```
################################################################################

# Brian Weinstein - bmw2148
# STAT W4201 001
# Homework 4
# 2016-02-24

# set working directory
setwd("~/Documents/advanced-data-analysis/homework_04")

# load packages
library(dplyr)
library(Sleuth3) # Data sets from Ramsey and Schafer's "Statistical Sleuth
(3rd ed)"
library(ggplot2); theme_set(theme_bw())
library(scales)
library(gmodels)
library(agricolae)



# Problem 1: Ramsey 5.23
#########################################################################

# load data
trexData <- Sleuth3::ex0523 %>%
  mutate(BoneNumber=as.factor(as.integer(gsub("Bone", "", Bone))))

# boxplots of oxygen composition in each bone
ggplot(trexData, aes(x=BoneNumber, y=Oxygen)) +
  geom_violin(alpha=0.15) +
  geom_boxplot() +
  geom_point(alpha=0.15) +
  labs(y="Oxygen Isotopic Composition\n(per mil deviations from SMOW)",
x="Bone Number")
ggsave(filename="writeup/1.png", width=6.125, height=3.5, units="in")

# use a one way ANOVA F-test
anovaTable <- anova(lm(Oxygen~Bone, data=trexData)); anovaTable

# compute the total sum of squares and the total degrees of freedom
sum(anovaTable$'Sum Sq')
sum(anovaTable$Df)

rm(list = ls()) # clear working environment



# Problem 2: Ramsey 5.25
#########################################################################

# load data
incomeEduData <- Sleuth3::ex0525 %>%
```

```
  mutate(LogIncome2005=log(Income2005))

# reorder Educ levels
incomeEduData$Educ <- relevel(incomeEduData$Educ, "<12")

# Part a ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###
###

# boxplots of income by education group
ggplot(incomeEduData, aes(x=Educ, y=(Income2005))) +
  geom_violin(alpha=0.15) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(y="Annual income in 2005, in U.S. dollars", x="Years of education",
title="Income vs Years of Education")
ggsave(filename="writeup/2a.png", width=7, height=4.5, units="in")

# check group sample sizes and standard deviations
incomeEduData %>%
  group_by(Educ) %>%
  summarize(numObs=n(), mean=mean(Income2005),
            median=median(Income2005), stdev=sd(Income2005))

# boxplots of LOG(income) by education group
ggplot(incomeEduData, aes(x=Educ, y=LogIncome2005)) +
  geom_violin(alpha=0.15) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(y="Annual income in 2005, in log(U.S. dollars)", x="Years of
education", title="Log Income vs Years of Education")
ggsave(filename="writeup/2b.png", width=7, height=4.5, units="in")

# check group sample sizes and standard deviations on log scale
incomeEduData %>%
  group_by(Educ) %>%
  summarize(numObs=n(), mean=mean(LogIncome2005),
            median=median(LogIncome2005), stdev=sd(LogIncome2005))

# use a one way ANOVA F-test on the log-transformed incomes
anovaTable <- anova(lm(LogIncome2005~Educ, data=incomeEduData)); anovaTable

# compute the total sum of squares and the total degrees of freedom
sum(anovaTable$'Sum Sq')
sum(anovaTable$Df)

# compute the limits for outlier definitions by group
logIncomeGroupSummaries <- incomeEduData %>%
  group_by(Educ) %>%
  summarize(pct25=quantile(LogIncome2005, probs=0.25, names=FALSE),
            pct50=median(LogIncome2005),
            pct75=quantile(LogIncome2005, probs=0.75, names=FALSE)) %>%
  mutate(iqr=(pct75-pct25),
         lowerBound=(pct25 - 1.5*iqr),
         upperBound=(pct75 + 1.5*iqr))
```

```
# create a dataset that excludes the outliers
incomeEduDataExclOutliers <- incomeEduData %>%
  left_join(x=., y=logIncomeGroupSummaries, by="Educ") %>%
  filter(LogIncome2005 >= lowerBound & LogIncome2005 <= upperBound)

# use a one way ANOVA F-test on the log-transformed incomes excluding outliers
anovaTableExclOutliers <- anova(lm(LogIncome2005~Educ,
data=incomeEduDataExclOutliers))
anovaTableExclOutliers

# Part b ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###
###

CompareTwoEducGroups <- function(data_frame=incomeEduData, Educ_groups){
  # define a function to perform a two sample t-test on log-transformed data
  # and returns a onse-sided pvalue, and estimate and confidence interval
  # on the back-transformed (anitlog) scale

  # Filter the dataset and relevel the Educ variable
  tempData <- filter(data_frame, Educ %in% Educ_groups) %>%
    mutate(Educ=relevel(factor(Educ), Educ_groups[1]))

  # Perform a two-sample t-test
  tt <- t.test(formula=LogIncome2005~Educ, data=tempData,
               var.equal=TRUE, conf.level=0.95,
               alternative="greater")

  # one-sided pvalue
  pval <- tt$p.value

  # take antilog of the estimate
  estimateOriginal <- exp(-diff(tt$estimate)[[1]])

  # Perform a two sided t-test for the confidence interval and take antilog
  confIntOriginal <- exp(t.test(formula=LogIncome2005~Educ, data=tempData,
                                var.equal=TRUE, conf.level=0.95,
                                alternative="two.sided")$conf.int)

  return(unlist(list(oneSidedPVal=pval, estimate=estimateOriginal,
                     confInt_lower=confIntOriginal[1],
                     confInt_upper=confIntOriginal[2])))

}

# Part b.i (>16 vs 16)
CompareTwoEducGroups(data_frame=incomeEduData, Educ_groups=c(">16", "16"))
CompareTwoEducGroups(data_frame=incomeEduDataExclOutliers,
Educ_groups=c(">16", "16"))

# boxplots of LOG(income) by education group
ggplot(incomeEduDataExclOutliers, aes(x=Educ, y=LogIncome2005)) +
  geom_violin(alpha=0.15) +
  geom_boxplot() +
  scale_y_continuous(labels = comma) +
  labs(y="Annual income in 2005, in log(U.S. dollars)", x="Years of
```

```
education", title="Log Income vs Years of Education, excluding outliers")
ggsave(filename="writeup/2c.png", width=7, height=4.5, units="in")

# Part b.ii (16 vs 13-15)
CompareTwoEducGroups(data_frame=incomeEduData, Educ_groups=c("16", "13-15"))

# Part b.iii (13-15 vs 12)
CompareTwoEducGroups(data_frame=incomeEduData, Educ_groups=c("13-15", "12"))

# Part b.iv (12 vs <12)
CompareTwoEducGroups(data_frame=incomeEduData, Educ_groups=c("12", "<12"))

rm(list = ls()) # clear working environment



# Problem 3: Ramsey 6.12
#########################################################################

# load data
handicapData <- Sleuth3::case0601

# check the order of Handicap factor levels
levels(handicapData$Handicap)

# test if the the avg of score means for
# amputee/crutches/wheelchair is equal to to hearing
fit.contrast(model=lm(Score ~ Handicap, data=handicapData),
             varname="Handicap", coeff=c(1/3, 1/3, -1, 0, 1/3),
             conf.int=0.95, df=TRUE)

rm(list = ls()) # clear working environment



# Problem 4: Ramsey 6.15
#########################################################################

# input data
testScoresData <- data.frame(group=c(1,2,3,4,5),
                             logo=c("L+D", "R", "R+L", "C", "C+L"),
                             method=c("Lecture and discussion",
                                      "Programmed text",
                                      "Programmed text with lectures",
                                      "Computer instruction",
                                      "Computer instruction with lectures"),
                             n=c(9, 9, 9, 9, 9),
                             average=c(30.20, 28.80, 26.20, 31.10, 30.20),
                             sd=c(3.82, 5.26, 4.66, 4.91, 3.53))

# compute the pooled standard deviation
sp <- sqrt(
  sum(((testScoresData$n) - 1) * (testScoresData$sd)^2) /
sum(((testScoresData$n) - 1))
)
```

```
sp

# estimate the linear contrast g
g <- sum(testScoresData$average[c(1, 4, 5)])/3 -
sum(testScoresData$average[c(2, 3)])/2
g

# compute standard error of the estimate of g
coefs <- c(1/3, -1/2, -1/2, 1/3, 1/3)
se <- sp * sqrt(sum(coefs^2 / testScoresData$n))
se

# compute 0.975 quantile of t distr with df=40
tquantile <- qt(p=0.975, df=40, lower.tail=TRUE); tquantile

# compute the 95% CI
g - tquantile * se
g + tquantile * se

rm(list = ls()) # clear working environment



# Problem 5: Ramsey 6.16
##########################################################################

# compute degrees of freedom for sp
df <- (6+6+6+6+6+6)-6; df

# Part a: multiplier for LSD
qt(p=(1-(0.05/2)), df=30)

# Part b: F-protected LSD

# no code needed

# Part c: multiplier for Tukey-Kramer
qtukey(p=(1-0.05), nmeans=6, df=30) / sqrt(2)

# Part d: multiplier for Bonferroni
qt(p=(1-(0.05/(2*15))), df=30)

# Part e: multiplier for Scheffe
sqrt(5 * qf(p=(1-0.05), df1=5, df2=30))

rm(list = ls()) # clear working environment



# Problem 6: Ramsey 6.23
##########################################################################

# load data
dietData <- Sleuth3::ex0623
```

```
# boxplots of weight loss in each group
ggplot(dietData, aes(x=Group, y=WtLoss24)) +
  geom_violin(alpha=0.15) +
  geom_boxplot() +
  labs(y="Weight Losses (kg)", x="Diet Group")
ggsave(filename="writeup/6.png", width=6.125, height=3.5, units="in")

# perform a one way ANOVA F-test to check for group differences
anovaTable <- anova(lm(WtLoss24~Group, data=dietData)); anovaTable

# compute the total sum of squares and the total degrees of freedom
sum(anovaTable$'Sum Sq')
sum(anovaTable$Df)

# tukey-kramer procedure
TukeyHSD(x=aov(lm(WtLoss24 ~ Group, data = dietData)), which="Group",
         ordered=TRUE, conf.level=0.95)

rm(list = ls()) # clear working environment
```