Bonnie Turek

Eco 634 – Lab 4 (Probability Distributions)

*I worked with Sonja on this lab

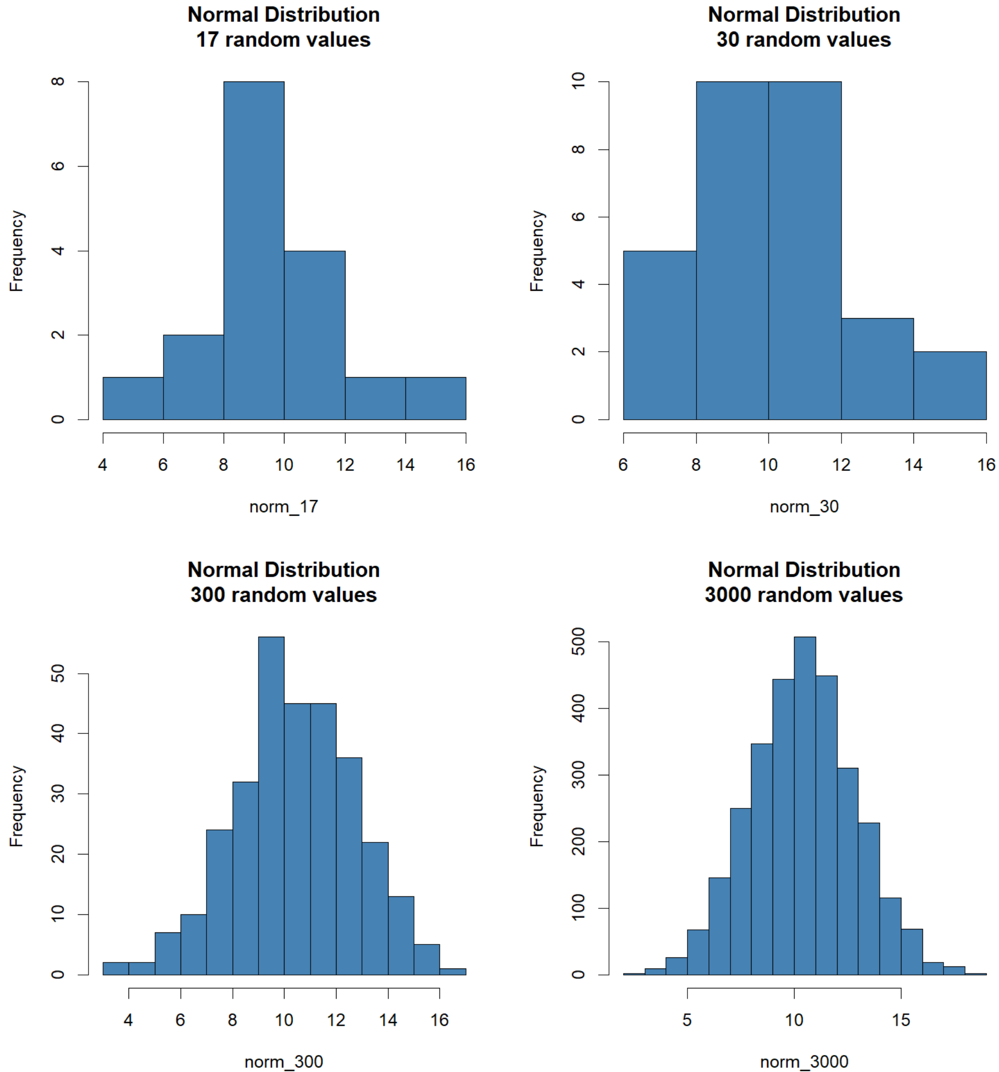
Q1. Show the code you used to create your four randomized normal distribution vectors:

```
set.seed(12)
my_mean = 10.4
my_sd = 2.4
norm_17 = rnorm(n = 17, mean = my_mean, sd = my_sd)
norm_30 = rnorm(n = 30, mean = my_mean, sd = my_sd)
norm_300 = rnorm(n = 300, mean = my_mean, sd = my_sd)
norm_3000 = rnorm(n = 3000, mean = my_mean, sd = my_sd)
```

Q2. Code to build the histogram figure and save it to images folder using PNG parameters:

```
require(here)
image_file = "lab_04_hist_01.png"
png(here("images", image_file), width = 1500, height = 1600, res = 180)
par(mfrow = c(2, 2))
hist(norm_17, main = "Normal Distribution\n17 random values", cex = 5, col = "steelblue")
hist(norm_30, main = "Normal Distribution\n30 random values", cex = 5, col = "steelblue")
hist(norm_300, main = "Normal Distribution\n300 random values", cex = 5, col = "steelblue")
hist(norm_3000, main = "Normal Distribution\n3000 random values", cex = 5, col = "steelblue")
dev.off()
```

Q3.



Q4. Qualitatively describe the differences among the three histograms.

The three histograms use different numbers of randomly generated values, and therefore they demonstrate a different variation of the normal distribution. There are obviously more histogram bins with the normal distribution of 3000 values versus the only 17 values. The histograms with larger numbers of values (i.e. 300 and 3000) better resemble the typical bell-curve shape of the normal distribution. The 17 and 30 value histograms are less noticeable as a normal distribution bell-curve as there's less frequency and fewer data points in general.

Q5. Explain why the shapes of the histograms are different.

The shapes of the histograms are different due to the sample size, n, for each of the normal distribution vectors we created. Small samples are subject to more sampling error, i.e. non-representative samples. These appear less evenly distributed into histogram bins. If you randomly generate more numbers, my histograms will look more like a normal distribution bell curve.

Q6. What are the parameters and their values for the standard Normal distribution?

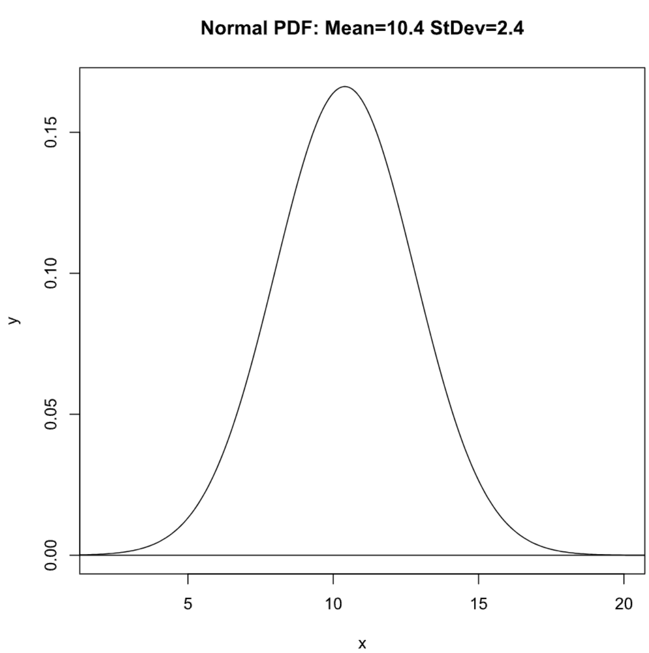
Parameters for the randomly generated numbers normal distribution are ‘n’ , ‘mean’ , and ‘sd’. These are the number of observations, the vector of means, and the vector of standard deviations. The default values are seen in the help for rnorm in R (rnorm(n, mean = 0, sd = 1)) Default values for mean is 0 and sd is 1. These are the most basic values that are used for the “standard” normal distribution.

In the previous questions Q1-Q5, we changed these values of the standard normal distribution and we were using a value of 10.4 for mean and a value of 2.4 for the standard deviation.

Q7. R code you used to create your figure - code that builds the figure as well as saves it to a file

```
x = seq(-35, 35, length.out = 1000)

y = dnorm(x, mean = 10.4, sd = 2.4)

image_file = "images/norm_1.svg"

svg(filename = image_file, width = 7, height = 7)

plot(x, y, main = "Normal PDF: Mean=10.4 StDev=2.4", type = "l", xlim = c(2, 20))

abline(h = 0)

dev.off()
```

Q8. SVG file will also be uploaded in Moodle with my lab document.



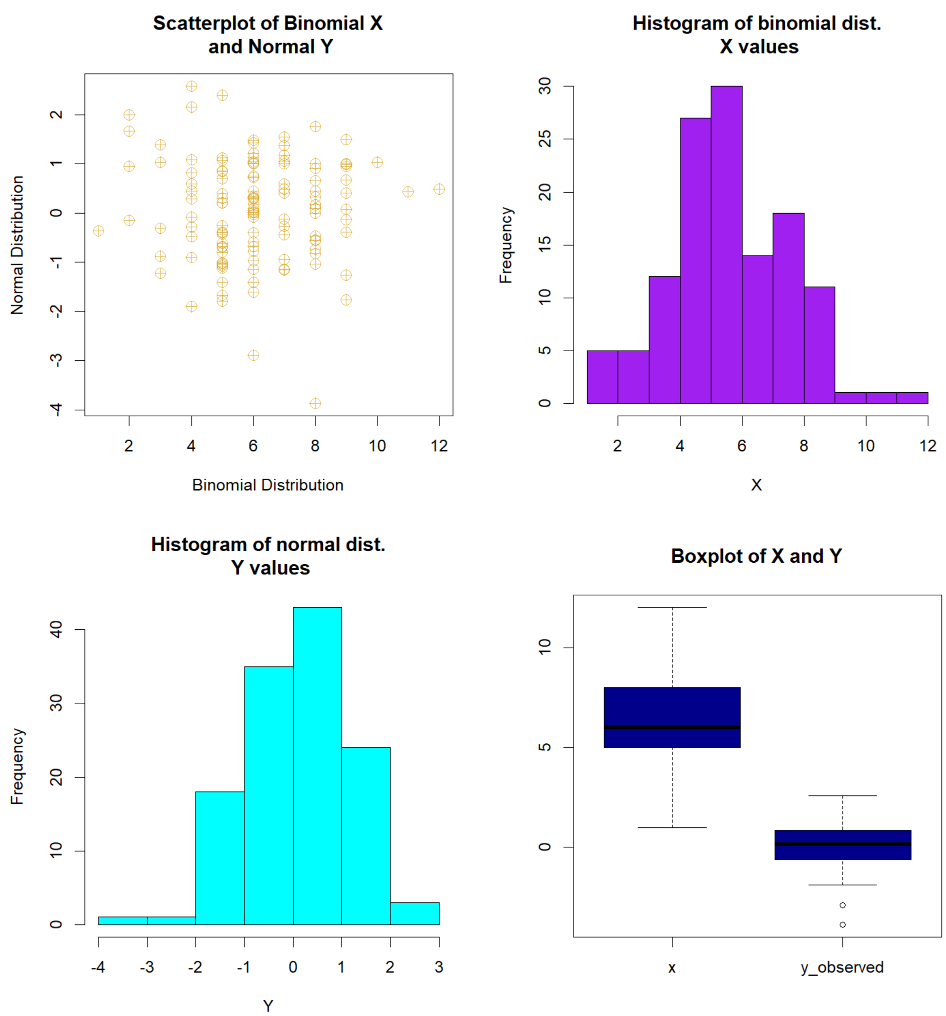
Q9. R code you used to create one of the random datasets in your figure:

```
image_file = "lab_04_ques10.png"

png(here("images", image_file), width = 1500, height = 1600, res = 180)

par(mfrow = c(2, 2))

set.seed(90)

n_pts = 125

my_size = 20

x = rbinom(n = n_pts, size = my_size, prob = 0.3)

dat = data.frame(x = x, y_observed = rnorm(n_pts))

plot(dat$x, dat$y_observed, xlab = "Binomial Distribution",

    ylab = "Normal Distribution",

    main = "Scatterplot of Binomial X\n and Normal Y",

    col = adjustcolor("goldenrod", alpha.f = 0.7), cex = 1.5, pch = 10)

hist(dat$x, xlab = "X", main = "Histogram of binomial dist.\n X values", col = "purple")

hist(dat$y_observed, xlab = "Y", main = "Histogram of normal dist.\n Y values", col = "cyan")

boxplot(dat, col = "darkblue", main = "Boxplot of X and Y")

dev.off()
```

Q10. Upload an image file of your figure. It may be in png (raster graphics), svg (vector graphics), or pdf (vector graphics) format:

*Also will be uploaded on Moodle

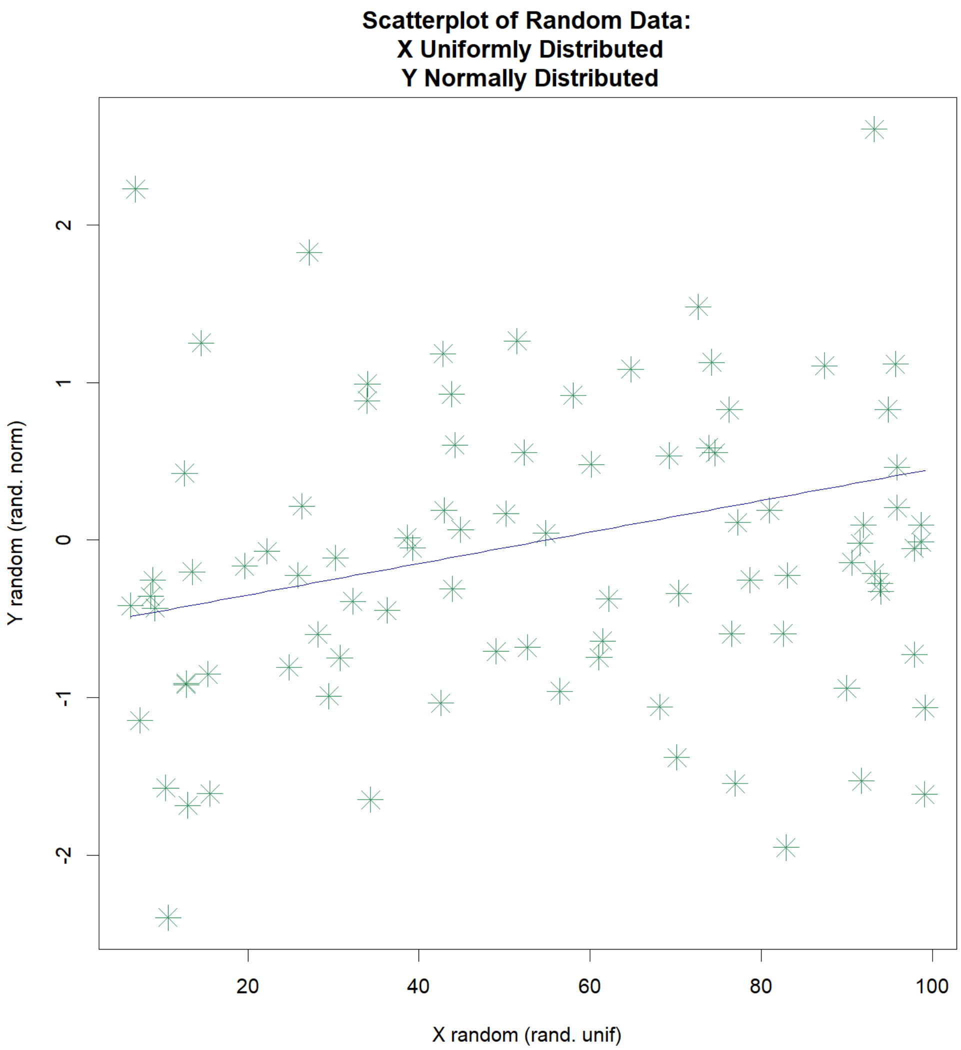
I worked with Sonja to create this dataset using binomial distribution and normal distribution. It is hard to interpret this and more than one dataset as we haven't fully covered these distribution types in lecture yet. We created various types of plots for that dataset as seen below.



Q11. R code you used to create one of the random datasets in your figure:

```
image_file = "lab_04_ques12.png"

png(here("images", image_file), width = 1500, height = 1600, res = 180)

set.seed(6)

n_pts = 90

x_min = 1

x_max = 100

x = runif(n = n_pts, min = x_min, max = x_max)

dat = data.frame(x = x, y_observed = rnorm(n_pts))

plot(dat$x, dat$y_observed, pch = 8, col = "seagreen", cex = 2,

    xlab = "X random (rand. unif)", ylab = "Y random (rand. norm)", main = "Scatterplot of Random
Data:\n X Uniformly Distributed\n Y Normally Distributed")

guess_x = 55

guess_y = 0

guess_slope = 0.01

curve(line_point_slope(x, guess_x, guess_y, guess_slope), add = T, col = "darkblue")

dev.off()
```

Q12. Upload an image file of your figure. It may be in png (raster graphics), svg (vector graphics), or pdf (vector graphics) format. *This will also be uploaded to Moodle



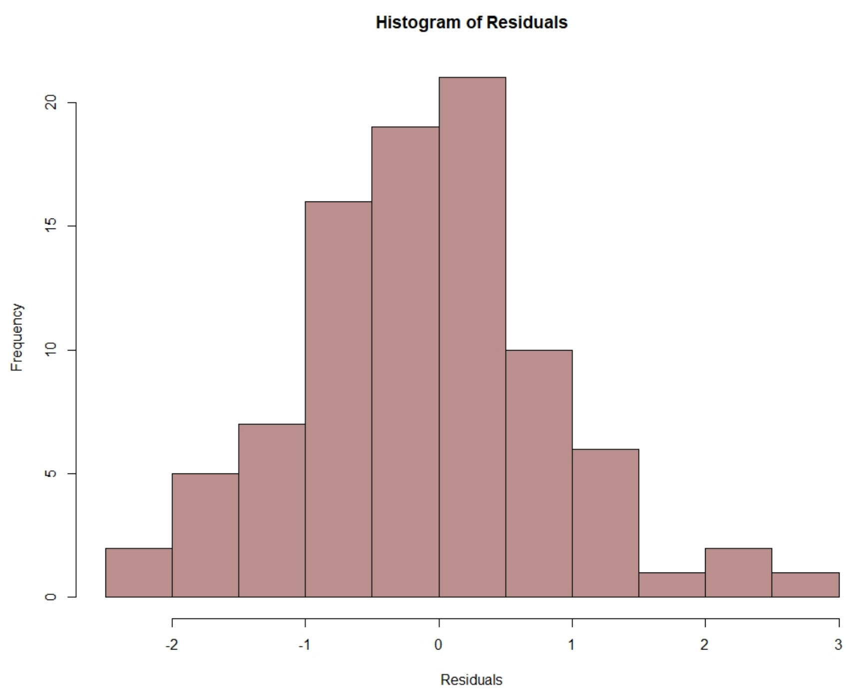
Q13. R code you used to create the columns of predicted values and residuals**:**

dat$y_predicted = line_point_slope(dat$x, guess_x, guess_y, guess_slope)

dat$resids = (dat$y_observed - dat$y_predicted)

Q14. In your report, include the two following figures:

1. A histogram of the model's residuals.



2. A scatterplot of your model's predicted values (on the x-axis) and residuals (on the y-axis)