of the species. In the level of significance, gestation period and in-dangerousness have significantly and negatively affected the sleep time of the species, while predation and sleep exposure have significantly and positively affected the sleep time of the species. In overall, the main factors that affect the sleep time of the species would be **gestation period and in-dangerousness**.

From the result of cross validation above, the training error in the regression model is estimated as **10.58154**.

From the result of bootstrapping above, the resample mean errors follow the normal distribution after the observation of Q-Q plot. Therefore, we can conclude that the estimation error between sample mean and population mean would be in range between 2.5% and 97.5% quantile of the distribution plot of the resample mean errors, which is approximated as

**[ -1.51903010381943 , 1.64181620488814 ]**

with 95% confidence level.