Regression Analysis

Sarah Jafari

2024-11-26

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

```r
options(repos = c(CRAN = "https://cloud.r-project.org"))
install.packages("car")    # Install the package
```

```
## Installing package into 'C:/Users/Owner/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'car' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##   C:\Users\Owner\AppData\Local\Temp\Rtmp8QQAY9\downloaded_packages
```

```r
library(car)        # Load the package
```

```
## Warning: package 'car' was built under R version 4.3.3
```

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

```
## Warning: package 'carData' was built under R version 4.3.3
```

```r
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.3.3
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked 'package:car':
from ## ##
        recode
```

```
## The following objects are masked'package:stats':
from ## ##
            filter,lag
```

```
## The following objects are masked'package:base':
from ## ##
            intersect,setdiff,setequal,         union
```

```
library(ggplot2)
```

```
data <- read.table ("class.data.txt",      header= TRUE)
data
```

```
## IDgendermajorquiz1quiz2quiz3quiz4quiz5 ##1 1
MMath 90 79 90 90 93 ##2 2 MMath 55 60 58 70 79
##3 3 FMath 60 72 75 80 77 ##4 4 MMath 66 48 89
70 72 ##5 5 FMath 63 60 54 55 61 ##6 6 MMath 61 48
63 60 83 ##7 7 MMath 40 42 83 80 56 ##8 8 MMath
50 44 11 60 71 ##9 9 MMath 75 80 93 90 85 ##1010
FMath 57 64 68 65 71 ##1111 MMath 71 71 87 86 93
##1212 MStat 93 94 97 92 94 ##1313 MStat 70 81 87 90
87 ##1414 FStat 67 87 82 92 92 ##1515 MStat 62 74 70
85 82 ##1616 FStat 72 67 63 60 74 ##1717 MStat 91 80
83 90 81 ##1818 FStat 91 76 87 70 86 ##1919 MStat 65
82 63 60 82 ##2020 MStat 62 57 84 65 47 ##2121
MStat 61 56 73 65 79 ##2222 MStat 38 68 43 92 81
##2323 MStat 75 80 100 86 85 ##2424 MStat 72 79 83
60 86 ##2525 MStat 48 51 73 75 67 ##2626 MStat 48
70 73 80 68 ##2727 FStat 78 79 66 80 76 ##2828
FStat 43 67 69 75 88 ##2929 FStat 60 44 45 60 63
##3030 MStat 35 46 37 70 38 ##3131 FStat 31 61 52 75
77 ##3232 FStat 74 74 79 86 79 ##3333 FStat 79 82
97 88 91 ##3434 FStat 72 68 80 90 76 ##3535 FStat
81 94 87 90 82 ##3636 FStat 71 57 48 60 86 ##3737
FComp 53 59 70 75 73 ##3838 MComp 60 77 84 85
87 ##3939 FComp 61 65 73 65 69
```

```
##4040        F  Co      4      5      9      9      5
##4141        M  mp      8      5      3      2      6
##4242        F  Co      7      71     9      9      8
##4343        M  mp      3      7      0      2      3
##4444        F  Co      6      6      5      5      6
##4545        F  mp      4      6      3      5      7
##4646        F  Co      6      5      8      9      6
##4747        F  mp      5      9      7      2      9
##4848        M  Co      9      6      9      8      9
##4949        F  mp      7      8      0      6      9
##5050        F  Co      9      8      9      9      9
##5151        M  mp      4      5      7      2      6
##5252        F  Co      4      5      4      7      4
                 mp      8      7      7      5      4
                 Co      8      9      6      7      9
# Summary statistics mp  0      5      3      5      3
summary(data[, 4:8]) Co  51     4      6      6      7
                 mp      7      8      0      0      9
                 Co      3      9      9      8      8
#       quiz1    mp    quiz2 6  7      0  quiz3 5   9       quiz4
#   Min.    :31.00 Co Min.  4 :42.00 3  Min. 8  11.00 8  Min.    :55.00
#   1stQu.:56.50  mp 1stQu.:6 57.00 5   1st Qu.5 63.00 4  1st  Qu.:65.00
#   Median 64.50  Co Median 2:70.50 4   Median 8 77.00 71  Median:80.00
#   Mean 65.60    mp Mean 91 :69.04 9   Mean 5  73.88 9  Mean    :77.42
#   3rdQu.:74.25  Co  3rdQu.:79.25 2    3rd Qu.8  87.25 5  3rd  Qu.:88.50
#   Max.    :97.00 mp Max.    :96.00    Max. 0  500.00    Max.      :92.00
#   Min.  quiz5
#   1stQu.:71.00  :38.00
#
#
#   Median:80.00
#   Mean    :77.67
#     3rdQu.:86.25
#
#   Max.    :99.00
#
```

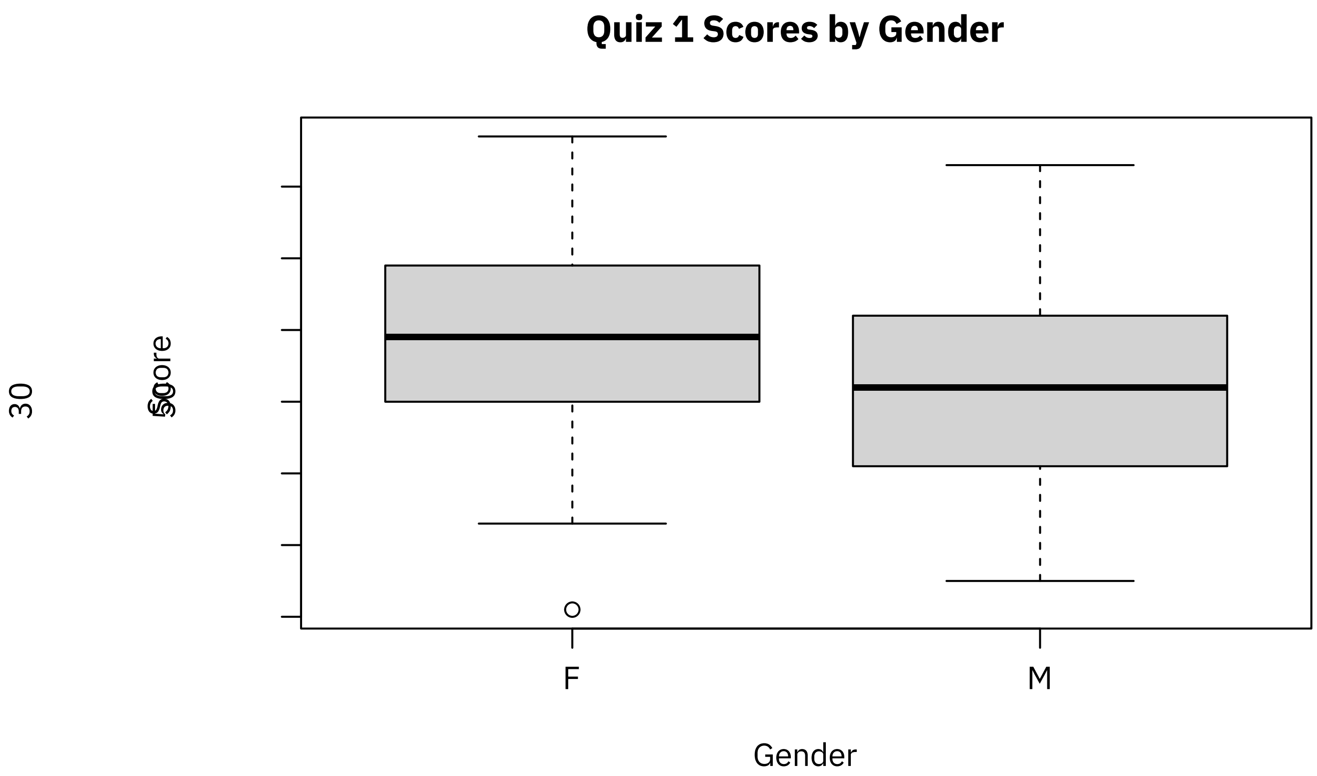
```
# by gender
#
gender_summary aggregate(. ~    gender, data =  data[, c("gender",    "quiz1",    "quiz2",    "quiz3", "quiz4", "q
#
print(gender_summary)
#
#
```

```
##    gender     quiz1       quiz2       quiz3      quiz4      quiz5
##1              F68.1538572.2692373.30769 76.76923 78.50000
##2           M63.0384665.8076974.46154 78.0769276.84615
#
#
```

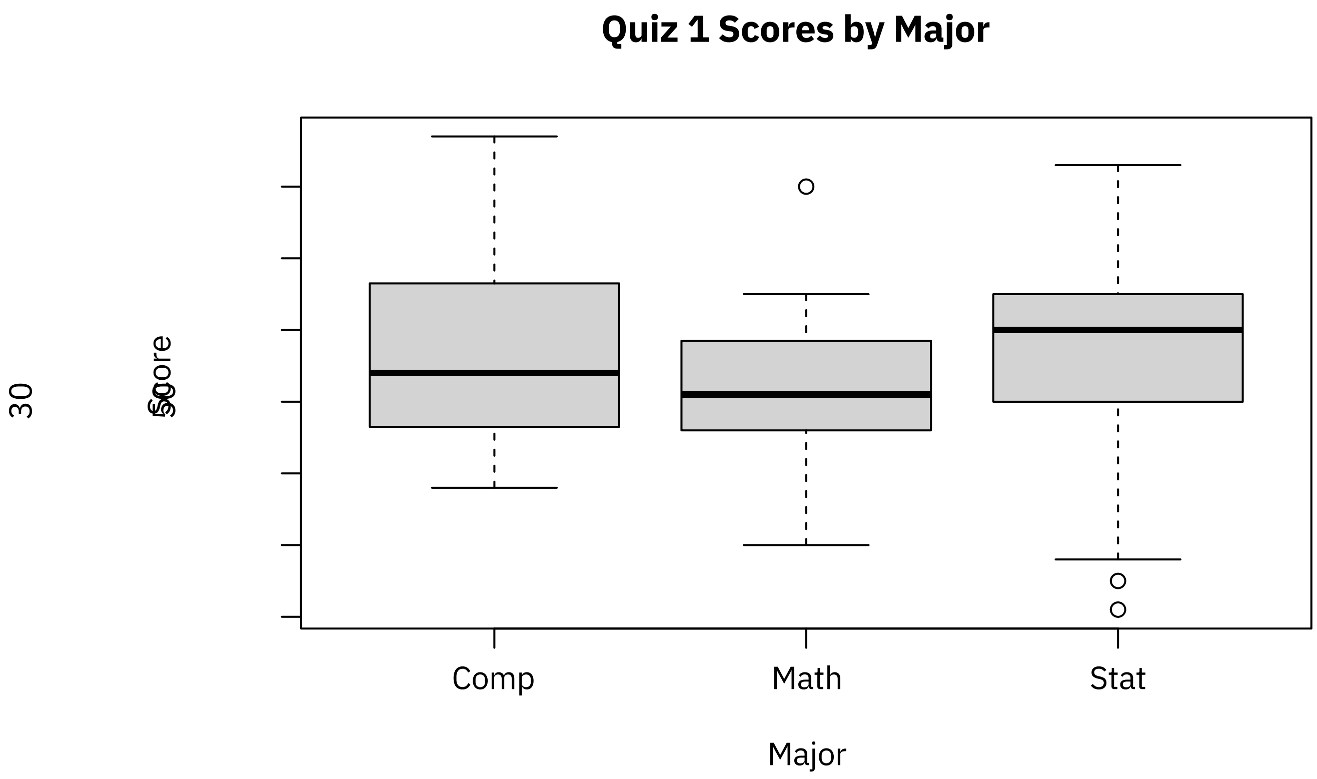
```
# by major
#
major_summary <- aggregate(. ~ major,  data =  data[, c("major",     "quiz1",     "quiz2",    "quiz3", "quiz4", "quiz
#
print(major_summary)
```

```
##    major      quiz1       quiz2       quiz3      quiz4       quiz5
## 1  Comp67.75000 71.75000 78.12500 80.2500078.18750
##     Math  62.54545 60.72727 70.0909173.27273 76.45455
2        Stat   65.56000 70.96000 72.8400077.4400077.88000
##
3
```

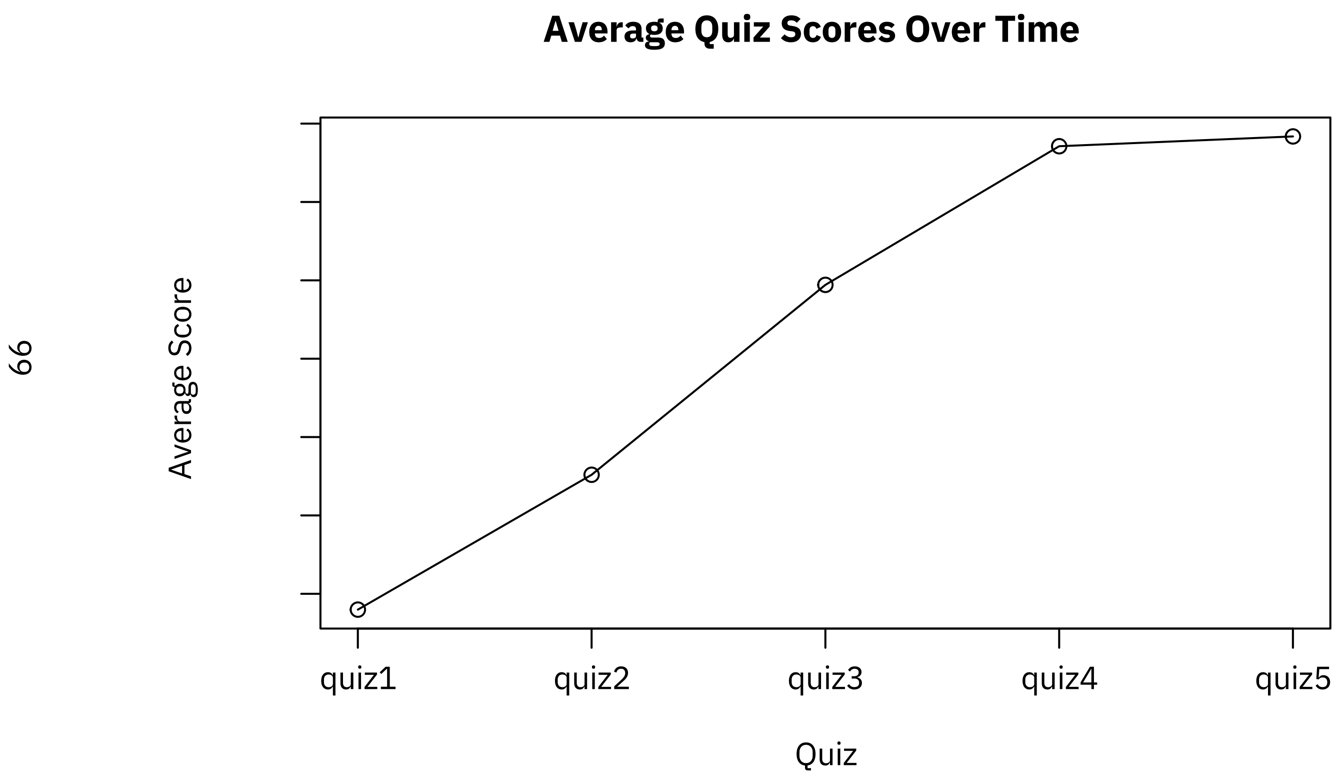
```
# boxplotr
boxplot(quiz1 ~      gender, data = data,   main=  "Quiz 1  Scores by Gender", xlab  =  "Gender", ylab = "Score")
```



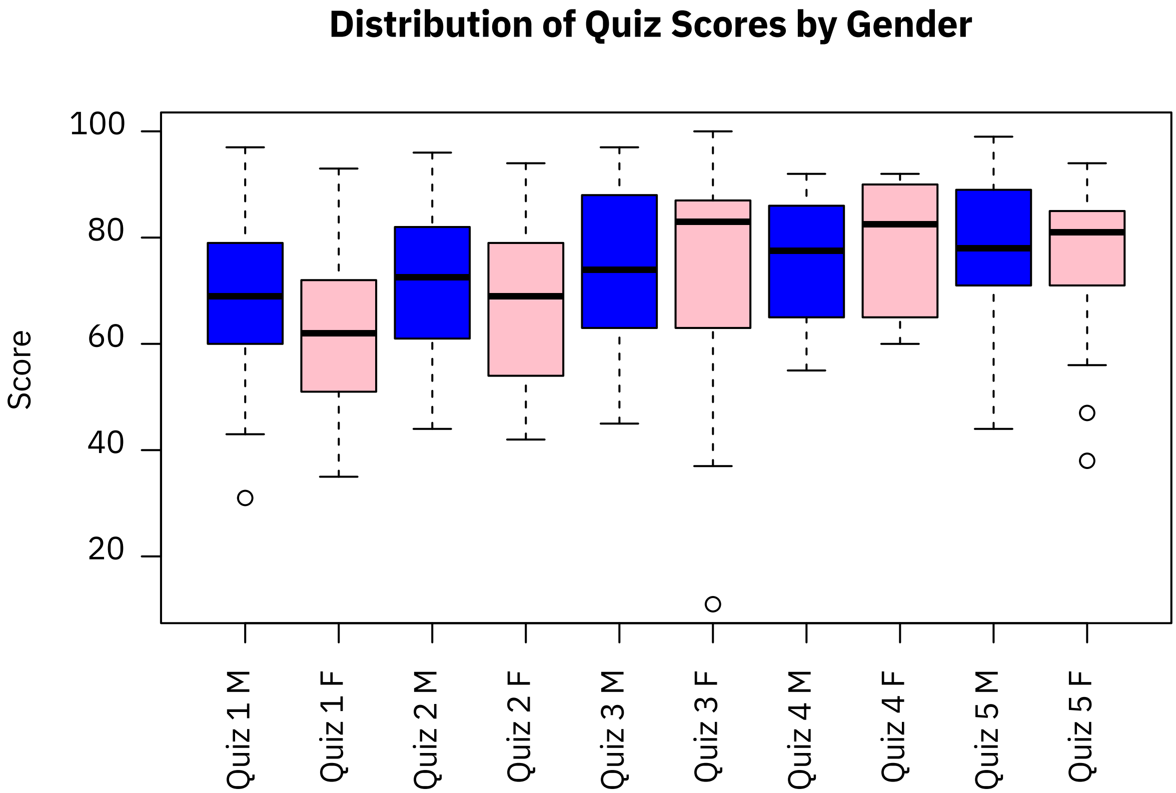
```
# boxplot by major
boxplot(quiz1 ~ major, data = data, main = "Quiz 1 Scores by Major", xlab = "Major", ylab = "Score")
```



```
# line plot to show improvement of scores with time
average_scores <- aggregate(data[, 4:8],   by = list(data$gender),   FUN=  mean)
quiz_means <- colMeans(data[, 4:8])
quiz_names <- names(data[, 4:8])
plot(quiz_means, type = "o", xaxt = "n",        main=  "AverageQuiz Scores OverTime",  xlab = "Quiz", ylab = "
axis(1, at      = 1:5, labels = quiz_names)
```



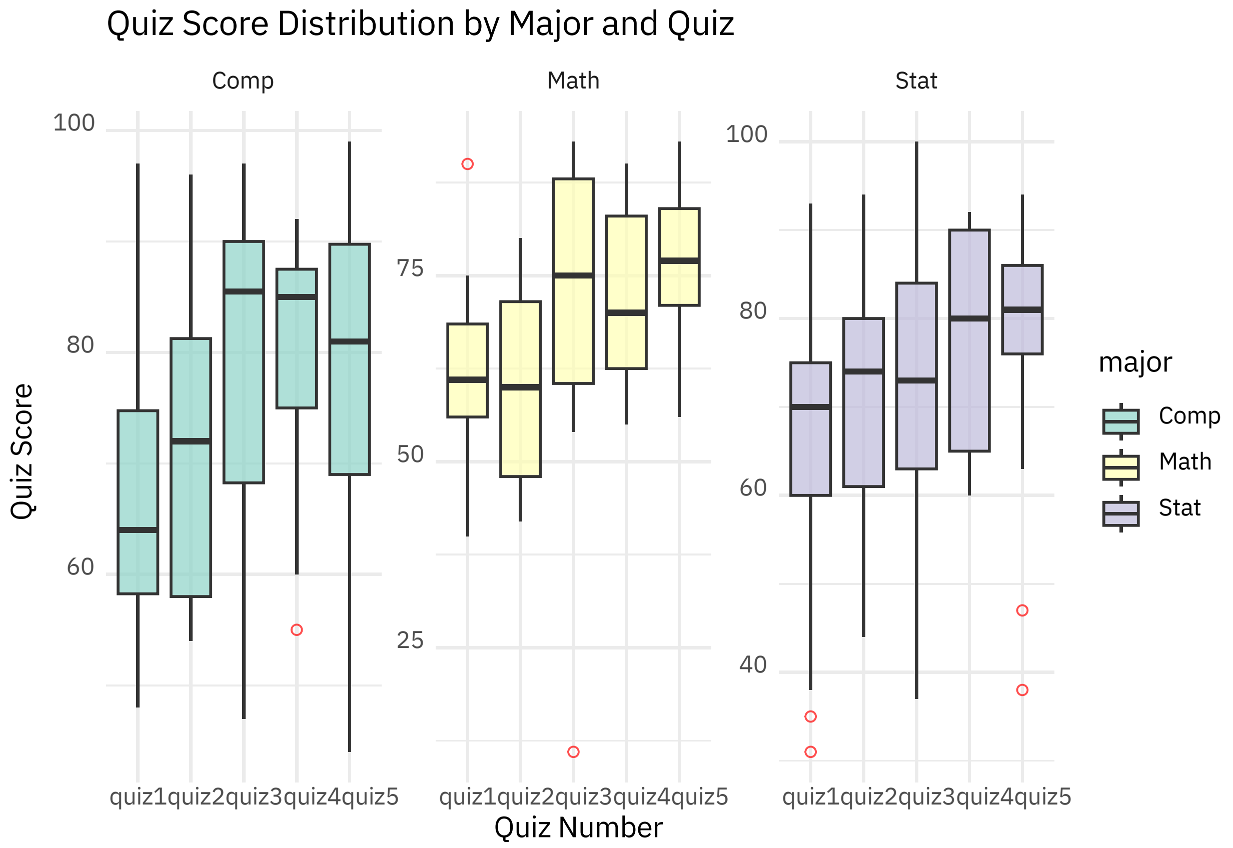
```
#Visualizing Gender-Based PerformanceTrends
boxplot(c(data$quiz1, data$quiz2, data$quiz3,  data$quiz4, data$quiz5) ~
        rep(data$gender, times = 5) +          "Quiz                     each
        rep(c("Quiz 1", "Quiz 2",      "Quiz 3",          4",  "Quiz 5"),        = nrow(data)),
      main = "Distribution of Quiz   Scores by Gender",
      xlab = "", ylab = "Score",
      col = c("blue", "pink"),
      names =   c("Quiz 1 M", "Quiz 1  F",  "Qui   2  M",  "Qui   2  F",
                  "Quiz 3 M", "Quiz 3  F",  z        4  M",  z        4  F",
                 "Quiz 5 M", "Quiz  5  F"),  "Qui             "Qui
      las = 2)                               z               z
```



```
# Calculate average quiz scores for each major
avg_scores_major <- aggregate(cbind(quiz1,quiz2, quiz3, quiz4, quiz5) ~ major, data = data, mean)
print(avg_scores_major)
```

```
## major quiz1 quiz2 quiz3 quiz4 quiz5 ## 1 Comp
67.75000 71.75000 78.12500 80.25000 78.18750 ## 2 Math
62.54545 60.72727 70.09091 73.27273 76.45455 ## 3 Stat
65.56000 70.96000 72.84000 77.44000 77.88000
```

```
data_long <- data %>%
    pivot_longer(cols = starts_with("quiz"),
                 names_to = "QuizNumber",
                 values_to = "Score")
ggplot(data_long, aes(x = QuizNumber, y = Score, fill = major)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 1, alpha = 0.7) +
  labs(title = "Quiz Score Distribution by Major and Quiz",
       x = "Quiz Number", y = "Quiz Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3") +
  facet_wrap(~ major, scales = "free_y")
```

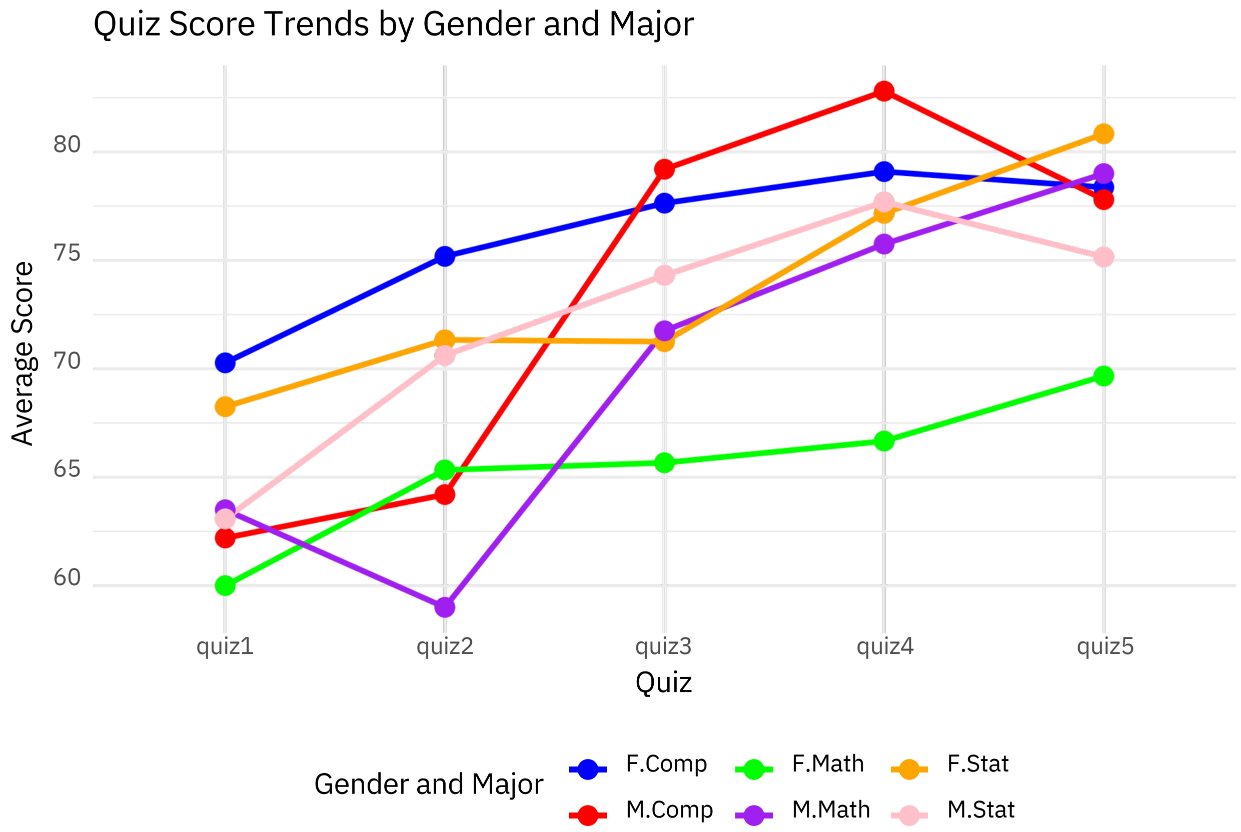


```
# Calculate the average score per quiz by genderandmajor
trend_data <- data_long %>%
  group_by(QuizNumber, gender, major) %>%
  summarise(Average_Score = mean(Score, na.rm = TRUE), .groups = "drop")
```

```
# Reshape the data from wide to longformat
data_long <- data %>%
pivot_longer(
    cols = starts_with("quiz"),
    names_to = "QuizNumber",
    values_to = "Score"
  )%>%
  mutate(QuizNumber = as.numeric(gsub("quiz", "", QuizNumber))) # Convert QuizNumber to numeric

# Convert categorical variables to factors
data_long$gender <- as.factor(data_long$gender)
data_long$major <- as.factor(data_long$major)
```

```
# Line Plot: trend by gender and major
ggplot(trend_data, aes(x = QuizNumber, y = Average_Score, group= interaction (gender, major), color = ir
  geom_line(size = 1)  +
  geom_point(size = 3)  +
  labs(title = "Quiz Score Trends by Genderand Major",
       x = "Quiz", y = "Average Score", color = "Gender and Major") +
  theme_minimal() +
scale_color_manual(values = c("blue", "red", "green", "purple", "orange", "pink", "cyan", "yellow")) +
theme(legend.position = "bottom")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



```
# Fit the Base Model
base_model <- lm(Score ~ QuizNumber + gender + major, data = data_long)
summary(base_model)
```

```
##
## Call:
## lm(formula = Score ~ QuizNumber + gender + major, data = data_long)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -57.380 -10.184   1.003  11.042  28.128
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
##(Intercept)     65.7244##QuizNumber2.620  25.079   <  2e-16 ***
##genderM          3.2538##majorMath  5.021 9.65e-07 ***
##majorStat ##-0.8749 Signif. codes:0.456    0.6488 *
## ## Residual standard error: 14.78-2.301    0.0222
on                -6.2314   0.6488
                  -2.0950        1.9189    -0.973    0.3314
                                 2.707      0.01      0.05
                                 9                     of
                      0 '***' 0.001 '**'           '*'    '.'     0.1 ' '   1
                                 2.1528          R-squared:
                                                 degrees    freedom
                                            255
## Multiple R-squared:    0.1114, Adjusted                   0.09744
## F-statistic: 7.991 on 4 and 255 DF,       p-value:  4.395e-06
```

```r
vif(base_model)
```

```
##                        GVIFDfGVIF^(1/(2*Df))
##QuizNumber1.0000001          1.000000
##gender    1.0961841    1.046988    ##major
1.0961842 1.023224
```

```r
cor(data_long[, sapply(data_long, is.numeric)])      # Correlations  for  numericpredictorss
```

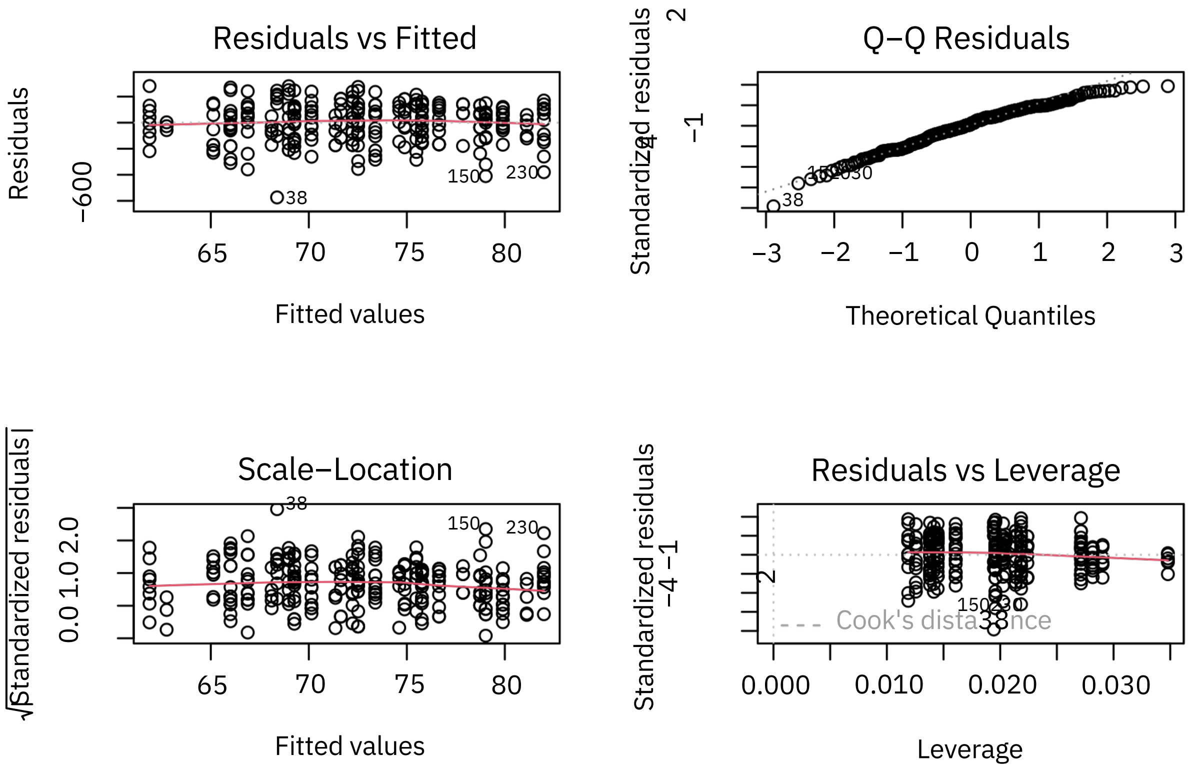
```
##                          IDQuizNumber       Score
##ID              1.00000000  0.00000000.09561389
## QuizNumber 0.00000000 1.0000000 0.29642185
##Score          0.09561389  0.2964218 1.00000000
```

```r
# Fit the Interaction Model
interaction_model <- lm(Score ~ QuizNumber *gender+ QuizNumber* major + gender* major, data = data_l

summary(interaction_model)
```

```
##
## Call:
## lm(formula = Score ~ QuizNumber * gender+ QuizNumber* major +
##     ##gender ##*major,data=data_long)
Residuals:

##      Min       1QMedian      3Q      Max
##  -58.80  -10.41  ##1.70##    11.52     29.02
Coefficients:     ##      ##
(Intercept)
##QuizNumber                EstimateStd.  Error  t  value    Pr(>|t|)
##genderM                   68.5824       4.2428    16.165    <2e-16 ***
##majorMath                  2.5089       1.2473     2.011    0.0454 * .
##majorStat                 -6.9837       5.4304    -1.286     0.1996
##QuizNumber:gender         -12.2324      7.2043    -1.698     0.0908
M                            -2.0121       5.3548    -0.376     0.7074
##QuizNumber:major           1.3715       1.3619     1.007     0.3149
Math                         0.5300       1.9218     0.276     0.7829
##QuizNumber:majorS          -0.1101      1.5278    -0.072     0.9426
tat                          7.2024       5.7410     1.255     0.2108
##genderM:majorMath          1.2717       4.4549     0.285     0.7755
##genderM:majorStat
```

```
## ---
## Signif. codes:   0 '***' 0.001 '**'     0.01  '*'  0.05 '.'   0.1 '  ' 1
##                                                  degrees of freedom
## Residual standard error: 14.83 on  250 R-squared:
## Multiple R-squared:  0.1224, Adjusted                      0.09084
## F-statistic: 3.875 on 9 and 250 DF,    p-value:  0.0001283
```

```
# Residual diagnostics for the base model
par(mfrow = c(2, 2))
plot(base_model)
```



```
library (lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.3.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.3.3

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
#Breusch-Pagan test
bptest(base_model)
```

```
##     ##
##  stu## dentizedBreusch-Pagantest
data:
                base_model
## BP = 2.4923, df = 4, p-value = 0.646
```

```
# Durbin Watson Test
durbinWatsonTest(base_model)
```

```
## ## lag## Autocorrelation D-W Statistic p-value
        1      0.4698759       1.04424          0
# Introducing a a lagged term for quiz
      Alternative hypothesis: rho != 0
```

```
                                                          scores
data_long$LaggedScore <- lag(data_long$Score)
base_model_lagged <- lm(Score ~ QuizNumber + LaggedScore + gender + major, data = data_long)
summary(base_model_lagged)
```

```
##
## Call:
## lm(formula = Score ~ QuizNumber + LaggedScore + gender + major,
##    ## data ## data_long)
Residuals:

## Min                 1Q    Median      3Q      Max
##-46.802   -8.156    1.626      8.824   30.154
##
## Coefficients:
##    ##(Intercept) Estimate ## Std. Error QuizNumber value    Pr(>|t|)
##LaggedScore 34.9234   ## genderM 4.572  7.638   4.55e-13 ***
##majorMath  ## 2.9968 t Stat ## 6 --- ##  5.129  5.81e-07 ***
Signif.    codes:   0.4192 ##   Residual 0.584  7.778   1.87e-13 ***
standard error: 13 -0.2.6405 on     4           -0.372      0.710
                         -3.9166        0.053      -1.592       0.113
                         -1.2442        9          -0.645       0.520
                                        1.7197       0.01        0.05
                                        2.460                    of
                      0 '***' 0.001 '**'           '*'              '.'      0.1 '   '   1
                                        4
                                                      degrees       freedom
                                        1.9303
                                                    253                  0.2759
##    (1observationdeletedduetomissingness)
## Multiple R-squared:   0.2899, Adjusted  R-squared:
## F-statistic: 20.66 on 5 and 253 DF,        p-value:  <  2.2e-16
```

```
# DW test again
durbinWatsonTest(base_model_lagged)
```

```
#      lag Autocorrelation D-W Statistic p-value
#       1         -0.0255119         2.049537     0.898
#     Alternative hypothesis: rho != 0
#
#
#
```

```
# Fit the Polynomial Model
polynomial_model <- lm(Score ~ poly (QuizNumber, 2) * gender * major, data = data_long)
summary(polynomial_model)
```

```
##
## Call:
## lm(formula = Score ~ poly(QuizNumber, 2) * gender * major, data = data_long)
##
## Residuals:
## Min           1Q    Median      3Q      Max
##-57.836   -9.195    2.293   10.537   30.000
##
## Coefficients:
##                                          Estimate Std. Error t value  Pr(>|t|)
##(Intercept)                                 76.109      2.013  37.808   <2e-16 ***
##poly(QuizNumber,2)1                         45.814     32.459   1.411   0.1594
##poly(QuizNumber,2)2                        -23.653     32.459  -0.729   0.4669
##genderM                                     -2.869      3.601  -0.797   0.4264
##majorMath                                  -10.642      4.349  -2.447   0.0151 *
##majorStat                                   -2.342      2.787  -0.841   0.4015
##poly(QuizNumber,2)1:genderM                 67.747     58.064   1.167   0.2445
##poly(QuizNumber,2)2:genderM                -25.300     58.064  -0.436   0.6634
##poly(QuizNumber,2)1:majorMath                1.313     70.119   0.019   0.9851
##poly(QuizNumber,2)2:majorMath               15.944     70.119   0.227   0.8203
##poly(QuizNumber,2)1:majorStat               24.877     44.937   0.554   0.5804
##poly(QuizNumber,2)2:majorStat               37.465     44.937   0.834   0.4053
##genderM:majorMath                            7.202      5.779   1.246   0.2139
##genderM:majorStat                            1.272      4.484   0.284   0.7770
##poly(QuizNumber,2)1:genderM:majorMath       -5.988     93.184  -0.064   0.9488
##poly(QuizNumber,2)2:genderM:majorMath       46.017     93.184   0.494   0.6219
##poly(QuizNumber,2)1:genderM:majorStat      -67.221     72.310  -0.930   0.3535
##poly(QuizNumber,2)2:genderM:majorStat      -27.947     72.310  -0.386   0.6995
## ---
## Signif. codes:
##                    0 '***' 0.001 '**'   0.01  '*'  0.05 '.'   0.1   ' '  1
## Residual standard error: 14.93 on 242 degrees of freedom
## Multiple R-squared:  0.1392, Adjusted R-squared:  0.07874
## F-statistic: 2.302 on 17 and 242 DF,  p-value: 0.002988
```

```
# Extract influential points
influential_points <- c(38,148, 150)
data_long[influential_points, ]
```

```
## # A tibble: 3 x 6
## ## ##ID ##2 ##3 gender major QuizNumber Score LaggedScore
       <int><fct>     <fct>         <dbl><int>      <int>
          8M         Mat             3     11         44
         30M         h               3     37         46
         30M         Stat            5     38         70
                     Stat
```

```
# Fit the model without influential points
model_no_influential <- lm(Score ~ QuizNumber + gender + major, data = data_long[-influential_points, ])

# Compare summaries of the original and new models
summary(base_model) # Original model
```

```
##
## Call:
## lm(formula = Score ~ QuizNumber + gender + major, data = data_long)
##
## Residuals:
## 1Q      Min             Median     3Q      Max
## -57.380 -10.184   1.003    11.042   28.128
##
## Coefficients:
##     ##(Intercept)Estimate##QuizNumberStd.Errort value    Pr(>|t|)
##genderM       65.7244##majorMath2.620 25.079   <  2e-16 ***
##majorStat ## --- 3.2538 Signif. 7 codes: 5.021 9.65e-07 ***
## ## Residual standard -0.8749 error: 0.648 14.78 -0.456    0.6488 *
on                  -6.2314       0            -2.301     0.0222
                    -2.0950       1.9189       -0.973      0.3314
                                  2.707          0.01       0.05
                                  9                           of
                  0 '***' 0.001 '**'                 '*'          '.'     0.1 '   '  1
                                  2.1528           R-squared:
                                                   degrees       freedom
                                          255
## Multiple R-squared:    0.1114, Adjusted                      0.09744
## F-statistic: 7.991 on 4 and 255 DF,        p-value:  4.395e-06
```

```
summary(model_no_influential)   # without influential   points
```

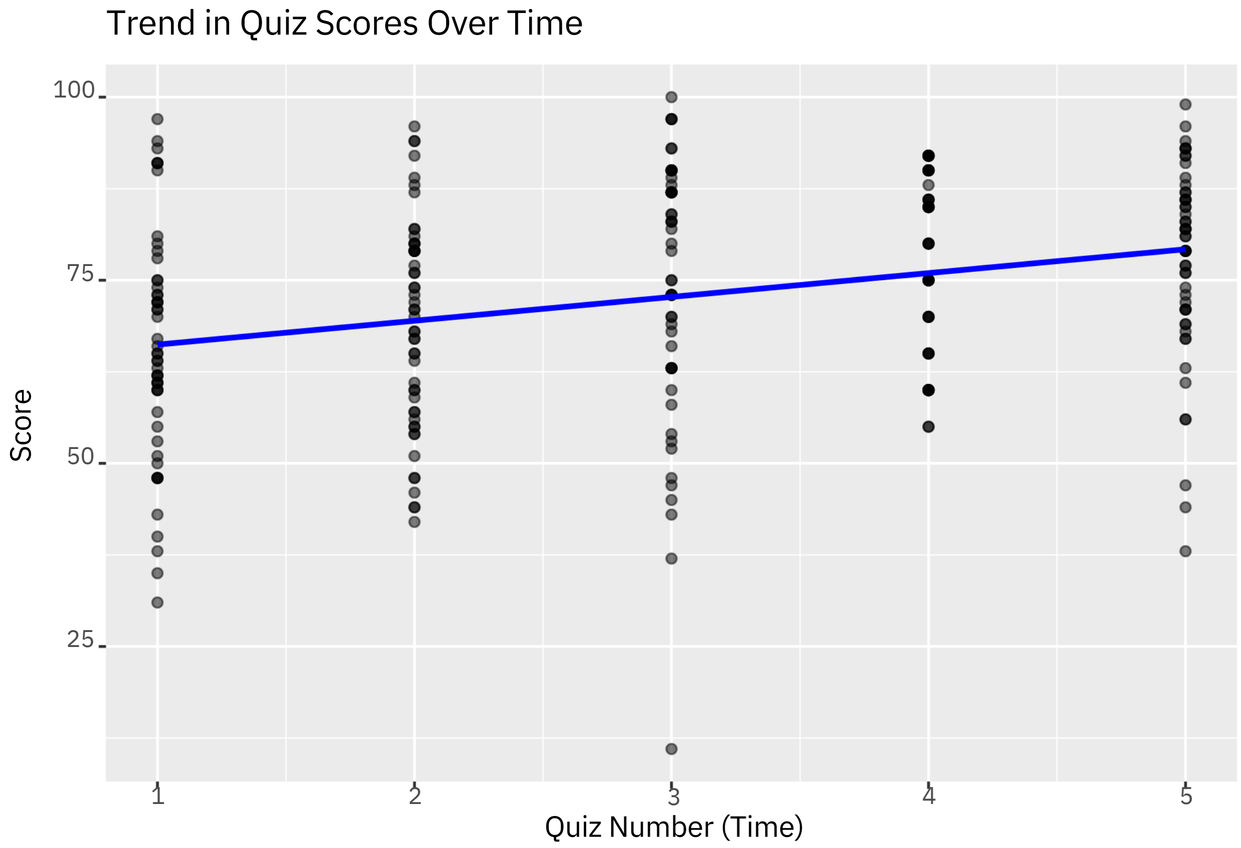
```
##
## Call:
## lm(formula = Score ~ QuizNumber + gender + major, data = data_long[-influential_points,
##     ##])    ##
Residuals:

## Min                1Q   Median     3Q      Max
##-38.036   -9.133     0.581    10.832   28.632
##
## Coefficients:
##                   EstimateStd.Errort      value    Pr(>|t|)
## (Intercept) 64.95072       2.4882      26.103   <  2e-16 ***
##QuizNumber 3.41699   ##6genderM    5.551    7.2e-08 ***
##majorMath ##0.03456 majorStat ## 0.61559 --- ## 0.019     0.9849 *
Signif.    codes: -5.54148## 1.8242 Residual -2.153     0.0323
standard error: 13.98 -1.58165      8          -0.809     0.4192
                                  2.57361        0.01        0.05
                                  2.04118                     of
                  0 '***' 0.001 '**'                 '*'          '.'     0.1 '   '  1
                                                   R-squared:
                                                   degrees       freedom
                                          252                   0.1109
## Multiple R-squared:   0.1248, Adjusted
## F-statistic: 8.985 on 4 and 252 DF,       p-value:  8.471e-07
```

```
# Compare with previous models usingAIC
AIC(base_model, interaction_model, polynomial_model)
```

```
##                       df        AIC
##base_model               62145.184
## interaction_model 11 2151.932
## polynomial_model19 2162.912
```

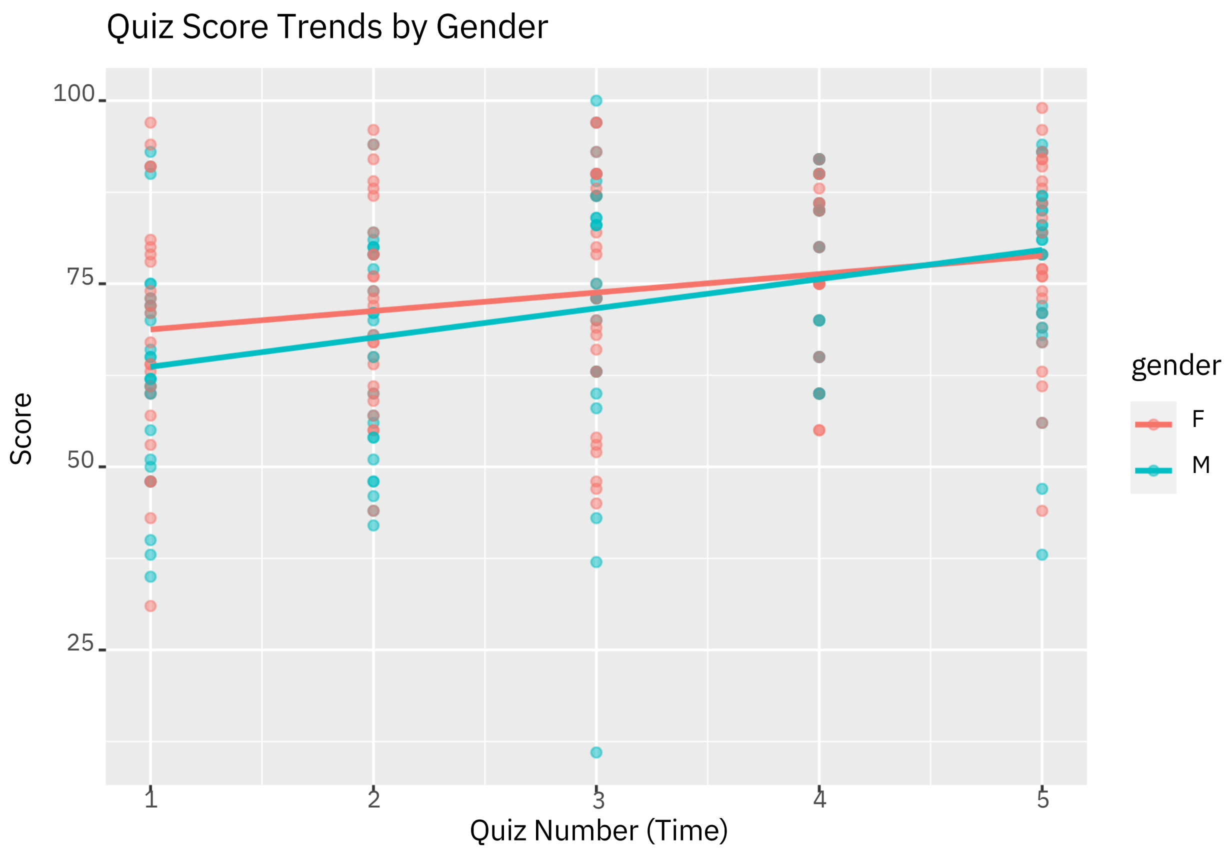
```
# Overall trend in quiz scores
library(ggplot2)
ggplot(data_long, aes(x = QuizNumber, y = Score)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Trend in Quiz Scores Over Time",
       x = "Quiz Number (Time)",
       y = "Score")
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



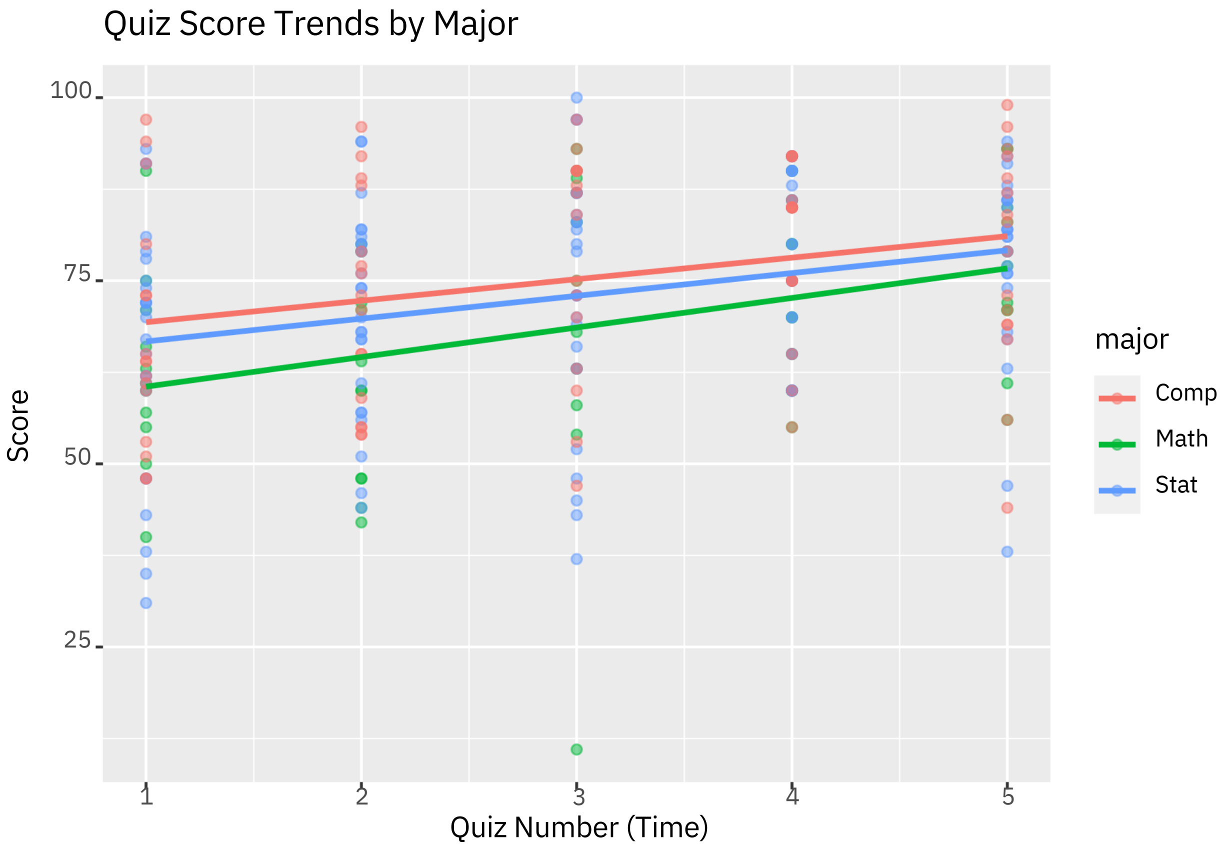
```
#     Trend     by
gender            aes(x = QuizNumber, y = Score, color = gender)) +
ggplot(data_long
, geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Quiz Score Trends by Gender",
       x = "Quiz Number (Time)",
       y = "Score")
```

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



```
# Trend by major
ggplot(data_long, aes(x = QuizNumber,y = Score, color = major)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Quiz Score Trends by Major",
       x = "Quiz Number (Time)",
       y = "Score")
```

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


Quiz Score Trends by Major
Score
100
75
50
25
1
2
3
4
5
Quiz Number (Time)
major
Comp
Math
Stat