Educational Technology Project - EduTech (Fall 2015, Spring 2016 and Fall 2016) Data Analysis

Process Data

```
# Set cwd
setwd("D:/Documents/Data Science/Educational Technology/R/EduTech")
#setwd("E:/Educational Technology/R/EduTech")
getwd()

# Load libraries
library(plyr)
library(tools)
library(ggplot2)

# Read in survey data sets
survey_fall15_soc = read.csv('Survey_CS6460_FALL15_SOC.csv')
survey_fall15_qc = read.csv('Survey_CS6460_FALL15_QC.csv')
survey_fall15_mc = read.csv('Survey_CS6460_FALL15_MC.csv')
survey_fall15_eoc = read.csv('Survey_CS6460_FALL15_EOC.csv')

survey_spr16_soc = read.csv('Survey_CS6460_SPR16_SOC.csv')
survey_spr16_qc = read.csv('Survey_CS6460_SPR16_QC.csv')
survey_spr16_mc = read.csv('Survey_CS6460_SPR16_MC.csv')
survey_spr16_eoc = read.csv('Survey_CS6460_SPR16_EOC.csv')

survey_fall16_soc = read.csv('Survey_CS6460_FALL16_SOC.csv')
survey_fall16_qc = read.csv('Survey_CS6460_FALL16_QC.csv')
survey_fall16_mc = read.csv('Survey_CS6460_FALL16_MC.csv')
survey_fall16_eoc = read.csv('Survey_CS6460_FALL16_EOC.csv')

# Read in grade data sets
grades_fall16 = read.csv('Grades_CS6460_FALL16.csv', na.strings="")
grades_spr16 = read.csv('Grades_CS6460_SPR16.csv', na.strings="")
```

```
# Create data subsets containing information of interest and change names
survey_fall15_soc = survey_fall15_soc[, c(1, 2, 3, 4, 5, 7, 8, 10)]
colnames(survey_fall15_soc) = c("student", "age", "gender", "birth", "residence",
                                "language", "english", "education")

survey_spr16_soc = survey_spr16_soc[, c(1, 2, 3, 4, 5, 7, 8, 10)]
colnames(survey_spr16_soc) = c("student", "age", "gender", "birth", "residence",
                               "language", "english", "education")

survey_fall16_soc = survey_fall16_soc[, c(1, 2, 3, 4, 5, 7, 8, 10)]
colnames(survey_fall16_soc) = c("student", "age", "gender", "birth", "residence",
                                "language", "english", "education")

survey_fall15_qc = survey_fall15_qc[, c(1, 2, 3)]
colnames(survey_fall15_qc) = c("student", "conf_p1_post", "conf_p2_pre")
```

```
survey_spr16_qc = survey_spr16_qc[, c(1, 2, 3)]
colnames(survey_spr16_qc) = c("student", "conf_p1_post", "conf_p2_pre")

survey_fall16_qc = survey_fall16_qc[, c(1, 13, 14)]
colnames(survey_fall16_qc) = c("student", "conf_p1_post", "conf_p2_pre")


survey_fall15_mc = survey_fall15_mc[, c(1, 2, 3)]
colnames(survey_fall15_mc) = c("student", "conf_p2_post", "conf_p3_pre")

survey_spr16_mc = survey_spr16_mc[, c(1, 2, 3)]
colnames(survey_spr16_mc) = c("student", "conf_p2_post", "conf_p3_pre")

survey_fall16_mc = survey_fall16_mc[, c(1, 2, 3)]
colnames(survey_fall16_mc) = c("student", "conf_p2_post", "conf_p3_pre")


survey_fall15_eoc = survey_fall15_eoc[, c(1, 2, 11)]
colnames(survey_fall15_eoc) = c("student", "hours", "conf_p3_post")

survey_spr16_eoc = survey_spr16_eoc[, c(1, 2, 10)]
colnames(survey_spr16_eoc) = c("student", "hours", "conf_p3_post")

survey_fall16_eoc = survey_fall16_eoc[, c(1, 6, 14)]
colnames(survey_fall16_eoc) = c("student", "hours", "conf_p3_post")

colnames(grades_spr16) = c("student", "part_total", "part_other",
      "part_feedback", "part_piazza", "paper", "pres", "project",
      "milestone1", "milestone2", "milestone3", "milestone4", "personal",
      "prop", "status", "assign1", "assign2", "assign3", "assign4", "part_raw")

colnames(grades_fall16) = c("student", "part_total", "part_other",
      "part_feedback", "part_piazza", "paper", "pres", "project",
      "milestone1", "milestone2", "personal", "prop", "status", "assign1",
      "assign2", "assign3", "assign4", "part_raw")


# Create function to convert grade scales
grade_covert = function(x){
  if(is.na(x)){
    num = 0
  } else if (x>0){
    num = (x/5)*50 - 5 + 50
  } else{
    num = 0
  }
  return(num)
}

# Convert letter grades to numbers
cols_spr16 = c("paper", "pres", "project",
      "milestone1", "milestone2", "milestone3", "milestone4", "personal",
      "prop", "assign1", "assign2", "assign3", "assign4")
```

```
for(i in seq(1, length(cols_spr16))){
    name = cols_spr16[i]
    grades_spr16[, name] = sapply(grades_spr16[, name], grade_covert)
}

cols_fall16 = c("paper", "pres", "project", "milestone1", "milestone2",
      "personal", "prop", "assign1", "assign2", "assign3", "assign4")

for(i in seq(1, length(cols_fall16))){
    name = cols_fall16[i]
    grades_fall16[, name] = sapply(grades_fall16[, name], grade_covert)
}

# Create grade summary variables
grades_spr16$assign_ave = 100*(grades_spr16$assign1 + grades_spr16$assign2 +
                               grades_spr16$assign3 + grades_spr16$assign4)/400

grades_spr16$milestone_ave = 100*(grades_spr16$milestone1 +
                     grades_spr16$milestone2 + grades_spr16$milestone3 +
                     grades_spr16$milestone4)/400

grades_spr16$project_ave = 100*(grades_spr16$paper + grades_spr16$pres +
                               grades_spr16$project)/300

grades_spr16$total = (grades_spr16$assign_ave*0.15 + grades_spr16$personal*0.05 +
                grades_spr16$prop*0.1 + 100*(grades_spr16$status/5)*0.05 +
                grades_spr16$milestone_ave*0.2 + grades_spr16$project_ave*0.3 +
                100*(grades_spr16$part_raw/75)*0.15)

grades_fall16$assign_ave = 100*(grades_fall16$assign1 + grades_fall16$assign2 +
                               grades_fall16$assign3 + grades_fall16$assign4)/400

grades_fall16$milestone_ave = 100*(grades_fall16$milestone1 +
                     grades_fall16$milestone2)/200

grades_fall16$project_ave = 100*(grades_fall16$paper + grades_fall16$pres +
                               grades_fall16$project)/300

grades_fall16$total = (grades_fall16$assign_ave*0.15 +
          grades_fall16$personal*0.1 + grades_fall16$prop*0.1 +
          100*(grades_fall16$status/5)*0.05 + grades_fall16$milestone_ave*0.15 +
          grades_fall16$project_ave*0.3 + 100*(grades_fall16$part_raw/75)*0.15)


# Drop unnecessary fields from grades dataframes
grades_spr16 = grades_spr16[,c("student", "assign_ave", "milestone_ave",
                               "project_ave", "total")]

grades_fall16 = grades_fall16[,c("student", "assign_ave", "milestone_ave",
                               "project_ave", "total")]
# Merge datasets
edutech_data_fall15 = merge(x = survey_fall15_soc, y = survey_fall15_qc,
                            by = "student", all.x = TRUE)
```

```
edutech_data_fall15 = merge(x = edutech_data_fall15, y = survey_fall15_mc,
                            by = "student", all.x = TRUE)
edutech_data_fall15 = merge(x = edutech_data_fall15, y = survey_fall15_eoc,
                            by = "student", all.x = TRUE)

edutech_data_spr16 = merge(x = survey_spr16_soc, y = survey_spr16_qc,
                           by = "student", all.x = TRUE)
edutech_data_spr16 = merge(x = edutech_data_spr16, y = survey_spr16_mc,
                           by = "student", all.x = TRUE)
edutech_data_spr16 = merge(x = edutech_data_spr16, y = survey_spr16_eoc,
                           by = "student", all.x = TRUE)
edutech_data_spr16 = merge(x = edutech_data_spr16, y = grades_spr16,
                           by = "student", all.x = TRUE)

edutech_data_fall16 = merge(x = survey_fall16_soc, y = survey_fall16_qc,
                            by = "student", all.x = TRUE)
edutech_data_fall16 = merge(x = edutech_data_fall16, y = survey_fall16_mc,
                            by = "student", all.x = TRUE)
edutech_data_fall16 = merge(x = edutech_data_fall16, y = survey_fall16_eoc,
                            by = "student", all.x = TRUE)
edutech_data_fall16 = merge(x = edutech_data_fall16, y = grades_fall16,
                            by = "student", all.x = TRUE)

# Add dummy columns to Fall 15 data
edutech_data_fall15$assign_ave = NA
edutech_data_fall15$milestone_ave = NA
edutech_data_fall15$project_ave = NA
edutech_data_fall15$total = NA

edutech_data_fall15$semester = "Fall 2015"
edutech_data_spr16$semester = "Spring 2016"
edutech_data_fall16$semester = "Fall 2016"

edutech = rbind(edutech_data_fall15, edutech_data_spr16, edutech_data_fall16)

# Drop unneeded datasets
rm(survey_fall15_soc, survey_fall15_qc, survey_fall15_mc, survey_fall15_eoc,
   survey_spr16_soc, survey_spr16_qc, survey_spr16_mc, survey_spr16_eoc,
   survey_fall16_soc, survey_fall16_qc, survey_fall16_mc, survey_fall16_eoc,
   edutech_data_fall15, edutech_data_spr16, edutech_data_fall16, grades_fall16,
   grades_spr16)

# Drop blank rows
edutech = subset(edutech, student != "")

# Replace blanks with NA
is.na(edutech) = (edutech=="")

# Convert factors into character strings
edutech$student = as.character(edutech$student)
edutech$birth = as.character(edutech$birth)
edutech$residence = as.character(edutech$residence)
edutech$language = as.character(edutech$language)
```

```r
# Drop blank factor levels
edutech$age = factor(edutech$age)
edutech$gender = factor(edutech$gender)
edutech$english = factor(edutech$english)
edutech$education = factor(edutech$education)
edutech$conf_p1_post = factor(edutech$conf_p1_post)
edutech$conf_p2_pre = factor(edutech$conf_p2_pre)
edutech$conf_p2_post = factor(edutech$conf_p2_post)
edutech$conf_p3_pre = factor(edutech$conf_p3_pre)
edutech$conf_p3_post = factor(edutech$conf_p3_post)
edutech$hours = factor(edutech$hours)

# Simplify level names
edutech$age = revalue(edutech$age, c("No Answer" = NA))
edutech$gender = revalue(edutech$gender, c("No Answer" = NA))
edutech$english = revalue(edutech$english, c("Native speaker"="Native",
                          "Fully fluent (non-native speaker)"="Fluent",
                          "Partially fluent" = "Partial", "No Answer" = NA))

edutech$education = revalue(edutech$education, c("Bachelors Degree"="Bachelors",
                        "Doctoral Degree"="Doctorate",
                        "High School (or international equivalent)"="High School",
                        "Masters Degree" = "Masters", "No Answer" = NA))

edutech$conf_p1_post = revalue(edutech$conf_p1_post, c("Very confident" = 5,
                       "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                       "Somewhat unconfident" = 2, "Very unconfident" = 1))

edutech$conf_p2_pre = revalue(edutech$conf_p2_pre, c("Very confident" = 5,
                       "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                       "Somewhat unconfident" = 2, "Very unconfident" = 1))

edutech$conf_p2_post = revalue(edutech$conf_p2_post, c("Very confident" = 5,
                       "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                       "Somewhat unconfident" = 2, "Very unconfident" = 1, "No Answer" = NA))

edutech$conf_p3_pre = revalue(edutech$conf_p3_pre, c("Very confident" = 5,
                       "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                       "Somewhat unconfident" = 2, "Very unconfident" = 1))

edutech$conf_p3_post = revalue(edutech$conf_p3_post, c("Very confident" = 5,
                       "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                       "Somewhat unconfident" = 2, "Very unconfident" = 1, "No Answer" = NA))

edutech$hours = revalue(edutech$hours, c("No Answer" = NA))

edutech$hours = factor(edutech$hours, levels = c("0-3", "3-6", "6-9", "9-12", "12-15",
                          "15-18", "18-21", "21+"))

edutech$age = factor(edutech$age, levels = c("Under 18", "18 to 24", "25 to 34",
                                             "35 to 44", "45 to 54", "55 to 64"))

# Create function for removing "1:" from text fields and convert to title case
```

```r
text_split = function(x){
  x = unlist(strsplit(x, ": "))[2]
  return(toTitleCase(x))
}

# Remove "1:" from text fields
edutech$birth = sapply(edutech$birth, text_split)
edutech$residence = sapply(edutech$residence, text_split)
edutech$language = sapply(edutech$language, text_split)

# Get lists of unique values
#unique(edutech$birth)
#unique(edutech$residence)
#unique(edutech$language)

# Clean birth country names
edutech$birth = ifelse(edutech$birth %in% c("United States", "USA", "U.S.A.", "US", "Usa",
                    "Us", "The United States of America", "uSA", "United States of America",
                    "U.S.", "U.S", "Denver City, Tx", "Ethiopia - US Army Base"), "USA",
                    edutech$birth)

edutech$birth = ifelse(edutech$birth %in% c("India", "INDIA"), "India", edutech$birth)
edutech$birth = ifelse(edutech$birth %in% c("China", "People's Republic of China",
                    "P.R.CHINA", "Hong Kong, SAR", "Hong Kong", "CHINA", "China P.R."),
                    "China", edutech$birth)
edutech$birth = ifelse(edutech$birth %in% c("South Korea", "Korea"), "Korea", edutech$birth)
edutech$birth = ifelse(edutech$birth %in% c("Addis Ababa", "Ethiopia"), "Ethiopia",
                        edutech$birth)
edutech$birth = ifelse(edutech$birth %in% c("United Kingdom", "England"), "UK",
                        edutech$birth)
edutech$birth = ifelse(edutech$birth == "NA", NA, edutech$birth)


# Clean residence country names
edutech$residence = ifelse(edutech$residence %in% c("United States", "USA", "U.S.A.",
                    "US", "Usa",
                    "The United States of America", "uSA", "United States of America",
                    "United State", "USa", "Los Angeles", "Houston", "U.S", "U.S.", "YSA",
                    "Us", "United STates", "America"), "USA", edutech$residence)

edutech$residence = ifelse(edutech$residence == "NA", NA, edutech$residence)
edutech$residence = ifelse(edutech$residence == "Myanmar, Hong Kong", "Myanmar",
                           edutech$residence)
edutech$residence = ifelse(edutech$residence %in% c("China", "Hong Kong"), "China",
                           edutech$residence)

# Clean language
edutech$language = ifelse(edutech$language %in% c("English", "American English", "ENGLISH",
                  "American", "English (US)", "English Language", "Englist",
                  "C++, but you Probably Mean \"English\"", "ENGLISH", "En", "JavaScript",
                  "Elijah", "Dallas"), "English", edutech$language)
edutech$language = ifelse(edutech$language %in% c("Chinese", "Mandarin", "China",
                  "Mandarin Chinese", "Cantonese", "Chiinese", "CHINESE", "Manderin",
```

```
                  "Java", "Python"), "Chinese", edutech$language)
edutech$language = ifelse(edutech$language %in% c("Marathi", "Telugu", "Bengali", "Gujarati",
                  "Kannada", "Hindi", "Tamil", "Odiya", "TAMIL", "Punjabi", "Hindo"),
                  "Indian", edutech$language)
edutech$language = ifelse(edutech$language == "Farsi/English", "Farsi", edutech$language)
edutech$language = ifelse(edutech$language == "Spanish/English", "Spanish", edutech$language)
edutech$language = ifelse(edutech$language == "NA", NA, edutech$language)

# Create factors
edutech$birth = factor(edutech$birth)
edutech$residence = factor(edutech$residence)
edutech$language = factor(edutech$language)
edutech$semester = factor(edutech$semester)

# Convert confidence scores to numeric
edutech$conf_p1_post = as.numeric(as.character(edutech$conf_p1_post))
edutech$conf_p2_pre = as.numeric(as.character(edutech$conf_p2_pre))
edutech$conf_p2_post = as.numeric(as.character(edutech$conf_p2_post))
edutech$conf_p3_pre = as.numeric(as.character(edutech$conf_p3_pre))
edutech$conf_p3_post = as.numeric(as.character(edutech$conf_p3_post))

# Calculate average confidence scores
edutech$conf_ave = (edutech$conf_p1_post + edutech$conf_p2_pre + edutech$conf_p2_post +
                   edutech$conf_p3_pre + edutech$conf_p3_post)/5

edutech$conf_pre_ave = (edutech$conf_p2_pre + edutech$conf_p3_pre)/2

edutech$conf_post_ave = (edutech$conf_p1_post + edutech$conf_p2_post +
                           edutech$conf_p3_post)/3

# Convert ranges to numeric values
edutech$age_num = revalue(edutech$age, c("18 to 24"=21, "25 to 34"=29.5, "35 to 44"=39.5,
                                         "45 to 54"=49.5, "55 to 64"=59.5, "Under 18" = 18))
edutech$age_num = as.numeric(as.character(edutech$age_num))

edutech$hours_num = revalue(edutech$hours, c("0-3"=1.5, "3-6"=4.5, "6-9"=7.5, "9-12"=10.5,
                  "12-15"=13.5, "15-18"=16.5, "18-21"=19.5, "21+"=21))
edutech$hours_num = as.numeric(as.character(edutech$hours_num))

# Create indicator variables
edutech$native_ind = ifelse(edutech$english == "Native", 1, 0)
edutech$higher_ind = ifelse(edutech$education %in% c("Masters", "Doctorate"), 1, 0)
edutech$gender_ind = ifelse(edutech$gender == "Male", 1, 0)
```

Explore Data

```
# Calculate summary statistics
summary(edutech)
```

```
##    student                age          gender        birth
##  Length:335         Under 18:  1   Female: 54   USA    :186
##  Class :character   18 to 24: 20   Male  :272   India  : 36
```

```
##  Mode  :character   25 to 34:173   NA's  :  9   China  : 34
##                     35 to 44: 91                Canada :  8
##                     45 to 54: 37                Vietnam:  5
##                     55 to 64:  8                (Other): 57
##                     NA's    :  5                NA's   :  9
##    residence          language      english          education
##  USA    :294   English   :242   Fluent :102   Bachelors  :236
##  Canada : 10   Chinese   : 31   Native :216   Doctorate  : 17
##  India  :  6   Indian    : 20   Partial:  8   High School:  1
##  China  :  4   Spanish   : 11   NA's   :  9   Masters     : 70
##  Kenya  :  3   Indonesian:  2                 NA's       : 11
##  (Other): 10   (Other)   : 20
##  NA's   :  8   NA's      :  9
##   conf_p1_post    conf_p2_pre     conf_p2_post    conf_p3_pre
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:4.000   1st Qu.:4.000   1st Qu.:4.000   1st Qu.:4.000
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000
##  Mean   :4.066   Mean   :4.149   Mean   :4.224   Mean   :4.237
##  3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:5.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000
##  NA's   :46      NA's   :46      NA's   :41      NA's   :40
##      hours     conf_p3_post     assign_ave    milestone_ave
##  9-12   :85   Min.   :1.000   Min.   :66.25   Min.   : 0.00
##  12-15  :61   1st Qu.:4.000   1st Qu.:94.50   1st Qu.:95.00
##  6-9    :43   Median :4.000   Median :95.00   Median :95.00
##  15-18  :25   Mean   :3.996   Mean   :93.97   Mean   :93.29
##  3-6    :18   3rd Qu.:5.000   3rd Qu.:95.00   3rd Qu.:95.00
##  (Other):20   Max.   :5.000   Max.   :95.00   Max.   :95.00
##  NA's   :83   NA's   :83      NA's   :94      NA's   :94
##   project_ave        total              semester      conf_ave
##  Min.   : 0.00   Min.   :41.45   Fall 2015  : 94   Min.   :1.800
##  1st Qu.:93.33   1st Qu.:84.62   Fall 2016  :124   1st Qu.:3.800
##  Median :95.00   Median :85.00   Spring 2016:117   Median :4.200
##  Mean   :91.85   Mean   :88.15                     Mean   :4.129
##  3rd Qu.:95.00   3rd Qu.:95.00                     3rd Qu.:4.400
##  Max.   :95.00   Max.   :96.00                     Max.   :5.000
##  NA's   :94      NA's   :94                        NA's   :117
##   conf_pre_ave   conf_post_ave      age_num        hours_num
##  Min.   :2.000   Min.   :1.667   Min.   :18.00   Min.   : 4.50
##  1st Qu.:4.000   1st Qu.:3.667   1st Qu.:29.50   1st Qu.:10.50
##  Median :4.000   Median :4.000   Median :29.50   Median :10.50
##  Mean   :4.201   Mean   :4.096   Mean   :34.68   Mean   :11.64
##  3rd Qu.:4.500   3rd Qu.:4.667   3rd Qu.:39.50   3rd Qu.:13.50
##  Max.   :5.000   Max.   :5.000   Max.   :59.50   Max.   :21.00
##  NA's   :67      NA's   :117     NA's   :5       NA's   :83
##    native_ind       higher_ind       gender_ind
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.0000
##  Median :1.0000   Median :0.0000   Median :1.0000
##  Mean   :0.6626   Mean   :0.2597   Mean   :0.8344
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##  NA's   :9                         NA's   :9
```

```r
# Calculate proportion of class by gender
prop.table(table(edutech$gender))
```

```
##
##    Female      Male
## 0.1656442 0.8343558
```

Analyze Data by Gender

```r
# Calculate age summary statistics
ddply(subset(edutech, !is.na(age_num) & !is.na(gender)), "gender", summarise,
      mean = mean(age_num),
      sd = sd(age_num), median = median(age_num), first_q = quantile(age_num, 0.25),
      third_q = quantile(age_num, 0.75))
```

```
##   gender     mean       sd median first_q third_q
## 1 Female 33.78704 8.102877   29.5    29.5    39.5
## 2   Male 34.78309 8.593571   29.5    29.5    39.5
```

```r
# Calculate study hours summary statistics
ddply(subset(edutech, !is.na(gender)&!is.na(hours_num)), "gender", summarise,
             mean = mean(hours_num), sd = sd(hours_num), median = median(hours_num),
             first_q = quantile(hours_num, 0.25), third_q = quantile(hours_num, 0.75))
```

```
##   gender     mean       sd median first_q third_q
## 1 Female 12.28571 4.335254   10.5     7.5   15.75
## 2   Male 11.51471 3.928835   10.5    10.5   13.50
```

```r
# Calculate confidence summary statistics
ddply(subset(edutech, !is.na(gender)&!is.na(conf_ave)), "gender", summarise,
             mean = mean(conf_ave), sd = sd(conf_ave), median = median(conf_ave),
             first_q = quantile(conf_ave, 0.25), third_q = quantile(conf_ave, 0.75))
```

```
##   gender     mean        sd median first_q third_q
## 1 Female 3.927778 0.6162611    4.0     3.6     4.4
## 2   Male 4.174157 0.5327740    4.2     3.8     4.6
```

```r
# Calculate confidence summary statistics
ddply(subset(edutech, !is.na(gender)&!is.na(conf_pre_ave)), "gender", summarise,
             mean = mean(conf_pre_ave), sd = sd(conf_pre_ave), median = median(conf_pre_ave),
             first_q = quantile(conf_pre_ave, 0.25), third_q = quantile(conf_pre_ave, 0.75))
```

```
##   gender     mean        sd median first_q third_q
## 1 Female 3.975000 0.5655721      4       4     4.0
## 2   Male 4.240991 0.5800491      4       4     4.5
```

```r
ddply(subset(edutech, !is.na(gender)&!is.na(conf_post_ave)), "gender", summarise,
             mean = mean(conf_post_ave), sd = sd(conf_post_ave),
             median = median(conf_post_ave), first_q = quantile(conf_post_ave, 0.25),
             third_q = quantile(conf_post_ave, 0.75))
```

```
##   gender     mean        sd median  first_q  third_q
## 1 Female 3.907407 0.6928712      4 3.583333 4.333333
## 2   Male 4.136704 0.5976843      4 3.666667 4.666667
```

```r
edutech_m = subset(edutech, gender == "Male")
edutech_f = subset(edutech, gender == "Female")

# Compare age
prop.table(table(edutech_m$age))
```

```
##
##    Under 18    18 to 24    25 to 34    35 to 44    45 to 54    55 to 64
## 0.003676471 0.069852941 0.496323529 0.290441176 0.117647059 0.022058824
```

```r
prop.table(table(edutech_f$age))
```

```
##
##   Under 18   18 to 24   25 to 34   35 to 44   45 to 54   55 to 64
## 0.00000000 0.01851852 0.68518519 0.18518519 0.07407407 0.03703704
```

```r
# Compare birth country
prop.table(table(edutech_m$birth))
```

```
##
##        Afghanistan          Argentina          Australia
##        0.003690037        0.011070111        0.003690037
##            Bahamas         Bangladesh             Canada
##        0.003690037        0.003690037        0.029520295
##              China     Czech Republic Dominican Republic
##        0.092250923        0.003690037        0.003690037
##            Ecuador        El Salvador            Finland
##        0.000000000        0.003690037        0.000000000
##              Ghana            Grenada           Honduras
##        0.000000000        0.003690037        0.003690037
##              India          Indonesia               Iran
##        0.110701107        0.007380074        0.003690037
##              Italy              Japan         Kazakhstan
##        0.003690037        0.007380074        0.003690037
##              Kenya              Korea            Lebanon
##        0.000000000        0.003690037        0.003690037
##          Lithuania         Luxembourg             Mexico
##        0.003690037        0.003690037        0.011070111
##            Moldova              Nepal          Nicaragua
##        0.000000000        0.007380074        0.003690037
##            Nigeria           Pakistan               Peru
##        0.003690037        0.011070111        0.007380074
##        Philippines             Poland              Qatar
##        0.007380074        0.003690037        0.000000000
##            Romania             Rwanda          Singapore
##        0.007380074        0.003690037        0.000000000
##             Taiwan        The Bahamas                 UK
##        0.000000000        0.003690037        0.011070111
##                USA            Vietnam
##        0.583025830        0.018450185
```

```r
prop.table(table(edutech_f$birth))
```

```
##
##        Afghanistan          Argentina          Australia
##         0.00000000         0.00000000         0.00000000
```

```
##            Bahamas         Bangladesh             Canada
##         0.00000000         0.00000000         0.00000000
##              China     Czech Republic Dominican Republic
##         0.16666667         0.00000000         0.00000000
##            Ecuador        El Salvador            Finland
##         0.01851852         0.00000000         0.01851852
##              Ghana            Grenada           Honduras
##         0.01851852         0.00000000         0.00000000
##              India          Indonesia               Iran
##         0.11111111         0.00000000         0.00000000
##              Italy              Japan         Kazakhstan
##         0.00000000         0.00000000         0.00000000
##              Kenya              Korea            Lebanon
##         0.03703704         0.01851852         0.00000000
##           Lithuania         Luxembourg             Mexico
##         0.00000000         0.00000000         0.00000000
##            Moldova              Nepal          Nicaragua
##         0.01851852         0.00000000         0.00000000
##            Nigeria           Pakistan               Peru
##         0.00000000         0.00000000         0.00000000
##        Philippines             Poland              Qatar
##         0.00000000         0.00000000         0.03703704
##            Romania             Rwanda          Singapore
##         0.01851852         0.00000000         0.01851852
##             Taiwan        The Bahamas                 UK
##         0.01851852         0.00000000         0.00000000
##                USA            Vietnam
##         0.50000000         0.00000000
```

```
# Compare country of residence
prop.table(table(edutech_m$residence))
```

```
##
##      Australia         Canada          China Czech Republic        Grenada
##    0.003676471    0.033088235    0.014705882    0.003676471    0.003676471
##      Guatemala          India          Japan          Kenya       Pakistan
##    0.003676471    0.014705882    0.003676471    0.003676471    0.003676471
##      Singapore    Switzerland    The Bahamas            USA
##    0.000000000    0.000000000    0.003676471    0.908088235
```

```
prop.table(table(edutech_f$residence))
```

```
##
##      Australia         Canada          China Czech Republic        Grenada
##     0.00000000     0.01851852     0.00000000     0.00000000     0.00000000
##      Guatemala          India          Japan          Kenya       Pakistan
##     0.00000000     0.03703704     0.01851852     0.03703704     0.00000000
##      Singapore    Switzerland    The Bahamas            USA
##     0.01851852     0.01851852     0.00000000     0.85185185
```

```
# Compare language background
prop.table(table(edutech_m$language))
```

```
##
##   Cambodian     Chinese       Czech        Dari     English       Farsi
## 0.003690037 0.084870849 0.003690037 0.003690037 0.752767528 0.003690037
```

```
##      Indian  Indonesian      Korean  Lithuanian   Malayalam      Nepali
## 0.059040590 0.007380074 0.003690037 0.003690037 0.000000000 0.007380074
##    Romanian     Russian     Spanish     Tagalog        Urdu  Vietnamese
## 0.003690037 0.003690037 0.036900369 0.007380074 0.007380074 0.007380074
```

```
prop.table(table(edutech_f$language))
```

```
##
##  Cambodian    Chinese      Czech       Dari    English      Farsi
## 0.00000000 0.14814815 0.00000000 0.00000000 0.68518519 0.00000000
##     Indian Indonesian     Korean Lithuanian  Malayalam     Nepali
## 0.07407407 0.00000000 0.01851852 0.00000000 0.03703704 0.00000000
##   Romanian    Russian    Spanish    Tagalog       Urdu Vietnamese
## 0.01851852 0.00000000 0.01851852 0.00000000 0.00000000 0.00000000
```

```
# Compare English skills
prop.table(table(edutech_m$english))
```

```
##
##     Fluent     Native    Partial
## 0.29151292 0.68265683 0.02583026
```

```
prop.table(table(edutech_f$english))
```

```
##
##     Fluent     Native    Partial
## 0.42592593 0.55555556 0.01851852
```

```
# Compare education
prop.table(table(edutech_m$education))
```

```
##
##   Bachelors   Doctorate High School     Masters
## 0.740740741 0.040740741 0.003703704 0.214814815
```

```
prop.table(table(edutech_f$education))
```

```
##
##   Bachelors   Doctorate High School     Masters
##   0.6603774   0.1132075   0.0000000   0.2264151
```

```
# Compare hours
prop.table(table(edutech_m$hours))
```

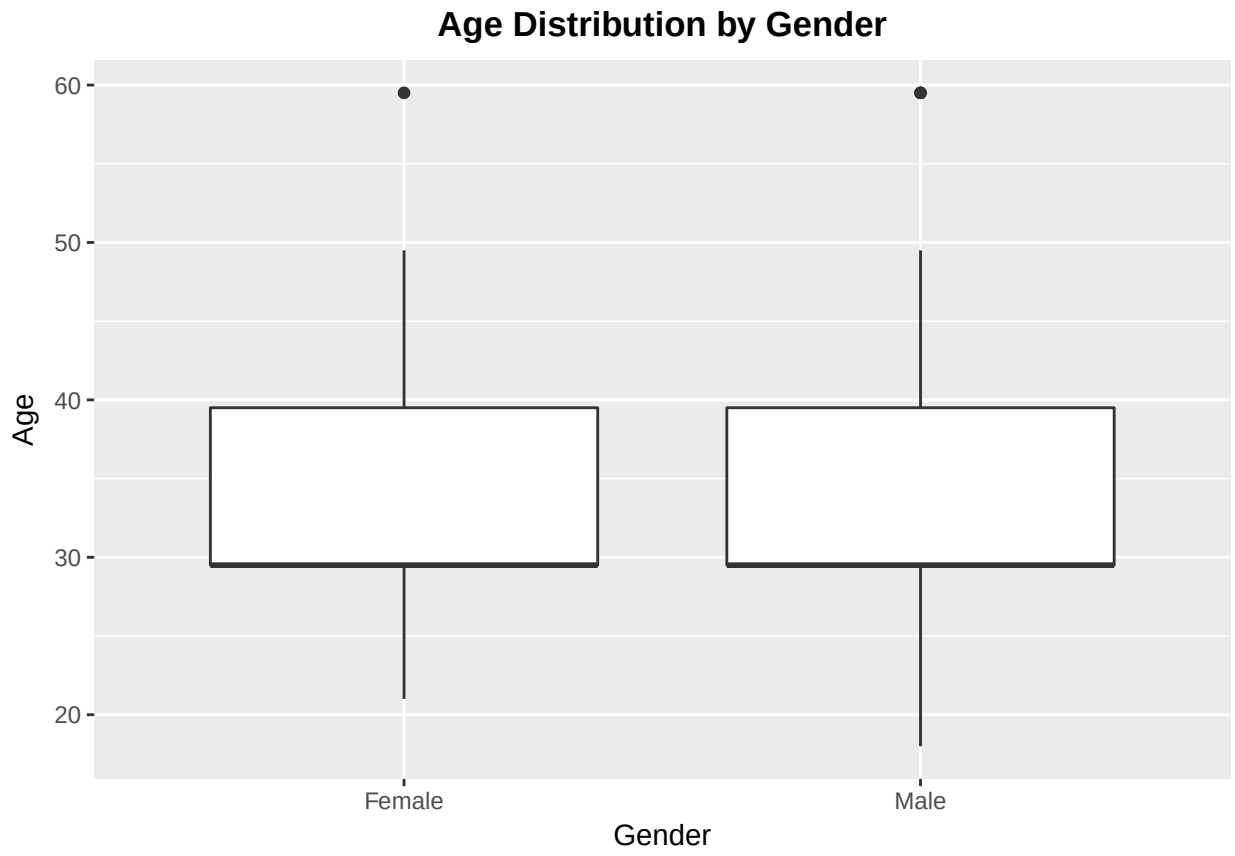
```
##
##        0-3        3-6        6-9       9-12      12-15      15-18
## 0.00000000 0.08823529 0.14215686 0.35294118 0.25980392 0.08823529
##      18-21        21+
## 0.03921569 0.02941176
```

```
prop.table(table(edutech_f$hours))
```

```
##
##        0-3        3-6        6-9       9-12      12-15      15-18
## 0.00000000 0.00000000 0.28571429 0.26190476 0.19047619 0.11904762
##      18-21        21+
## 0.09523810 0.04761905
```

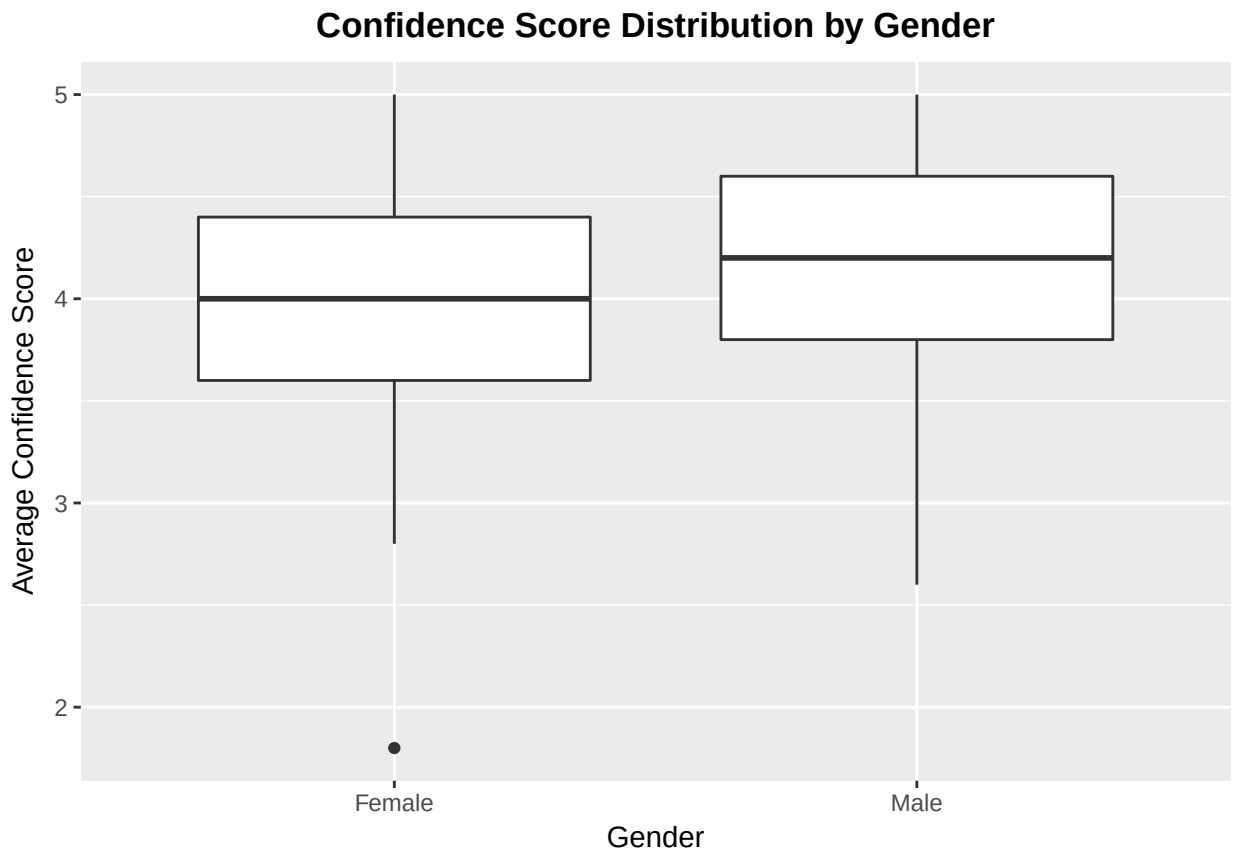
```
#Boxplot of age distribution by gender
ggplot(subset(edutech, !is.na(gender)), aes(gender, age_num)) +
```

```
geom_boxplot() +
labs(title = "Age Distribution by Gender",
     x = "Gender", y = "Age") +
theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

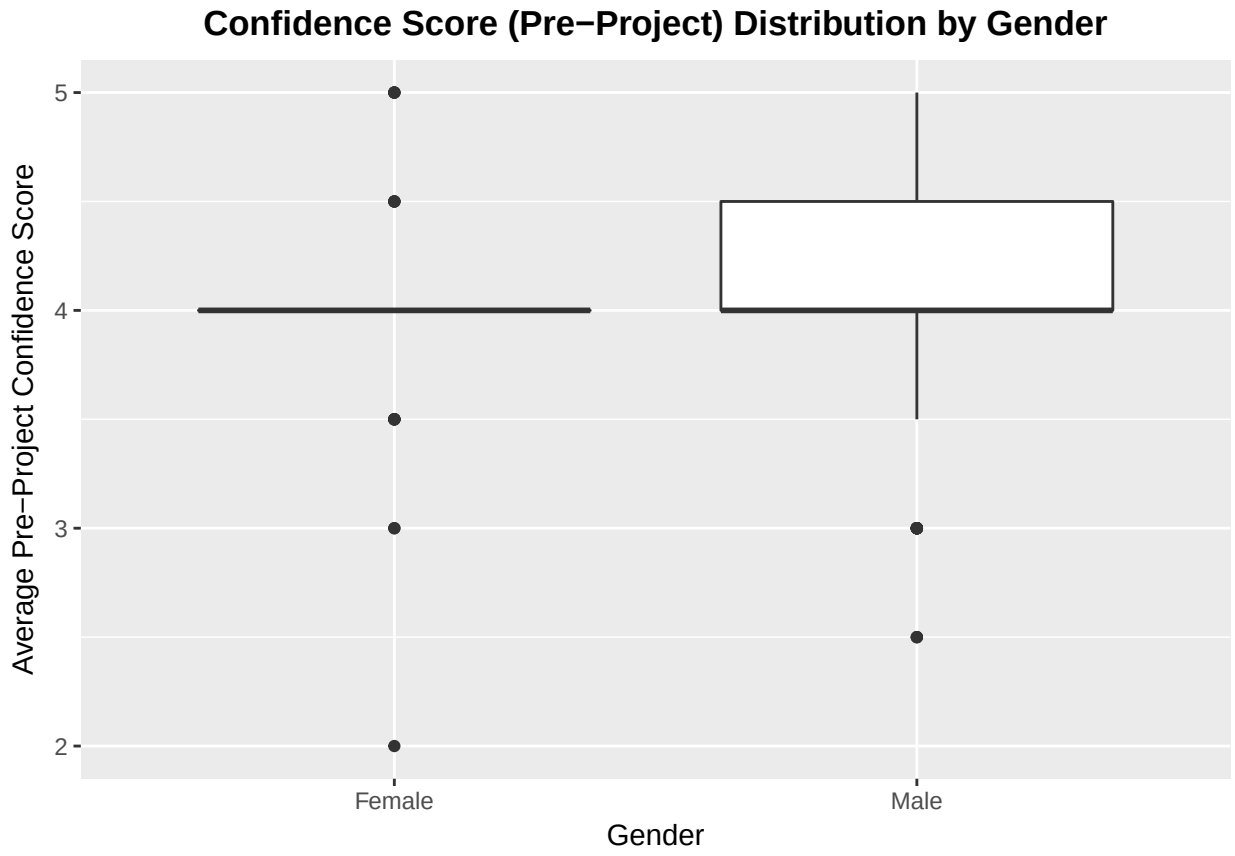



```
# Boxplot of hours spent studying by gender
ggplot(subset(edutech, !is.na(hours_num) & !is.na(gender)), aes(gender, hours_num)) +
  geom_boxplot() +
 labs(title = "Study Hours Distribution by Gender",
      x = "Gender", y = "Study Hours") +
 theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

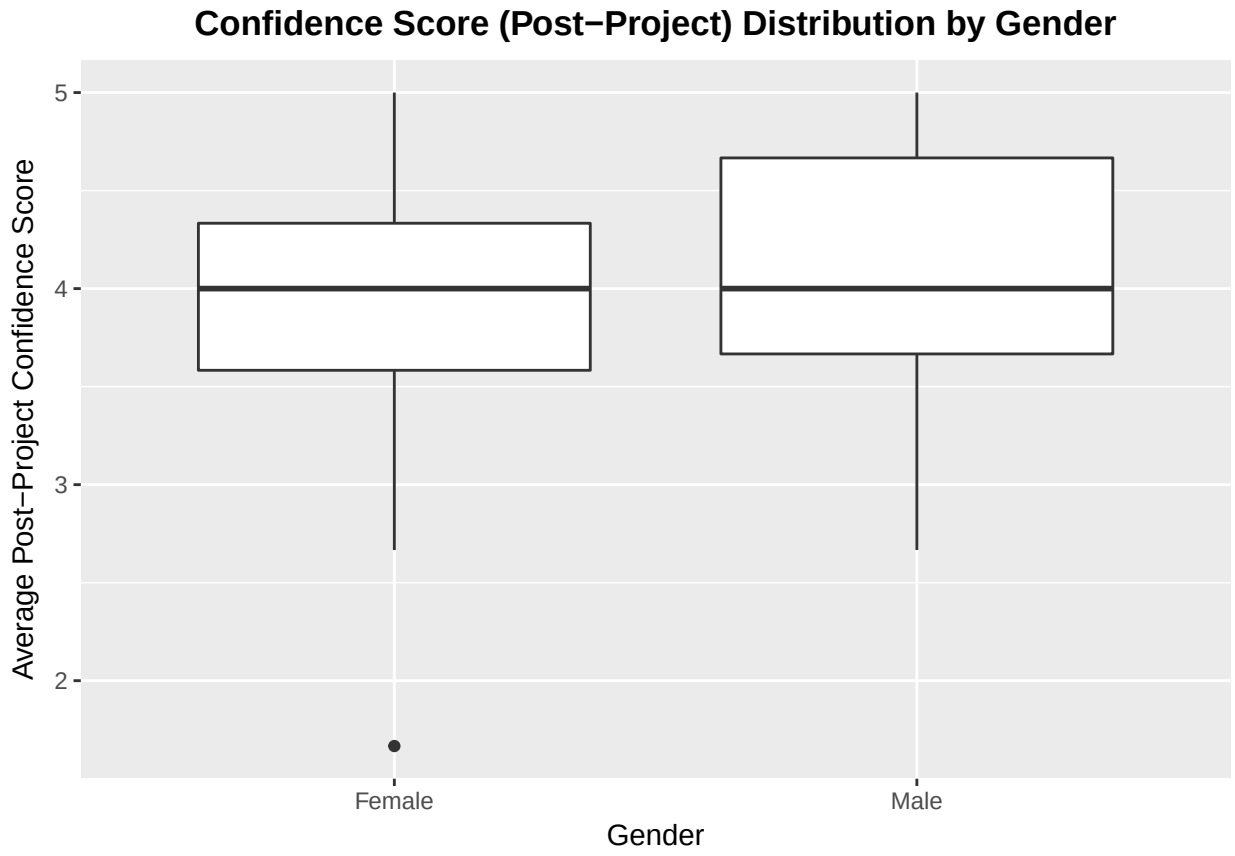

```
# Boxplot of confidence score by gender
ggplot(subset(edutech, !is.na(conf_ave) & !is.na(gender)), aes(gender, conf_ave)) +
  geom_boxplot() +
 labs(title = "Confidence Score Distribution by Gender",
      x = "Gender", y = "Average Confidence Score") +
 theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```



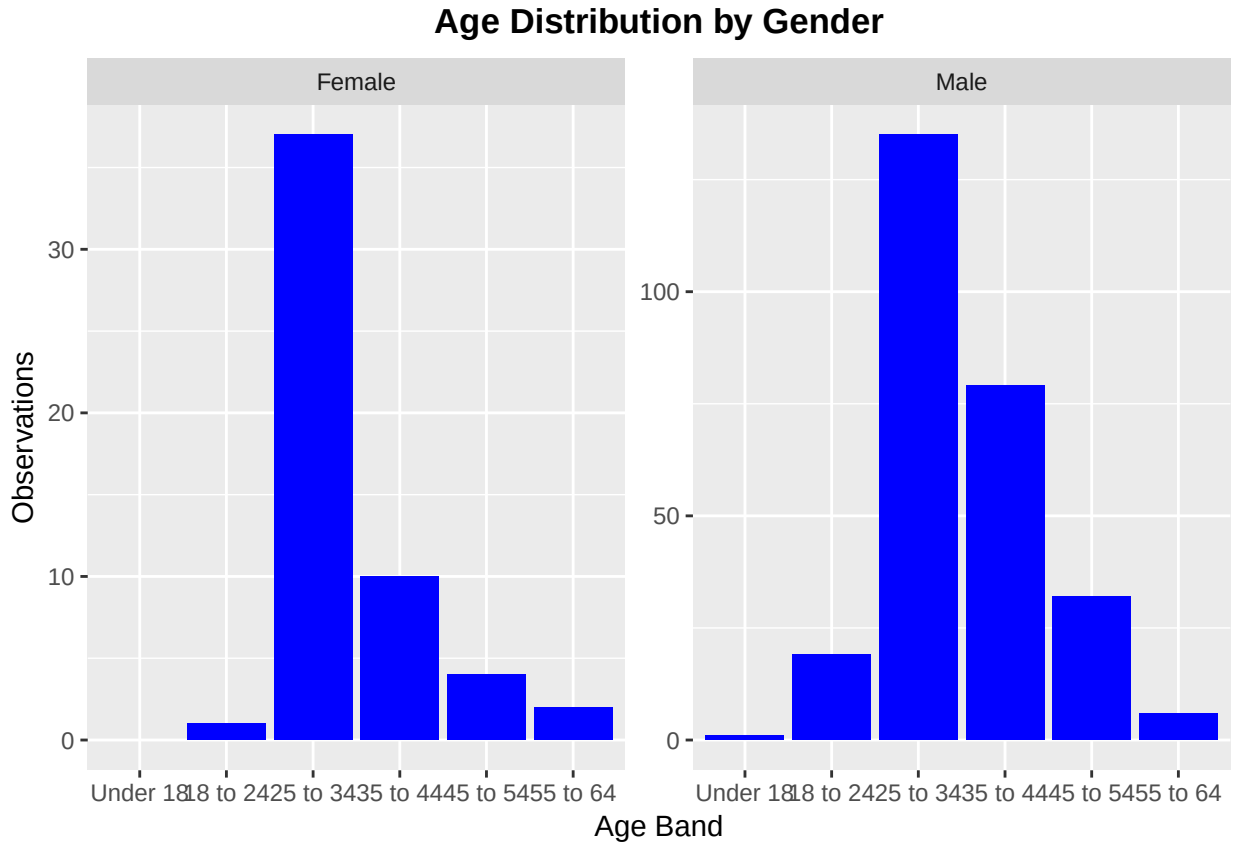

```
# Boxplot of confidence score (pre-project) by gender
ggplot(subset(edutech, !is.na(conf_pre_ave) & !is.na(gender)), aes(gender,
    conf_pre_ave)) +  geom_boxplot() +
 labs(title = "Confidence Score (Pre-Project) Distribution by Gender",
      x = "Gender", y = "Average Pre-Project Confidence Score") +
 theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```



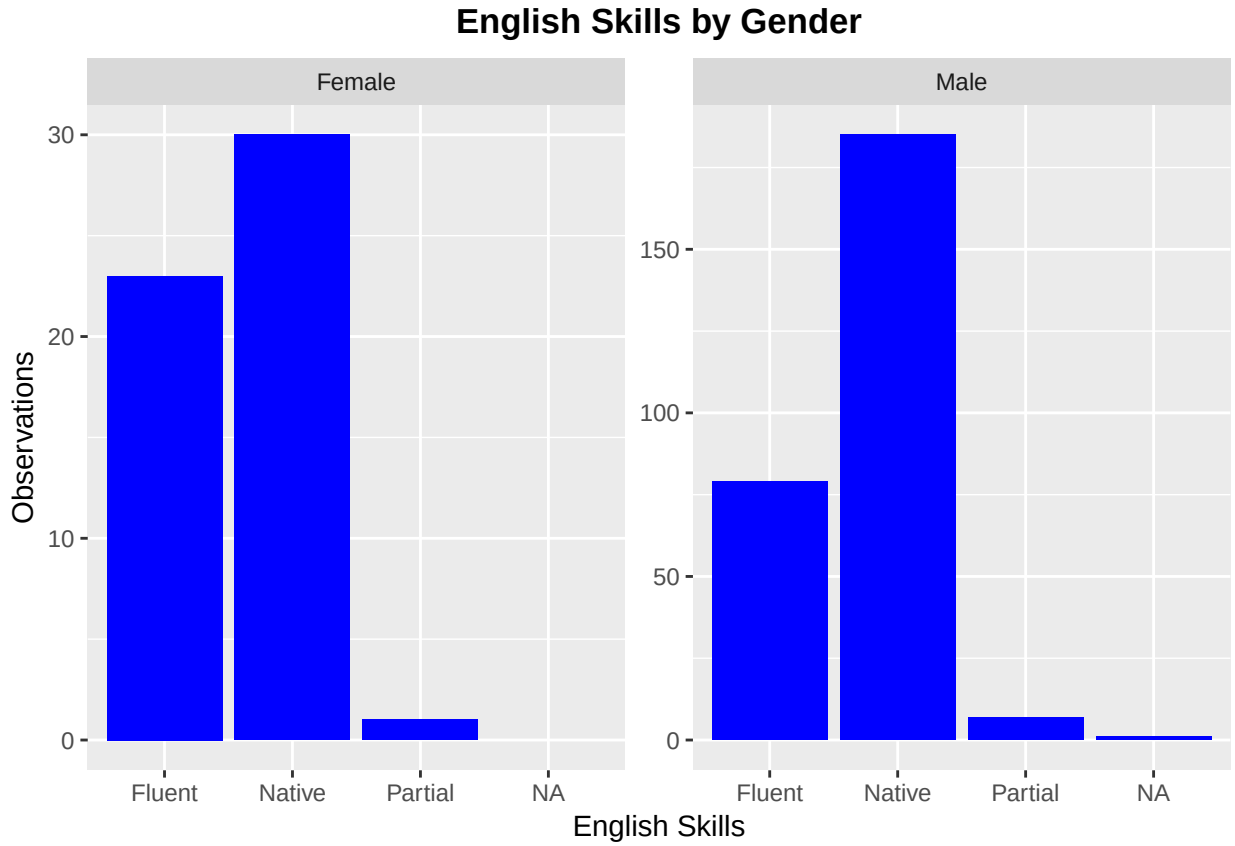

```
# Boxplot of confidence score (post-project) by gender
ggplot(subset(edutech, !is.na(conf_post_ave) & !is.na(gender)), aes(gender,
    conf_post_ave)) +  geom_boxplot() +
 labs(title = "Confidence Score (Post-Project) Distribution by Gender",
      x = "Gender", y = "Average Post-Project Confidence Score") +
 theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```



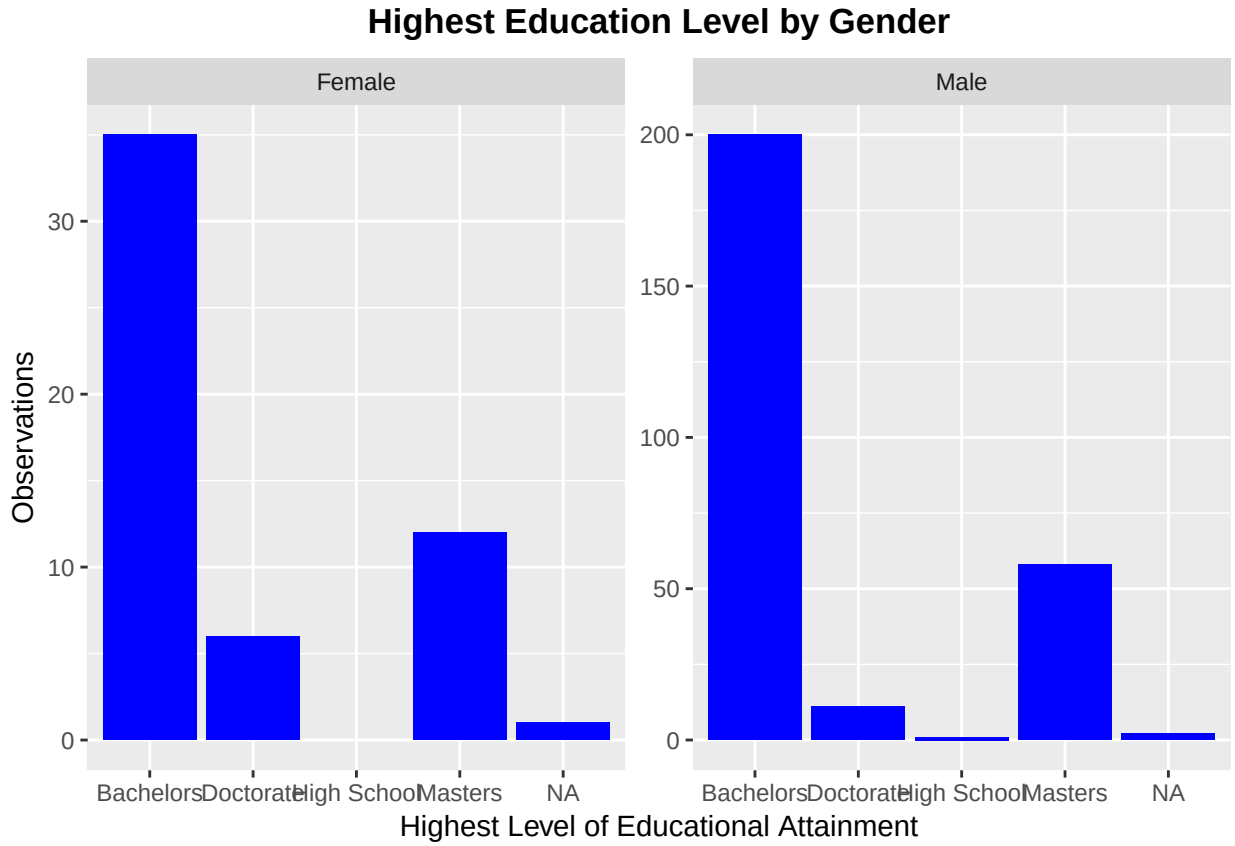

```
# Bar chart comparing age by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = age)) +
    geom_bar(fill = "blue") +
    facet_wrap(~gender, scales = "free_y") +
    labs(title = "Age Distribution by Gender",
      x = "Age Band",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```




```r
# Bar chart comparing English skills by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = english)) +
    geom_bar(fill = "blue") +
    facet_wrap(~gender, scales = "free_y") +
    labs(title = "English Skills by Gender",
      x = "English Skills",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

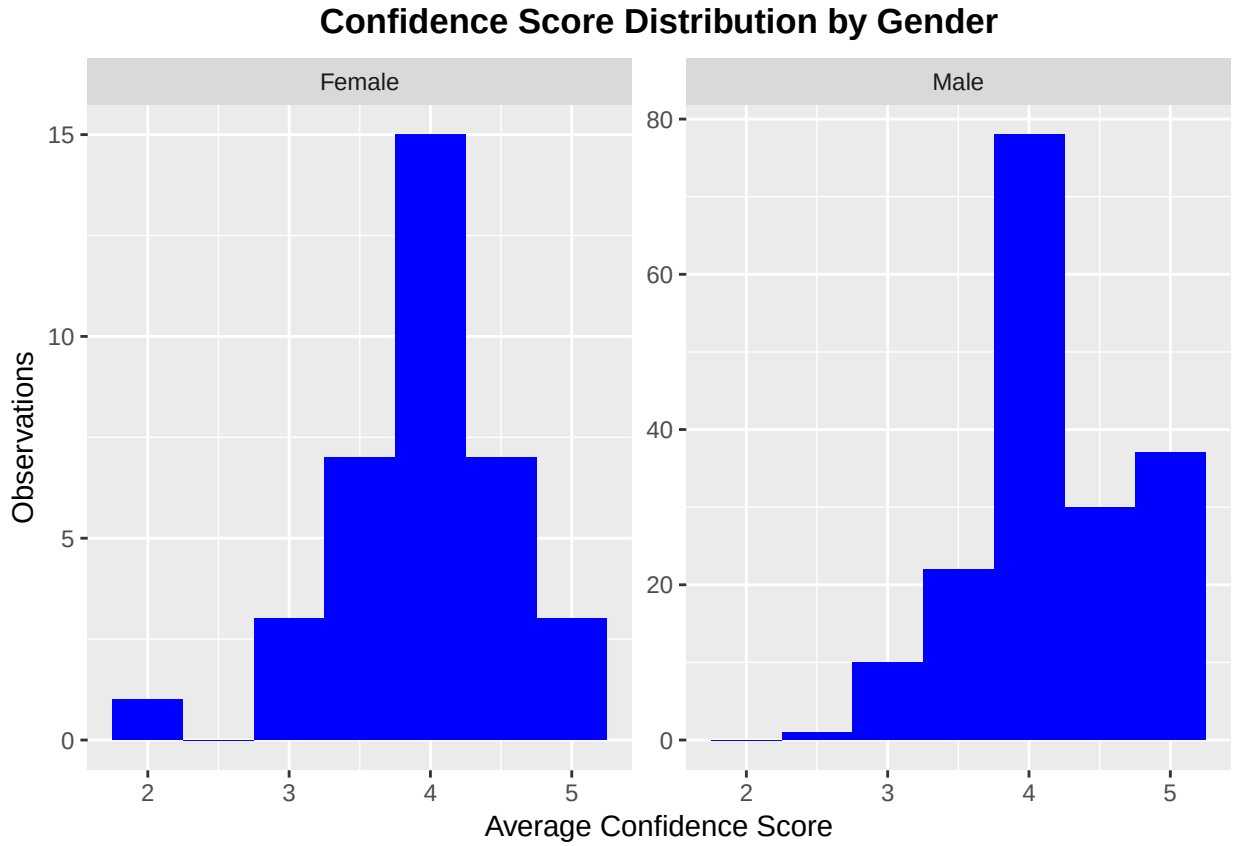



```
# Bar chart comparing education by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = education)) +
    geom_bar(fill = "blue") +
    facet_wrap(~gender, scales = "free_y") +
    labs(title = "Highest Education Level by Gender",
      x = "Highest Level of Educational Attainment",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```



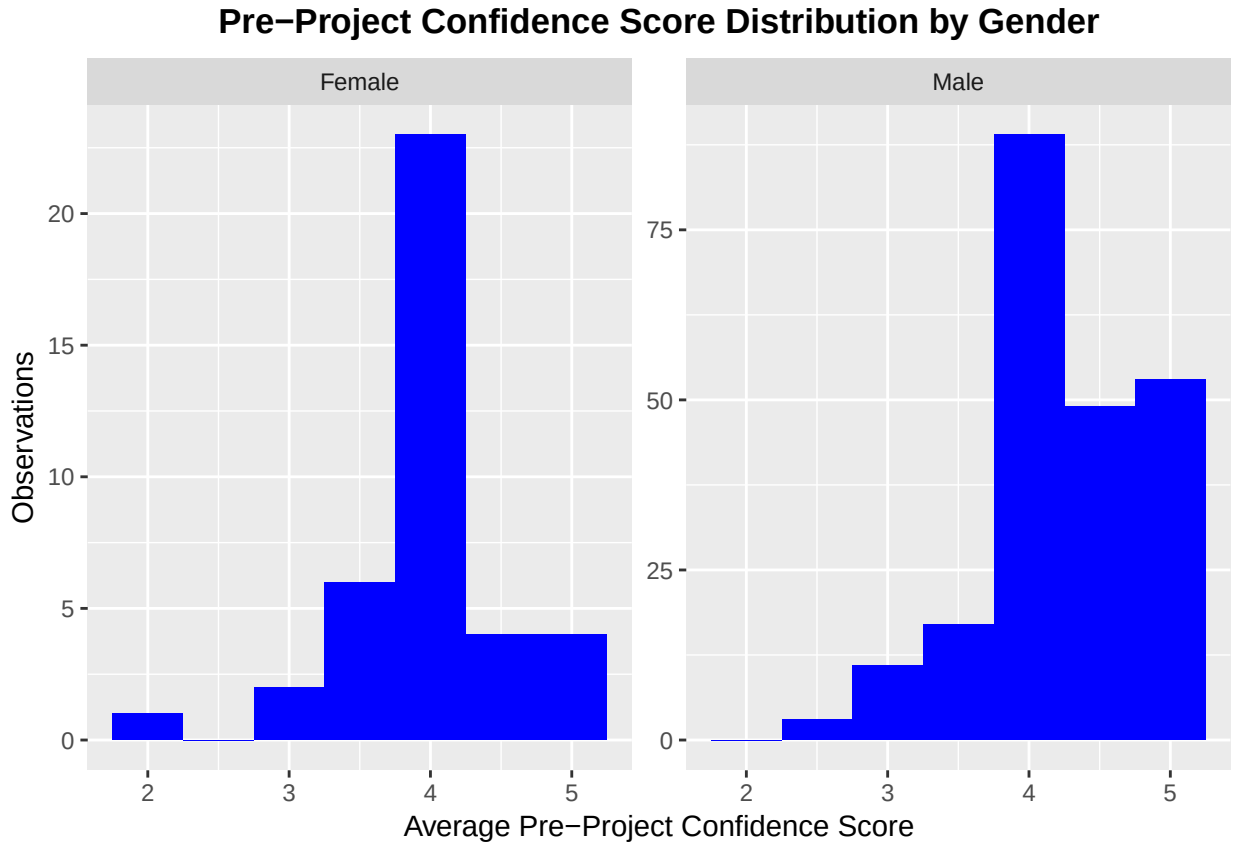

```
# Histogram of conf_ave by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = conf_ave)) +
    geom_histogram(fill = "blue", binwidth = 0.5) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Confidence Score Distribution by Gender",
      x = "Average Confidence Score",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

```
## Warning: Removed 112 rows containing non-finite values (stat_bin).
```



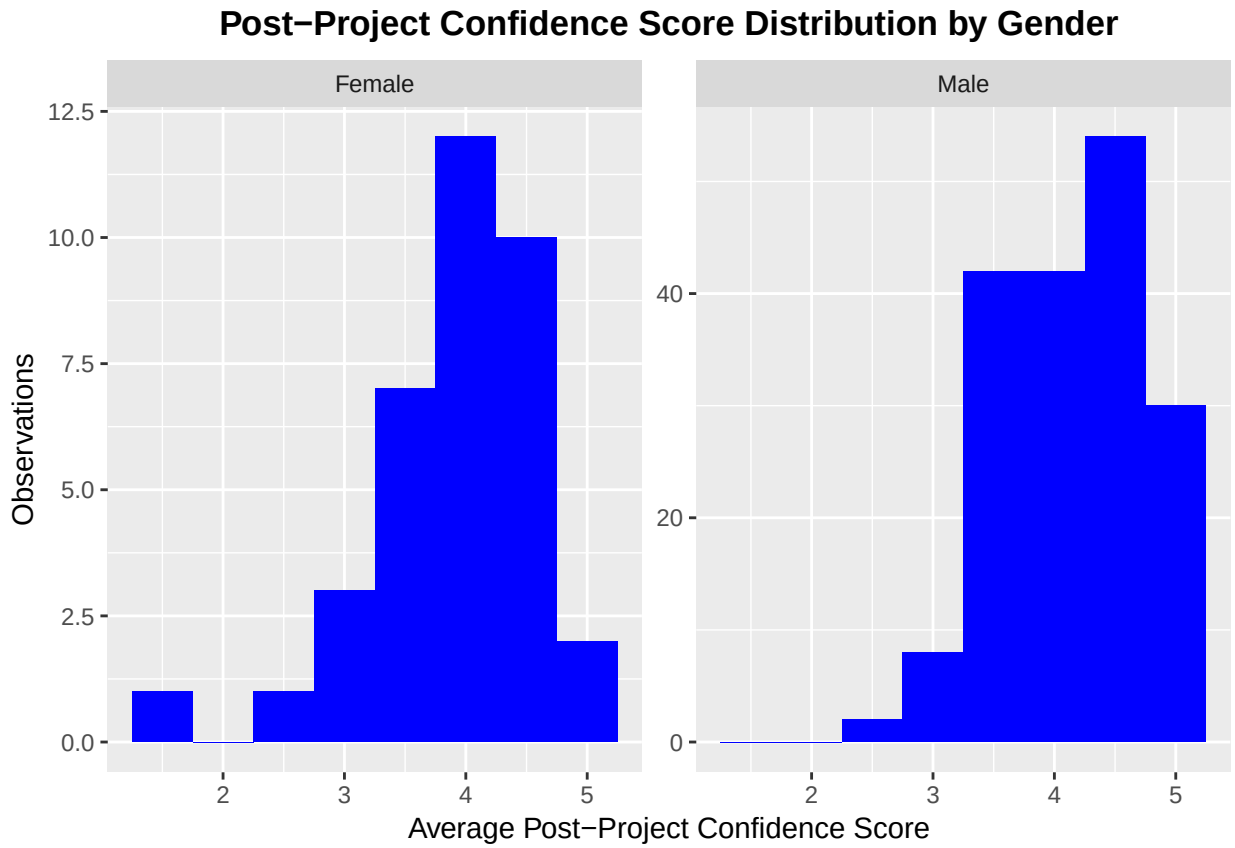

```
# Histogram of conf_pre_ave by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = conf_pre_ave)) +
    geom_histogram(fill = "blue", binwidth = 0.5) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Pre-Project Confidence Score Distribution by Gender",
      x = "Average Pre-Project Confidence Score",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

```
## Warning: Removed 64 rows containing non-finite values (stat_bin).
```



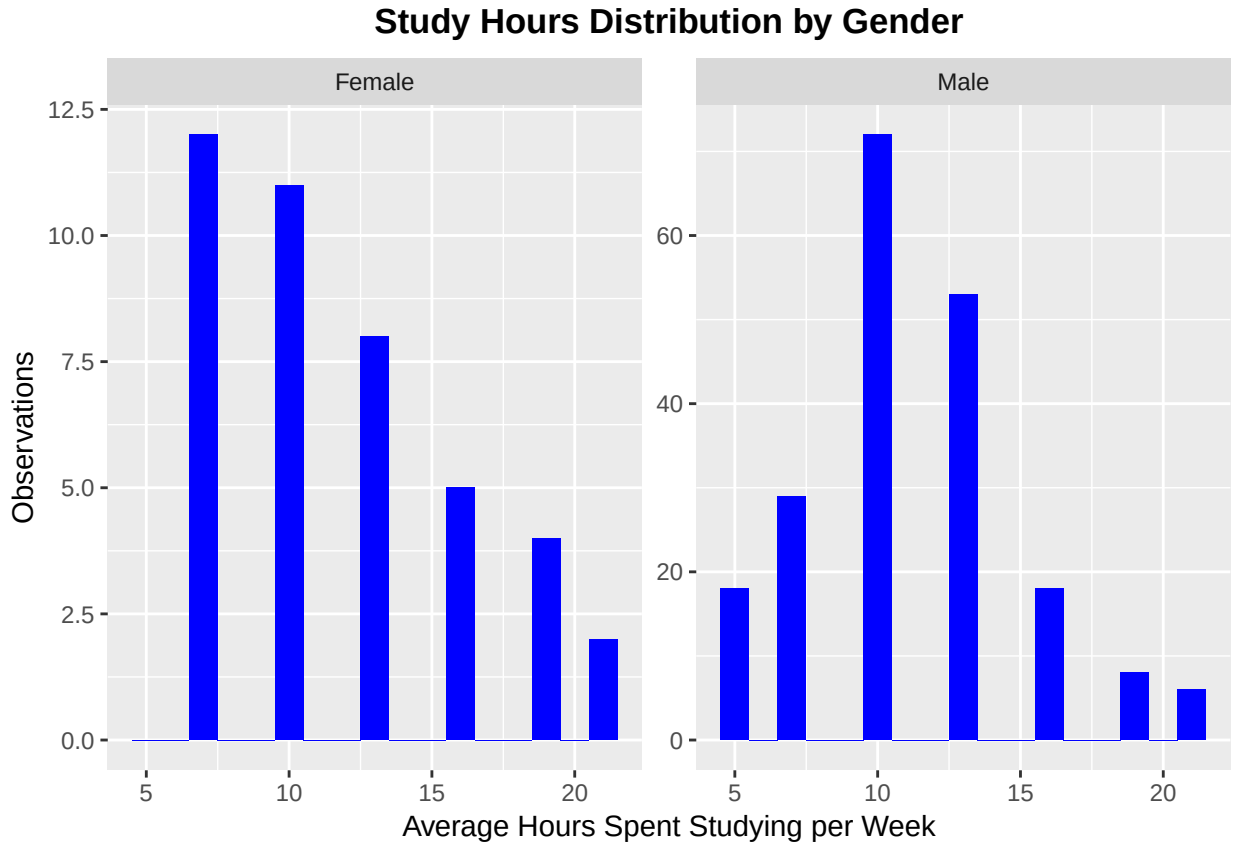

```
# Histogram of conf_post_ave by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = conf_post_ave)) +
    geom_histogram(fill = "blue", binwidth = 0.5) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Post-Project Confidence Score Distribution by Gender",
      x = "Average Post-Project Confidence Score",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

```
## Warning: Removed 112 rows containing non-finite values (stat_bin).
```




```
# Histogram of study hours by gender
ggplot(subset(edutech, !is.na(gender)), aes(x = hours_num)) +
    geom_histogram(fill = "blue", binwidth = 1) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Study Hours Distribution by Gender",
      x = "Average Hours Spent Studying per Week",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```

```
## Warning: Removed 80 rows containing non-finite values (stat_bin).
```




```
# Age tests
t.test(edutech_m$age_num, edutech_f$age_num)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$age_num and edutech_f$age_num
## t = 0.81672, df = 78.547, p-value = 0.4166
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.431674  3.423777
## sample estimates:
## mean of x mean of y
##  34.78309  33.78704
```

```
wilcox.test(age_num ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  age_num by gender
## W = 6731, p-value = 0.2883
## alternative hypothesis: true location shift is not equal to 0
```

```
# Higher ed tests
t.test(edutech_m$higher_ind, edutech_f$higher_ind)
```

```
##
```

```
##  Welch Two Sample t-test
##
## data:  edutech_m$higher_ind and edutech_f$higher_ind
## t = -1.1389, df = 71.744, p-value = 0.2585
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.21908656  0.05977284
## sample estimates:
## mean of x mean of y
## 0.2536765 0.3333333
```

```
wilcox.test(higher_ind ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  higher_ind by gender
## W = 7929, p-value = 0.2279
## alternative hypothesis: true location shift is not equal to 0
```

```
# Native speaker test
t.test(edutech_m$native_ind, edutech_f$native_ind)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$native_ind and edutech_f$native_ind
## t = 1.7199, df = 72.406, p-value = 0.08972
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.02020013  0.27440267
## sample estimates:
## mean of x mean of y
## 0.6826568 0.5555556
```

```
wilcox.test(native_ind ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  native_ind by gender
## W = 6387, p-value = 0.07207
## alternative hypothesis: true location shift is not equal to 0
```

```
# Average confidence score tests
t.test(edutech_m$conf_ave, edutech_f$conf_ave)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$conf_ave and edutech_f$conf_ave
## t = 2.2357, df = 46.172, p-value = 0.03024
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.02458082 0.46817823
## sample estimates:
## mean of x mean of y
```

```
##  4.174157  3.927778
```

```
wilcox.test(conf_ave ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  conf_ave by gender
## W = 2520, p-value = 0.04189
## alternative hypothesis: true location shift is not equal to 0
```

```
# Average pre-project confidence score tests
t.test(edutech_m$conf_pre_ave, edutech_f$conf_pre_ave)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$conf_pre_ave and edutech_f$conf_pre_ave
## t = 2.7272, df = 54.836, p-value = 0.008561
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.07052063 0.46146135
## sample estimates:
## mean of x mean of y
##  4.240991  3.975000
```

```
wilcox.test(conf_pre_ave ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  conf_pre_ave by gender
## W = 3248.5, p-value = 0.004486
## alternative hypothesis: true location shift is not equal to 0
```

```
# Average post-project confidence score tests
t.test(edutech_m$conf_post_ave, edutech_f$conf_post_ave)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$conf_post_ave and edutech_f$conf_post_ave
## t = 1.8512, df = 46.121, p-value = 0.07055
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.0200101  0.4786035
## sample estimates:
## mean of x mean of y
##  4.136704  3.907407
```

```
wilcox.test(conf_post_ave ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  conf_post_ave by gender
## W = 2654, p-value = 0.09972
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Study hours
t.test(edutech_m$hours_num, edutech_f$hours_num)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$hours_num and edutech_f$hours_num
## t = -1.066, df = 55.716, p-value = 0.291
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.2201000  0.6780832
## sample estimates:
## mean of x mean of y
##  11.51471  12.28571
```

```r
wilcox.test(hours_num ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  hours_num by gender
## W = 4550, p-value = 0.5144
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Total grade
t.test(edutech_m$total, edutech_f$total)
```

```
##
##  Welch Two Sample t-test
##
## data:  edutech_m$total and edutech_f$total
## t = 1.1517, df = 67.135, p-value = 0.2535
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.8982223  3.3490027
## sample estimates:
## mean of x mean of y
##  88.39769  87.17230
```

```r
wilcox.test(total ~ gender, data=edutech)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  total by gender
## W = 3159.5, p-value = 0.1557
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Check for multicollinearity
cor_subset = edutech[, c("age_num", "native_ind", "higher_ind", "gender_ind")]
cor(na.omit(cor_subset))
```

```
##               age_num  native_ind higher_ind  gender_ind
## age_num    1.00000000  0.03724957  0.1416418  0.04444265
## native_ind 0.03724957  1.00000000 -0.1403130  0.09998062
## higher_ind 0.14164179 -0.14031301  1.0000000 -0.06617940
## gender_ind 0.04444265  0.09998062 -0.0661794  1.00000000
```

```r
# Fit regression to confidence score
conf_lm = lm(conf_ave~gender + age_num + native_ind + higher_ind + semester,
             data=na.omit(edutech))

summary(conf_lm)
```

```
##
## Call:
## lm(formula = conf_ave ~ gender + age_num + native_ind + higher_ind +
##     semester, data = na.omit(edutech))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.20027 -0.30534 -0.05455  0.34311  1.15113
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           4.0698832  0.2308412  17.631   <2e-16 ***
## genderMale            0.2566210  0.1271159   2.019   0.0454 *
## age_num              -0.0004754  0.0055641  -0.085   0.9320
## native_ind           -0.0460818  0.1008011  -0.457   0.6483
## higher_ind           -0.0071892  0.1042549  -0.069   0.9451
## semesterSpring 2016 -0.1998023  0.0892645  -2.238   0.0268 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5351 on 142 degrees of freedom
## Multiple R-squared:  0.05761,    Adjusted R-squared:  0.02442
## F-statistic: 1.736 on 5 and 142 DF,  p-value: 0.1301
```

```r
# Fit regression to pre-project confidence score
conf_pre_lm = lm(conf_pre_ave~gender + age_num + native_ind + higher_ind + semester,
             data=na.omit(edutech))

summary(conf_pre_lm)
```

```
##
## Call:
## lm(formula = conf_pre_ave ~ gender + age_num + native_ind + higher_ind +
##     semester, data = na.omit(edutech))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87912 -0.25019 -0.08141  0.34439  1.18885
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           4.296733   0.241938  17.760   <2e-16 ***
## genderMale            0.293468   0.133227   2.203   0.0292 *
## age_num              -0.006797   0.005832  -1.166   0.2458
## native_ind           -0.081164   0.105647  -0.768   0.4436
## higher_ind           -0.029435   0.109267  -0.269   0.7880
## semesterSpring 2016 -0.152920   0.093556  -1.635   0.1044
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.5608 on 142 degrees of freedom
## Multiple R-squared:  0.06015,    Adjusted R-squared:  0.02706
## F-statistic: 1.818 on 5 and 142 DF,  p-value: 0.113
```

```
# Fit regression to post-project confidence score
conf_post_lm = lm(conf_post_ave~gender + age_num + native_ind + higher_ind + semester,
             data=na.omit(edutech))

summary(conf_post_lm)
```

```
##
## Call:
## lm(formula = conf_post_ave ~ gender + age_num + native_ind +
##     higher_ind + semester, data = na.omit(edutech))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.41437 -0.35916 -0.00726  0.42835  1.19447
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.918650   0.262227  14.944   <2e-16 ***
## genderMale           0.232056   0.144399   1.607   0.1103
## age_num              0.003739   0.006321   0.592   0.5551
## native_ind          -0.022693   0.114506  -0.198   0.8432
## higher_ind           0.007641   0.118430   0.065   0.9486
## semesterSpring 2016 -0.231057   0.101401  -2.279   0.0242 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6078 on 142 degrees of freedom
## Multiple R-squared:  0.05128,    Adjusted R-squared:  0.01787
## F-statistic: 1.535 on 5 and 142 DF,  p-value: 0.1826
```

```
# Fit regression to study hours
hours_lm = lm(hours_num~gender + age_num + native_ind + higher_ind + semester,
             data=na.omit(edutech))

summary(hours_lm)
```

```
##
## Call:
## lm(formula = hours_num ~ gender + age_num + native_ind + higher_ind +
##     semester, data = na.omit(edutech))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6852 -2.1407 -0.5268  1.9832  9.4503
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          9.10745    1.63066   5.585 1.15e-07 ***
## genderMale          -0.73897    0.89794  -0.823   0.4119
## age_num              0.08342    0.03930   2.122   0.0355 *
## native_ind           0.97709    0.71206   1.372   0.1722
```

```
## higher_ind          -1.32422    0.73645  -1.798   0.0743 .
## semesterSpring 2016 -0.28971    0.63056  -0.459   0.6466
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.78 on 142 degrees of freedom
## Multiple R-squared:  0.07322,    Adjusted R-squared:  0.04059
## F-statistic: 2.244 on 5 and 142 DF,  p-value: 0.05318
```

```
# Fit regression to grade
grades_lm = lm(total~gender + age_num + native_ind + higher_ind + semester,
             data=na.omit(edutech))

summary(grades_lm)
```

```
##
## Call:
## lm(formula = total ~ gender + age_num + native_ind + higher_ind +
##     semester, data = na.omit(edutech))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.167   0.021   0.826   1.641   2.998
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         83.27319    2.14412  38.838   <2e-16 ***
## genderMale          -0.41673    1.18069  -0.353    0.725
## age_num              0.03076    0.05168   0.595    0.553
## native_ind           0.35390    0.93627   0.378    0.706
## higher_ind           0.20410    0.96835   0.211    0.833
## semesterSpring 2016  9.49967    0.82911  11.458   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.97 on 142 degrees of freedom
## Multiple R-squared:  0.4845, Adjusted R-squared:  0.4664
## F-statistic:  26.7 on 5 and 142 DF,  p-value: < 2.2e-16
```