Jessica Bonin
Analysis of Environmental Data
Lab 4 Report
October 10, 2021
Worked with Juliana Berube, Julia Vineyard, Andrew Gordon

1. Show the code you used to create your vectors

```
lab.mean = 10.4
lab.sd = 2.4

norm_17 = rnorm(n = 17, mean = lab.mean, sd = lab.sd)
norm_30 = rnorm(n = 30, mean = lab.mean, sd = lab.sd)
norm_300 = rnorm(n = 300, mean = lab.mean, sd = lab.sd)
norm_3000 = rnorm(n = 3000, mean = lab.mean, sd = lab.sd)
```

2. Include the R code you used to create your figure. Your answer should include code that builds the figure as well as saves it to a file.

```
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, xlab = "Random Data", ylab = "Frequency", main = "17 Data Points")
hist(norm_30, xlab = "Random Data", ylab = "Frequency", main = "30 Data Points")
hist(norm_300, xlab = "Random Data", ylab = "Frequency", main = "300 Data Points")
hist(norm_3000, xlab = "Random Data", ylab = "Frequency", main = "3000 Data
Points")

dev.off()
```

3. Upload your `lab_04_hist_01.png` file to Moodle. Make sure you double check the image size and resolution requirements.

   Done

4. Qualitatively describe the differences among the histograms.

   With the increase of data points, there is an increase in number of bars on the histogram. In addition, as the number of data points increases, the shape of the histogram becomes more “bell shaped” or normally distributed.

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

   The more data points that are run, the closer the distribution to the normal curve.

6. What are the parameters and their values for the standard Normal distributions?

- Mean = 0
- SD = 1
- Total area under the curve = 1

7. Include the R code you used to create your figure. Your answer should include code that builds the figure as well as saves it to a file.

```
image_file = "norm_1.png"
png(here("images", image_file))

x = seq(0, 20, length.out = 1000)
y = dnorm(x, mean = 10.4, sd = 2.4, log = FALSE)

plot(x, y, main = "Normal PDF: mean = 10.4, sd = 2.4", type = "l", xlim = c(0, 20))
abline(h = 0)

dev.off()
```

8. Upload `norm_1.svg`.

   Done

9. Show the R code you used to create one of the random datasets in your figure.

```
set.seed(97)
n_pts = 100
x_min = 1
x_max = 80
x = runif(n = n_pts, min = x_min, max = x_max)
dat = data.frame(x = x, y_observed = runif(n_pts))
```

10. Upload an image file of your figure.

    Done

11. Show the R code you used to create one of the random datasets in your figure.

```
set.seed(97)
n_pts = 25
x_min = 1
x_max = 200
x = runif(n = n_pts, min = x_min, max = x_max)
dat = data.frame(x = x, y_observed = runif(n_pts))

guess_x = 50
guess_y = 0.3
guess_slope = .003

plot(y_observed ~ x, data = dat, pch = 8, main = "Scatterplot:
Measuring Error", xlab = "x-value", ylab = "Observed y-value")
```

```
curve(line_point_slope(x, guess_x, guess_y, guess_slope), add = T)
```

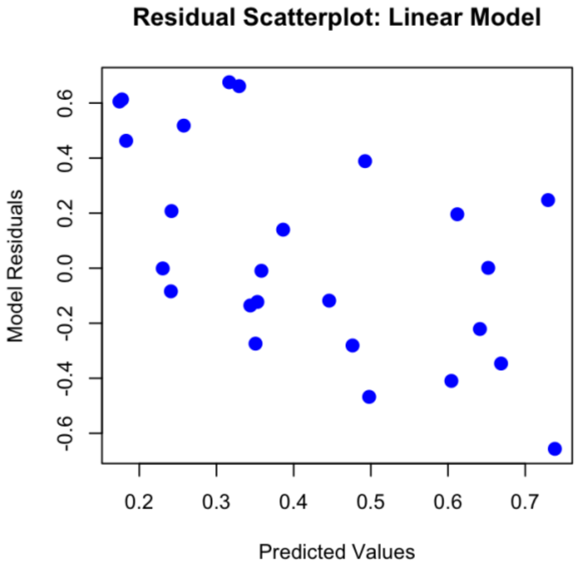
12. Upload an image file of your figure.

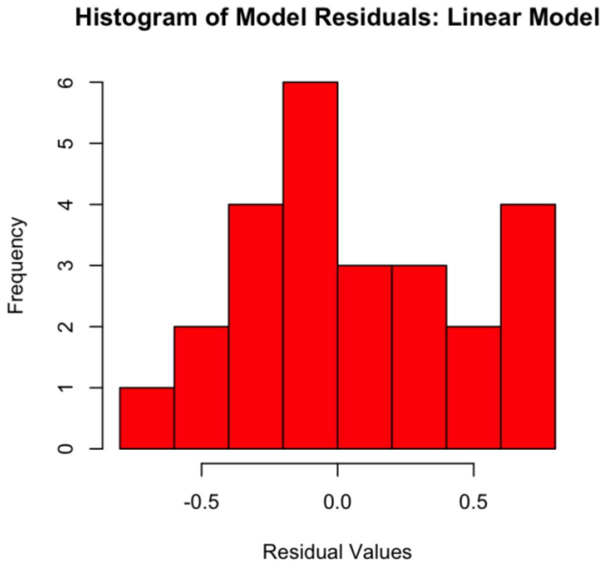
    Done

13. Paste the R code you used to create the columns of predicted values and residuals.

```
line_point_slope(dat$x, guess_x, guess_y, guess_slope)
dat = data.frame(x = x, y_observed = runif(n_pts), y_predicted = 
line_point_slope(dat$x, guess_x, guess_y, guess_slope))
dat = data.frame(x = x, y_observed = rnorm(n_pts),
                 y_predicted = line_point_slope(dat$x, guess_x, guess_y, 
guess_slope),
                 resids = dat$y_observed-dat$y_predicted)
```

14. In your report, include the two following figures
    - A histogram of the model's residuals
    - A scatterplot of your model's predicted values (on the x-axis) and residuals (on the y-axis)


Residual Scatterplot: Linear Model
Model Residuals
Predicted Values


Histogram of Model Residuals: Linear Model
Frequency
Residual Values