Educational Technology Project - Combined Data Analysis - EduTech (Fall 2015, Spring 2016 and Fall 2016), KBAI (Summer 2015 and Summer 2016) and HCI (Fall 2016) Data Analysis

Process Data

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

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

# Read in survey data sets
CS6460_fall15_soc = read.csv('Survey_CS6460_FALL15_SOC.csv')
CS6460_fall15_qc = read.csv('Survey_CS6460_FALL15_QC.csv')
CS6460_fall15_mc = read.csv('Survey_CS6460_FALL15_MC.csv')
CS6460_fall15_eoc = read.csv('Survey_CS6460_FALL15_EOC.csv')

CS6460_spr16_soc = read.csv('Survey_CS6460_SPR16_SOC.csv')
CS6460_spr16_qc = read.csv('Survey_CS6460_SPR16_QC.csv')
CS6460_spr16_mc = read.csv('Survey_CS6460_SPR16_MC.csv')
CS6460_spr16_eoc = read.csv('Survey_CS6460_SPR16_EOC.csv')

CS6460_fall16_soc = read.csv('Survey_CS6460_FALL16_SOC.csv')
CS6460_fall16_qc = read.csv('Survey_CS6460_FALL16_QC.csv')
CS6460_fall16_mc = read.csv('Survey_CS6460_FALL16_MC.csv')
CS6460_fall16_eoc = read.csv('Survey_CS6460_FALL16_EOC.csv')

CS7637_sum15_soc = read.csv('Survey_CS7637_SUM15_SOC.csv')
CS7637_sum15_qc = read.csv('Survey_CS7637_SUM15_QC.csv')
CS7637_sum15_mc = read.csv('Survey_CS7637_SUM15_MC.csv')
CS7637_sum15_eoc = read.csv('Survey_CS7637_SUM15_EOC.csv')

CS7637_sum16_soc = read.csv('Survey_CS7637_SUM16_SOC.csv')
CS7637_sum16_qc = read.csv('Survey_CS7637_SUM16_QC.csv')
CS7637_sum16_mc = read.csv('Survey_CS7637_SUM16_MC.csv')
CS7637_sum16_eoc = read.csv('Survey_CS7637_SUM16_EOC.csv')

CS6750_fall16_soc = read.csv('Survey_CS6750_FALL16_SOC.csv')
CS6750_fall16_qc = read.csv('Survey_CS6750_FALL16_QC.csv')
CS6750_fall16_mc = read.csv('Survey_CS6750_FALL16_MC.csv')
CS6750_fall16_eoc = read.csv('Survey_CS6750_FALL16_EOC.csv')
```

```r
# Create data subsets containing information of interest and change names
# CS6460 - EduTech
CS6460_fall15_soc = CS6460_fall15_soc[, c(1, 2, 3, 4, 5, 7, 8, 10)]
colnames(CS6460_fall15_soc) = c("student", "age", "gender", "birth", "residence",
                                "language", "english", "education")

CS6460_spr16_soc = CS6460_spr16_soc[, c(1, 2, 3, 4, 5, 7, 8, 10)]
colnames(CS6460_spr16_soc) = c("student", "age", "gender", "birth", "residence",
                               "language", "english", "education")

CS6460_fall16_soc = CS6460_fall16_soc[, c(1, 2, 3, 4, 5, 7, 8, 10)]
colnames(CS6460_fall16_soc) = c("student", "age", "gender", "birth", "residence",
                                "language", "english", "education")


CS6460_fall15_qc = CS6460_fall15_qc[, c(1, 2, 3)]
colnames(CS6460_fall15_qc) = c("student", "conf_p1_post", "conf_p2_pre")

CS6460_spr16_qc = CS6460_spr16_qc[, c(1, 2, 3)]
colnames(CS6460_spr16_qc) = c("student", "conf_p1_post", "conf_p2_pre")

CS6460_fall16_qc = CS6460_fall16_qc[, c(1, 13, 14)]
colnames(CS6460_fall16_qc) = c("student", "conf_p1_post", "conf_p2_pre")


CS6460_fall15_mc = CS6460_fall15_mc[, c(1, 2, 3)]
colnames(CS6460_fall15_mc) = c("student", "conf_p2_post", "conf_p3_pre")

CS6460_spr16_mc = CS6460_spr16_mc[, c(1, 2, 3)]
colnames(CS6460_spr16_mc) = c("student", "conf_p2_post", "conf_p3_pre")

CS6460_fall16_mc = CS6460_fall16_mc[, c(1, 2, 3)]
colnames(CS6460_fall16_mc) = c("student", "conf_p2_post", "conf_p3_pre")


CS6460_fall15_eoc = CS6460_fall15_eoc[, c(1, 2, 11)]
colnames(CS6460_fall15_eoc) = c("student", "hours", "conf_p3_post")

CS6460_spr16_eoc = CS6460_spr16_eoc[, c(1, 2, 10)]
colnames(CS6460_spr16_eoc) = c("student", "hours", "conf_p3_post")

CS6460_fall16_eoc = CS6460_fall16_eoc[, c(1, 6, 14)]
colnames(CS6460_fall16_eoc) = c("student", "hours", "conf_p3_post")

# CS7637 - KBAI
CS7637_sum15_soc = CS7637_sum15_soc[, c(1, 2, 3, 4, 5, 7, 8, 16)]
colnames(CS7637_sum15_soc) = c("student", "age", "gender", "birth", "residence",
                               "language", "english", "education")

CS7637_sum16_soc = CS7637_sum16_soc[, c(1, 2, 3, 4, 5, 7, 8, 11)]
colnames(CS7637_sum16_soc) = c("student", "age", "gender", "birth", "residence",
                               "language", "english", "education")
```

```
CS7637_sum15_qc = CS7637_sum15_qc[, c(1, 4, 5)]
colnames(CS7637_sum15_qc) = c("student", "conf_p1_post", "conf_p2_pre")

CS7637_sum16_qc = CS7637_sum16_qc[, c(1, 3, 4)]
colnames(CS7637_sum16_qc) = c("student", "conf_p1_post", "conf_p2_pre")

CS7637_sum15_mc = CS7637_sum15_mc[, c(1, 4, 5)]
colnames(CS7637_sum15_mc) = c("student", "conf_p2_post", "conf_p3_pre")

CS7637_sum16_mc = CS7637_sum16_mc[, c(1, 3, 4)]
colnames(CS7637_sum16_mc) = c("student", "conf_p2_post", "conf_p3_pre")

CS7637_sum15_eoc = CS7637_sum15_eoc[, c(1, 3, 2)]
colnames(CS7637_sum15_eoc) = c("student", "hours", "conf_p3_post")

CS7637_sum16_eoc = CS7637_sum16_eoc[, c(1, 3, 2)]
colnames(CS7637_sum16_eoc) = c("student", "hours", "conf_p3_post")

# CS6750 - HCI
CS6750_fall16_soc = CS6750_fall16_soc[, c(1, 2, 3, 4, 5, 7, 8, 11)]
colnames(CS6750_fall16_soc) = c("student", "age", "gender", "birth", "residence",
                                "language", "english", "education")

CS6750_fall16_qc = CS6750_fall16_qc[, c(1, 2, 3)]
colnames(CS6750_fall16_qc) = c("student", "conf_p1_post", "conf_p2_pre")

CS6750_fall16_mc = CS6750_fall16_mc[, c(1, 2, 3)]
colnames(CS6750_fall16_mc) = c("student", "conf_p2_post", "conf_p3_pre")

CS6750_fall16_eoc = CS6750_fall16_eoc[, c(1, 3, 2)]
colnames(CS6750_fall16_eoc) = c("student", "hours", "conf_p3_post")


# Merge EduTech datasets
edutech_data_fall15 = merge(x = CS6460_fall15_soc, y = CS6460_fall15_qc,
                            by = "student", all.x = TRUE)
edutech_data_fall15 = merge(x = edutech_data_fall15, y = CS6460_fall15_mc,
                            by = "student", all.x = TRUE)
edutech_data_fall15 = merge(x = edutech_data_fall15, y = CS6460_fall15_eoc,
                            by = "student", all.x = TRUE)

edutech_data_spr16 = merge(x = CS6460_spr16_soc, y = CS6460_spr16_qc,
                           by = "student", all.x = TRUE)
edutech_data_spr16 = merge(x = edutech_data_spr16, y = CS6460_spr16_mc,
                           by = "student", all.x = TRUE)
edutech_data_spr16 = merge(x = edutech_data_spr16, y = CS6460_spr16_eoc,
                           by = "student", all.x = TRUE)

edutech_data_fall16 = merge(x = CS6460_fall16_soc, y = CS6460_fall16_qc,
                            by = "student", all.x = TRUE)
edutech_data_fall16 = merge(x = edutech_data_fall16, y = CS6460_fall16_mc,
                            by = "student", all.x = TRUE)
edutech_data_fall16 = merge(x = edutech_data_fall16, y = CS6460_fall16_eoc,
```

```r
                            by = "student", all.x = TRUE)


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)

edutech$course = "EduTech"

# Drop unneeded datasets
rm(CS6460_fall15_soc, CS6460_fall15_qc, CS6460_fall15_mc, CS6460_fall15_eoc,
   CS6460_spr16_soc, CS6460_spr16_qc, CS6460_spr16_mc, CS6460_spr16_eoc,
   CS6460_fall16_soc, CS6460_fall16_qc, CS6460_fall16_mc, CS6460_fall16_eoc,
   edutech_data_fall15, edutech_data_spr16, edutech_data_fall16)

# Merge KBAI datasets
kbai_data_sum15 = merge(x = CS7637_sum15_soc, y = CS7637_sum15_qc,
                        by = "student", all.x = TRUE)
kbai_data_sum15 = merge(x = kbai_data_sum15, y = CS7637_sum15_mc,
                        by = "student", all.x = TRUE)
kbai_data_sum15 = merge(x = kbai_data_sum15, y = CS7637_sum15_eoc,
                        by = "student", all.x = TRUE)

kbai_data_sum16 = merge(x = CS7637_sum16_soc, y = CS7637_sum16_qc,
                        by = "student", all.x = TRUE)
kbai_data_sum16 = merge(x = kbai_data_sum16, y = CS7637_sum16_mc,
                        by = "student", all.x = TRUE)
kbai_data_sum16 = merge(x = kbai_data_sum16, y = CS7637_sum16_eoc,
                        by = "student", all.x = TRUE)


kbai_data_sum15$semester = "Summer 2015"
kbai_data_sum16$semester = "Summer 2016"

kbai = rbind(kbai_data_sum15, kbai_data_sum16)

kbai$course = "KBAI"

# Drop unneeded datasets
rm(kbai_data_sum15, kbai_data_sum16, CS7637_sum15_eoc, CS7637_sum15_mc, CS7637_sum15_qc,
   CS7637_sum15_soc, CS7637_sum16_eoc, CS7637_sum16_mc, CS7637_sum16_qc, CS7637_sum16_soc)

# Merge HCI datasets
hci = merge(x = CS6750_fall16_soc, y = CS6750_fall16_qc, by = "student", all.x = TRUE)
hci = merge(x = hci, y = CS6750_fall16_mc, by = "student", all.x = TRUE)
hci = merge(x = hci, y = CS6750_fall16_eoc, by = "student", all.x = TRUE)

hci$semester = "Fall 2016"

hci$course = "HCI"
```

```r
# Drop unneeded datasets
rm(CS6750_fall16_soc, CS6750_fall16_qc, CS6750_fall16_mc, CS6750_fall16_eoc)

# Stack data sets
combined = rbind(kbai, edutech, hci)

# Drop unneeded datasets
rm(kbai, edutech, hci)

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

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

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

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

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

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

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

combined$conf_p2_post = revalue(combined$conf_p2_post, c("Very confident" = 5,
                    "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                    "Somewhat unconfident" = 2, "Very unconfident" = 1))
```

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

combined$conf_p3_post = revalue(combined$conf_p3_post, c("Very confident" = 5,
                    "Somewhat confident" = 4, "Neither confident nor unconfident" = 3,
                    "Somewhat unconfident" = 2, "Very unconfident" = 1))

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

combined$hours = revalue(combined$hours, c("<3 hours per week" = "0-3",
                "3 - 6 hours per week" = "3-6", "6 - 9 hours per week" = "6-9",
                "9 - 12 hours per week" = "9-12", "12 - 15 hours per week" = "12-15",
                "15 - 18 hours per week" = "15-18", "18 - 21 hours per week" = "18-21",
                "21 or more hours per week" = "21+"))

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

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

combined$course = factor(combined$course, levels = c("KBAI", "HCI", "EduTech"))
combined$semester = factor(combined$semester, levels = c("Fall 2016", "Summer 2016",
                                              "Spring 2016", "Fall 2015", "Summer 2015"))

# Create function for removing "1:" from text fields and convert to title case
text_split = function(x){
  x = unlist(strsplit(x, ": "))[2]
  return(toTitleCase(x))
}

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

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

# Clean birth country names
combined$birth = ifelse(combined$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",
                    combined$birth)

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

# Create alternative birth groupings
combined$birth2 = combined$birth
combined$birth2 = ifelse(combined$birth %in% c("Syria", "Taiwan", "Vietnam",
    "Pakistan", "Japan", "Korea", "Kuwait", "Philippines", "Indonesia",
    "Sri Lanka", "Singapore", "Nepal", "Turkey", "Kazakhstan", "Iran",
    "Afghanistan", "Thailand", "Myanmar", "Lebanon", "Tunisia", "UAE",
    "Bangladesh", "Qatar", "Malaysia"), "Other Asia", combined$birth2)
combined$birth2 = ifelse(combined$birth %in% c("Ukraine", "Italy", "Norway",
    "Serbia", "Moldova", "Czech Republic", "Poland", "Russia", "Switzerland",
    "Germany", "Bulgaria", "UK", "Finland", "Romania", "Lithuania",
    "Luxembourg"), "Europe", combined$birth2)
combined$birth2 = ifelse(combined$birth %in% c("Puerto Rico", "Canada",
    "Dominican Republic", "Mexico", "Dominica", "El Salvador", "Cuba",
    "Haiti", "Bahamas", "Guatemala", "Panama", "Grenada", "Honduras",
    "Nicaragua", "The Bahamas", "Trinidad and Tobago"), "Other Nth America",
    combined$birth2)
combined$birth2 = ifelse(combined$birth %in% c("Peru", "Ecuador", "Colombia",
    "Brazil", "Argentina", "Chile"), "Sth America", combined$birth2)
combined$birth2 = ifelse(combined$birth %in% c("Nigeria", "Kenya",
    "South Africa", "Ethiopia", "Ghana", "Rwanda"), "Africa", combined$birth2)

combined$birth2 = ifelse(combined$birth %in% c("Australia", "New Zealand"),
    "Other", combined$birth2)

unique(combined$birth2)

# Clean residence country names
combined$residence = ifelse(combined$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", "JS"), "USA", combined$residence)

combined$residence = ifelse(combined$residence == "NA", NA, combined$residence)
combined$residence = ifelse(combined$residence == "Myanmar, Hong Kong", "Myanmar",
                            combined$residence)
combined$residence = ifelse(combined$residence %in% c("China", "Hong Kong"), "China",
                            combined$residence)
combined$residence = ifelse(combined$residence == "United Kingdom", "UK", combined$residence)

# Clean language
combined$language = ifelse(combined$language %in% c("English", "American English", "ENGLISH",
                  "American", "English (US)", "English Language", "Englist",
                  "C++, but you Probably Mean \"English\"", "ENGLISH", "En", "JavaScript",
                  "Elijah", "Dallas", "First",
                  "English and French", "English, Cantonese", "Java",
                  "Conative American Sign Language and English"), "English",
                  combined$language)
combined$language = ifelse(combined$language %in% c("Chinese", "Mandarin", "China",
                  "Mandarin Chinese", "Cantonese", "Chiinese", "CHINESE", "Manderin",
                  "Java", "Python"), "Chinese", combined$language)
combined$language = ifelse(combined$language %in% c("Marathi", "Telugu", "Bengali",
                  "Gujarati",
                  "Kannada", "Hindi", "Tamil", "Odiya", "TAMIL", "Punjabi", "Hindo",
                  "Indian Language"), "Indian", combined$language)
combined$language = ifelse(combined$language %in% c("Principal", "Korean", "South Korean"),
                  "Korean", combined$language)
combined$language = ifelse(combined$language == "Farsi/English", "Farsi", combined$language)
combined$language = ifelse(combined$language == "Spanish/English", "Spanish",
                           combined$language)
combined$language = ifelse(combined$language %in% c("Swiss German", "German", "Germany"),
                  "German", combined$language)
combined$language = ifelse(combined$language %in% c("Persian", "Persian (Farsi)"), "Farsi",
                  combined$language)
combined$language = ifelse(combined$language %in% c("Thai", "ABAP"), "Thai",
                  combined$language)
combined$language = ifelse(combined$language == "NA", NA, combined$language)


# Create factors
combined$birth = factor(combined$birth)
combined$birth2 = factor(combined$birth2)
combined$residence = factor(combined$residence)
combined$language = factor(combined$language)
combined$semester = factor(combined$semester)

# Convert confidence scores to numeric
combined$conf_p1_post = as.numeric(as.character(combined$conf_p1_post))
```

```
## Warning: NAs introduced by coercion
```

```
combined$conf_p2_pre = as.numeric(as.character(combined$conf_p2_pre))
```

```
## Warning: NAs introduced by coercion
```

```
combined$conf_p2_post = as.numeric(as.character(combined$conf_p2_post))
```

```
## Warning: NAs introduced by coercion
```

```
combined$conf_p3_pre = as.numeric(as.character(combined$conf_p3_pre))
```

```
## Warning: NAs introduced by coercion
```

```
combined$conf_p3_post = as.numeric(as.character(combined$conf_p3_post))
```

```
## Warning: NAs introduced by coercion
```

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

combined$conf_pre_ave = (combined$conf_p2_pre + combined$conf_p3_pre)/2
```

```
combined$conf_post_ave = (combined$conf_p1_post + combined$conf_p2_post +
                            combined$conf_p3_post)/3

# Convert ranges to numeric values
combined$age_num = revalue(combined$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))
combined$age_num = as.numeric(as.character(combined$age_num))

combined$hours_num = revalue(combined$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))
combined$hours_num = as.numeric(as.character(combined$hours_num))

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

# Drop NA values
combined = subset(combined, !is.na(student))
```

Explore Data

```
# Calculate proportion & frequency of data set by course
prop.table(table(combined$course))
```

```
##
##       KBAI        HCI    EduTech
## 0.58366534 0.08266932 0.33366534
```

```
count(combined$course)
```

```
##         x freq
## 1    KBAI  586
## 2     HCI   83
## 3 EduTech  335
```

```
# Calculate counts by course and semester
data.frame(table(combined[,c("semester", "course")]))
```

```
##       semester  course Freq
## 1    Fall 2016    KBAI    0
## 2  Summer 2016    KBAI  299
## 3  Spring 2016    KBAI    0
## 4    Fall 2015    KBAI    0
## 5  Summer 2015    KBAI  287
## 6    Fall 2016     HCI   83
## 7  Summer 2016     HCI    0
## 8  Spring 2016     HCI    0
## 9    Fall 2015     HCI    0
## 10 Summer 2015     HCI    0
## 11   Fall 2016 EduTech  124
## 12 Summer 2016 EduTech    0
## 13 Spring 2016 EduTech  117
## 14   Fall 2015 EduTech   94
## 15 Summer 2015 EduTech    0
```

```
# Calculate counts by course and semester and gender
data.frame(table(combined[,c("semester", "course", "gender")]))
```

```
##       semester  course gender Freq
## 1    Fall 2016    KBAI Female    0
## 2  Summer 2016    KBAI Female   37
## 3  Spring 2016    KBAI Female    0
## 4    Fall 2015    KBAI Female    0
## 5  Summer 2015    KBAI Female   39
## 6    Fall 2016     HCI Female   20
## 7  Summer 2016     HCI Female    0
## 8  Spring 2016     HCI Female    0
## 9    Fall 2015     HCI Female    0
## 10 Summer 2015     HCI Female    0
## 11   Fall 2016 EduTech Female   25
## 12 Summer 2016 EduTech Female    0
## 13 Spring 2016 EduTech Female   12
## 14   Fall 2015 EduTech Female   17
## 15 Summer 2015 EduTech Female    0
## 16   Fall 2016    KBAI   Male    0
## 17 Summer 2016    KBAI   Male  257
## 18 Spring 2016    KBAI   Male    0
## 19   Fall 2015    KBAI   Male    0
## 20 Summer 2015    KBAI   Male  244
## 21   Fall 2016     HCI   Male   60
## 22 Summer 2016     HCI   Male    0
## 23 Spring 2016     HCI   Male    0
## 24   Fall 2015     HCI   Male    0
## 25 Summer 2015     HCI   Male    0
## 26   Fall 2016 EduTech   Male   95
## 27 Summer 2016 EduTech   Male    0
## 28 Spring 2016 EduTech   Male  105
## 29   Fall 2015 EduTech   Male   72
## 30 Summer 2015 EduTech   Male    0
```

```
# Determine number of duplicates
student_cnt = count(combined, "student")
student_cnt = student_cnt[order(-student_cnt$freq),]
multiple = subset(student_cnt, freq > 1)

dim(multiple)[1]
```

```
## [1] 141
```

```
min(multiple$freq)
```

```
## [1] 2
```

```
max(multiple$freq)
```

```
## [1] 3
```

```
dim(subset(multiple, freq == 2))[1]
```

```
## [1] 132
```

```r
dim(subset(multiple, freq == 3))[1]
```

```
## [1] 9
```

```r
# For duplicates, keep the most recent occurrence of student in data set and drop
#earlier values
combined = with(combined, combined[order(course, semester),])

combined = combined[!duplicated(combined$student),]

# Calculate summary statistics
summary(combined)
```

```
##    student                age         gender         birth
##  Length:854         Under 18:  0   Female:123   USA    :468
##  Class :character   18 to 24: 95   Male   :715   India  : 87
##  Mode  :character   25 to 34:467   NA's   : 16   China  : 79
##                     35 to 44:201                 Canada : 17
##                     45 to 54: 62                 Korea  : 11
##                     55 to 64: 15                 (Other):178
##                     NA's    : 14                 NA's   : 14
##      residence      language      english          education
##  USA      :741   English:603   Fluent :269   Bachelors  :617
##  Canada   : 22   Chinese: 76   Native :551   Doctorate  : 53
##  India    : 16   Indian : 46   Partial: 16   High School:  2
##  China    :  6   Spanish: 31   NA's   : 18   Masters    :163
##  Australia:  5   Korean :  9                 NA's       : 19
##  (Other)  : 49   (Other): 72
##  NA's     : 15   NA's   : 17
##   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.:3.000   1st Qu.:3.000   1st Qu.:4.000   1st Qu.:3.000
##  Median :4.000   Median :4.000   Median :4.000   Median :4.000
##  Mean   :3.868   Mean   :3.875   Mean   :3.993   Mean   :3.717
##  3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.:5.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000
##  NA's   :118     NA's   :118     NA's   :140     NA's   :139
##      hours      conf_p3_post          semester          course
##  9-12   :156   Min.   :1.000   Fall 2016  :134   KBAI   :585
##  12-15  :130   1st Qu.:3.000   Summer 2016:299   HCI    : 65
##  6-9    : 88   Median :4.000   Spring 2016: 75   EduTech:204
##  15-18  : 81   Mean   :3.738   Fall 2015  : 60
##  18-21  : 61   3rd Qu.:4.000   Summer 2015:286
##  (Other): 89   Max.   :5.000
##  NA's   :249   NA's   :250
##                 birth2       conf_ave      conf_pre_ave   conf_post_ave
##  USA              :468   Min.   :1.400   Min.   :1.000   Min.   :1.000
##  India            : 87   1st Qu.:3.400   1st Qu.:3.500   1st Qu.:3.333
##  Other Asia       : 81   Median :4.000   Median :4.000   Median :4.000
##  China            : 79   Mean   :3.848   Mean   :3.786   Mean   :3.878
##  Other Nth America: 48   3rd Qu.:4.400   3rd Qu.:4.500   3rd Qu.:4.333
##  (Other)          : 77   Max.   :5.000   Max.   :5.000   Max.   :5.000
##  NA's             : 14   NA's   :321     NA's   :201     NA's   :318
##     age_num        hours_num       native_ind       higher_ind
##  Min.   :21.00   Min.   : 1.50   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:29.50   1st Qu.:10.50   1st Qu.:0.0000   1st Qu.:0.0000  
##  Median :29.50   Median :13.50   Median :1.0000   Median :0.0000  
##  Mean   :32.94   Mean   :12.72   Mean   :0.6591   Mean   :0.2529  
##  3rd Qu.:39.50   3rd Qu.:16.50   3rd Qu.:1.0000   3rd Qu.:1.0000  
##  Max.   :59.50   Max.   :21.00   Max.   :1.0000   Max.   :1.0000  
##  NA's   :14      NA's   :249     NA's   :18                       
##    gender_ind    
##  Min.   :0.0000  
##  1st Qu.:1.0000  
##  Median :1.0000  
##  Mean   :0.8532  
##  3rd Qu.:1.0000  
##  Max.   :1.0000  
##  NA's   :16
```

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

```
## 
##   Female     Male 
## 0.146778 0.853222
```

```r
# Calculate proportion & frequency of data set by course
prop.table(table(combined$course))
```

```
## 
##       KBAI        HCI    EduTech 
## 0.68501171 0.07611241 0.23887588
```

```r
count(combined$course)
```

```
##         x freq
## 1    KBAI  585
## 2     HCI   65
## 3 EduTech  204
```

```r
# Calculate proportion of data set by semester
prop.table(table(combined$semester))
```

```
## 
##   Fall 2016 Summer 2016 Spring 2016   Fall 2015 Summer 2015 
##  0.15690867  0.35011710  0.08782201  0.07025761  0.33489461
```

Analyze Data by Gender

```r
# Calculate age summary statistics
ddply(subset(combined, !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 33.60569 8.847120   29.5    29.5    39.5
## 2   Male 32.81119 7.974224   29.5    29.5    39.5
```

```r
# Calculate study hours summary statistics
ddply(subset(combined, !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 13.42105 4.957434   13.5    10.5    16.5
## 2   Male 12.58449 4.660305   13.5    10.5    16.5
```

```r
# Calculate confidence summary statistics
ddply(subset(combined, !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 3.779310 0.7153041      4     3.3     4.2
## 2   Male 3.858636 0.6869549      4     3.4     4.4
```

```r
# Calculate confidence summary statistics
ddply(subset(combined, !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 3.659794 0.8150485      4     3.5     4.0
## 2   Male 3.802368 0.8428607      4     3.5     4.5
```

```r
ddply(subset(combined, !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 3.827586 0.8099224      4 3.333333 4.333333
## 2   Male 3.886381 0.7698325      4 3.333333 4.333333
```

```r
combined_m = subset(combined, gender == "Male")
combined_f = subset(combined, gender == "Female")

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

```
## 
##   Under 18   18 to 24   25 to 34   35 to 44   45 to 54   55 to 64 
## 0.00000000 0.11888112 0.54965035 0.24335664 0.07552448 0.01258741
```

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

```
## 
##   Under 18   18 to 24   25 to 34   35 to 44   45 to 54   55 to 64 
## 0.00000000 0.08130081 0.59349593 0.21951220 0.05691057 0.04878049
```

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

```
## 
##         Afghanistan           Argentina           Australia 
##         0.001398601         0.004195804         0.006993007
##             Bahamas          Bangladesh              Brazil 
##         0.002797203         0.001398601         0.008391608 
##            Bulgaria              Canada               Chile 
##         0.002797203         0.023776224         0.001398601 
##               China            Colombia                Cuba 
##         0.076923077         0.001398601         0.001398601 
##      Czech Republic            Dominica  Dominican Republic 
##         0.001398601         0.001398601         0.002797203 
##             Ecuador         El Salvador            Ethiopia 
##         0.001398601         0.002797203         0.002797203 
##             Finland             Germany               Ghana 
##         0.000000000         0.005594406         0.000000000 
##             Grenada           Guatemala               Haiti 
##         0.001398601         0.001398601         0.001398601 
##            Honduras               India           Indonesia 
##         0.001398601         0.099300699         0.002797203 
##                Iran               Italy               Japan 
##         0.005594406         0.002797203         0.005594406 
##          Kazakhstan               Kenya               Korea 
##         0.001398601         0.004195804         0.011188811 
##              Kuwait             Lebanon           Lithuania 
##         0.001398601         0.002797203         0.001398601 
##          Luxembourg            Malaysia              Mexico 
##         0.001398601         0.001398601         0.013986014 
##             Moldova             Myanmar               Nepal 
##         0.000000000         0.002797203         0.006993007 
##         New Zealand           Nicaragua             Nigeria 
##         0.001398601         0.001398601         0.004195804 
##              Norway            Pakistan              Panama 
##         0.002797203         0.012587413         0.005594406 
##                Peru         Philippines              Poland 
##         0.005594406         0.004195804         0.002797203 
##         Puerto Rico               Qatar             Romania 
##         0.001398601         0.000000000         0.002797203 
##              Russia              Rwanda              Serbia 
##         0.005594406         0.001398601         0.001398601 
##           Singapore        South Africa           Sri Lanka 
##         0.002797203         0.001398601         0.001398601 
##         Switzerland               Syria              Taiwan 
##         0.001398601         0.001398601         0.009790210 
##            Thailand         The Bahamas Trinidad and Tobago 
##         0.002797203         0.000000000         0.001398601 
##             Tunisia              Turkey                 UAE 
##         0.001398601         0.004195804         0.001398601 
##                  UK             Ukraine                 USA 
##         0.005594406         0.004195804         0.573426573 
##             Vietnam 
##         0.012587413
```

```
prop.table(table(combined_f$birth))
```

```
## 
##         Afghanistan           Argentina           Australia 
##         0.000000000         0.008130081         0.008130081 
##             Bahamas          Bangladesh              Brazil
##         0.000000000         0.000000000         0.000000000
##            Bulgaria              Canada               Chile
##         0.000000000         0.000000000         0.000000000
##               China            Colombia                Cuba
##         0.195121951         0.008130081         0.016260163
##      Czech Republic            Dominica  Dominican Republic
##         0.000000000         0.000000000         0.000000000
##             Ecuador         El Salvador            Ethiopia
##         0.016260163         0.000000000         0.000000000
##             Finland             Germany               Ghana
##         0.008130081         0.000000000         0.008130081
##             Grenada           Guatemala               Haiti
##         0.000000000         0.000000000         0.000000000
##            Honduras               India           Indonesia
##         0.000000000         0.130081301         0.000000000
##                Iran               Italy               Japan
##         0.000000000         0.008130081         0.000000000
##          Kazakhstan               Kenya               Korea
##         0.000000000         0.024390244         0.024390244
##              Kuwait             Lebanon           Lithuania
##         0.000000000         0.000000000         0.000000000
##          Luxembourg            Malaysia              Mexico
##         0.000000000         0.000000000         0.000000000
##             Moldova             Myanmar               Nepal
##         0.008130081         0.000000000         0.008130081
##         New Zealand           Nicaragua             Nigeria
##         0.000000000         0.000000000         0.000000000
##              Norway            Pakistan              Panama
##         0.000000000         0.000000000         0.000000000
##                Peru         Philippines              Poland
##         0.000000000         0.016260163         0.000000000
##         Puerto Rico               Qatar             Romania
##         0.000000000         0.008130081         0.008130081
##              Russia              Rwanda              Serbia
##         0.000000000         0.000000000         0.008130081
##           Singapore        South Africa           Sri Lanka
##         0.008130081         0.000000000         0.000000000
##         Switzerland               Syria              Taiwan
##         0.000000000         0.000000000         0.008130081
##            Thailand         The Bahamas Trinidad and Tobago
##         0.000000000         0.000000000         0.000000000
##             Tunisia              Turkey                 UAE
##         0.000000000         0.000000000         0.000000000
##                  UK             Ukraine                 USA
##         0.000000000         0.008130081         0.455284553
##             Vietnam
##         0.008130081
```

```
# Compare birth country2
prop.table(table(combined_m$birth2))
```

```
## 
##            Africa             China            Europe             India 
##       0.013986014       0.076923077       0.041958042       0.099300699 
##             Other        Other Asia Other Nth America       Sth America
##       0.008391608       0.099300699       0.064335664       0.022377622
##               USA
##       0.573426573
```

```
prop.table(table(combined_f$birth2))
```

```
##
##            Africa             China            Europe             India
##       0.032520325       0.195121951       0.048780488       0.130081301
##             Other        Other Asia Other Nth America       Sth America
##       0.008130081       0.081300813       0.016260163       0.032520325
##               USA
##       0.455284553
```

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

```
##
##      Australia        Bahamas         Brazil         Canada          Chile
##    0.005602241    0.001400560    0.001400560    0.029411765    0.001400560
##          China       Colombia Czech Republic    El Salvador        Germany
##    0.008403361    0.001400560    0.000000000    0.001400560    0.002801120
##        Grenada      Guatemala          India      Indonesia        Ireland
##    0.001400560    0.001400560    0.019607843    0.001400560    0.001400560
##         Israel          Italy          Japan          Kenya       Malaysia
##    0.000000000    0.000000000    0.002801120    0.002801120    0.001400560
##        Myanmar    Netherlands    New Zealand       Pakistan         Panama
##    0.001400560    0.002801120    0.001400560    0.002801120    0.001400560
##           Peru      Singapore    South Korea         Sweden    Switzerland
##    0.001400560    0.005602241    0.004201681    0.001400560    0.001400560
##         Taiwan    The Bahamas        Tunisia            UAE             UK
##    0.002801120    0.000000000    0.001400560    0.001400560    0.002801120
##        Ukraine            USA        Vietnam
##    0.001400560    0.879551821    0.001400560
```

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

```
##
##      Australia        Bahamas         Brazil         Canada          Chile
##    0.008130081    0.000000000    0.000000000    0.008130081    0.000000000
##          China       Colombia Czech Republic    El Salvador        Germany
##    0.000000000    0.000000000    0.000000000    0.000000000    0.000000000
##        Grenada      Guatemala          India      Indonesia        Ireland
##    0.000000000    0.000000000    0.016260163    0.000000000    0.000000000
##         Israel          Italy          Japan          Kenya       Malaysia
##    0.008130081    0.008130081    0.016260163    0.024390244    0.000000000
##        Myanmar    Netherlands    New Zealand       Pakistan         Panama
##    0.000000000    0.000000000    0.000000000    0.000000000    0.000000000
##           Peru      Singapore    South Korea         Sweden    Switzerland
##    0.000000000    0.008130081    0.000000000    0.000000000    0.000000000
##         Taiwan    The Bahamas        Tunisia            UAE             UK
##    0.000000000    0.000000000    0.000000000    0.000000000    0.000000000
##        Ukraine            USA        Vietnam
##    0.000000000    0.902439024    0.000000000
```

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

```
##
##         Arabic      Bulgarian        Burmese      Cambodian        Chinese
##    0.004213483    0.001404494    0.002808989    0.000000000    0.078651685
##          Czech           Dari        English          Farsi       Filipino
##    0.001404494    0.000000000    0.733146067    0.007022472    0.000000000
##         French         German Haitian Creole         Indian     Indonesian
##    0.001404494    0.004213483    0.001404494    0.053370787    0.002808989
##        Italian       Japanese         Korean     Lithuanian      Malayalam
##    0.000000000    0.001404494    0.008426966    0.001404494    0.002808989
##         Nepali      Norwegian         Polish     Portuguese       Romanian
##    0.005617978    0.002808989    0.001404494    0.008426966    0.001404494
##        Russian        Serbian        Spanish        Swahili        Tagalog
##    0.007022472    0.001404494    0.036516854    0.001404494    0.002808989
##           Thai        Turkish      Ukrainian           Urdu     Vietnamese
##    0.002808989    0.004213483    0.001404494    0.008426966    0.008426966
```

```r
prop.table(table(combined_f$language))
```

```
##
##         Arabic      Bulgarian        Burmese      Cambodian        Chinese
##    0.008130081    0.000000000    0.000000000    0.000000000    0.162601626
##          Czech           Dari        English          Farsi       Filipino
##    0.000000000    0.000000000    0.642276423    0.000000000    0.008130081
##         French         German Haitian Creole         Indian     Indonesian
##    0.000000000    0.000000000    0.000000000    0.065040650    0.000000000
##        Italian       Japanese         Korean     Lithuanian      Malayalam
##    0.008130081    0.000000000    0.024390244    0.000000000    0.016260163
##         Nepali      Norwegian         Polish     Portuguese       Romanian
##    0.000000000    0.000000000    0.000000000    0.000000000    0.008130081
##        Russian        Serbian        Spanish        Swahili        Tagalog
##    0.000000000    0.000000000    0.040650407    0.000000000    0.008130081
##           Thai        Turkish      Ukrainian           Urdu     Vietnamese
##    0.000000000    0.000000000    0.000000000    0.000000000    0.008130081
```

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

```
##
##     Fluent     Native    Partial
## 0.30239100 0.67932489 0.01828411
```

```r
prop.table(table(combined_f$english))
```

```
##
##     Fluent     Native    Partial
## 0.43902439 0.53658537 0.02439024
```

```r
# Compare education
prop.table(table(combined_m$education))
```

```
##
##   Bachelors   Doctorate High School     Masters
## 0.753521127 0.049295775 0.002816901 0.194366197
```

```r
prop.table(table(combined_f$education))
```

```
##
##   Bachelors   Doctorate High School     Masters
##   0.6504065   0.1463415   0.0000000   0.2032520
```

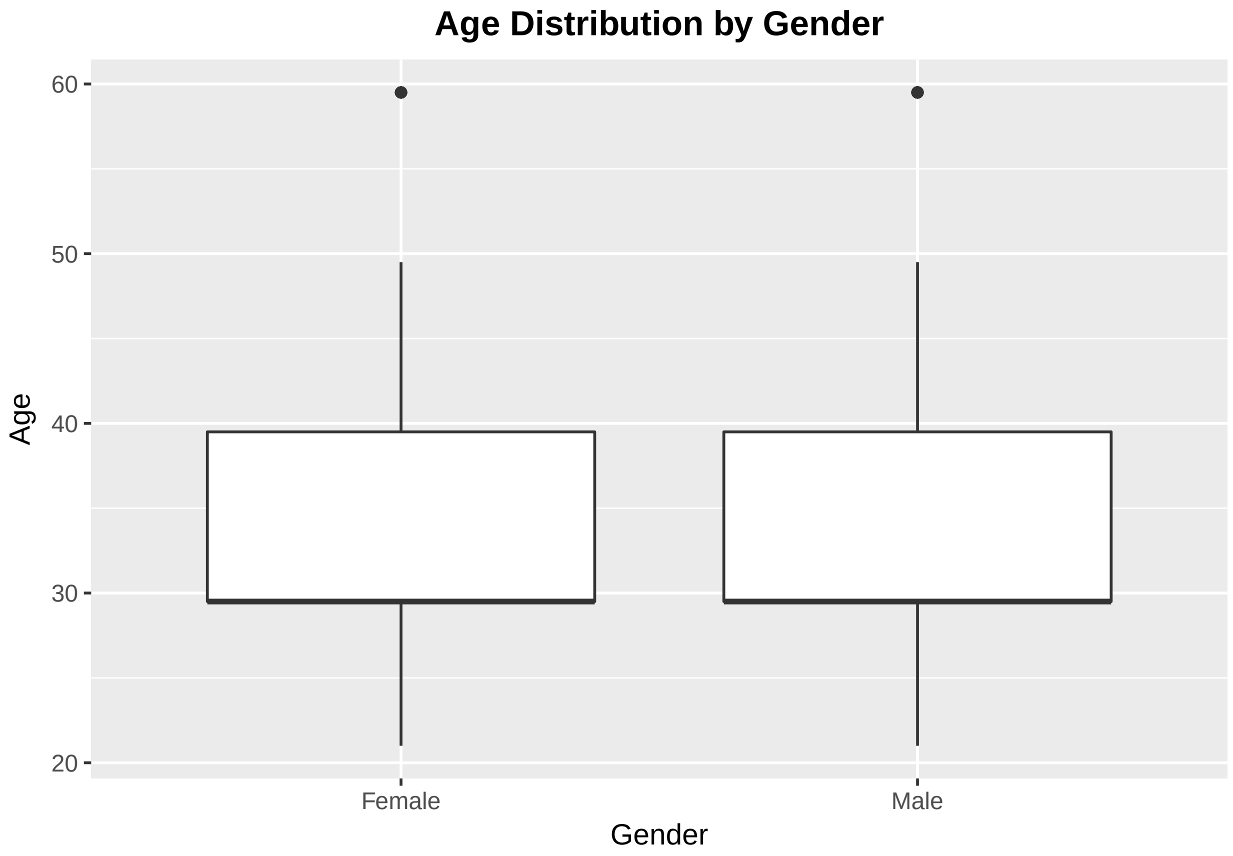
```
# Compare hours
prop.table(table(combined_m$hours))
```

```
##
##         0-3         3-6         6-9        9-12       12-15       15-18
## 0.005964215 0.071570577 0.145129225 0.262425447 0.220675944 0.135188867
##       18-21         21+
## 0.093439364 0.065606362
```

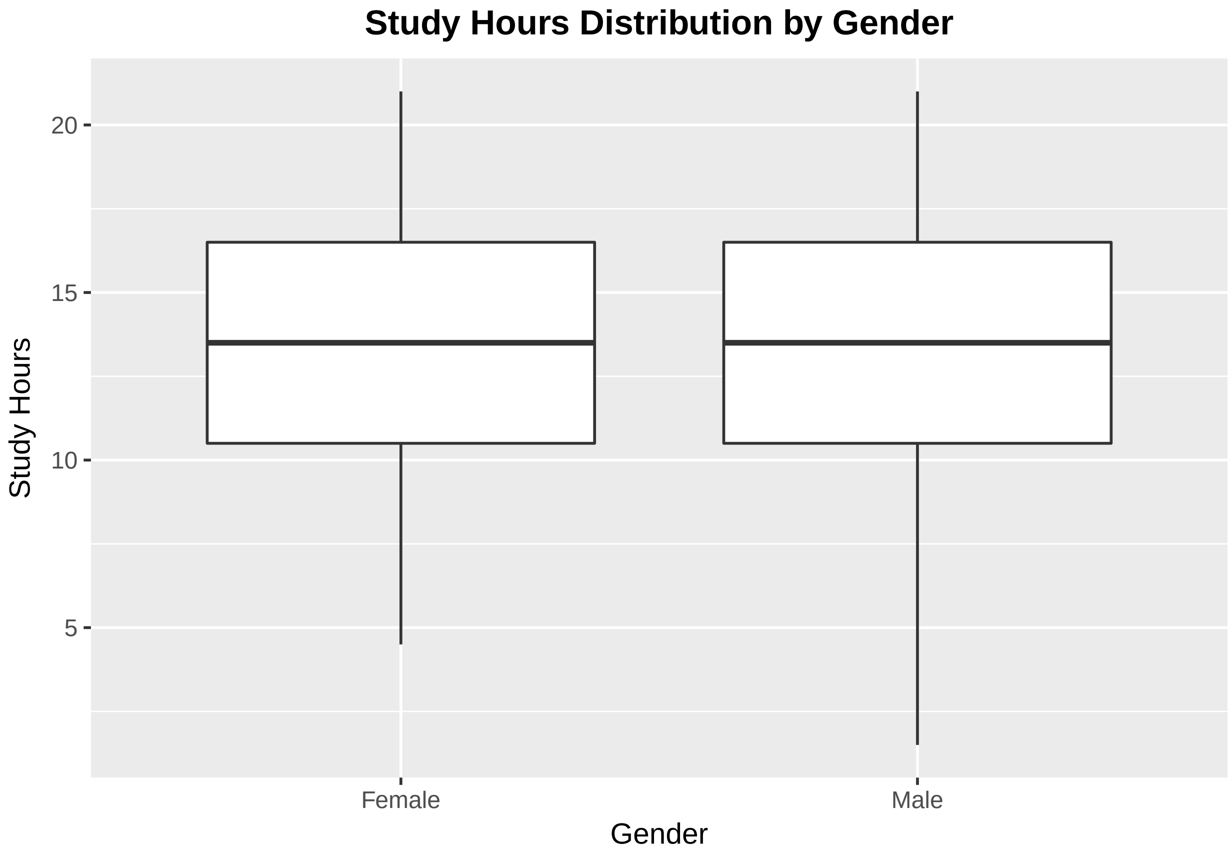
```
prop.table(table(combined_f$hours))
```

```
##
##        0-3        3-6        6-9       9-12      12-15      15-18
## 0.00000000 0.06315789 0.13684211 0.23157895 0.20000000 0.12631579
##      18-21        21+
## 0.12631579 0.11578947
```

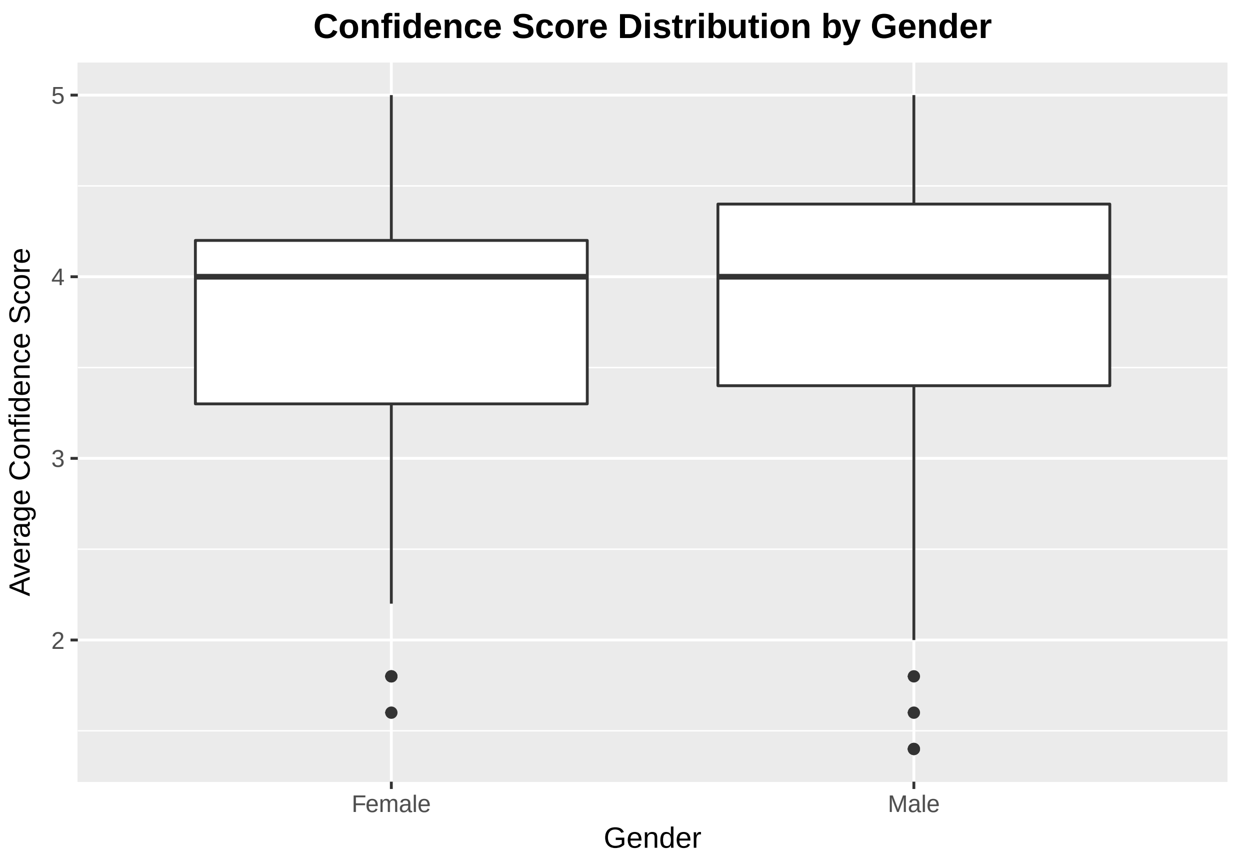
```
#Boxplot of age distribution by gender
ggplot(subset(combined, !is.na(gender)), aes(gender, age_num)) +
 geom_boxplot() +
 labs(title = "Age Distribution by Gender",
      x = "Gender", y = "Age") +
 theme(plot.title = element_text(lineheight=.8, face="bold", hjust=0.5))
```



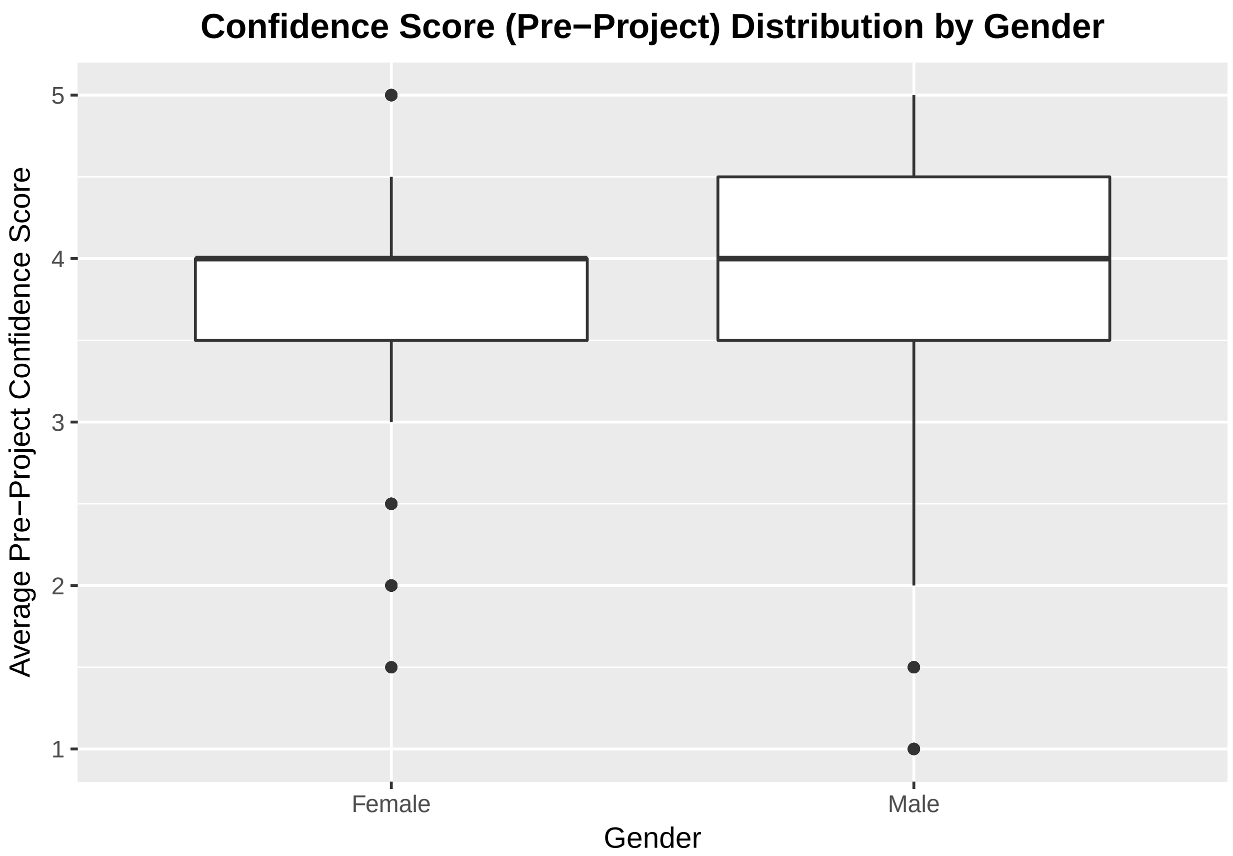
```r
# Boxplot of hours spent studying by gender
ggplot(subset(combined, !is.na(hours_num) & !is.na(gender)), aes(gender, hours_num)) +
  geom_boxplot() +
 labs(title = "Study Hours Distribution by Gender",
      x = "Gender", y = "Study Hours") +
 theme(plot.title = element_text(lineheight=.8, face="bold",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



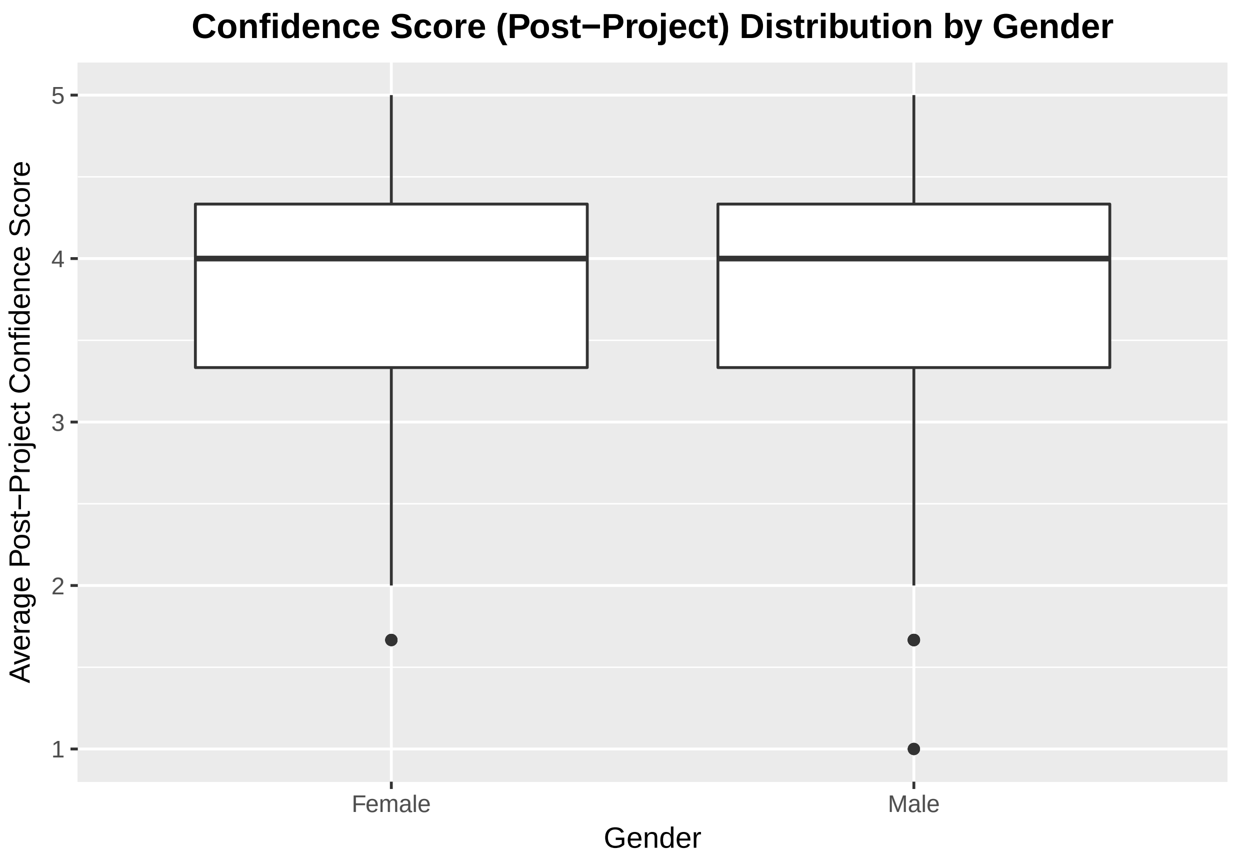
```r
# Boxplot of confidence score by gender
ggplot(subset(combined, !is.na(conf_ave) & !is.na(gender)), aes(gender, conf_ave)) +
  geom_boxplot() +
 labs(title = "Confidence Score Distribution by Gender",
      x = "Gender", y = "Average Confidence Score") +
 theme(plot.title = element_text(lineheight=.8,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



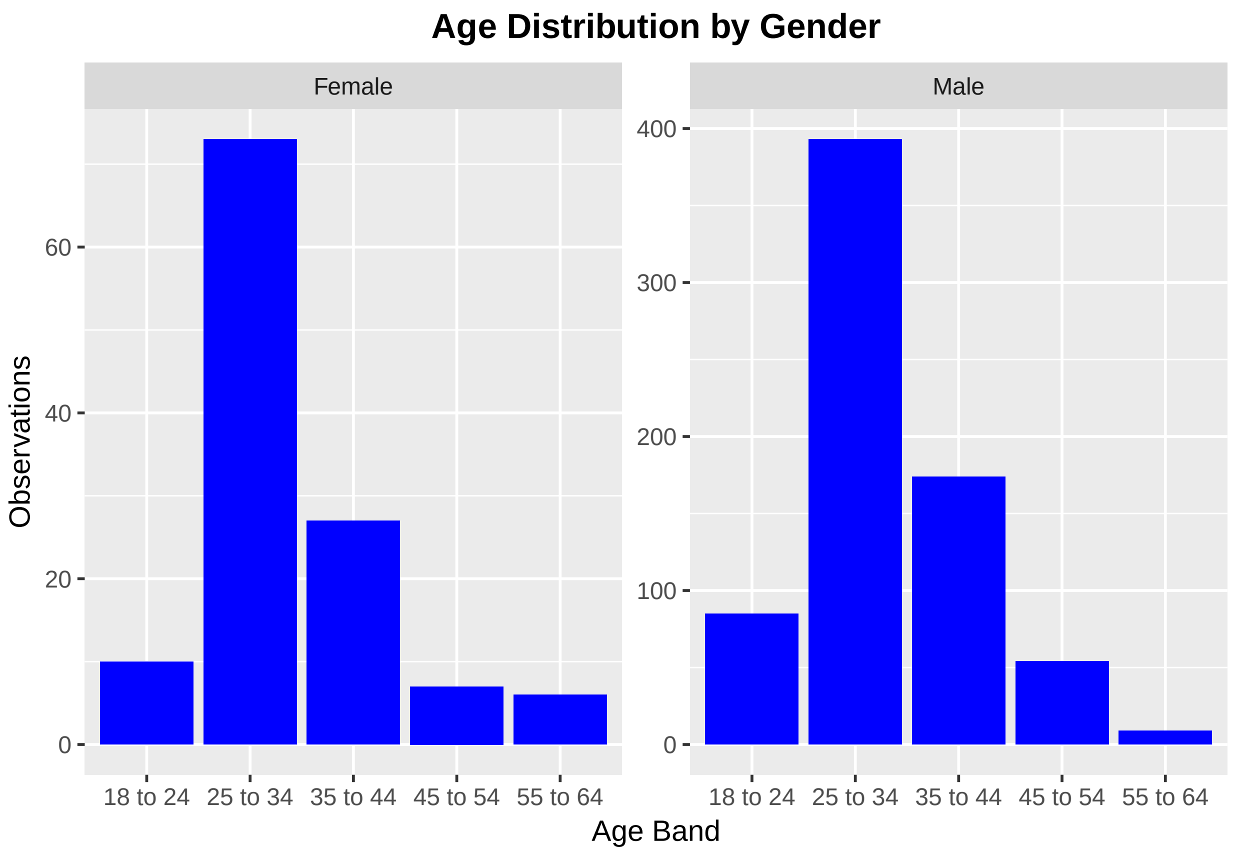
```
# Boxplot of confidence score (pre-project) by gender
ggplot(subset(combined, !is.na(conf_pre_ave) & !is.na(gender)), aes(gender,
    conf_pre_ave)) +  geom_boxplot() +
 labs(title = "Confidence Score (Pre-Project) Distribution by Gender",
      x = "Gender", y = "Average Pre-Project Confidence Score")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



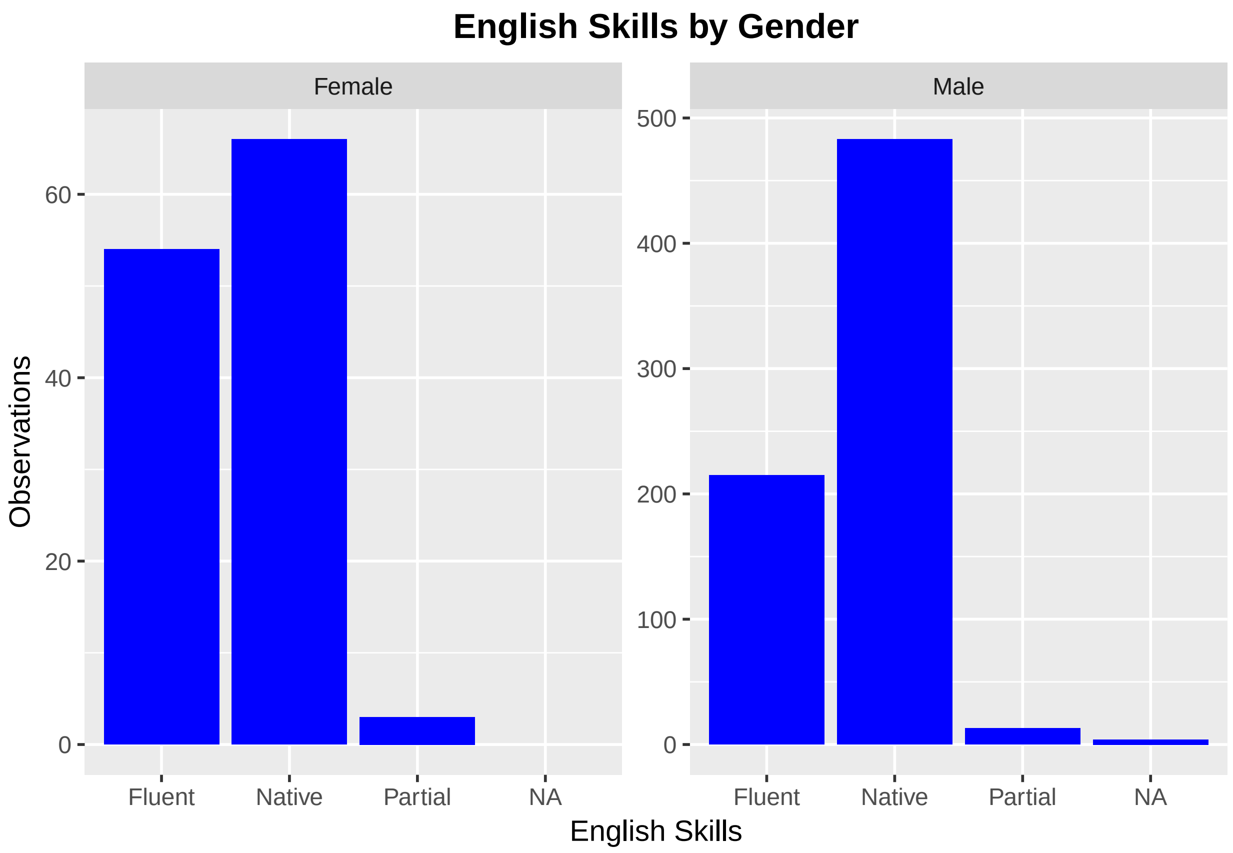
```r
# Boxplot of confidence score (post-project) by gender
ggplot(subset(combined, !is.na(conf_post_ave) & !is.na(gender)), aes(gender,
    conf_post_ave)) +  geom_boxplot() +
 labs(title = "Confidence Score (Post-Project) Distribution by Gender",
      x = "Gender", y = "Average Post-Project Confidence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



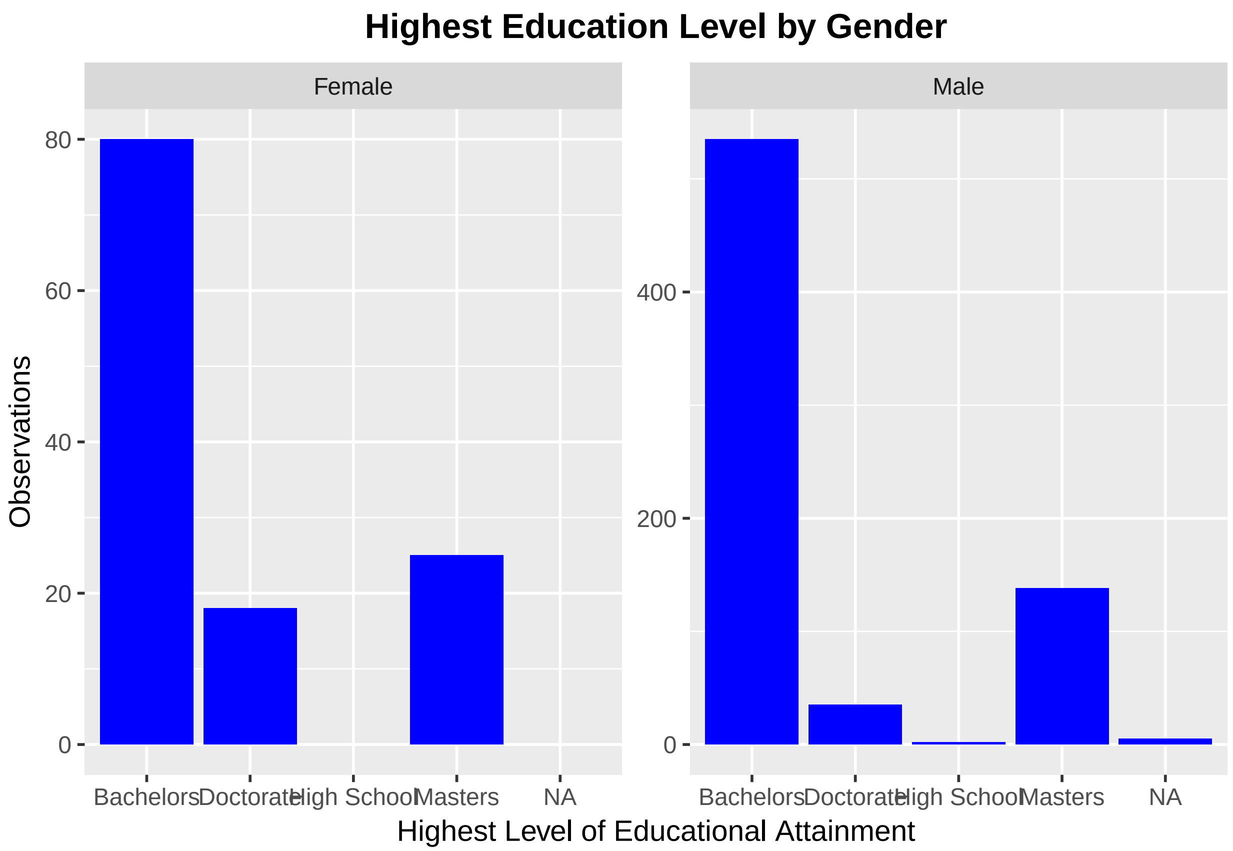
```r
# Bar chart comparing age by gender
ggplot(subset(combined, !is.na(gender)), aes(x = age)) +
    geom_bar(fill = "blue") +
    facet_wrap(~gender, scales = "free_y") +
    labs(title = "Age Distribution by Gender",
      x = "Age Band",
      y = "Observations") +
    theme(plot.title = element_text(lineheight=.8,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



```r
# Bar chart comparing English skills by gender
ggplot(subset(combined, !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))
```

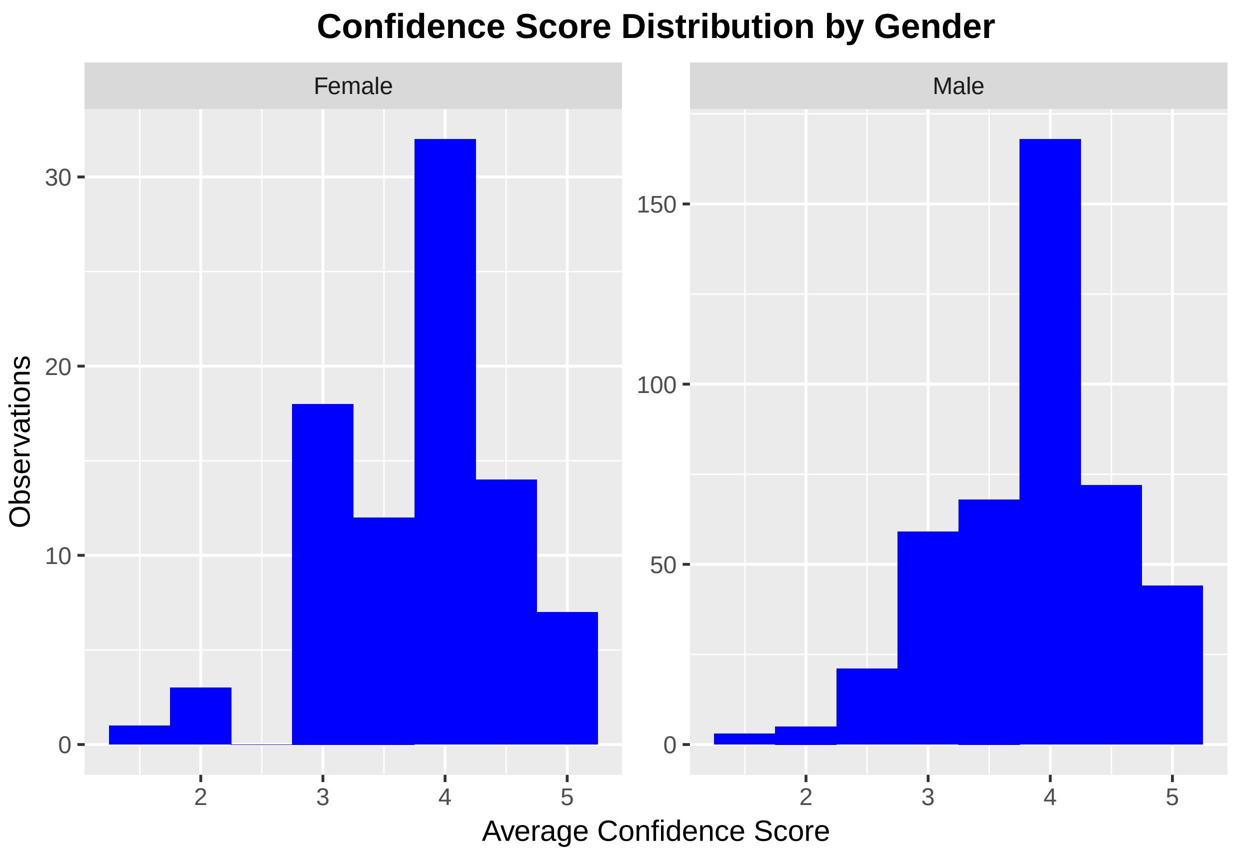


```r
# Bar chart comparing education by gender
ggplot(subset(combined, !is.na(gender)), aes(x = education)) +
    geom_bar(fill = "blue") +
    facet_wrap(~gender, scales = "free_y") +
    labs(title = "Highest Education Level by Gender",
      x = "Highest Level of Educational Attainment",
      y =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



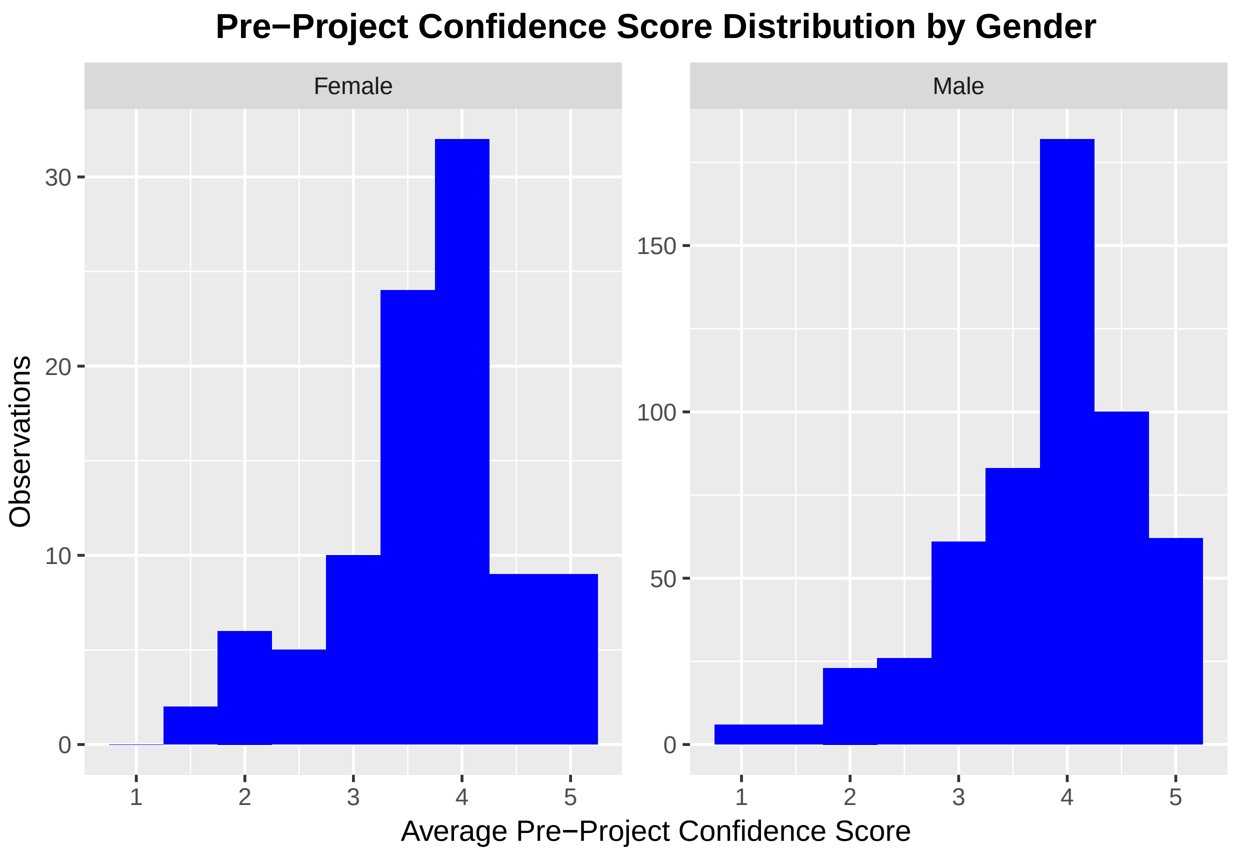
```
# Histogram of conf_ave by gender
ggplot(subset(combined, !is.na(gender)), aes(x = conf_ave)) +
    geom_histogram(fill = "blue", binwidth = 0.5) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Confidence Score Distribution by Gender",
      x = "Average Confidence Score",
      y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

```
## Warning: Removed 311 rows containing non-finite values (stat_bin).
```



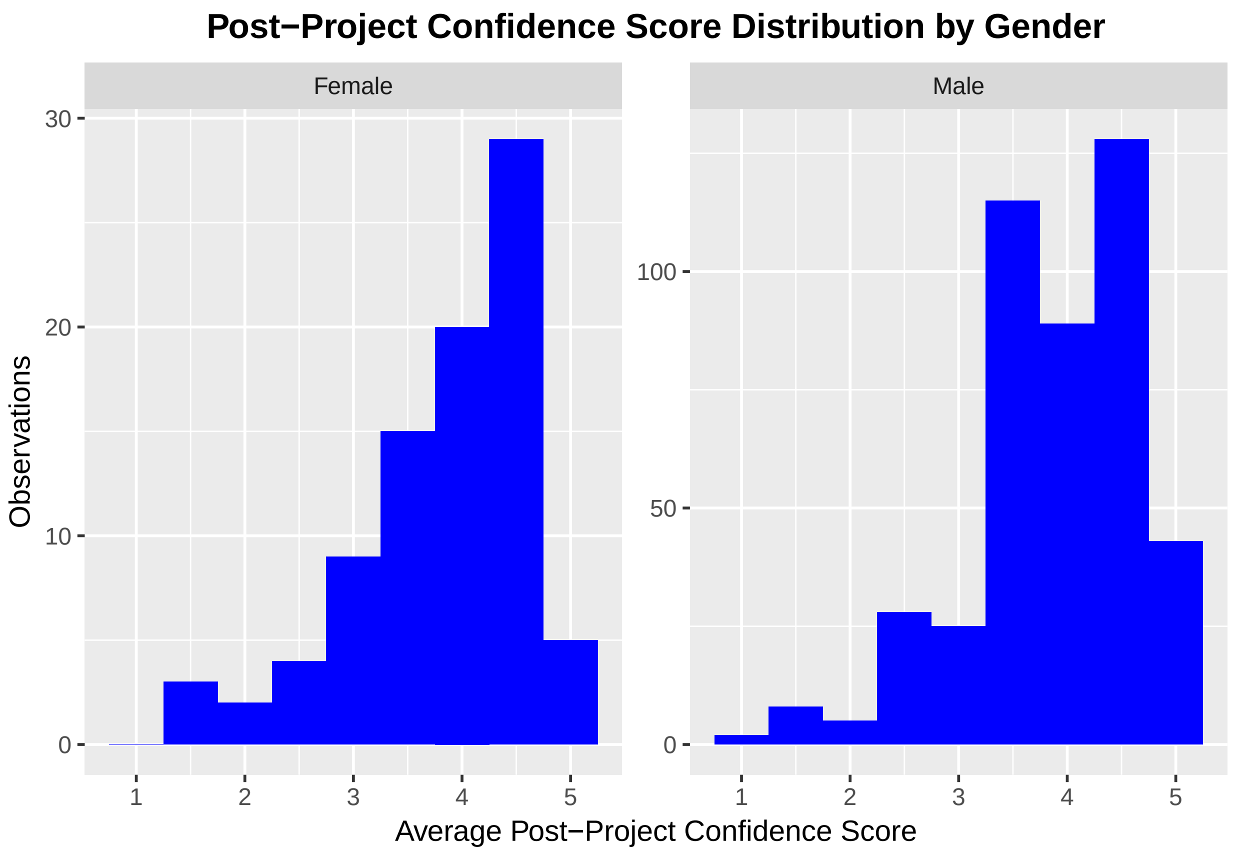
```r
# Histogram of conf_pre_ave by gender
ggplot(subset(combined, !is.na(gender)), aes(x = conf_pre_ave)) +
    geom_histogram(fill = "blue", binwidth = 0.5) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Pre-Project Confidence Score Distribution by Gender",
      x = "Average Pre-Project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

```
## Warning: Removed 192 rows containing non-finite values (stat_bin).
```



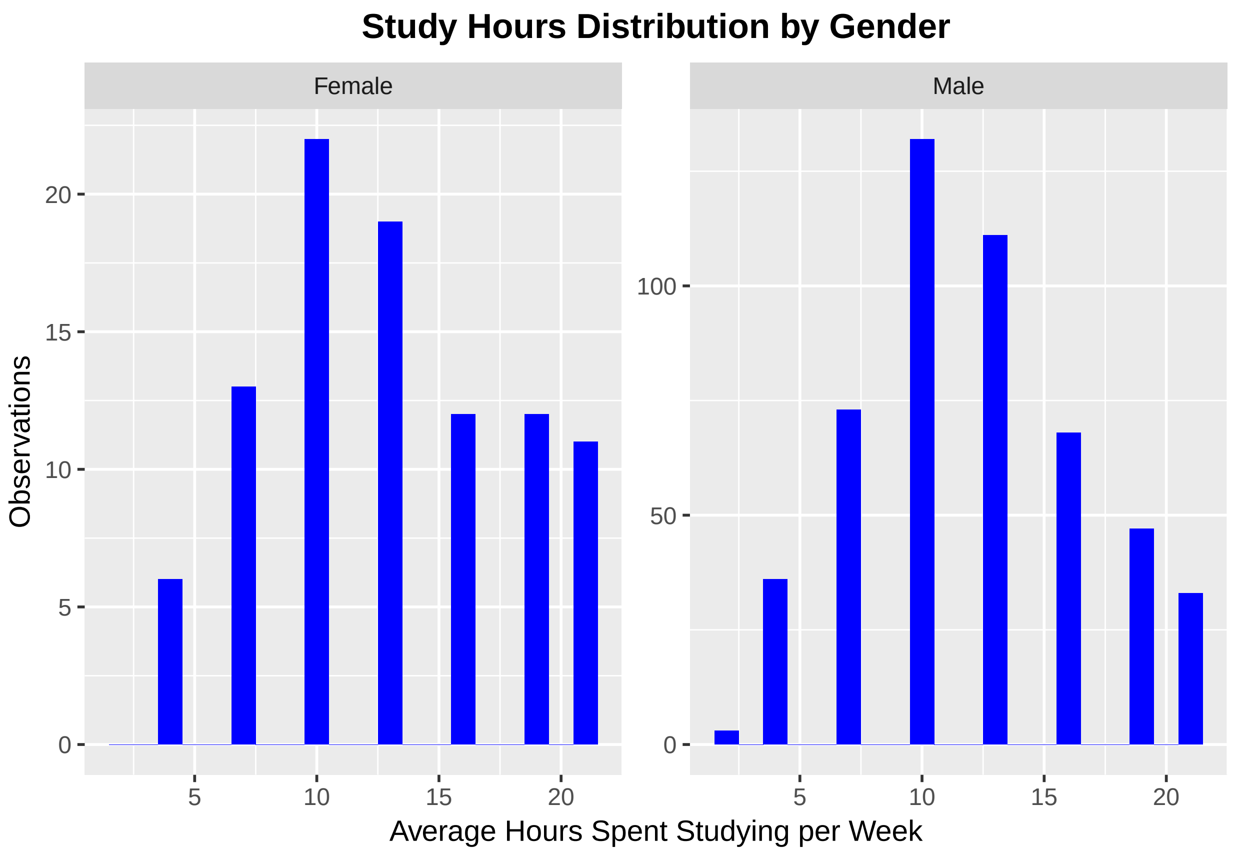
```
# Histogram of conf_post_ave by gender
ggplot(subset(combined, !is.na(gender)), aes(x = conf_post_ave)) +
    geom_histogram(fill = "blue", binwidth = 0.5) +
    facet_wrap(~gender, scale = "free_y") +
    labs(title = "Post-Project Confidence Score Distribution by Gender",
      x = "Average Post-Project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

```
## Warning: Removed 308 rows containing non-finite values (stat_bin).
```



```r
# Histogram of study hours by gender
ggplot(subset(combined, !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 240 rows containing non-finite values (stat_bin).
```



```
# Age tests
t.test(combined_m$age_num, combined_f$age_num)
```

```
##
##  Welch Two Sample t-test
##
## data:  combined_m$age_num and combined_f$age_num
## t = -0.93291, df = 157.96, p-value = 0.3523
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.4765707  0.8875662
## sample estimates:
## mean of x mean of y
##  32.81119  33.60569
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  age_num by gender
## W = 45246, p-value = 0.5691
## alternative hypothesis: true location shift is not equal to 0
```

```
# Higher ed tests
t.test(combined_m$higher_ind, combined_f$higher_ind)
```

```
##
```

```
##  Welch Two Sample t-test
##
## data:  combined_m$higher_ind and combined_f$higher_ind
## t = -2.3373, df = 157.44, p-value = 0.02068
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.19859167 -0.01667924
## sample estimates:
## mean of x mean of y
## 0.2419580 0.3495935
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  higher_ind by gender
## W = 48706, p-value = 0.01176
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Native speaker test
t.test(combined_m$native_ind, combined_f$native_ind)
```

```
##
##  Welch Two Sample t-test
##
## data:  combined_m$native_ind and combined_f$native_ind
## t = 2.9476, df = 160.87, p-value = 0.003679
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.04710786 0.23837120
## sample estimates:
## mean of x mean of y
## 0.6793249 0.5365854
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  native_ind by gender
## W = 37485, p-value = 0.002072
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Average confidence score tests
t.test(combined_m$conf_ave, combined_f$conf_ave)
```

```
##
##  Welch Two Sample t-test
##
## data:  combined_m$conf_ave and combined_f$conf_ave
## t = 0.95128, df = 119.45, p-value = 0.3434
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.08578545  0.24443748
## sample estimates:
## mean of x mean of y
##  3.858636  3.779310
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  conf_ave by gender
## W = 18074, p-value = 0.4092
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Average pre-project confidence score tests
t.test(combined_m$conf_pre_ave, combined_f$conf_pre_ave)
```

```
##
##  Welch Two Sample t-test
##
## data:  combined_m$conf_pre_ave and combined_f$conf_pre_ave
## t = 1.58, df = 134.86, p-value = 0.1164
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.03588629  0.32103455
## sample estimates:
## mean of x mean of y
##  3.802368  3.659794
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  conf_pre_ave by gender
## W = 23305, p-value = 0.04451
## alternative hypothesis: true location shift is not equal to 0
```

```r
# Average post-project confidence score tests
t.test(combined_m$conf_post_ave, combined_f$conf_post_ave)
```

```
##
##  Welch Two Sample t-test
##
## data:  combined_m$conf_post_ave and combined_f$conf_post_ave
## t = 0.624, df = 118.5, p-value = 0.5338
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1277819  0.2453710
## sample estimates:
## mean of x mean of y
##  3.886381  3.827586
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  conf_post_ave by gender
## W = 18786, p-value = 0.7077
## alternative hypothesis: true location shift is not equal to 0
```

```
# Study hours
t.test(combined_m$hours_num, combined_f$hours_num)
```

```
##
##  Welch Two Sample t-test
##
## data:  combined_m$hours_num and combined_f$hours_num
## t = -1.5226, df = 127.33, p-value = 0.1303
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.9237565  0.2506373
## sample estimates:
## mean of x mean of y
##  12.58449  13.42105
```

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

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  hours_num by gender
## W = 26092, p-value = 0.1474
## alternative hypothesis: true location shift is not equal to 0
```

```
# Check for multicollinearity
cor_subset = combined[, 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  0.01369383  0.20039288 -0.03582584
## native_ind   0.01369383  1.00000000 -0.18574516  0.10671440
## higher_ind   0.20039288 -0.18574516  1.00000000 -0.08601896
## gender_ind  -0.03582584  0.10671440 -0.08601896  1.00000000
```

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

summary(conf_lm)
```

```
##
## Call:
## lm(formula = conf_ave ~ gender + age_num + native_ind + higher_ind +
##     semester + course, data = na.omit(combined))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.38702 -0.40547  0.06085  0.45871  1.48032
##
## Coefficients: (1 not defined because of singularities)
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.276011   0.184114  23.225  < 2e-16 ***
## genderMale           0.122244   0.078065   1.566   0.1180
## age_num             -0.001823   0.003518  -0.518   0.6046
## native_ind          -0.072906   0.063975  -1.140   0.2550
## higher_ind           0.072196   0.067416   1.071   0.2847
## semesterSummer 2016 -0.547924   0.126438  -4.334 1.77e-05 ***
## semesterSpring 2016 -0.272650   0.145331  -1.876   0.0612 .
## semesterFall 2015   -0.077793   0.151992  -0.512   0.6090
## semesterSummer 2015 -0.629657   0.122669  -5.133 4.05e-07 ***
## courseHCI            0.069708   0.158930   0.439   0.6611
## courseEduTech              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6522 on 515 degrees of freedom
## Multiple R-squared:  0.1268, Adjusted R-squared:  0.1115
## F-statistic: 8.308 on 9 and 515 DF,  p-value: 1.34e-11
```

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

summary(conf_pre_lm)
```

```
##
## Call:
## lm(formula = conf_pre_ave ~ gender + age_num + native_ind + higher_ind +
##     semester + course, data = na.omit(combined))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.67544 -0.45721  0.02048  0.48904  1.65116
##
## Coefficients: (1 not defined because of singularities)
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.291345   0.216146  19.854  < 2e-16 ***
## genderMale           0.152983   0.091647   1.669   0.0957 .
## age_num             -0.002231   0.004130  -0.540   0.5894
## native_ind          -0.085034   0.075106  -1.132   0.2581
## higher_ind           0.058131   0.079145   0.734   0.4630
## semesterSummer 2016 -0.618053   0.148436  -4.164 3.67e-05 ***
## semesterSpring 2016 -0.189015   0.170615  -1.108   0.2684
## semesterFall 2015    0.003795   0.178435   0.021   0.9830
## semesterSummer 2015 -0.791671   0.144011  -5.497 6.08e-08 ***
## courseHCI            0.135545   0.186581   0.726   0.4679
## courseEduTech              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7657 on 515 degrees of freedom
## Multiple R-squared:  0.161,  Adjusted R-squared:  0.1463
## F-statistic: 10.98 on 9 and 515 DF,  p-value: 1.033e-15
```

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

summary(conf_post_lm)
```

```
##
## Call:
```

```
## lm(formula = conf_post_ave ~ gender + age_num + native_ind +
##     higher_ind + semester + course, data = na.omit(combined))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8206 -0.4212  0.1084  0.5644  1.3664
##
## Coefficients: (1 not defined because of singularities)
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.265789   0.214146  19.920  < 2e-16 ***
## genderMale           0.101751   0.090799   1.121 0.262972
## age_num             -0.001551   0.004092  -0.379 0.704847
## native_ind          -0.064821   0.074410  -0.871 0.384090
## higher_ind           0.081573   0.078413   1.040 0.298684
## semesterSummer 2016 -0.501172   0.147062  -3.408 0.000706 ***
## semesterSpring 2016 -0.328407   0.169036  -1.943 0.052583 .
## semesterFall 2015   -0.132185   0.176784  -0.748 0.454970
## semesterSummer 2015 -0.521648   0.142678  -3.656 0.000282 ***
## courseHCI            0.025817   0.184854   0.140 0.888981
## courseEduTech              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7586 on 515 degrees of freedom
## Multiple R-squared:  0.0626, Adjusted R-squared:  0.04622
## F-statistic: 3.821 on 9 and 515 DF,  p-value: 0.0001085
```

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

summary(hours_lm)
```

```
##
## Call:
## lm(formula = hours_num ~ gender + age_num + native_ind + higher_ind +
##     semester + course, data = na.omit(combined))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5684  -3.1733   0.0447   3.1881   9.1881
##
## Coefficients: (1 not defined because of singularities)
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           9.0189     1.2507   7.211 2.00e-12 ***
## genderMale           -0.8459     0.5303  -1.595  0.11132
## age_num               0.1091     0.0239   4.564 6.28e-06 ***
## native_ind           -0.6957     0.4346  -1.601  0.11002
## higher_ind            0.1313     0.4580   0.287  0.77443
## semesterSummer 2016   2.1934     0.8589   2.554  0.01095 *
## semesterSpring 2016  -0.3832     0.9873  -0.388  0.69806
## semesterFall 2015    -0.1171     1.0325  -0.113  0.90976
## semesterSummer 2015   2.2825     0.8333   2.739  0.00637 **
## courseHCI            -2.9420     1.0796  -2.725  0.00665 **
## courseEduTech             NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.43 on 515 degrees of freedom
## Multiple R-squared:  0.1373, Adjusted R-squared:  0.1222
## F-statistic: 9.105 on 9 and 515 DF,  p-value: 7.782e-13
```