Megan Mitchell
ECO 634
October 10th, 2021

Lab 04: Uncertainty and Error

#1

```
norm_17 = rnorm(n= 17, mean= 10.4 , sd = 2.4)
norm_30 = rnorm(n= 30, mean= 10.4, sd = 2.4)
norm_300 = rnorm(n= 300, mean= 10.4, sd = 2.4)
norm_3000 = rnorm(n= 3000, mean= 10.4 , sd = 2.4)
```

#2

```
png(file="C:/Users/Megan Mitchell/OneDrive - University of
Massachusetts/Environmental_Data/ECO_634_Data/Lab_04_Hist_01.png",
    width=1500, height=1600)
par(mfrow = c(2,2))
hist(norm_17, main = "Histogram with 17 observations", xlab= "")
hist(norm_30, main = "Histogram with 30 observations",  xlab= "")
hist(norm_300, main = "Histogram with 300 observations",  xlab= "")
hist(norm_3000, main = "Histogram with 3000 observations",  xlab= "")
dev.off()
```

#3 (Uploaded on Moodle)

#4

The shapes of the four histograms all differ from one another because of differences in sample size, n. The histograms with a sample size of n=300 and n=3000 have a somewhat bell-shaped distribution, although n=3000 is more normally distributed versus the left skew in n=300. The histogram with n=30 is a bell that is very skewed to the left whereas the plot with n=17 is somewhat left-skewed and bi-model shaped.

#5

The differences in distribution across histograms may be attributable to the variation in sample size, n. The histogram titled "Histogram with 3000 observations" has a much larger sample size than the first three histograms. As sample size increases, the deviation of samples away from the mean (10.4) decreases and data becomes normally distributed. The "Histogram with 300 observations" begins to approach normal distribution because of a larger sample size (n=300), but still has more variation than the plot with n=3000 observations. Normal distribution was not the case for the histograms with n=17 and n=30 observations because small sample sizes can lead to more sample variation away from the mean. However, we can see that a sample size of n=30 is more evenly distributed than a sample size of n=17.

#6

The standard values and parameters for a normal distribution is mean =0 and standard deviation (sd) = 1. In our lab activity, the mean = 10.4 and the sd = 2.4.

#7

```
svg(filename = if(onefile) "C:/Users/Megan Mitchell/OneDrive - University of
Massachusetts/Environmental_Data/ECO_634_Data/norm_1.svg",
        width=7, height=7, pointsize = 12,
        onefile = FALSE, family = "sans", bg = "white",
        antialias = c("default", "none", "gray", "subpixel"),
        symbolfamily)

svg("norm_1.svg")
x = seq(-50, 50, length.out = 1000)
y = dnorm(x, mean = 10.4, sd = 2.4)
plot(x, y, main = "Standard Normal PDF, mean = 10.4, sd = 2.4", type = "l", xlim = c(2,
18))
abline(h = 0)
dev.off()
```

#8 (uploaded on Moodle)

#9

```
set.seed(711)
n_pts = 25
x_min = 0
x_max = 25
x = runif(n = n_pts, min = x_min, max = x_max)

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

png(file="C:/Users/Megan Mitchell/OneDrive - University of
Massachusetts/Environmental_Data/ECO_634_Data/My_4_plots.png", width=1500,
height=1600)

plot(y_observed ~ x, data = dat, pch = 5, main = "Megan's Second Scatterplot", xlab = "",
ylab= "", col = "orange")
```

#10 (uploaded on Moodle)

#11

```
set.seed(299)
n = 50
x_min = 1
x_max = 3

x = runif(n = n, min = x_min, max = x_max)
dat = data.frame(x = x, y_observed = rnorm(n))

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))
}
plot(y_observed ~ x, data = dat, pch = 15, main = "Megan's scatterplot with fitted linear
model", xlab = "", ylab= "", col = "maroon3")
guess_x = 1.1
guess_y = 0.1
guess_slope = 0.9
curve(line_point_slope(x, guess_x, guess_y, guess_slope), add = TRUE)
```

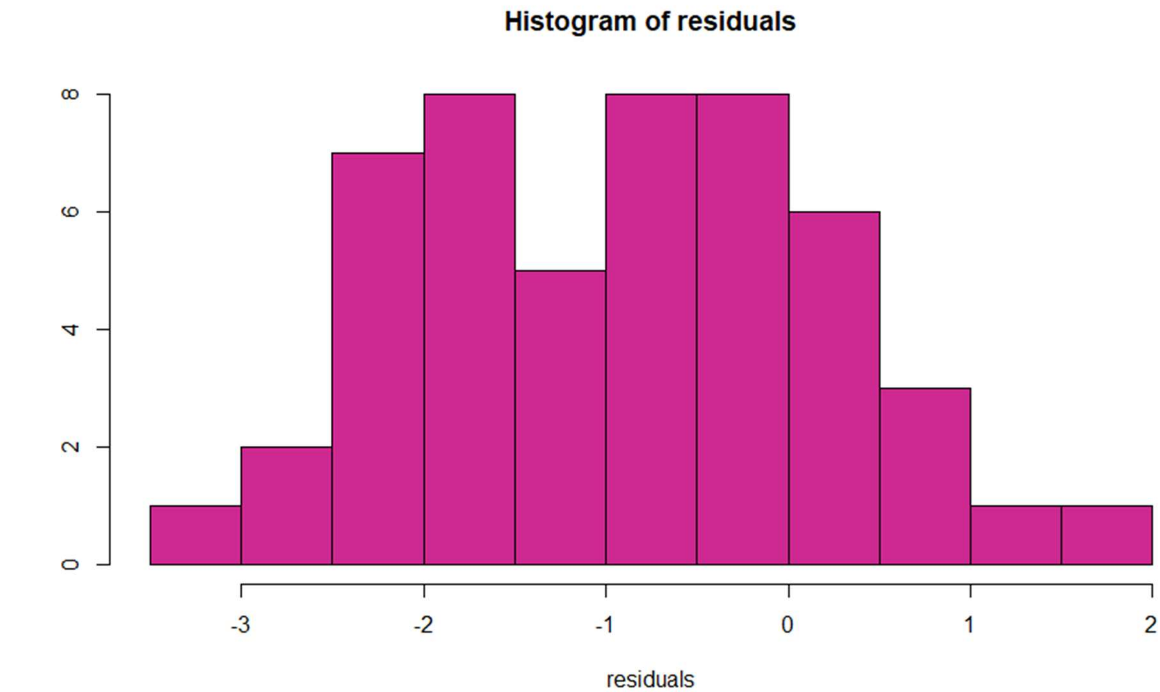
#12 (uploaded on Moodle)

#13

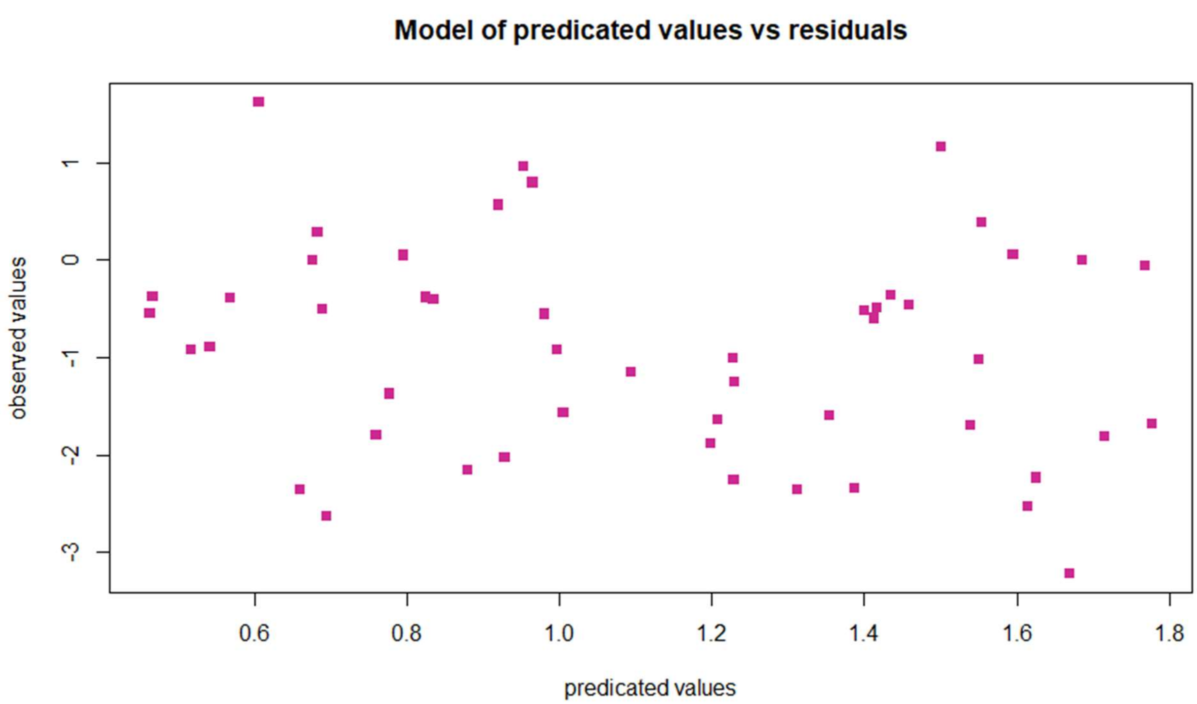
```
y_predicted = line_point_slope(dat$x, guess_x, guess_y, guess_slope)
dat$resids = (dat$y_observed - dat$y_predicted)
sum(dat$resids)
dat$abs_resids = abs(dat$resids)
dat$abs_resids

hist(dat$resids, main = "Histogram of residuals", xlab = "residuals", ylab = "",
col="maroon3")
plot(dat$y_predicted, dat$resids, pch = 15, main = "Model of predicated values vs
residuals", xlab= "predicated values", ylab= "observed values", col="maroon3")
```

#14



Histogram of residuals
0
2
4
6
8
-3
-2
-1
0
1
2
residuals



Model of predicated values vs residuals
observed values
-3
-2
-1
0
1
0.6
0.8
1.0
1.2
1.4
1.6
1.8
predicated values