Data Management and Exploratory Data Analysis Report of a Online Cyber Security Course 'CSC8631 Coursework (Semester 1, 2021)'

Samar Abdullah

30/11/2021

1 Business Understanding

This report is based on an inquiry into a Future Learn-hosted online course. Future Learn is an online learning platform that has worked with a number of world-renowned universities and organisations to provide a diverse set of courses. One of them is Newcastle University's Cyber Security, for which we will conduct a forensic examination in the form of a data analysis in this report.

1.1 Business Objectives

Because Future Learn is an educational website, they are primarily interested in anything that would improve the learning experience, increase student interaction, and encourage students to enroll in more online courses. As a result, we may claim that someone taking their course obtained skills or knowledge relating to the course they signed up for, and the course was given in a stimulating and engaging manner for the student. According to From Bricks to Clicks, the government is now addressing challenges such as "identifying at-risk students."

Despite the importance of the students' well-being, this report will look into how they interact with the course. We hope to use this method to assess the course's progress and determine where the course is prospering and where it is not. After we've addressed these difficulties, we'll be able to devise a strategy for improving the course as needed.

1.2 Assess the Situation

1.2.1 Sources of Data and Knowledge

Newcastle University contributed the data for this study, as they are presumed to have direct access to the online course data "CSC8631".

1.2.2 Data Assessment

WWe were supplied with a large data set from an online Cyber Security course over the course of seven runs; it is safe to claim that the course only ran seven times. The information supplied is divided into several csv files, which can be summarized as follows:

- Survey questions
  - Archetype - psychological qualities of users
  - Weekly sentimental
- feedback from students on the course
  - Departure
- Statistics
  - Registration
  - Physical Activity
  - Answers to Questions
  - (>run2) video
  - Members of the team (>run1)

The Cyber Security course information is presented in the form of videos and notes, which are broken down into sections, or chapters. Sub-sections are contained within these "steps," and this is what is referred to throughout the data set table titles. It is crucial to note, however, that the video data is only present after run two, and it is presumed that movies were supplied to the students in runs one and two, but that data

assimilation was not yet possible. After run one, there is also a Team member file that provides information on any roles that were assigned inside the course, such as mentors and course organisers.

From 2016 to 2018, Cyber Security was held seven times, each for three weeks - see information below. All of the information was gathered from the course pdf papers - see the data file in the Project Template folder for further information.

Table 1: Summary of start and end dates for each run

Run	StartDate	EndDate
1	05/09/2016	26/09/2016
2	20/03/2017	10/04/2017
3	18/09/2017	09/10/2017
4	13/11/2017	04/12/2017
5	05/02/2018	26/02/2018
6	11/06/2018	02/07/2018
7	10/09/2018	01/10/2018

1.2.3 Requirements

Reproducibility is a critical need for this project. All analysis will be done in R, specifically ProjectTemplate, reports will be compiled in RMarkDown, and Git version control will be used to ensure this. The project will last four weeks and will be completed on December 3th, 2021. In terms of the legality of the data, we assume that we have complete agreement from the data owner, which is granted through Newcastle University's CSC8631 - Data Management and Exploratory Data.

1.2.4 Assumptions

The data is thought to have come directly from the course's online database, so it's safe to presume it's accurate. Future Learn's competitors are also presumed to have had no direct impact on the data set because its major competitor, Udemy (2009), was created before Future Learn (2012). Economic considerations, likewise, are considered to have no bearing on the data's quality. The responses to the online questionnaire and the movie are not supposed to be mandatory or even graded. As a result, it's considered that they're mostly used as a learning tool.

1.3 Data Mining Goals

1.3.1 Goals

The data mining tries to address the business requirement of "raising student interaction with the course" by extrapolating trends in the quiz response data set and how they fluctuate with the number of times this course was run and throughout the course duration. The problem can be solved mostly, but not exclusively, using predictive linear regression. As a result, depending on how the data is presented, a range of data mining approaches are used. This is, after all, an exploratory data analysis.

1.3.2 Sucess Criteria

It would be acceptable to call data mining a success if a correlation between quiz question responses and interaction in the online course can be established, and how this develops throughout the course's number of runs and duration. For instance, how does participation in the course questions alter as the number of runs increases and the course progresses?

1.3 Data Mining Goals

The purpose of this data mining technique is to look at the data set surrounding the posed question of "identifying pupils who may need further support" and see whether there is any evidence to back this up in the data. Finding a correlation would be a successful conclusion of this investigation.

1.4 Project Plan

To achieve these objectives, a review of the quiz.responses files will be conducted, with the hope of identifying tendencies. Because the data to be analysed has already been provided, a significant amount of data mining time would be saved. As a result, it is expected that the majority of the project's time and effort would be spent learning about data preparation, such as data cleaning and reformatting. This has been categorised as a proportion of project time in the table below.

1) Business understanding - 5%
2) Data mining - 10%
3) Data preparation - 60%
4) Modeling - 20%
5) Evaluation - 10%
6) Deployment - 5%

Stages 3 and 4 of the project will be iteratively repeated, with each iteration making empirical judgments based on the preceding model's outcomes. The majority of the project's lifespan is planned to be spent in the reformatting and modelling stages, which will involve multiple iterations. Depending on the outcomes, this process will be repeated a finite number of times, with each cycle adding to the prior findings. Similarly, if the results are exhausted, the project will explore tackling a new data mining problem.

2 Data Understannding

TThe data for the project is stored in multiple files, the most important of which is the quiz.response file, which is created for each course run. Run one's quiz replies to display the head of the data frame.

```
head(cyber.security.1_question.response)
```

```
## # A tibble: 6 x 10
##   learner_id quiz_question question_type week_number step_number question_number
##   <chr>      <chr>         <chr>               <int>       <int>           <int>
## 1 77454a73-~ 1.7.1         MultipleChoi~           1           7               1
## 2 77454a73-~ 1.7.1         MultipleChoi~           1           7               1
## 3 a4fa6f89-~ 1.7.1         MultipleChoi~           1           7               1
## 4 a4fa6f89-~ 1.7.1         MultipleChoi~           1           7               1
## 5 a4fa6f89-~ 1.7.1         MultipleChoi~           1           7               1
## 6 f27eec8c-~ 1.7.1         MultipleChoi~           1           7               1
## # ... with 4 more variables: response <chr>, cloze_response <lgl>,
## #   submitted_at <chr>, correct <chr>
```

There are a few things to note regarding this data collection. To begin, all of the questions are multiple choice, and the column 'quiz question' is a combination of the columns 'week number,"step number,' and'quiz number.' Second, because 'cloze response' has no data, it can be ruled out. Finally, each student's attempt is recorded, including the time and response that the student picked, as well as whether or not they correctly answered the question.

These can easily be negated because all of the questions are multiple choice and the 'cloze reposne' field is empty. When it comes to the quiz question fields, the 'quiz question' field can be ignored because the other elements give the information in a more code-friendly way, allowing us to pick and choose which ones are relevant during the modelling step.

It's also vital to maintain track of which run the data came from when reformatting quiz.responses. Furthermore, it is critical to maintain the complete user id, or question number, when reformatting the quiz data, so that if any further merging is required, they are consistent with the rest of the huge data set.

There are references to team members from run two onward.

```
head(cyber.security.2_team.members)
```

```
## # A tibble: 6 x 5
##   id                                   first_name last_name team_role  user_role
##   <chr>                                <chr>      <chr>     <chr>      <chr>
## 1 f27eec8c-eaf1-4e6a-90f0-d6d5b653285d FIRST      LAST      host       organisa~
## 2 77454a73-6b8b-46a2-8dee-35f36b6c4fc1 FIRST      LAST      host       organisa~
## 3 a4fa6f89-a596-4d00-9397-420a348c398d FIRST      LAST      lead_educ~ organisa~
## 4 21d74c76-2b0d-4dfd-a252-f6dcf2100874 FIRST      LAST      educator   learner
## 5 3e58d103-57b3-4d46-ac62-69a8b30b7835 FIRST      LAST      educator   learner
## 6 85ea97bb-17d6-4bf7-ad74-dfb91f45aeb6 FIRST      LAST      educator   learner
```

These are essentially hierarchical roles in the course, and while they may have an impact on quiz responses (i.e., the admin testing the operation of the course questions), the weighting on the total results is insignificant. As a result, data frames for teams were omitted.

The 'video.stats', enrollment, and'step.activity' files were not included in this study because the data mining goals only referenced to the quiz questions.

2.1 Data Quality

The data appears to be of acceptable quality, and all of the fields appear to have consistent formatting, such as capitalization and spacing. There are a few rows, however, where data is lacking, such as no student id. However, in data preparation, these fields can simply be ignored.

```
sum(cyber.security.1_question.response$learner_id == "")
```

```
## [1] 401
```

As can be seen, there are 401 empty fields; nonetheless, this represents only 0.5 percent of the total rows.

```
sum(cyber.security.1_question.response$learner_id == "" )/
  length(cyber.security.1_question.response$learner_id )*
  100
```

```
## [1] 0.5207657
```

Because of the way the data is displayed, finding anomalies in the data is rather challenging. However, after reformatting the data, some differences may become more obvious.

```
summary(cyber.security.1_question.response)
```

```
##   learner_id        quiz_question      question_type       week_number
##  Length:77002       Length:77002       Length:77002       Min.   :1.000
##  Class :character   Class :character   Class :character   1st Qu.:1.000
##  Mode  :character   Mode  :character   Mode  :character   Median :2.000
##                                                           Mean   :2.085
##                                                           3rd Qu.:3.000
##                                                           Max.   :3.000
##   step_number    question_number   response         cloze_response
##  Min.   : 7.00   Min.   :1.000   Length:77002       Mode:logical
##  1st Qu.: 7.00   1st Qu.:2.000   Class :character   NA's:77002
##  Median : 8.00   Median :3.000   Mode  :character
##  Mean   :11.57   Mean   :3.572
##  3rd Qu.:18.00   3rd Qu.:5.000
##  Max.   :19.00   Max.   :9.000
##  submitted_at         correct
##  Length:77002       Length:77002
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

3 Data Preperation

3.1 Select Data

A resultant data frame was sought for the data preparation that could be utilised to answer several permutations of the data mining inquiry. 'quiz question' and 'cloze response' had to be deleted first when referring to the quiz.response csv file. Furthermore, the date was formatted as 'YYYY-MM-DD HH:MM:SS UTC'. We are only interested in the time since the start of the course for this application, and thus it was easier to display the time in seconds because it had to be standardised around the start of the course. As a result, when all runs are referred to as a single data frame, they are all harmonised around 0.

3.1.1 Pre-processing

The data was not displayed in such a way that any actual analysis could be done to assess its importance or association at this level of the data selection. As a result, the data was tampered with. 'cleanQuizData()' and 'quizDataClean()' were used to do this. First, there's 'cleanQuizData(),' which takes two arguments: the quiz data frame and the course start date. The lengthy and inconvenient heading was shortened, resulting in less error-prone coding and more challenging representation when plotting data frames. Finally, the date was added up and converted to seconds, which was a more comprehensible format. The dates were standardised, however, because all of the course runs began at different times. The function was performed as part of the munging section's pre-processing, with all of the resulting data frames being saved in the cache. The learner id was excessively long (but required), therefore it was maintained for cross-referencing purposes. This function produced 7 data frames in the following format as a consequence of the cleaning.

```
quizStu <- function(quiz, courseStartDate){
```

```
  #convert the course start date to seconds
  cs = as.numeric(as.POSIXct(courseStartDate ))

  #renaming the df and its columns
  colnames(quiz) = c("id", "qq", "qt", "wn", "sn", "qn", "r", "cr", "t", "ans")

  #removing the columns that are not needed, or give no info
  quiz = select(quiz, -c(qt, cr))

  #displaying the date in seconds and removing substituting the date in which the course started
  quiz$t = as.numeric(as.POSIXct(quiz$t))-cs

  return(quiz)
}
```

Table 2: First two rows of quizStat1 data frame

id	qq	wn	sn	qn	r	t	ans
77454a73-6b8b-46a2-8dee-35f36b6c4fc1	1.7.1	1	7	1	1,2	-5232175	false
77454a73-6b8b-46a2-8dee-35f36b6c4fc1	1.7.1	1	7	1	1,2,3	-5230975	true

The data frame headings have been made more manageable, as seen in the above data frame.

- id: the students id
- qq: quiz question
- wn: week number
- sn: section number
- r: responses
- t: time relative to the start of the course
- ans: was the response correct

It's worth noting that the time is negative because the quiz questions were answered before the course ever began.

The second function, 'quizDataClean(),' did the heavy lifting, reformatting and altering the data into a manner that allowed some correlations to be seen. This function output a data frame with the student id as the reference, using the resultant data frame from 'cleanQuizData()'. The function returned, for example, the number of responses given by a student ('numAns'); the number of distinct questions replied by the student ('numQues'); the number of accurate answers given ('numCorr'); and the time of the first and last question answered ('st' and 'ft', respectively).

```
quizStuPre <- function(quizStat){

  #create a data frame with unique user id
  quizData <- data.frame(id = unique(quizStat$id),
                         numAns="",
                         numQues="",
                         numCorr="",
                         ft="",
                         st="")
```

```
  for(i in 1:nrow(quizData)){
    count = 0 #count the number of occurrences (i.e. question attempts)
    count2 = 0 #reset the number of correct answers
    count3 = 0 # resets the number of different questions answered
    question = ""
    flag = 1

    #loops the number unique id values
    for(j in 1:nrow(quizStat)){

      if(quizData$id[i] == quizStat$id[j]){
        count = count+1
        if(flag == 1){
          quizData$st[i] = quizStat$t[j] #store the FIRST time student answered question
          flag = 0
        }
        if(quizStat$ans[j] == "true"){
          count2 = count2+1
        }
        if(quizStat$qq[j] != question){
          question = quizStat$qq[j]
          count3 = count3+1
        }
      }
    }
    quizData$numQues[i] = count3#store the number of different questions answered
    quizData$numCorr[i] = count2 # store the number of correct answers
    quizData$numAns[i] = count #store the number of attempts
    quizData$ft[i] = quizStat$t[count] #store the LAST time student answered question

  }

  return(quizData)
}
```

This function is part of the pre-processing and is stored in the munge subdirectory due to its high computing requirements. The first two columns of the resulting 'quiz.response.1' are displayed below.

Table 3: First two rows of quizStuPre1 data frame

id	numAns	numQues	numCorr	ft	st
77454a73-6b8b-46a2-8dee-35f36b6c4fc1	39	17	18	7831	-5232175
a4fa6f89-a596-4d00-9397-420a348c398d	40	19	19	7859	-4805410

The above df correspond to the following;

- id: the students id
- numAns: the total number attempts at all questions
- numQues: the total number of questions answered
- numCorr: the number of correct answers provided
- ft: final time a question was answered
- st: first time a question was answered

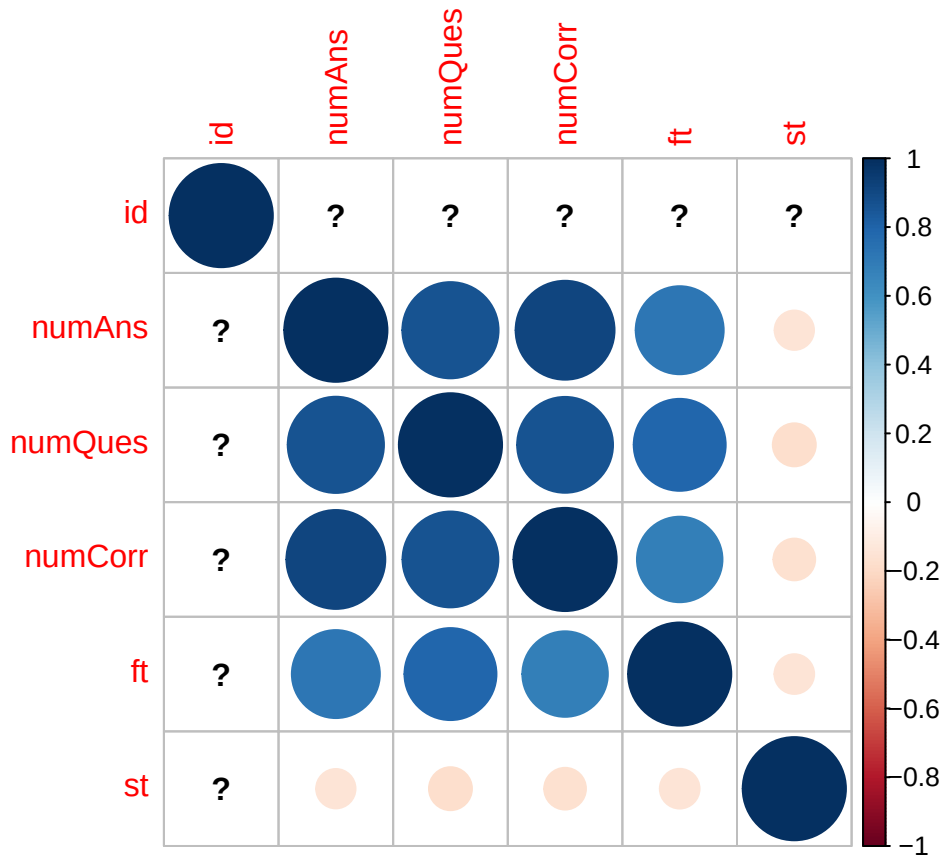
3.1.2 Relationships

For run one, a correlation matrix was constructed to acquire a better understanding of the data (excluded the student id).

```
  #quizStuPre1 <- select(quizStuPre1, -c(id))
  quizStuPre1 <- as.data.frame(sapply(quizStuPre1, as.numeric))
```

```
## Warning in lapply(X = X, FUN = FUN, ...): NAs introduced by coercion
```

```
  corrplot(cor(quizStuPre1), method="circle")
```



There are some strong correlations in this figure, particularly between 'numCorr' and 'numAns,' and we can also observe that all of the start times ('st') have a negative correlation. The strongest associations are listed below, in order of strength.

1) Correct Answers vs. Question Attempts (0.92)
2) Question Answers vs. Question Attempts (0.87)
3) Answers to questions vs. correct answers (0.86)

3.2 Clean Data

```
summary(quizStuPre1)
```

```
##        id           numAns          numQues         numCorr
##  Min.   : NA    Min.   :  1.00   Min.   : 1.00   Min.   :  0.00
##  1st Qu.: NA    1st Qu.:  9.00   1st Qu.: 6.00   1st Qu.:  6.00
##  Median : NA    Median : 17.00   Median :10.00   Median : 10.00
##  Mean   :NaN    Mean   : 22.58   Mean   :13.36   Mean   : 12.37
##  3rd Qu.: NA    3rd Qu.: 36.00   3rd Qu.:22.00   3rd Qu.: 20.00
##  Max.   : NA    Max.   :401.00   Max.   :22.00   Max.   :236.00
##  NA's   :3410
##        ft                 st
##  Min.   :-5232175   Min.   :-5232175
##  1st Qu.:-2795990   1st Qu.:  112605
##  Median :    3204   Median :  306961
##  Mean   :-1106865   Mean   :  523251
##  3rd Qu.:    7287   3rd Qu.:  724466
##  Max.   :   44231   Max.   : 2893905
##
```

The nosie has to be removed in order to acquire a full grasp of the paired correlations that our created data exhibited in a scatter plot matrix. When graphing the summary - as shown in 'numCorr' and 'numAns' - this became clear. To reduce noise in the data, these abnormalities were removed using the following code, where more than 75 attempts were considered abnormal, and since there were only 22 questions, this was reduced to the maximum number of attempts.

```
quizStuPre1 <- quizStuPre1[!(quizStuPre1$numAns > 75), ]
quizStuPre1 <- quizStuPre1[!(quizStuPre1$numCorr > 22), ]
```

Note:

- Because there were only 22 questions, some students may have answered the same question multiple times).
- To maintain consistency, the same arguments must be used on the other runs.

Now that the summary has been run again, the results appear to be much cleaner.

```
summary(quizStuPre1)
```

```
##        id           numAns          numQues         numCorr
##  Min.   : NA    Min.   : 1.00   Min.   : 1.00   Min.   : 0.00
##  1st Qu.: NA    1st Qu.: 9.00   1st Qu.: 6.00   1st Qu.: 6.00
##  Median : NA    Median :17.00   Median :10.00   Median :10.00
##  Mean   :NaN    Mean   :22.44   Mean   :13.33   Mean   :12.27
##  3rd Qu.: NA    3rd Qu.:36.00   3rd Qu.:22.00   3rd Qu.:20.00
##  Max.   : NA    Max.   :67.00   Max.   :22.00   Max.   :22.00
##  NA's   :3398
##        ft                 st
##  Min.   :-5232175   Min.   :-5232175
##  1st Qu.:-2795990   1st Qu.:  112605
##  Median :    3204   Median :  307068
```

```
##  Mean   :-1110810   Mean   :  523730
##  3rd Qu.:    7287   3rd Qu.:  725191
##  Max.   :   14975   Max.   : 2893905
##
```

3.3 Construct Data

The following derivations were compiled from the results of the previous stage and may be useful in an analysis. This was made up of

- dt: the amount of time that has passed between the first and last question
- acc: the proportion of correct to incorrect answers supplied
- scr: proportion of right responses to various problems
- tot: total number of questions answered

All of the following fields were added using the quizStuCon() function, and the resulting data frames were given the name 'quizStuConX,' with the 'X' referring to the ratio of 1:7.

```
quizStuCon <- function(quizData){
  quizData <- data.frame(quizData,
                tot = (as.numeric(quizData$numQues) / max(as.numeric(quizData$numQues))),
                dt <- Mod((as.numeric(quizData$ft) - as.numeric(quizData$st))),
                acc <- (as.numeric(quizData$numCorr)/as.numeric(quizData$numAns)),
                scr <- (as.numeric(quizData$numQues)/as.numeric(quizData$numAns))
                )
  return(quizData)
}
```

Furthermore, the data frame contained 'char' values that needed to be converted to 'nums' variables, which was done using the 'dfToNum()' function.

```
  #converts the df variables to a num other than id
  dfToNum <- function(data){
    df <- data
    df <- select(df, -c(id))
    df <- as.data.frame(sapply(df, as.numeric))
    df <- data.frame(data$id, df )
    return(df)
  }
```

Because the time fields' values were already scaled around 0 and not the actual data, it made logical to scale them all. This took away the complication of the larger, more difficult-to-understand figures. This was done again for all 7 'quizStuConX' df's.

```
  quizStuCon1$dt <- scale(quizStuCon1$dt)
  quizStuCon1$ft <- scale(quizStuCon1$ft)
  quizStuCon1$st <- scale(quizStuCon1$st)
```

The resultant data frame for run one, i.e. 'quizStuCon1', is shown below.

3.4 Interrogate Data

After all of the necessary changes to the quiz response df's, the runs needed to be merged into a single data frame that could be used to model the data. Weak correlations in the start times'st' were discovered in the scatter plot matrix, thus they were deleted. We also eliminated the student'id' because we no longer need to refer to it. Furthermore, it was critical to keep the run from which the data originated when merging the df's, which was accomplished as follows.

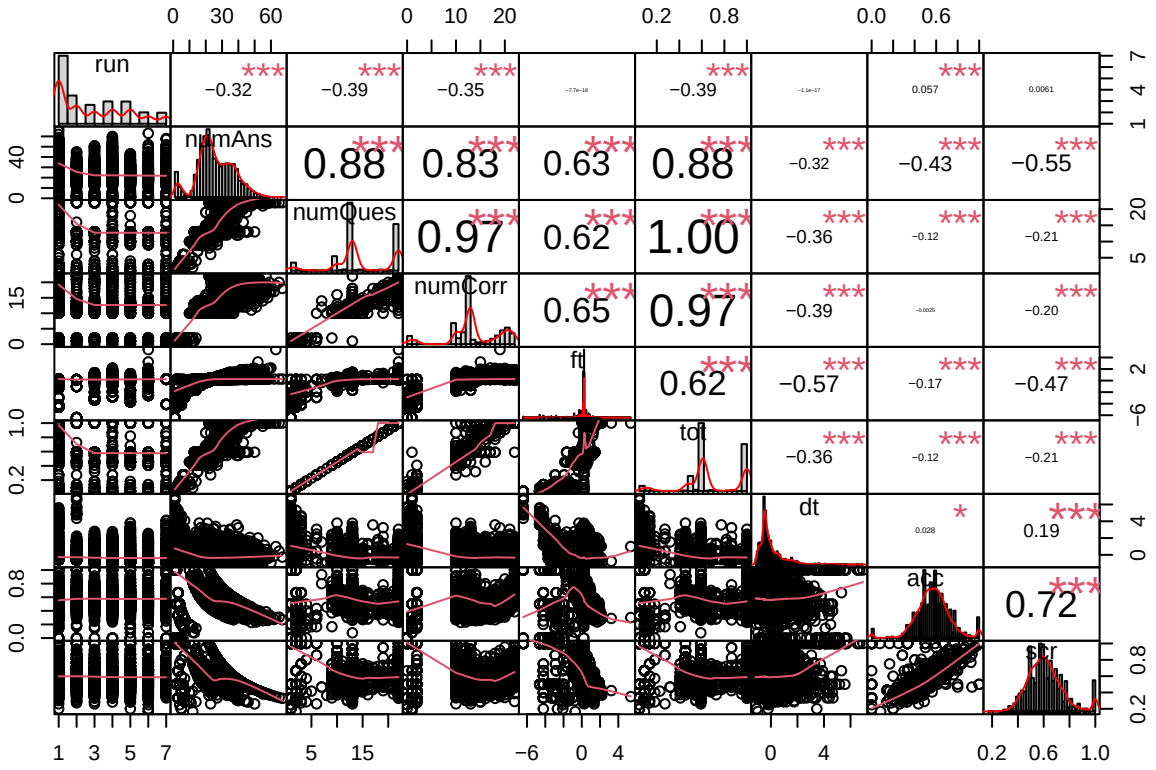
```
#Merge all quiz data df's

df1 <- data.frame(run=1, select(quizStuCon1, -c(st, id)))
df2 <- data.frame(run=2, select(quizStuCon2, -c(st, id)))
df3 <- data.frame(run=3, select(quizStuCon3, -c(st, id)))
df4 <- data.frame(run=4, select(quizStuCon4, -c(st, id)))
df5 <- data.frame(run=5, select(quizStuCon5, -c(st, id)))
df6 <- data.frame(run=6, select(quizStuCon6, -c(st, id)))
df7 <- data.frame(run=7, select(quizStuCon7, -c(st, id)))

quizStuMod <- rbind( df1,  df2, df3, df4, df5, df6, df7)
```

The final pairs plot revealed various fields that appeared to have linear connections, with the 'tot' appearing to have the most.

```
  chart.Correlation(quizStuMod, histogram=TRUE, pch=19)
```



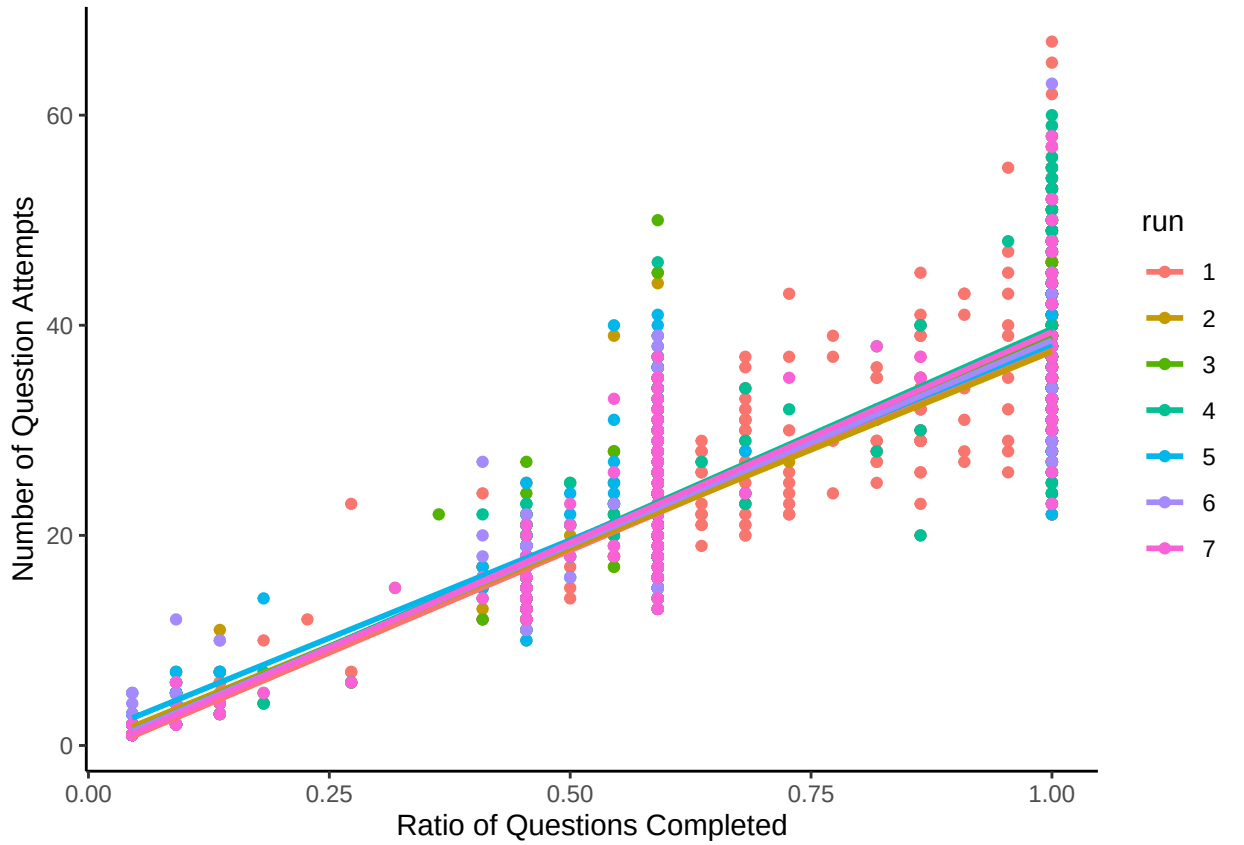
4 Modeling

At this stage of the modelling process, linear regression and a density plot will be used.

The data mining purpose was to see how the quiz question interaction fluctuates across the course's number of runs. Because there was a substantial association between the overall percentage of course questions finished and the number of right answers provided, it would be fascinating to examine how these matched up throughout all seven runs. The resulting fields were shown in a scatter plot using a linear correlation line.

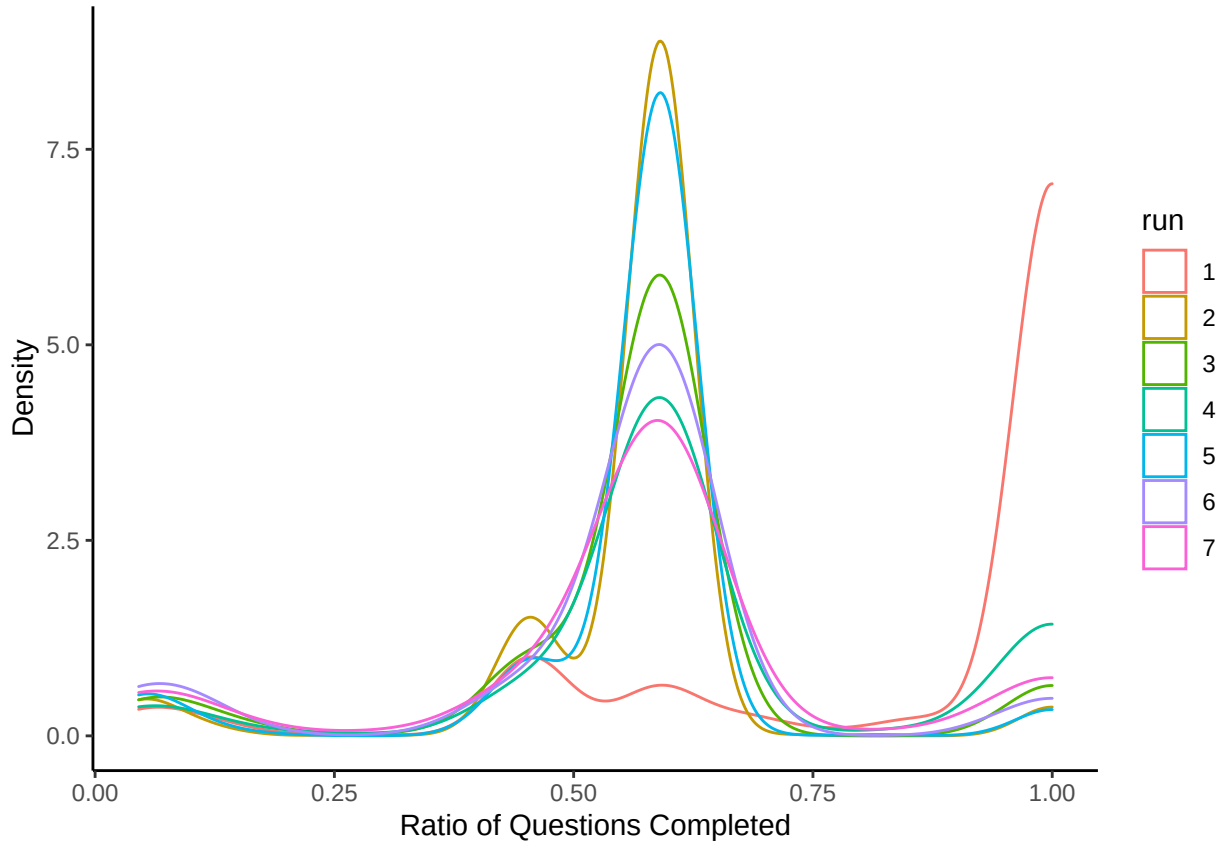
```
#plot a scatter plot with linear line of correlation between runs
ggplot(quizStuMod, aes(x = tot, y = numAns, col = factor(run))) +
  geom_point() +
  stat_smooth(method = "lm", se=F) +
  labs(x = "Ratio of Questions Completed",
       y = "Number of Question Attempts",
       color = "run" ) +
  theme_classic()
```

```
## `geom_smooth()` using formula 'y ~ x'
```



The percentage of course questions completed and the number of responses given have a strong linear relationship. As a result, we can claim that the question mistake rate remains constant throughout, and we can deduce from the correlation graph that the percentage of the course finished grows as the number of correct answers and attempts increases. However, people who finish more than 70% of the course are more likely to answer 100% of the questions. Maybe they have a stake in it? However, what is the distribution of total course questions completed by students in terms of density?

```
ggplot(quizStuMod, aes(x=tot, color=factor(run))) +
geom_density() +
labs(x = "Ratio of Questions Completed",
     y = "Density",
     color = "run" ) +
theme_classic()
```



The statistics are astounding; aside from the run one, there are very few students who are completing the course, with the majority of dropouts appearing to be at the same percentage of questions finished year after year. There is some catalyst event at this time that is causing people to drop out of the course, especially the larger surge that occurs after students have finished more than half of the course. For the number of times this course was offered, the following table shows the proportion of students who completed more than 75% of the questions.

Table 4: Summary of start and end dates for each run

Run	Percentage
1	74.91
2	3.20
3	7.84
4	21.68
5	3.41
6	6.98
7	13.07

This demonstrates that it varies somewhat arbitrarily.

5 Data Mining

5.1 Data Mining Goals

To figure out why the majority of students were only completing 55 percent to 60 percent of the course questions, the data had to be manipulated again to determine if the lack of involvement was due to a specific question or week in the course. Rather than using the student id as the subject, the question number was used this time.

6 Data Preperation

Please refer to the 'PrePro2.R' file in the scr folder within the project files for all data preparation code.

6.1 Pre-Processing

Most of the pre-processing was previously done because we were using the same data sets as before. The generated df from 'quizStat()' was passed to the 'quizQuePre()' function as an argument.

```
quizQuePre <- function(quizStat){

  #create a dataframe with unique user qq
  quizData <- data.frame(qq = unique(quizStat$qq),
                         numAns="",
                         numStu="",
                         numCorr="",
                         wn="",
                         sn="",
                         qn="")

  for(i in 1:nrow(quizData)){
    count = 0 #count the number of occurrences (i.e. question attempts)
    count2 = 0 #reset the number of correct answers
    count3 = 0 # resets the number of different questions answered
    student = ""
    flag = 1

    #loops the number unique qq values
    for(j in 1:nrow(quizStat)){

      if(quizData$qq[i] == quizStat$qq[j]){
        count = count+1
        if(flag == 1){
          quizData$wn[i] = quizStat$wn[j] #store the FIRST time student answered question
          quizData$sn[i] = quizStat$sn[j]
          quizData$qn[i] = quizStat$qn[j]
          flag = 0
        }
        if(quizStat$ans[j] == "true"){
```

```
          count2 = count2+1
        }
        if(quizStat$id[j] != student){
          student = quizStat$id[j]
          count3 = count3+1
        }
      }
    }
    quizData$numStu[i] = count3#store the number of different questions answered
    quizData$numCorr[i] = count2 # store the number of correct answers
    quizData$numAns[i] = count #store the number of attempts


  }

  return(quizData)
}
```

This function's output for run one is shown below. All pre-processing was saved in cache once more.

Table 5: First two rows of quizQuePre1 data frame

qq	numAns	numStu	numCorr	wn	sn	qn
1.7.1	5019	3443	3172	1	7	1
1.7.2	4468	3316	3167	1	7	2

The headings of the Data Frame (df) above correspond to the following:

- qq : question
- numAns : the total number attempts for the question

- numStu : the total number of students that answered the question
- numCorr : the number of correct answers provided for the question

- wn : week number
- sn : section number
- qn : question number

6.2 Construct Data

We used the 'quizQueCon()' function to calculate the ratio of students who completed the quiz, the accuracy of the answers for each question, and the ratio of students to answers, which we then added to the data frame.

```
quizQueCon <- function(quizData) {
  quizData <- data.frame(quizData,
                  tot = (as.numeric(quizData$numStu) / max(as.numeric(quizData$numStu))),
                  acc = (as.numeric(quizData$numCorr)/as.numeric(quizData$numAns)),
                  scr = (as.numeric(quizData$numStu)/as.numeric(quizData$numAns))
                  )
  return(quizData)
}
```

The resulting df was then processed, turning the numeric values from 'char' to 'num' variables - practically all columns except the quiz questions, which were 'qq' variables.

```
#converts the df variables to a num other than id
dfToNum <- function(data){
  df <- data
  df <- select(df, -c(qq))
  df <- as.data.frame(sapply(df, as.numeric))
  df <- data.frame(qq=data$qq, df )
  return(df)
}
```

Because the number of students in each run of the course varied greatly, the number of students was normalised for each run.

```
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}

quizQueCon1$numStu <- normalize(quizQueCon1$numStu)
```

6.3 Interrogating Data

All seven runs were integrated into a single function, with special attention paid to the run from whence they originated.

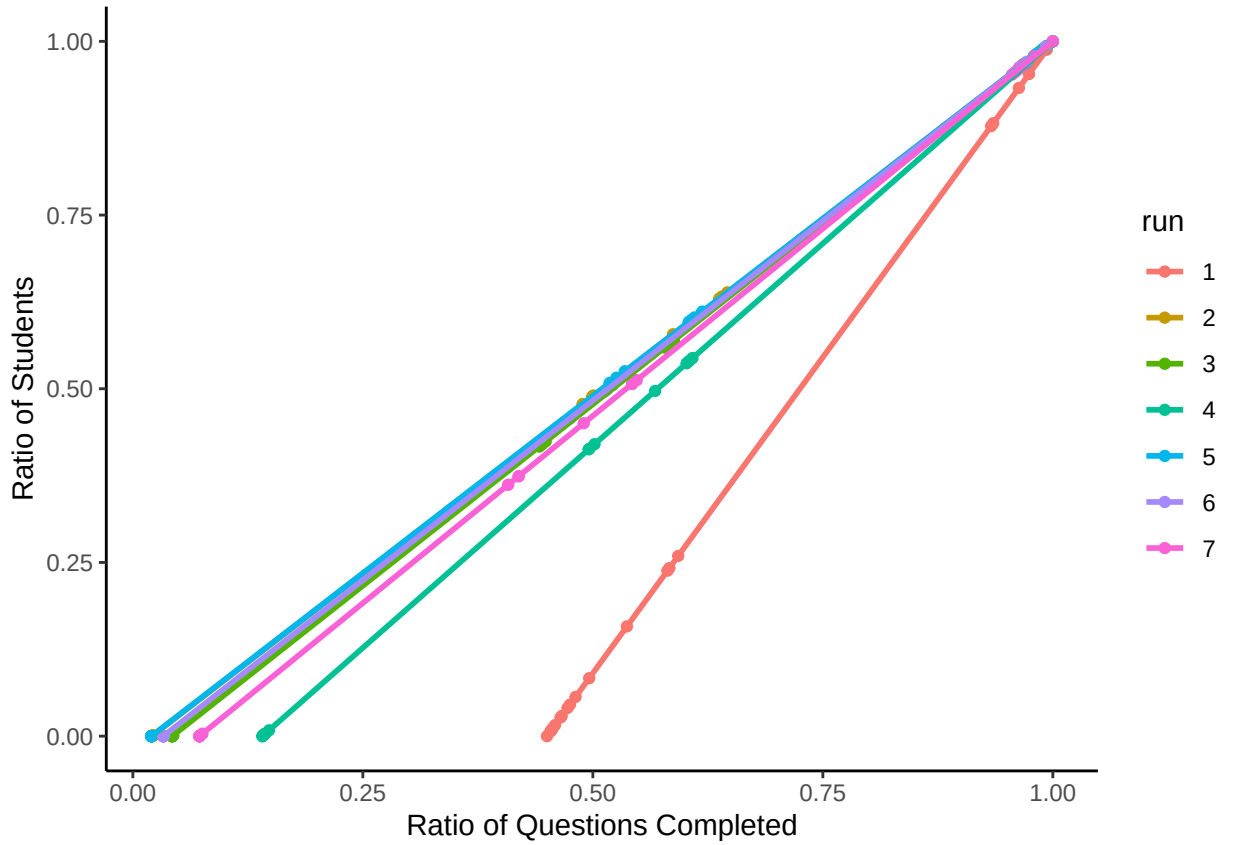
```
# run one was ignored as the section question varied from the other
df1 <- data.frame(run=1, quizQueCon1)
df2 <- data.frame(run=2, quizQueCon2)
df3 <- data.frame(run=3, quizQueCon3)
df4 <- data.frame(run=4, quizQueCon4)
df5 <- data.frame(run=5, quizQueCon5)
df6 <- data.frame(run=6, quizQueCon6)
df7 <- data.frame(run=7, quizQueCon7)
quizQueMod <- rbind( df1, df2,df3, df4, df5, df6, df7)
```

7 Modeling

Returning to the data mining question, it would be fascinating to know how the total ratio of quiz questions compared to the number of students over the 7 runs.

```
ggplot(quizQueMod, aes(x = tot, y = numStu, col = factor(run))) +
    geom_point() +
    stat_smooth(method = "lm", se=F) +
    labs(x="Ratio of Questions Completed",
         y="Ratio of Students",
         col="run") +
    theme_classic()
```

```
## `geom_smooth()` using formula 'y ~ x'
```



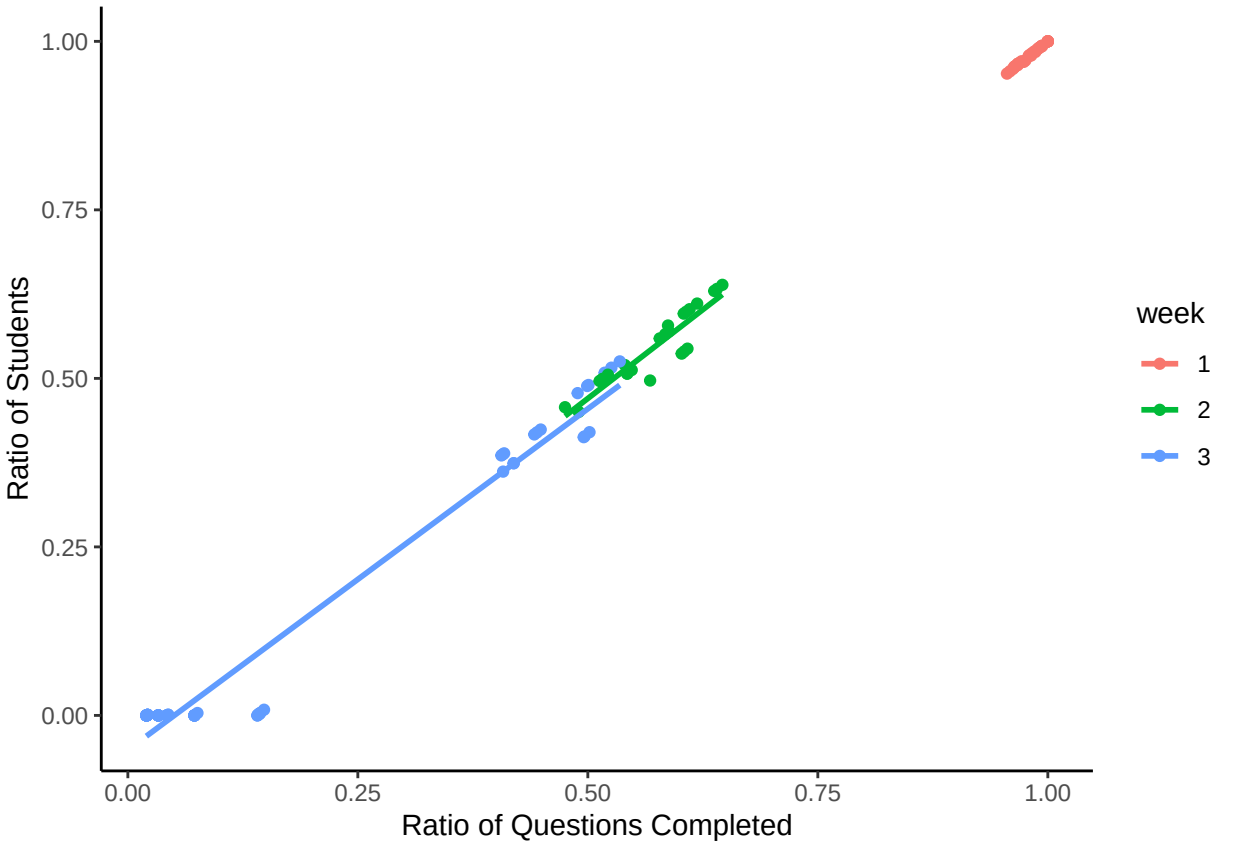
When we look at run one, the distribution is radically different, and it's actually not that bad, with a minimum of 45 percent of online quiz questions completed. The other six runs, on the other hand, are not in this category. Run 1 was removed from the df to gain a better understanding of why the other runs were not as effective, and the new df is dubbed 'quizQueMod1'.

```
# excluding run 1 from the model
quizQueMod1 <- rbind(df2,df3, df4, df5, df6, df7)
```

When this is plotted against the week number, a clear, virtually clusterable distribution emerges as the weeks go. In fact, these figures are rather consistent over the runs.

```
#scatter plot with linear regression
ggplot(quizQueMod1, aes(x = tot, y = numStu, col = factor(wn))) +
    geom_point() +
    stat_smooth(method = "lm", se=F) +
    labs(x="Ratio of Questions Completed", y="Ratio of Students", col= "week") +
    theme_classic()
```

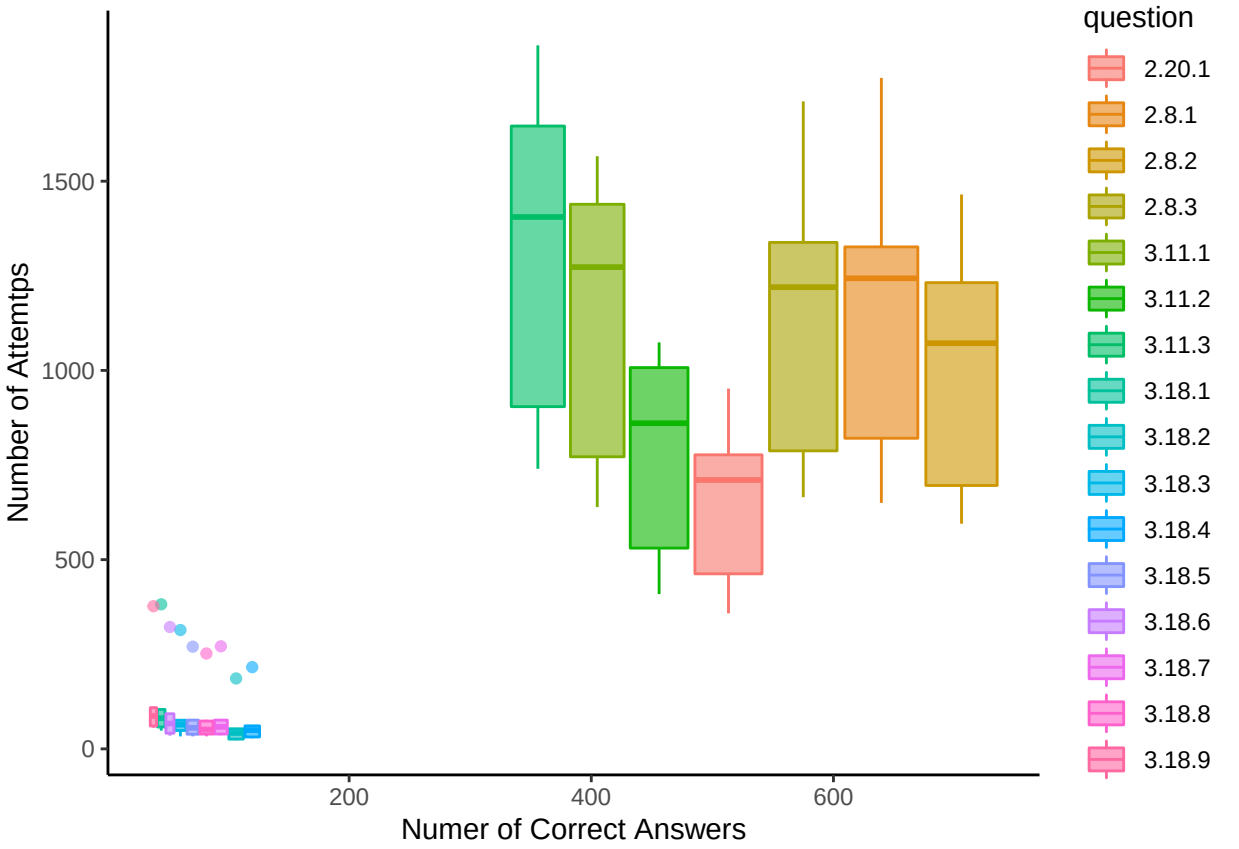
```
## `geom_smooth()` using formula 'y ~ x'
```



Further investigation has revealed that the first week should be ignored because nearly all of the questions are answered in week one. However, in week two, this decreases considerably, and in week three, there is a distinct difference, resulting in two clusters. What does a box plot of the completed questions look like for weeks two and three?

```
 #removing week one
quizQueMod2 <-quizQueMod1[(!quizQueMod1$wn == 1), ]

# Box plot
ggplot(quizQueMod2, aes(y=numAns, x=numCorr, color=factor(qq), fill=factor(qq))) +
   geom_boxplot(alpha=0.6) +
   labs(x="Numer of Correct Answers", y="Number of Attemtps", color="question", fill="question") +
   theme_classic()
```



When looking at the amount of questions a student responds during week two, there appears to be a minor rise in attempts to answer the question correctly as the content progresses, with fewer students attempting the final question of the week. In week three, the problem seemed to worsen, with much fewer pupils tackling questions four and upwards. Week three, on the other hand, has much more questions than week two.

8 Evaluation

8.1 Evaluate Results

The data mining aims were somewhat adjusted in two versions of CRISP DM as a result of the results. The first examines the quiz.response data in terms of the student and measures numerous factors related to the student's engagement with the online quiz question, while the second examined the identical set of data but this time with the quiz question as the subject. The findings of run one revealed several linear correlations between the data sets, demonstrating consistency throughout the course. We can conclude that the more questions a student answers, the more attempts he or she will have, which is to be expected. What's more noteworthy is that, except from run one, graphing the density of the percentage of the course finished revealed that the bulk of students were dropping out after 50-70 percent. The outcome, however, did not demonstrate why run one was so much more successful. The second run attempted to answer this question by looking for patterns in the answers to the previous questions. When students were separated into their runs, the data revealed a linear relationship between the number of students and the percentage of the course they completed. Run one was eliminated since it did not represent the rest of the runs. By doing so, and utilizing the week number as a factor, it was discovered that practically all students in week one completed the quiz question, week two had a 50-60 percent completion rate, and week three had two clusters of 40-55 percent and 0-15 percent completion rate. We can estimate the number of students who will complete the quizzes based on the trend from week one. Then, by graphing the questions, it was possible to see a pattern

that indicated the question was becoming more difficult throughout the week. This appeared to escalate to the point that kids simply refused to answer the questions. However, these findings could also indicate that students were not finishing the last sections of each week's course.

In terms of the business goal, we can say that as the course progresses, the amount of time students spend interacting with the course material declines linearly, and that for this course, most students will likely cease taking online quizzes between weeks two and three. This could be because the student is losing interest in the course or because the questions are becoming more difficult each week. However, there is a considerable increase in quiz questions in week three, which may discourage students from engaging with the course; however, this did not appear to have an impact on one's achievement. However, we can forecast what kind of interaction we'll have in week two by looking at the linear distribution in week one. As a result, if the number of questions answered does not lead to the desired outcome by the conclusion of the course, a plan of action that raises the students' awareness or importance of the quiz questions could be implemented. Finally, students should be encouraged to complete all of the course materials, not just the initial sections, in order to promote course involvement. As a result, the number of quiz questions should grow, as there are nine questions in the later portion that have not been attempted in week three, accounting for 40% of the online quiz questions.

8.2 Review Process & Next Steps

The project as a whole looked at ways to improve interaction and opted to do so based on quiz results. This does not, however, justify the course's interaction, as there are additional video stats, for example. Further examinations into the other data presented would be required to gain a complete understanding of the course interaction. Furthermore, instead of forming assumptions about the course, a meeting with the course administrators should be scheduled to acquire more knowledge. Overall, the study was a success in terms of uncovering certain correlations that provided answers to the business problems. Future research in this subject could examine into why run one was so much more successful with quiz question responses.