Homework1

Rick Galbo, Bills Fan, Not #BillsMafia member

January 24, 2016

Problem 1

a) Upon initial inspection, we see that the median test score is 77 which is much higher than the previous median of 70.

```
testScores <- c(79,74,88,80,80,66,65,86,84,80,78,72,71,74,86,96,77,
                81,76,80,76,75,78,87,87,74,85,84,76,77,76,74,85,74,76,
                77,76,74,81,76)

testScores_med <- median(testScores)

#summarize data
summary(testScores)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   65.00   74.75   77.00   78.53   81.75   96.00
```

b) The null and alternative hypothesis at alpha level equal to 0.05 for single sided binomial test.
Ho: median = 70
Ha: median > 70

```
diff <- testScores - 70
table(sign(diff))
```

```
##
## -1  1
##  2 38
```

```
#by hand
1 - pbinom(37,40,0.5)
```

```
## [1] 7.466952e-10
```

```
#using binomial function
binom.test(38, 40, alternative = 'greater')
```

```
##
##  Exact binomial test
##
## data:  38 and 40
## number of successes = 38, number of trials = 40, p-value =
## 7.467e-10
## alternative hypothesis: true probability of success is greater than 0.5
## 95 percent confidence interval:
```

```
##  0.850848 1.000000
## sample estimates:
## probability of success
##                   0.95
```

c) The binomial hypothesis test done here both "by hand" and using the binomial test function

d) We can see that both p-values agree with the value 7.466952e-10. Both support the rejection of the null hypothesis at the $a = 0.05$ level meaning there is evidence the median actually did increase.

Problem 2

Start by creating a data frame with a numerical score variable and a treatment categorical variable. Then use the `tapply` function to calculate the medians of the different groups and see they're different.

```
dat <- data.frame(score = c(3,4,6,3,1,2,1,1),
                  treatment = c(rep('T1',4),rep('T2',4)))

scoreMean <- tapply(dat$scor, dat$treatment, mean)
scoreMean
```

```
##   T1   T2
## 4.00 1.25
```

```
obsDiffMean<-diff(scoreMean)
```

To compute a p-value to check if the groups are significantly different, using the perm library's function `permTS` to check if the median values of all possible permutations of the labeling are the same.

```
#by hand permutation test
library(gtools)
perms<-combinations(8,4)
dim(perms)
```

```
## [1] 70  4
```

```
permDiffs<-rep(NA,dim(perms)[1])

for (i in 1:dim(perms)[1]){
  scoresTemp<-c(dat$score[perms[i,]],dat$score[setdiff(c(1:8),perms[i,])] )
  datPermuted<-data.frame(score=scoresTemp,treatment=dat$treatment)
  permDiffs[i]<-diff(tapply(datPermuted$score,datPermuted$treatment,mean))
}
sort(permDiffs)
```

```
##  [1] -2.75 -2.25 -2.25 -1.75 -1.75 -1.75 -1.75 -1.75 -1.75 -1.75 -1.25
## [12] -1.25 -1.25 -1.25 -1.25 -0.75 -0.75 -0.75 -0.75 -0.75 -0.75
## [23] -0.75 -0.75 -0.75 -0.75 -0.25 -0.25 -0.25 -0.25 -0.25 -0.25 -0.25
## [34] -0.25 -0.25  0.25  0.25  0.25  0.25  0.25  0.25  0.25  0.25  0.25
## [45]  0.75  0.75  0.75  0.75  0.75  0.75  0.75  0.75  0.75  0.75  1.25
## [56]  1.25  1.25  1.25  1.25  1.25  1.75  1.75  1.75  1.75  1.75  1.75
## [67]  1.75  2.25  2.25  2.75
```

```r
#calc p-val
sum(permDiffs<=obsDiffMean)/length(permDiffs)
```

```
## [1] 0.01428571
```

```r
#prepackaged permutation test
library(perm)
permTS(dat$score~dat$treatment, alternative='greater',exact=TRUE)
```

```
##
##  Exact Permutation Test (network algorithm)
##
## data:  dat$score by dat$treatment
## p-value = 0.01429
## alternative hypothesis: true mean dat$treatment=T1 - mean dat$treatment=T2 is greater than 0
## sample estimates:
## mean dat$treatment=T1 - mean dat$treatment=T2
##                                          2.75
```

Can see from the small p-values that this test is significant but the p-value for this specific situation doesn't require a sample data set or any computation to find. If the groups are perfectly separated by the size of their elements, meaning that all elements in group one are larger than all elements in group two, this creates a maximal condition. All other permutations of the data will not produce as extreme of a result. The one sided p-value of this occurring:

$$\frac{1}{\binom{\#of observations}{groupsize}}$$

Problem 3

```r
siblings<-data.frame(hometown=c(rep("rural",24),rep("urban",17)),
                     siblings=c(3,2,1,1,2,1,3,2,2,2,2,5,1,4,1,1,1,1,6,2,2,
                                2,1,1,1,0,1,1,0,0,1,1,1,8,1,1,1,0,1,1,2))

#Wilcoxon Rank Sum
siblings$rank<-rank(siblings$siblings)
W1<-sum(siblings$rank[siblings$hometown=='rural'])
W1
```

```
## [1] 614.5
```

```r
set.seed(1234)
nsims<-10000
rankSumPerms<-rep(NA,nsims)

for (i in 1:nsims){
  rankSumPerms[i]<- sum(sample(1:41,24,replace=FALSE))
}

#pvalue
sum(rankSumPerms>=W1)/nsims
```

```
## [1] 0.0014
```

a) Checking for difference between groups:
Ho: The sums of the ranks between rural and urban areas is the same.
Ha: The sums of the ranks between rural and urban areas is not the same.
Here we can see that the p-value of 0.0014 is significant at $a = 0.05$ level which allows us to reject the null hypothesis.

b) It would not be efficient to attempt a complete permutation test because the number of possible permutations on the data are $\binom{41}{24} = 151,584,480,450$. Instead it is acceptable to do a simulation for this test.

Problem 4

Create a data set with two groups of similar values and one that has a large outlier.

```
prob4 <- data.frame(group=c(rep('uno',4),rep('dos',4)),value=c(1:7,45))

wilcox.test(value~group,data = prob4)
```

```
##
##  Wilcoxon rank sum test
##
## data:  value by group
## W = 16, p-value = 0.02857
## alternative hypothesis: true location shift is not equal to 0
```

```
t.test(value~group, data=prob4)
```

```
##
##  Welch Two Sample t-test
##
## data:  value by group
## t = 1.3548, df = 3.026, p-value = 0.2677
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -17.72172  44.22172
## sample estimates:
## mean in group dos mean in group uno
##             15.75              2.50
```

Can see that the outlier in the data set threw off the two sample t-test while the Wilcoxon rank-test still had significance.

Problem 5

```
#Treatment 1 data
trt1 <- c(21.9,20.2,19.4,20.3,19.6,20.4,18.4,20.1,22.0,18.9)
#Treatment 2 data
trt2 <- c(20.2,13.8,21.8,19.2,19.6,25.5,17.0,17.6,19.5,22.2)

ansari.test(trt1,trt2)
```

```
## Warning in ansari.test.default(trt1, trt2): cannot compute exact p-value
## with ties


##
##  Ansari-Bradley test
##
## data:  trt1 and trt2
## AB = 64, p-value = 0.1707
## alternative hypothesis: true ratio of scales is not equal to 1
```

Ho: The variance of the two samples is the same.
Ha: The sample variance is not the same.

This test doesn't reject the null hypothesis at the standard alpha level.