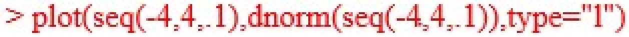
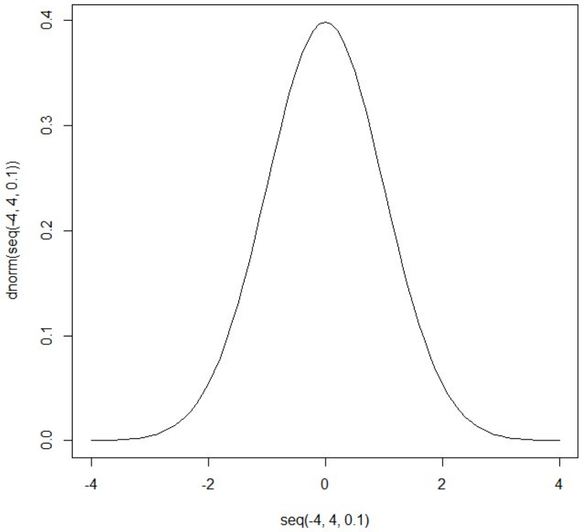
Baruch College, STA-CIS 3920, Exercise # 1
7/14/2020 William Hall 20

Exercise 1.1: Graphing the standard Normal (Z) density function

a. I am going to create a plot of Normal density function in R. The R command is:

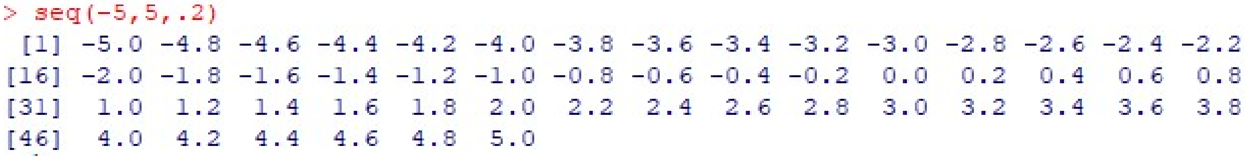


```
> plot(seq(-4,4,.1),dnorm(seq(-4,4,.1)),type="l")
```



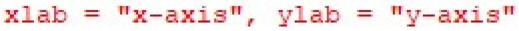
b. The seq function generates a sequence of numbers. In the graph shown, the range of numbers are -4 to 4 in increments of 0.1. With the plot function above, it visualizes the parameters of the seq function. Dnorm is a density function for the normal distribution. The 'Type = "1"' parameter creates a line through points on the graph.

c. As explained in the R documentation, this form of seq generates equally spaced values within the range specified using *length.out*. In my example below, I performed my own seq command. As you can see, the length.out created equally spaced out points every 0.2 within the range of -5 to 5.



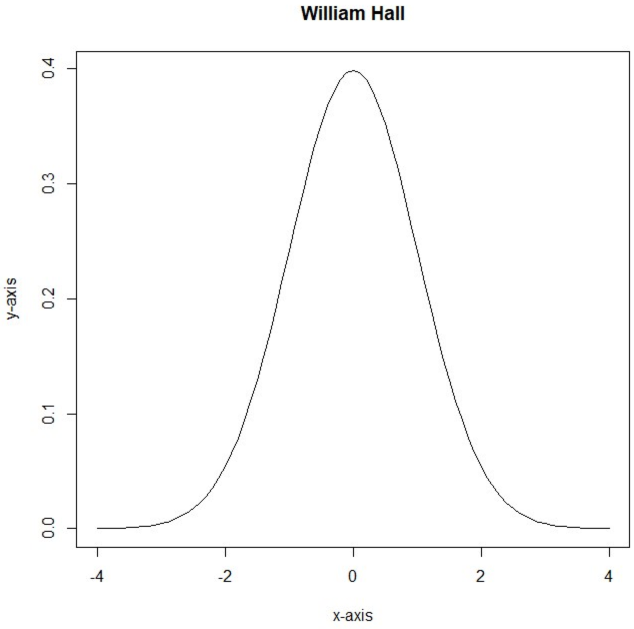
```
> seq(-5,5,.2)
 [1] -5.0 -4.8 -4.6 -4.4 -4.2 -4.0 -3.8 -3.6 -3.4 -3.2 -3.0 -2.8 -2.6 -2.4 -2.2
[16] -2.0 -1.8 -1.6 -1.4 -1.2 -1.0 -0.8 -0.6 -0.4 -0.2  0.0  0.2  0.4  0.6  0.8
[31]  1.0  1.2  1.4  1.6  1.8  2.0  2.2  2.4  2.6  2.8  3.0  3.2  3.4  3.6  3.8
[46]  4.0  4.2  4.4  4.6  4.8  5.0
```

d. I have started to make a title named "William Hall" using the main function: `main="William Hall",` After that, I made labels for the x and y axis.

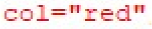


```
xlab = "x-axis", ylab = "y-axis"
```

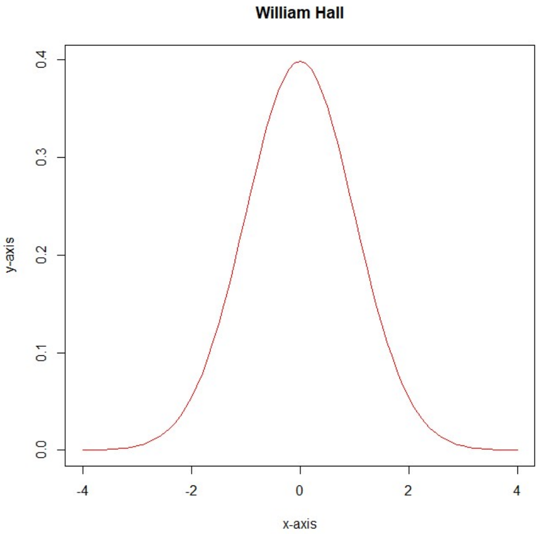
This resulted in the following graph:

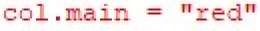


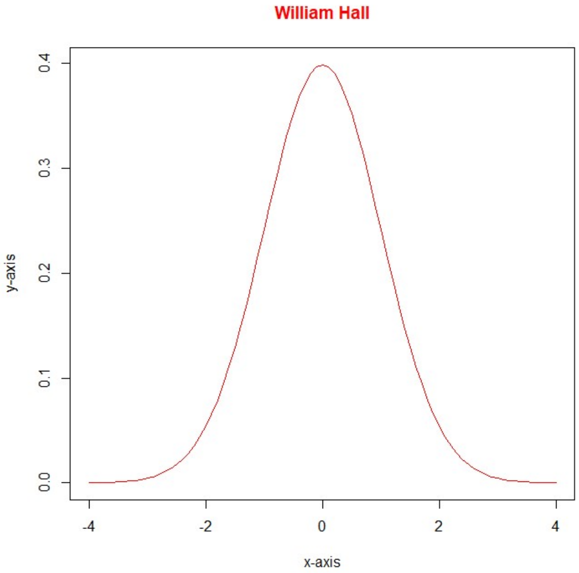
e. First, I used the *col* command to change the color of the line to red.



```
col="red"
```

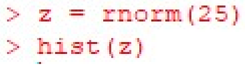


After that I used  `col.main = "red"` to change the color to the title.

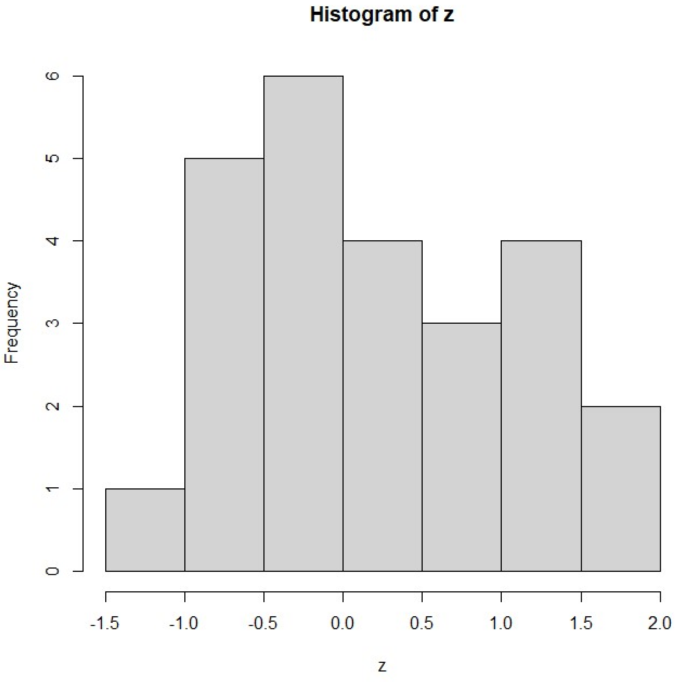


<u>Exercise 1.2: Using a histogram to summarize sample points from Normal dist.</u>

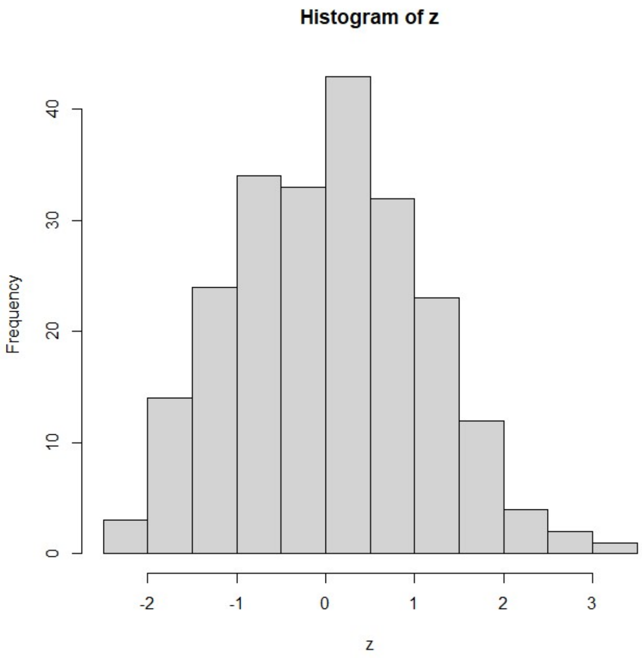
a. When entering the following code into the prompt:
It generated the following histogram:



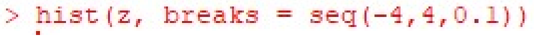
```
> z = rnorm(25)
> hist(z)
```

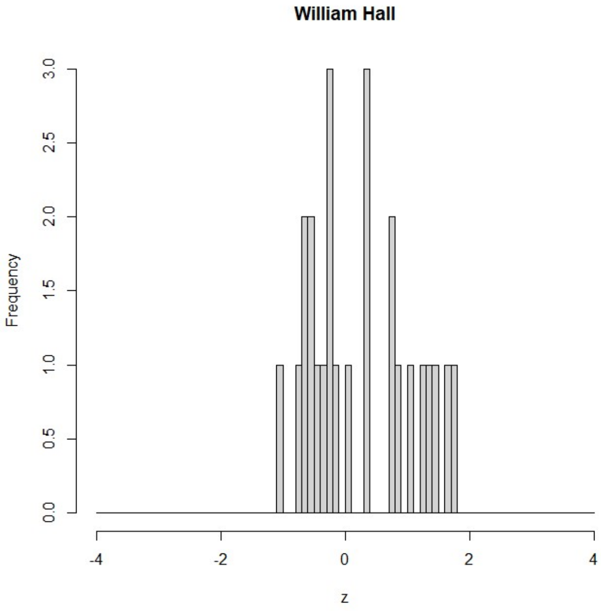


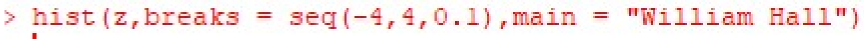
b. When entering 200 more points into the vector “z”, it produced the following histogram:



From what I noticed, the frequency in histogram (b) increased as did the points on the x-axis. There were more bins and were much higher than the previous histogram with less points.

c. When entering  `> hist(z, breaks = seq(-4,4,0.1))` I noticed the breaks command controls the number of bins produced within the histogram. In the new histogram, there are more bins and expand further out within the range.



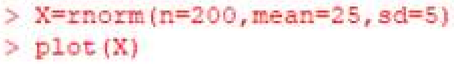


```
> hist(z,breaks = seq(-4,4,0.1),main = "William Hall")
```

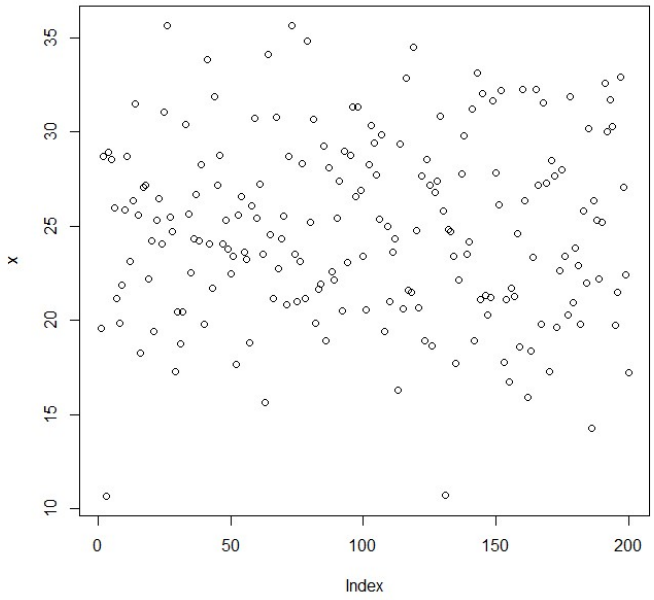
The main function allowed me to change the title of the histogram.

Exercise 1.3: Plotting time series of sample points from Normal dist’s

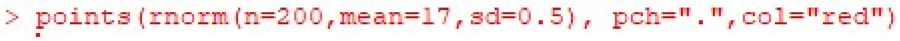
This is the original points provided using



```
> X=rnorm(n=200,mean=25,sd=5)
> plot(X)
```

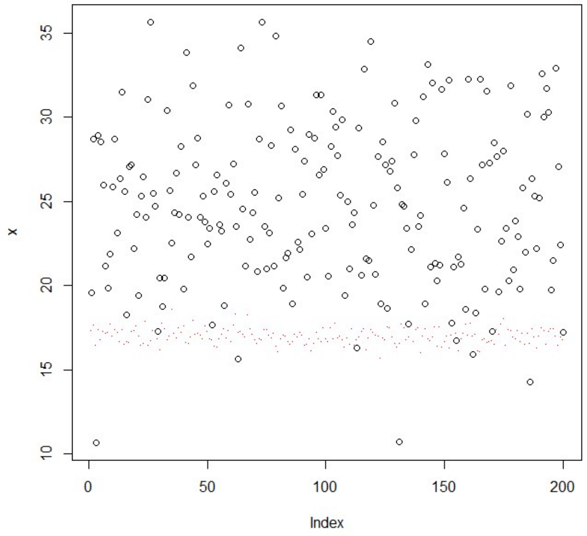


The second graph using



```
> points(rnorm(n=200,mean=17,sd=0.5), pch=".",col="red")
```

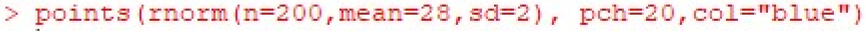
generated the following graph:



.

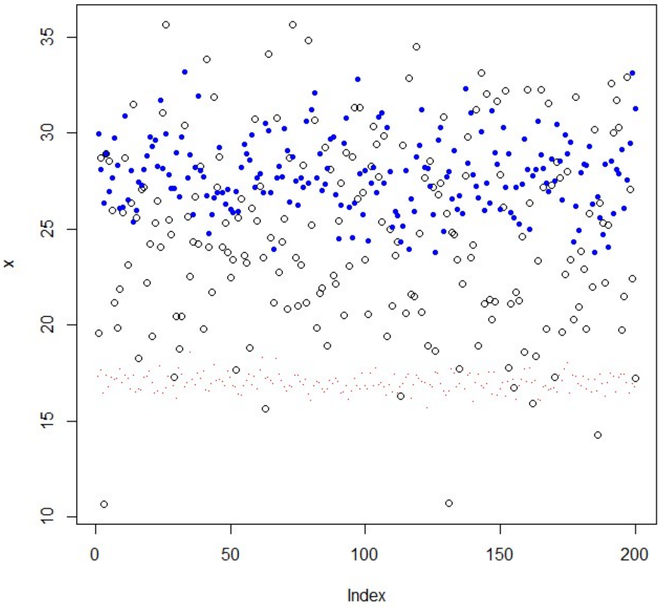
It appears that the code produced many more dots that are tinier and red. The concentration of the tiny dots appear to be densely concentrated around the mean (17). They’re tightly compacted due to the standard deviation (0.5).

In the final example,

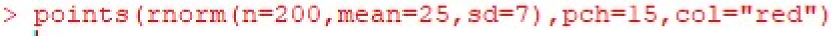


```
> points(rnorm(n=200,mean=28,sd=2), pch=20,col="blue")
```

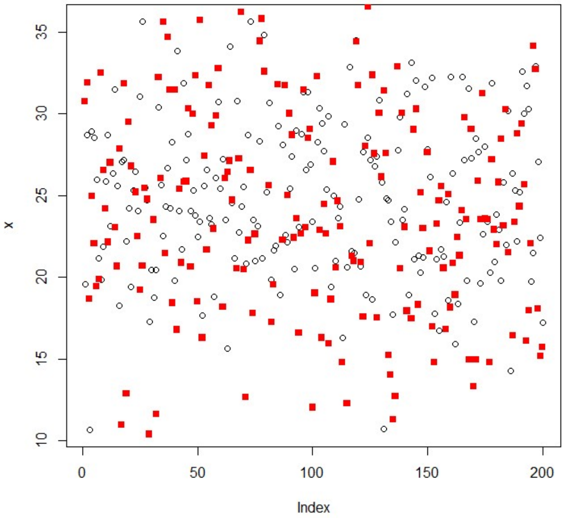
It appears that the blue dots are generally concentrated around the mean (28) but more spread apart due to the increased standard deviation (2).



Next, I would like to see how spread apart the points will be with a higher standard deviation and a mean around 25. While generating my own, I will use the vector "x" for comparison. This is the code I will be using:

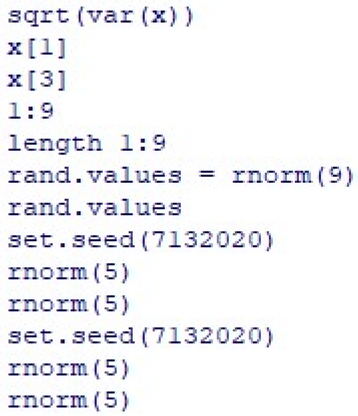


```
> points(rnorm(n=200,mean=25,sd=7),pch=15,col="red")
```



As shown, the red points are spread further away from the mean due to a higher standard deviation.

Appendix: Practicing commands



```
sqrt(var(x))
x[1]
x[3]
1:9
length 1:9
rand.values = rnorm(9)
rand.values
set.seed(7132020)
rnorm(5)
rnorm(5)
set.seed(7132020)
rnorm(5)
rnorm(5)
```