Boon or Bane? Is E-Learning affecting the grades of students?

Rommel Bartolome

Summary

E-Learning has been very widespread in past century due to the rise of computers. Several benefits of e-learning includes reduced costs, scalability, consistency and ability to accomodate different time zones. However, there are skeptics which say that e-learning can be a "bane" as it removes the focus of the students due to the freedom it gives. In this exercise, we discover knowledge on a dataset where students use an elearning facility wherein while using the platform, their online footprint is being recorded. From these, we engineered four features: (1) focus_level - times where they were browsing non-related sites, (2) time_finished - time it takes to finish the exercise, (3) idle_time - time where they were doing nothing, and (4) activity_level - number of clicks, keystrokes and mouse wheeling they made. We then find correlation of these engineered features with grades, using a linear model. It was found that among all of the features, only the time_finished was the significant variable and the model with only this variable gives the best prediction. This means that the faster a student finishes the exercise, the higher the student's final grade will be. Among all the variables, this is the most unrelated to e-learning. This also means that a student browsing non-related sites, has lots of idle time and those with little activity level does not necessarily mean they will get lower grades. As such, while e-learning may remove the focus of the students as they are free to visit any site during the session, it does not necessarily mean that e-learning will be a "bane", as it does not significantly affect the student's grade.

Data Reading

We read the data first, but before that we will need the following tools to do this:

```
library(plyr)
library(dplyr)
library(tidyverse)
library(gtools)
library(knitr)
library(openxlsx)
library(GGally)
```

Looking at the documentation, the data set contains the students' time series of activities during six sessions of laboratory sessions of the course of digital electronics. There are 6 folders containing the students' data per session. Each 'Session' folder contains up to CSV files each dedicated to a specific student log during that session. The number of files in each folder changes due to the number of students present in each session. Each file contains 13 features. The features are the following:

Feature	Description
session	It shows the number of laboratory session from 1 to 6.
student_Id	It shows the Id of student from 1 to 115.
exercise	It shows the Id of the exercise the student is working on.
activity	The activities are labeled based on the title of web pages that are on focus.
start_time	It shows the start date and time of a specific activity.
end_time	It shows the end date and time of a specific activity.
idle_time	It shows the duration of idle time between the start and end time of an activity.
mouse_wheel	It shows the amount of mouse wheel during an activity.
mouse_wheel_click	It shows the number of mouse wheel clicks during an activity.

Feature	Description
mouse_click_left	It shows the number of mouse left clicks during an activity.
mouse_click_right	It shows the number of mouse right clicks during an activity.
mouse_movement	It shows the distance covered by the mouse movements during an activity.
keystroke	It shows the number of keystrokes during an activity.

To read the data, we will create a for loop that will run through all of the six Session folders, and then extract that and bind with the previous "sessions", then write it in a .csv file to save it:

```
colnames <- c("session", "student_Id", "exercise", "activity", "start_time",
              "end_time", "idle_time", "mouse_wheel", "mouse_wheel_click",
              "mouse_click_left", "mouse_click_right", "mouse_movement",
              "keystroke")

data <- NULL

for (i in 1:6){
  setwd(paste0("C:/Users/rommel.bartolome/Documents/
               Stat 227/EPM Dataset 2/Data/Processes/Session ",i,"/"))
  filenames <- list.files()
  filenames <- mixedsort(filenames)
  dataframe <- ldply(.data = filenames, .fun = read.csv, col.names = colnames)
  data <- rbind(data, dataframe)
}

setwd("C:/Users/rommel.bartolome/Documents/Stat 227")
data %>% write.csv("education_data.csv")
```

We will also read the grades values:

```
finalgrades1 <- read.xlsx("final_grades.xlsx", sheet = 1)
finalgrades2 <- read.xlsx("final_grades.xlsx", sheet = 2)
int_grades <- read.xlsx("intermediate_grades.xlsx")
```

We have now successfully read the files provides. The next part would be exploring our data.

Exploratory Data Analysis

The first thing we would like to see is how is the main data looking:

```
education <- read.csv("education_data.csv")[,-1]
education %>% head()
```

```
##   session student_Id exercise activity          start_time            end_time
## 1       1          1       Es  Aulaweb  2.10.2014 11:25:35  2.10.2014 11:25:42
## 2       1          1       Es    Blank  2.10.2014 11:25:43  2.10.2014 11:25:43
## 3       1          1       Es    Deeds  2.10.2014 11:25:44  2.10.2014 11:26:17
## 4       1          1       Es    Other  2.10.2014 11:26:18  2.10.2014 11:26:18
## 5       1          1       Es    Other  2.10.2014 11:26:19  2.10.2014 11:26:27
## 6       1          1       Es    Blank  2.10.2014 11:26:28  2.10.2014 11:26:28
##   idle_time mouse_wheel mouse_wheel_click mouse_click_left mouse_click_right
## 1       218           0                 0                4                 0
## 2         0           0                 0                0                 0
## 3    154117           6                 0                8                 0
```

```
## 4         0           0                 0                 2                 0
## 5       460           0                 0                 4                 0
## 6         0           0                 0                 1                 0
##   mouse_movement keystroke
## 1            397         0
## 2             59         0
## 3           1581         4
## 4            103         0
## 5            424         8
## 6             93         0
```

```
education %>% nrow()
```

```
## [1] 229798
```

From here, we can see that there are 13 columns and around 229,798 rows.

We also check the data from the final and intermediate grades:

```
finalgrades1 %>% head()
```

```
##   Student.ID ES.1.1.(2.points) ES.1.2.(3.points) ES.2.1.(2.points)
## 1          3                 2                 3                 1
## 2          6                 2                 3                 2
## 3          7                 2                 3                 1
## 4         10                 2                 3                 2
## 5         13                 2                 3                 2
## 6         15                 2                 3                 1
##   ES.2.2.(3.points) ES.3.1.(1.points) ES.3.2.(2.points) ES.3.3.(2.points)
## 1               2.0                 1                 2                 2
## 2               3.0                 1                 2                 2
## 3               1.5                 1                 2                 0
## 4               1.5                 1                 2                 0
## 5               1.5                 1                 2                 2
## 6               2.0                 1                 2                 2
##   ES.3.4.(2.points) ES.3.5.(3.points) ES.4.1.(15.points) ES.4.2.(10.points)
## 1                 2                 3               15.0                 10
## 2                 0                 3               15.0                  7
## 3                 0                 3                5.0                  4
## 4                 2                 3               11.0                  1
## 5                 2                 3               14.5                 10
## 6                 2                 3               15.0                 10
##   ES.5.1.(2.points) ES.5.2.(10.points) ES.5.3.(3.points) ES.6.1.(25.points)
## 1                 1                  5                3.0                 18
## 2                 2                  9                3.0                 13
## 3                 0                  0                3.0                 17
## 4                 2                 10                1.5                  7
## 5                 2                  2                3.0                 25
## 6                 2                  4                1.5                  2
##   ES.6.2.(15.points) TOTAL.(100.points)
## 1                 15               85.0
## 2                 15               82.0
## 3                 10               52.5
## 4                 10               59.0
## 5                 15               90.0
## 6                 15               67.5
```

```
finalgrades2 %>% head()
```

```
##   Student.ID ES.1.1.(2.points) ES.1.2.(3.points) ES.2.1.(2.points)
## 1          1               2.0                 3                 1
## 2          2               2.0                 3                 2
## 3          4               2.0                 3                 1
## 4          5               2.0                 3                 2
## 5          7               2.0                 3                 1
## 6          8               0.5                 3                 0
##   ES.2.2.(3.points) ES.3.1.(1.points) ES.3.2.(2.points) ES.3.3.(2.points)
## 1               0.5                 1                 2                 2
## 2               0.5                 1                 2                 0
## 3               0.5                 1                 2                 0
## 4               1.5                 1                 2                 2
## 5               1.5                 1                 2                 2
## 6               0.0                 1                 2                 0
##   ES.3.4.(2.points) ES.3.5.(3.points) ES.4.1.(15.points) ES.4.2.(10.points)
## 1                 2                 3                 15                10
## 2                 2                 3                 15                 2
## 3                 2                 0                  3                 4
## 4                 2                 3                  3                 2
## 5                 2                 3                 15                10
## 6                 2                 0                  0                 0
##   ES.5.1.(2.points) ES.5.2.(10.points) ES.5.3.(3.points) ES.6.1.(25.points)
## 1               2.0                10.0               3.0                25
## 2               0.0                 5.0               1.5                 5
## 3               0.0                 1.5               0.0                 5
## 4               1.5                 9.0               1.5                 2
## 5               1.0                 2.5               0.0                20
## 6               0.0                 0.0               0.0                 0
##   ES.6.2.(15.points) TOTAL.(100.points)
## 1                 13                94.5
## 2                  0                44.0
## 3                  5                30.0
## 4                  1                38.5
## 5                 12                78.0
## 6                  0                 8.5
```

```
int_grades %>% head()
```

```
##   Student.Id Session.2 Session.3 Session.4 Session.5 Session.6
## 1          1       5.0       0.0       4.5       4.0      2.25
## 2          2       4.0       3.5       4.5       4.0      1.00
## 3          3       3.5       3.5       4.5       4.0      0.00
## 4          4       6.0       4.0       5.0       3.5      2.75
## 5          5       5.0       4.0       5.0       4.0      2.75
## 6          6       5.5       3.5       4.5       3.0      3.00
```

We now check the summary statistics:

```
education %>% summary()
```

```
##     session        student_Id        exercise             activity
##  Min.   :1.000   Min.   :  1.00   Es_4_5: 22351   Other     : 33128
##  1st Qu.:2.000   1st Qu.: 27.00   Es_6_3: 19549   Blank     : 24291
##  Median :4.000   Median : 53.00   Es_6_2: 16128   Diagram   : 20815
```

```
##  Mean   :3.695   Mean   : 53.63    Es_6_1: 13617    Properties: 19677
##  3rd Qu.:5.000   3rd Qu.: 81.00    Es_5_4: 12376    Aulaweb   :  8251
##  Max.   :6.000   Max.   :115.00    Es_1_1: 11531    FSM_Es_6_3:  7709
##                                   (Other):134246   (Other)    :115927
##                 start_time                     end_time
##   13.11.2014 11:58:3 :    18    13.11.2014 11:33:27:    17
##   13.11.2014 11:59:1 :    18    13.11.2014 11:41:43:    17
##   13.11.2014 11:41:44:    17    13.11.2014 11:59:0 :    17
##   13.11.2014 11:41:52:    17    13.11.2014 11:33:32:    16
##   13.11.2014 11:42:42:    17    13.11.2014 11:37:14:    16
##   13.11.2014 11:51:53:    17    13.11.2014 11:40:8 :    16
##  (Other)             :229694   (Other)             :229699
##    idle_time           mouse_wheel      mouse_wheel_click  mouse_click_left
##  Min.   :-2.059e+14   Min.   :   0.00   Min.   : 0.00000   Min.   :   0.000
##  1st Qu.: 0.000e+00   1st Qu.:   0.00   1st Qu.: 0.00000   1st Qu.:   2.000
##  Median : 8.000e+01   Median :   0.00   Median : 0.00000   Median :   2.000
##  Mean   :-1.852e+09   Mean   :   2.75   Mean   : 0.00546   Mean   :   7.083
##  3rd Qu.: 5.779e+03   3rd Qu.:   0.00   3rd Qu.: 0.00000   3rd Qu.:   5.000
##  Max.   : 7.245e+09   Max.   :2904.00   Max.   :60.00000   Max.   :1096.000
##
##  mouse_click_right  mouse_movement      keystroke
##  Min.   :  0.0000   Min.   :    0.0   Min.   :   0.000
##  1st Qu.:  0.0000   1st Qu.:   62.0   1st Qu.:   0.000
##  Median :  0.0000   Median :  138.0   Median :   0.000
##  Mean   :  0.3369   Mean   :  415.1   Mean   :   6.294
##  3rd Qu.:  0.0000   3rd Qu.:  336.0   3rd Qu.:   0.000
##  Max.   :168.0000   Max.   :85949.0   Max.   :4754.000
##
```

```
finalgrades1 %>% summary()
```

```
##    Student.ID     ES.1.1.(2.points) ES.1.2.(3.points) ES.2.1.(2.points)
##  Min.   :  3.00   Min.   :0.000     Min.   :0.000     Min.   :0.000
##  1st Qu.: 28.75   1st Qu.:2.000     1st Qu.:3.000     1st Qu.:1.000
##  Median : 57.50   Median :2.000     Median :3.000     Median :1.500
##  Mean   : 55.54   Mean   :1.885     Mean   :2.865     Mean   :1.433
##  3rd Qu.: 83.50   3rd Qu.:2.000     3rd Qu.:3.000     3rd Qu.:2.000
##  Max.   :106.00   Max.   :2.000     Max.   :3.000     Max.   :2.000
##  ES.2.2.(3.points) ES.3.1.(1.points) ES.3.2.(2.points) ES.3.3.(2.points)
##  Min.   :0.00      Min.   :0.0000    Min.   :0.000     Min.   :0.000
##  1st Qu.:0.00      1st Qu.:1.0000    1st Qu.:2.000     1st Qu.:0.000
##  Median :1.50      Median :1.0000    Median :2.000     Median :2.000
##  Mean   :1.26      Mean   :0.9423    Mean   :1.904     Mean   :1.423
##  3rd Qu.:2.00      3rd Qu.:1.0000    3rd Qu.:2.000     3rd Qu.:2.000
##  Max.   :3.00      Max.   :1.0000    Max.   :2.000     Max.   :2.000
##  ES.3.4.(2.points) ES.3.5.(3.points) ES.4.1.(15.points) ES.4.2.(10.points)
##  Min.   :0.000     Min.   :0.000     Min.   : 0.000     Min.   : 0.000
##  1st Qu.:2.000     1st Qu.:3.000     1st Qu.: 3.000     1st Qu.: 4.000
##  Median :2.000     Median :3.000     Median :13.000     Median : 7.000
##  Mean   :1.721     Mean   :2.375     Mean   : 9.769     Mean   : 6.413
##  3rd Qu.:2.000     3rd Qu.:3.000     3rd Qu.:15.000     3rd Qu.:10.000
##  Max.   :2.000     Max.   :3.000     Max.   :15.000     Max.   :10.000
##  ES.5.1.(2.points) ES.5.2.(10.points) ES.5.3.(3.points) ES.6.1.(25.points)
##  Min.   :0.000     Min.   : 0.000     Min.   :0.000     Min.   : 0.00
##  1st Qu.:0.000     1st Qu.: 0.000     1st Qu.:0.000     1st Qu.: 2.75
```

```
##  Median :1.000     Median : 4.000     Median :1.750     Median :12.00
##  Mean   :1.058     Mean   : 4.192     Mean   :1.827     Mean   :12.15
##  3rd Qu.:2.000     3rd Qu.: 8.000     3rd Qu.:3.000     3rd Qu.:24.00
##  Max.   :2.000     Max.   :10.000     Max.   :3.000     Max.   :25.00
##  ES.6.2.(15.points) TOTAL.(100.points)
##  Min.   : 0.000     Min.   : 7.00
##  1st Qu.: 2.250     1st Qu.:38.88
##  Median :10.000     Median :67.25
##  Mean   : 8.846     Mean   :60.07
##  3rd Qu.:15.000     3rd Qu.:82.00
##  Max.   :15.000     Max.   :98.00
```

```
finalgrades2 %>% summary()
```

```
##    Student.ID     ES.1.1.(2.points) ES.1.2.(3.points) ES.2.1.(2.points)
##  Min.   :  1.00   Min.   :0.000     Min.   :1.500     Min.   :0.000
##  1st Qu.: 30.75   1st Qu.:2.000     1st Qu.:3.000     1st Qu.:1.000
##  Median : 55.50   Median :2.000     Median :3.000     Median :1.000
##  Mean   : 53.61   Mean   :1.879     Mean   :2.847     Mean   :1.323
##  3rd Qu.: 76.50   3rd Qu.:2.000     3rd Qu.:3.000     3rd Qu.:2.000
##  Max.   :106.00   Max.   :2.000     Max.   :3.000     Max.   :2.000
##  ES.2.2.(3.points) ES.3.1.(1.points) ES.3.2.(2.points) ES.3.3.(2.points)
##  Min.   :0.000     Min.   :0.0000    Min.   :0.000     Min.   :0.000
##  1st Qu.:0.500     1st Qu.:1.0000    1st Qu.:2.000     1st Qu.:0.000
##  Median :0.500     Median :1.0000    Median :2.000     Median :2.000
##  Mean   :1.113     Mean   :0.9677    Mean   :1.839     Mean   :1.145
##  3rd Qu.:2.000     3rd Qu.:1.0000    3rd Qu.:2.000     3rd Qu.:2.000
##  Max.   :3.000     Max.   :1.0000    Max.   :2.000     Max.   :2.000
##  ES.3.4.(2.points) ES.3.5.(3.points) ES.4.1.(15.points) ES.4.2.(10.points)
##  Min.   :0.000     Min.   :0.000     Min.   : 0.000     Min.   : 0.000
##  1st Qu.:2.000     1st Qu.:2.250     1st Qu.: 4.000     1st Qu.: 2.000
##  Median :2.000     Median :3.000     Median :12.000     Median : 4.000
##  Mean   :1.839     Mean   :2.258     Mean   : 9.468     Mean   : 4.718
##  3rd Qu.:2.000     3rd Qu.:3.000     3rd Qu.:15.000     3rd Qu.: 8.375
##  Max.   :2.000     Max.   :3.000     Max.   :15.000     Max.   :10.000
##  ES.5.1.(2.points) ES.5.2.(10.points) ES.5.3.(3.points) ES.6.1.(25.points)
##  Min.   :0.0000    Min.   : 0.000     Min.   :0.000     Min.   : 0.000
##  1st Qu.:0.0000    1st Qu.: 0.000     1st Qu.:0.000     1st Qu.: 0.000
##  Median :1.0000    Median : 3.500     Median :1.500     Median : 5.000
##  Mean   :0.8548    Mean   : 4.081     Mean   :1.290     Mean   : 8.968
##  3rd Qu.:1.8750    3rd Qu.: 7.500     3rd Qu.:1.875     3rd Qu.:15.750
##  Max.   :2.0000    Max.   :10.000     Max.   :3.000     Max.   :25.000
##  ES.6.2.(15.points) TOTAL.(100.points)
##  Min.   : 0.000     Min.   : 8.50
##  1st Qu.: 0.000     1st Qu.:30.00
##  Median : 5.000     Median :45.00
##  Mean   : 5.169     Mean   :49.76
##  3rd Qu.:10.000     3rd Qu.:69.50
##  Max.   :15.000     Max.   :97.50
```

```
int_grades %>% summary()
```

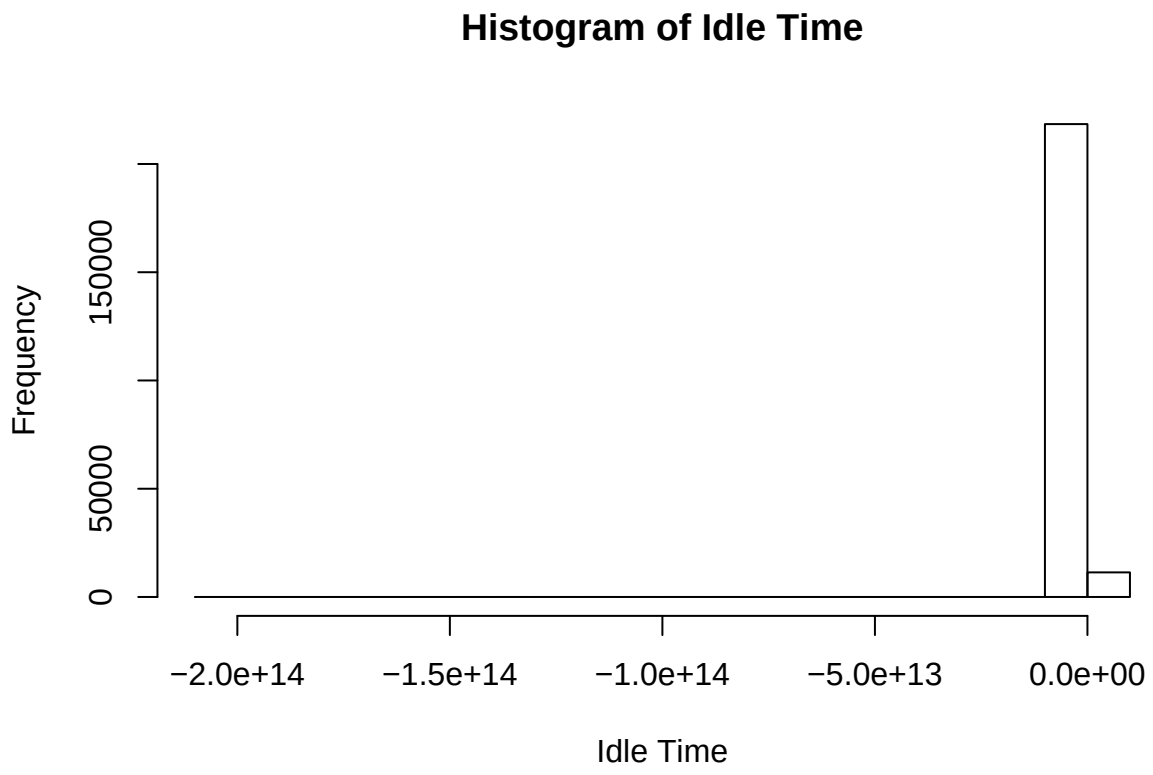
```
##    Student.Id      Session.2       Session.3       Session.4       Session.5
##  Min.   :  1.0   Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.00
##  1st Qu.: 29.5   1st Qu.:0.000   1st Qu.:0.500   1st Qu.:4.000   1st Qu.:3.00
```

```
##  Median : 58.0   Median :3.500   Median :2.500   Median :4.500   Median :3.50
##  Mean   : 58.0   Mean   :2.887   Mean   :2.135   Mean   :3.943   Mean   :3.03
##  3rd Qu.: 86.5   3rd Qu.:4.500   3rd Qu.:3.500   3rd Qu.:5.000   3rd Qu.:4.00
##  Max.   :115.0   Max.   :6.000   Max.   :4.000   Max.   :5.000   Max.   :4.00
##    Session.6
##  Min.   :0.000
##  1st Qu.:0.125
##  Median :2.000
##  Mean   :1.696
##  3rd Qu.:2.750
##  Max.   :4.000
```

The tables above show the summary statistics of the variables. This is helpful in checking what the usual values of the variables.

We explore the histogram values of the some of the variables:

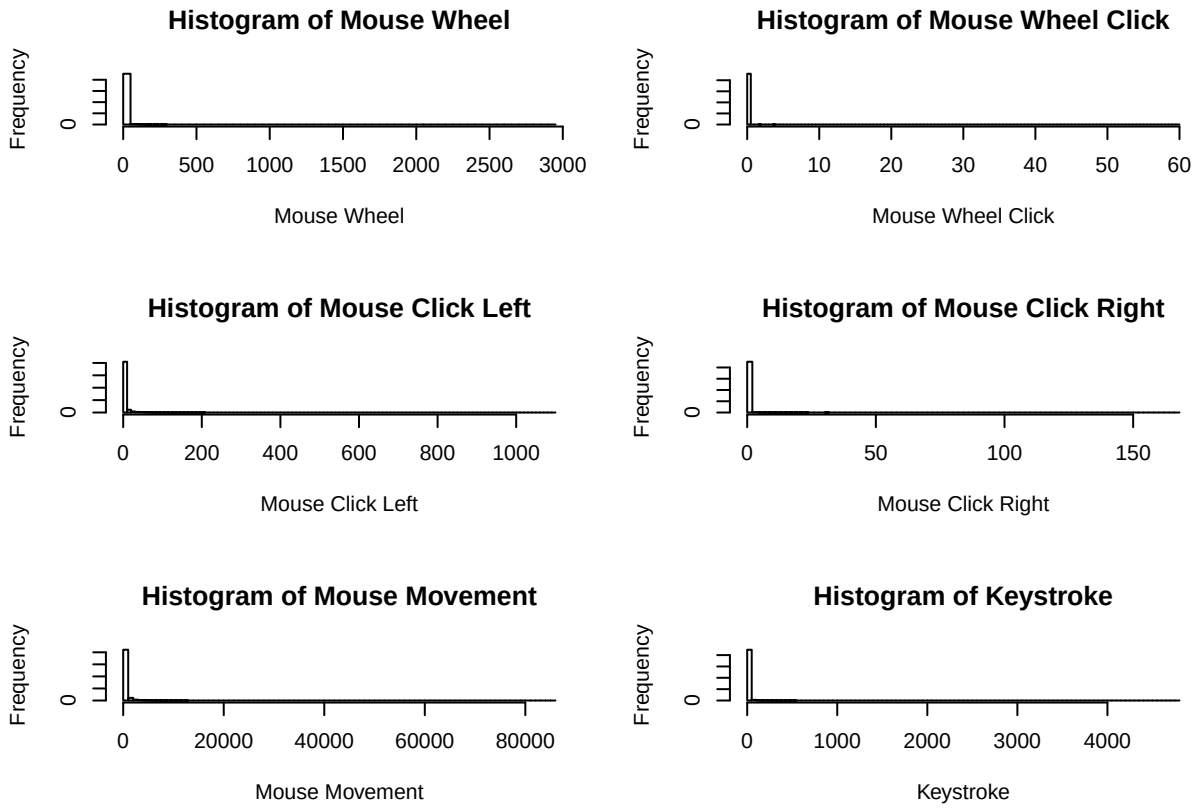
```
hist(education$idle_time, main = "Histogram of Idle Time", xlab = "Idle Time")
```



From this we can see that majority of the students have zero idle time. We also check the histogram of other variables:

```
par(mfrow=c(3,2))
hist(education$mouse_wheel, main = "Histogram of Mouse Wheel",
     xlab = "Mouse Wheel", breaks = 100)
hist(education$mouse_wheel_click, main = "Histogram of Mouse Wheel Click",
     xlab = "Mouse Wheel Click", breaks = 100)
hist(education$mouse_click_left, main = "Histogram of Mouse Click Left",
```

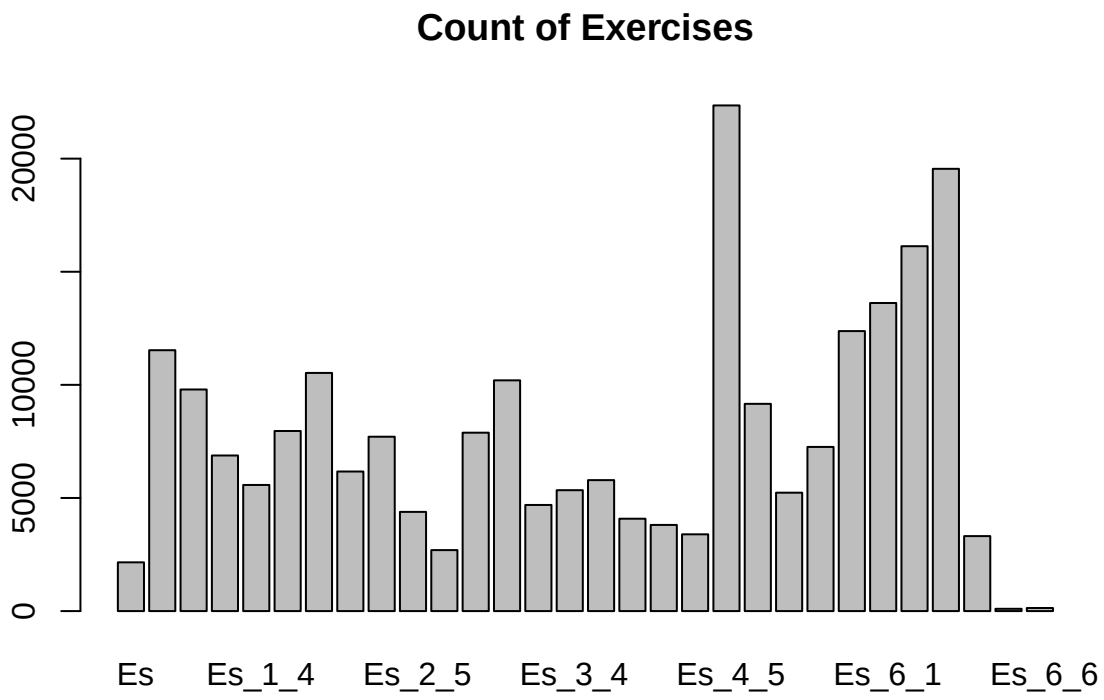
```
     xlab = "Mouse Click Left", breaks = 100)
hist(education$mouse_click_right, main = "Histogram of Mouse Click Right",
     xlab = "Mouse Click Right", breaks = 100)
hist(education$mouse_movement, main = "Histogram of Mouse Movement",
     xlab = "Mouse Movement", breaks = 100)
hist(education$keystroke, main = "Histogram of Keystroke",
     xlab = "Keystroke", breaks = 100)
```



From the values above, we can see that similar to the idle time, most students have very limited number of keystrokes, clicks and movements.

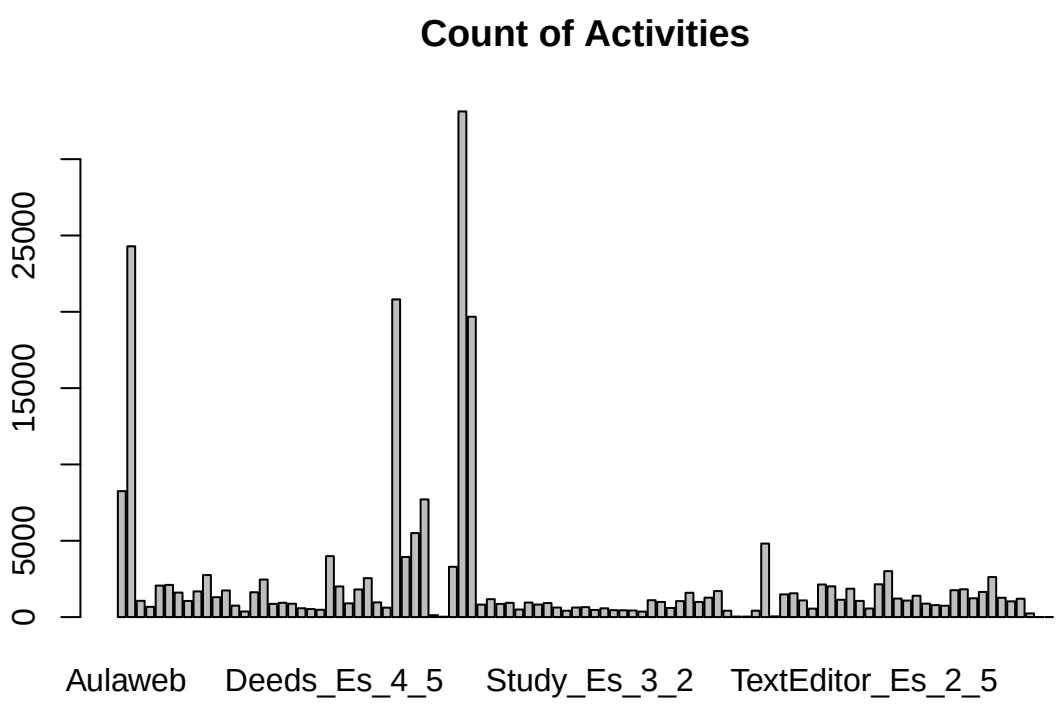
Also, we would want to check the counts of the activity and exercise:

```
plot(education$exercise, main = 'Count of Exercises')
```



From here, we can see that there are some exercises with fewer counts. We also check the activities:

```
plot(education$activity, main = 'Count of Activities')
```



Again, similar to the first one. There are some activities that have more counts. However, a spike was seen on "Others", which are times When the student is not viewing any pages related to academic work. This includes, for majority of the cases, the student irrelevant activity to the course (e.g. if the student is on Facebook).

Knowledge Discovery

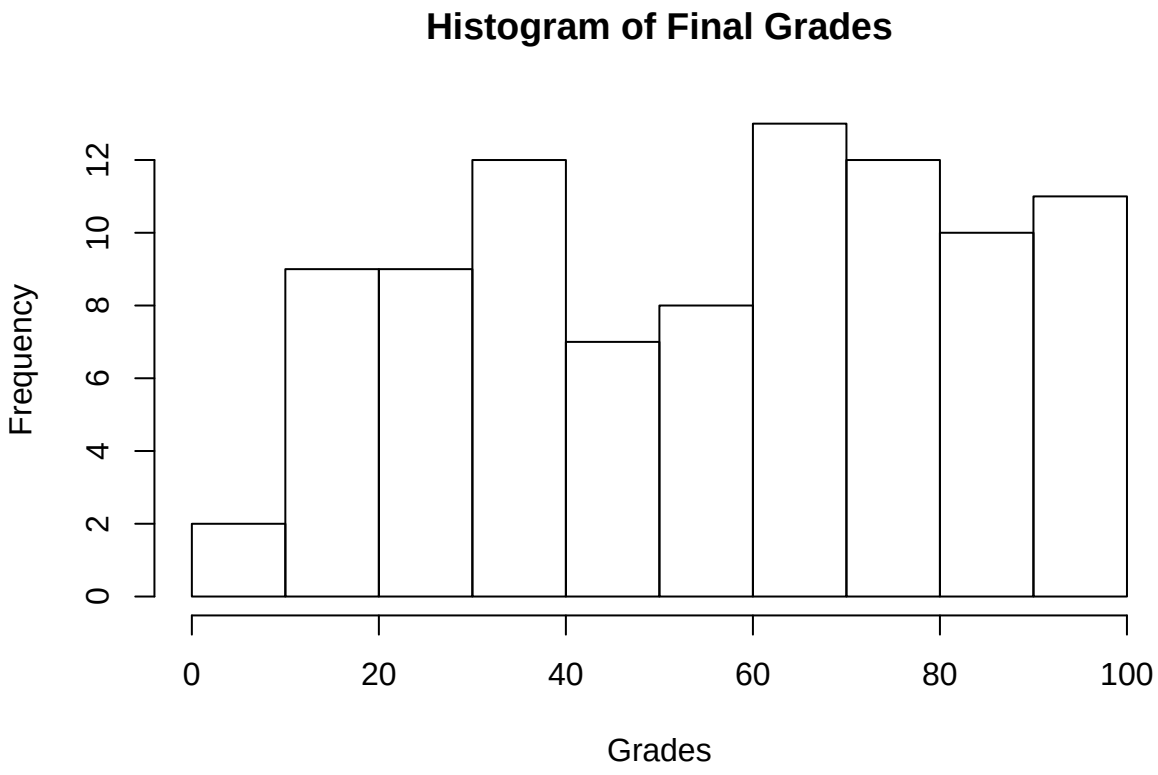
After checking the data, what we can now formulate what we may want to discover. From the grades data, we can see that there are students that have very low marks. What we want to ask is if this is due to the fact that they are just doing nothing or is doing other things in the computer. We would also like to know if more activity in the computer means higher grade. As such we would like to engineer these features, with grades as our response variable:

1. focus Is the student viewing pages related to the exercise? Here, we would just like to classify everything as related or non-related.
2. idle_time How much idle time was spent by the student?
3. time_finished How long did the student finish? We will find the difference of the start time and the end time.
4. activity level Did the student have more activity (keystrokes, mouse movements)? We will consolidate this into just one feature.

We create these features, but first, we would need to find the students with final grades only:

```
## change name from "Total (100 points)" to just "Total"
finalgrades <- rbind(finalgrades1, finalgrades2)
colnames(finalgrades)[18] <- "Total"
```

```r
finalgrades <- finalgrades %>% dplyr::select(Student.ID, Total) %>% aggregate(by = list(finalgrades$Stu
finalgrades <- finalgrades[,-1]
hist(finalgrades$Total, main = "Histogram of Final Grades", xlab = "Grades")
```



From here, we can see the almost uniform distribution of the grades from 10-100. We engineer the other variables, using only the student with grades:

```r
#helper function for normalization
normalize <- function(x) {
return ((x - min(x)) / (max(x) - min(x))) }

education_filtered_1 <- filter(education, student_Id %in% finalgrades$Student.ID)

education_filtered_2 <- education_filtered_1 %>%
  mutate(time_finished = as.numeric(education_filtered_1$end_time %>%
           as.POSIXct(format = "%m.%d.%Y %H:%M:%S") -
           education_filtered_1$start_time %>%
           as.POSIXct(format = "%m.%d.%Y %H:%M:%S"))) %>%
  mutate(unfocus = ifelse(as.character(education_filtered_1$activity) == " Other", 1, 0)) %>%
  mutate(activity_level =
           normalize(education_filtered_1$mouse_wheel) +
           normalize(education_filtered_1$mouse_wheel_click) +
           normalize(education_filtered_1$mouse_click_left) +
           normalize(education_filtered_1$mouse_click_right) +
           normalize(education_filtered_1$mouse_movement) +
           normalize(education_filtered_1$keystroke)) %>%
```

```
  na.omit()

education_filtered_3 <- education_filtered_2 %>%
  dplyr::select(student_Id, idle_time, time_finished, unfocus, activity_level) %>%
  aggregate(by = list(education_filtered_2$student_Id), FUN = mean)
```

From above, we engineered the new variables time_finished, unfocus and activity level.

We check the value of the our newly created engineered table:

```
education_filtered_3 <- education_filtered_3[,-1]
education_filtered_3 %>% head()
```

```
##   student_Id idle_time time_finished    unfocus activity_level
## 1          1  175628.7      12.32307 0.11747851    0.016171442
## 2          2  781660.9      17.71622 0.19425676    0.010835350
## 3          4  177586.5      10.52787 0.14177564    0.008750627
## 4          5  174650.6      10.64270 0.20756116    0.013489874
## 5          6  274752.0      12.76096 0.08062235    0.015003678
## 6          7  769966.9      17.50000 0.11961722    0.014677122
```

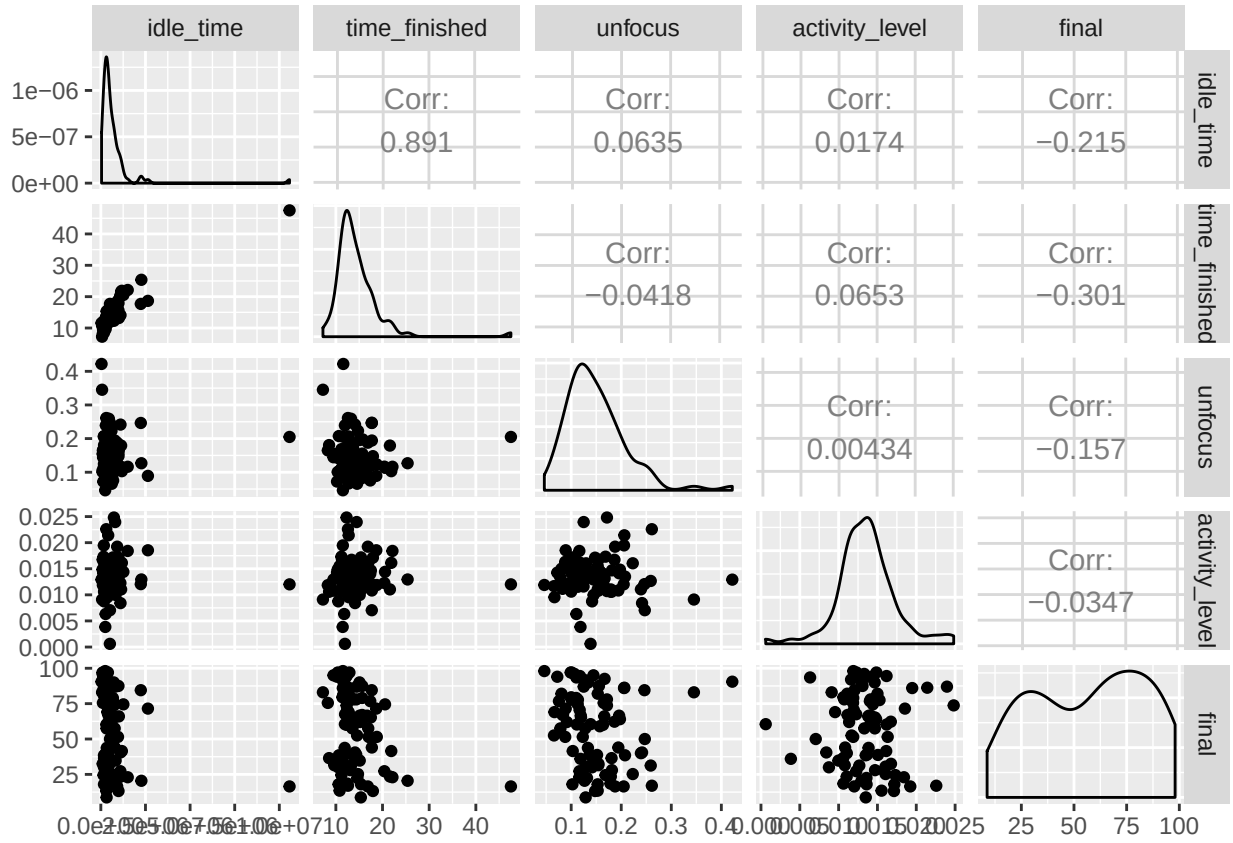
We join this with the grades table:

```
final_education_1 <- merge(education_filtered_3, finalgrades, by.x = "student_Id", by.y = "Student.ID")
final_education_1 <- final_education_1[,-1]
names(final_education_1)[5] <- 'final'
final_education_1 %>% head()
```

```
##   idle_time time_finished    unfocus activity_level final
## 1  175628.7      12.32307 0.11747851    0.016171442 94.50
## 2  781660.9      17.71622 0.19425676    0.010835350 44.00
## 3  177586.5      10.52787 0.14177564    0.008750627 30.00
## 4  174650.6      10.64270 0.20756116    0.013489874 38.50
## 5  274752.0      12.76096 0.08062235    0.015003678 82.00
## 6  769966.9      17.50000 0.11961722    0.014677122 65.25
```

We check the most correlated with 'final':

```
final_education_1 %>%
  ggpairs(progress=FALSE)
```



Above, we compare the correlation of final with all other engineered features. We can see strong correlation with idle_time and time_finished.

We split the data in a test and train split:

```
smp_size <- floor(0.75 * nrow(final_education_1))
set.seed(123)
train_ind <- sample(seq_len(nrow(final_education_1)), size = smp_size)

education_train <- final_education_1[train_ind, ]
education_test <- final_education_1[-train_ind, ]
```

We fit a linear model:

```
full_model <- lm(final ~ ., data=education_train)
full_model %>% summary
```

```
##
## Call:
## lm(formula = final ~ ., data = education_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.991 -20.198   4.128  23.593  54.260
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.055e+02  2.369e+01   4.452 3.93e-05 ***
```

```
## idle_time       7.600e-06  6.163e-06   1.233   0.2225
## time_finished  -3.242e+00  1.550e+00  -2.092   0.0409 *
## unfocus        -8.786e+01  5.684e+01  -1.546   0.1276
## activity_level  3.978e+02  8.346e+02   0.477   0.6354
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.92 on 58 degrees of freedom
## Multiple R-squared:  0.1378, Adjusted R-squared:  0.07835
## F-statistic: 2.318 on 4 and 58 DF,  p-value: 0.06778
```

```
model_coefs <- summary(full_model)$coeff[-1, 4]
significant_predictors <- model_coefs[model_coefs < 0.10]
significant_predictors
```

```
## time_finished
##     0.0408562
```

For now, it appears that only the time finished appears to be the predictor of the final grade.

We try backward elimination:

```
# Run backward elimination
initial_model <- lm(final ~ 1, data = education_train)
back_elim <- step(object = full_model, scope = list(lower = initial_model), direction = "backward")
```

```
## Start:  AIC=419.68
## final ~ idle_time + time_finished + unfocus + activity_level
##
##                  Df Sum of Sq   RSS    AIC
## - activity_level  1     164.6 42187 417.92
## - idle_time       1    1101.7 43124 419.31
## <none>                        42022 419.68
## - unfocus         1    1731.3 43753 420.22
## - time_finished   1    3169.9 45192 422.26
##
## Step:  AIC=417.92
## final ~ idle_time + time_finished + unfocus
##
##                 Df Sum of Sq   RSS    AIC
## - idle_time      1    1071.3 43258 417.50
## <none>                       42187 417.92
## - unfocus        1    1692.1 43879 418.40
## - time_finished  1    3109.7 45296 420.40
##
## Step:  AIC=417.5
## final ~ time_finished + unfocus
##
##                 Df Sum of Sq   RSS    AIC
## - unfocus        1    1361.0 44619 417.46
## <none>                       43258 417.50
## - time_finished  1    4007.5 47266 421.09
##
## Step:  AIC=417.46
## final ~ time_finished
##
```

```
##                 Df Sum of Sq   RSS    AIC
## <none>                       44619 417.46
## - time_finished  1    4119.9 48739 421.02
```

We try forward elimination:

```
# Run forward elimination
forward_elim <- step(object = initial_model, scope = list(upper = full_model), direction = "forward")
```

```
## Start:  AIC=421.02
## final ~ 1
##
##                   Df Sum of Sq   RSS    AIC
## + time_finished    1    4119.9 44619 417.46
## + idle_time        1    2235.5 46503 420.06
## <none>                         48739 421.02
## + unfocus          1    1473.4 47266 421.09
## + activity_level   1      70.0 48669 422.93
##
## Step:  AIC=417.46
## final ~ time_finished
##
##                   Df Sum of Sq   RSS    AIC
## <none>                         44619 417.46
## + unfocus          1   1361.01 43258 417.50
## + idle_time        1    740.15 43879 418.40
## + activity_level   1    106.27 44513 419.30
```

We try a stepwise elimination:

```
step_sel <- step(object = initial_model, scope = list(upper = full_model), direction = "both")
```

```
## Start:  AIC=421.02
## final ~ 1
##
##                   Df Sum of Sq   RSS    AIC
## + time_finished    1    4119.9 44619 417.46
## + idle_time        1    2235.5 46503 420.06
## <none>                         48739 421.02
## + unfocus          1    1473.4 47266 421.09
## + activity_level   1      70.0 48669 422.93
##
## Step:  AIC=417.46
## final ~ time_finished
##
##                   Df Sum of Sq   RSS    AIC
## <none>                         44619 417.46
## + unfocus          1    1361.0 43258 417.50
## + idle_time        1     740.2 43879 418.40
## + activity_level   1     106.3 44513 419.30
## - time_finished    1    4119.9 48739 421.02
```

Now, we evaluated the metrics:

```
fitstat <- function(x) {
  xsum <- summary(x)
  resid <- x$residuals
```

```
  fit <- x$fitted.values
  return(c(R2 = xsum$r.squared,
           R2Adj = xsum$adj.r.squared,
           AIC = AIC(x),
           BIC = BIC(x),
           MSE = mean(resid^2),
           MAPE = mean(abs(resid/fit))
  ))}

sapply(list('Full Model' = full_model,
            'Forward Elimination' = forward_elim,
            'Backward Elimination' = back_elim,
            'Stepwise Selection' = step_sel), fitstat)
```

```
##         Full Model Forward Elimination Backward Elimination Stepwise Selection
## R2      0.13781310          0.08453072           0.08453072         0.08453072
## R2Adj   0.07835193          0.06952303           0.06952303         0.06952303
## AIC   600.46367227        598.24145045         598.24145045       598.24145045
## BIC   613.32248063        604.67085463         604.67085463       604.67085463
## MSE   667.01740782        708.23848291         708.23848291       708.23848291
## MAPE    0.40295191          0.43954167           0.43954167         0.43954167
```

All models are the same:

```
back_elim
```

```
##
## Call:
## lm(formula = final ~ time_finished, data = education_train)
##
## Coefficients:
##   (Intercept)  time_finished
##         77.95          -1.51
```

It means that only the time finished affects the final grade. If a student passes their worksheet earlier, the higher their grade is.