Math 159, Spring 2010
Jo Hardin
HW #8
due Tuesday 4/13/10

Section 4.1: Exercises 3, 5 (Do not turn in: answers in back of book)

Section 5.11:
Exercises: 2 & 4 ; Problem 1 (note that you don't have to find the entire dist to do the problems)

Section 16.5: 52 (pg 16-49), 55, 64, 67

- Any data you need from the supplement can be found at `http://bcs.whfreeman.com/ips6e/`. Click on data sets. You need to save the files (the ones you want to use, not all of them) into the same folder where R lives. This might be called "Statistics" on your user space. To input a file into R, for example:

```
price2<-read.table("ex16_055.txt",header=T,sep="\t")
```

- The R code for 3 different permutation tests is given below (2x2 table, rxc table, difference in variances):

```
# Fisher's Exact Test (2x2 tables)

thresh<-c(rep("success",11),rep("failure",13))
group<-c(rep("A",3),rep("B",8),rep("A",9),rep("B",4))

table(thresh,group)
obs.stat<-table(thresh,group)[2,1]

perm.stat<-c()  # creating a vector to hold the test statistic values


for(i in 1:999){
perm.stat<-c(perm.stat,table(thresh,sample(group,24,replace=F))[2,1])
}

hist(perm.stat)
(sum(perm.stat<=obs.stat)+1)/1000
```

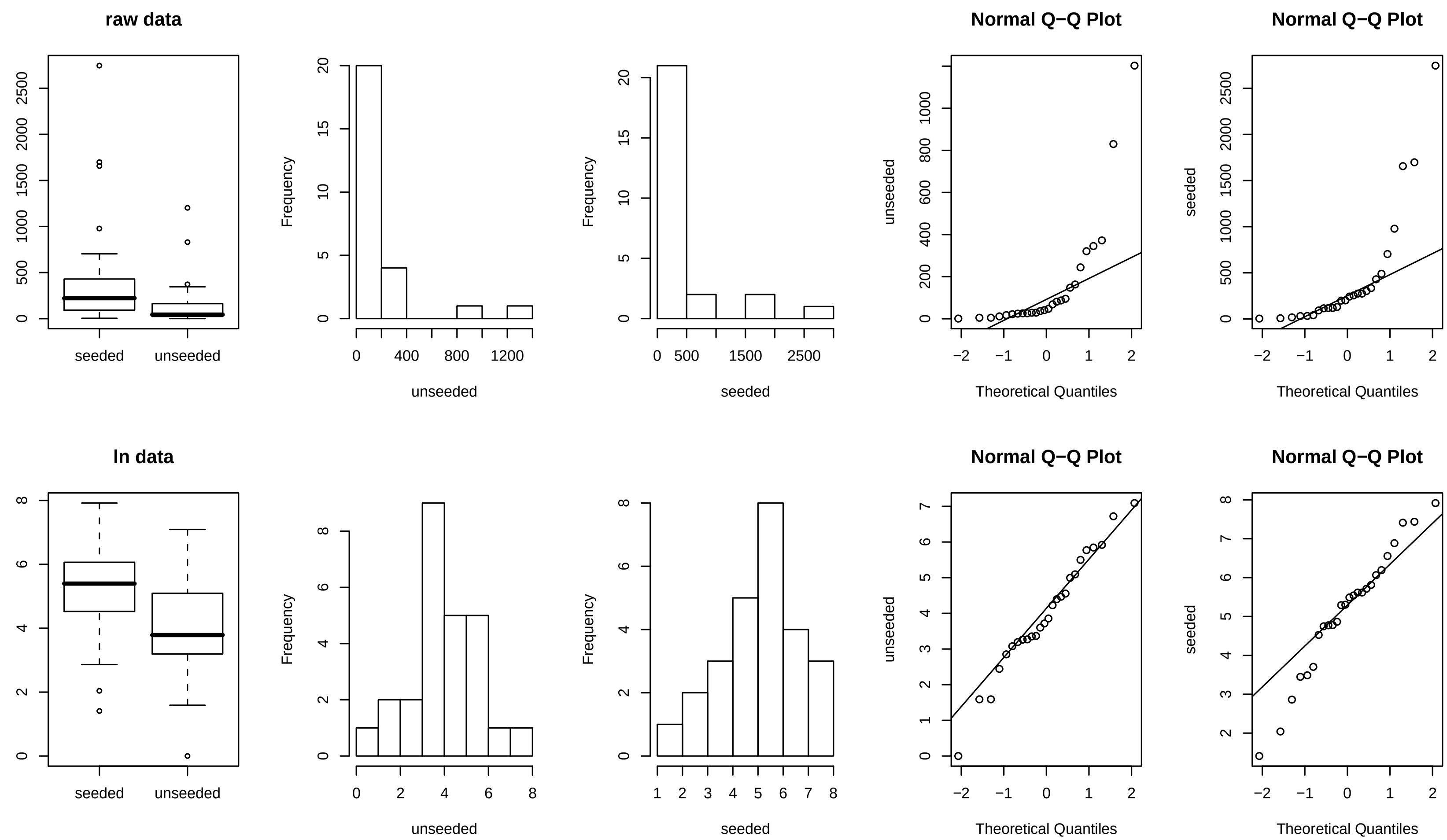
```
# rxc tables  (grades given for 2 different profs)

prof<-c(rep("Schafer",125),rep("Moore",107),rep("Rossman",92))
grade<-c(rep("A",12),rep("B",45),rep("C",49),rep("D",6),rep("F", 13),
 rep("A",10),rep("B",32),rep("C",43),rep("D",18),rep("F", 4),
 rep("A",15),rep("B",19),rep("C",32),rep("D",20),rep("F", 6))


table(prof,grade)
obs.stat<-chisq.test(table(prof,grade))$stat

perm.stat<-c()  # creating a vector to hold the test statistic values


for(i in 1:999){
perm.stat<-c(perm.stat,chisq.test(table(prof,sample(grade,324,replace=F)))$stat)
}

hist(perm.stat)
(sum(perm.stat<=obs.stat)+1)/1000


# Difference in Variances (exercise 16.71):

bp<-read.table("ta16_007.txt",header=T,sep="\t")
par(mfrow=c(2,2))

hist(bp[1:10,6])
hist(bp[11:21,6])
qqnorm(bp[1:10,6])
qqnorm(bp[11:21,6])

obs.stat<-var(bp[1:10,6])/var(bp[11:21,6])

perm.stat<-c()

for(i in 1:999){
bp2<-sample(bp[,6],21,replace=F)
perm.stat<-c(perm.stat,var(bp2[1:10])/var(bp2[11:21]))
}

hist(perm.stat)
(sum(perm.stat>=obs.stat) + 1)/1000
```

- In class example. Cloud seeding data: seeding or not seeding was randomly allocated to 52 days when seeding was appropriate. The pilot did not know whether or not the plane was seeding. Rain is measured in acre-feet.

  After running tests to compare means and variances we obtain the following p-values:

	comparison of means		comparison of variances	
	Permutation	t-test	Permutation	F-test
Raw Data	0.031	0.054	0.068	0.000067
Logged Data	0.010	0.014	0.535	0.897

```
par(mfrow=c(2,5))
boxplot(cloud[,1]~cloud[,2],main="raw data")
hist(cloud[1:26,1],xlab="unseeded",main="")
hist(cloud[27:52,1],xlab="seeded",main="")
qqnorm(cloud[1:26,1],ylab="unseeded")
qqline(cloud[1:26,1])
qqnorm(cloud[27:52,1],ylab="seeded")
qqline(cloud[27:52,1])

#unlogged data:

obs.um.stat<-tapply(cloud[,1],cloud[,2],mean)[1] -
tapply(cloud[,1],cloud[,2],mean)[2]
perm.um.stat<-c()

for(i in 1:999){
cloud2<-sample(cloud[,1],52,replace=F)
 perm.um.stat<-c(perm.um.stat,tapply(cloud2,cloud[,2],mean)[1] -
tapply(cloud2,cloud[,2],mean)[2])
}

obs.uv.stat<-tapply(cloud[,1],cloud[,2],var)[1] /
tapply(cloud[,1],cloud[,2],var)[2]
perm.uv.stat<-c()

for(i in 1:999){
cloud2<-sample(cloud[,1],52,replace=F)
 perm.uv.stat<-c(perm.uv.stat,tapply(cloud2,cloud[,2],var)[1] /
  tapply(cloud2,cloud[,2],var)[2])
}

(sum(obs.um.stat<=perm.um.stat)+1)/1000
(sum(obs.uv.stat<=perm.uv.stat)+1)/1000
t.test(cloud[,1]~cloud[,2])
var.test(cloud[,1]~cloud[,2])
```


raw data
seeded
unseeded
Frequency
unseeded
Frequency
seeded
Normal Q−Q Plot
unseeded
Theoretical Quantiles
Normal Q−Q Plot
seeded
Theoretical Quantiles
ln data
seeded
unseeded
Frequency
unseeded
Frequency
seeded
Normal Q−Q Plot
unseeded
Theoretical Quantiles
Normal Q−Q Plot
seeded
Theoretical Quantiles

- To find a CI for the difference in locations of the distributions of the seeded vs. unseeded rainfall, we shift the unseeded values by some amount, and do our usual hypothesis test. Our CI will be those values we don't reject for a given α. Here:

	b test value								
	20	50	100	300	400	500	550	570	600
p	0.03	0.05	0.11	0.56	0.80	0.94	0.98	0.98	0.99

  where p is the proportion of simulated test statistics greater than or equal to our observed value (after shifting by an amount "b").

  R code:

```
for(b in c(0,20,50,100,300,400,500,550,570,600)){

cloud.ci<-cbind(rainfall=c(cloud[,1]+c(rep(b,26),rep(0,26))),cloud[,2])

obs.um.ci.stat<-tapply(cloud.ci[,1],cloud.ci[,2],mean)[2] -
tapply(cloud.ci[,1],cloud.ci[,2],mean)[1]

perm.um.ci.stat<-c()

for(i in 1:3999){
 cloud.ci2<-sample(cloud.ci[,1],52,replace=F)
 perm.um.ci.stat<-c(perm.um.ci.stat,tapply(cloud.ci2,cloud[,2],mean)[1] -
tapply(cloud.ci2,cloud[,2],mean)[2])
}

print((sum(obs.um.ci.stat>=perm.um.ci.stat)+1)/4000)
}
```