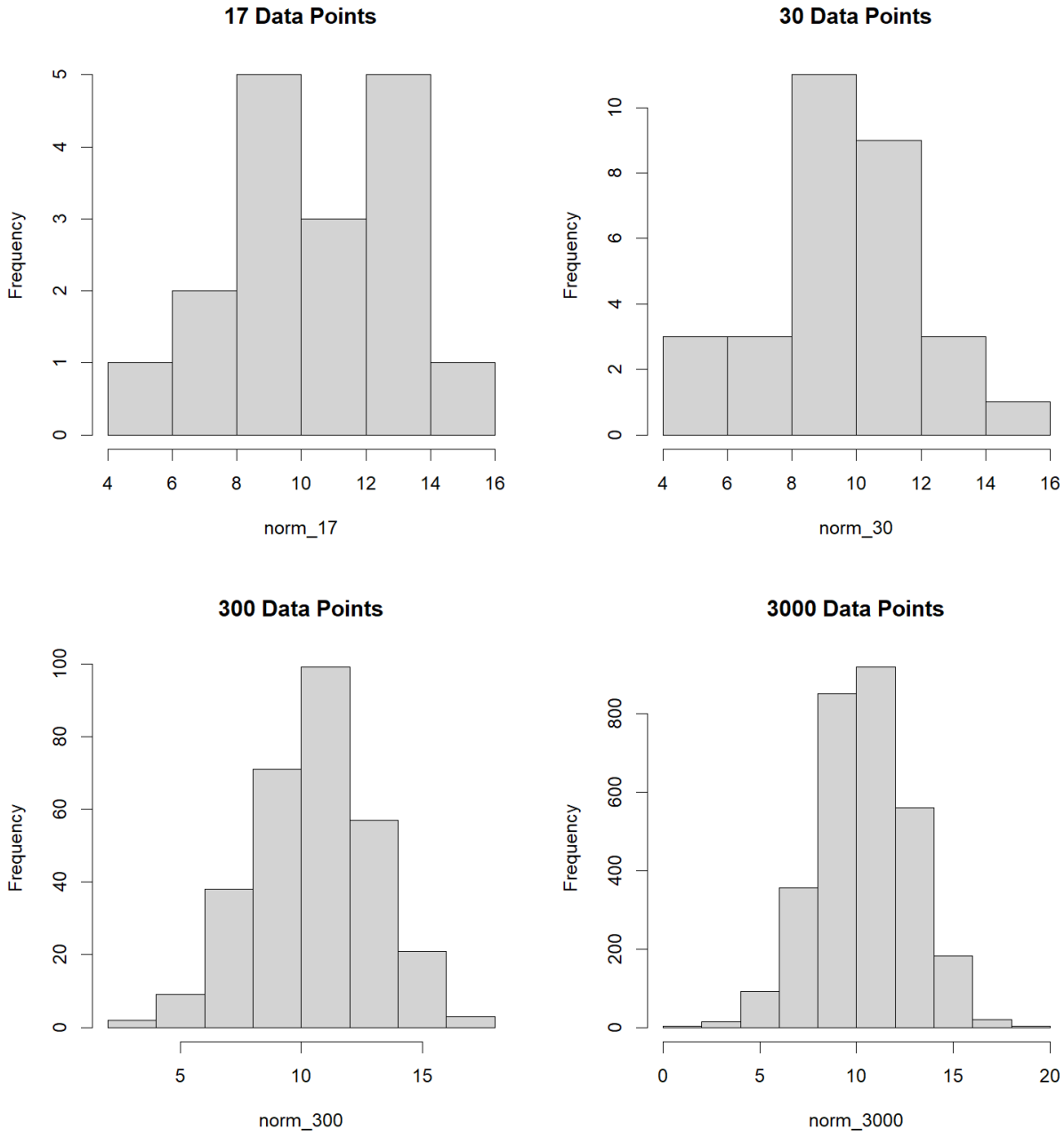
9/29/2021
Heather Siart
ECO-634-Environmental Data Analysis Lab
Group members: Matt, Mandi, Erika, Jahiya, Bonnie

Lab 4 Uncertainty and Error

Q1:
```
# mean 10.4, standard deviation 2.4, elements are the number in the name
m = c(10.4)
d = c(2.4)

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

Q2:
```
par(mfrow = c(2, 2))

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

png(filename = here("images","lab_04_hist_01.png"), width =1500, height = 1600, units = "px",
res = 180)
par(mfrow = c(2,2))
hist(norm_17, main = "17 Data Points")
hist(norm_30, main = "30 Data Points")
hist(norm_300, main = "300 Data Points")
hist(norm_3000, main = "3000 Data Points")

dev.off()
```

Q3:



Q4: As you add more samples and increase the sample size, the histogram begins to look more like a normal distribution. The more samples you have the more your data starts to round out to give you a better picture of the distribution. With a smaller sample size, you have more sample error, and it shows in the graph. The more samples the better!

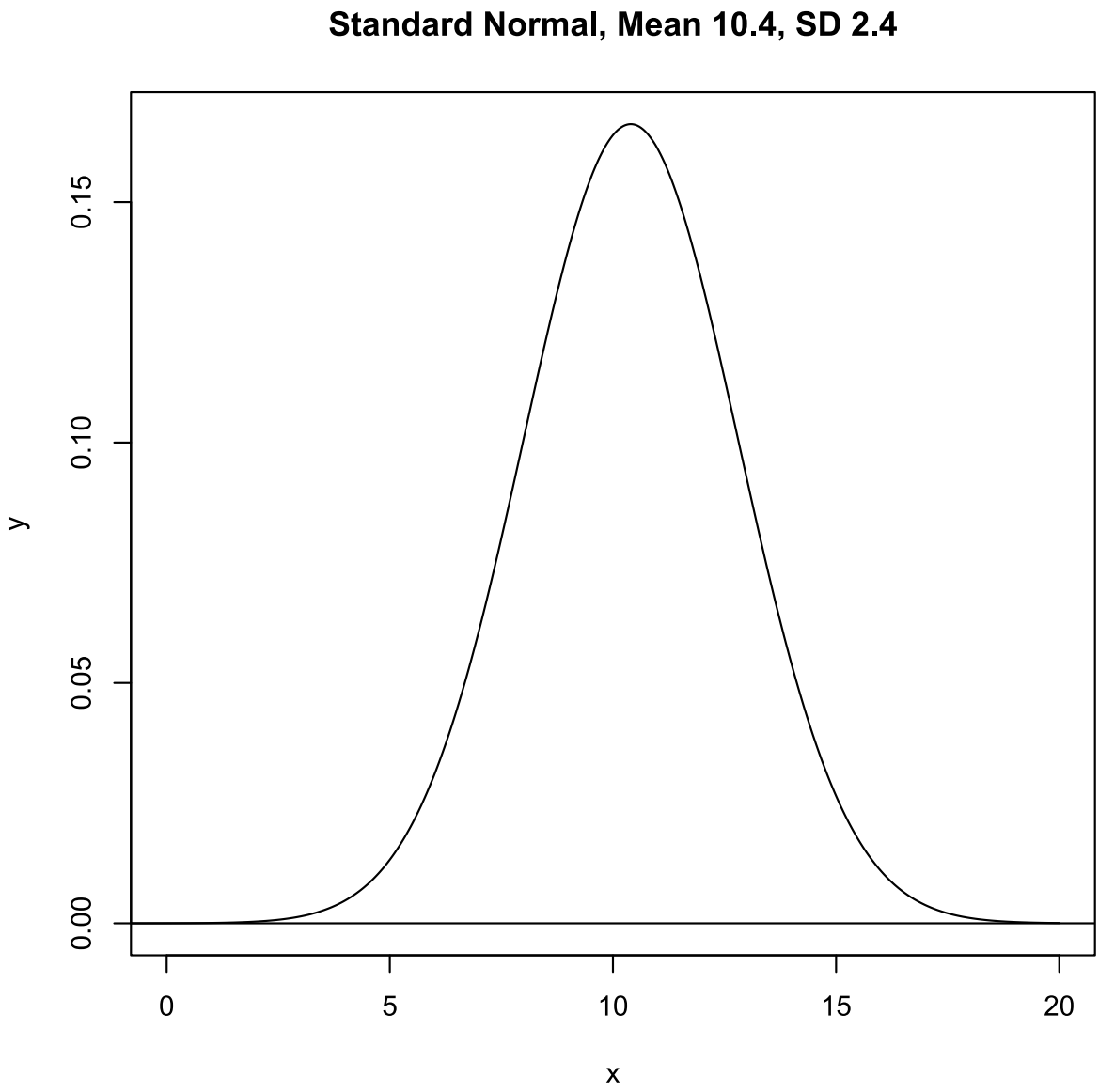
Q5: The shapes of the histograms are different because you have a larger sample size. Having more samples from a population allows you to have a better understanding of the population.

Q6: The parameters and their values for the standard normal distribution is the mean and the standard deviation. For the histograms above the mean is 10.4 and the standard deviation is 2.4.

Q7: svg(filename = here("images","norm_1.svg"))

```
require(here)
x = seq(-20, 20, length.out = 1000)
y = dnorm(x, mean = 10.4, sd = 2.4)

plot(x, y, main = "Standard Normal, Mean 10.4, SD 2.4", type = "l", xlim = c(0, 20))
abline(h = 0)

dev.off()
```

Q8:

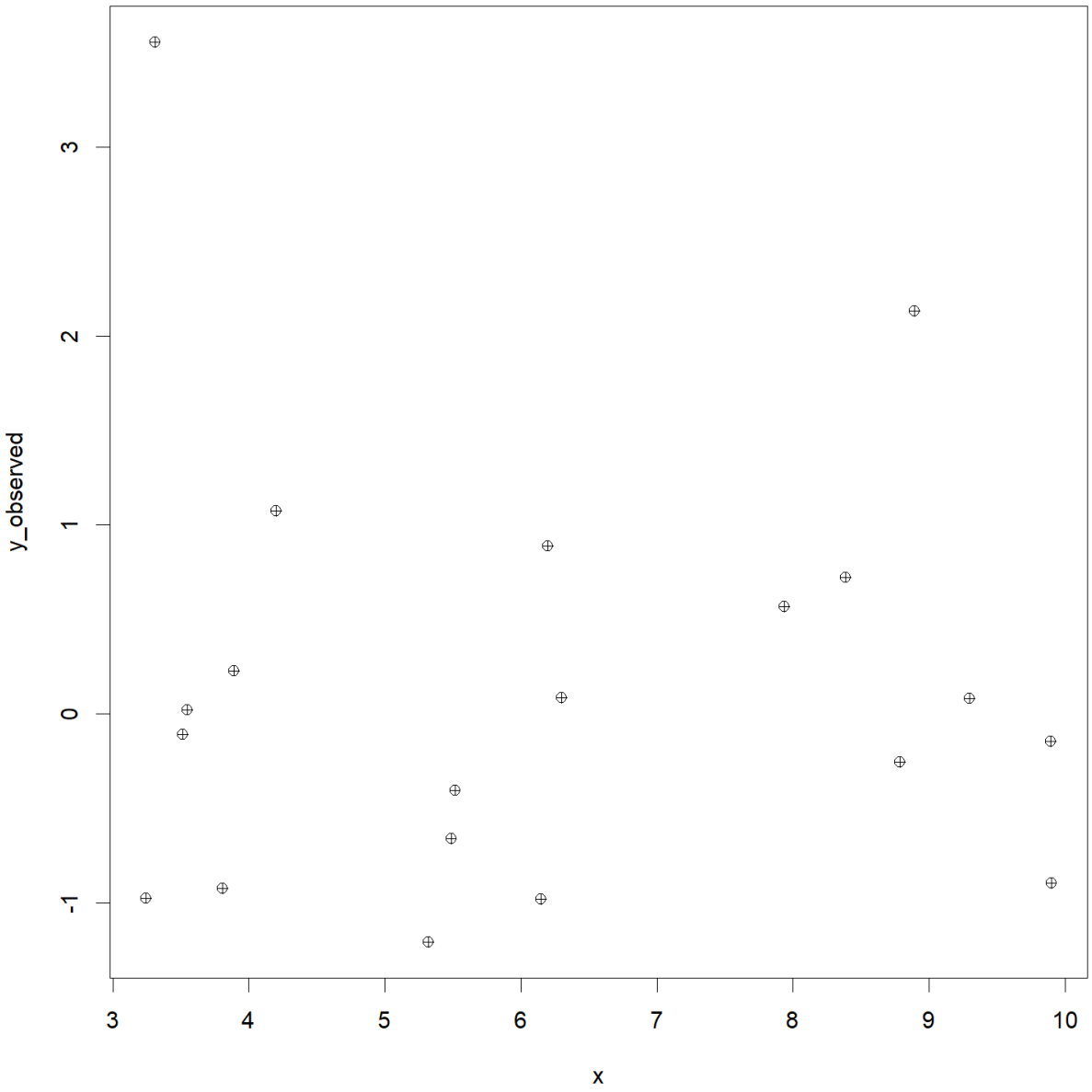


Q9:

```
png(filename = here("images","lab_04_hist_05.png"), width =1500, height = 1600, units = "px",
res = 180)
require(here)
```

```
n_pts = 20
x_min = 3
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(y_observed ~ x, data = dat, pch = 10)

guess_x = 6
guess_y = 0
guess_slope = 0.1

dev.off()
```

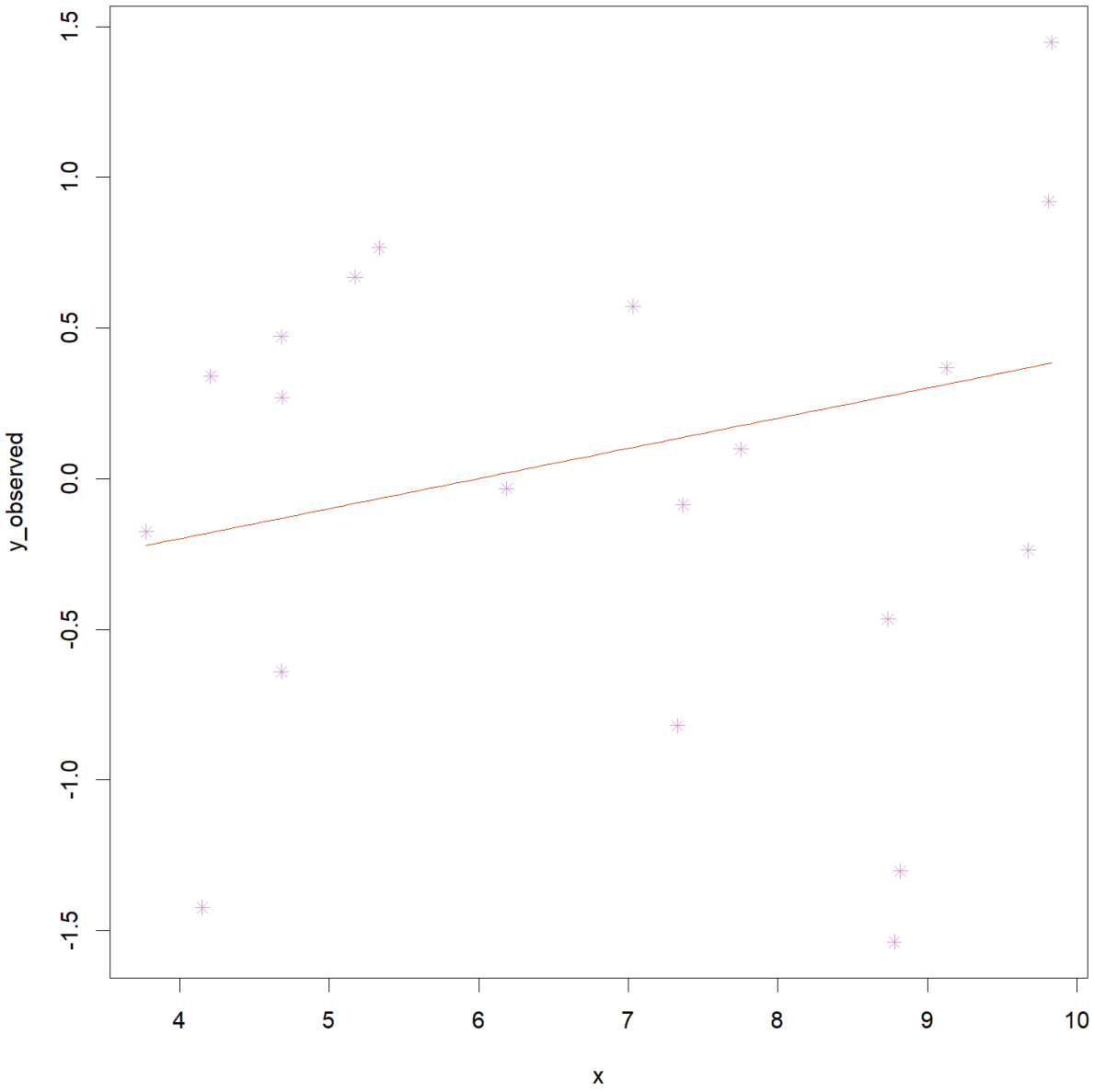
Q10:



Q11:
```
png(filename = here("images","lab_04_hist_03.png"), width =1500, height = 1600, units = "px",
res = 180)
require(here)
n_pts = 20
x_min = 3
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(y_observed ~ x, data = dat, pch = 10)

guess_x = 6
guess_y = 0
guess_slope = 0.1

plot(y_observed ~ x, data = dat, pch = 8, col = "plum3")
curve(line_point_slope(x, guess_x, guess_y, guess_slope), add = T, col = "orangered3")

dev.off()
```

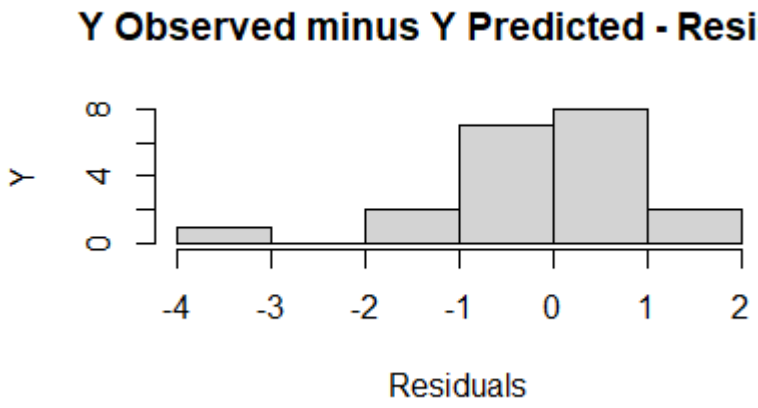
Q12:



Q13:

```
guess_x = 15
guess_y = 0
guess_slope = 0.3

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

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

dat$y_predicted = y_predicted
```

dat$resid = dat$y_predicted - dat$y_observed
dat



Q14: