STA302H1 – Final Project Descriptive Statistics

Danny Chen

August 10, 2021

Import STA302H1 Study Time and COVID Contemplation Time vs. Quiz Performance Dataset

Data Cleaning

First, I'll clean my data.

```r
cleaned_sta302_performance_data <- sta302_performance_data %>%
    # Create a new "country" column, which is just "Country" but whose entries are factors.
    mutate(country = as.factor(Country)) %>%

    # Remove the "X" column: it's simply the row number, which isn't very useful.
    # Remove the "Country" column: column "country" already exists
    select(-X, -Country) %>%

    # Rearrange similar columns side-by-side.
    relocate(country,
             COVID.hours..W1., COVID.hours..W2.,
             COVID.hours..W3., COVID.hours..W4.,
             STA302.hours..W1., STA302.hours..W2.,
             STA302.hours..W3., STA302.hours..W4.,
             Quiz_1_score, Quiz_2_score,
             Quiz_3_score, Quiz_4_score)
```

Helper Functions

```r
num_column_NAs = function(predictor_variable) {
  sum(is.na(predictor_variable))
}
```

```r
row_nums_of_NA_columns = function(data, predictor_variable) {
  which(is.na(predictor_variable))
}
```

```r
rows_with_num_NAs = function(data, num_NAs) {
  return (rowSums(is.na(data)) == num_NAs)
}
```

```r
row_nums_of_NA_rows = function(data, num_NAs) {
  return (which(rows_with_num_NAs(data, num_NAs)))
}
```

```r
display_histogram <- function(data, predictor_variable, histogram_title, x_axis_label) {
  ggplot(data = tibble(data), mapping = aes(x = predictor_variable)) +
    geom_histogram(col = "black", fill = "red", bins = 30) +
    labs(title = histogram_title, y = "Frequency", x = x_axis_label) +
    geom_vline(mapping = aes(xintercept = mean(predictor_variable, na.rm = TRUE)),
               color = "blue", linetype = "solid") +
    geom_vline(mapping = aes(xintercept = median(predictor_variable, na.rm = TRUE)),
               color = "dark green", linetype = "dotted")
}
```

```r
display_boxplot <- function(data, predictor_variable, boxplot_title, y_axis_label) {
  ggplot(mapping = aes(x = Country, y = predictor_variable, color = Country)) +
    geom_boxplot(mapping = aes(x = Country, y = predictor_variable)) +
    labs(title = boxplot_title,
         x = "Country",
         y = y_axis_label)
}
```

Rows With At Least One NA

Rows with at least one NA deserve closer examination.
Some of the rows might only have 1 - 2 NAs and are therefore salvageable, which is OK.
Other rows may contain 3 or more NAs, and might indicate students who have dropped STA302H1. We'd like to exclude them from our analysis.

```
rows_with_0_NAs = cleaned_sta302_performance_data[
  rows_with_num_NAs(cleaned_sta302_performance_data, 0),
]
rows_with_1_NAs = cleaned_sta302_performance_data[
  rows_with_num_NAs(cleaned_sta302_performance_data, 1),
]
rows_with_2_NAs = cleaned_sta302_performance_data[
  rows_with_num_NAs(cleaned_sta302_performance_data, 2),
]
rows_with_3_NAs = cleaned_sta302_performance_data[
  rows_with_num_NAs(cleaned_sta302_performance_data, 3),
]
rows_with_4_NAs = cleaned_sta302_performance_data[
  rows_with_num_NAs(cleaned_sta302_performance_data, 4),
]
```

Let's count the number of rows with 0 - 4 NAs.

```
print(as_tibble(data.frame(
  nrows_0_NAs = nrow(rows_with_0_NAs),
  nrows_1_NAs = nrow(rows_with_1_NAs),
  nrows_2_NAs = nrow(rows_with_2_NAs),
  nrows_3_NAs = nrow(rows_with_3_NAs),
  nrows_4_NAs = nrow(rows_with_4_NAs)
)))
```

```
## # A tibble: 1 x 5
##   nrows_0_NAs nrows_1_NAs nrows_2_NAs nrows_3_NAs nrows_4_NAs
##         <int>       <int>       <int>       <int>       <int>
## 1         143           9          16          19           1
```

And then we'll determine which row numbers have 0 - 4 NAs.

```
row_nums_of_NA_rows(cleaned_sta302_performance_data, 0)
```

```
##   [1]   1   2   3   4   7   8  11  13  14  15  18  20  21  22  24  25  26  27
##  [19]  29  31  32  33  35  36  37  38  42  44  45  48  50  54  55  57  60  61
##  [37]  62  63  65  66  67  68  70  71  72  73  74  75  76  77  79  81  82  83
##  [55]  84  85  86  87  88  89  92  93  94  97  99 101 103 104 105 106 107 108
##  [73] 109 110 111 112 114 115 116 118 119 122 123 124 126 127 128 129 130 134
##  [91] 135 136 137 139 140 141 142 144 146 147 149 150 151 152 153 154 155 156
## [109] 157 158 159 160 161 162 163 164 165 166 167 169 170 171 172 173 174 175
## [127] 176 177 178 179 180 183 184 185 186 187 190 191 193 196 199 200 201
```

```
row_nums_of_NA_rows(cleaned_sta302_performance_data, 1)
```

```
## [1]  34  78  80 117 132 138 143 145 197
```

```
row_nums_of_NA_rows(cleaned_sta302_performance_data, 2)
```

```
##  [1]  10  12  43  52  59  90  95  96  98 100 121 125 131 181 189 192
```

```
row_nums_of_NA_rows(cleaned_sta302_performance_data, 3)
```

```
##  [1]   5   6  28  69 113 188 195 202 203 205 207 208 209 211 215 216 217 218 221
```

```
row_nums_of_NA_rows(cleaned_sta302_performance_data, 4)
```

```
## [1] 223
```

Columns with NAs

```
perform_data = cleaned_sta302_performance_data
print(as_tibble(data.frame(
  week1_covid = num_column_NAs(perform_data$COVID.hours..W1.),
  week2_covid = num_column_NAs(perform_data$COVID.hours..W2.),
  week3_covid = num_column_NAs(perform_data$COVID.hours..W3.),
  week4_covid = num_column_NAs(perform_data$COVID.hours..W4.)
)))
```

```
## # A tibble: 1 x 4
##   week1_covid week2_covid week3_covid week4_covid
##         <int>       <int>       <int>       <int>
## 1          26          22          21          40
```

```
print(as_tibble(data.frame(
  week1_sta302 = num_column_NAs(perform_data$STA302.hours..W1.),
  week2_sta302 = num_column_NAs(perform_data$STA302.hours..W2.),
  week3_sta302 = num_column_NAs(perform_data$STA302.hours..W3.),
  week4_sta302 = num_column_NAs(perform_data$STA302.hours..W4.)
)))
```

```
## # A tibble: 1 x 4
##   week1_sta302 week2_sta302 week3_sta302 week4_sta302
##          <int>        <int>        <int>        <int>
## 1           26           22           20           40
```

```
print(as_tibble(data.frame(
  quiz1_score = num_column_NAs(perform_data$Quiz_1_score),
  quiz2_score = num_column_NAs(perform_data$Quiz_2_score),
  quiz3_score = num_column_NAs(perform_data$Quiz_3_score),
  quiz4_score = num_column_NAs(perform_data$Quiz_4_score)
)))
```

```
## # A tibble: 1 x 4
##   quiz1_score quiz2_score quiz3_score quiz4_score
##         <int>       <int>       <int>       <int>
## 1          13          36          31          34
```

Columns only with Quiz Grades

```
quiz_grades <- cleaned_sta302_performance_data %>%
  select(Quiz_1_score, Quiz_2_score,
         Quiz_3_score, Quiz_4_score)
```

```
missed_0_quizzes = quiz_grades[
  rows_with_num_NAs(quiz_grades, 0),
]
missed_1_quizzes = quiz_grades[
  rows_with_num_NAs(quiz_grades, 1),
]
missed_2_quizzes = quiz_grades[
  rows_with_num_NAs(quiz_grades, 2),
]
missed_3_quizzes = quiz_grades[
  rows_with_num_NAs(quiz_grades, 3),
]
missed_4_quizzes = quiz_grades[
  rows_with_num_NAs(quiz_grades, 4),
]
```

Number of Missed Quizzes

```
print(as.tibble(data.frame(
  miss_0_quizzes = nrow(missed_0_quizzes),
  miss_1_quizzes = nrow(missed_1_quizzes),
  miss_2_quizzes = nrow(missed_2_quizzes),
  miss_3_quizzes = nrow(missed_3_quizzes),
  miss_4_quizzes = nrow(missed_4_quizzes)
)))
```

```
## Warning: ‘as.tibble()‘ was deprecated in tibble 2.0.0.
## Please use ‘as_tibble()‘ instead.
## The signature and semantics have changed, see ‘?as_tibble‘.
```

```
## # A tibble: 1 x 5
##   miss_0_quizzes miss_1_quizzes miss_2_quizzes miss_3_quizzes miss_4_quizzes
##            <int>          <int>          <int>          <int>          <int>
## 1            176             20              3             24              4
```

Who Missed Quizzes?

```
which(rows_with_num_NAs(quiz_grades, 0))
```

```
##  [1]  1  2  3  4  7  8 10 11 12 13 14 15 18 20 21 22 24 25
## [19] 26 27 29 31 32 33 35 36 37 38 42 43 44 45 48 50 52 54
##  [37]  55  57  59  60  61  62  63  65  66  67  68  70  71  72  73  74  75  76
##  [55]  77  79  81  82  83  84  85  86  87  88  89  90  92  93  94  95  96  97
##  [73]  98  99 100 101 103 104 105 106 107 108 109 110 111 112 114 115 116 118
##  [91] 119 121 122 123 124 125 126 127 128 129 130 131 134 135 136 137 139 140
## [109] 141 142 144 146 147 149 150 151 152 153 154 155 156 157 158 159 160 161
## [127] 162 163 164 165 166 167 169 170 171 172 173 174 175 176 177 178 179 180
## [145] 181 183 184 185 186 187 189 190 191 192 193 196 199 200 201 202 203 204
## [163] 205 207 208 209 210 211 215 216 217 218 219 220 221 222
```

```
which(rows_with_num_NAs(quiz_grades, 1))
```

```
##  [1]   5   6  28  34  56  69  78  80 113 117 132 138 143 145 188 195 197 206 223
## [20] 225
```

```
which(rows_with_num_NAs(quiz_grades, 2))
```

```
## [1]   9  17 226
```

```
which(rows_with_num_NAs(quiz_grades, 3))
```

```
##  [1]  16  19  23  30  39  40  41  47  49  51  53  64  91 102 120 148 168 182 194
## [20] 198 212 213 214 227
```

```
which(rows_with_num_NAs(quiz_grades, 4))
```

```
## [1]  46  58 133 224
```

Who to Exclude from the Dataset?

Identify rows with at least 3 missing quiz marks. These indicate students who have dropped STA302H1, and who should be excluded from the final data.

Notice that we didn't check the number of NAs for country of origin, COVID hours, and STA302H1 hours, since some students either forgot or abstained. So there's no reason to exclude these students from our final dataset.

```
library(dplyr)
row_nums_to_exclude <- union(
  which(rows_with_num_NAs(quiz_grades, 3)),
  which(rows_with_num_NAs(quiz_grades, 4))) # that's 28 people.
row_nums_to_exclude
```

```
##  [1]  16  19  23  30  39  40  41  47  49  51  53  64  91 102 120 148 168 182 194
## [20] 198 212 213 214 227  46  58 133 224
```

```
cleaned_sta302_performance_data2 =
  cleaned_sta302_performance_data[-row_nums_to_exclude,]
```

Rows with Mistyped Columns

Rows whose columns are mis-typed may need to be corrected via imputation.

```
rows_with_mistyped_columms = cleaned_sta302_performance_data2[c(38, 83, 84, 117),]
# row 83: Country -> "canada" -- DONE
# row 84: Country -> "canada" -- DONE

# row 117: COVID.hours..W4. -> 0.5 hours -- DONE

# row 38:  STA302.hours..W3. -> 5.5<U+00A0> -- DONE
# row 117: STA302.hours..W4. -> 7.5 hours -- DONE
```

```
# library(janitor)
# use it to clean up data.
```

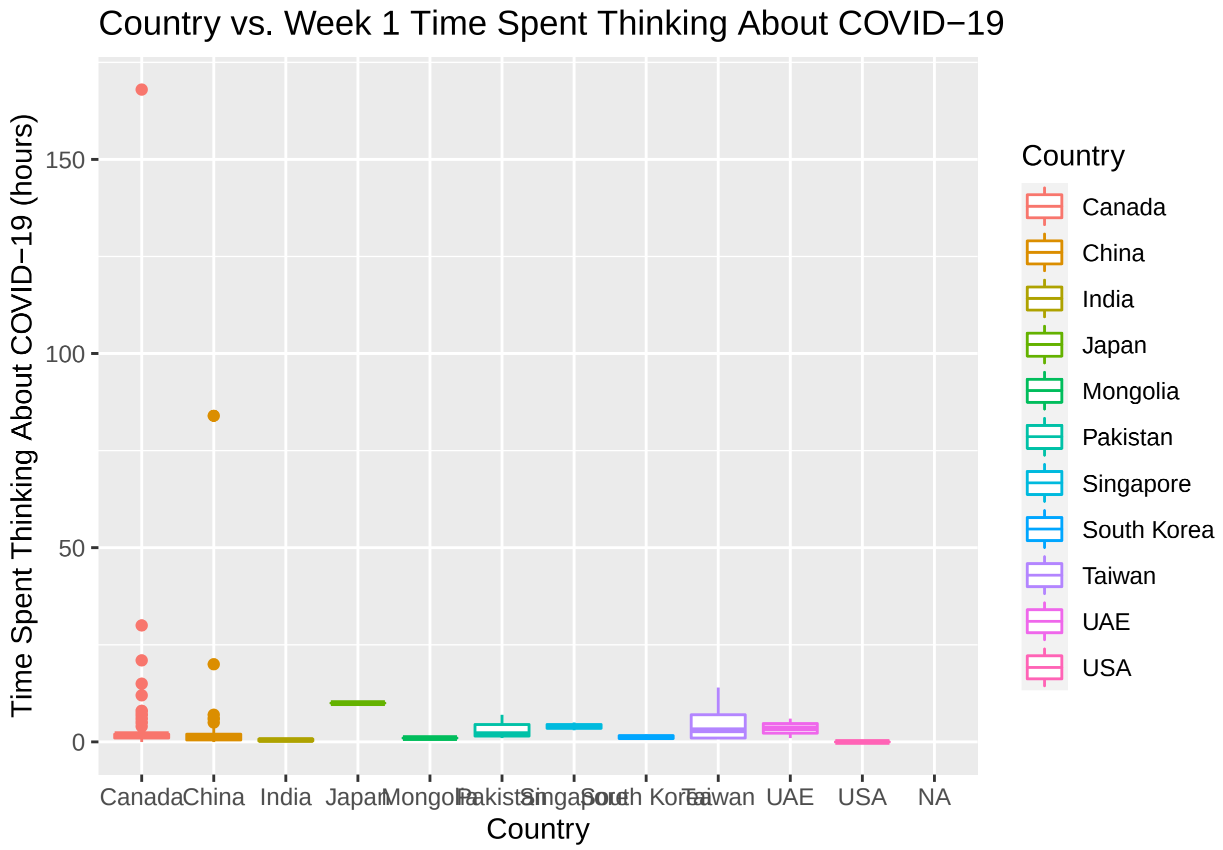
Rows Without Country Entry

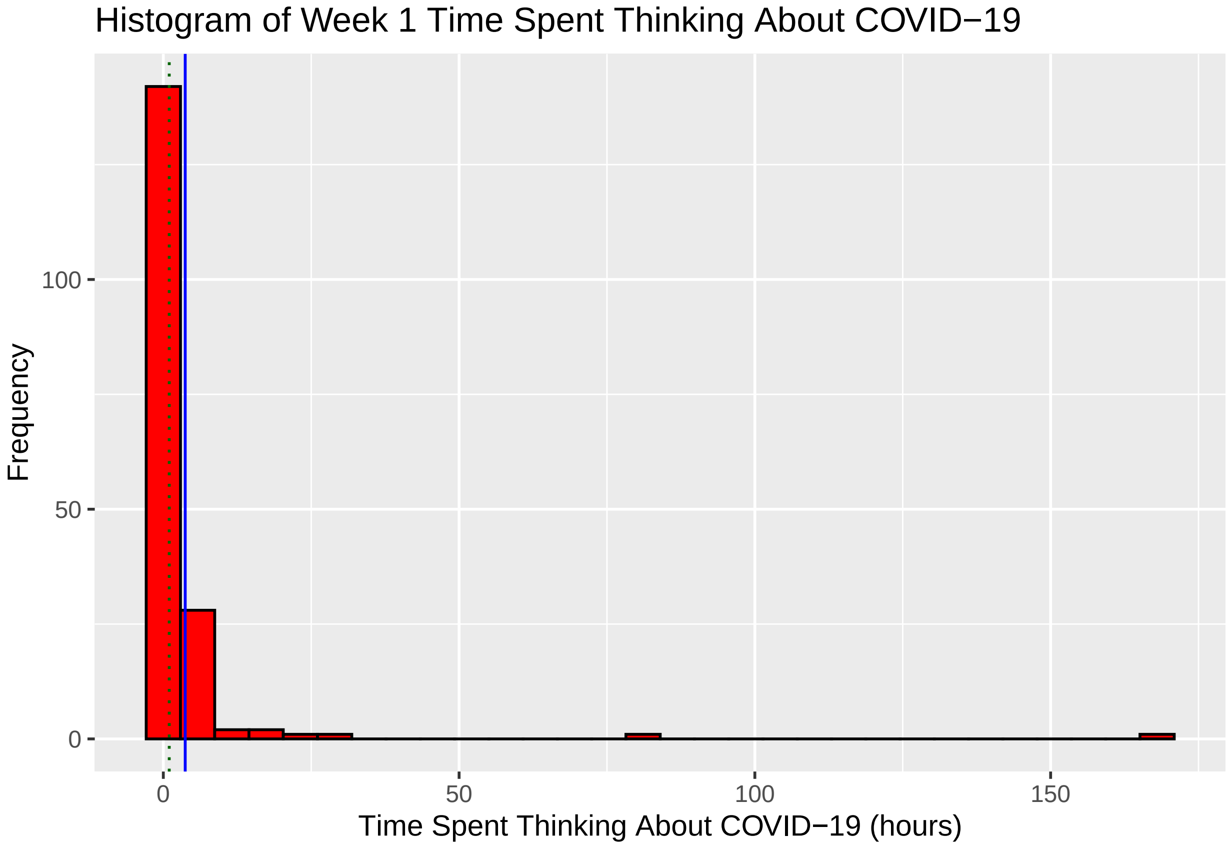
Taking out the country column can come in handy for functions like cor() where factors aren't allowed.

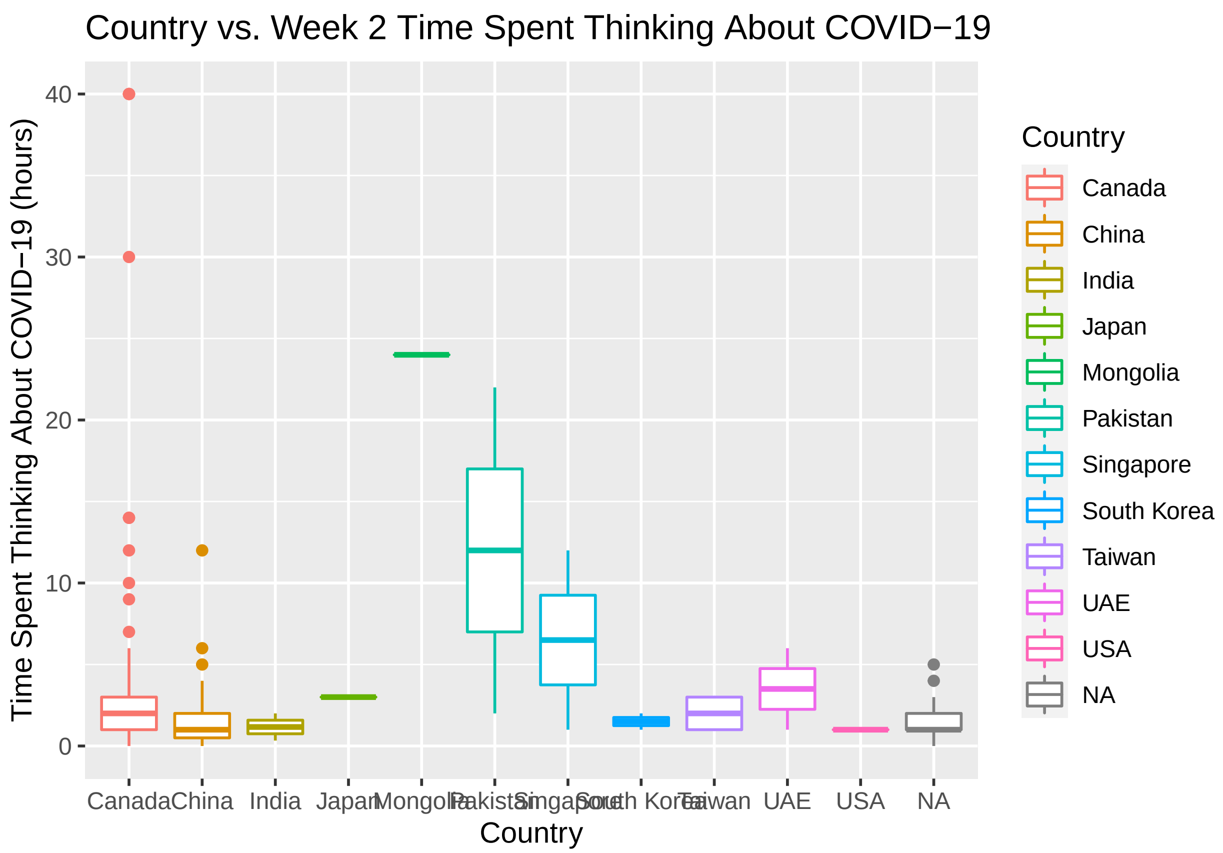
```
rows_with_no_country = cleaned_sta302_performance_data2 %>%
  select(-country)
```

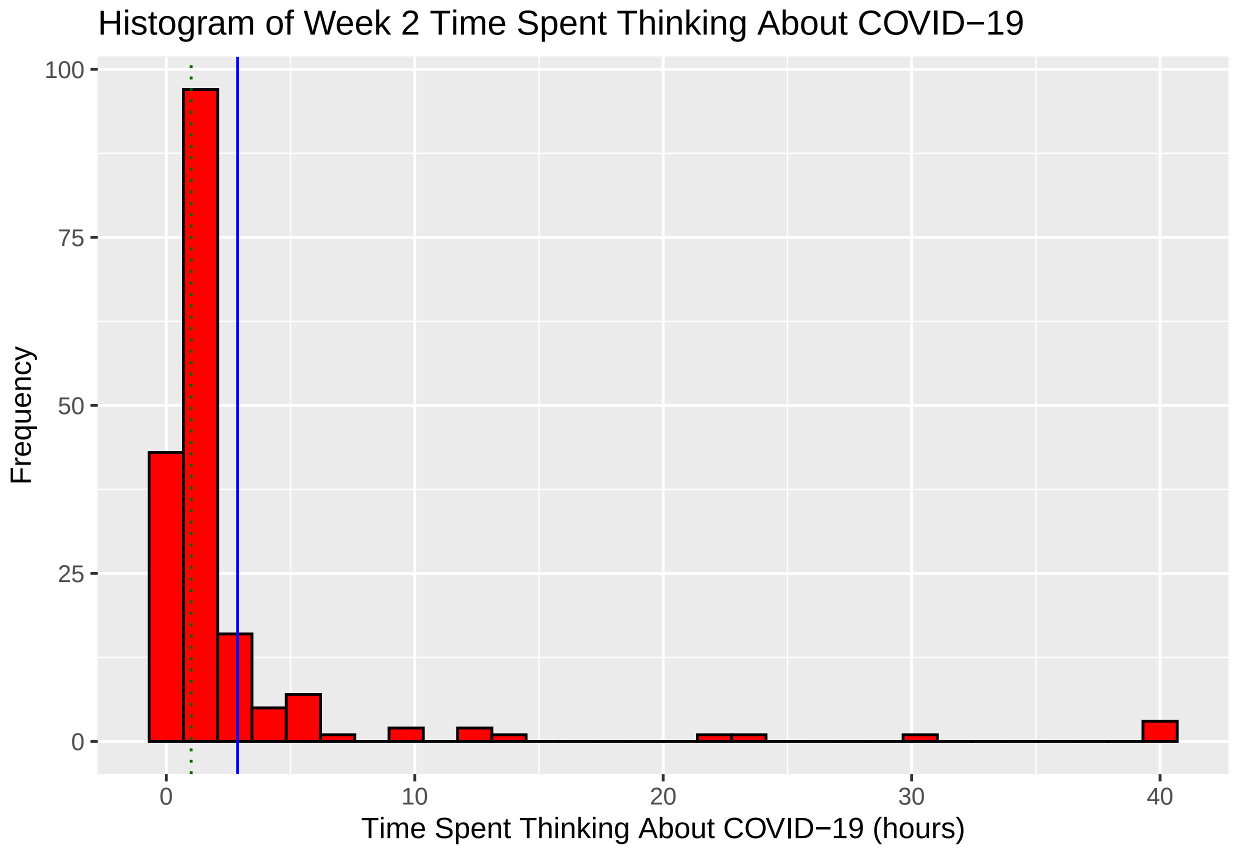
Find Significance Predictor Variables, Select Predictor Variables Based on Criterion

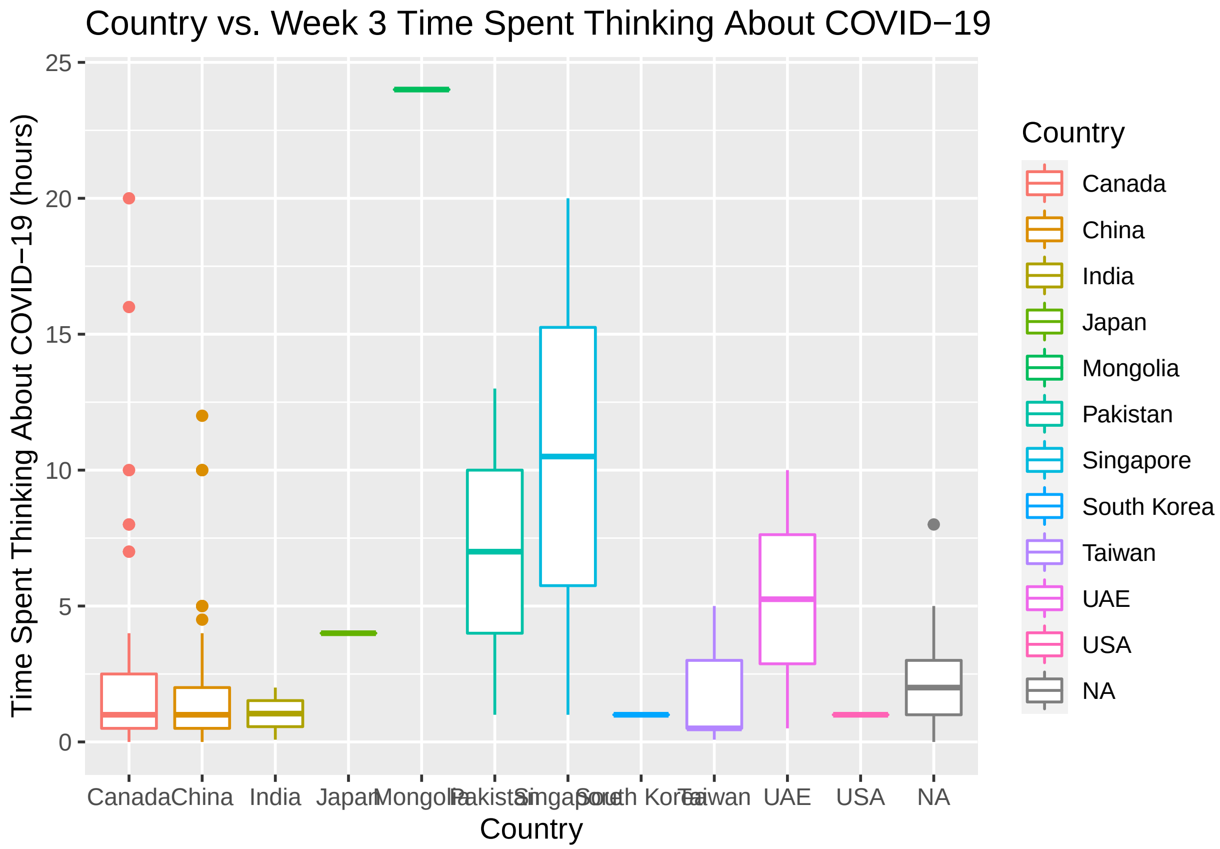
```
# use week 5b slides -- choose model selection criterion to pick predictor variables.


# use lm() on a bunch of predictor variables to determine significant
# predictor variables.
```

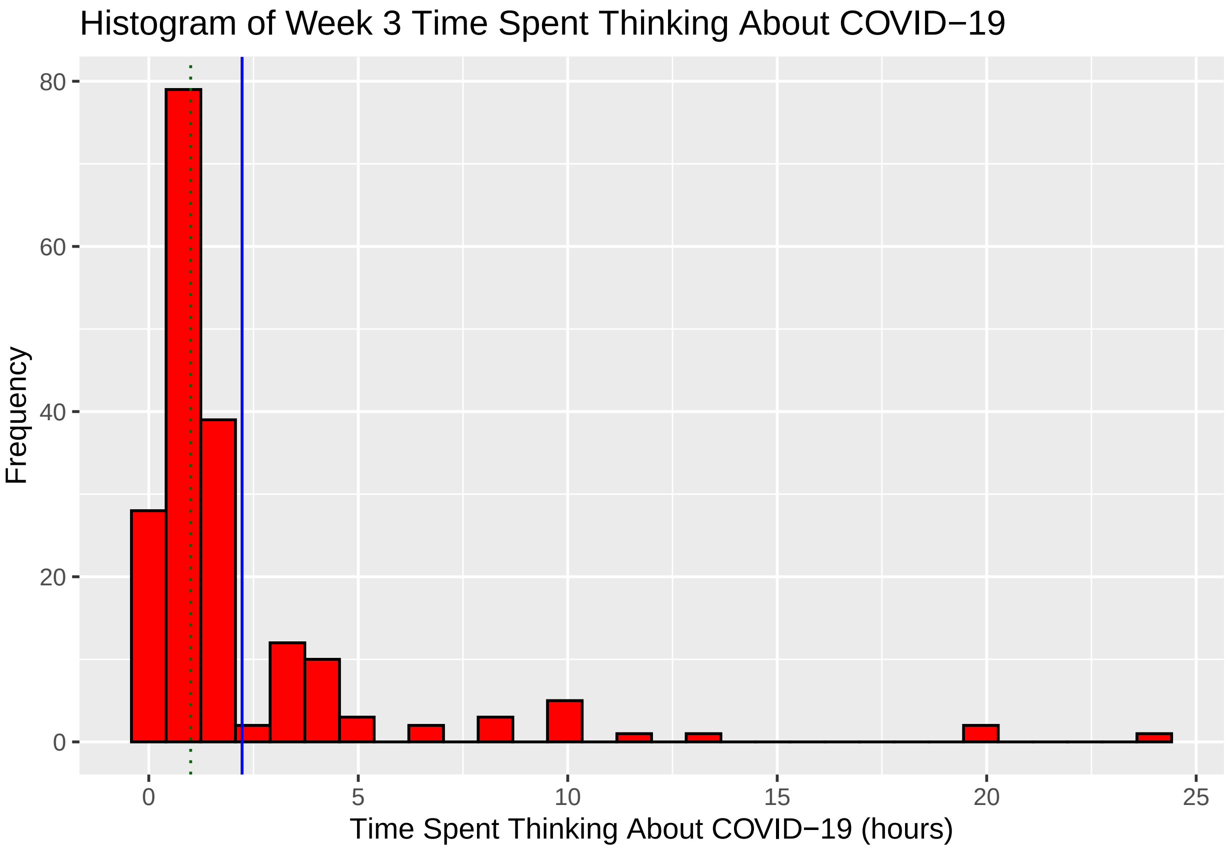

Country vs. Week 1 Time Spent Thinking About COVID−19
Time Spent Thinking About COVID−19 (hours)
0
50
100
150
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA

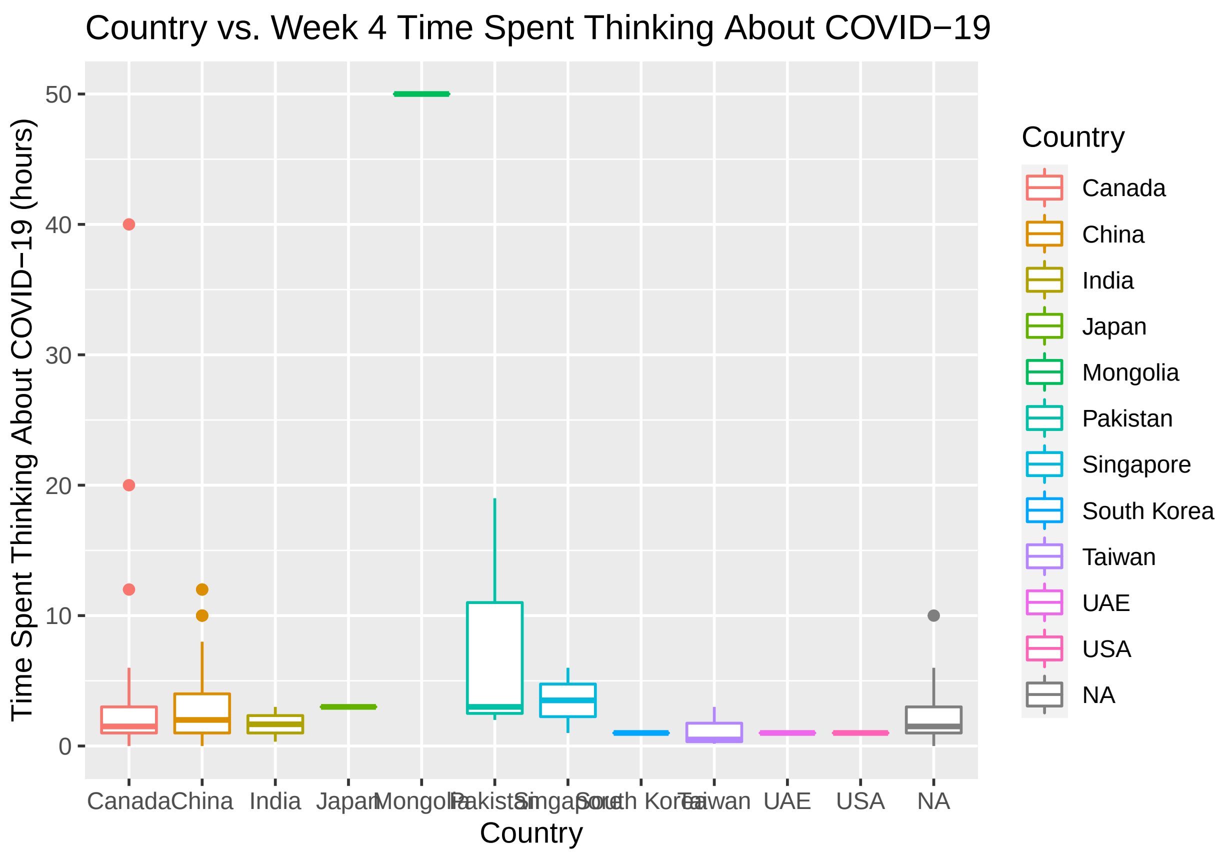

Histogram of Week 1 Time Spent Thinking About COVID−19
Frequency
0
50
100
0
50
100
150
Time Spent Thinking About COVID−19 (hours)

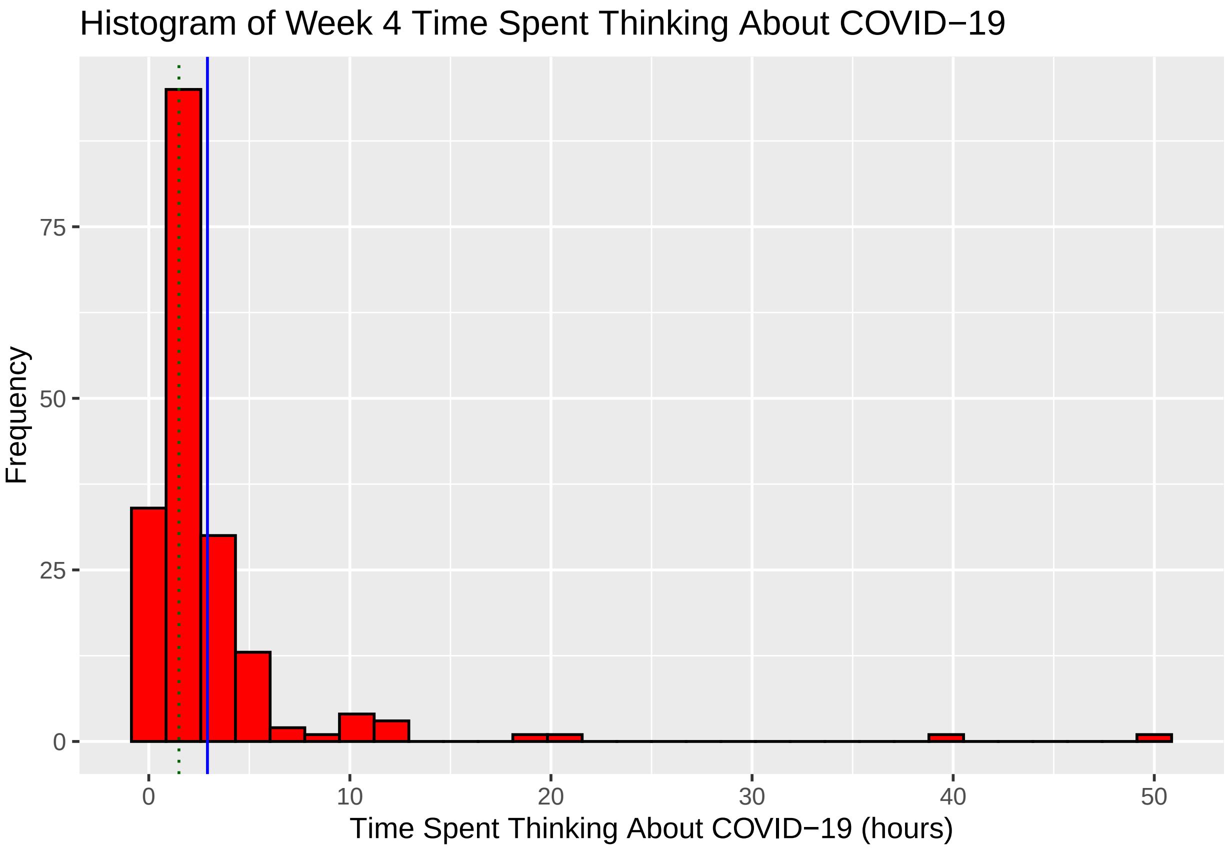

Country vs. Week 2 Time Spent Thinking About COVID−19

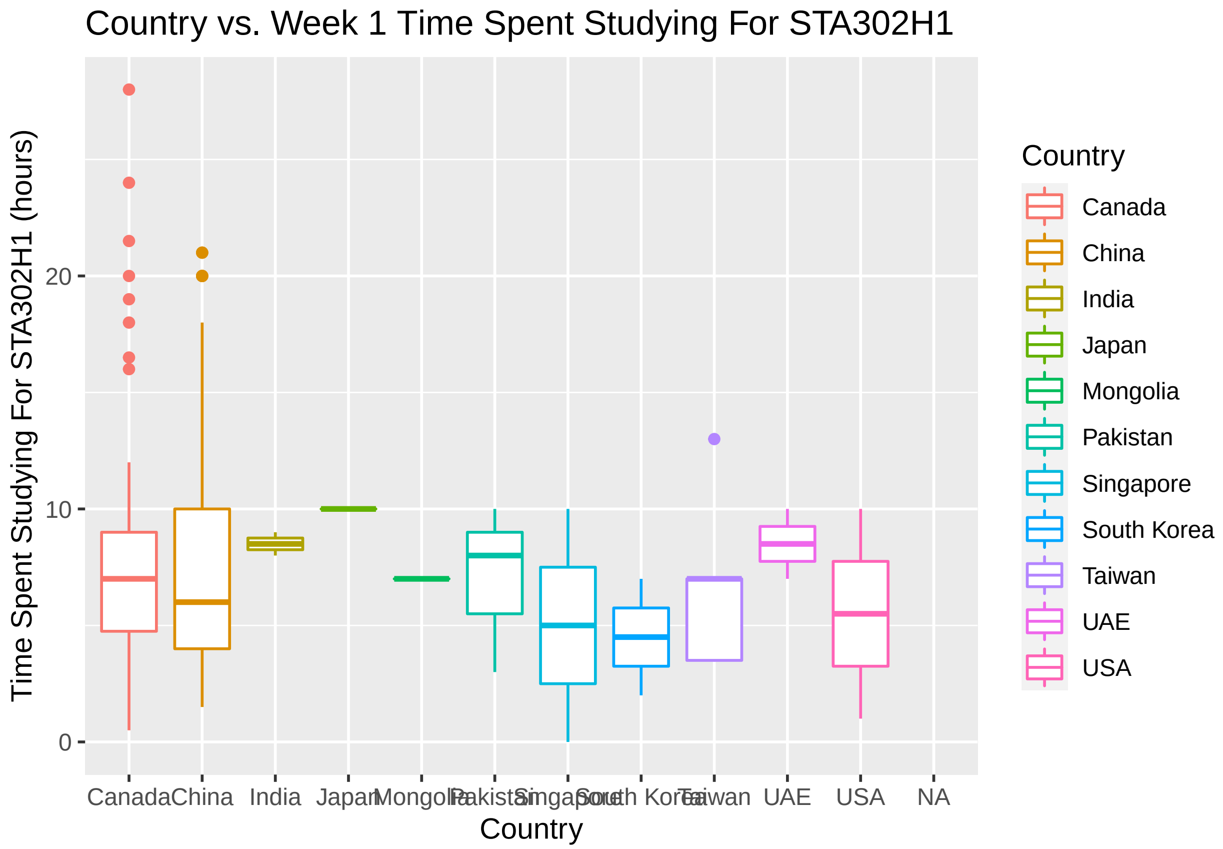

Histogram of Week 2 Time Spent Thinking About COVID−19

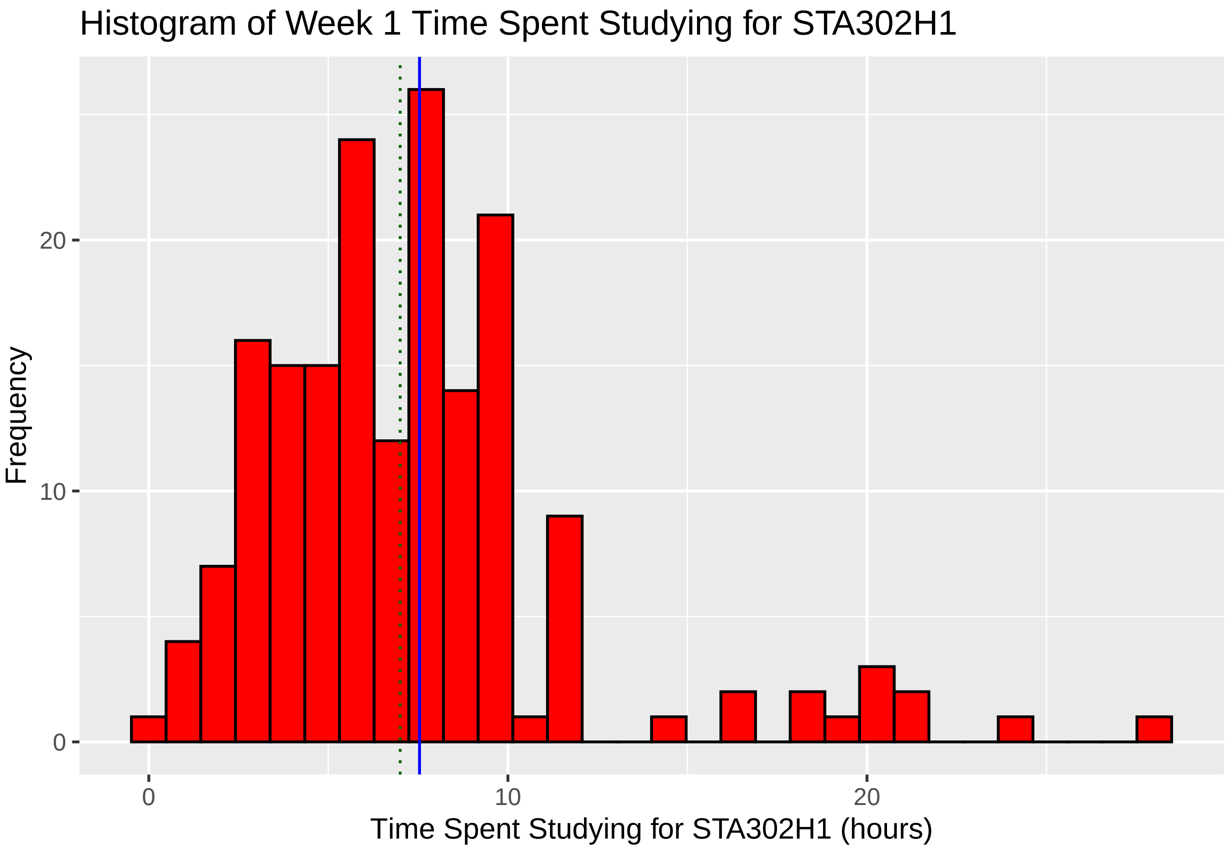

Country vs. Week 3 Time Spent Thinking About COVID−19
Time Spent Thinking About COVID−19 (hours)
0
5
10
15
20
25
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA

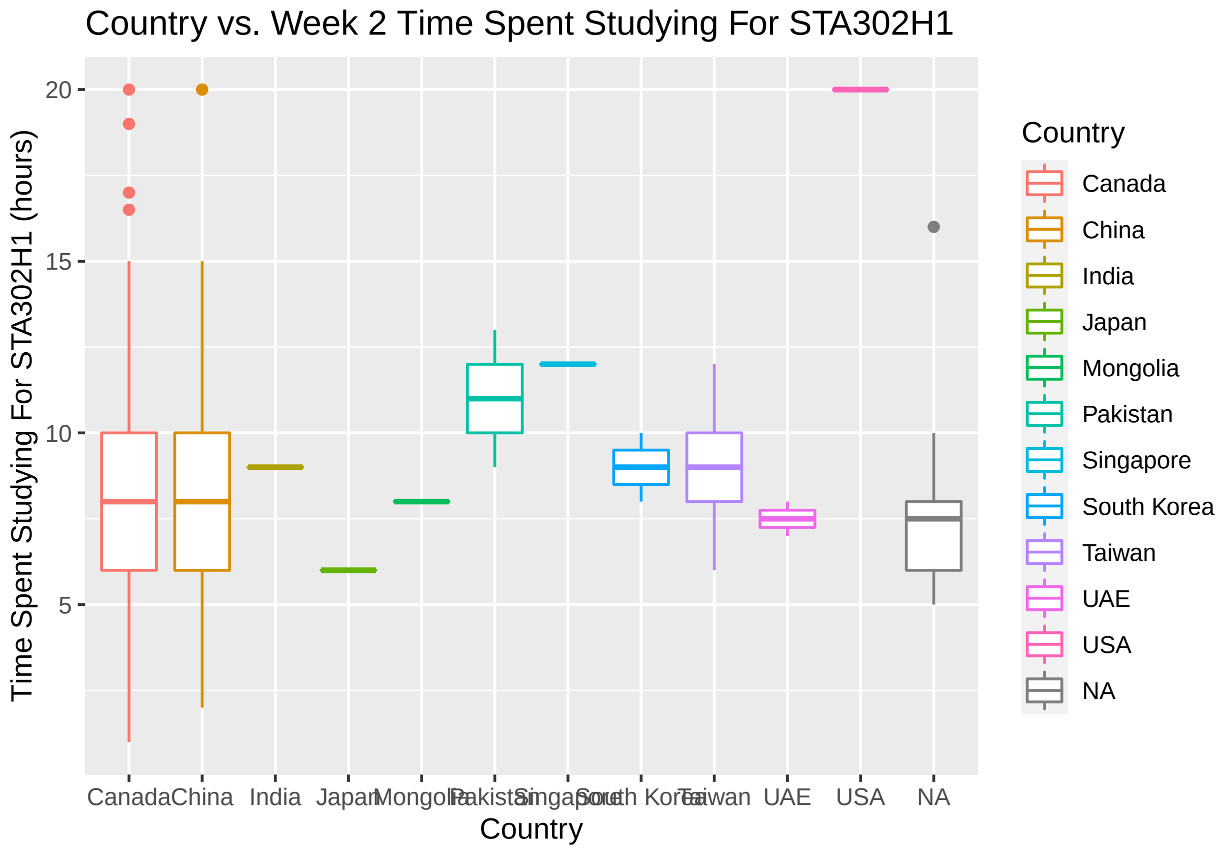

Histogram of Week 3 Time Spent Thinking About COVID−19
Frequency
0
20
40
60
80
0
5
10
15
20
25
Time Spent Thinking About COVID−19 (hours)

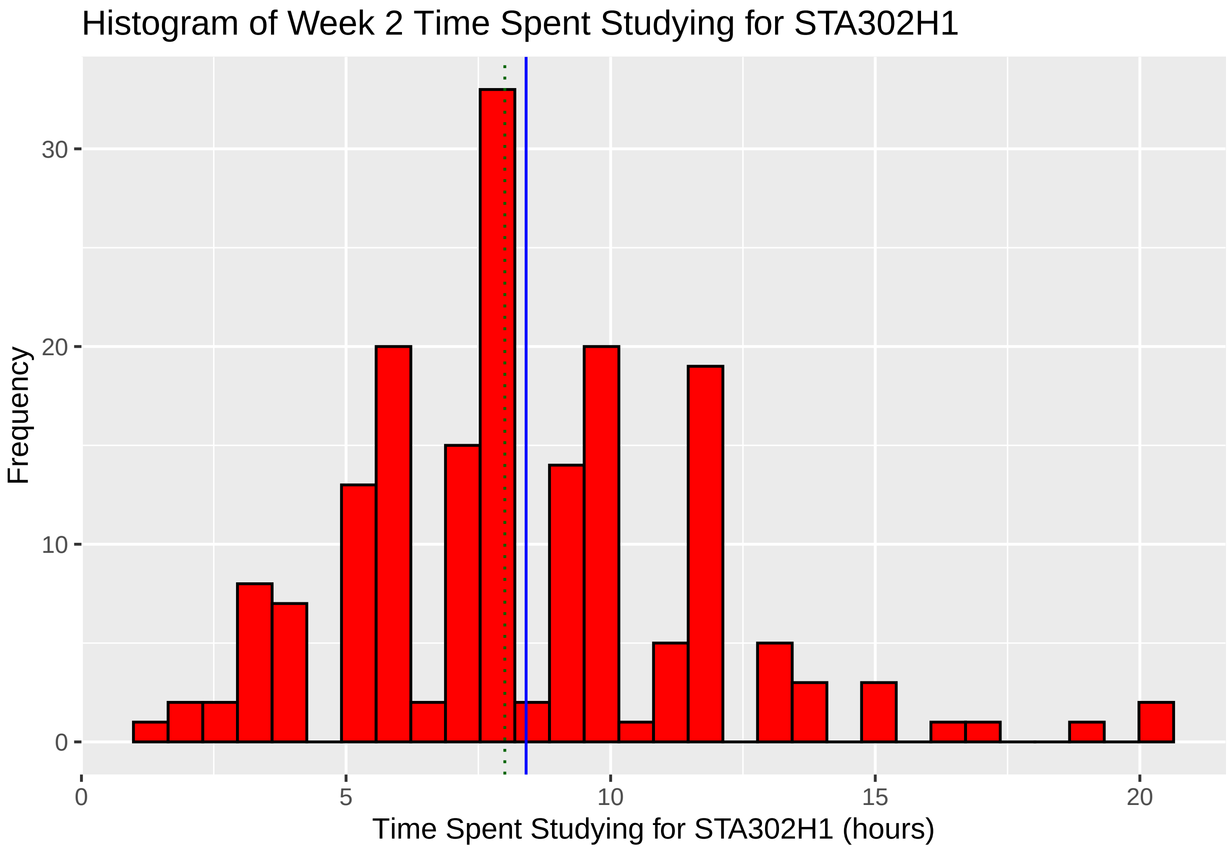

Country vs. Week 4 Time Spent Thinking About COVID−19
Time Spent Thinking About COVID−19 (hours)
0
10
20
30
40
50
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA

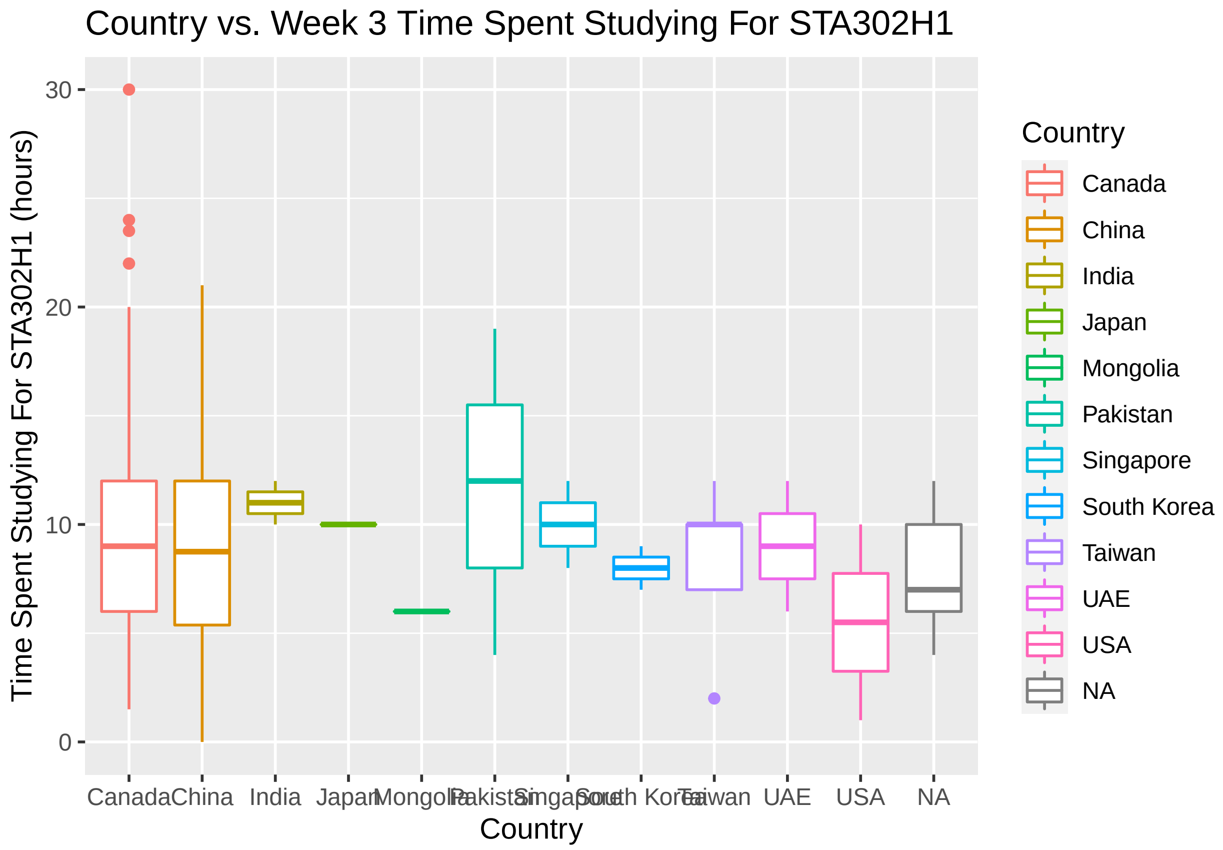

Histogram of Week 4 Time Spent Thinking About COVID−19
Frequency
0
25
50
75
0
10
20
30
40
50
Time Spent Thinking About COVID−19 (hours)

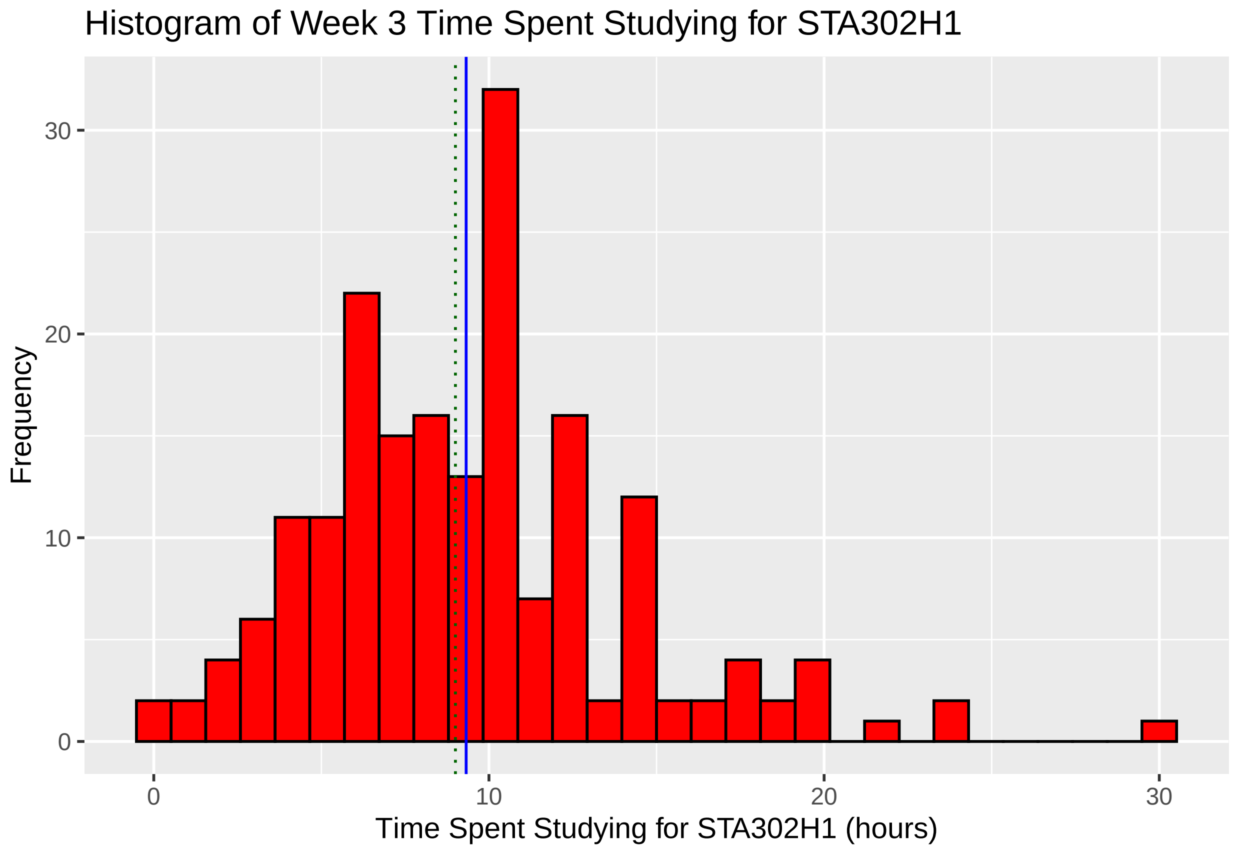

Country vs. Week 1 Time Spent Studying For STA302H1

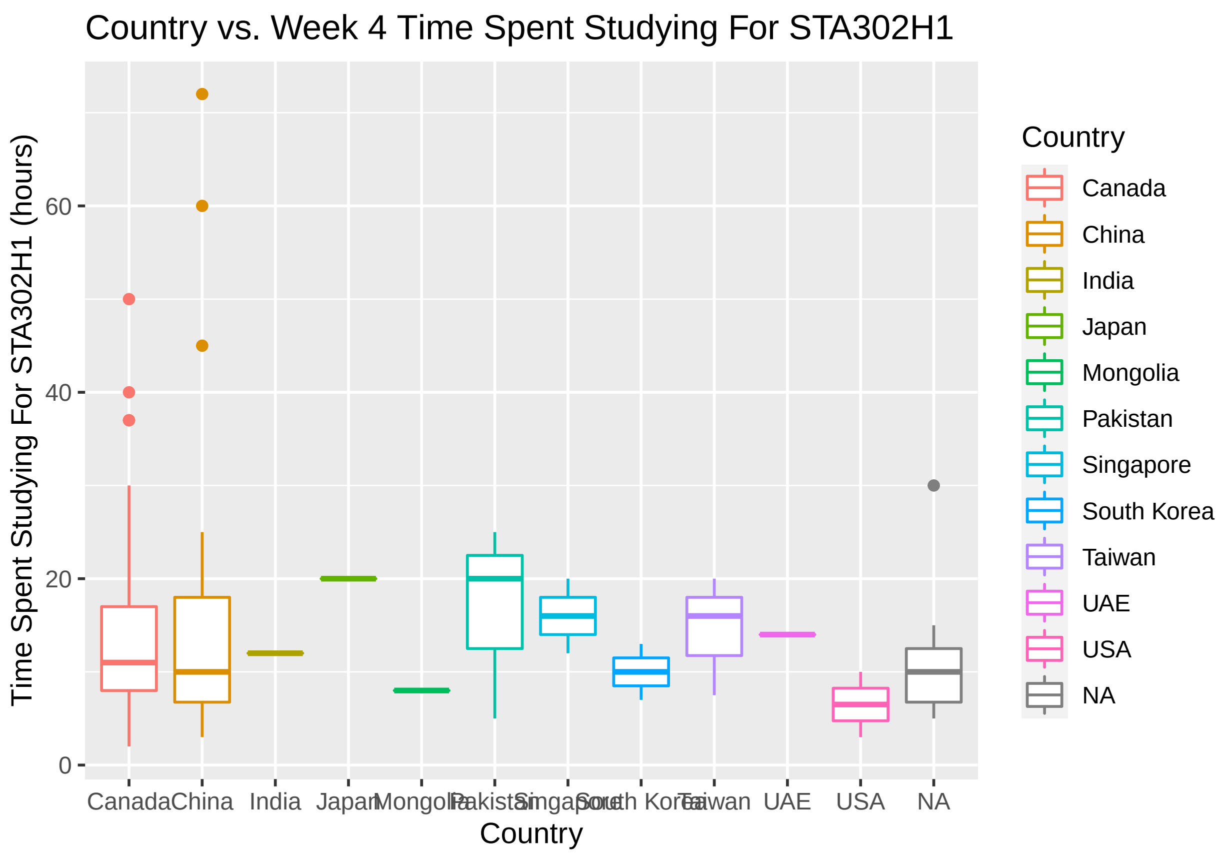

Histogram of Week 1 Time Spent Studying for STA302H1

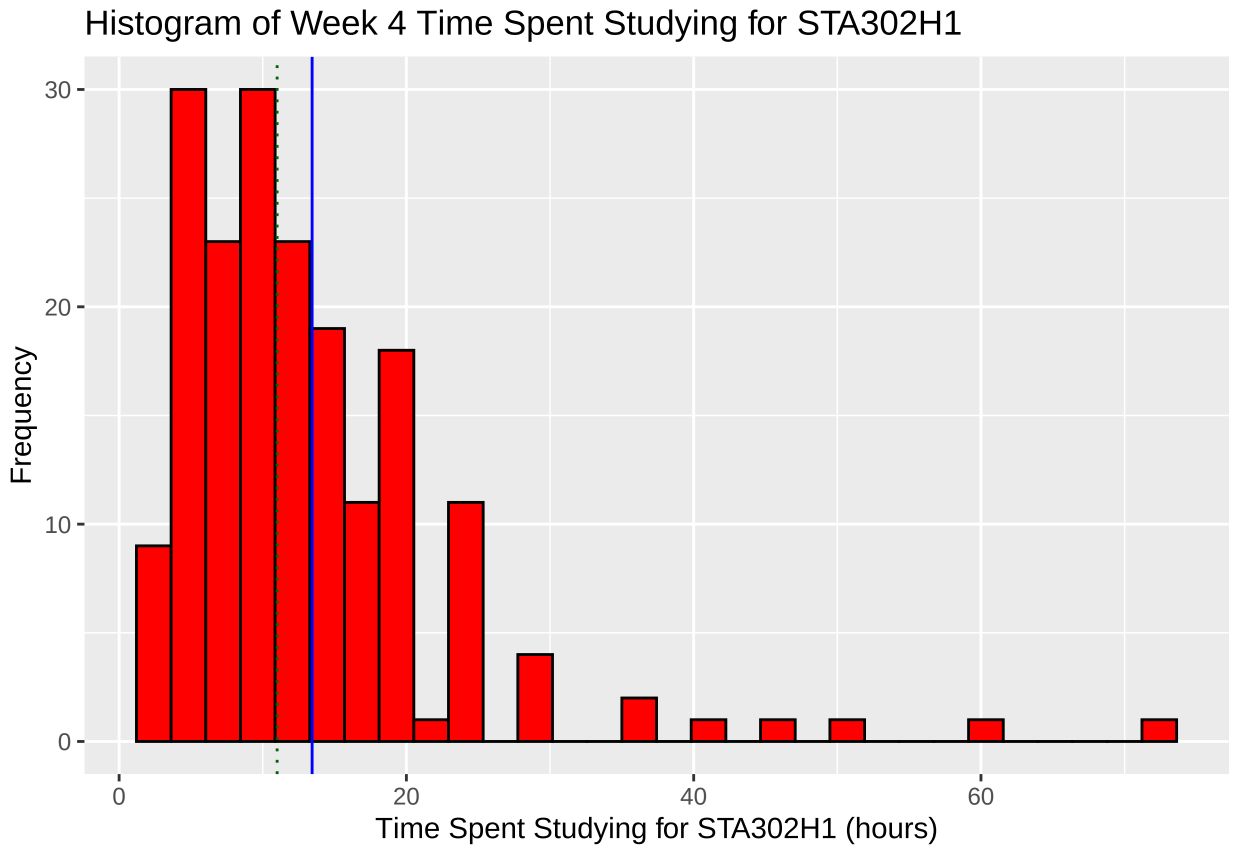

Country vs. Week 2 Time Spent Studying For STA302H1
Time Spent Studying For STA302H1 (hours)
20
15
10
5
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA

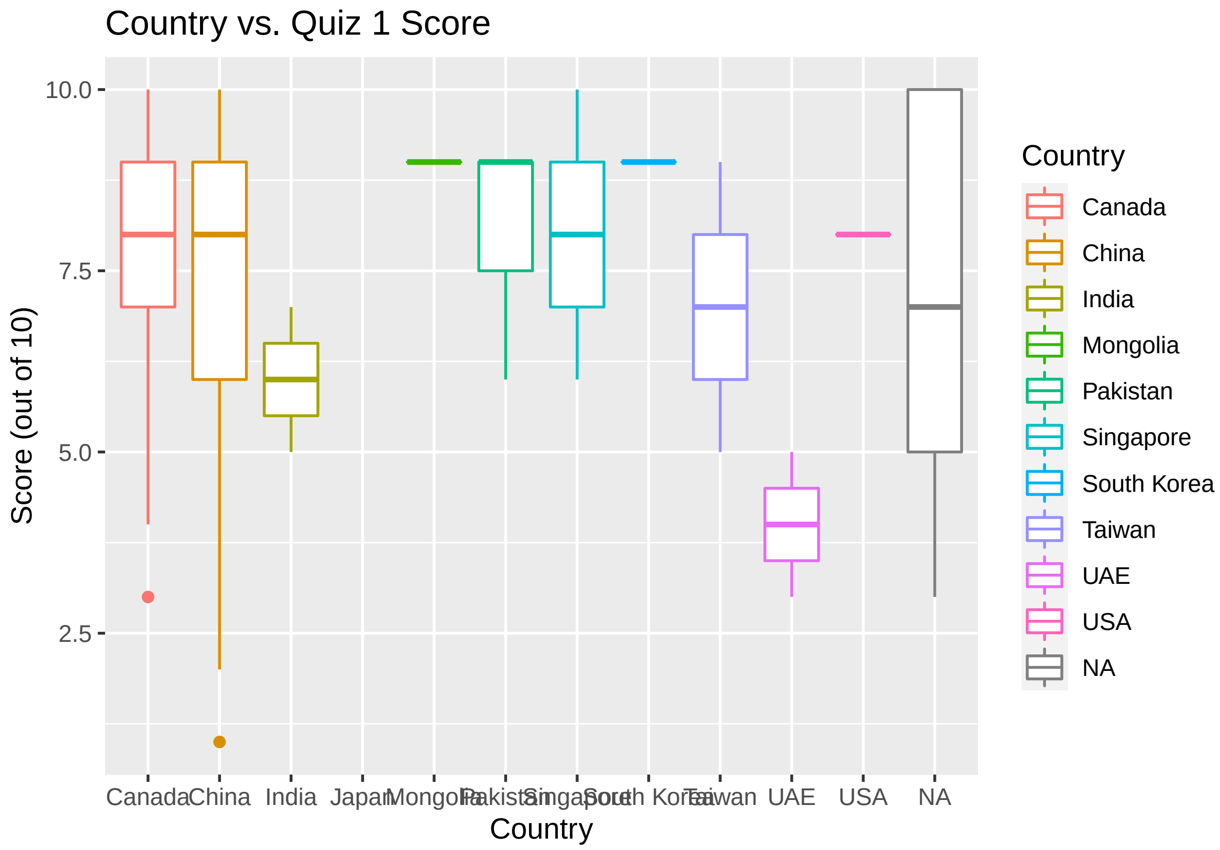

Histogram of Week 2 Time Spent Studying for STA302H1
Frequency
30
20
10
0
0
5
10
15
20
Time Spent Studying for STA302H1 (hours)

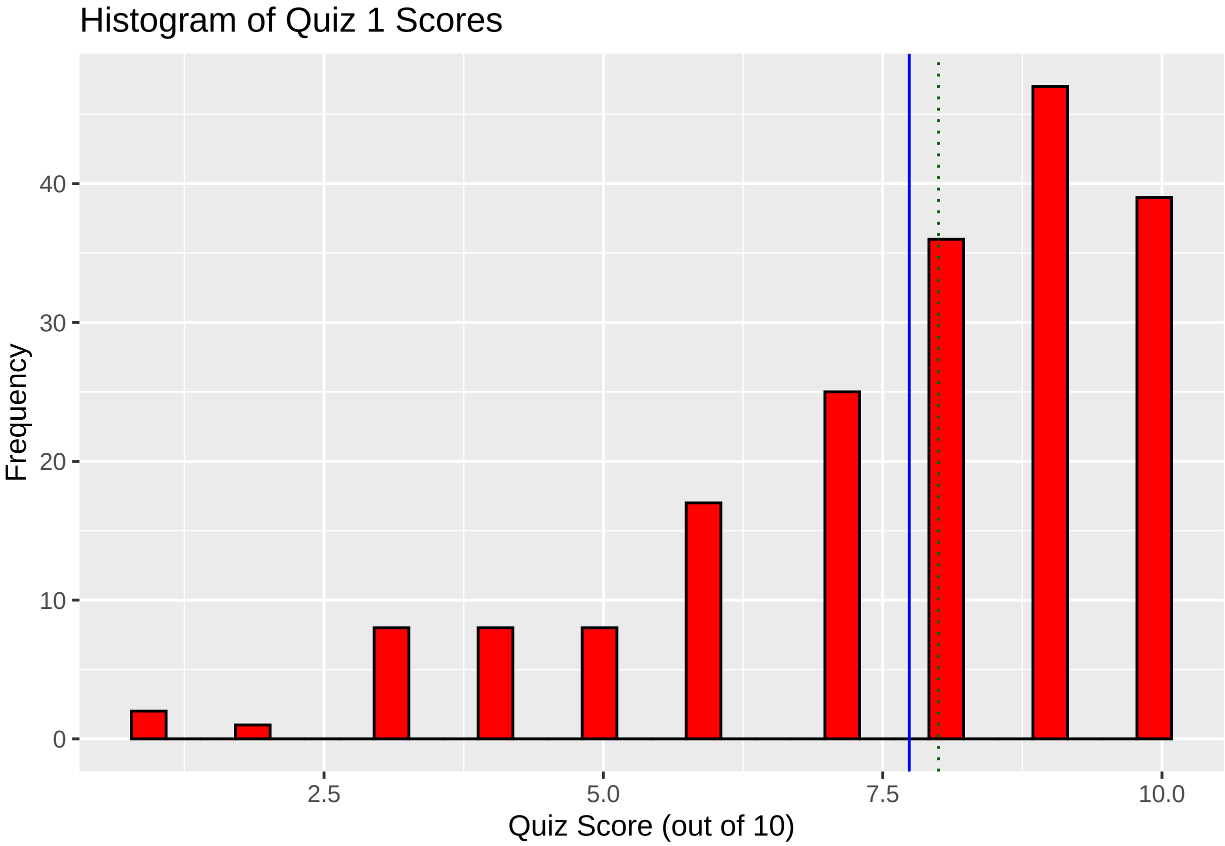

Country vs. Week 3 Time Spent Studying For STA302H1

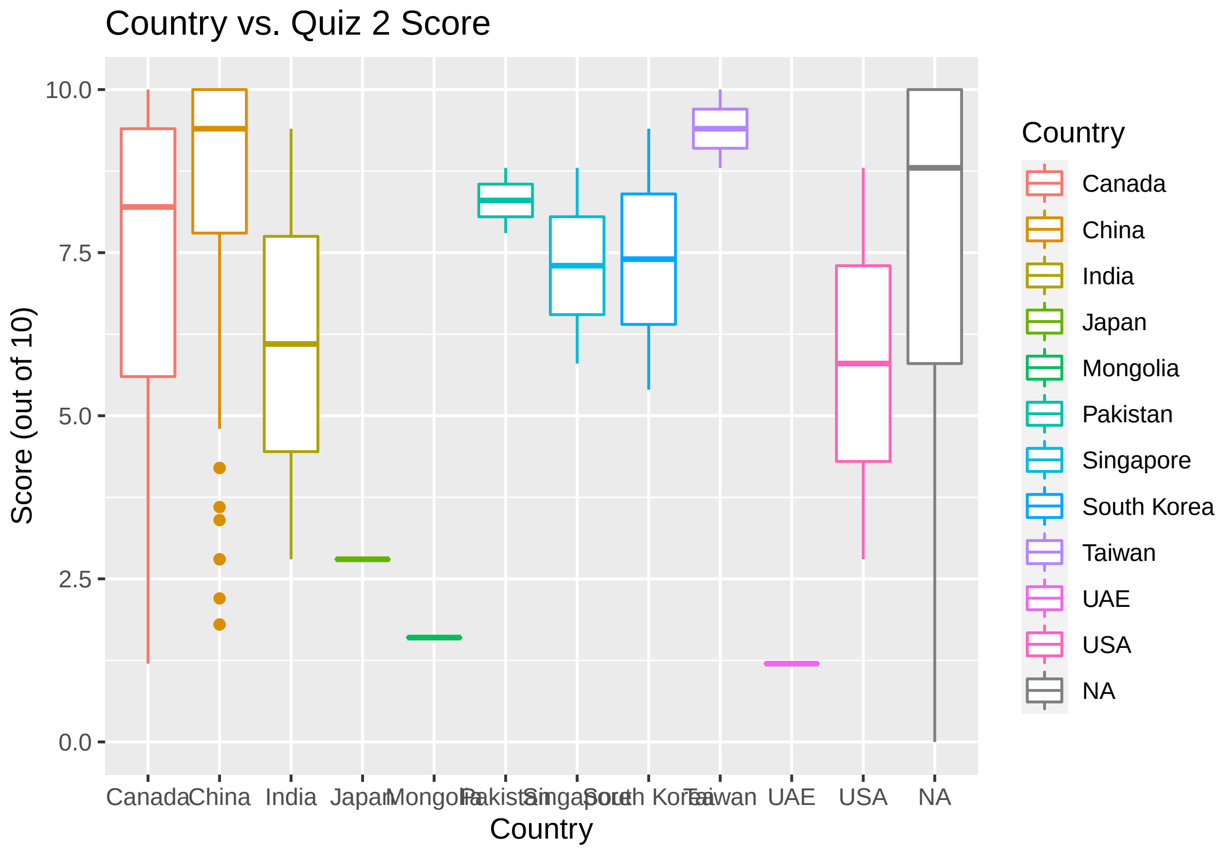

Histogram of Week 3 Time Spent Studying for STA302H1

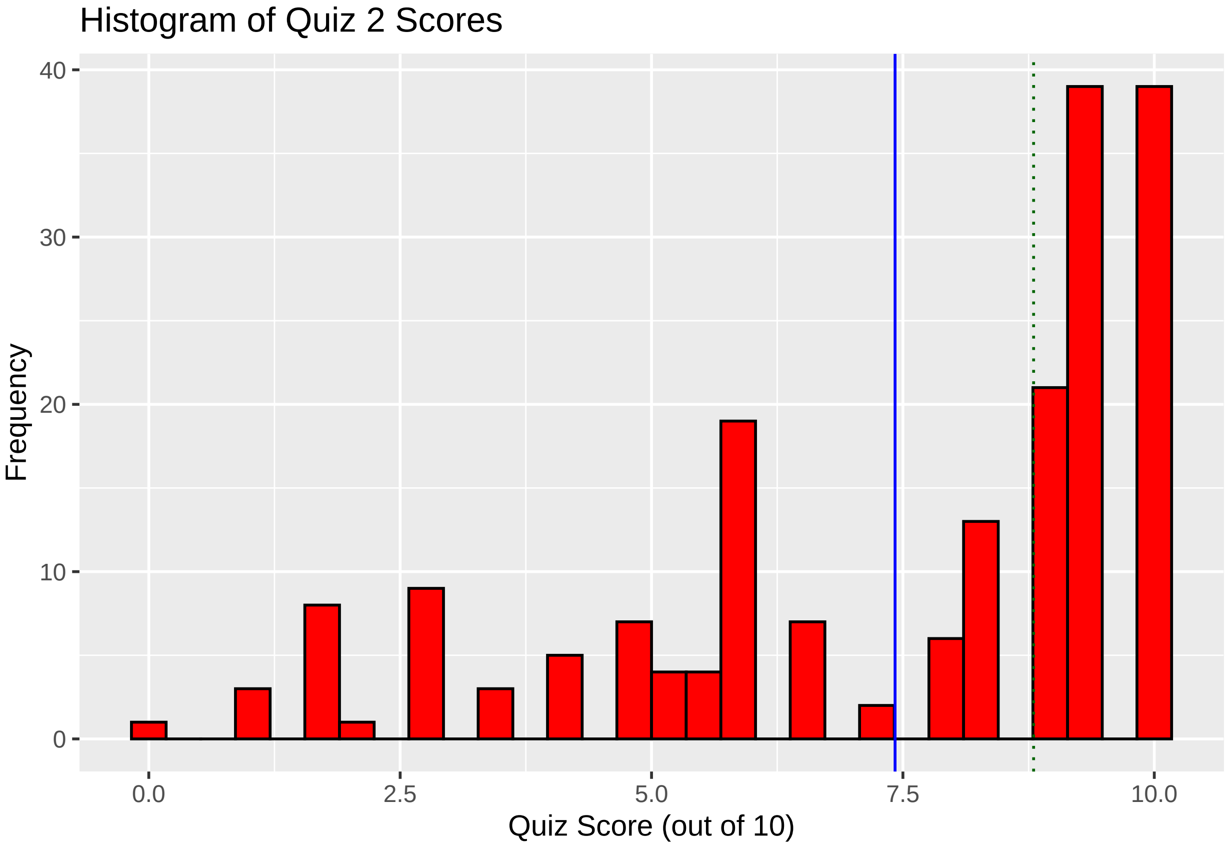

Country vs. Week 4 Time Spent Studying For STA302H1
Time Spent Studying For STA302H1 (hours)
0
20
40
60
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA

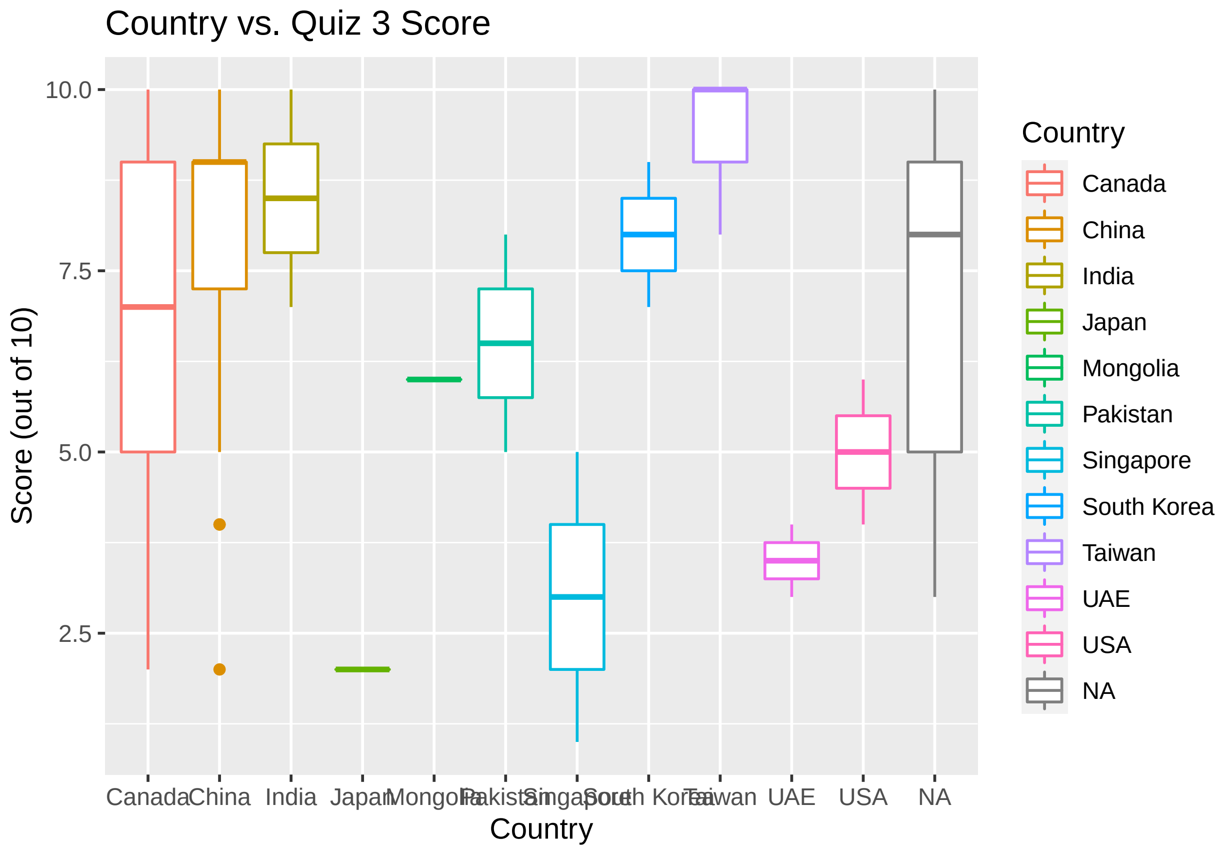

Histogram of Week 4 Time Spent Studying for STA302H1
Frequency
0
10
20
30
0
20
40
60
Time Spent Studying for STA302H1 (hours)

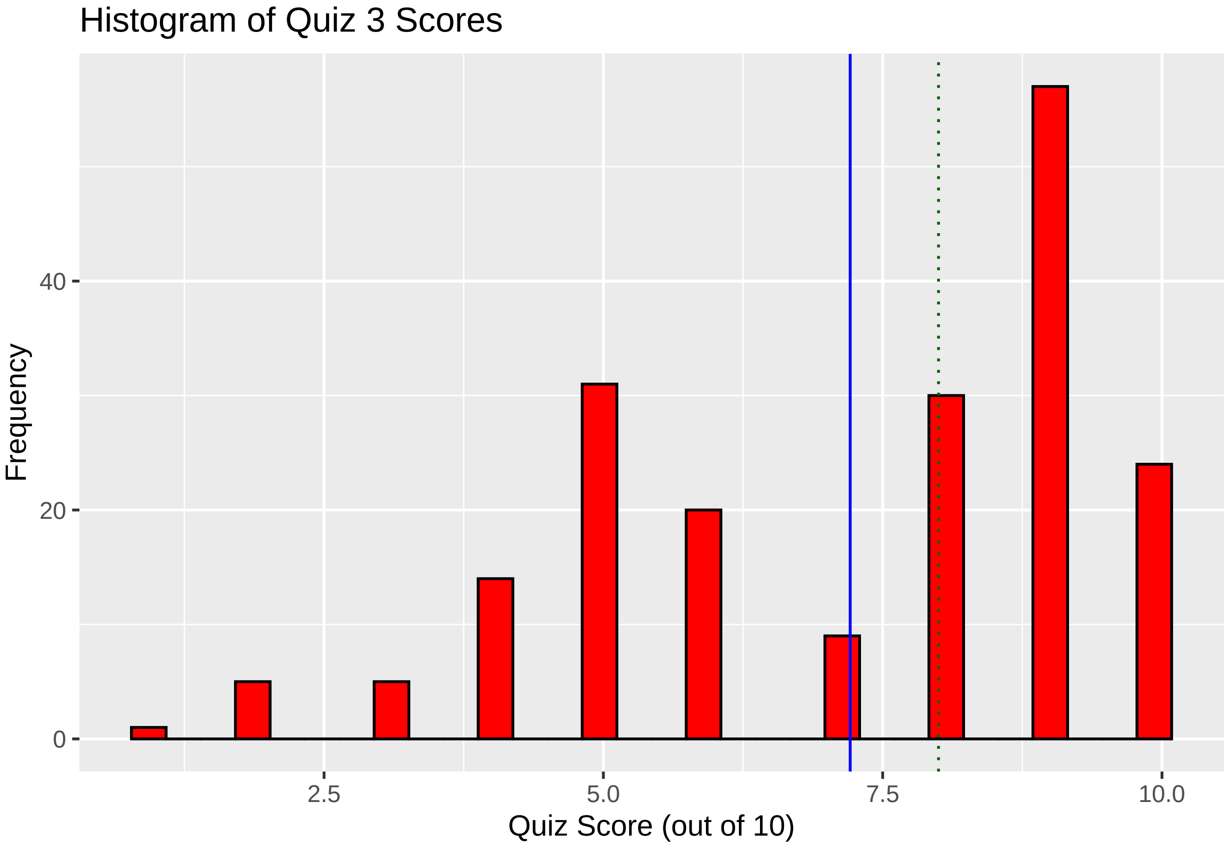

Country vs. Quiz 1 Score
10.0
7.5
5.0
2.5
Score (out of 10)
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA

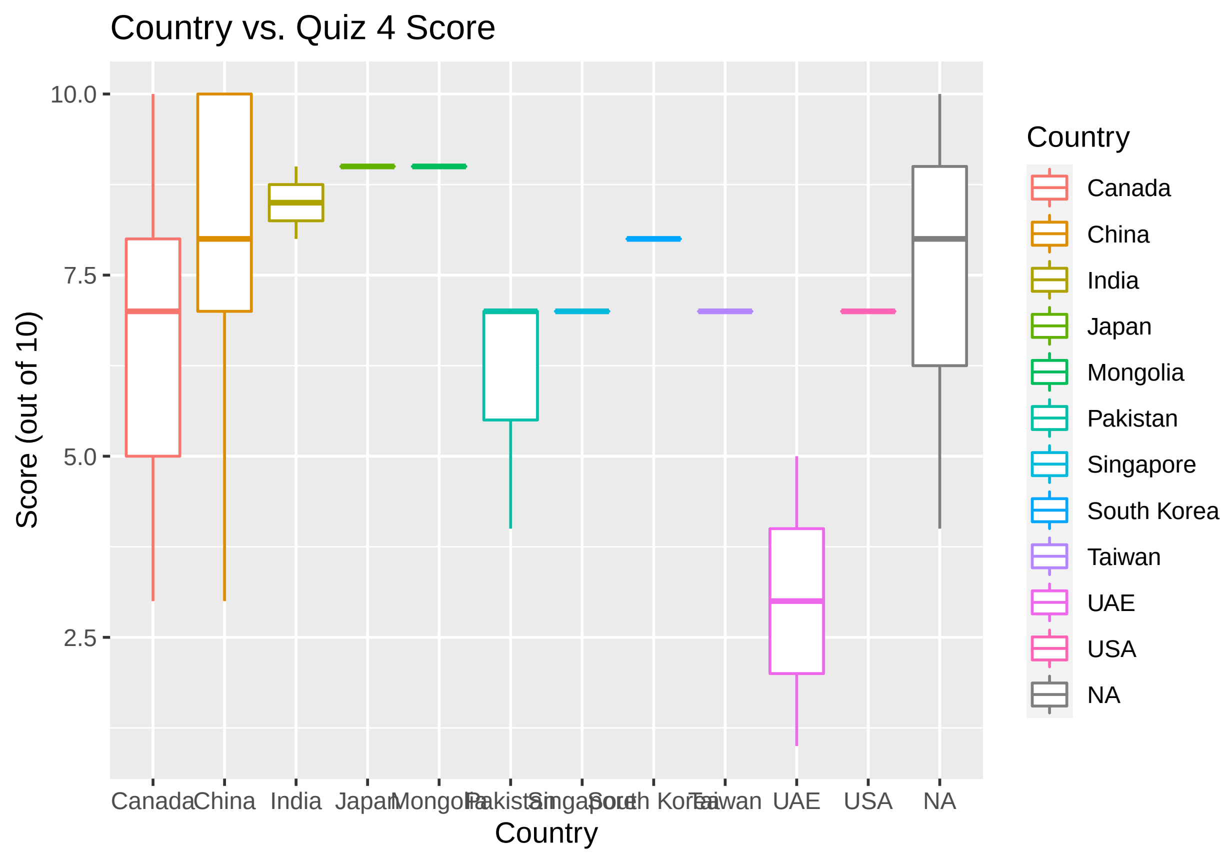

Histogram of Quiz 1 Scores
40
30
20
10
0
Frequency
2.5
5.0
7.5
10.0
Quiz Score (out of 10)


Country vs. Quiz 2 Score
Score (out of 10)
10.0
7.5
5.0
2.5
0.0
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA

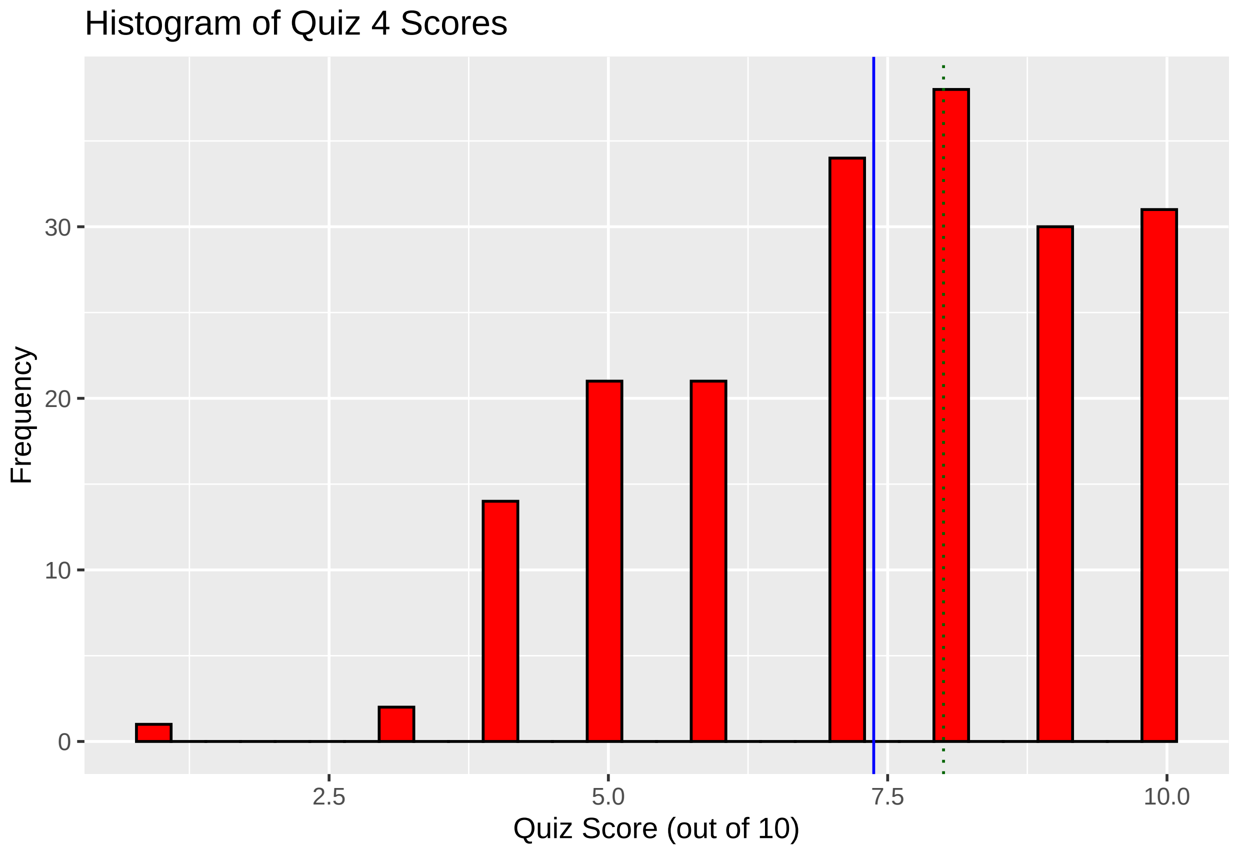

Histogram of Quiz 2 Scores
Frequency
40
30
20
10
0
0.0
2.5
5.0
7.5
10.0
Quiz Score (out of 10)


Country vs. Quiz 3 Score
Score (out of 10)
10.0
7.5
5.0
2.5
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA
Country
Country
Canada
China
India
Japan
Mongolia
Pakistan
Singapore
South Korea
Taiwan
UAE
USA
NA


Histogram of Quiz 3 Scores
Frequency
40
20
0
2.5
5.0
7.5
10.0
Quiz Score (out of 10)



Country vs. Quiz 4 Score



Histogram of Quiz 4 Scores

5-Number Summary Statistics

```
summary(cleaned_sta302_performance_data2$COVID.hours..W1.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     0.0     1.0     1.0     3.7     2.0   168.0      21
```

```
summary(cleaned_sta302_performance_data2$COVID.hours..W2.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   1.000   2.869   2.000  40.000      19
```

```
summary(cleaned_sta302_performance_data2$COVID.hours..W3.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   0.500   1.000   2.227   2.000  24.000      11
```

```
summary(cleaned_sta302_performance_data2$COVID.hours..W4.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   1.500   2.917   3.000  50.000      13
```

```
summary(cleaned_sta302_performance_data2$STA302.hours..W1.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   5.000   7.000   7.539   9.000  28.000      21
```

```
summary(cleaned_sta302_performance_data2$STA302.hours..W2.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   1.000   6.000   8.000   8.403  10.000  20.000      19
```

```
summary(cleaned_sta302_performance_data2$STA302.hours..W3.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    6.00    9.00    9.32   12.00   30.00      10
```

```
summary(cleaned_sta302_performance_data2$STA302.hours..W4.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    2.00    7.00   11.00   13.44   16.00   72.00      13
```

```r
summary(cleaned_sta302_performance_data2$Quiz_1_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   1.000   7.000   8.000   7.738   9.000  10.000       8
```

```r
summary(cleaned_sta302_performance_data2$Quiz_2_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   0.000   5.800   8.800   7.422   9.400  10.000       8
```

```r
summary(cleaned_sta302_performance_data2$Quiz_3_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   1.000   5.000   8.000   7.209   9.000  10.000       3
```

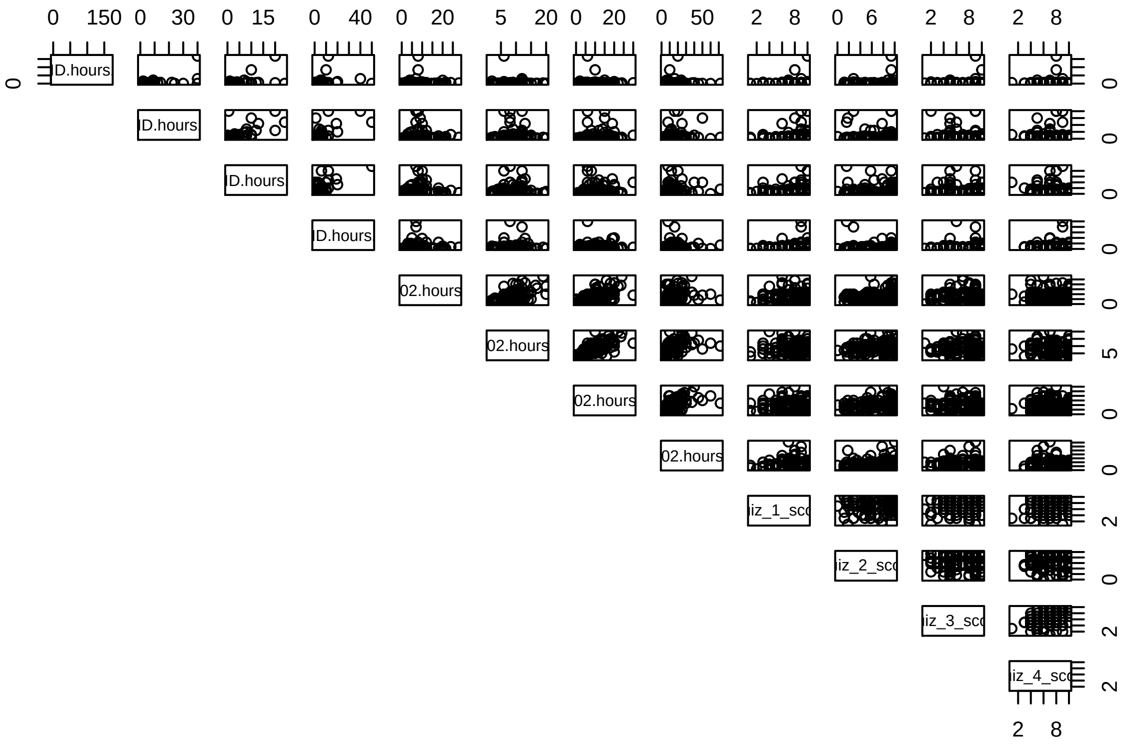
```r
summary(cleaned_sta302_performance_data2$Quiz_4_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   1.000   6.000   8.000   7.375   9.000  10.000       7
```

Scatterplots

Comprehensive pairwise scatterplot

```
pairs(~COVID.hours..W1. + COVID.hours..W2. + COVID.hours..W3. + COVID.hours..W4. +
      STA302.hours..W1. + STA302.hours..W2. + STA302.hours..W3. + STA302.hours..W4. +
      Quiz_1_score + Quiz_2_score + Quiz_3_score + Quiz_4_score,
      data = cleaned_sta302_performance_data2, lower.panel = NULL)
```



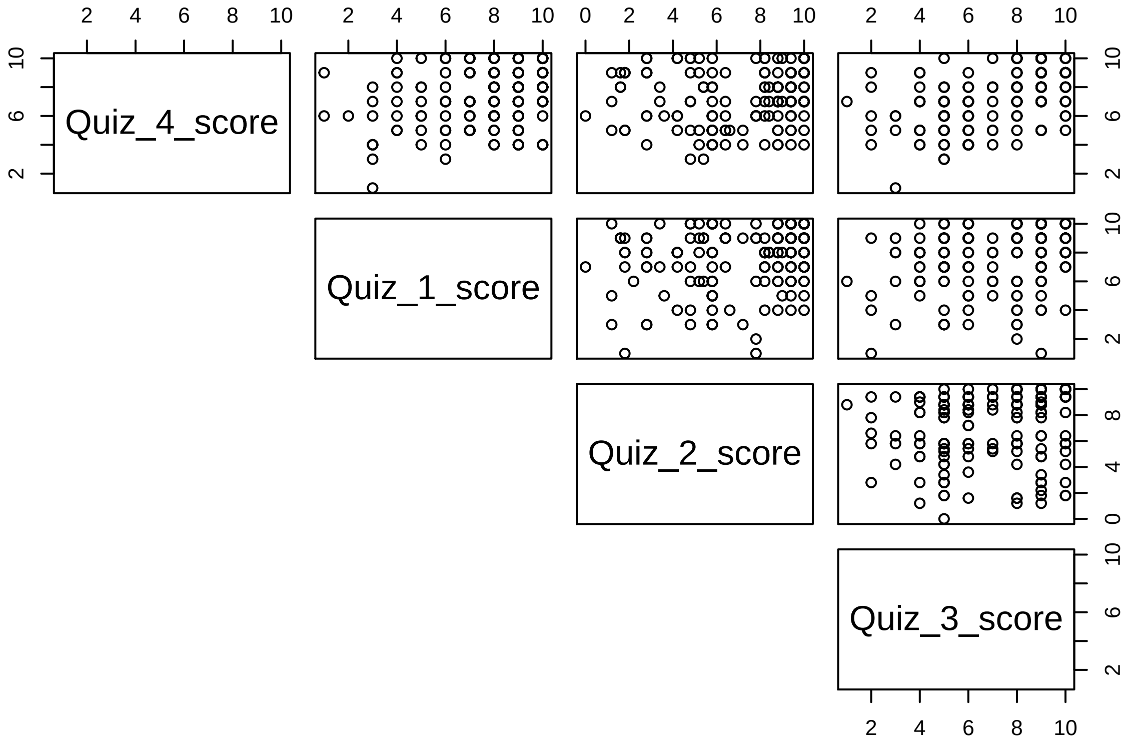
```
## GGally
# ggpairs -- removes bottom half of pairs plot
# ggpairs(data = cleaned_sta302_performance_data2)
```

Slightly Zoomed In Pairwise Scatterplots

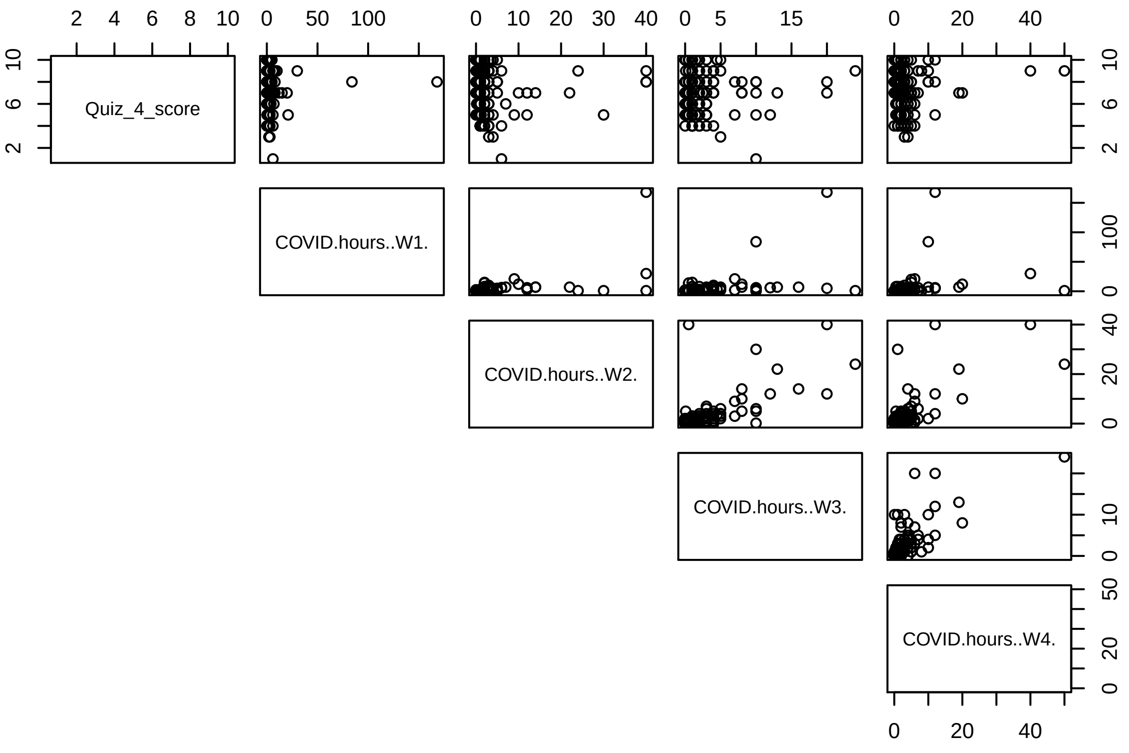
We can zoom in a bit by creating 3 - 4 pairs() functions:

- quiz4 ~ quiz 1, 2, 3
- quiz4 ~ covid 1, 2, 3, 4
- quiz4 ~ sta302h1 1, 2, 3, 4

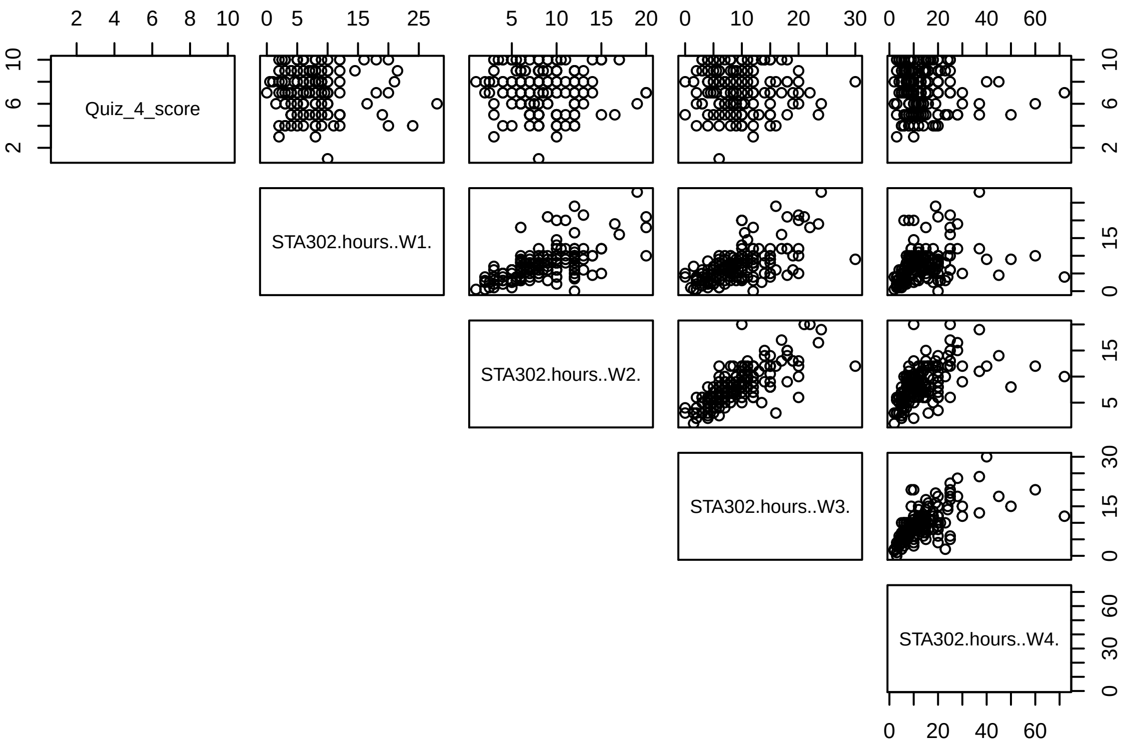
```
pairs(~Quiz_4_score + Quiz_1_score + Quiz_2_score
      + Quiz_3_score, lower.panel = NULL)
```



```
pairs(~Quiz_4_score + COVID.hours..W1. + COVID.hours..W2.
      + COVID.hours..W3. + COVID.hours..W4., lower.panel = NULL)
```



```
pairs(~Quiz_4_score + STA302.hours..W1. + STA302.hours..W2.
      + STA302.hours..W3. + STA302.hours..W4., lower.panel = NULL)
```



Top 4 - 5 Interesting Scatterplots

Let's pick out 4 - 5 scatterplots that have interesting relationships.

I'll back up my choices with their correlation (R value).

Correlation Matrix

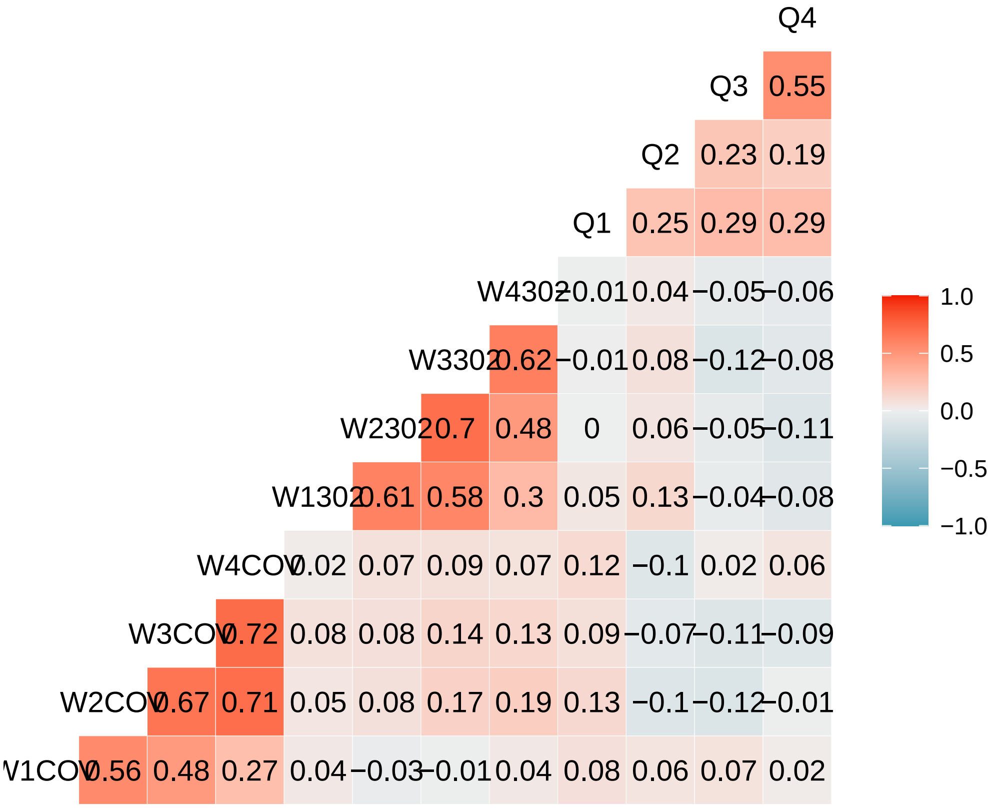
All Countries

We can find correlation matrix to determine candidate significant predictor values.

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
colnames(rows_with_no_country) <- c("W1COV", "W2COV", "W3COV", "W4COV",
                                    "W1302", "W2302", "W3302", "W4302",
                                    "Q1", "Q2", "Q3", "Q4")
ggcorr(rows_with_no_country, label = TRUE, label_round = 2)
```



```
round(cor(rows_with_no_country, use = "pairwise.complete.obs", method = "pearson"), 2)
```

```
##       W1COV W2COV W3COV W4COV W1302 W2302 W3302 W4302    Q1    Q2    Q3    Q4
## W1COV  1.00  0.56  0.48  0.27  0.04 -0.03 -0.01  0.04  0.08  0.06  0.07  0.02
## W2COV  0.56  1.00  0.67  0.71  0.05  0.08  0.17  0.19  0.13 -0.10 -0.12 -0.01
## W3COV  0.48  0.67  1.00  0.72  0.08  0.08  0.14  0.13  0.09 -0.07 -0.11 -0.09
## W4COV  0.27  0.71  0.72  1.00  0.02  0.07  0.09  0.07  0.12 -0.10  0.02  0.06
## W1302  0.04  0.05  0.08  0.02  1.00  0.61  0.58  0.30  0.05  0.13 -0.04 -0.08
## W2302 -0.03  0.08  0.08  0.07  0.61  1.00  0.70  0.48  0.00  0.06 -0.05 -0.11
## W3302 -0.01  0.17  0.14  0.09  0.58  0.70  1.00  0.62 -0.01  0.08 -0.12 -0.08
## W4302  0.04  0.19  0.13  0.07  0.30  0.48  0.62  1.00 -0.01  0.04 -0.05 -0.06
## Q1     0.08  0.13  0.09  0.12  0.05  0.00 -0.01 -0.01  1.00  0.25  0.29  0.29
## Q2     0.06 -0.10 -0.07 -0.10  0.13  0.06  0.08  0.04  0.25  1.00  0.23  0.19
## Q3     0.07 -0.12 -0.11  0.02 -0.04 -0.05 -0.12 -0.05  0.29  0.23  1.00  0.55
## Q4     0.02 -0.01 -0.09  0.06 -0.08 -0.11 -0.08 -0.06  0.29  0.19  0.55  1.00
```

By Individual Country

```
# TODO: You could also create separate correlation matrices for each country.
```