

Stat 462: Lab 4 solutions

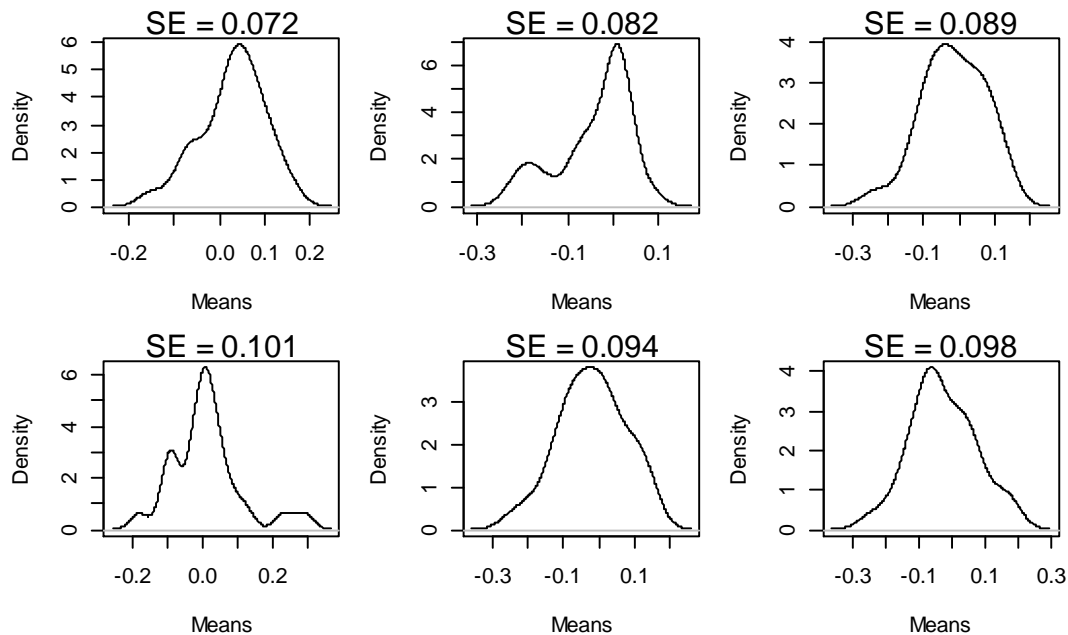
```
# 3.3
replicate( n = 4, sample(1:7) )

# 3.4
# a)
y <- rnorm(100)
cat( "Mean is:", mean(y), "and standard deviation is:", sd(y), "\n" )

# b)
ybar <- numeric(25)
for( i in 1:25 ){
  y <- rnorm(100)
  ybar[i] <- mean(y)
}
cat( "Standard error of the means is:", sd(ybar), "\n" )

# c)
sefun <- function( n=100, m=25 ){
  ybar <- numeric(m)
  for( i in 1:m ){
    y <- rnorm(n)
    ybar[i] <- mean(y)
  }
  return( list( se = sd(ybar), means = ybar ) )
}
# Produce a series of plots of the means
par( mfrow = c(2,3), mar = c(4,4,2,1) )
for( j in 1:6 ){
  res <- sefun()
  plot( density( res$means ), xlab = "Means", main = "" )
  mtext( paste( "SE =", round( res$se, 3 ) ), 3 )
}
```

One thing to notice about the following plots is that although the distribution of the means is Normal, most of the distributions do not *look* Normal because of the small sample size. A question to explore is: How large of a sample do we need before the distribution looks like a Normal distribution?



```
# 3.11
y <- c(87, 53, 72, 90, 78, 85, 83)
cat( "Mean:", mean(y), " Variance:", var(y), "\n" )
# Mean: 78.28571 Variance: 159.9048

x <- rpois(7, 78.3)
cat( "Mean:", mean(x), " Variance:", var(x), "\n" )
```

If you repeat the Poission simulation many times, it is quite uncommon to see a variance as large as 160 (as in the data), although not impossible. It is unlikely that these data follow a Poission distribution, and we might consider the possibility that they are overdispersed (have more variation than would be expected).

```
# 3.13
Pb <- matrix( c( 0.6, 0.2, 0.4, 0.2, 0.4, 0.3, 0.2, 0.4, 0.3 ), ncol =
3 )

# Using the code from the text...
Markov <- function( N = 100, initial.value = 1, P ){
  X <- numeric( N )
  X[1] <- initial.value + 1 # States 0:2; subscripts 1:3
  n <- nrow( P )
  for ( i in 2:N ){
    X[i] <- sample( 1:n, size = 1, prob = P[ X[i-1], ] )
  }
  X - 1
}
```

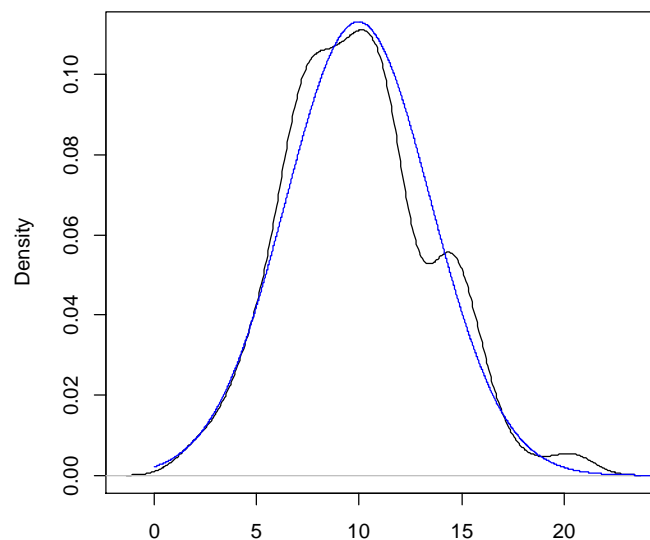
```

# a)
# Simulate 1000 values
sims <- Markov( 1000, P = Pb )
table( sims )/1000

# b)
library(zoo); library(lattice)
plotmarkov <- function( n = 3000, start = 0, window = 300
                        , transition = Pb, npanels = 5 ){
  xc2 <- Markov(n, start, transition)
  mav0 <- rollmean( as.integer(xc2==0), window )
  mav1 <- rollmean( as.integer(xc2==1), window )
  npanel <- cut( 1:length(mav0)
                , breaks = seq( 1, length(mav0), npanels+1 )
                , include.lowest = TRUE )
  df <- data.frame( av0 = mav0, av1 = mav1
                  , x = 1:length(mav0), gp = npanel )
  print( xyplot( av0 + av1 ~ x | gp, data = df
                , layout = c(1,npanels), type = "l"
                , par.strip.text = list( cex=0.65 )
                , scales = list( x = list( relation = "free" ) ) )
        )
}

# Final question
# I will take my data from wind measurements at NYC
wind <- airquality$Wind
cat( "Mean:", mean(wind), " Variance:", var(wind), "\n" )
# Mean: 9.957516 Variance: 12.41154
plot( density( wind ) )
lines( seq(0,25,0.01)
      , dnorm( seq(0,25,0.01), mean(wind), sd(wind) )
      , col = "blue" )
# This distribution looks very close to Normal (NOT necessary)

```



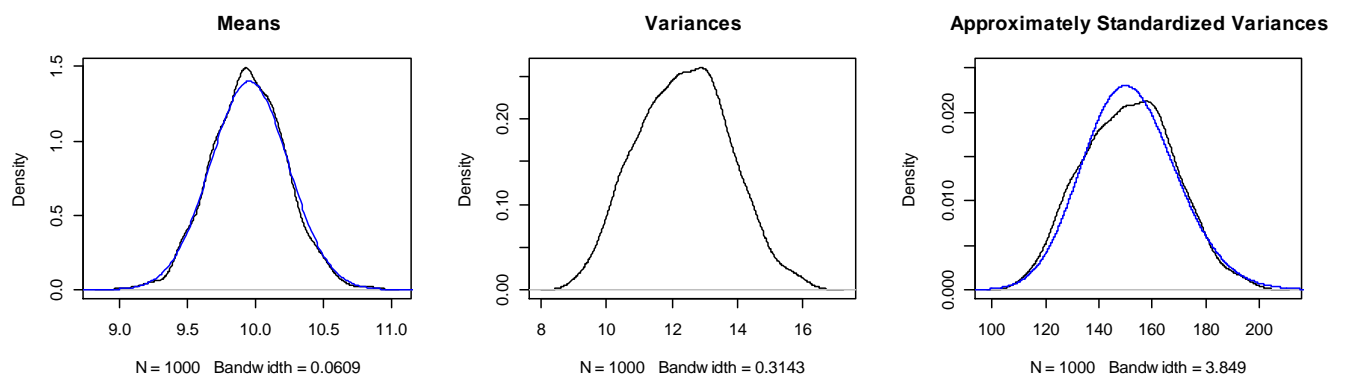
```

# Note that it is NOT necessary to have Normally distributed
# data here.
# Now bootstrap resample the data to get means and variances
# Define a function that takes one bootstrap sample
# and computes the mean and variance
drawSamp <- function( x ){
  xSamp <- sample( x, length(x), replace = TRUE )
  c( mean = mean(xSamp), var = var(xSamp) )
}
# Now use replicate function to repeat the bootstrap
# sample procedure 1000 times
windSamps <- replicate( 1000, drawSamp( x = wind ) )

# The result is a 2 by 1000 matrix, so transpose and
# convert to data.frame
windSamps <- as.data.frame( t( windSamps ) )

# Plot the distributions of the means and variances
par( mfrow = c(1,3) )
with( windSamps, plot( density( mean ), main = "Means" ) )
lines( seq(8.5,11.5,0.01)
      , dnorm( seq(8.5,11.5,0.01), mean(wind)
              , sd(wind)/sqrt(length(wind))) )
      , col = "blue" )
with( windSamps, plot( density( var ), main = "Variances" ) )
with( windSamps, plot( density( var/var(wind)*(length(wind)-1)
      , main = "Approximately Standardized Variances"
      , ylim = c(0,0.025) ) )
lines( seq(80,220,0.01)
      , dchisq( seq(80,220,0.01), length(wind)-1 )
      , col = "blue" )

```



The means should be approximately Normally distributed due to CLT, and in my case they look to be that way (it helps to have a larger dataset and/or Normally distributed data to begin with). If your original data are from a Normal distribution, then the distribution of the variances should follow a scaled Chi-squared distribution. The rightmost panel shows the variances scaled by the original sample variance and the degrees of freedom compared to a Chi-squared distribution. That is, $(n - 1) s^2 / s_{sample}^2 \cong (n - 1) s^2 / \sigma^2 \sim \chi_{n-1}^2$