Namburu_Setu

Skewness and kurtosis statistics are used to assess normality or lack thereof during EDA. These statistics are affected by distributional shape and sample size. This exercise will investigate how variable the skewness statistic is when sampling from a standard normal distribution. The results will provide a baseline for evaluating data samples in the future. The skewness for the standard normal distribution is 0.0. Samples from a normal distribution will depart from this number, sometimes substantially.

Problem Statement (10 points)

Load the "moments" package which supplies *skewness()* and *kurtosis()* functions. Use three different samples size: 10, 40 and 100. For each sample size, draw 1000 random samples, and compute the skewness for each. Use *par(mfrow = c(3, 1))* and present histograms for these results. Compute the 2.5% (0.025) and 97.5% (0.975) quantiles, and the first and third quartiles for each histogram and present.

The function defined below may be used to generate the samples for a given *n*. This function produces a 1000 x *n* dimensional array. Use *apply()* to compute the skewness for each row.

```
example <- function(n){
  set.seed(123)
  result <- matrix(0, nrow = 1000, ncol = n)
  for (k in 1:1000) {
    result[k, ] <- rnorm(n, mean = 0, sd = 1)
  }
  return(result)
}

# We can define our 1000 x n matrices by passing our desired n to example(). For example, for
# n = 10:

sample10 <- example(10)


# You will need to add code here for creating the same objects with n = 40 and n = 100.
sample40 <- example(40)
sample100 <- example(100)
```

In the following code 'chunk,' you will need to add code passing each 1000 x *n* matrix to an instance of *apply()*. Your matrices will be passed to *apply()* as X. You must specify MARGIN = 1 and FUN = skewness to have skewness calculated for each row; MARGIN = 2 would calculate the skewness for each column. You will then present the vectors of skewness results for n = 10, 40 and 100 as histograms and determine the 2.5% (0.025) and 97.5% (0.975) quantiles, and the first and third quartiles for each histogram and present. Use *quantile()*.

```
library(moments)  # install.packages("moments")

skewness10 <- apply(sample10, MARGIN = 1, FUN = skewness)
quantile10 <- quantile(skewness10,c(0.025,0.25,0.75,0.975))
cat("(0.025,0.25,0.75,0.975) quantiles of skewness statistic computed from 1000 normal random
samples of size n=10 respectively: ", quantile10)
```

```
## (0.025,0.25,0.75,0.975) quantiles of skewness statistic computed from 1000 normal random sa
mples of size n=10 respectively:  -1.075816 -0.3386289 0.3627693 1.10436
```

```
skewness40 <- apply(sample40, MARGIN = 1, FUN = skewness)
quantile40 <- quantile(skewness40,c(0.025,0.25,0.75,0.975))
cat("\n\n(0.025,0.25,0.75,0.975) quantiles of skewness statistic computed from 1000 normal ran
dom samples of size n=40 respectively: ", quantile40)
```

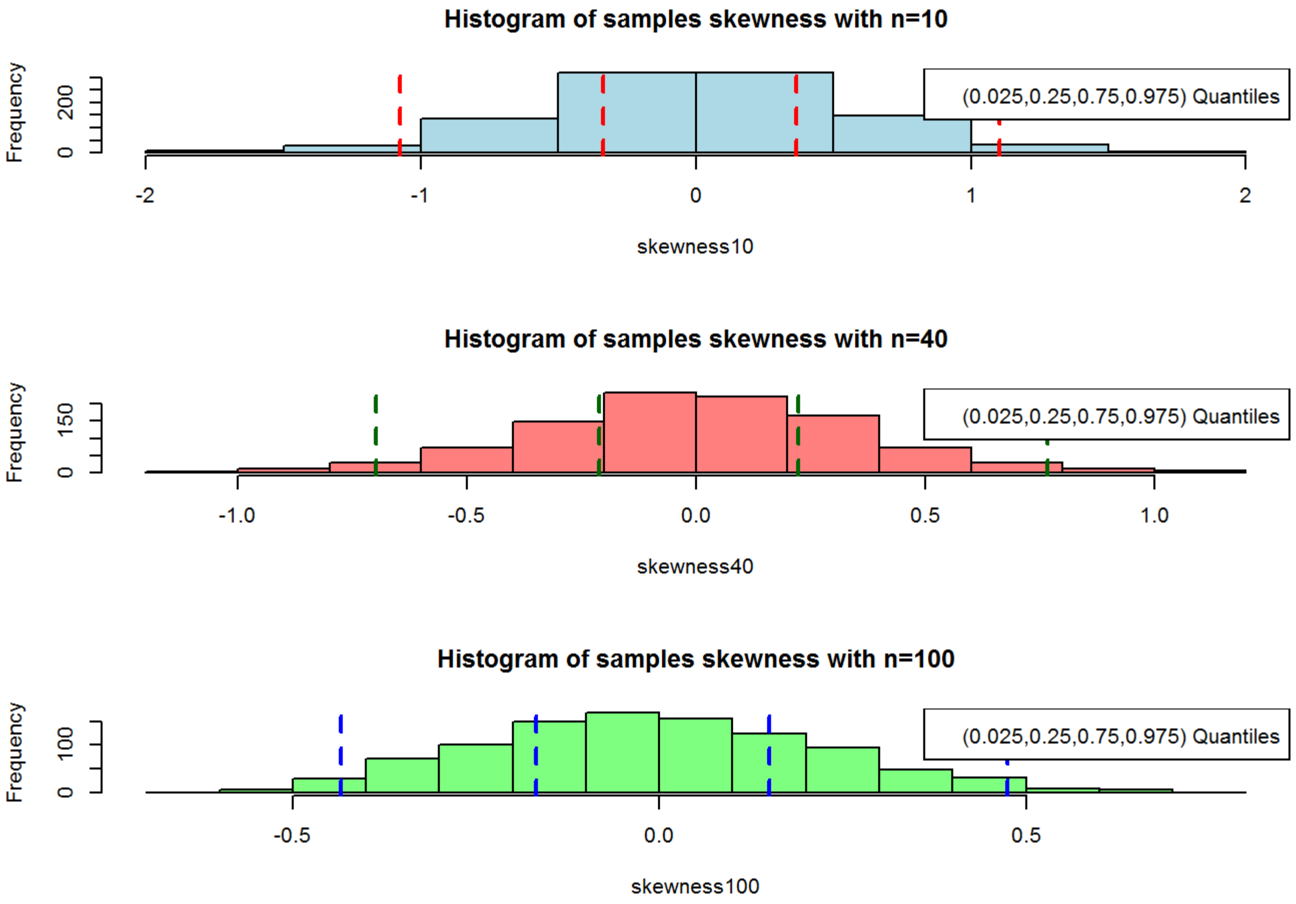
```
##
##
## (0.025,0.25,0.75,0.975) quantiles of skewness statistic computed from 1000 normal random sa
mples of size n=40 respectively:  -0.6986338 -0.2113579 0.2238934 0.7661049
```

```
skewness100 <- apply(sample100, MARGIN = 1, FUN = skewness)
quantile100 <- quantile(skewness100,c(0.025,0.25,0.75,0.975))
cat("\n\n(0.025,0.25,0.75,0.975) quantiles of skewness statistic computed from 1000 normal ran
dom samples of size n=100 respectively: ", quantile100)
```

```
##
##
## (0.025,0.25,0.75,0.975) quantiles of skewness statistic computed from 1000 normal random sa
mples of size n=100 respectively:  -0.4349885 -0.1682871 0.150487 0.474412
```

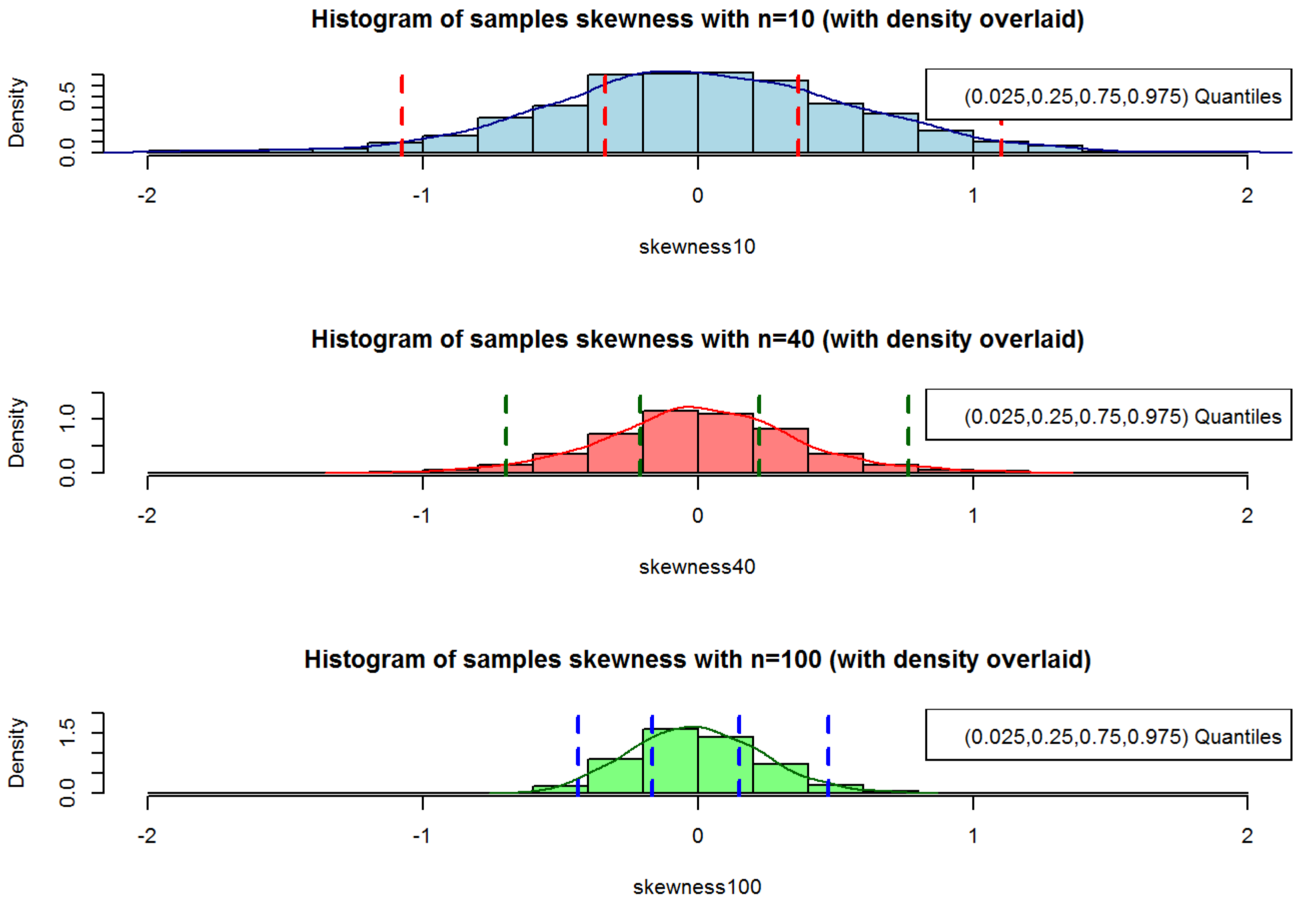
You did the work correctly. I would ask that next time for assignments like this to show your numerical results in a table or in summary form.

```
###Basic histograms

par(mfrow = c(3, 1))

hist(skewness10, col = "lightblue", main = "Histogram of samples skewness with n=10")
abline(v = quantile10, col = "red",lty=2, lwd=2)
#text(quantile10,rep(200,length=4),c("2.5% \nquantile","1st \nquartile","3rd \nquartile","97.5
% \nquantile"), adj=c(-0.1,1),col="red")
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")

hist(skewness40,col = rgb(1,0,0,0.5),  main = "Histogram of samples skewness with n=40")
abline(v = quantile40, col = "darkgreen",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")

hist(skewness100,col = rgb(0,1,0,0.5),  main = "Histogram of samples skewness with n=100")
abline(v = quantile100, col = "blue",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")
```



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

###Histograms with density overlaid (kernal density estimation)

par(mfrow = c(3, 1))


cells <- seq(-2,2,0.2)

hist(skewness10,col = "lightblue", probability = T, breaks = cells, main = "Histogram of sampl
es skewness with n=10 (with density overlaid)")
points(density(skewness10),type="l",col="darkblue")
abline(v = quantile10, col = "red",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")




hist(skewness40,col = rgb(1,0,0,0.5), probability = T, breaks = cells,ylim = c(0.0,1.5), main
= "Histogram of samples skewness with n=40 (with density overlaid)")
points(density(skewness40),type="l",col="red")
abline(v = quantile40, col = "darkgreen",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")

hist(skewness100,col = rgb(0,1,0,0.5), probability = T, breaks = cells,ylim = c(0.0,2.0), main
 = "Histogram of samples skewness with n=100 (with density overlaid)")
points(density(skewness100),type="l",col="darkgreen")
```

```
abline(v = quantile100, col = "blue",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")
```



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

Additional (5 points)

Additional credit (5 points) will be provided if results are also shown for kurtosis. Follow the instructions shown above for skewness.) Kurtosis for a standard normal distribution is 3.0 using the “moments” package *kurtosis()* function. Sample values will depart from this number, sometimes substantially.

```
library(moments)  # install.packages("moments")

kurtosis10 <- apply(sample10, MARGIN = 1, FUN = kurtosis)
quantile10 <- quantile(kurtosis10,c(0.025,0.25,0.75,0.975))
cat("(0.025,0.25,0.75,0.975) quantiles of kurtosis statistic computed from 1000 normal random
samples of size n=10 respectively: ", quantile10)
```

```
## (0.025,0.25,0.75,0.975) quantiles of kurtosis statistic computed from 1000 normal random sa
mples of size n=10 respectively:  1.50665 1.909665 2.797319 4.276703
```

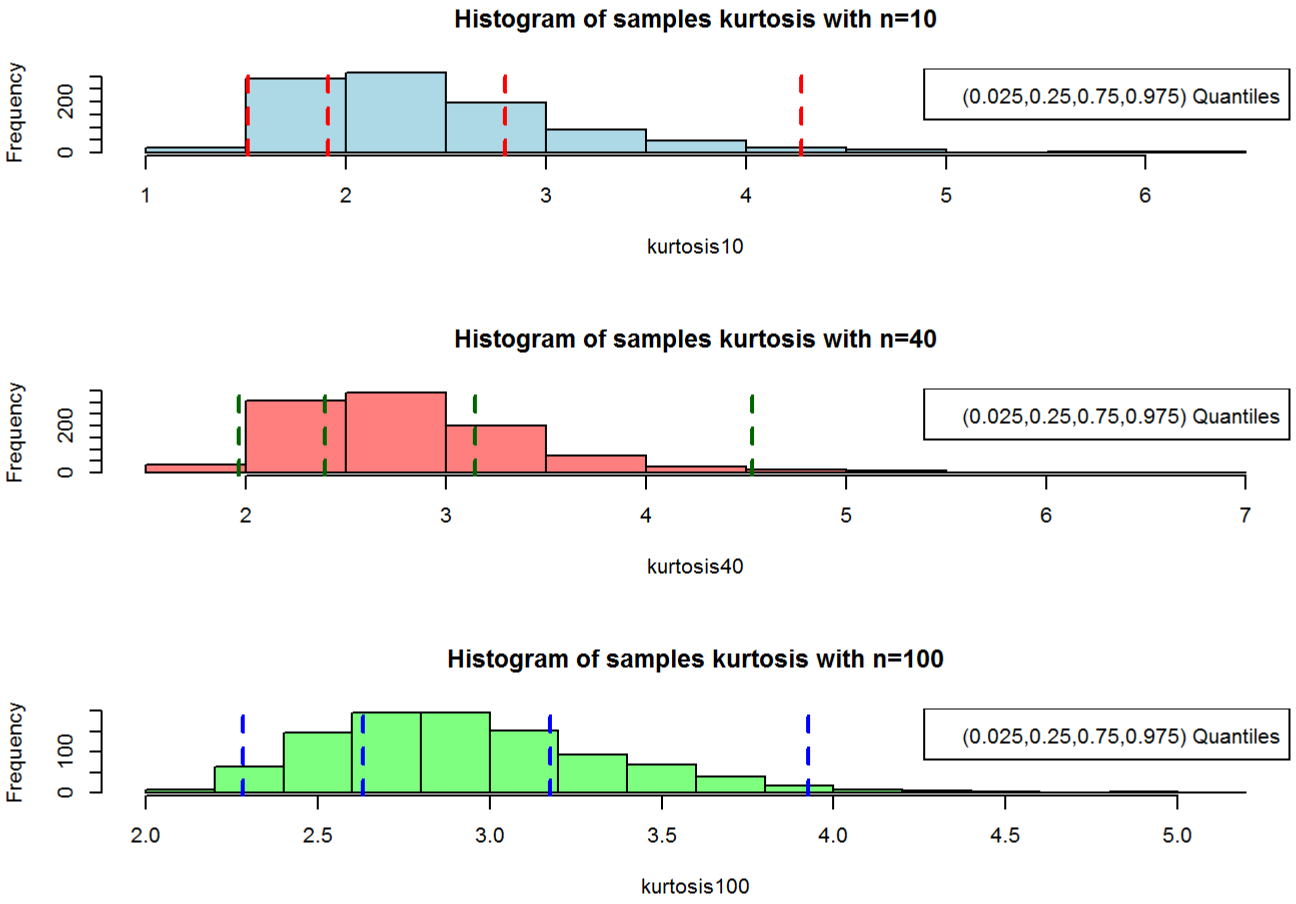
```
kurtosis40 <- apply(sample40, MARGIN = 1, FUN = kurtosis)
quantile40 <- quantile(kurtosis40,c(0.025,0.25,0.75,0.975))
cat("\n\n(0.025,0.25,0.75,0.975) quantiles of kurtosis statistic computed from 1000 normal ran
dom samples of size n=40 respectively: ", quantile40)
```

```
##
##
## (0.025,0.25,0.75,0.975) quantiles of kurtosis statistic computed from 1000 normal random sa
mples of size n=40 respectively:  1.962334 2.396368 3.146233 4.532775
```

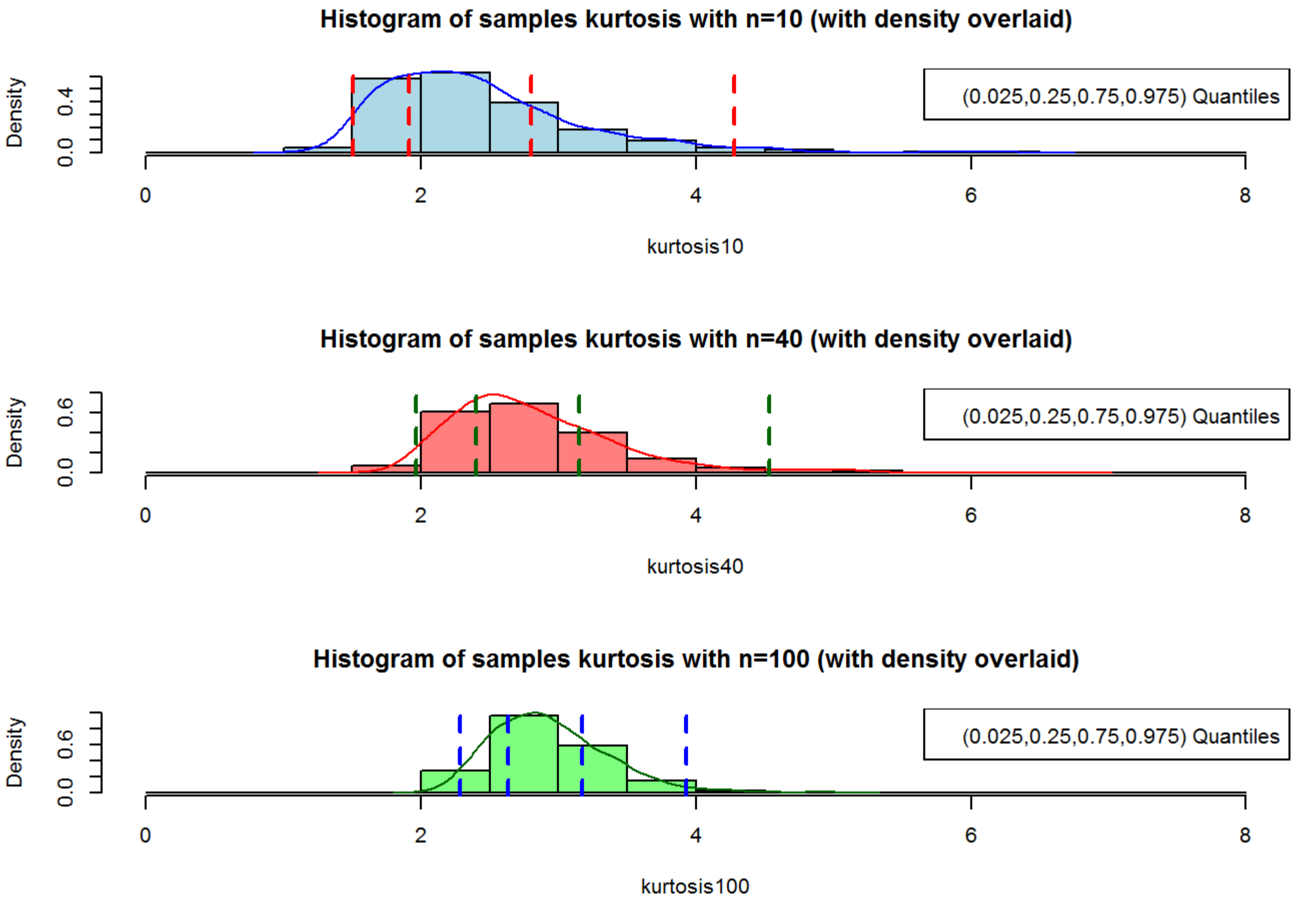
```
kurtosis100 <- apply(sample100, MARGIN = 1, FUN = kurtosis)
quantile100 <- quantile(kurtosis100,c(0.025,0.25,0.75,0.975))
cat("\n\n(0.025,0.25,0.75,0.975) quantiles of kurtosis statistic computed from 1000 normal ran
dom samples of size n=100 respectively: ", quantile100)
```

```
##
##
## (0.025,0.25,0.75,0.975) quantiles of kurtosis statistic computed from 1000 normal random sa
mples of size n=100 respectively:  2.280478 2.629944 3.174297 3.926302
```

```
par(mfrow = c(3, 1))

hist(kurtosis10, col = "lightblue", main = "Histogram of samples kurtosis with n=10")
abline(v = quantile10, col = "red",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")


hist(kurtosis40,col = rgb(1,0,0,0.5),  main = "Histogram of samples kurtosis with n=40")
abline(v = quantile40, col = "darkgreen",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")


hist(kurtosis100,col = rgb(0,1,0,0.5),  main = "Histogram of samples kurtosis with n=100")
abline(v = quantile100, col = "blue",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")
```



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

###Histograms with density overlaid (Using kernal density estimation)

par(mfrow = c(3, 1))

cells <- seq(0,8,0.5)

hist(kurtosis10,col = "lightblue", probability = T, breaks = cells, main = "Histogram of sampl
es kurtosis with n=10 (with density overlaid)")
points(density(kurtosis10),type="l",col="blue")
abline(v = quantile10, col = "red",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")

hist(kurtosis40,col = rgb(1,0,0,0.5), probability = T, breaks = cells,ylim=c(0.0,0.8), main =
"Histogram of samples kurtosis with n=40 (with density overlaid)")
points(density(kurtosis40),type="l",col="red")
abline(v = quantile40, col = "darkgreen",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")

hist(kurtosis100,col = rgb(0,1,0,0.5), probability = T, breaks = cells,ylim=c(0.0,1.0), main =
 "Histogram of samples kurtosis with n=100 (with density overlaid)")
points(density(kurtosis100),type="l",col="darkgreen")
abline(v = quantile100, col = "blue",lty=2, lwd=2)
legend("topright", legend="(0.025,0.25,0.75,0.975) Quantiles")
```



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

15 points You did extensive work, actually more than necessary. Please show the results of calculations such as these in a summary form as mentioned above.