Jahiya Clark
Lab 4
Prof. Nelson
10/06/21

Lab 4

1.

```
set.seed(1)

pop_sd = 2.4
pop_mean = 10.4

norm_17 = rnorm(n = 17, mean = pop_mean, sd = pop_sd)
norm_30 = rnorm(n = 30, mean = pop_mean, sd = pop_sd)
norm_300 = rnorm(n = 300, mean = pop_mean, sd = pop_sd)
norm_3000 = rnorm(n = 3000, mean = pop_mean, sd = pop_sd)

png(
  filename = here("images", "lab_04_hist_01.png"),
  width = 1500, height = 1600,
  res = 180)
```

2.

```
par(mfrow = c(2, 2))

hist(norm_17)
hist(norm_30)
hist(norm_300)
hist(norm_3000)
```

3. Upload picture

4. For the histograms with the sample size 17 and 30, the shape is right skewed with tails trailing. Yet, the histograms with the sample size of 300 and 3000 have a more normal bell curve shape.

5. The shape of each graph can easily be attributed to sample size. That is because the more samples to describe the parameter the more the distribution actually reflects the population.

6. mean us 10.4 and sd is 2.4

7.

```
x = seq(-2, 25, length.out = 10000)
y = dnorm(x, mean = 10.4, sd = 2.4)

plot(x, y, main = "Standard Normal PDF: mean=10.4 and sd=2.4 ", type = "l", xlim = c(0, 20),
ylim = c(0,0.2))
abline(h = 0)

pdf(
  file = here( "images", "norm_1.pdf"),
  )
```

8. upload the photo

9.

```
n_pts = 100
x_min = 0
x_max = 10
x = runif(n = n_pts, min = x_min, max = x_max)
dat = data.frame(x = x, y_observed = rnorm(n_pts))
plot(dat, ylim= c(-0.75, 1.5), main = "Plot of Random Data", pch = 10, cex = 1.3, col =
"#00FF7F")
```

10. upload the photo

11.

```
n_pts = 100
x_min = 0
x_max = 10
x = runif(n = n_pts, min = x_min, max = x_max)
dat = data.frame(x = x, y_observed = rnorm(n_pts))
plot(dat, ylim= c(-0.75, 1.5), main = "Plot of Random Data", pch = 10, cex = 1.3, col =
"#00FF7F")


line_point_slope = function(x, x1, y1, slope)
{
  get_y_intercept =
    function(x1, y1, slope)
      return(-(x1 * slope) + y1)

  linear =
    function(x, yint, slope)
      return(yint + x * slope)
```

```
  return(linear(x, get_y_intercept(x1, y1, slope), slope))
}

x= dat$x
x1= 5
y1 = 0.4
slope1= 0.2
```

12. Upload image

13.

```
line_point_slope(x, x1, y1, slope1)

dat$y_predicted= c(line_point_slope(x, x1, y1, slope1))

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

14. upload picture