STA 360 Lab 2: R review

Isaac Fan

1/29/2021

Preliminaries

Please turn in a PDF of this Rmd file on Sakai by **Friday, February 5th at 11:59 PM**. Exercises 5 and 6 will be graded for completion. No other exercises will be graded, so it is your choice to do them.

Exercise 5 (For completion)

(a) Generate 500 samples from a Beta distribution with shape parameter $[a, b] = [0.5, 0.5]$ and store the samples in a variable called `W`.

```
W <- rbeta(500, .5, .5, ncp = 0)
```

(b) Generate 1000 samples from a binomial distribution with parameters $[n, p] = [100, 0.2]$ and store them in a variable called `B` (hint: read the documentation to make sure you're inputting the parameters in the correct spots). Calculate the mean of `B` and compare this to the mean of this binomial distribution (np).

```
B <- rbinom(1000, 100, .2)
```

```
mean(W)
```

```
## [1] 0.5019461
```

```
mean(B)
```

```
## [1] 20.132
```

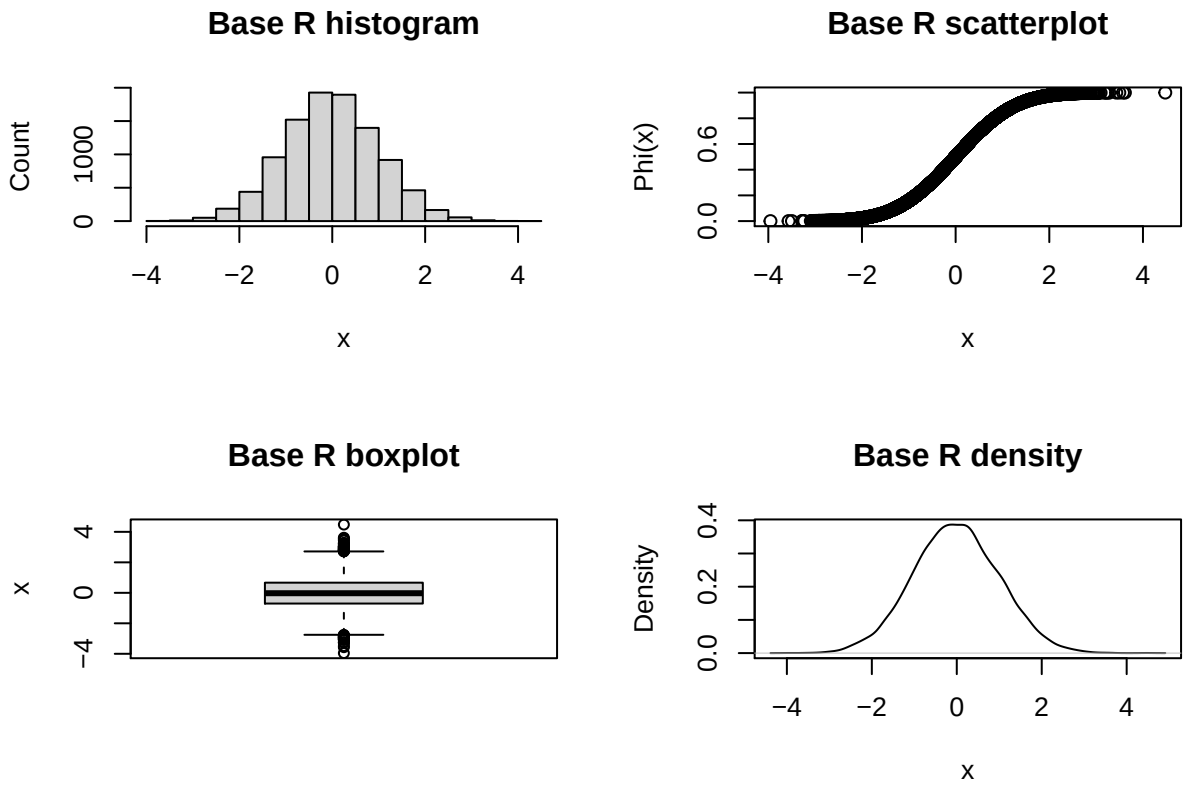
The mean of the Binomial distribution is significantly higher than the mean of the Beta distribution by roughly 40 times.

Plotting

When making plots in R, you have two main options: (1) the base R plotting function `plot` and (2) the package `ggplot2`.

The base R plotting functions are nice for quick, simple visualizations of data. Here are some examples:

```r
norm_samples <- rnorm(10000)
#
par(mfrow = c(2, 2)) # Set the number of rows and columns for display panels
#
hist(norm_samples,
     main = "Base R histogram",
     xlab = "x", ylab = "Count")
#
plot(x = norm_samples, y = pnorm(norm_samples),
     main = "Base R scatterplot",
     xlab = "x", ylab = "Phi(x)")
#
boxplot(norm_samples,
        main = "Base R boxplot",
        ylab = "x")
#
plot(density(norm_samples),
     main = "Base R density",
     xlab = "x", ylab = "Density")
```



Plotting with `ggplot2` is generally a little bit easier when working with data in large tables. It has a gallery of built-in themes, and there are many extensions that make producing complicated visualizations relatively straightforward.

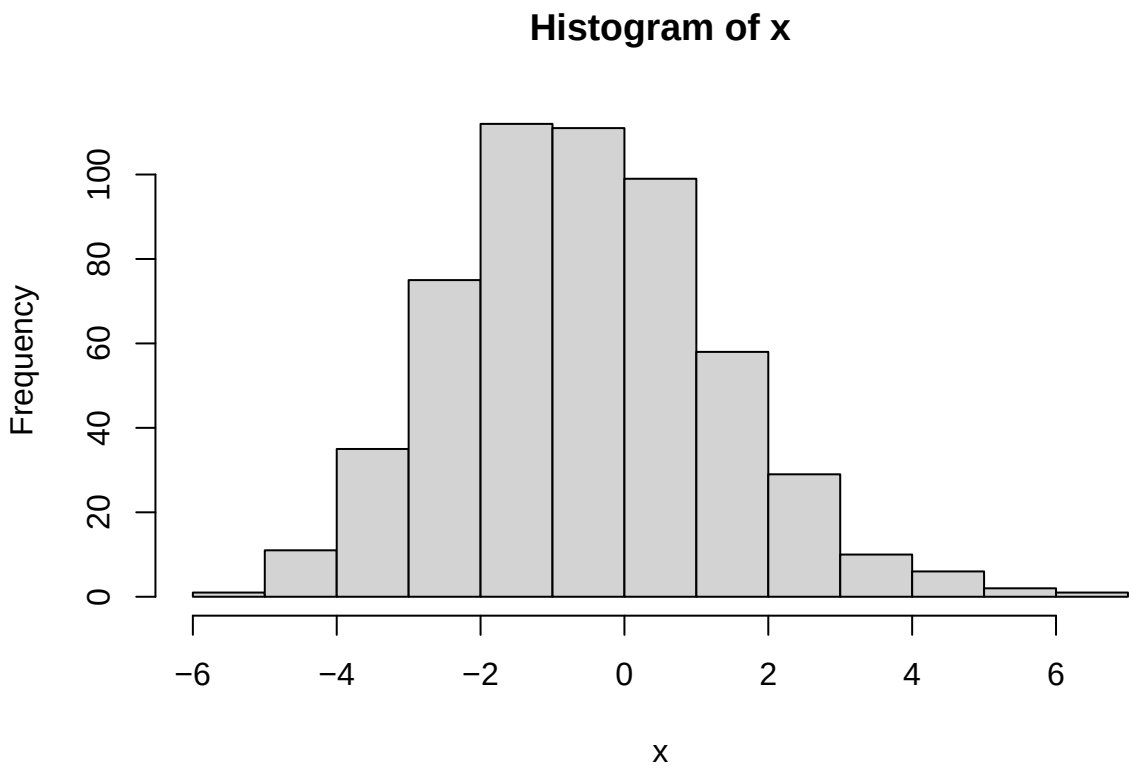
Exercise 6 (For Completion)

The following line of code produces a time series of length 550 and stores it in the variable `x`.

```
#
x0 <- rt(1, df = 5)
xprev <- x0
x <- rep(0, 550)
x[1] <- xprev
for(i in 1:549){
  x[i+1] <- 0.8*xprev + rnorm(1)
  xprev <- x[i+1]
}
```

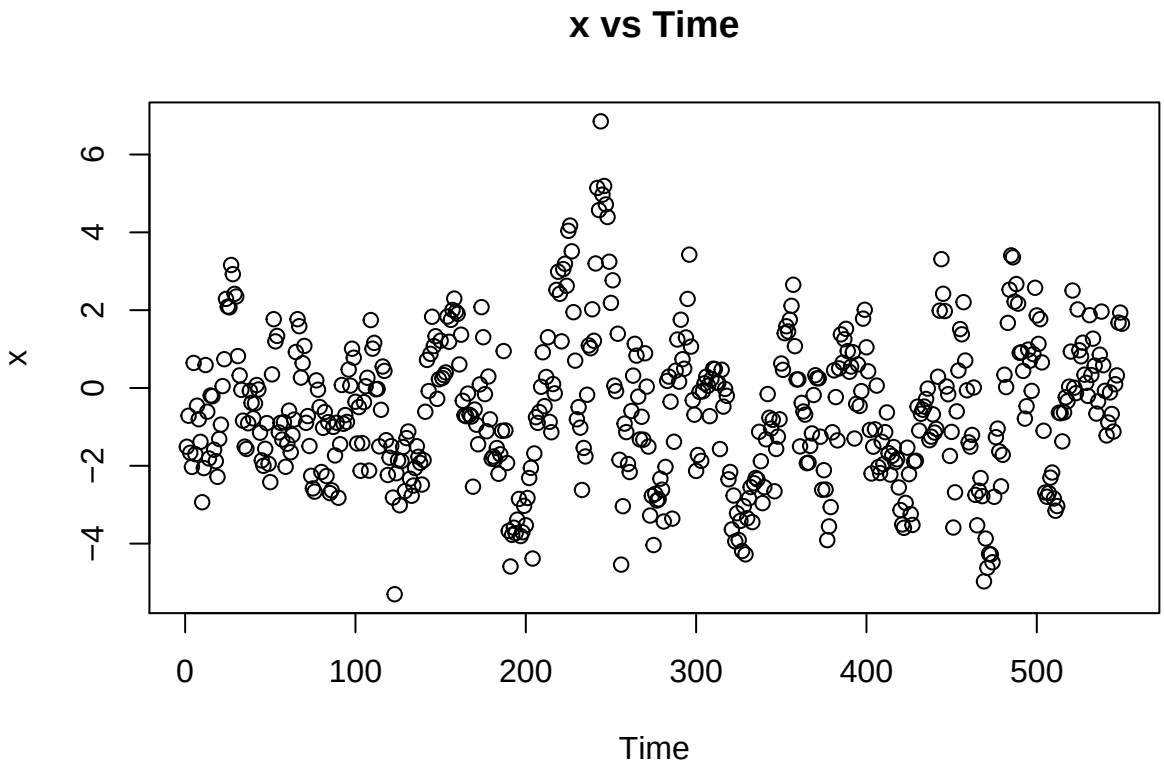
(a) Make a histogram of `x`.

```
hist(x)
```



(b) Make a plot of `x` (on the y-axis) against time (on the x-axis).

```
plot(x = time(x), y = x,
     main = "x vs Time",
     xlab = "Time", ylab = "x")
```



R tutorials and resources

For more information on the R programming language please refer to some or all of the following resources:

- R for Data Science
- Advanced R
- ggplot2
- RStan documentation
- ColorBrewer