BM_HW3

Coco

10/6/2018

Problem 2

1

The first question is to assess if their body mass index (BMI) has changed 6 years after quitting smoking. Perform an appropriate hypothesis test and interpret your findings.

```
heavysmoke_df <- read_csv(file = "./data/HeavySmoke.csv")
```

```
## Parsed with column specification:
## cols(
##   ID = col_integer(),
##   BMI_base = col_double(),
##   BMI_6yrs = col_double()
## )
```

```
heavysmoke_df = janitor::clean_names(heavysmoke_df) %>%
  mutate(diff = bmi_6yrs-bmi_base) %>%
  mutate(mean = sum(diff)/length(bmi_6yrs)) %>%
  mutate(sd = (mean - diff)^2)

s_d_1= sqrt(sum((heavysmoke_df$sd))/(length(heavysmoke_df$id)-1))
t_statistics = 3.36 / (s_d_1/sqrt(9))
t_critical = qt(0.975,9)
t.test(heavysmoke_df$bmi_6yrs,heavysmoke_df$bmi_base,paired = T)
```

```
##
##  Paired t-test
##
## data:  heavysmoke_df$bmi_6yrs and heavysmoke_df$bmi_base
## t = 4.3145, df = 9, p-value = 0.001949
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.598291 5.121709
## sample estimates:
## mean of the differences
##                    3.36
```

2

The investigators suspected an overall change in weight over the years, so they decided toenroll a control group of 50-64 years of age that never smoked (data NeverSmoke.csv). Perform an appropriate test to compare the BMI changes between women that quit smoking and women who never smoked. Interpret the findings.

```
neversmoke_df <- read_csv(file = "./data/NeverSmoke.csv") %>%
  janitor::clean_names() %>%
```

```r
  mutate(diff = bmi_6yrs-bmi_base) %>%
  mutate(mean = sum(diff)/length(bmi_6yrs)) %>%
  mutate(sd = (mean - diff)^2)
```

```
## Parsed with column specification:
## cols(
##   ID = col_integer(),
##   BMI_base = col_double(),
##   BMI_6yrs = col_double()
## )
```

```r
s_d_2 = sqrt(sum((neversmoke_df$sd))/(length(neversmoke_df$id)-1))
s_sqr = (9*s_d_1^2+9*s_d_2^2)/(18)

t_stat = (3.36 - 1.55) / (sqrt(s_sqr)*sqrt(1/9+1/9))
t_crit = qt(0.975,18)

diff_heavy = heavysmoke_df$bmi_6yrs - heavysmoke_df$bmi_base
diff_never = neversmoke_df$bmi_6yrs - neversmoke_df$bmi_base

f_crit=qf(0.975,9,9)

var.test(diff_heavy,diff_never,alternative = "two.sided")
```

```
##
##  F test to compare two variances
##
## data:  diff_heavy and diff_never
## F = 1.1627, num df = 9, denom df = 9, p-value = 0.826
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.2888038 4.6811133
## sample estimates:
## ratio of variances
##           1.162722
```

```r
res = t.test(diff_heavy,diff_never,var.equal = FALSE, paired = FALSE)
names(res)
```

```
## [1] "statistic"   "parameter"   "p.value"     "conf.int"    "estimate"
## [6] "null.value"  "alternative" "method"      "data.name"
```

4

```r
 power.t.test(power = .90, delta = 3.0, sd=2.0, sig.level = 0.05, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 10.40147
##           delta = 3
##              sd = 2
##       sig.level = 0.05
##           power = 0.9
```

```
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
power.t.test(power = .80, delta = 3.0, sd=2.0, sig.level = 0.05, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 8.06031
##           delta = 3
##              sd = 2
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
power.t.test(power = .90, delta = 3.0, sd=2.0, sig.level = 0.025, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 12.38853
##           delta = 3
##              sd = 2
##       sig.level = 0.025
##           power = 0.9
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
power.t.test(power = .80, delta = 3.0, sd=2.0, sig.level = 0.025, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 9.821082
##           delta = 3
##              sd = 2
##       sig.level = 0.025
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
power.t.test(power = .90, delta = 1.7, sd=1.5, sig.level = 0.05, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 17.38011
##           delta = 1.7
##              sd = 1.5
##       sig.level = 0.05
##           power = 0.9
##     alternative = two.sided
```

```
##
## NOTE: n is number in *each* group
```

```r
    power.t.test(power = .80, delta = 1.7, sd=1.5, sig.level = 0.05, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 13.25404
##           delta = 1.7
##              sd = 1.5
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
    power.t.test(power = .90, delta = 1.7, sd=1.5, sig.level = 0.025, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 20.64058
##           delta = 1.7
##              sd = 1.5
##       sig.level = 0.025
##           power = 0.9
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
    power.t.test(power = .80, delta = 1.7, sd=1.5, sig.level = 0.025, alternative = c("two.sided"))
```

```
##
##      Two-sample t test power calculation
##
##               n = 16.12566
##           delta = 1.7
##              sd = 1.5
##       sig.level = 0.025
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

Problem 3

```r
knee_df = read_csv(file = "./data/Knee.csv") %>%
  janitor::clean_names()
```

```
## Parsed with column specification:
## cols(
##   Below = col_integer(),
##   Average = col_integer(),
##   Above = col_integer()
```

```
## )
```

```r
summary(knee_df$below)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##      29      36      40      38      42      43       2
```

```r
summary(knee_df$average)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   28.00   30.25   32.00   33.00   35.00   39.00
```

```r
summary(knee_df$above)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   20.00   21.00   22.00   23.57   24.50   32.00       3
```

```r
knee_df = gather(knee_df, key = type, value = value, below:above) %>%
  filter(!is.na(value))
knee_df$type=as_factor(knee_df$type)

res <- aov(value~type,data = knee_df)
summary(res)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)    
## type         2  795.2   397.6   19.28 1.45e-05 ***
## Residuals   22  453.7    20.6                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
res
```

```
## Call:
##    aov(formula = value ~ type, data = knee_df)
## 
## Terms:
##                     type Residuals
## Sum of Squares  795.2457  453.7143
## Deg. of Freedom        2        22
## 
## Residual standard error: 4.541297
## Estimated effects may be unbalanced
```

```r
qf(0.01, df1 = 2, df2 = 22)
```

```
## [1] 0.01005493
```

perform pairwise comparisons with the appropriate adjustments (Bonferroni, Tukey, and Dunnett – 'below average' as reference)

```r
pairwise.t.test(knee_df$value,knee_df$type,p.adjust.method = 'bonferroni')
```

```
## 
##  Pairwise comparisons using t tests with pooled SD 
## 
## data:  knee_df$value and knee_df$type 
## 
##         below   average
## average 0.0898  -      
## above   1.1e-05 0.0011
```

```
##
## P value adjustment method: bonferroni
```

```r
TukeyHSD(res)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ type, data = knee_df)
##
## $type
##                     diff       lwr        upr     p adj
## average-below  -5.000000 -10.41130  0.4113011 0.0736833
## above-below   -14.428571 -20.33278 -8.5243579 0.0000102
## above-average  -9.428571 -15.05051 -3.8066356 0.0010053
```

```r
library(multcomp)
```

```
## Loading required package: mvtnorm
## Loading required package: survival
## Loading required package: TH.data
## Loading required package: MASS
##
## Attaching package: 'MASS'
## The following object is masked from 'package:dplyr':
##
##     select
##
## Attaching package: 'TH.data'
## The following object is masked from 'package:MASS':
##
##     geyser
```

```r
dunnetttest<-glht(res, linfct=mcp(type="Dunnett"))
summary(dunnetttest)
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Dunnett Contrasts
##
##
## Fit: aov(formula = value ~ type, data = knee_df)
##
## Linear Hypotheses:
##                      Estimate Std. Error t value Pr(>|t|)
## average - below == 0   -5.000      2.154  -2.321   0.0543 .
## above - below == 0    -14.429      2.350  -6.139 6.93e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```

Problem 4

```r
library(datasets)
data("UCBAdmissions")

ucb_df <- as.data.frame(UCBAdmissions) %>%
  janitor::clean_names()

prop_men = sum(filter(ucb_df,gender=="Male", admit=="Admitted")$freq)/sum(filter(ucb_df,gender=="Male")$

prop_women = sum(filter(ucb_df,gender=="Female", admit=="Admitted")$freq)/sum(filter(ucb_df,gender=="Fe
```