R Homework 6

November 8th, 2018

Fisher Ankney

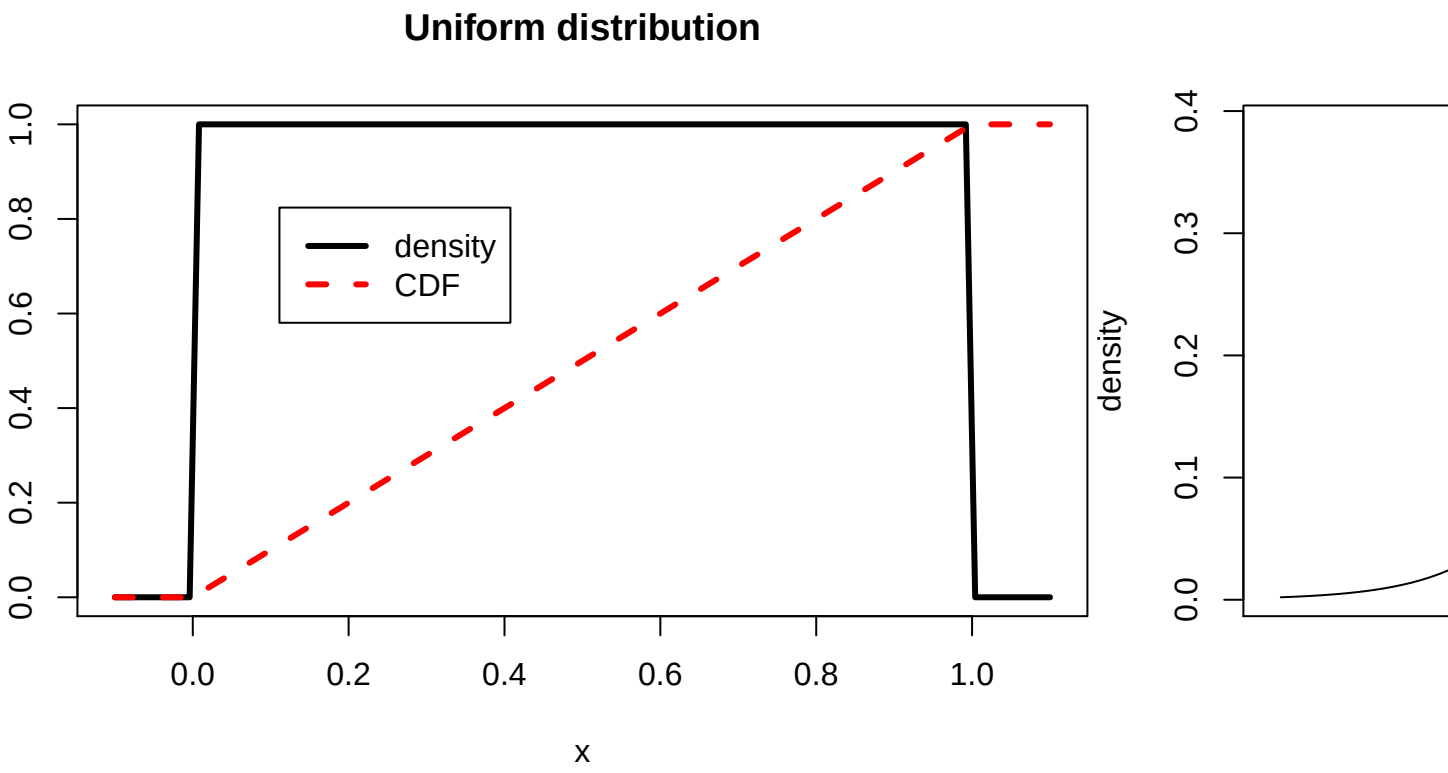
Statistics 5193

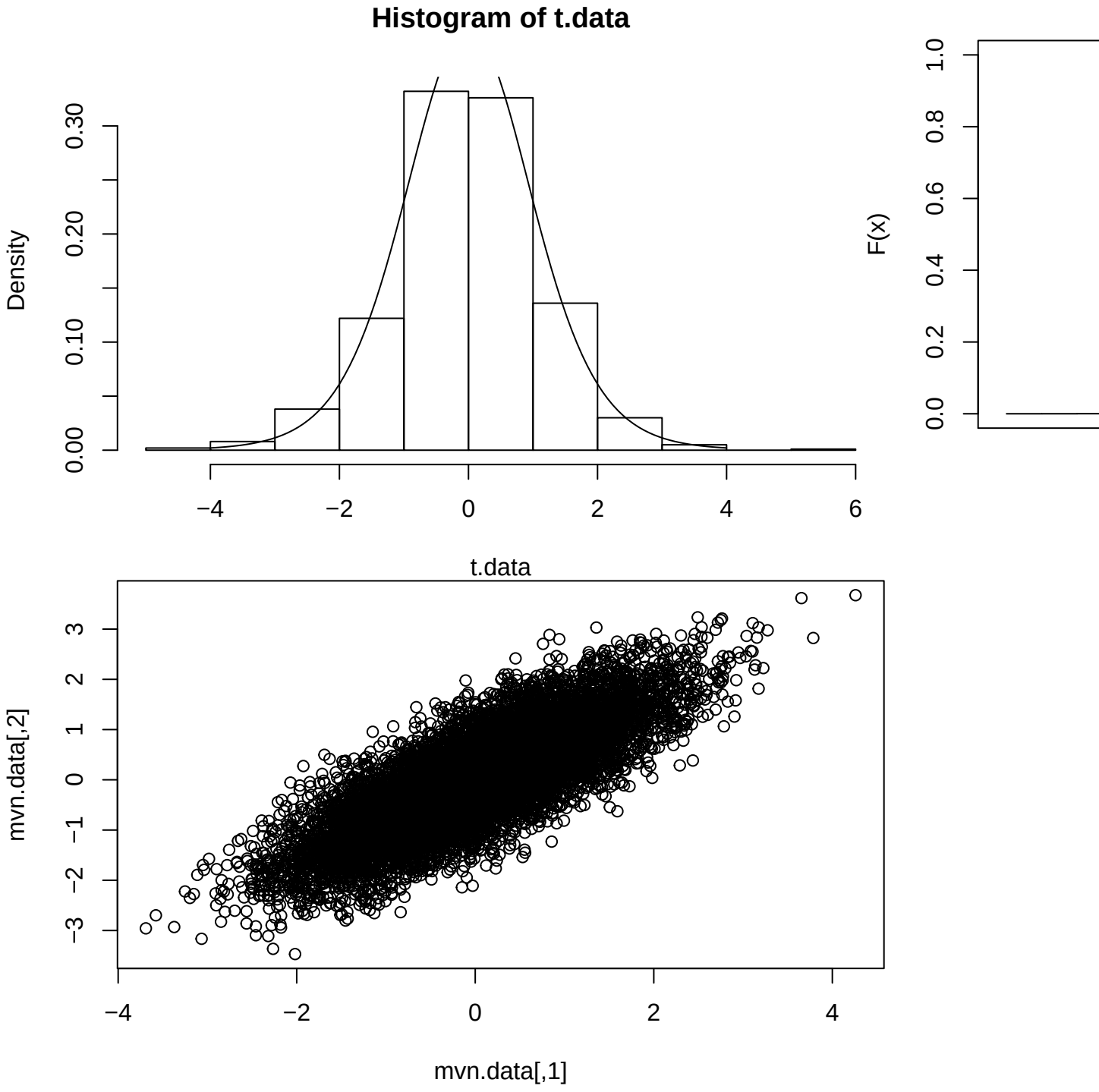
Question 1a.

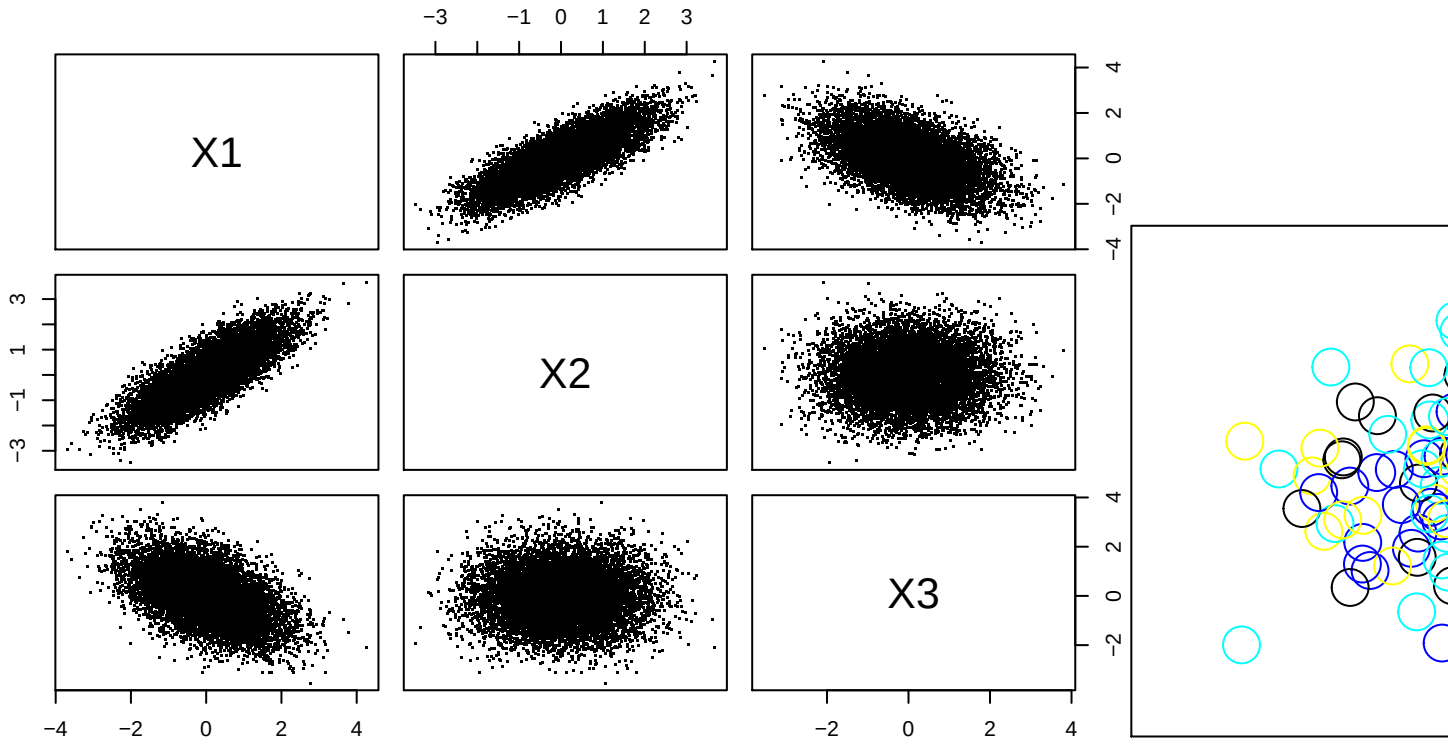
Consider the roster data set from the lecture video.

- Read the help file for the source function, and use the source function to generate the workspace from the lecture video. Print the roster data set to the console and also print a list of the objects in your workspace.

```
help(source)
setwd("~/Documents/data_science/r_stat_5193/scripts")
source("Day8Script.R")
```




Histogram of t.data
Density
0.00
0.10
0.20
0.30
−4
−2
0
2
4
6
t.data
F(x)
0.0
0.2
0.4
0.6
0.8
1.0
mvn.data[,2]
−3
−2
−1
0
1
2
3
−4
−2
0
2
4
mvn.data[,1]




```
help(source)
roster
```

```
##     Firstname   Lastname Math Science English score grade
## 6      Cheryl    Cushing  512      85      28  0.35     C
## 1        John      Davis  502      95      25  0.56     B
## 9        Joel    England  573      89      27  0.70     B
## 4       David      Jones  358      82      15 -1.16     F
## 8        Greg       Knox  625      95      30  1.34     A
## 5      Janice Markhammer  495      75      20 -0.63     D
## 3  Bullwinkle      Moose  412      80      18 -0.86     D
## 10       Mary    Rayburn  522      86      18 -0.18     C
## 2      Angela   Williams  600      99      22  0.92     A
## 7      Reuven    Ytzrhak  410      80      15 -1.05     F
```

```
ls()
```

```
##  [1] "col"       "English"   "First"     "Firstname" "Last"
##  [6] "Lastname"  "Math"      "mu"        "mvn.data"  "n"
## [11] "name"      "rho"       "roster"    "Science"   "score"
## [16] "Sigma"     "Student"   "sums"      "t.data"    "x"
## [21] "y"         "z"
```

Question 1b.

Print the objects score and roster$score to the console. Find the code in the script that is responsible for their difference and print it to the console.

```
score
```

```
##  [1]  0.56  0.92 -0.86 -1.16 -0.63  0.35 -1.05  1.34  0.70 -0.18
```

```
roster$score
```

```
##  [1]  0.35  0.56  0.70 -1.16  1.34 -0.63 -0.86 -0.18  0.92 -1.05
```

```
"roster <- roster[order(Lastname,Firstname),]"
```

```
## [1] "roster <- roster[order(Lastname,Firstname),]"
```

Question 1c.

Create a new variable in the alphabetized data set called "Easy Grade" that assigns a student an A if they are in the top 25% of the class, a B if they score in the middle half of the class, and a C if they are in the bottom 25%. Print the data set to the console.

```
y <- quantile(score, c(.75,.25))

roster$`Easy Grade`[roster$score > y[1]] <- "A"
roster$`Easy Grade`[roster$score < y[1] & roster$score >= y[2]] <- "B"
roster$`Easy Grade`[roster$score < y[2]] <- "C"

roster
```

```
##     Firstname   Lastname Math Science English score grade Easy Grade
## 6      Cheryl    Cushing  512      85      28  0.35     C          B
## 1        John      Davis  502      95      25  0.56     B          B
## 9        Joel    England  573      89      27  0.70     B          A
## 4       David      Jones  358      82      15 -1.16     F          C
## 8        Greg       Knox  625      95      30  1.34     A          A
## 5      Janice Markhammer  495      75      20 -0.63     D          B
## 3  Bullwinkle      Moose  412      80      18 -0.86     D          C
## 10       Mary    Rayburn  522      86      18 -0.18     C          B
## 2      Angela   Williams  600      99      22  0.92     A          A
## 7      Reuven    Ytzrhak  410      80      15 -1.05     F          C
```

Question 1d.

Add a variable grade called MathGrade for each student using the easy grading scale above, but applied to Math Scores. Print the data set to the console

```
y <- quantile(Math, c(.75,.25))

roster$MathGrade[roster$Math > y[1]] <- "A"
roster$MathGrade[roster$Math < y[1] & roster$Math >= y[2]] <- "B"
roster$MathGrade[roster$Math < y[2]] <- "C"

roster
```

```
##     Firstname   Lastname Math Science English score grade Easy Grade
## 6      Cheryl    Cushing  512      85      28  0.35     C          B
## 1        John      Davis  502      95      25  0.56     B          B
## 9        Joel    England  573      89      27  0.70     B          A
## 4       David      Jones  358      82      15 -1.16     F          C
## 8        Greg       Knox  625      95      30  1.34     A          A
## 5      Janice Markhammer  495      75      20 -0.63     D          B
## 3  Bullwinkle      Moose  412      80      18 -0.86     D          C
## 10       Mary    Rayburn  522      86      18 -0.18     C          B
## 2      Angela   Williams  600      99      22  0.92     A          A
## 7      Reuven    Ytzrhak  410      80      15 -1.05     F          C
##    MathGrade
## 6          B
## 1          B
## 9          A
## 4          C
## 8          A
## 5          B
## 3          C
## 10         B
## 2          A
## 7          C
```

Question 1e.

Sort the data set according to Math score (highest score first) and print it to the console. Hint: Go back and read your solution to 1b

```
roster_sort <- roster[order(-roster$Math),]
roster_sort
```

```
##     Firstname   Lastname Math Science English score grade Easy Grade
## 8        Greg       Knox  625      95      30  1.34     A          A
## 2      Angela   Williams  600      99      22  0.92     A          A
## 9        Joel    England  573      89      27  0.70     B          A
## 10       Mary    Rayburn  522      86      18 -0.18     C          B
## 6      Cheryl    Cushing  512      85      28  0.35     C          B
## 1        John      Davis  502      95      25  0.56     B          B
## 5      Janice Markhammer  495      75      20 -0.63     D          B
## 3  Bullwinkle      Moose  412      80      18 -0.86     D          C
## 7      Reuven    Ytzrhak  410      80      15 -1.05     F          C
## 4       David      Jones  358      82      15 -1.16     F          C
##    MathGrade
## 8          A
## 2          A
## 9          A
## 10         B
## 6          B
## 1          B
## 5          B
## 3          C
## 7          C
## 4          C
```

Question 1f.

```
full_name <- paste(roster_sort$Firstname, roster_sort$Lastname)
row.names(roster_sort) <- full_name

roster_sort_drop <- roster_sort[,c(-1,-2)]
roster_sort_drop
```

```
##                   Math Science English score grade Easy Grade MathGrade
## Greg Knox          625      95      30  1.34     A          A         A
## Angela Williams    600      99      22  0.92     A          A         A
## Joel England       573      89      27  0.70     B          A         A
## Mary Rayburn       522      86      18 -0.18     C          B         B
## Cheryl Cushing     512      85      28  0.35     C          B         B
## John Davis         502      95      25  0.56     B          B         B
## Janice Markhammer  495      75      20 -0.63     D          B         B
## Bullwinkle Moose   412      80      18 -0.86     D          C         C
## Reuven Ytzrhak     410      80      15 -1.05     F          C         C
## David Jones        358      82      15 -1.16     F          C         C
```

Question 2.

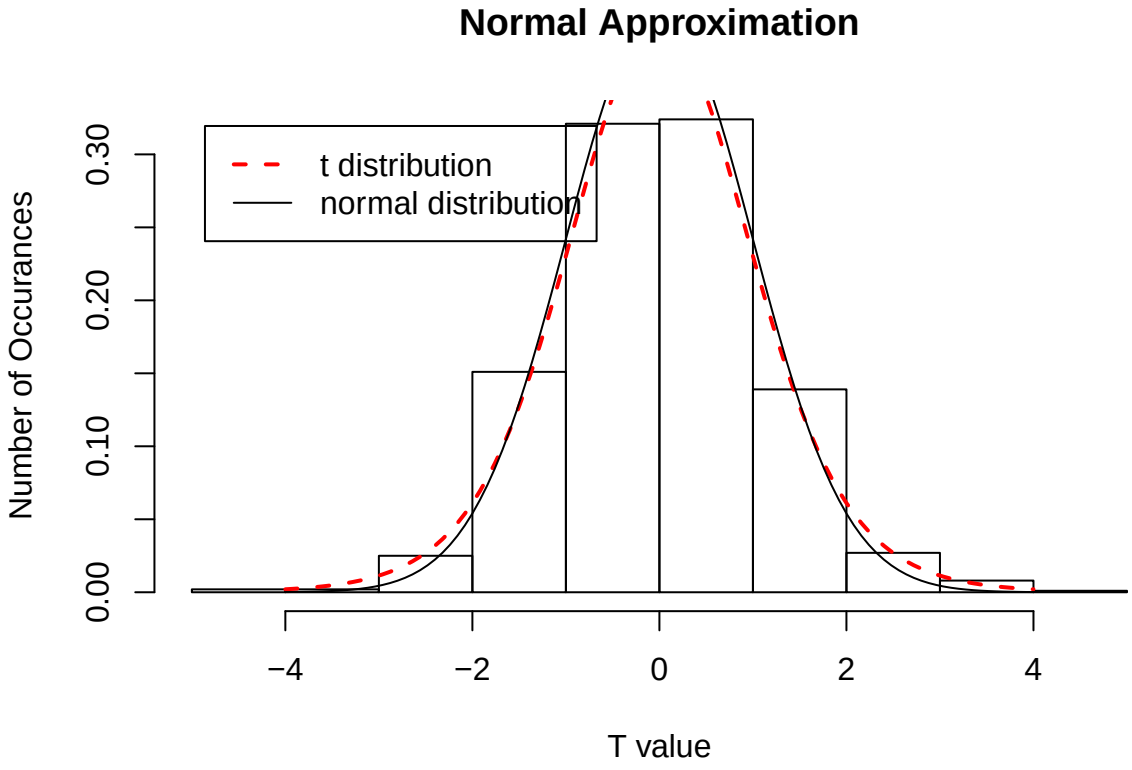
```
t_data <- rt(1000,10)

hist(t_data,
     main = "Normal Approximation",
     ylab = "Number of Occurances",
     xlab = "T value",
     probability = T)


curve(dt(x, df = 10),
      xlim = c(-4,4),
      lwd = 2,
      lty = 2,
      col = 'red',
      add = T)

curve(dnorm(x),
      xlim = c(-4,4),
      add = T)


legend(x="topleft",
      legend = c("t distribution", "normal distribution"),
      lty = c(2,1), col = c('red','black'),lwd=c(2,1),
      inset = 0.05)
```

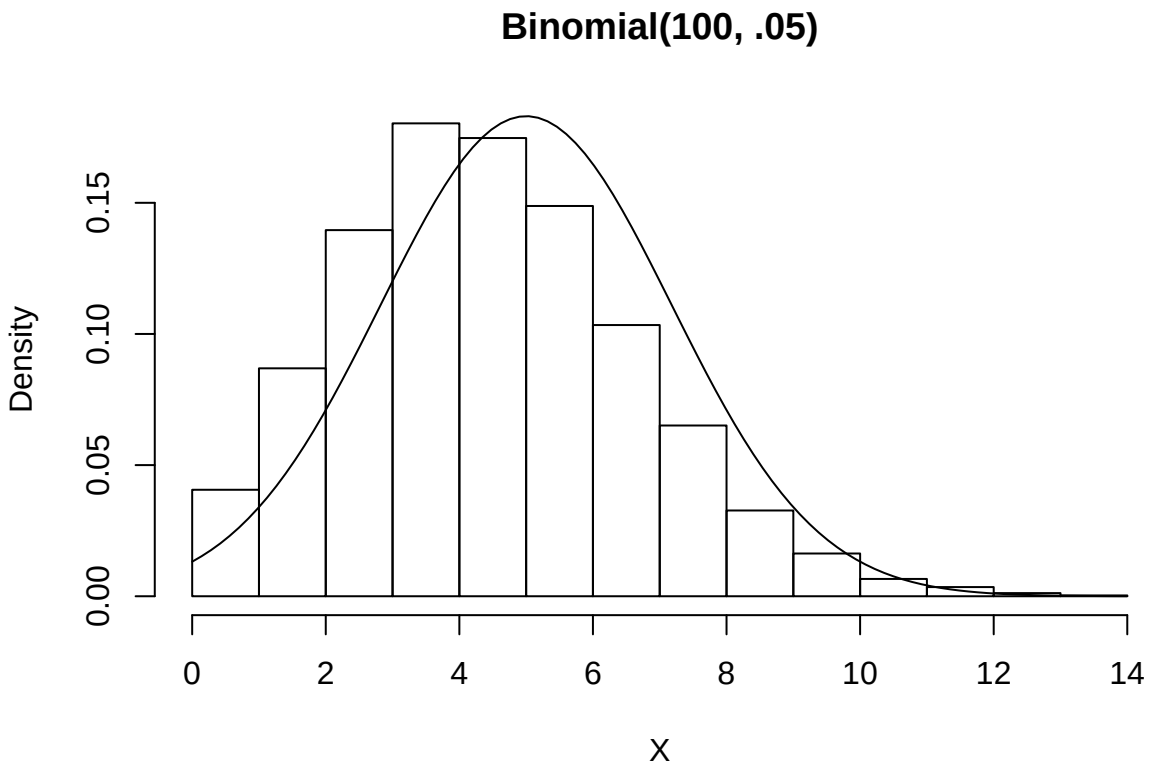


Question 3ab.

```
hist(rbinom(10000, 100, 0.05),
     probability = T,
     main = 'Binomial(100, .05)',
     xlab = 'X'
     )

mu <- 100 * 0.05
sigma <- sqrt((100*0.05)*0.95)

curve(dnorm(x, mu, sigma), add = T)
```

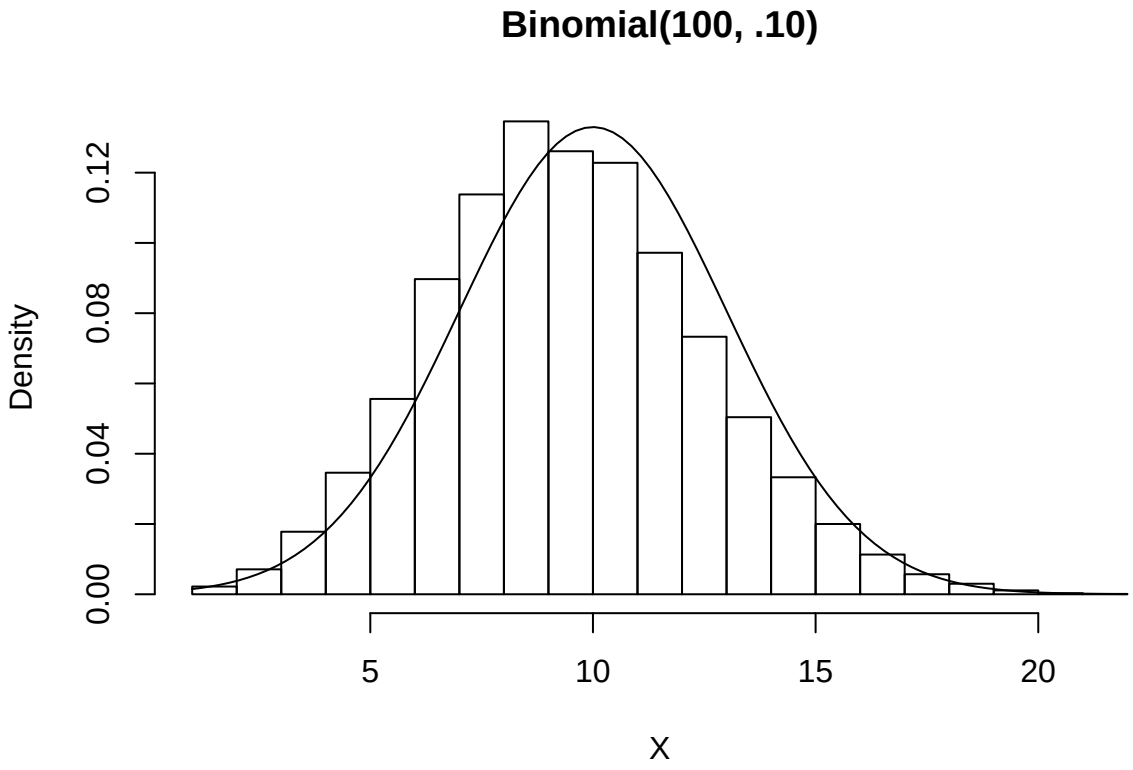


Question 3c.

```
hist(rbinom(10000, 100, 0.10),
     probability = T,
     main = 'Binomial(100, .10)',
     xlab = 'X'
     )

mu <- 100 * 0.10
sigma <- sqrt((100*0.10)*0.90)

curve(dnorm(x, mu, sigma), add = T)
```



Question 3d.

Yes, this seems reasonable based off the above plots.