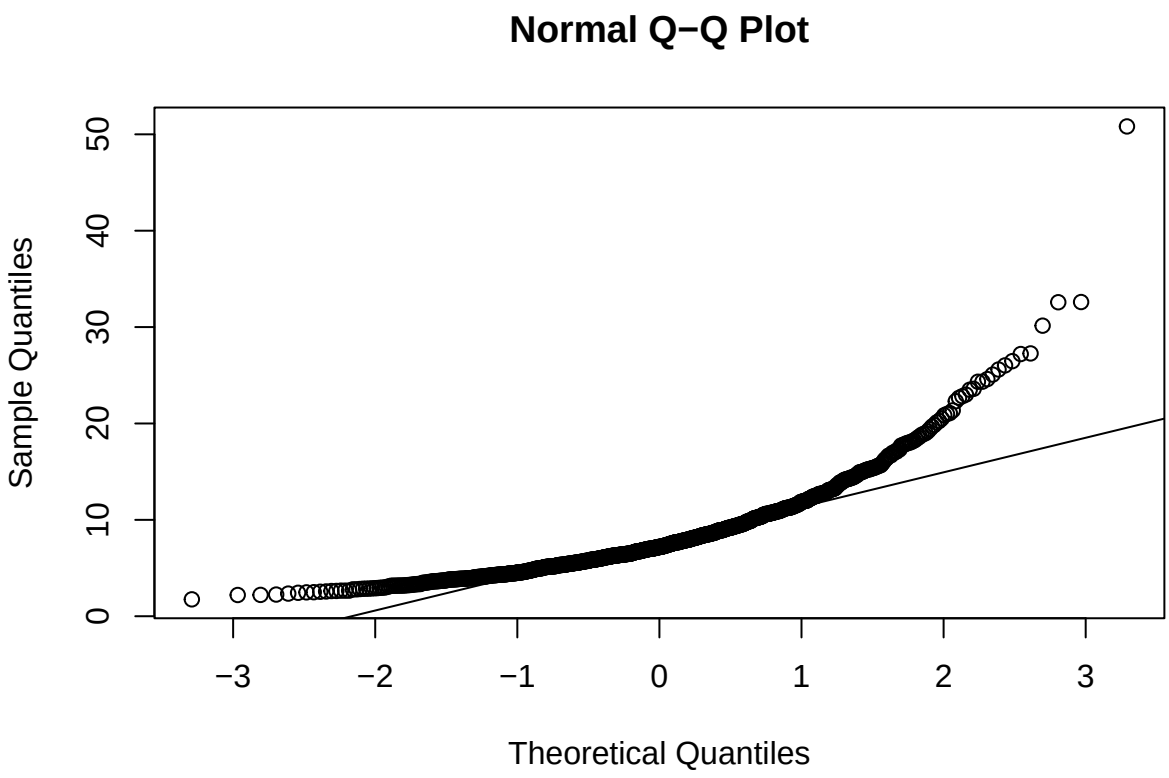
Kamala Dadashova

Problem1: The data in the file offset.csv contains data from measurements on 1000 machined metal parts representing the offset of the center of a drilled hole in cm with respect to the outside diameter of the part.

a) Make a histogram and Q-Q plot of the data. Is a normal distribution a plausible fit for this data? Justify your answer.

```
ex1 <- read.csv("offset.csv")
qqnorm(ex1$x)  # check normality with qqplot
qqline(ex1$x)  # add target line to qqplot
```



According to our observation, we can tell that it is asymmetrical and not a normal distribution because the scatter plot deviates from the qqline. It looks like a gamma distribution.

b) Estimate the parameters for a gamma distribution fit to the data using method of moments.

We have a gamma distribution with parameters α and λ. We know that the mean and variance of gamma distribution are

$$E(X) = \alpha\beta, Var(X) = \alpha\beta^2$$

Let $\bar{X}$ be sample mean and $\frac{1}{n}\sum_i (X_i - \bar{X})^2$ be a sample variance. Then using MOM method, we write

$$\alpha\beta = \bar{X}, \alpha\beta^2 = \frac{1}{n}\sum_i (X_i - \bar{X})^2$$

We solve above system and find

$$\hat{\beta} = \frac{\frac{1}{n}\sum_i (X_i - \bar{X})^2}{\bar{X}}$$

$$\hat{\alpha} = \frac{\bar{X}}{\beta} = \frac{\bar{X}^2}{\frac{1}{n}\sum_i (X_i - \bar{X})^2}$$

```
variance.data=var(ex1$x)
variance.data
```

```
## [1] 20.82667
```

```
mean.data=mean(ex1$x)
mean.data
```

```
## [1] 8.324163
```

```
alpha.hat=mean.data^2/variance.data
alpha.hat
```

```
## [1] 3.327065
```

```
beta.hat=variance.data/mean.data
beta.hat
```

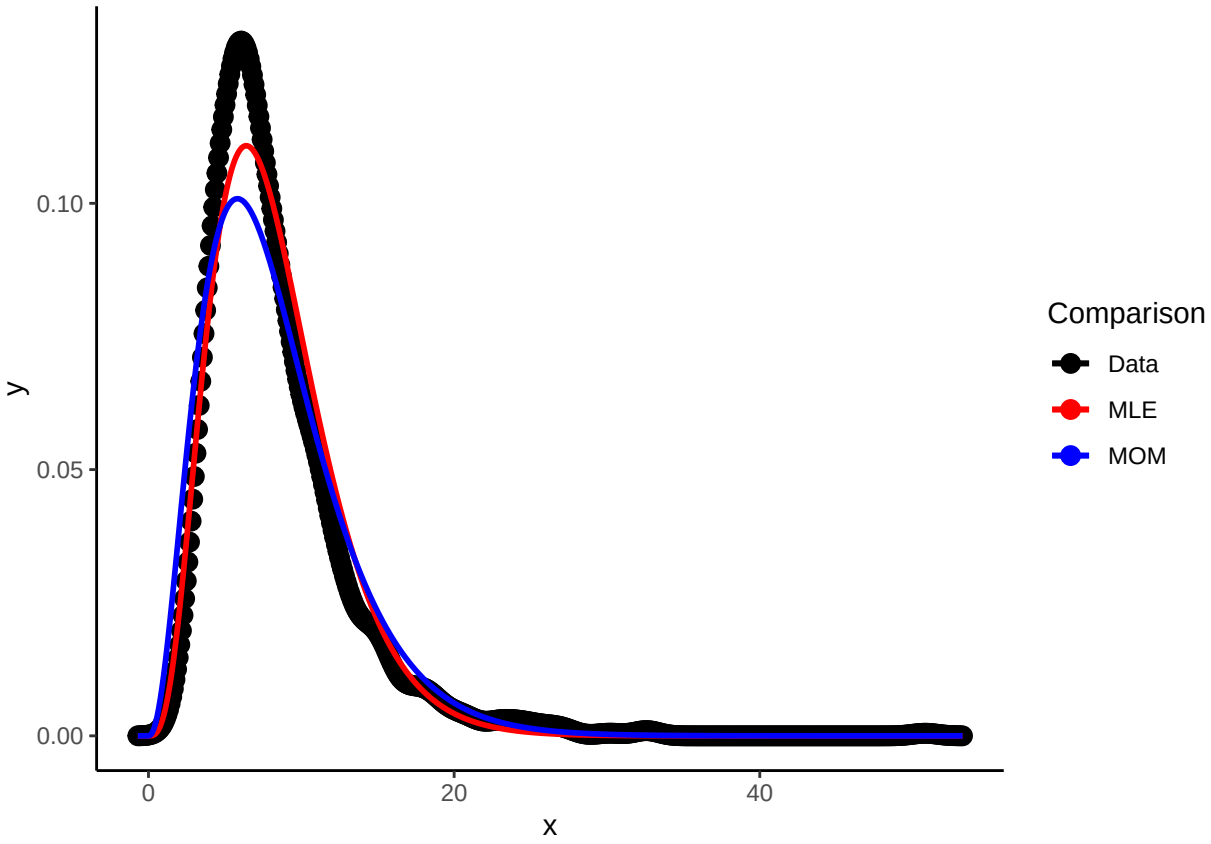
```
## [1] 2.501954
```

c) Estimate the parameters for a gamma distribution fit with an iterative maximum likelihood process using the fitdistr() command from the MASS package. You should use your estimates from part (b) as the initial values in fitdistr(). Comment on any differences in the MOM and MLE estimates.

```
 library(MASS)
 fitdistr(ex1$x,"gamma", list(shape=alpha.hat,scale=beta.hat),lower=c(0,0))
```

```
##      shape        scale
##   4.32069126   1.92658170
##  (0.18624699) (0.08806718)
```

```
library(ggplot2)
den=density(ex1$x)
data=data.frame(x=den$x,y=den$y)
ggplot(data=data,aes(x=x, y=y, color = "Data"))+geom_point(size=3)+
  geom_line(aes(x=x,y=dgamma(x, shape=4.32069126,scale= 1.92658170),color = "MLE"),size=1)+
  geom_line(aes(x=x,y=dgamma(x, shape=3.327065,scale= 2.501954),color = "MOM"),size=1)+
  theme_classic()+scale_color_manual(name = "Comparison", breaks = c("Data", "MLE", "MOM"),
values = c("Data" = "black", "MLE" = "red", "MOM" = "blue"))
```



From the plot, one can state that MLE estimate gives better fitting comparing to MOM.

d) Estimate the parameters for a log-normal distribution using method of moments.

We know that if $X \sim LogN(\mu, \sigma^2)$, then $\log(X) \sim N(\mu, \sigma^2)$. Therefore, we take log of our data. Note that $E(X) = \mu$ and $Var(X) = \sigma^2$ for normal distribution. Hence we set $\mu = \bar{X}$ and $\sigma^2 = s^2$, where $\bar{X}$ and s^2 are mean and variance of sample.

```
log.data=log(ex1$x)
mu.hat=mean(log.data)
mu.hat
```

```
## [1] 1.999
```

```
sigma.hat=sd(log.data)
sigma.hat
```

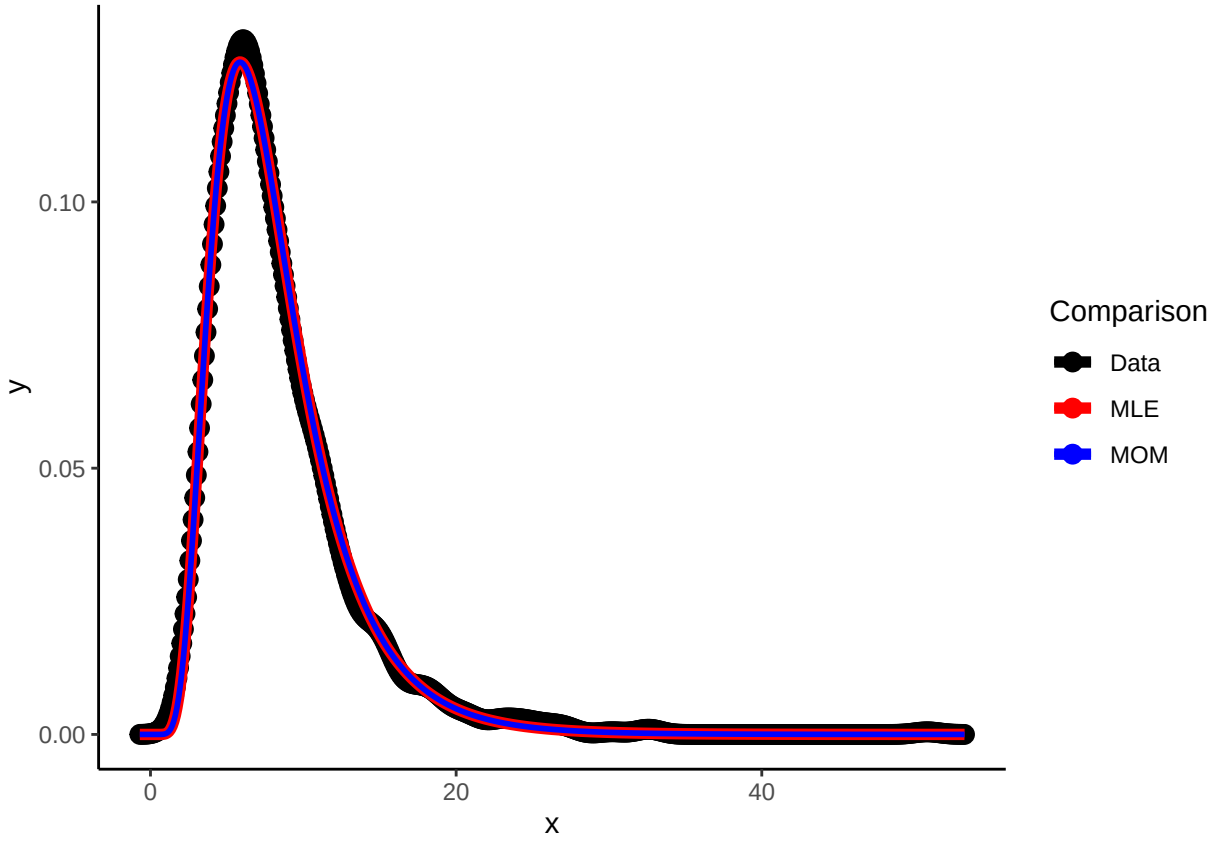
```
## [1] 0.4804472
```

e) Estimate the parameters for a log-normal distribution fit using the fitdistr() command from the MASS package. Note fitdistr() uses a closed form fit for the lognormal distribution, so initial values are not needed. Comment on any differences or similarities in the MOM and MLE estimates.

```
result_fit = fitdistr(ex1$x,"lognormal")
result_fit
```

```
##     meanlog       sdlog
##   1.99899996   0.48020691
##  (0.01518548) (0.01073775)
```

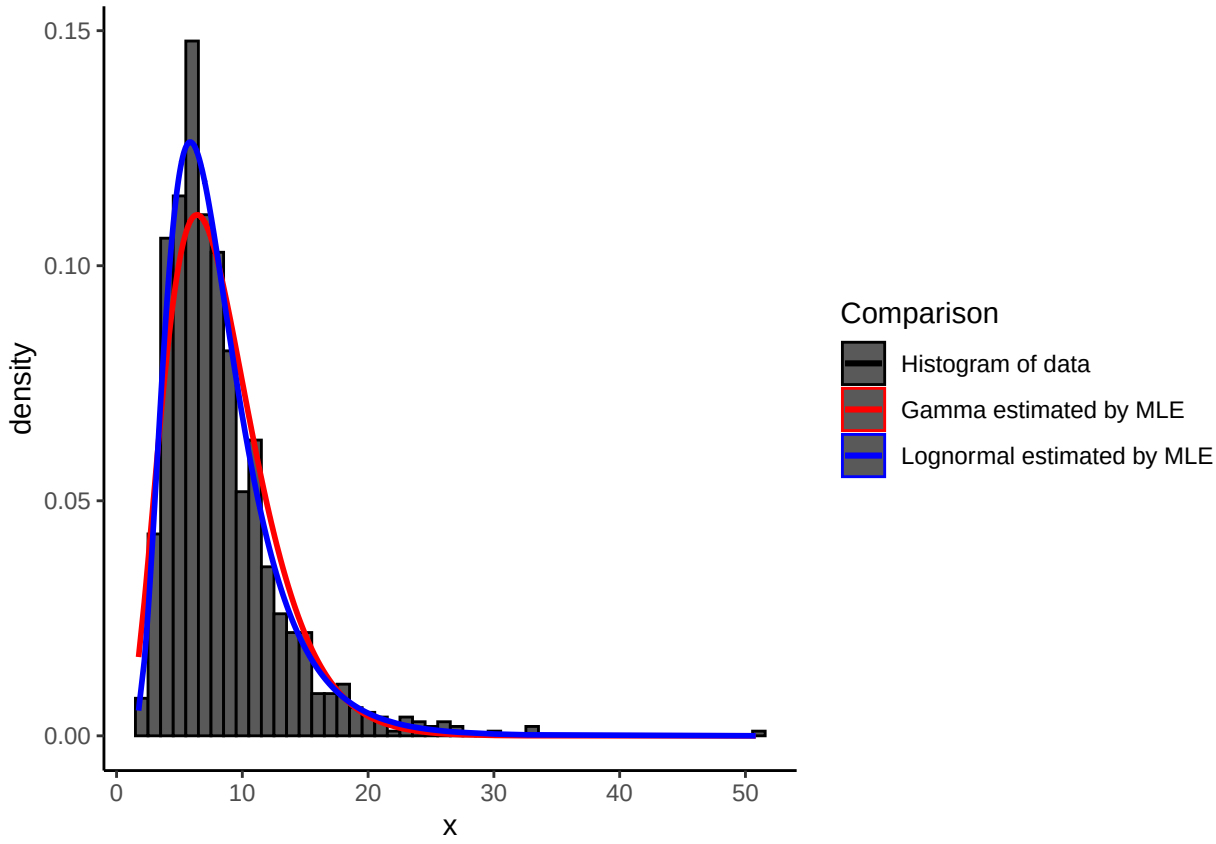
```
den=density(ex1$x)
data=data.frame(x=den$x,y=den$y)
ggplot(data=data,aes(x=x, y=y, color = "Data"))+geom_point(size=3)+
  geom_line(aes(x=x,y=dlnorm(x, meanlog=1.99899996 ,sdlog=  0.48020691),color = "MLE"),size=2)+
  geom_line(aes(x=x,y=dlnorm(x, meanlog=1.999 ,sdlog=0.4804472 ),color = "MOM"),size=1)+
  theme_classic()+scale_color_manual(name = "Comparison", breaks = c("Data", "MLE", "MOM"),
 values = c("Data" = "black", "MLE" = "red", "MOM" = "blue"))
```



As we observe, both MLE and MOM estimates almost give better fitting to data.

f) Plot the estimated gamma and log-normal densities on top of a histogram (plot the histogram with freq=F, then use the lines() command to overlay the gamma distributions). Which fit do you think is more representative of the sample?

```
den=density(ex1$x)
data=data.frame(x=den$x,y=den$y)
data = data.frame(x= ex1$x)
ggplot(data=data, aes(x=x, color = "Histogram of data"))+ geom_histogram(aes(y= ..density..), bins = 50)
geom_line(aes(x=x,y=dgamma(x, shape=4.32069126,scale= 1.92658170),
              color = "Gamma estimated by MLE"),size=1)+
  geom_line(aes(x=x,y=dlnorm(x, meanlog=1.99899996 ,sdlog=  0.48020691 ),
                color = "Lognormal estimated by MLE"),size=1)+
  theme_classic()+
  scale_color_manual(name = "Comparison",
  breaks = c("Histogram of data", "Gamma estimated by MLE", "Lognormal estimated by MLE"),
  values = c("Histogram of data" = "black", "Gamma estimated by MLE" = "red", "Lognormal estimated by ML
```



As we observe, lognormal distribution is more reprsentetive of the sample.

g) Estimate the sampling distribution and standard error the mean estimate of the log-normal distribution using the bootstrap. Plot a histogram of your estimated sampling distribution.

```
set.seed(42)
offset.smpl <- sample(ex1$x,50,replace=F)
B=10000  # run 10000 bootstrap samples
meanlog=rep(0,B)
sdlog=rep(0,B)
for(i in 1:B){
  rsample <- sample(offset.smpl,50,replace=T) # sample n=30 from population with replacement
  result_fit = fitdistr(rsample,"lognormal")
  meanlog[i]=result_fit$estimate["meanlog"]
  sdlog[i]=result_fit$estimate["sdlog"]
}

meanlog.bar <- mean(meanlog)    # calculate mean and standard error estimates
meanlog.bar
```

```
## [1] 1.969403
```

```
sdlog.bar<-mean(sdlog)
sdlog.bar
```

```
## [1] 0.4653262
```

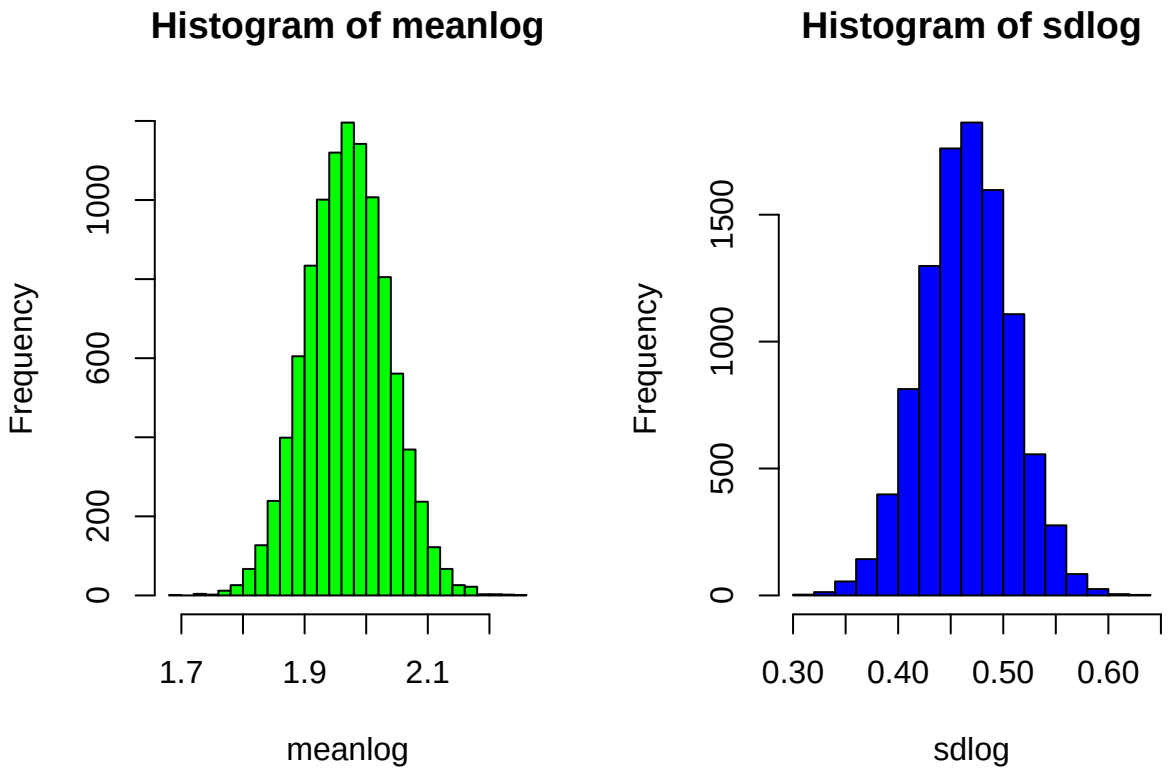
```
sd.errormean<- sd(meanlog)
sd.errormean
```

```
## [1] 0.06627875
```

```
sd.errorsd<- sd(sdlog)
sd.errorsd
```

```
## [1] 0.04220882
```

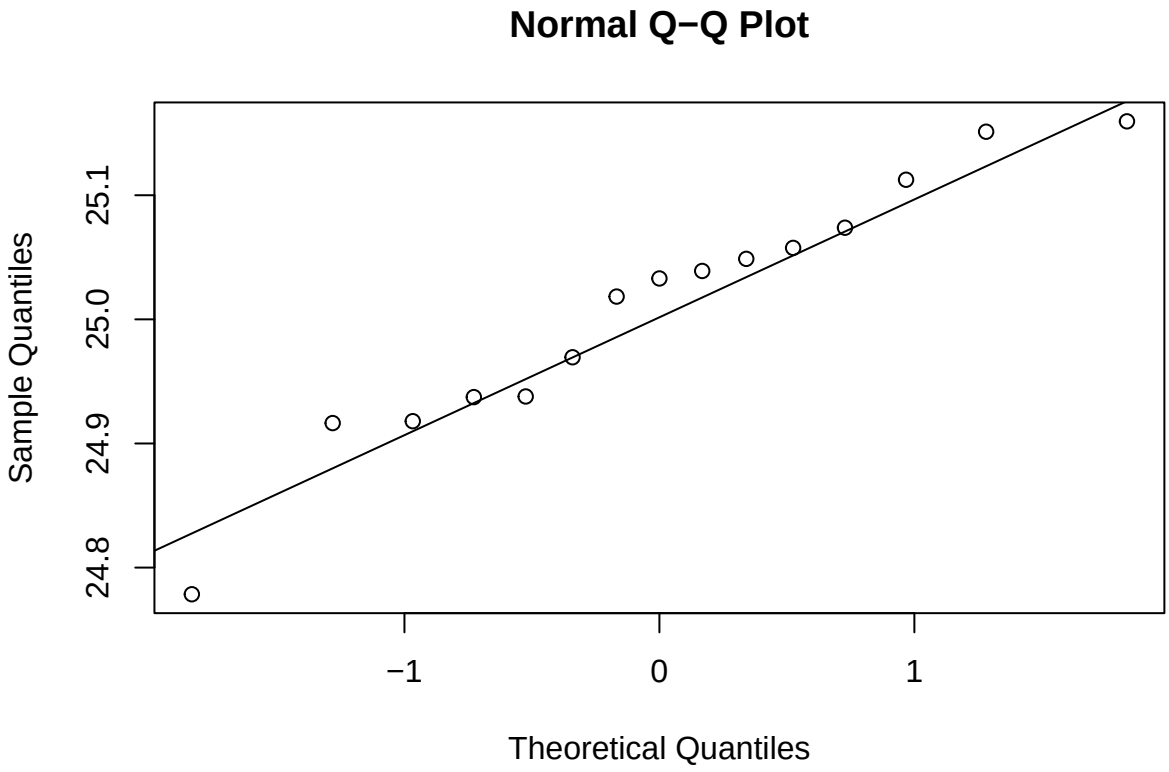
```
par(mfrow=c(1,2))
hist(meanlog,breaks=20,col="green")
hist(sdlog,breaks=20,col="blue")
```



Problem 2: A certain cylindrical machined part has nominal outside diameter of 25mm. Data for quality control measurements of 15 of these parts is provided in the file od.csv.

a) Calculate the upper bound of an exact one-sided 95% confidence interval for the variance of the diameter measurements. Explain why you believe this is an exact interval.

```
ex2 <- read.csv("od.csv")
qqnorm(ex2$x)  # check normality with qqplot
qqline(ex2$x)  # add target line to qqplot
```



So, the data is approximately normally distributed. We can compute an exact confidence interval for small sample size

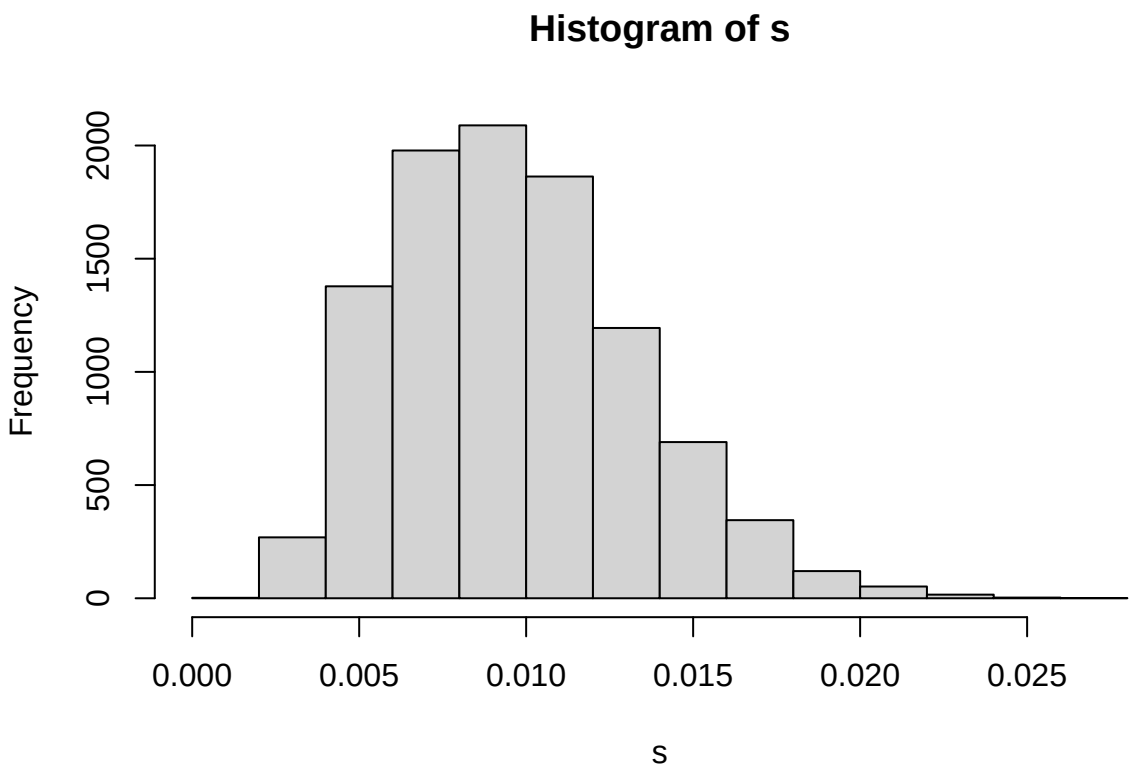
```
#exact interval
ci_exact=(15-1)*var(ex2$x)/qchisq(0.05,15-1)
ci_exact
```

```
## [1] 0.02206764
```

Exact confidence interval is $(-\infty, 0.02206764)$.

b) Find the upper bound of the one-sided 95% confidence interval using a bootstrap percentile interval. Compare the result to that from part (a) and provide a rationale for any observed differences.

```
B<-10000
s<-rep(NA,B)
set.seed(1)
for(i in 1:B){
  rsample <- sample(ex2$x,15,replace=T)
  s[i] <- var(rsample)
}
hist(s)
```




```
#Bootstrap percentile interval
quantile(s,0.95)
```

```
##        95%
## 0.01617577
```

We have $(-\infty, 0.01617577)$ 95% upper confidence interval with bootstrap. Comparing to exact confidence interval, it is narrower. It happens because of limitation of the bootstrap percentile CI.

c) Test the research hypothesis that the standard deviation of the outside diameter measurement is less than 0.1mm.

We have null hypothesis $H_0 : \sigma^2 = 0.01$, $H_a : \sigma^2 < 0.01$ The null hypothesis fall both exact$(-\infty, 0.02206764)$ and bootstrap upper $(-\infty, 0.01617577)$ confidence interval. So, we fail to reject the null hypothesis.

Problem 3: In class we used a simulation to determine when the large sample approximation was valid for testing the mean of a normal distribution by comparing the results of tests with large sample approximation to the exact t-test for 10,000 simulated random samples from a known normal distribution. The large sample approximation for testing variance of a normal distribution is based on the z-statistic

$$Z = \frac{S - \sigma_0}{\sigma_0/\sqrt{2n}}$$

Conduct a simulation to determine how large the sample size must be for the approximate test to be valid by comparing the approximate test to the exact test for the following situation

$$H_0 : \sigma^2 = 16$$
$$H_a : \sigma^2 > 16$$

with a significance level of 0.05. Simulating random samples drawn from a normal distribution with $\mu = 10$ and $\sigma^2 = 16$, perform the simulation for sample sizes $n = 3, 10, 30, 60$ and 100. Of the 5 sample sizes, which is

the smallest sample size you would recommend using the approximate method? Show supporting histograms and rejection rates from the simulation to support your recommendation.

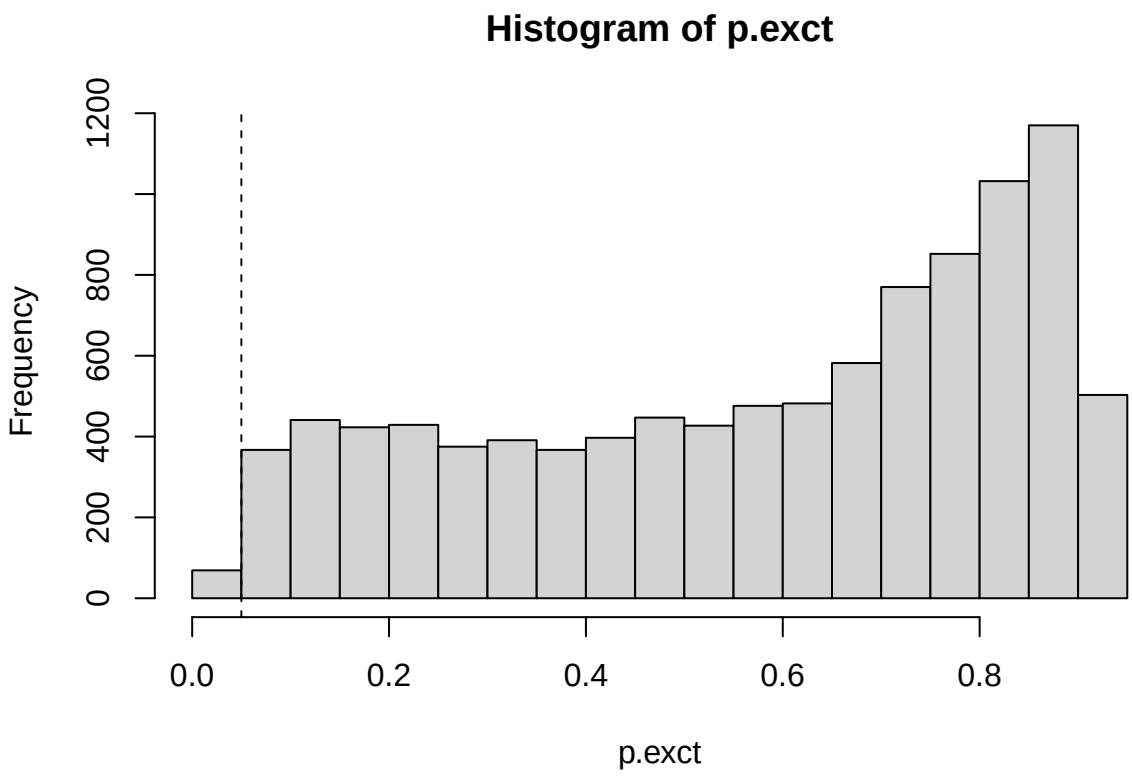
```
#  Variance Test Simulation
############################################################

# compare approximate large sample mean test to exact t-test for normally distributed data
# for various sample sizes.

set.seed(1)
M <- 10000  # set number of simulated tests
n <- 3  # sample size
sigma_0 <- 4   # null hypothesis for true mean; alt sigma2 > sigma2_0
sd.smpl <- rep(0,M)   # create dummy vectors to store simulation statistics and p-values
stat <- rep(0,M)
p.app <- rep(0,M)  # large sample p-value
p.exct <- rep(0,M)  # exact p-value
rej.app <- rep(0,M) # "1" if reject for large sample test, "0" if fail to reject
rej.exct <- rep(0,M) # "1" if reject for exact test, "0" if fail to reject
for (i in 1:M){
  smpl <- rnorm(n,10,4)  # generate sample from N(10,16)
  sd.smpl[i] <- sd(smpl)  # calculate sample sd
  stat[i] <- (sd(smpl)-sigma_0)/(sigma_0/sqrt(2*n))  # test statistic
  p.app[i] <- pnorm(stat[i],lower.tail=F)  # large sample p-value; Ha: sigma2>sigma2_0
  if(p.app[i]<0.05) rej.app[i] <- 1  # if p-value < 0.05, set rej array value to "1"
  p.exct[i] <- pt(stat[i],(n-1),lower.tail=F)  # exact p=value; Ha: sigma2>sigma2_0
  if(p.exct[i]<0.05) rej.exct[i] <- 1  # if p-value < 0.05, set rej array value to "1"
}
mean(rej.app)  # % rejected for large sample test
```

```
## [1] 0.063
```

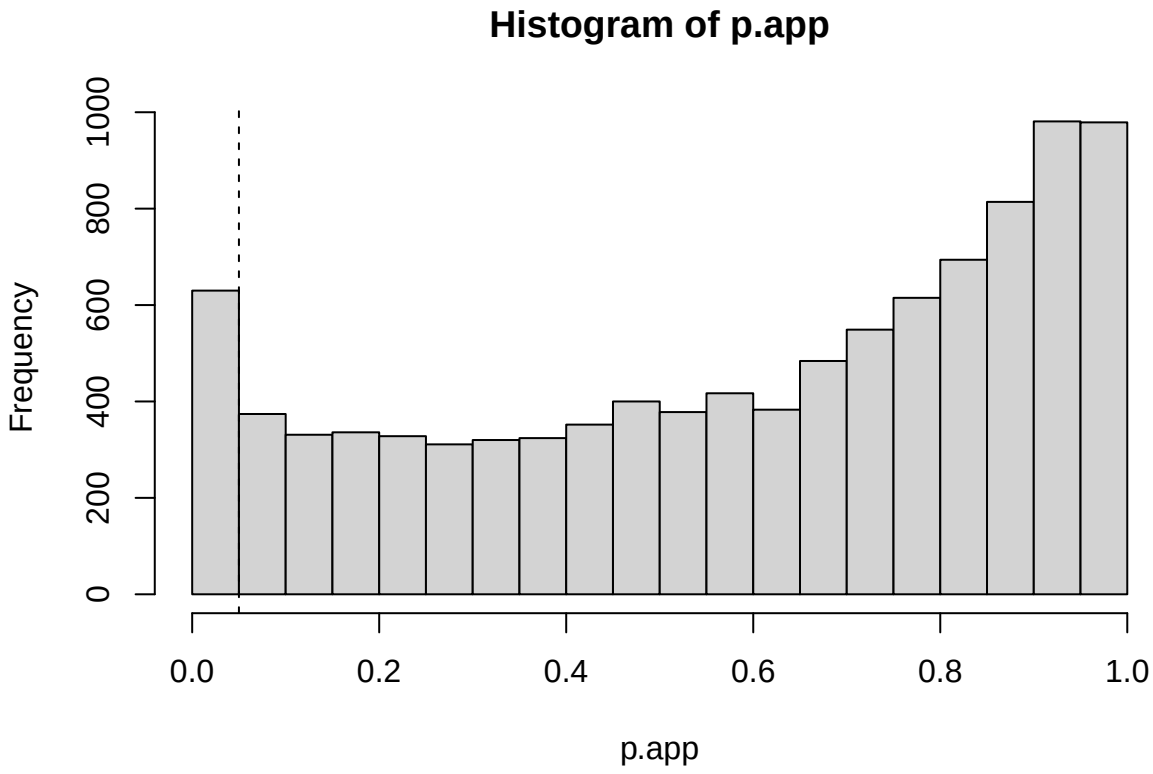
```
mean(rej.exct)  # % rejected for exact test
```

```
## [1] 0.0069
```

```
par(mfrow=c(1,1))

hist(p.exct,breaks=20) # plot distribution of exact p=values
abline(v=0.05,lty=2)
```

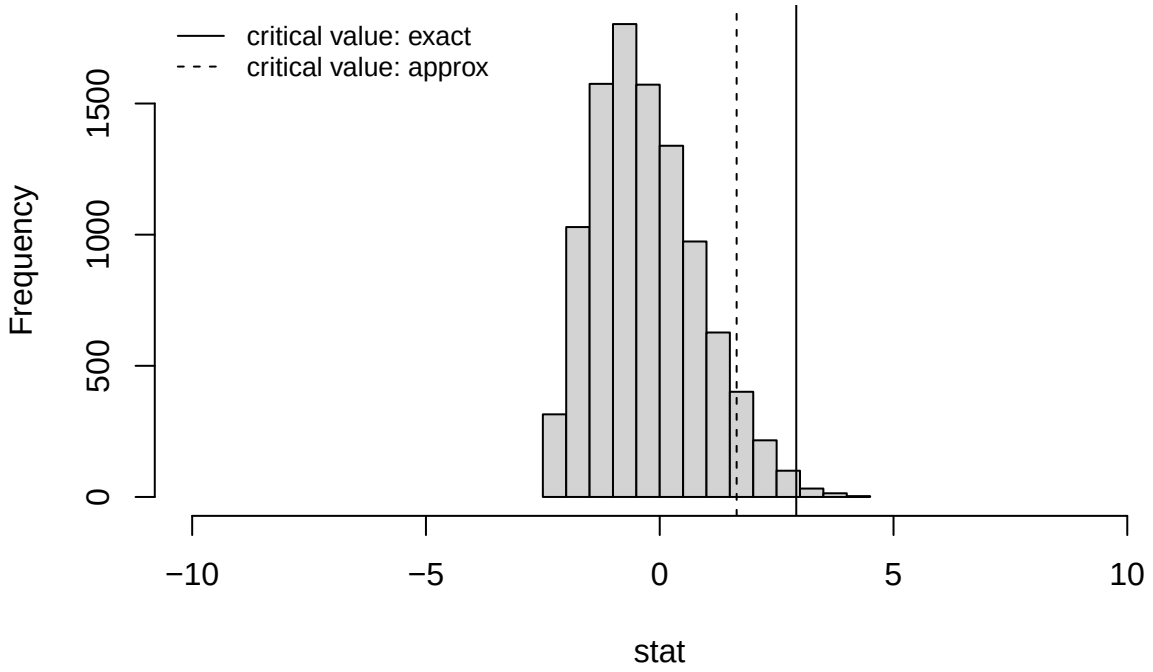



```
hist(p.app,breaks=20) # plot distribution of large sample p=values
abline(v=0.05,lty=2)
```




```
par(mfrow=c(1,1))   # plot distribution of sample statistics
hist(stat,breaks=20,xlim=c(-10,10), main="")
```

```r
# vertical lines at critical values for approx (solid), exact (dashed)
abline(v=c(qnorm(.95),qt(.95,n-1)),lty=c(2,1))
legend("topleft",legend=c("critical value: exact",
                          "critical value: approx"),
                          lty=c(1,2), bty="n", cex=0.8)
```



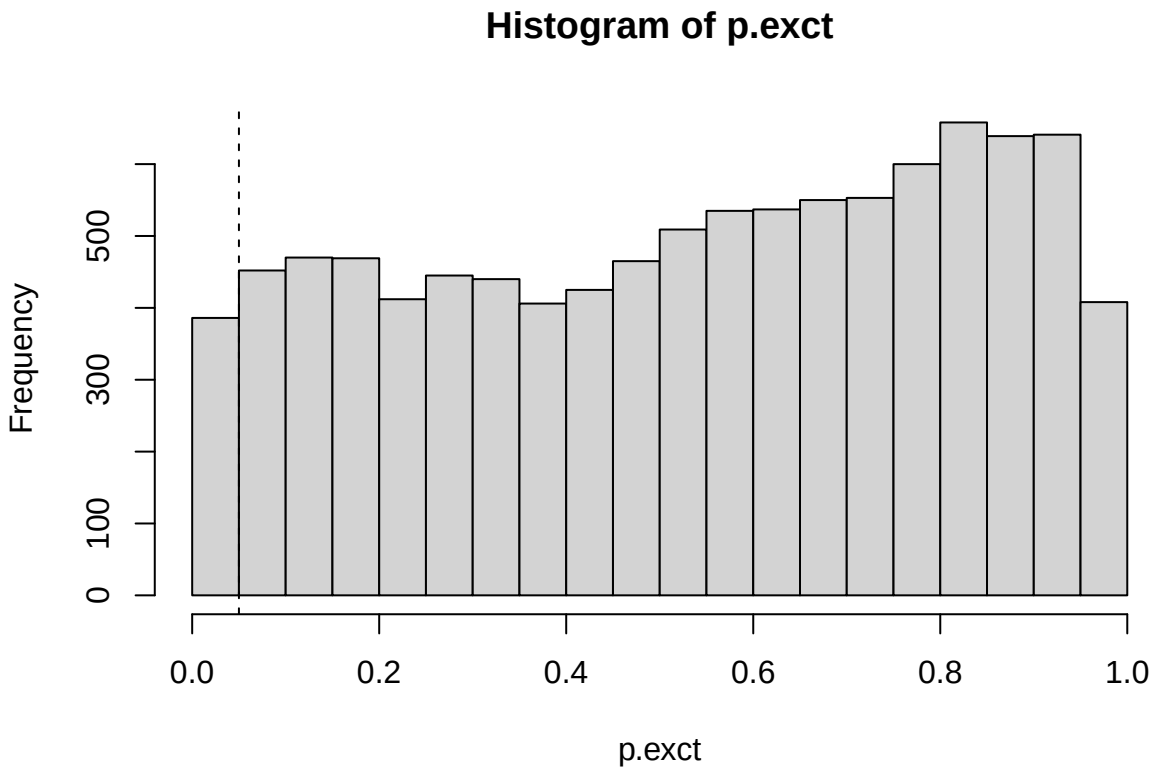

```r
#  Variance Test Simulation
################################################################

# compare approximate large sample mean test to exact t-test for normally distributed data
# for various sample sizes.

set.seed(1)
M <- 10000  # set number of simulated tests
n <- 10  # sample size
sigma_0 <- 4   # null hypothesis for true mean; alt sigma2 > sigma2_0
sd.smpl <- rep(0,M)   # create dummy vectors to store simulation statistics and p-values
stat <- rep(0,M)
p.app <- rep(0,M)  # large sample p-value
p.exct <- rep(0,M)  # exact p-value
rej.app <- rep(0,M) # "1" if reject for large sample test, "0" if fail to reject
rej.exct <- rep(0,M) # "1" if reject for exact test, "0" if fail to reject
for (i in 1:M){
  smpl <- rnorm(n,10,4)  # generate sample from N(10,16)
  sd.smpl[i] <- sd(smpl)  # calculate sample sd
  stat[i] <- (sd(smpl)-sigma_0)/(sigma_0/sqrt(2*n))  # test statistic
  p.app[i] <- pnorm(stat[i],lower.tail=F)  # large sample p-value; Ha: sigma2>sigma2_0
  if(p.app[i]<0.05) rej.app[i] <- 1  # if p-value < 0.05, set rej array value to "1"
  p.exct[i] <- pt(stat[i],(n-1),lower.tail=F)  # exact p=value; Ha: sigma2>sigma2_0
  if(p.exct[i]<0.05) rej.exct[i] <- 1  # if p-value < 0.05, set rej array value to "1"
}
mean(rej.app)  # % rejected for large sample test
```

```
## [1] 0.0533
```

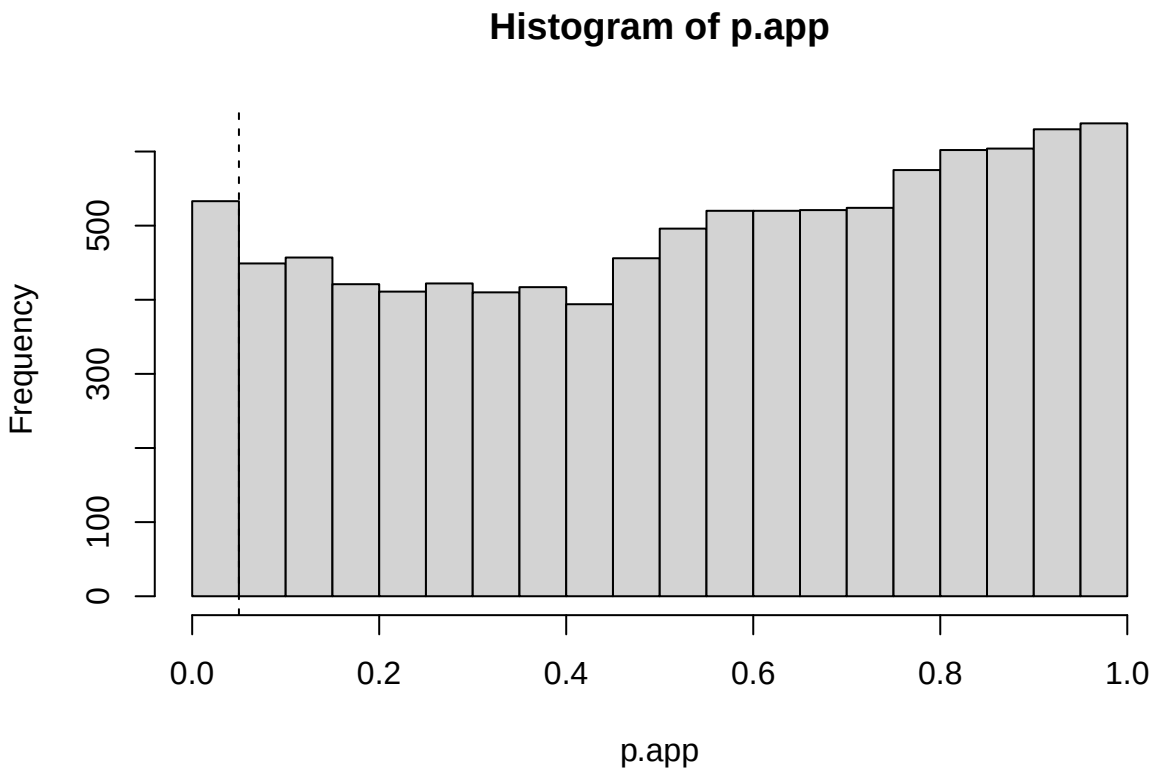
```
mean(rej.exct)  # % rejected for exact test
```

```
## [1] 0.0386
```

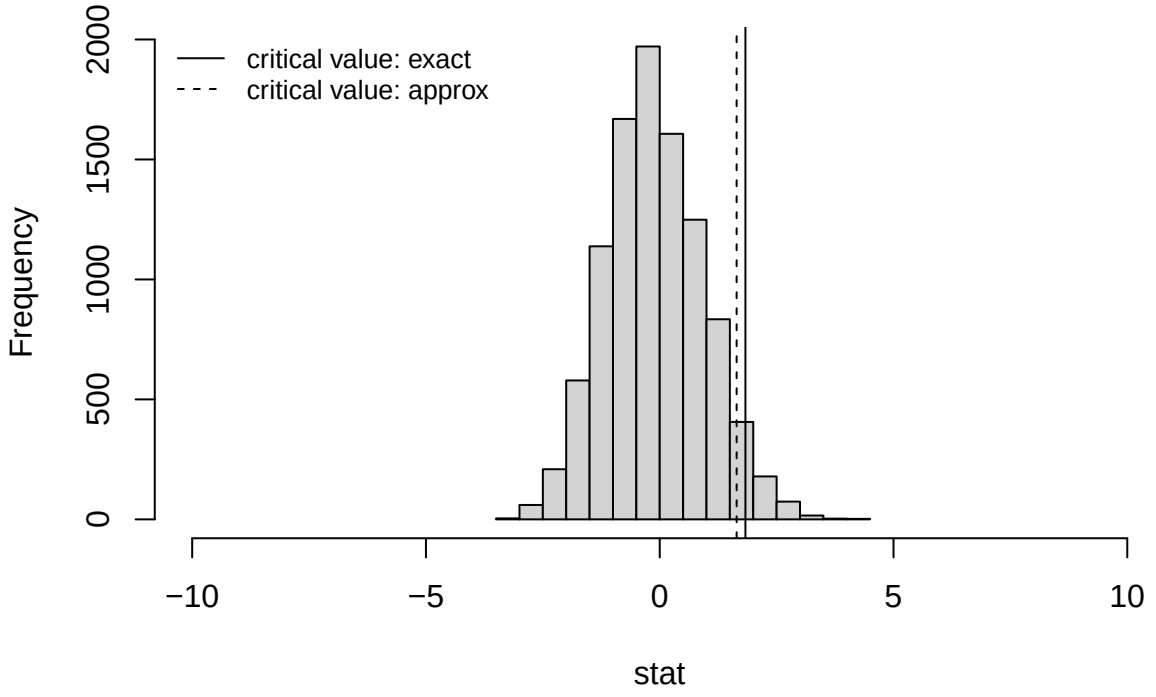
```
par(mfrow=c(1,1))

hist(p.exct,breaks=20) # plot distribution of exact p=values
abline(v=0.05,lty=2)
```




```
hist(p.app,breaks=20) # plot distribution of large sample p=values
abline(v=0.05,lty=2)
```




```
par(mfrow=c(1,1))   # plot distribution of sample statistics
hist(stat,breaks=20,xlim=c(-10,10), main="")
# vertical lines at critical values for approx (solid), exact (dashed)
abline(v=c(qnorm(.95),qt(.95,n-1)),lty=c(2,1))
legend("topleft",legend=c("critical value: exact",
                          "critical value: approx"),
                          lty=c(1,2), bty="n", cex=0.8)
```



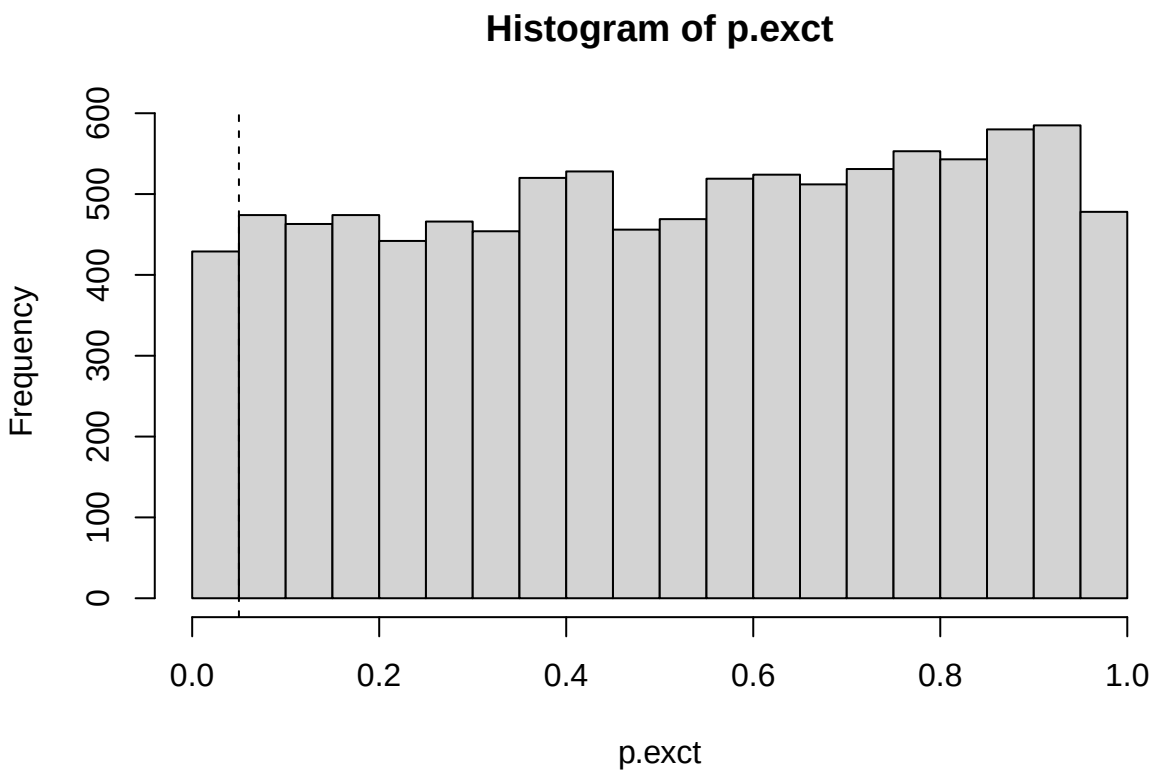
```r
#  Variance Test Simulation
############################################################

# compare approximate large sample mean test to exact t-test for normally distributed data
# for various sample sizes.

set.seed(1)
M <- 10000  # set number of simulated tests
n <- 30  # sample size
sigma_0 <- 4   # null hypothesis for true mean; alt sigma2 > sigma2_0
sd.smpl <- rep(0,M)   # create dummy vectors to store simulation statistics and p-values
stat <- rep(0,M)
p.app <- rep(0,M)  # large sample p-value
p.exct <- rep(0,M)  # exact p-value
rej.app <- rep(0,M) # "1" if reject for large sample test, "0" if fail to reject
rej.exct <- rep(0,M) # "1" if reject for exact test, "0" if fail to reject
for (i in 1:M){
  smpl <- rnorm(n,10,4)  # generate sample from N(10,16)
  sd.smpl[i] <- sd(smpl)  # calculate sample sd
  stat[i] <- (sd(smpl)-sigma_0)/(sigma_0/sqrt(2*n))  # test statistic
  p.app[i] <- pnorm(stat[i],lower.tail=F)  # large sample p-value; Ha: sigma2>sigma2_0
  if(p.app[i]<0.05) rej.app[i] <- 1  # if p-value < 0.05, set rej array value to "1"
  p.exct[i] <- pt(stat[i],(n-1),lower.tail=F)  # exact p=value; Ha: sigma2>sigma2_0
  if(p.exct[i]<0.05) rej.exct[i] <- 1  # if p-value < 0.05, set rej array value to "1"
}
mean(rej.app)  # % rejected for large sample test
```

```
## [1] 0.0483
```

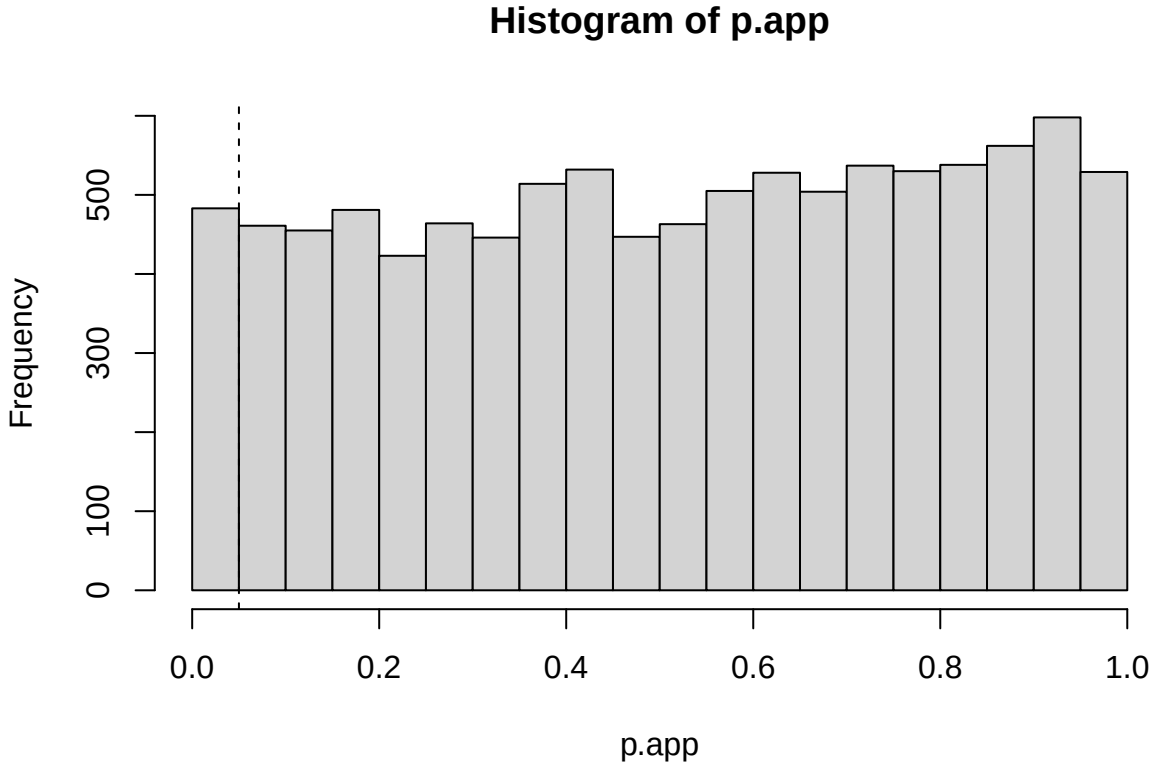
```r
mean(rej.exct)  # % rejected for exact test
```

```
## [1] 0.0429
```

```r
par(mfrow=c(1,1))

hist(p.exct,breaks=20) # plot distribution of exact p=values
abline(v=0.05,lty=2)
```

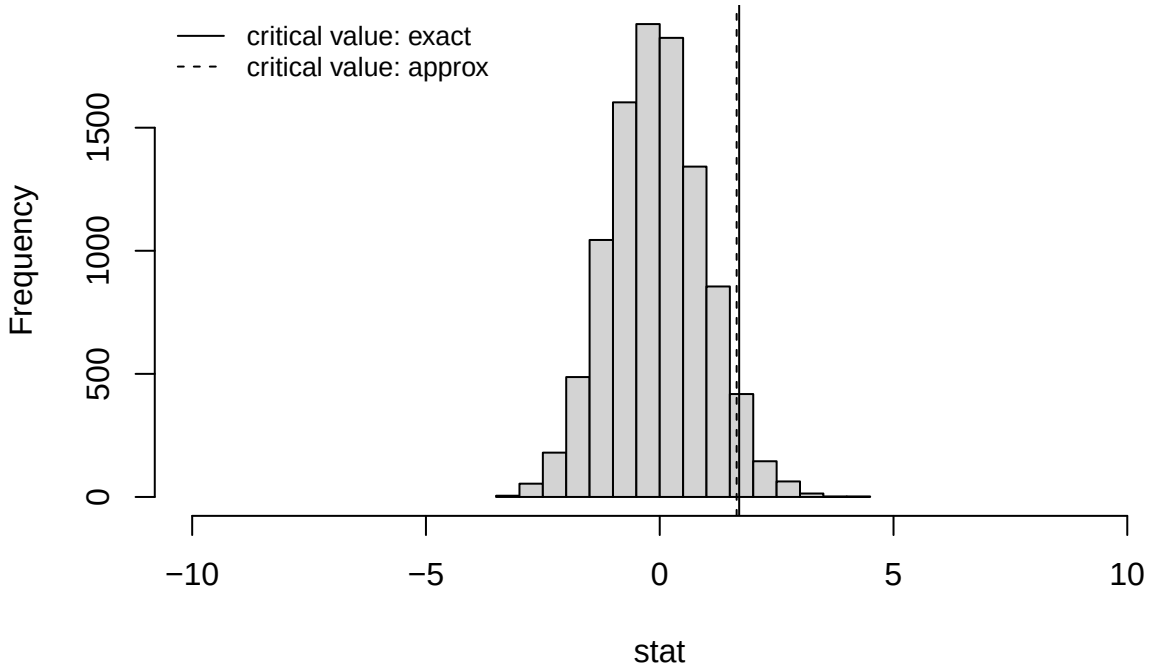



```
hist(p.app,breaks=20) # plot distribution of large sample p=values
abline(v=0.05,lty=2)
```




```
par(mfrow=c(1,1))   # plot distribution of sample statistics
hist(stat,breaks=20,xlim=c(-10,10), main="")
```

```
# vertical lines at critical values for approx (solid), exact (dashed)
abline(v=c(qnorm(.95),qt(.95,n-1)),lty=c(2,1))
legend("topleft",legend=c("critical value: exact",
                          "critical value: approx"),
                          lty=c(1,2), bty="n", cex=0.8)
```



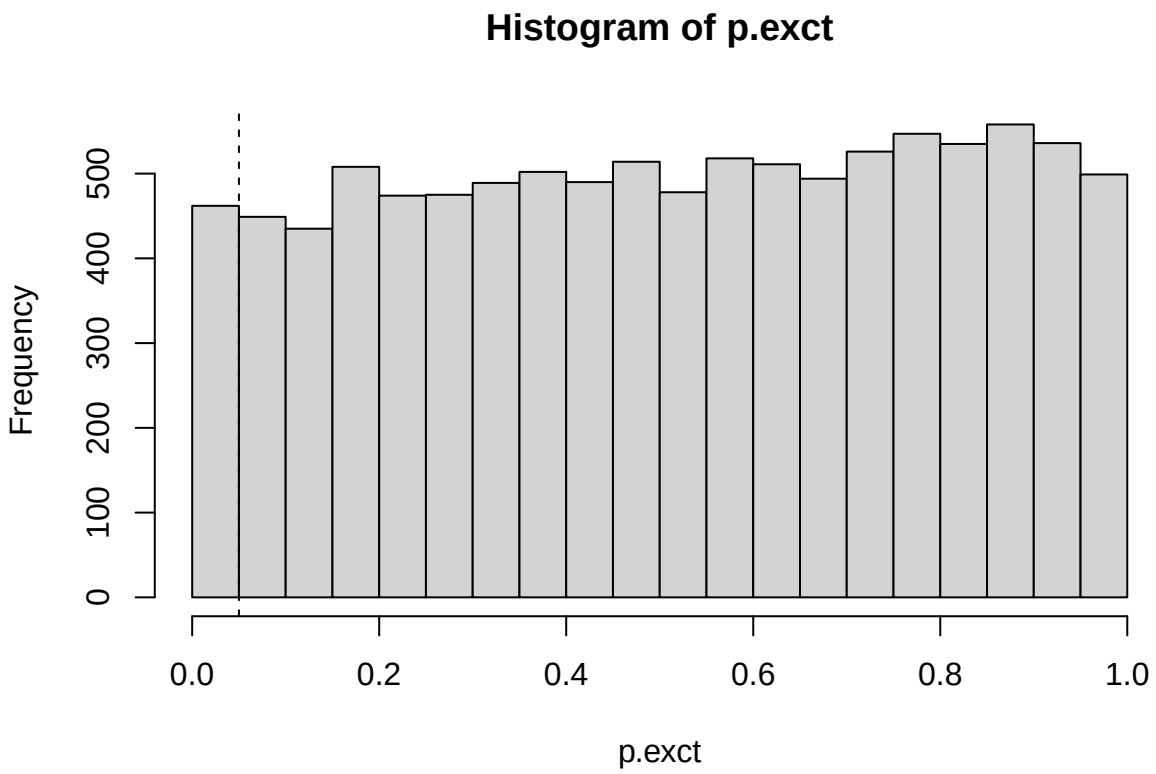

```
#  Variance Test Simulation
############################################################

# compare approximate large sample mean test to exact t-test for normally distributed data
# for various sample sizes.

set.seed(1)
M <- 10000  # set number of simulated tests
n <- 60  # sample size
sigma_0 <- 4   # null hypothesis for true mean; alt sigma2 > sigma2_0
sd.smpl <- rep(0,M)   # create dummy vectors to store simulation statistics and p-values
stat <- rep(0,M)
p.app <- rep(0,M)  # large sample p-value
p.exct <- rep(0,M)  # exact p-value
rej.app <- rep(0,M) # "1" if reject for large sample test, "0" if fail to reject
rej.exct <- rep(0,M) # "1" if reject for exact test, "0" if fail to reject
for (i in 1:M){
  smpl <- rnorm(n,10,4)  # generate sample from N(10,16)
  sd.smpl[i] <- sd(smpl)  # calculate sample sd
  stat[i] <- (sd(smpl)-sigma_0)/(sigma_0/sqrt(2*n))  # test statistic
  p.app[i] <- pnorm(stat[i],lower.tail=F)  # large sample p-value; Ha: sigma2>sigma2_0
  if(p.app[i]<0.05) rej.app[i] <- 1  # if p-value < 0.05, set rej array value to "1"
  p.exct[i] <- pt(stat[i],(n-1),lower.tail=F)  # exact p=value; Ha: sigma2>sigma2_0
  if(p.exct[i]<0.05) rej.exct[i] <- 1  # if p-value < 0.05, set rej array value to "1"
}
mean(rej.app)  # % rejected for large sample test
```

```
## [1] 0.0491
```

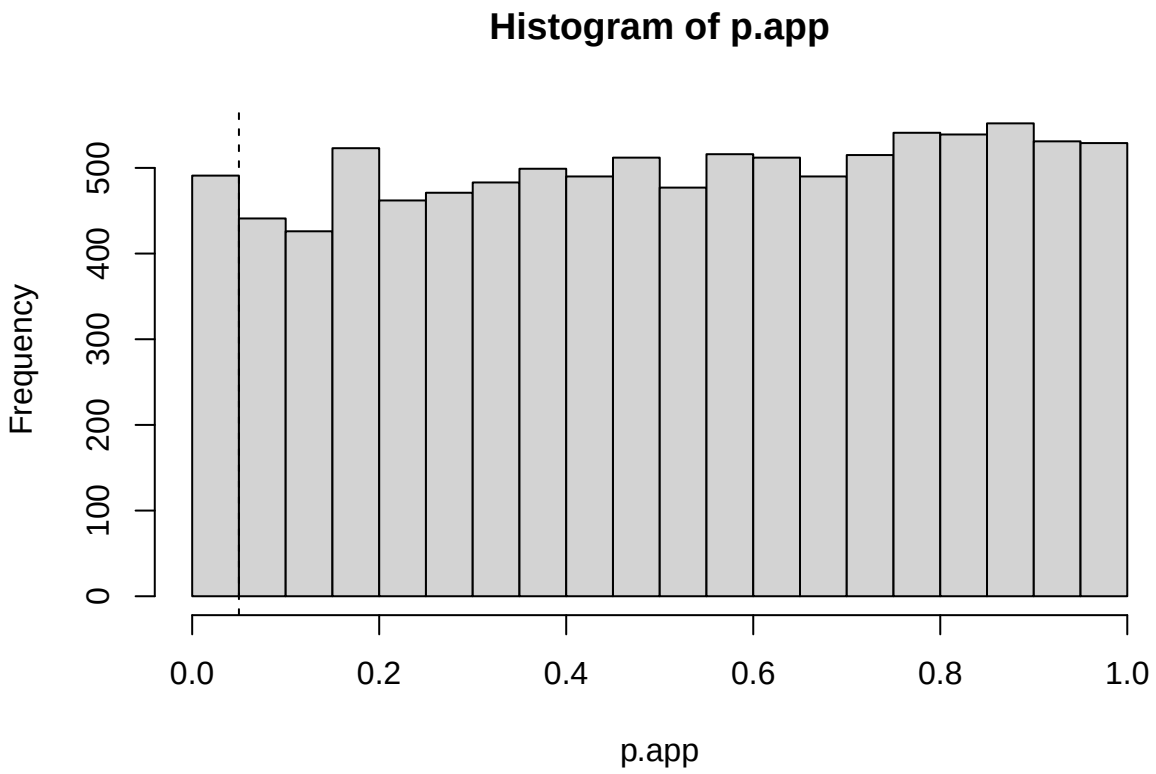
```
mean(rej.exct)  # % rejected for exact test
```

```
## [1] 0.0462
```

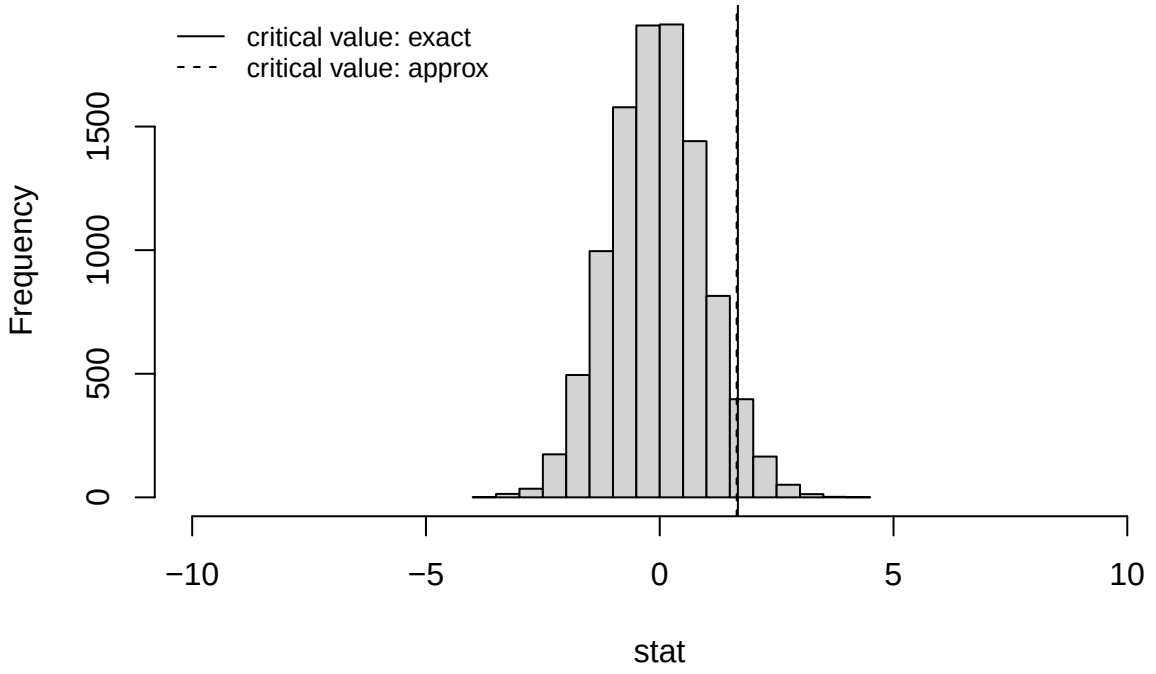
```
par(mfrow=c(1,1))

hist(p.exct,breaks=20) # plot distribution of exact p=values
abline(v=0.05,lty=2)
```




```
hist(p.app,breaks=20) # plot distribution of large sample p=values
abline(v=0.05,lty=2)
```




```
par(mfrow=c(1,1))   # plot distribution of sample statistics
hist(stat,breaks=20,xlim=c(-10,10), main="")
# vertical lines at critical values for approx (solid), exact (dashed)
abline(v=c(qnorm(.95),qt(.95,n-1)),lty=c(2,1))
legend("topleft",legend=c("critical value: exact",
                          "critical value: approx"),
                          lty=c(1,2), bty="n", cex=0.8)
```



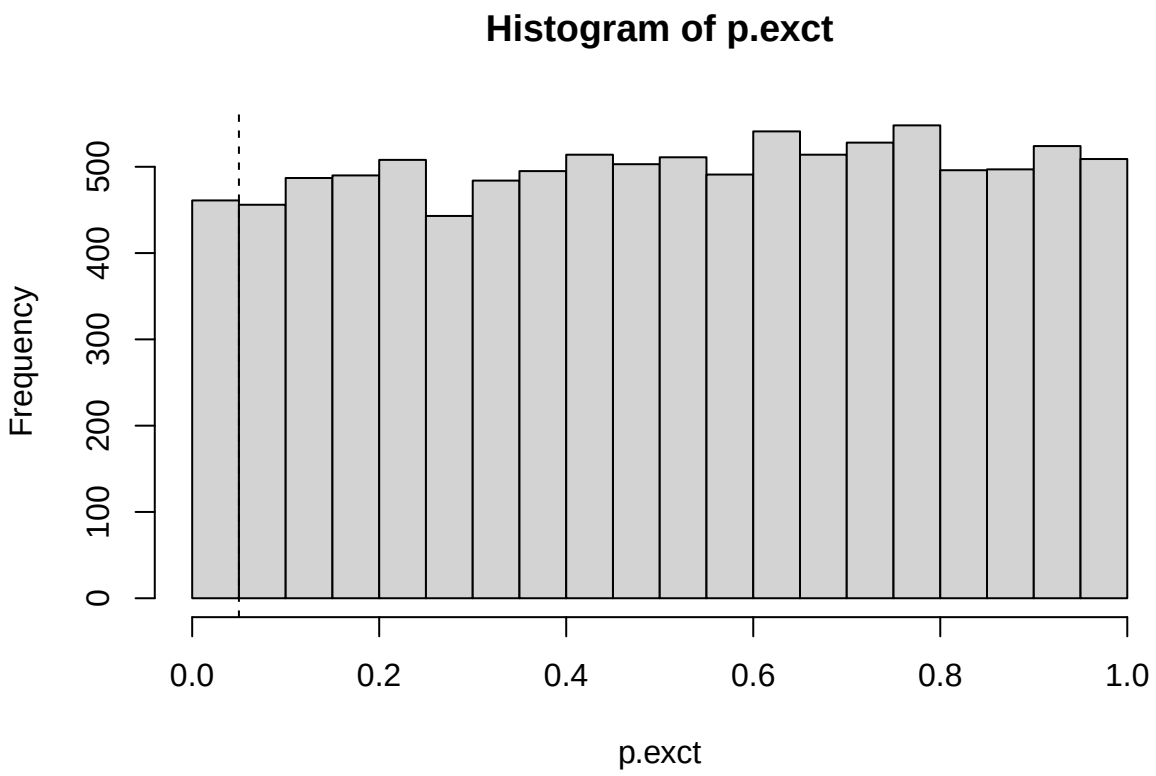
```r
#  Variance Test Simulation
############################################################

# compare approximate large sample mean test to exact t-test for normally distributed data
# for various sample sizes.

set.seed(1)
M <- 10000  # set number of simulated tests
n <- 100  # sample size
sigma_0 <- 4   # null hypothesis for true mean; alt sigma2 > sigma2_0
sd.smpl <- rep(0,M)   # create dummy vectors to store simulation statistics and p-values
stat <- rep(0,M)
p.app <- rep(0,M)  # large sample p-value
p.exct <- rep(0,M)  # exact p-value
rej.app <- rep(0,M) # "1" if reject for large sample test, "0" if fail to reject
rej.exct <- rep(0,M) # "1" if reject for exact test, "0" if fail to reject
for (i in 1:M){
  smpl <- rnorm(n,10,4)  # generate sample from N(10,16)
  sd.smpl[i] <- sd(smpl)  # calculate sample sd
  stat[i] <- (sd(smpl)-sigma_0)/(sigma_0/sqrt(2*n))  # test statistic
  p.app[i] <- pnorm(stat[i],lower.tail=F)  # large sample p-value; Ha: sigma2>sigma2_0
  if(p.app[i]<0.05) rej.app[i] <- 1  # if p-value < 0.05, set rej array value to "1"
  p.exct[i] <- pt(stat[i],(n-1),lower.tail=F)  # exact p=value; Ha: sigma2>sigma2_0
  if(p.exct[i]<0.05) rej.exct[i] <- 1  # if p-value < 0.05, set rej array value to "1"
}
mean(rej.app)  # % rejected for large sample test
```

```
## [1] 0.0474
```

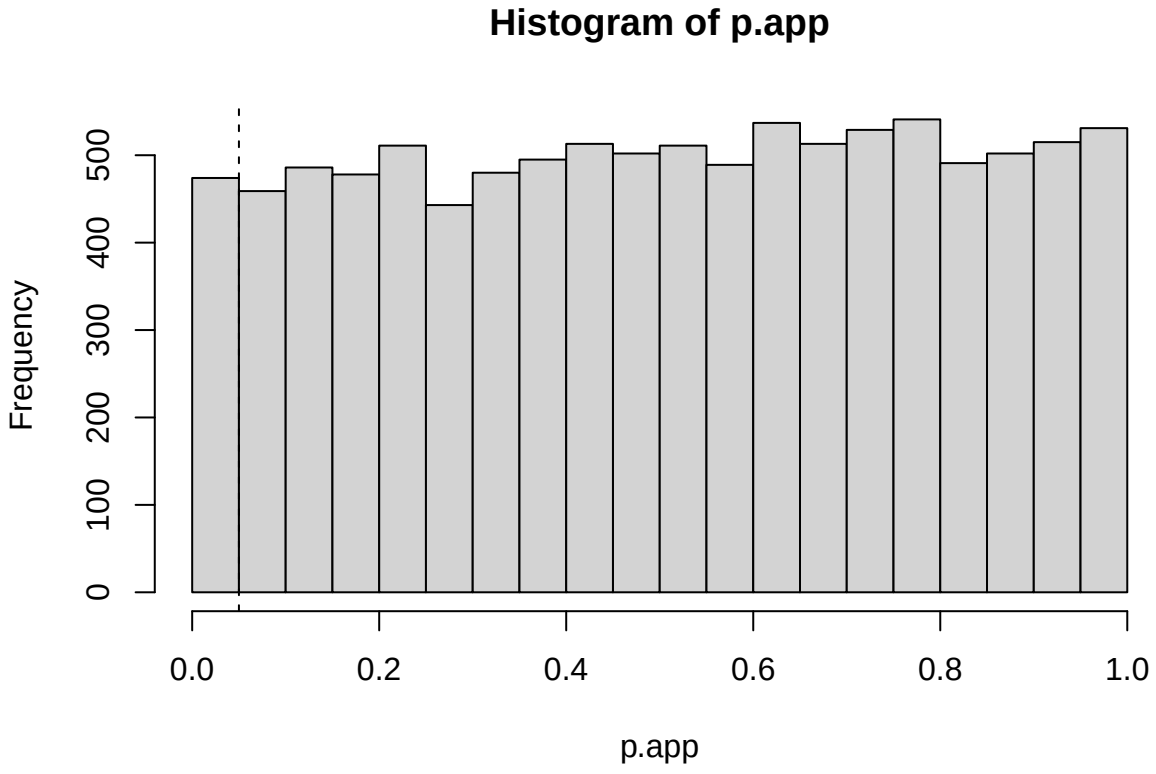
```r
mean(rej.exct)  # % rejected for exact test
```

```
## [1] 0.0461
```

```r
par(mfrow=c(1,1))

hist(p.exct,breaks=20) # plot distribution of exact p=values
abline(v=0.05,lty=2)
```

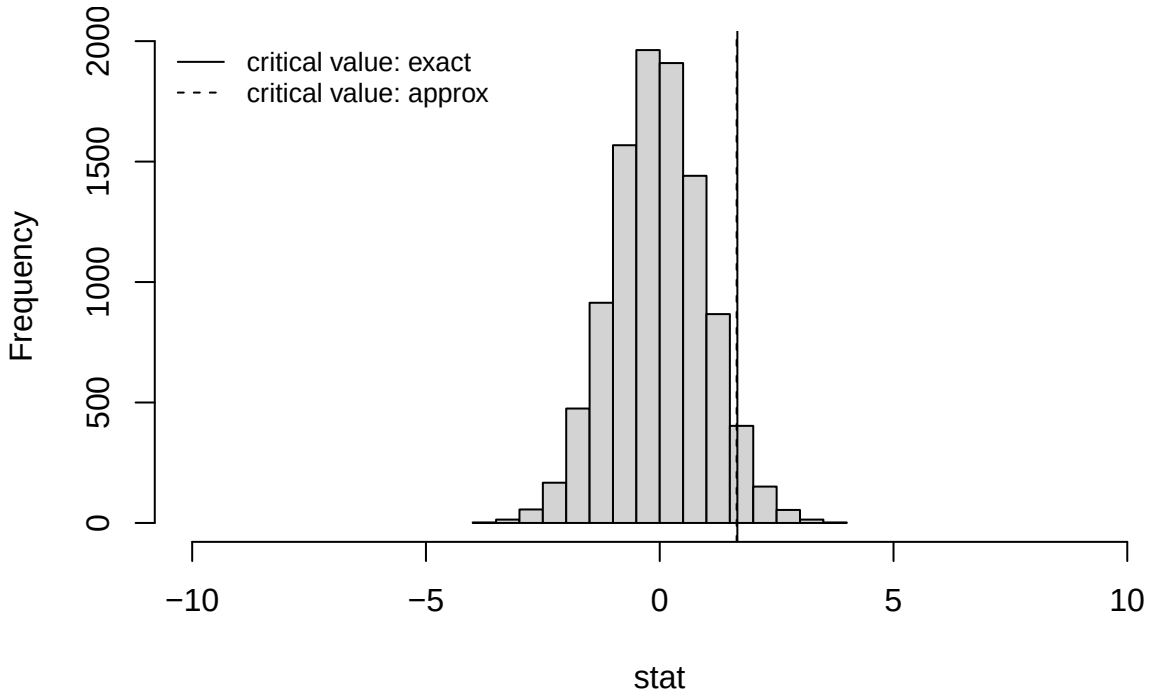



```
hist(p.app,breaks=20) # plot distribution of large sample p=values
abline(v=0.05,lty=2)
```




```
par(mfrow=c(1,1))   # plot distribution of sample statistics
hist(stat,breaks=20,xlim=c(-10,10), main="")
```

```r
# vertical lines at critical values for approx (solid), exact (dashed)
abline(v=c(qnorm(.95),qt(.95,n-1)),lty=c(2,1))
legend("topleft",legend=c("critical value: exact",
                          "critical value: approx"),
                          lty=c(1,2), bty="n", cex=0.8)
```



We would recommend the smallest sample size to be equal to n=60. Histograms for both exact and approximated p value are almost uniform compare to n=3,10,30. We can also observe that critical values for both cases are very near each other. Regarding rejection rate, $r_{exact} = 0.0491$ and $r_{approx} = 0.0462$, so they are also close to each other. Those facts support our choice of sample size.

Problem 4: Suppose we have a random sample $X_1, X_2, \ldots, X_n$ such that the X's follow a standard beta distribution with parameters α and $\beta = 2\alpha$.

a) Find the method of moments (MOM) estimator for α.

For standard beta distribution, we have

$$E(X) = \frac{\alpha}{\alpha + \beta}$$

$$V(X) = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}$$

We then set

$$\bar{X} = \frac{\alpha}{\alpha + \beta}$$

$$s^2 = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}$$

where $\bar{X}$ and s^2 are sample mean and variance. From the first equation, we have

$$(\alpha + \beta)\bar{X} = \alpha$$

We solve for β, and get

$$\beta = \frac{\alpha}{\bar{X}} - \alpha$$

From the second equation, we have

$$\alpha\beta = s^2(\alpha+\beta)^2(\alpha+\beta+1)$$
$$\alpha(\frac{\alpha}{\bar{X}} - \alpha) = s^2(\alpha + \frac{\alpha}{\bar{X}} - \alpha)^2(\alpha + \frac{\alpha}{\bar{X}} - \alpha + 1)$$

$$\frac{\alpha^2}{\bar{X}} - \alpha^2 = s^2(\frac{\alpha}{\bar{X}})^2(\frac{\alpha}{\bar{X}} + 1)$$

$$\alpha^2(\frac{1}{\bar{X}} - 1) = \alpha^2(\frac{1}{\bar{X}^2})(\frac{\alpha}{\bar{X}} + 1)s^2$$

$$(\frac{1}{\bar{X}} - 1) = (\frac{1}{\bar{X}^2})(\frac{\alpha}{\bar{X}} + 1)s^2$$
$$(\frac{1}{\bar{X}} - 1)\frac{\bar{X}^2}{s^2} = (\frac{\alpha}{\bar{X}} + 1)$$
$$\hat{\alpha}_{MOM} = \bar{X}\left(\frac{\bar{X}(1-\bar{X})}{s^2} - 1\right)$$

b) Write the log-likelihood function $l(\alpha)$ Th pdf of standard beta distribution is

$$f(X) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}X^{\alpha-1}(1-X)^{\beta-1}$$

$$f(X_1, X2, ...X_n, \alpha, \beta) = \prod_{i=1}^{n} f(X_i) = \left(\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\right)^n \left(\prod_{i=1}^{n} X_i\right)^{\alpha-1} \left(\prod_{i=1}^{n}(1-X_i)\right)^{\beta-1}$$

Next, we take log of both sides,

$$ln(f(X_1, X_2, ..., X_n) = n\ln(\Gamma(\alpha+\beta)) - n\Gamma(\alpha) - n\Gamma(\beta) +$$
$$+(\alpha-1)\sum_{i=1}^{n}\ln(X_i) + (\beta-1)\sum_{i=1}^{n}\ln(1-X_i)$$

We are given that $\beta = 2\alpha$, thus

$$ln(f(X_1, X_2, ..., X_n) = n\ln(\Gamma(3\alpha)) - n\Gamma(\alpha) - n\Gamma(2\alpha) +$$
$$+(\alpha-1)\sum_{i=1}^{n}\ln(X_i) + (2\alpha-1)\sum_{i=1}^{n}\ln(1-X_i)$$

c) Derive an equation that the MLE of α must satisfy.

We take derivative of log-likelihood function in part (b) respect to α and set it equal to zero

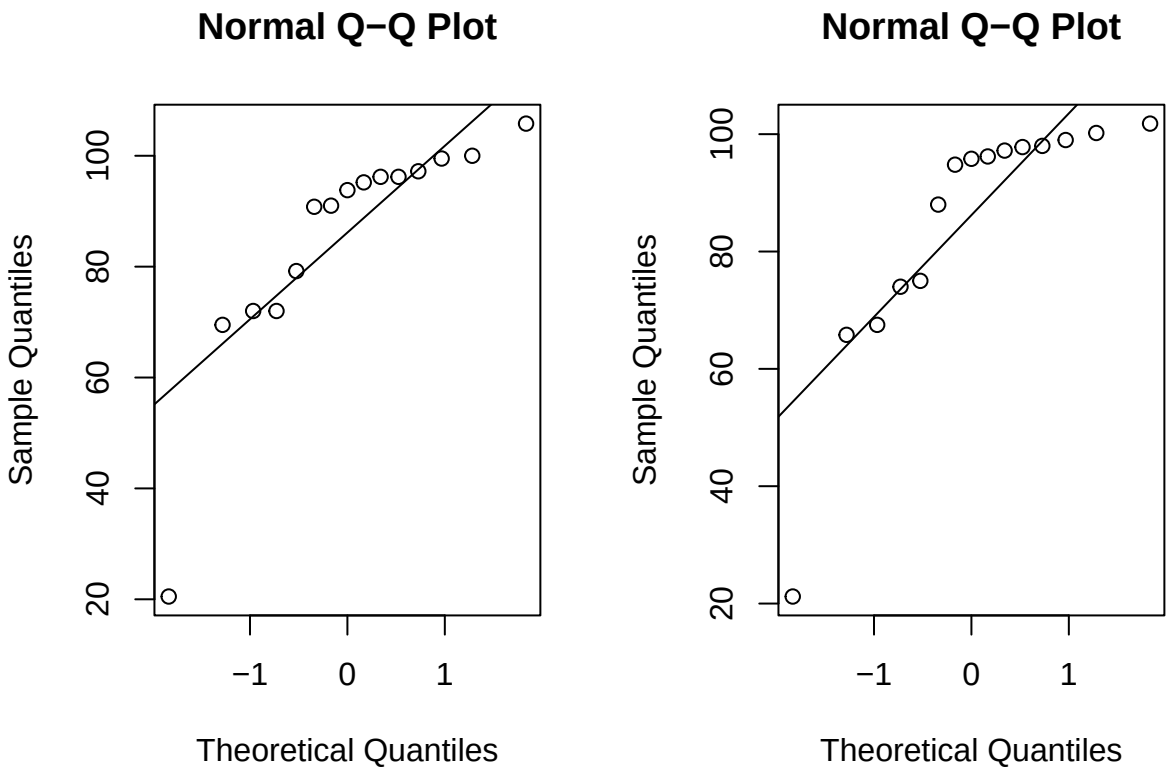
$$n\frac{\Gamma'(3\alpha)}{\Gamma(3\alpha)} - n\Gamma'(\alpha) - n\Gamma'(2\alpha) + \sum_{i=1}^{n}\ln(X_i) + 2\sum_{i=1}^{n}\ln(1-X_i) = 0$$

Prblem 5: Lin, Sutton, and Quarashi (1979) compared microbiological and hydroxylamine methods for analysis of ampicillin doses. In a series of experiments, 15 specimens were tested using the microbiological method (population 1), and then the hydroxylamine method (population 2). Using the dosage data recorded in ampicillin.csv, test the hypothesis

$$H_0 : \mu_1 - \mu_2 = 0, \quad H_a : \mu_1 - \mu_2 > 0$$

using significance level $\alpha = 0.05$. Show your test results for p-value, rejection region, and confidence interval. State your conclusion.

```
ex6 <- read.csv("ampicillin.csv")
par(mfrow=c(1,2))
qqnorm(ex6$microbiological)
qqline(ex6$microbiological)
qqnorm(ex6$hydroxylaminie)
qqline(ex6$hydroxylaminie)
```



As we observe both population are approximately normally distributed. We have $n_1 = n_2 = 15$ so they are dependent.

```
delta=ex6$microbiological-ex6$hydroxylaminie
x.bar=mean(delta)
x.bar
```

```
## [1] 0.44
```

```
s=sd(delta)
s
```

```
## [1] 4.630767
```

```
#test statistic.
t <- (x.bar)/s*sqrt(15)
t
```

```
## [1] 0.367998
```

```
p.value <- pt(t,14,lower.tail=F)    # p-value
p.value
```

```
## [1] 0.3591895
```

```
t.alpha <- qt(0.95,14)      # rejection region
t.alpha
```

```
## [1] 1.76131
```

```
ci <- x.bar-t.alpha*s/sqrt(15) # confidence interval
ci
```

```
## [1] -1.665926
```

Since p value, which is $0.3591895 > 0.05$, rejection interval $(1.76131, \infty)$ does not include t value which is 0.367998, and lower confidence interval $(-1.665926, \infty)$ includes zero mean, we fail to reject the null hypothesis.

Problem 6: An experiment is planned to compare the mean of a control group $\bar{X}$ to the mean of an independent sample of a group given a treatment $\bar{Y}$. Suppose there are 25 subjects in each group, the observations are are reasonably normally distributed, and the standard deviation of a single measurement in either group is $\sigma = 5$.

a) What is the standard error of $\bar{X} - \bar{Y}$.

We are given that the sample is independent and standard deviation of each one is equal. Using our note, this case matches with pooled variance. So,

```
Sp<-5
n1=25
n2=25
SE=Sp*sqrt((1/n1)+(1/n2))
SE
```

```
## [1] 1.414214
```

b) With a significance level of $\alpha = 0.05$ what is the rejection region of the test of the null hypothesis $H_0 : \mu_X = \mu_Y$ versus the alternative $H_a : \mu_X > \mu_Y$.

We firstly want mean difference to be zero and we need to find variance of $\bar{X} - \bar{Y}$, which we will find using the following

$$\sigma^2_{\bar{X}-\bar{Y}} = \frac{\sigma^2_{\bar{X}}}{n_1} + \frac{\sigma^2_{\bar{Y}}}{n_2}$$

where $n_1 = n_2 = 25$ and $\sigma_{\bar{X}} = \sigma_{\bar{Y}} = 5$.

```
sigma1=5
sigma2=5
sigma2.difference=(sigma1^2)/n1+(sigma2^2)/n2
sigma2.difference
```

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

So, we have $\bar{X} - \bar{Y} \sim \mathcal{N}(0, 2)$ and rejection region is computed as

```
rejection.region=qnorm(0.95, mean=0,sd=sqrt(sigma2.difference))
rejection.region
```

```
## [1] 2.326174
```

Rejection region is $(2.326174, \infty)$.

c) What is the power of the test if $\alpha = 0.05$ and $\mu_X = \mu_Y + 1$?

```
type2error=pnorm(rejection.region,mean=1, sd=sqrt(sigma2.difference))
Power=1-type2error
Power
```

```
## [1] 0.1741873
```

Problem 7: Nylon bars were tested for brittleness (Bennett and Franklin 1954). Each of 280 bars was molded under similar conditions and was tested in five places. Assuming that each bar has uniform composition, the number of breaks on a given bar should be binomially distributed with five trials and an unknown probability p of failure. If the bars are all of the same uniform strength, p should be the same for all of them. The following table summarizes the outcome of the experiment:

Breaks/Bar	Frequency
0	157
1	69
2	35
3	17
4	1
5	1

Under the given assumption, the data in the table consist of 280 observations of independent binomial random variables. Find the mle of p.

Each jˆth bar was tested in 5 different places. The probability that bar breaks at those places for each test is p_j. The number of breaks, X_j are binomially distributed, i.e. $X_j \sim Bin(n, p_j)$ where $n = 5$. We have the following probability mass function

$$b_{j,i} = \binom{5}{i} p_j^i (1 - p_j)^{5-i}, \quad 0 \leq i \leq 5$$

It represents the probability of jˆth bar being broken at i places. From the statement in the problem, we have a null hypothesis which is all p_j are same for observations $1 \leq j \leq 280$. As a result, $b_{j,i}$ are also same. Let η_i be a probability that a randomly selected will break at i places. We define

$$H_0 : \eta_i = \binom{5}{i} p^i (1 - p)^{5-i}, \quad 0 \leq i \leq 5$$

$$H_a : \eta_1 + \eta_2 + \eta_3 + \eta_4 + \eta_5 = 1.$$

From the table, we have $N_0 = 157, N_1 = 69, N_2 = 35, N_3 = 17, N_4 = 1, N_5 = 1$ Considering H_0, we have the following likelihood function

$$\frac{280!}{N_0!N_1!N_2!N_3!N_4!N_5!} \eta_0^{N_0} \eta_1^{N_1} \eta_2^{N_2} \eta_3^{N_3} \eta_4^{N_4} \eta_5^{N_5}$$

Then, log-likelihood function is

$$l(p) = \log(280!) - \sum_{i=0}^{5} log(N_i!) + \sum_{i=0}^{5} N_i \log(\eta_i)$$

$$l(p) = \log(280!) - \sum_{i=0}^{5} log(N_i!) + \sum_{i=0}^{5} N_i \left(\log \binom{5}{i} + i \log(p) + (5 - i) log(1 - p) \right)$$

We find derivatives,

$$l'(p) = \sum_{i=0}^{5} N_i \left(\frac{i}{p} - \frac{5-i}{1-p} \right)$$

$$l''(p) = \sum_{i=0}^{5} N_i \left(-\frac{i}{p^2} - \frac{5-i}{(1-p)^2} \right) < 0$$

So, $l(p)$ gets its maximum value at $\hat{p}$ where $l'(p) = 0$.

$$l'(p) = \sum_{i=0}^{5} N_i \left(\frac{i}{p} - \frac{5-i}{1-p} \right) = 0$$

$$\sum_{i=0}^{5} N_i \left(i(1-p) - (5-i)p \right) = 0$$

$$\sum_{i=0}^{5} N_i \left(i - 5p \right) = 0$$

$$\hat{p} = \frac{\sum_{i=0}^{5} i N_i}{5 \left(\sum_{i=1}^{5} N_i \right)}$$

```
p.hat=((0*157+1*69+2*35+3*17+4*1+5*1)/280)*0.2
p.hat
```

```
## [1] 0.1421429
```