

practice5

Qiang Liu

2022-10-04

```
> ## sampling times
> N <- 10000
> library(moments)
>
> ## mu function for sampling statisitics
> normSample <- function(n){
+ x <- unlist(lapply(1:N, function(i){
+ mean(rnorm(n))
+ })))
+
+ ## output
+ out <- c(mu = mean(x), s2 = var(x), g1 = skewness(x), g2 = kurtosis(x)-3)
+
+ ## histogram
+ hist(x, probability = T, breaks = 30, xlab = "", main = sprintf("n=%s",n))
+ ## legend
+ lgd <- sprintf("%s: %s", names(out), round(out, 5))
+ legend('topright', legend = lgd, ncol = 1, box.lwd = NA, cex = 0.5, title = "sample")
+ ## in theory
+ th <- c(mu=0, s2=1/n, g1=0, g2=0)
+ lgd2 <- sprintf("%s: %s", names(th), round(th, 5))
+ legend('topleft', legend = lgd2, ncol = 1, box.lwd = NA, cex = 0.5, text.col = 'red', title = "population")
+ px <- seq(-3,3,length.out=100)
+ py <- dnorm(px, mean = 0, sd = sqrt(1/n))
+ lines(px, py, col = "red", lwd = 2, lty = 5)
+ out
+ }
> op <- par(mfrow = c(2, 2))
> normSample(5)
```

```
##          mu          s2          g1          g2
## 0.004345735 0.204996568 0.013429907 0.099349392
```

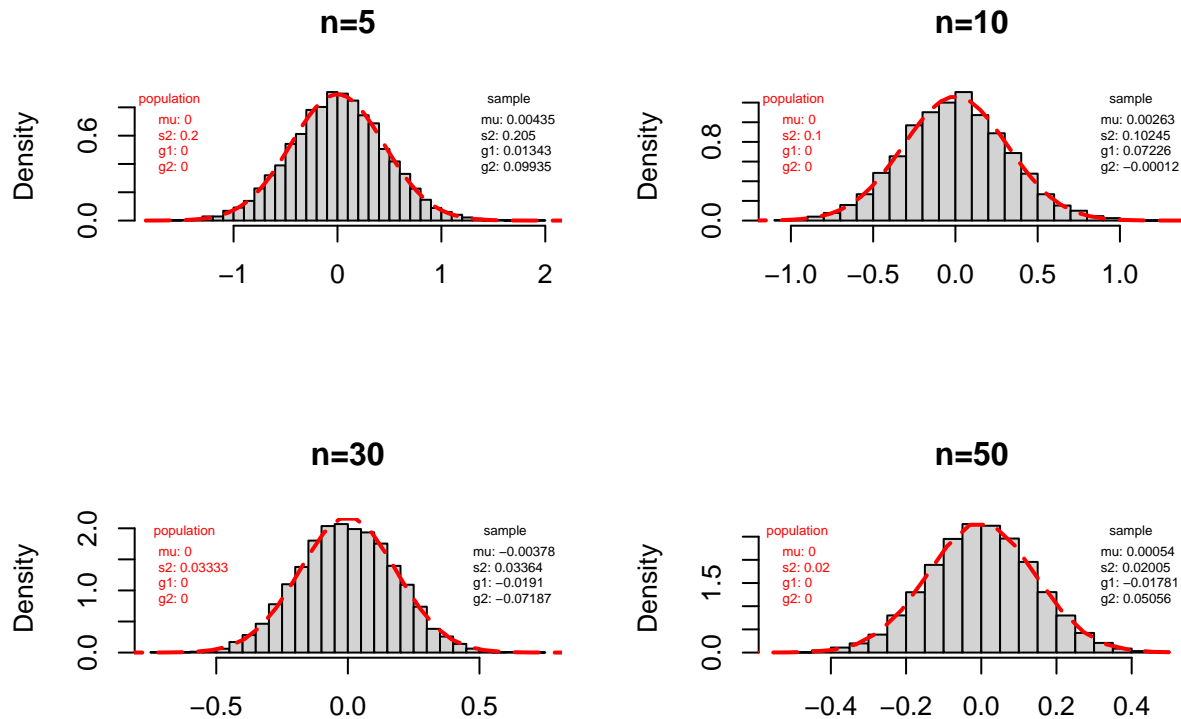
```
> normSample(10)
```

```
##          mu          s2          g1          g2
## 0.0026285704 0.1024466357 0.0722576475 -0.0001191087
```

```
> normSample(30)
```

```
##          mu          s2          g1          g2
## -0.003782377  0.033635820 -0.019098651 -0.071867960
```

```
> normSample(50)
```



```
##          mu          s2          g1          g2
##  0.000537082  0.020045941 -0.017808264  0.050556734
```

```
> par(op)
```

```
> N <- 10000
> library(moments)
> ## mu function for sampling statistics
> expSample <- function(n){
+ x <- unlist(lapply(1:N, function(i){
+ mean(rexp(n))
+ })))
+ 2
+ ## output
+ out <- c(mu = mean(x), s2 = var(x), g1 = skewness(x), g2 = kurtosis(x)-3)
+ ## histogram
+ hist(x, probability = T, breaks = 30, xlab = "", main=sprintf("n=%s",n))
```

```

+ ## legend
+ lgd <- sprintf("%s: %s", names(out), round(out, 5))
+ legend('topright', legend = lgd, ncol = 1, box.lwd = NA, cex = 0.5, title = "sample")
+ ## in theory
+ th <- c(mu = 0, s2 = 1, g1 = 0, g2 = 0)
+ lgd2 <- sprintf("%s: %s", names(th), round(th, 5))
+ legend('topleft', legend = lgd2, ncol = 1, box.lwd = NA, cex = 0.5, text.col = 'red', title = "population")
+ px <- seq(-3,3,length.out = 100)
+ py <- dnorm(px, mean = 1, sd = sqrt(1/n))
+ lines(px, py, col = "red", lwd = 2, lty = 5)
+ ## return
+ out
+ }
> op <- par(mfrow = c(2, 2))
> expSample(5)

```

```

##          mu          s2          g1          g2
## 1.0012160 0.1979380 0.8562142 1.0766576

```

```

> expSample(10)

```

```

##          mu          s2          g1          g2
## 1.00638335 0.09990011 0.63363871 0.59878829

```

```

> expSample(30)

```

```

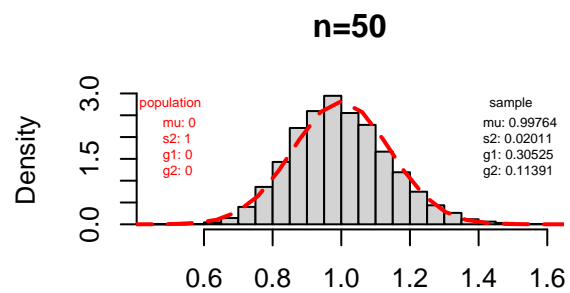
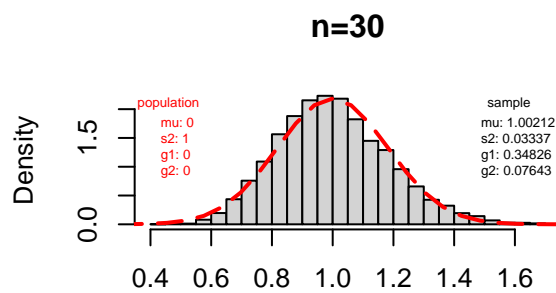
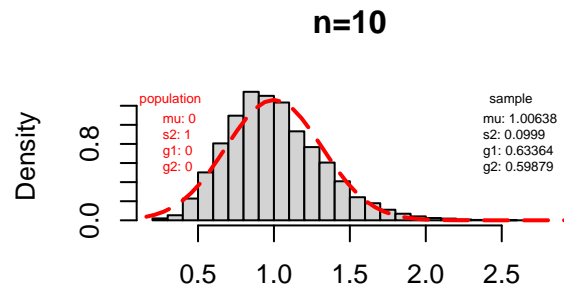
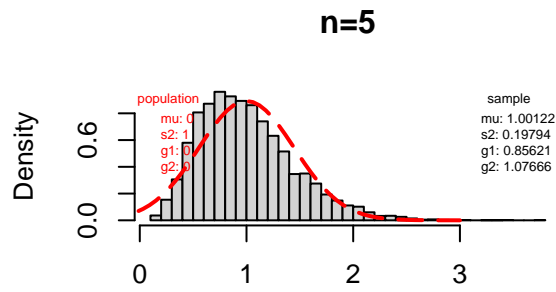
##          mu          s2          g1          g2
## 1.00211990 0.03337395 0.34826391 0.07642792

```

```

> expSample(50)

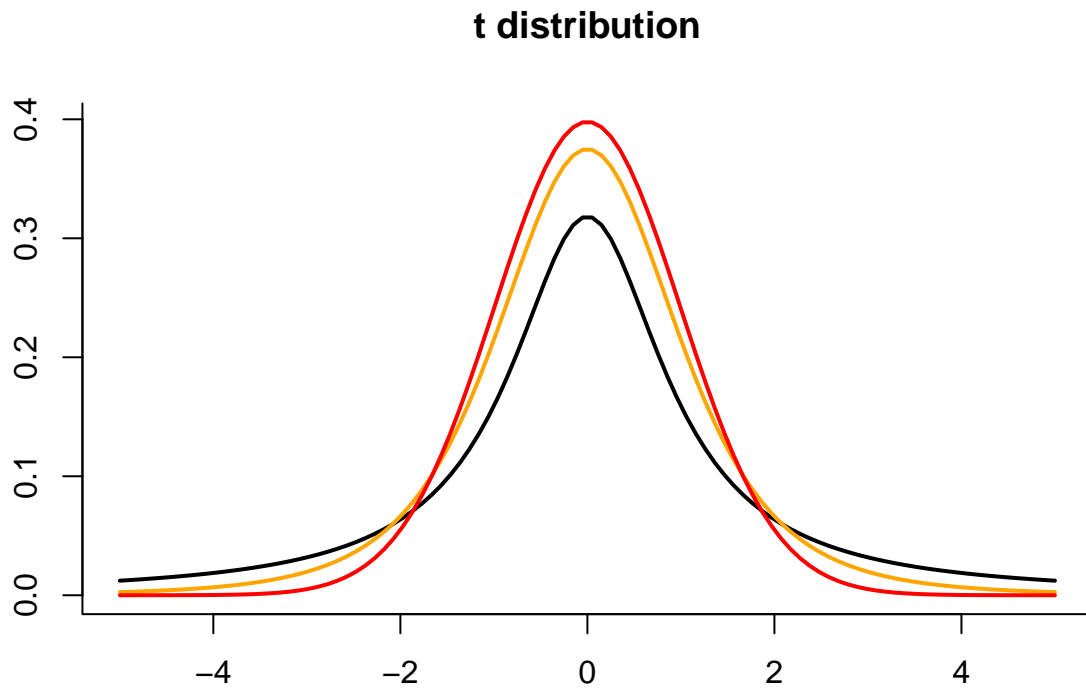
```



```
##      mu      s2      g1      g2
## 0.99763777 0.02011265 0.30525267 0.11391189
```

```
> par(op)
```

```
> x <- seq(-5, 5, length.out = 100)
> df1 <- dt(x, df = 1)
> df4 <- dt(x, df = 4)
> df100 <- dt(x, df = 100)
> plot(x, df1, ylim = range(c(df1, df4, df100)), bty = "l",
+ type = "l", lwd = 2, xlab = "", ylab = "", main = "t distribution")
> lines(x, df4, col = "orange", lwd = 2)
> lines(x, df100, col = "red", lwd = 2)
```



```

> ## sampling times
> N <- 10000
> library(moments)
> ## mu function for sampling statistics
> tSample <- function(n){
+ x <- unlist(lapply(1:N, function(i){
+ o <- rnorm(n)
+ mean(o) / (sd(o)/sqrt(n))
+ })))
+ ## output
+ out <- c(mu=mean(x),s2=var(x),g1=skewness(x),g2=kurtosis(x)-3)
+ ## histogram
+ hist(x, probability = T, breaks = 30, xlab = "", main=sprintf("df=%s",n))
+ ## legend
+ lgd <- sprintf("%s: %s", names(out), round(out, 5))
+ legend('topright', legend = lgd, ncol = 1, box.lwd = NA, cex = 0.5, title = "sample")
+ ## in theory
+ th <- c(mu=0, s2=n/(n-2), g1=0, g2=6/(n-4))
+ lgd2 <- sprintf("%s: %s", names(th), round(th, 5))
+ legend('topleft', legend = lgd2, ncol = 1, box.lwd = NA, cex = 0.5, text.col = 'red', title = "population")
+ px <- seq(-3,3,length.out=100)
+ py <- dt(px, df = n)
+ lines(px, py, col="red", lwd=2, lty=5)

```

```

+ ## return
+ out
+ }
> op <- par(mfrow=c(2, 2))
> tSample(5)

```

```

##          mu          s2          g1          g2
## 0.009620651 1.934760278 -0.241757270 6.099616144

```

```

> tSample(10)

```

```

##          mu          s2          g1          g2
## -0.0220168 1.3375886 -0.1690316 1.8991228

```

```

> tSample(20)

```

```

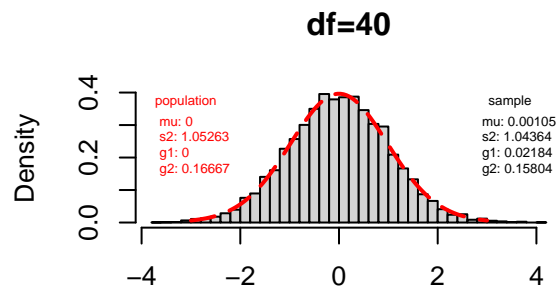
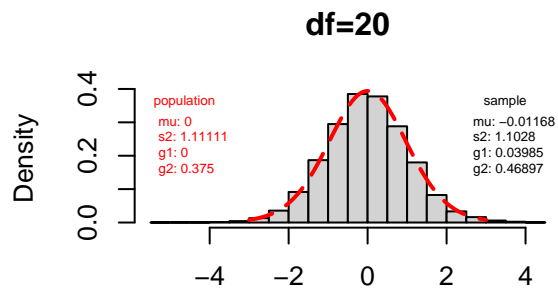
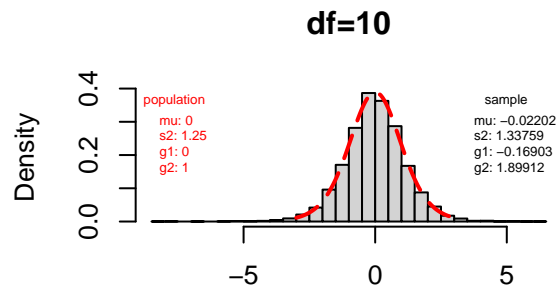
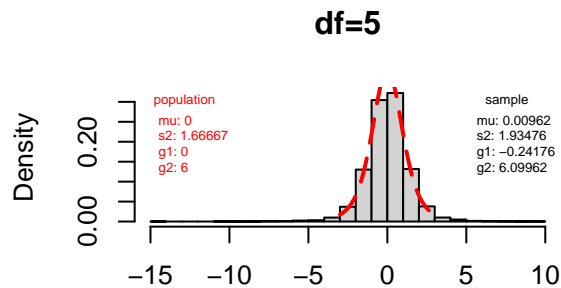
##          mu          s2          g1          g2
## -0.01168138 1.10280387 0.03984733 0.46897160

```

```

> tSample(40)

```



```

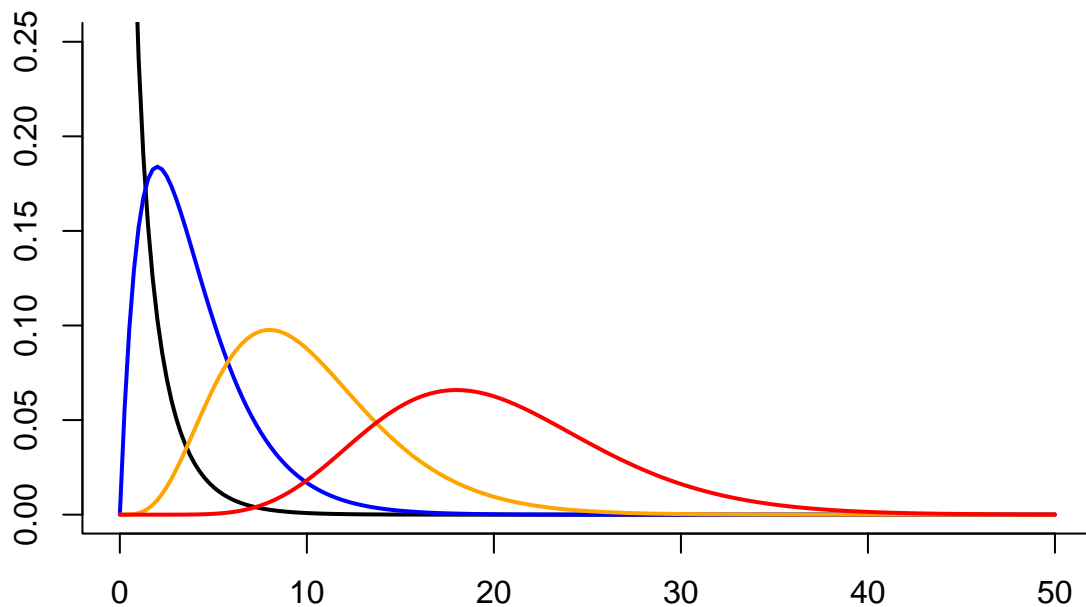
##          mu          s2          g1          g2
## 0.001052543 1.043639230 0.021843582 0.158038404

```

```
> par(op)
```

```
> x <- seq(0, 50, length.out = 200)
> df1 <- dchisq(x, df=1)
> df4 <- dchisq(x, df=4)
> df10 <- dchisq(x, df=10)
> df20 <- dchisq(x, df=20)
> df1[is.infinite(df1)] <- NA
> plot(x, df1, ylim = c(0, 0.25), bty = "l",
+ type = "l", lwd = 2, xlab = "", ylab = "", main = "Chi-Squared distribution")
> lines(x, df4, col = "blue", lwd = 2)
> lines(x, df10, col = "orange", lwd = 2)
> lines(x, df20, col = "red", lwd = 2)
```

Chi-Squared distribution

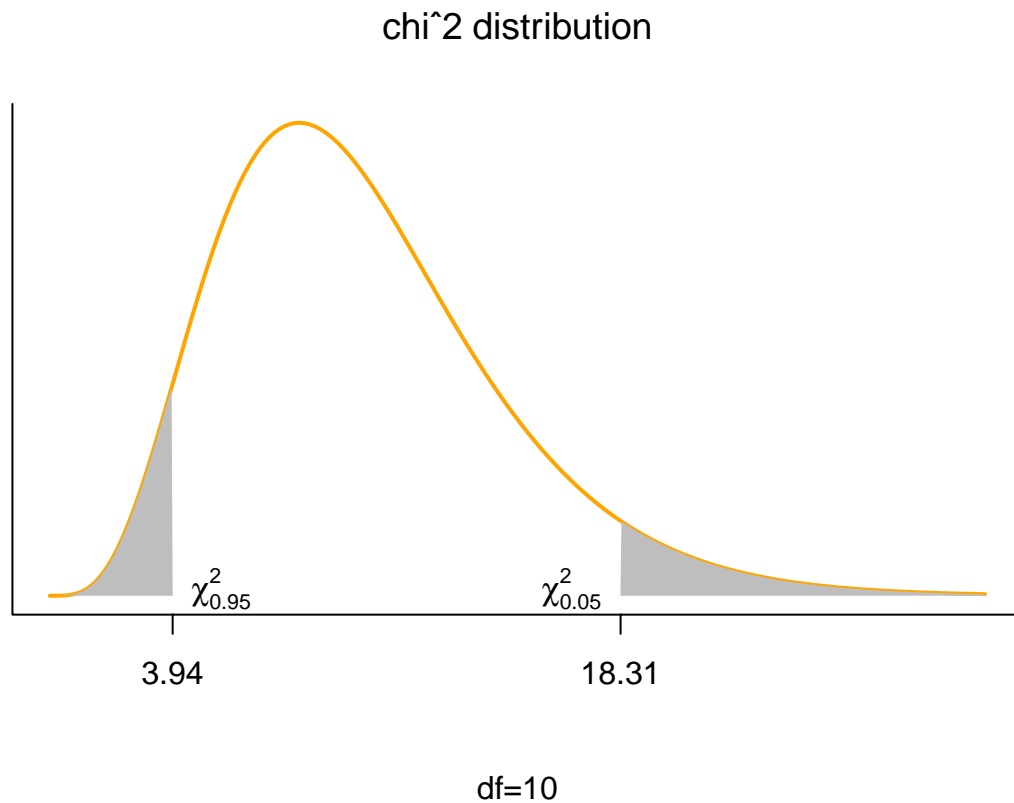


```
> from <- 0
> to <- 30
> df <- 10
> x <- seq(from, to, length.out = 500)
> df10 <- dchisq(x, df=df)
> plot(x, df10, bty = "l", col = "orange", yaxt='n', xaxt='n',
```

```

+ type = "l", lwd = 2, xlab = "", ylab = "",
+ main = expression(chi^2~"distribution"), sub = sprintf("df=%s", df))
> x0 <- qchisq(0.05, df=df, lower.tail=F)
> x1 <- x[x > x0]
> y1 <- dchisq(x1, df=df)
> polygon(c(x0,x1,to), c(0,y1,0), border = NA, col = "grey")
> text(x0, 0, labels = expression(chi[0.05]^2), pos=2, xpd=T)
> axis(1, at=round(x0, 2))
> x0 <- qchisq(0.05, df=df, lower.tail=T)
> x2 <- x[x < x0]
> y2 <- dchisq(x2, df=df)
> polygon(c(from,x2,x0), c(0,y2,0), border = NA, col = "grey")
> text(x0, 0, labels = expression(chi[0.95]^2), pos=4, xpd=T)
> axis(1, at=round(x0, 2))

```



```

> ## sampling times
> N <- 10000
> library(moments)
> ## mu function for sampling statisitics
> chisqSample <- function(n){
+ x <- unlist(lapply(1:N, function(i){
+ o <- rnorm(n)
+ n*var(o)

```



```

+ )))
+ ## output
+ out <- c(mu=mean(x),s2=var(x),g1=skewness(x),g2=kurtosis(x)-3)
+ ## histogram
+ h <- hist(x, probability = T, breaks = 30, xlab = "", main=sprintf("df=%s",n))
+ ## legend
+ lgd <- sprintf("%s: %s", names(out), round(out, 5))
+ legend('topright', legend = lgd, ncol = 1, box.lwd = NA, cex = 0.5, title = "sample")
+ ## in theory
+ th <- c(mu=n, s2=n*2, g1=sqrt(8/n), g2=12/n)
+ lgd2 <- sprintf("%s: %s", names(th), round(th, 5))
+ legend('topleft', legend = lgd2, ncol = 1, box.lwd = NA, cex = 0.5, text.col = 'red', title = "population")
+ px <- seq(min(h$mids),max(h$mids),length.out=100)
+ py <- dchisq(px, df = n)
+ lines(px, py, col="red", lwd=2, lty=5)
+ ## return
+ out
+ }
> op <- par(mfrow=c(2, 2))
> chisqSample(2)

```

```

##          mu          s2          g1          g2
## 2.043184  8.504342  2.817582 11.288735

```

```
> chisqSample(10)
```

```

##          mu          s2          g1          g2
## 9.989683 22.935252  1.030904  1.808076

```

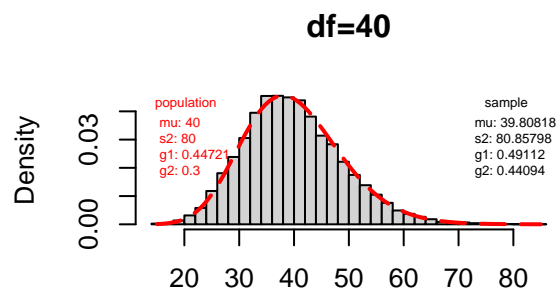
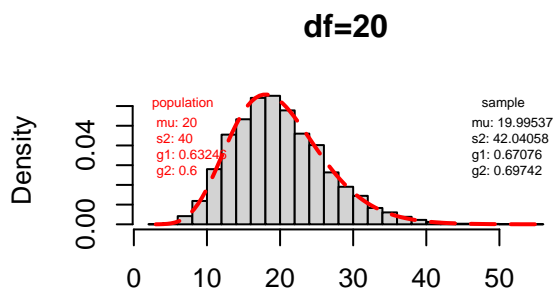
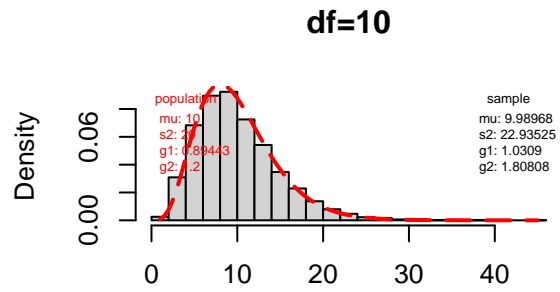
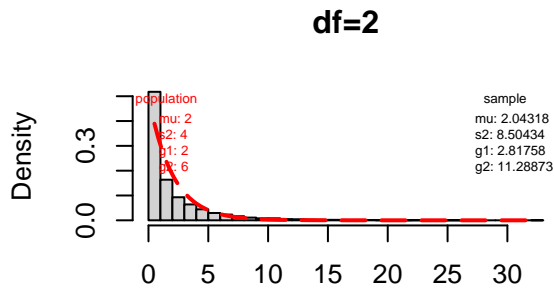
```
> chisqSample(20)
```

```

##          mu          s2          g1          g2
## 19.995369 42.040581  0.670763  0.697423

```

```
> chisqSample(40)
```



```
##          mu          s2          g1          g2
## 39.8081817 80.8579845  0.4911232  0.4409428
```

```
> par(op)
```

```
> ## sampling times
> N <- 10000
> library(moments)
> ## mu function for sampling statistics
> chisqSample2 <- function(n){
+ x <- unlist(lapply(1:N, function(i){
+ o <- rexp(n)
+ n*var(o)
+ })))
+ ## output
+ out <- c(mu=mean(x), s2=var(x), g1=skewness(x), g2=kurtosis(x)-3)
+ ## histogram
+ h <- hist(x, probability = T, breaks = 30, xlab = "", main=sprintf("df=%s",n))
+ ## legend
+ lgd <- sprintf("%s: %s", names(out), round(out, 5))
+ legend('topright', legend = lgd, ncol = 1, box.lwd = NA, cex = 0.5, title = "sample")
+ ## in theory
+ th <- c(mu=n, s2=n*2, g1=sqrt(8/n), g2=12/n)
```

```

+ lgd2 <- sprintf("%s: %s", names(th), round(th, 5))
+ legend('top', legend = lgd2, ncol = 1, box.lwd = NA, cex = 0.5, text.col = 'red', title = "population")
+ px <- seq(min(h$mids), max(h$mids), length.out=100)
+ py <- dchisq(px, df = n)
+ lines(px, py, col="red", lwd=2, lty=5)
+ ## return
+ out
+ }
> op <- par(mfrow=c(2, 2))
> chisqSample2(2)

```

```

##      mu      s2      g1      g2
## 1.941880 19.227561 6.627902 77.083125

```

```
> chisqSample2(10)
```

```

##      mu      s2      g1      g2
## 10.000443 83.527792 2.977479 15.612505

```

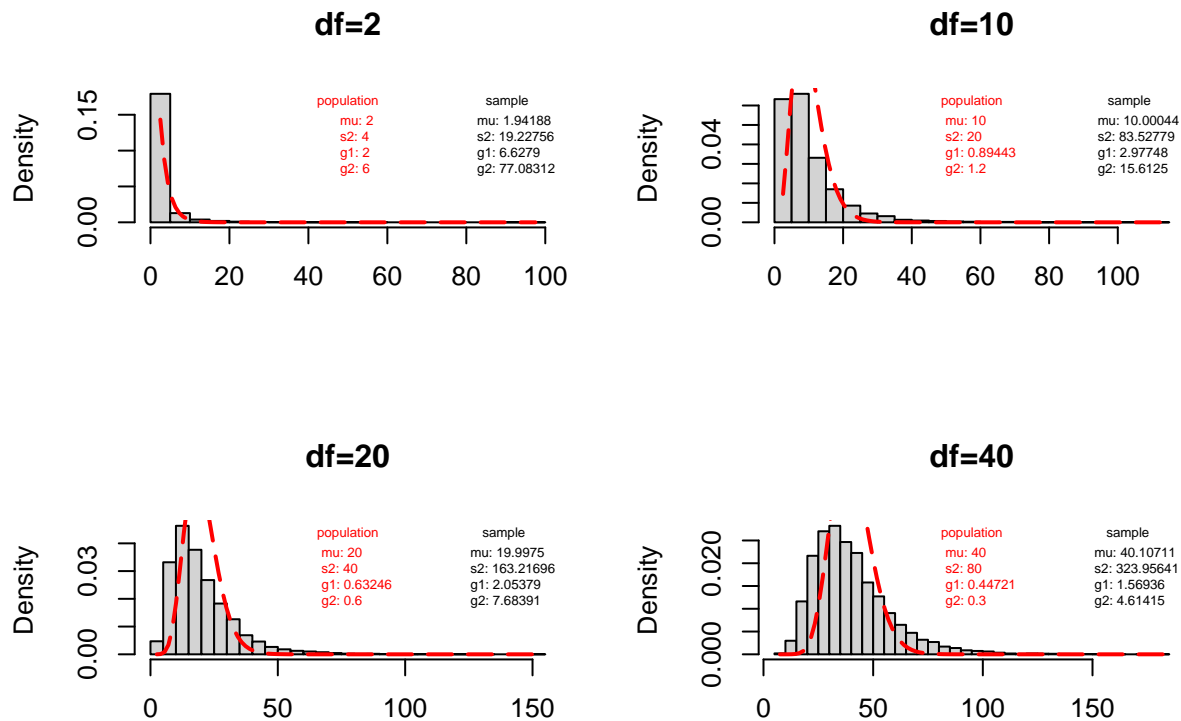
```
> chisqSample2(20)
```

```

##      mu      s2      g1      g2
## 19.997501 163.216958 2.053789 7.683915

```

```
> chisqSample2(40)
```



```
##          mu          s2          g1          g2
## 40.107110 323.956410  1.569364  4.614155
```

```
> par(op)
```

```
> x <- seq(0, 5, length.out = 100)
> df1 <- df(x, df1 = 4, df2 = 20)
> df2 <- df(x, df1 = 20, df2 = 4)
> plot(x, df1, ylim = c(0, 0.8), bty = "l", col = "orange",
+ type = "l", lwd = 2, xlab = "", ylab = "", main = "F distribution")
> lines(x, df2, col = "blue", lwd = 2)
```

