Review Questions

Madelyn Caufield

1) $E(\bar{y}) = E\,\frac{1}{4}(Y_1+Y_2+Y_3+Y_4)$

$E(\bar{y}) = \frac{1}{4}[E(Y_1)+E(Y_2)+E(Y_3)+E(Y_4))$

$E(\bar{y}) = \frac{1}{4} \times 4\mu$

$E(\bar{y}) = \mu$

$Var(\bar{y}) = var[\frac{1}{4}(Y_1+Y_2+Y_3+Y_4)]$

$Var(\bar{Y}) = \frac{1}{16} var(Y_1+Y_2+Y_3+Y_4)$

$var(\bar{Y}) = \frac{1}{16} \times 4\sigma^2$

$Var(\bar{Y}) = 0.25\sigma^2$

$E[W_1] = \left(\frac{1}{8}+\frac{1}{8}+\frac{1}{4}+\frac{1}{2}\right)\mu = \mu$

$E[W_1] = \mu$

$Var(W_1) = \left(\frac{1}{64}+\frac{1}{64}+\frac{1}{16}+\frac{1}{4}\right)\sigma^2 = \frac{11}{32}\sigma^2$

$Var(W_1) = \frac{11}{32}\sigma^2$

Since both $E[\bar{y}] = E[W_1] = \mu$, the best estimator is the one with less standard deviations. $\bar{Y}$ is the better estimator.

2) $W_a = \sum_{i=1}^{n} a_i Y_i$ $\quad E[Y_i] = \mu$

$var(Y_i) = \sigma^2$

$E[W_a] = E\left(\sum_{i=1}^{n} a_i Y_i\right) = \sum_{i=1}^{n} a_i E[Y_i] = \mu \sum_{i=1}^{n} a_i$

For W_a to be an unbiased estimator of μ,

$E[W_a] = \mu \longrightarrow \sum_{i=1}^{n} a_i = 1$

$Var(W_a) = var\left(\sum_{i=1}^{n} a_i Y_i\right) = \sum_{i=1}^{n} a_i^2\, var(Y_i) + \sum\sum_{i \neq j} a_i a_j\, cov(Y_i, Y_j)$

$var(W_a) = \sum_{i=1}^{n} a_i^2 \sigma^2 = \sigma^2 \sum_{i=1}^{n} a_i^2$

$$\sum_{i=1}^{n} a_i = 1$$

$$\left(\sum_{i=1}^{n} a_i\right)^2 \le n \sum_{i=1}^{n} a_i^2$$

$$\sum_{i=1}^{n} a_i^2 \ge \frac{1}{n}$$

least possible value of $\sum_{i=1}^{n} a_i^2 = \frac{1}{n}$

$\min var(w_a) = \frac{\sigma^2}{n}$, for $y = \frac{1}{n}\sum_{i=1}^{n} y_i$, $a_i^* = \frac{1}{n}$, $i = 1, 2, .., n$

$$\sum_{i=1}^{n} a_i^{*2} = \sum_{i=1}^{n} \frac{1}{n^2} = \frac{n}{n^2} = \frac{1}{n}$$

$\hat{y}$ is BLUE with variance $\frac{\sigma^2}{n}$

3) $\left[\hat{\mu} - 1.96 \cdot \frac{\sigma}{\sqrt{T}}, \hat{\mu} + 1.96 \cdot \frac{\sigma}{\sqrt{T}}\right]$ $\left[\sigma^2 - 1.96 \cdot \frac{\sqrt{2}\sigma^2}{\sqrt{T}}, \sigma^2 + 1.96 \cdot \frac{\sqrt{2}\sigma^2}{\sqrt{T}}\right]$

GS: $\left[0.01 - 1.96 \cdot \frac{0.1}{\sqrt{100}}, 0.01 + 1.96 \cdot \frac{0.1}{\sqrt{100}}\right]$

GS CI for μ_i: $[-0.0096, 0.0296]$

GS: $\left[0.01 - 1.96 \cdot \frac{\sqrt{2}\, 0.01}{\sqrt{100}}, 0.01 + 1.96 \cdot \frac{\sqrt{2}\, 0.01}{\sqrt{100}}\right]$

GS CI for σ_i^2: $[0.00722, 0.01277]$

AIG: $\left[-0.03 - 1.96 \cdot \frac{0.3}{\sqrt{100}}, -0.03 + 1.96 \cdot \frac{0.3}{\sqrt{100}}\right]$

AIG CI for μ_i: $[-0.0888, 0.0288]$

AIG: $\left[0.009 - 1.96 \cdot \frac{\sqrt{2}\, 0.009}{\sqrt{T}}, 0.009 + 1.96 \cdot \frac{\sqrt{2}\, 0.009}{\sqrt{T}}\right]$

AIG CI for σ_i^2: $[0.00650, 0.01149]$

$[\hat{\theta} - 2 \cdot \hat{SE}(\hat{\theta}), \hat{\theta} + 2 \cdot \hat{SE}(\hat{\theta})]$

$\left[0.01 - 2\sqrt{\frac{1-0.4}{100}}, 0.01 + 2\sqrt{\frac{1-0.4}{100}}\right]$

CI for ρ_{12}: $[-0.14491, 0.16491]$

$H_0: \mu = 0$
$H_1: \mu \neq 0$

$$Z_1 = \frac{\bar{X} - \mu}{\sigma / \sqrt{T}} = \frac{0.01 - 0}{0.1 / \sqrt{100}} = 1$$

Z_1 is inside $[-1.96, 1.96]$ so we fail to reject H_0.

$$Z_2 = \frac{-0.03 - 0}{0.3 / \sqrt{100}} = -1$$

Z_2 is inside $[-1.96, 1.96]$ so we fail to reject H_0.

$H_0: \mu_1 = \mu_2$
$H_1: \mu_1 \neq \mu_2$

$\bar{X}_1 = 0.01$ $\quad \sigma_1 = 0.1$
$\bar{X}_2 = -0.03$ $\quad \sigma_2 = 0.3$

$$Z = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\sigma_1^2 / T} + \sqrt{\sigma_2^2 / T}} = \frac{0.01 + 0.03}{0.1/_T + 0.3/_T} = 1$$

$|Z| = 1 < 1.96$, do not reject $H_0: \mu_1 = \mu_2$

$H_0: \sigma_i^2 = 0.0225$
$H_1: \sigma_i^2 \neq 0.0225$

$1 - \alpha = .05$
$.05 - 1 = -\alpha$
$\alpha = 0.95$

$$Z_1 = \frac{0.01 - 0.0225}{0.1 / \sqrt{100}} = -1.25$$

Z_1 is between $[-1.96, 1.96]$ so we fail to reject $H_0: \sigma_i^2 = 0.0225$

$$Z_2 = \frac{-0.03 - 0.0225}{0.3 / \sqrt{100}} = -1.75$$

Z_2 is between $[-1.96, 1.96]$ so we fail to reject $H_0: \sigma_i^2 = 0.0225$

ECON 432 Homework 3 R Excercises

Madelyn Caufield

Feb 13,2021

Contents

```
library(mvtnorm)
```

```
CalcSigmaMtx <- function(rho.xy, sig.x, sig.y) {
    # ---- Assert simulate from bivariate normal with rho
    sig.xy = rho.xy * sig.x * sig.y
    matrix(c(sig.x^2, sig.xy, sig.xy, sig.y^2), 2, 2, byrow = TRUE)
}

GenerateBiNormMtx <- function(rho.xy, n = 100, seed = 123, mu.x, mu.y, sig.x,
    sig.y, sigma.xy) {
    # ---- Assert use the rmvnorm() function to simulate from bivariate normal
    set.seed(seed)
    xy.vals = rmvnorm(n, mean = c(mu.x, mu.y), sigma = sigma.xy)
}

layout(matrix(1:1, 1, 1, byrow = TRUE))

muX = 0.01
sigX = 0.25
muY = 0.05
sigY = 0.15
```

```
sigmaXY <- CalcSigmaMtx(rho.xy = 0.99, sigX, sigX)
sigmaXY
```

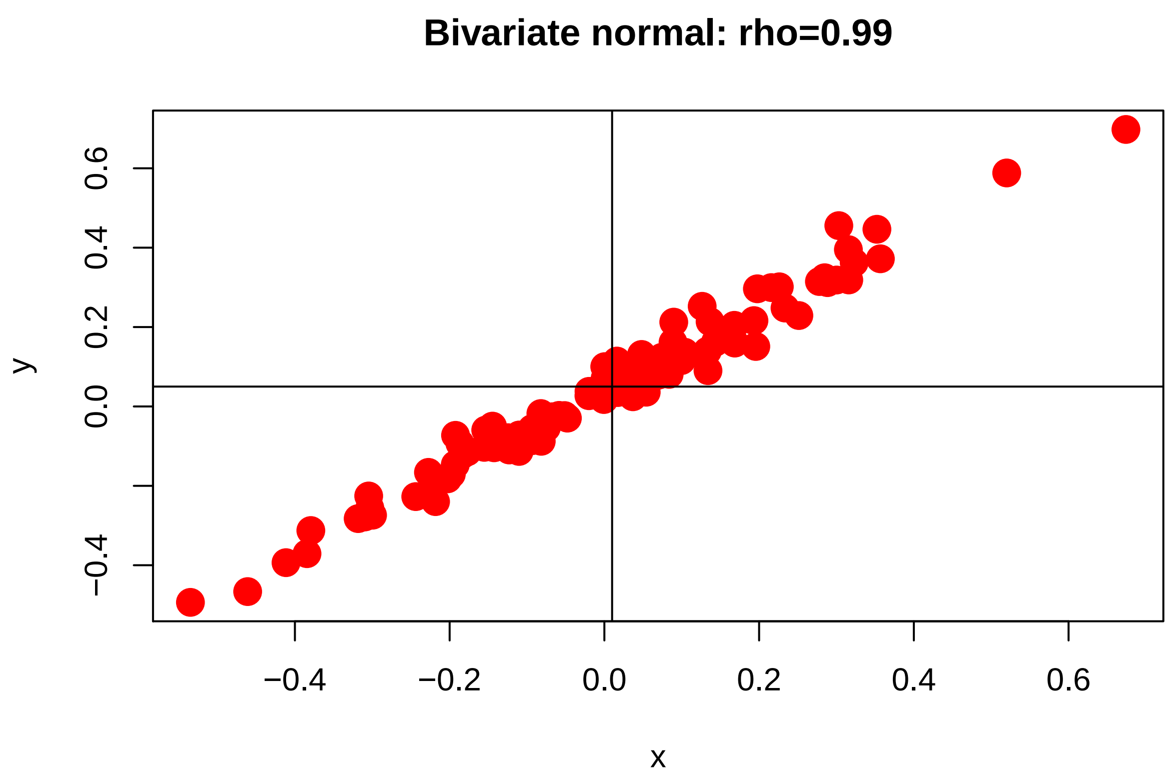
```
##          [,1]     [,2]
## [1,] 0.062500 0.061875
## [2,] 0.061875 0.062500
```

```
valXY <- GenerateBiNormMtx(rho.xy = 0.99, n = 100, seed = 123, muX, muY, sigX,
    sigY, sigmaXY)
head(valXY)
```

```
##            [,1]        [,2]
## [1,] -0.1335479 -0.08529046
## [2,]  0.3158891  0.31868412
## [3,]  0.3154002  0.39504468
```

```
## [4,] -0.1102236 -0.11337299
## [5,] -0.1927160 -0.14668621
## [6,]  0.3000983  0.31849162
```

```
plot(valXY[, 1], valXY[, 2], pch = 16, cex = 2, col = "red", xlab = "x", ylab = "y")
title("Bivariate normal: rho=0.99")
abline(h = muY, v = muX)
```




```
sigma2XY <- CalcSigmaMtx(rho.xy = 0.9, sigX, sigX)
sigma2XY
```

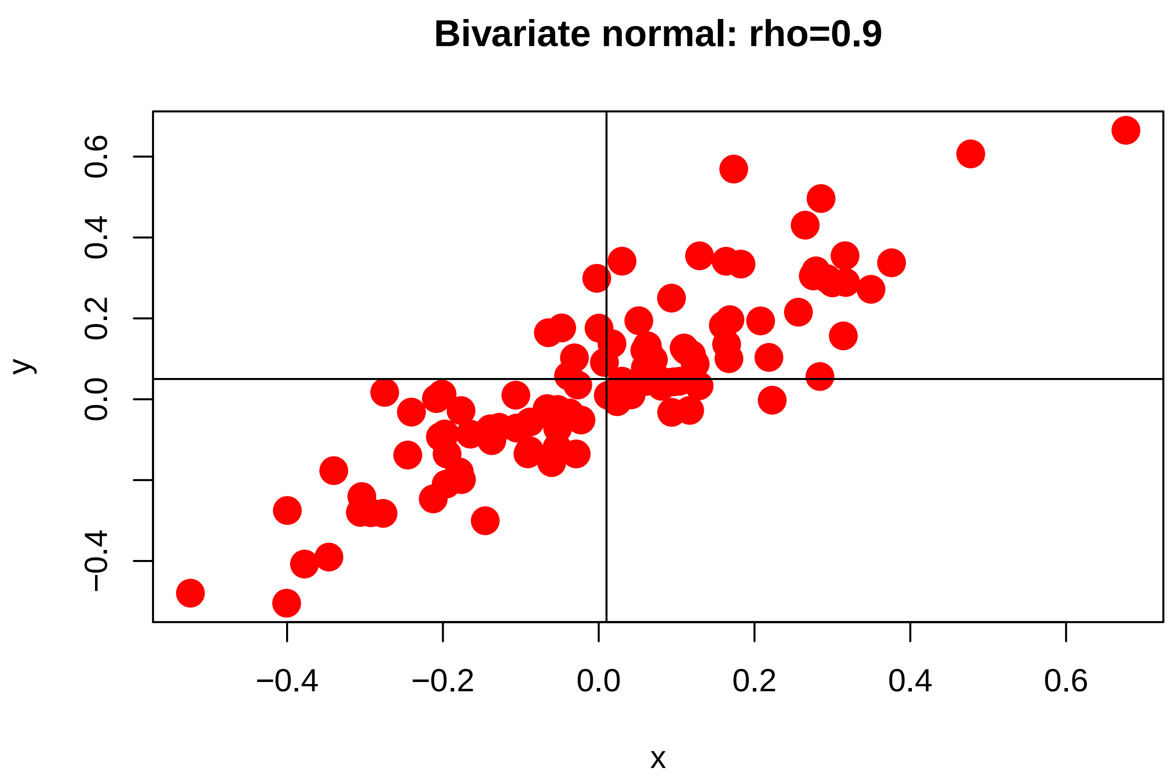
```
##         [,1]    [,2]
## [1,] 0.06250 0.05625
## [2,] 0.05625 0.06250
```

```
val2XY <- GenerateBiNormMtx(rho.xy = 0.9, n = 100, seed = 123, muX, muY, sigX,
    sigY, sigma2XY)
head(val2XY)
```

```
##             [,1]        [,2]
## [1,] -0.13928620 -0.07317384
## [2,]  0.34954130  0.27188876
## [3,]  0.26509975  0.43046645
## [4,] -0.06032943 -0.15677993
```

```
## [5,] -0.19466690 -0.13559909
## [6,]  0.31706937  0.28874299
```

```
plot(val2XY[, 1], val2XY[, 2], pch = 16, cex = 2, col = "red", xlab = "x", ylab = "y")
title("Bivariate normal: rho=0.9")
abline(h = muY, v = muX)
```



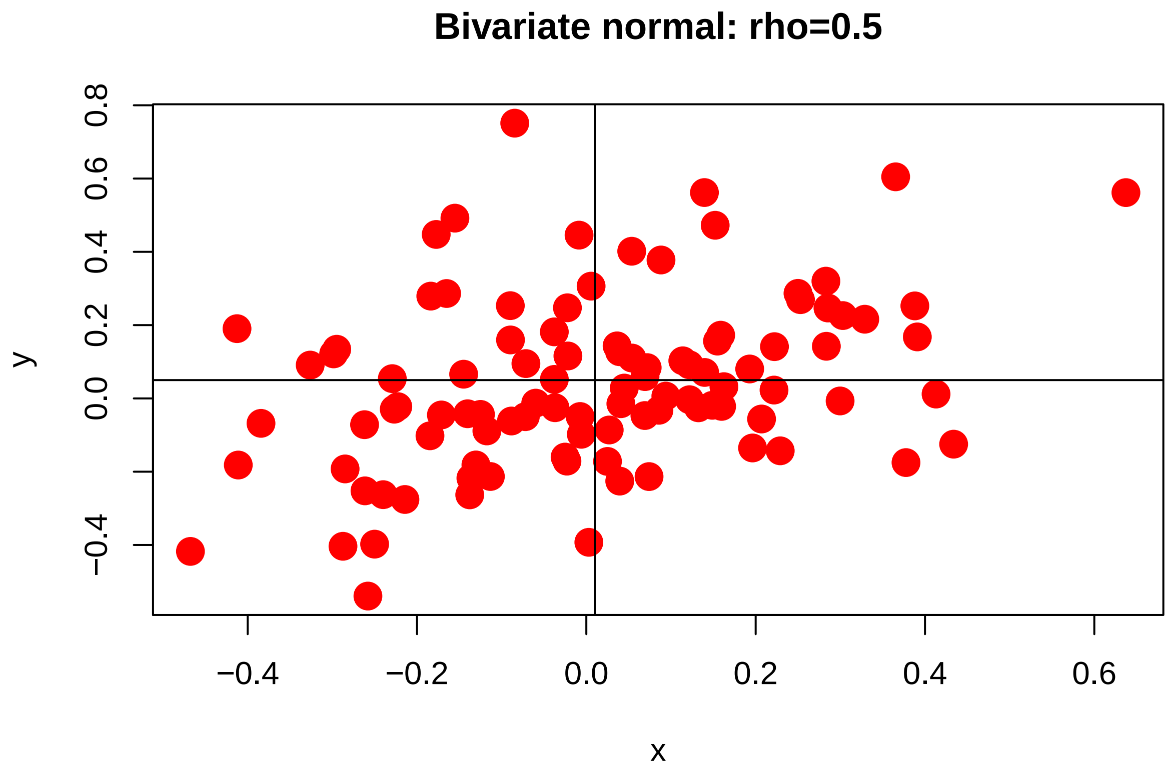

```
sigma3XY <- CalcSigmaMtx(rho.xy = 0.5, sigX, sigX)
sigma3XY
```

```
##         [,1]    [,2]
## [1,] 0.06250 0.03125
## [2,] 0.03125 0.06250
```

```
val3XY <- GenerateBiNormMtx(rho.xy = 0.5, n = 100, seed = 123, muX, muY, sigX,
    sigY, sigma3XY)
head(val3XY)
```

```
##             [,1]        [,2]
## [1,] -0.14023806 -0.04184904
## [2,]  0.39096138  0.16788232
## [3,]  0.15219346  0.47252192
## [4,]  0.03944723 -0.22566536
## [5,] -0.18469868 -0.10206175
## [6,]  0.32887472  0.21609229
```

```r
plot(val3XY[, 1], val3XY[, 2], pch = 16, cex = 2, col = "red", xlab = "x", ylab = "y")
title("Bivariate normal: rho=0.5")
abline(h = muY, v = muX)
```



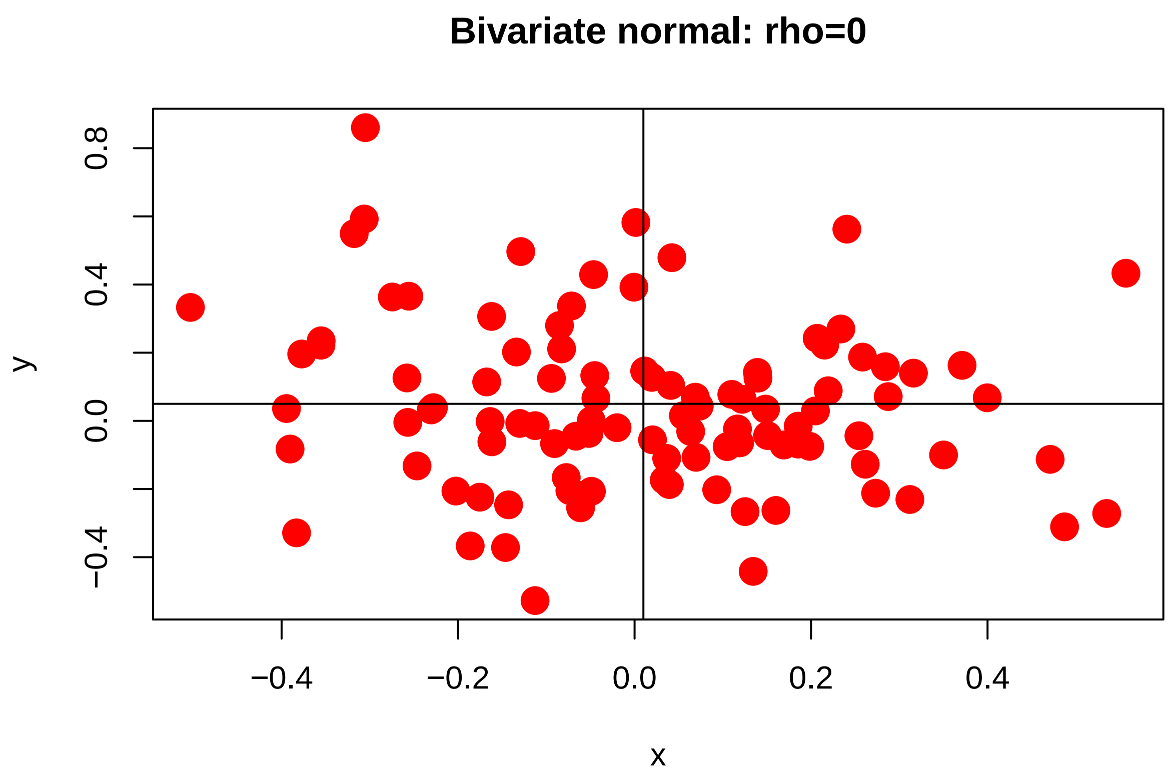

```r
sigma4XY <- CalcSigmaMtx(rho.xy = 0, sigX, sigX)
sigma4XY
```

```
##        [,1]   [,2]
## [1,] 0.0625 0.0000
## [2,] 0.0000 0.0625
```

```r
val4XY <- GenerateBiNormMtx(rho.xy = 0, n = 100, seed = 123, muX, muY, sigX,
    sigY, sigma4XY)
head(val4XY)
```

```
##             [,1]         [,2]
## [1,] -0.13011891 -0.007544372
## [2,]  0.39967708  0.067627098
## [3,]  0.04232193  0.478766247
## [4,]  0.12522905 -0.266265309
## [5,] -0.16171321 -0.061415493
## [6,]  0.31602045  0.139953457
```

```r
plot(val4XY[, 1], val4XY[, 2], pch = 16, cex = 2, col = "red", xlab = "x", ylab = "y")
title("Bivariate normal: rho=0")
abline(h = muY, v = muX)
```



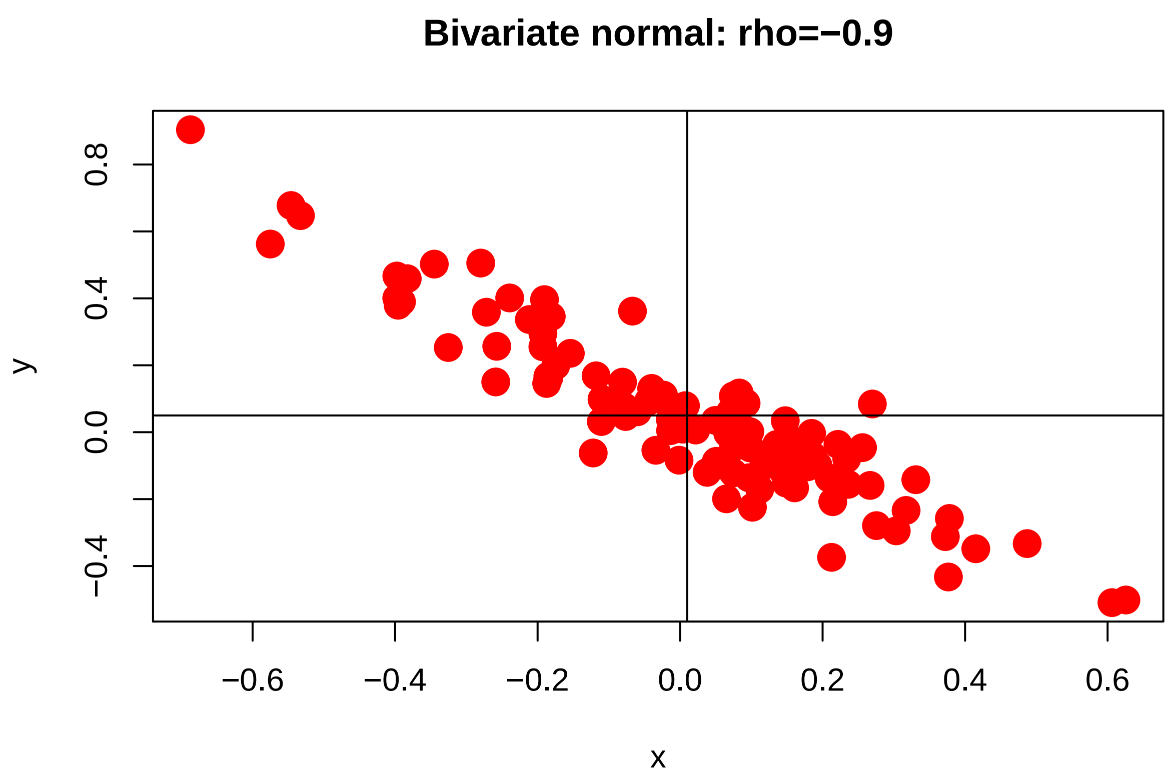

```r
sigma5XY <- CalcSigmaMtx(rho.xy = -0.9, sigX, sigX)
sigma5XY
```

```
##          [,1]     [,2]
## [1,]  0.06250 -0.05625
## [2,] -0.05625  0.06250
```

```r
val5XY <- GenerateBiNormMtx(rho.xy = -0.9, n = 100, seed = 123, muX, muY, sigX,
    sigY, sigma5XY)
head(val5XY)
```

```
##             [,1]        [,2]
## [1,] -0.07816388  0.07565726
## [2,]  0.33081820 -0.14201731
## [3,] -0.19032594  0.39613483
## [4,]  0.27560034 -0.27917359
## [5,] -0.07632391  0.04679076
## [6,]  0.22152287 -0.03630493
```

```r
plot(val5XY[, 1], val5XY[, 2], pch = 16, cex = 2, col = "red", xlab = "x", ylab = "y")
title("Bivariate normal: rho=-0.9")
abline(h = muY, v = muX)
```




```r
setwd("/Users/madelyncaufield/Desktop/ECON432 Data Science for Financial Engineering/Homework 3")
df <- read.csv("MSFT.csv")
head(df)
```

```
##         Date  Open  High   Low Close Adj.Close   Volume
## 1 2010-01-04 30.62 31.10 30.59 30.95  24.10536 38409100
## 2 2010-01-05 30.85 31.10 30.64 30.96  24.11315 49749600
## 3 2010-01-06 30.88 31.08 30.52 30.77  23.96516 58182400
## 4 2010-01-07 30.63 30.70 30.19 30.45  23.71593 50559700
## 5 2010-01-08 30.28 30.88 30.24 30.66  23.87950 51197400
## 6 2010-01-11 30.71 30.76 30.12 30.27  23.57575 68754700
```

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
msft.df = df %>%
          select(Date, Adj.Close)
head(msft.df)
```

```
##         Date Adj.Close
## 1 2010-01-04  24.10536
## 2 2010-01-05  24.11315
## 3 2010-01-06  23.96516
## 4 2010-01-07  23.71593
## 5 2010-01-08  23.87950
## 6 2010-01-11  23.57575
```

```r
class(msft.df$Date)
```

```
## [1] "character"
```

```r
msft.df$Date <- as.Date(msft.df$Date,format = '%Y-%m-%d')
class(msft.df$Date)
```

```
## [1] "Date"
```

```r
msft.df$cc.ret[2:T] <-  log(msft.df$Adj.Close[2:T]/msft.df$Adj.Close[1:(T-1)])
head(msft.df$cc.ret)
```

```
## [1] 0.0000000000 0.0003230295 0.0000000000 0.0003230295 0.0000000000
## [6] 0.0003230295
```

```r
df.g <- msft.df[,c('Date','cc.ret')]
df.g <- msft.df[-c(1),]

head(df.g)
```

```
##         Date Adj.Close       cc.ret
## 2 2010-01-05  24.11315 0.0003230295
## 3 2010-01-06  23.96516 0.0000000000
## 4 2010-01-07  23.71593 0.0003230295
## 5 2010-01-08  23.87950 0.0000000000
## 6 2010-01-11  23.57575 0.0003230295
## 7 2010-01-12  23.41997 0.0000000000
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
fitdistr(df.g$cc.ret, "normal")
```

```
##        mean              sd
##   1.615727e-04   1.615147e-04
##  (3.058355e-06) (2.162584e-06)
```

```r
x = df.g$cc.ret
mu.r = 0.0008232918
sd.r = 0.0160125173
v = sd.r^2
n = length(x)
skewness <- function(x) {
    third.moment <- (1/(n - 2)) * sum((x - mu.r)^3)
    third.moment/(v^(3/2))
}

skewness(df.g$cc.ret)
```

```
## [1] -8.324767e-05
```

The coefficient of skewness is approximately equal to 0. This indicates that the the graph is mostly symmetric, however, it is approximately -0.25 less than 0 meaning it has more of a heavy left tail than a normal distribution.

```r
x = df.g$cc.ret
mu.r = 0.0008232918
sd.r = 0.0160125173
v = sd.r^2
n = length(x)
skewness <- function(x) {
    third.moment <- (1/(n - 2)) * sum((x - mu.r)^4)
    third.moment/(v^(4/2))
}
s = skewness(df.g$cc.ret)

s-3
```

```
## [1] -2.999996
```

The coefficient for excess kurtosis is positive, meaning that the data for cc returns has heavier tails than the normal distribution.

```r
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
jarque.bera.test(df.g$cc.ret)
```

```
##
##  Jarque Bera Test
##
## data:  df.g$cc.ret
## X-squared = 464.83, df = 2, p-value < 2.2e-16
```

We would reject the null hypothesis that the data is normally distributed since the p value is less than 0.05

In view of the testing results from the Jarque-Bera test, we can trust the maximum likelihood estimators of mu and sd. The significant results from the JB test that the data is not normally distributed is consistent with our coefficients of skewness and kurtosis. From the sample skewness we saw that the data had a heavier left tail than normal and from the excess kurtosis, we saw that the cc returns data had heavier tails than that of the normal.