Assignment 1: Basic R

Emily Parsons

January 9, 2020

1. What are the names of the columns in this dataset?

```
tgpp <-read.csv("C:/Users/Emily/Desktop/Stats Class/Data/tgpp.csv", header = TRUE)
```

```
head(tgpp)
```

	plot <int>	year <int>	record_id <int>	corner <int>	scale <dbl>	richness <int>	easting <int>	northing <int>	slope <int>
1	205	1998	187	*NA*	100	60	727000	4080000	3
2	205	1998	188	1	10	36	727000	4080000	3
3	205	1998	189	2	10	34	727000	4080000	3
4	205	1998	190	3	10	37	727000	4080000	3
5	205	1998	191	4	10	33	727000	4080000	3
6	205	1998	192	1	1	21	727000	4080000	3

6 rows | 1-10 of 12 columns

The names of the columns are: plot, year, record_id, corner, scale, richness, easting, northing, slope, ph, yrsslb

2. How many rows and columns does this data set have?

```
nrow(tgpp)
```

```
## [1] 4080
```

```
ncol(tgpp)
```

```
## [1] 11
```

This data set has 4080 rows and 11 columns

3. What kind of object is each data column?

```
tgpp <-read.csv("C:/Users/Emily/Desktop/Stats Class/Data/tgpp.csv", header = TRUE)
sapply(tgpp, class)
```

```
##      plot      year record_id    corner     scale  richness   easting
## "integer" "integer" "integer" "integer" "numeric" "integer" "integer"
##  northing     slope        ph    yrsslb
## "integer" "integer" "numeric" "numeric"
```

Columns 1-4 and 5-9 are integers, column 5,10, and 11 are numeric. ‘Class’ tells you the type of object your data is and sapply ran this function across the values in the different columns in the data set.

4. What are the values of the datafile for rows 1, 5, and 8 at columns 3, 7, and 10?

```
tgpp[1,3]
```

```
## [1] 187
```

```
tgpp[1,7]
```

```
## [1] 727000
```

```
tgpp[1,10]
```

```
## [1] 6.9
```

```
tgpp[5,3]
```

```
## [1] 191
```

```
tgpp[5,7]
```

```
## [1] 727000
```

```
tgpp[5,10]
```

```
## [1] 6.9
```

```
tgpp[8,3]
```

```
## [1] 194
```

```
tgpp[8,7]
```

```
## [1] 727000
```

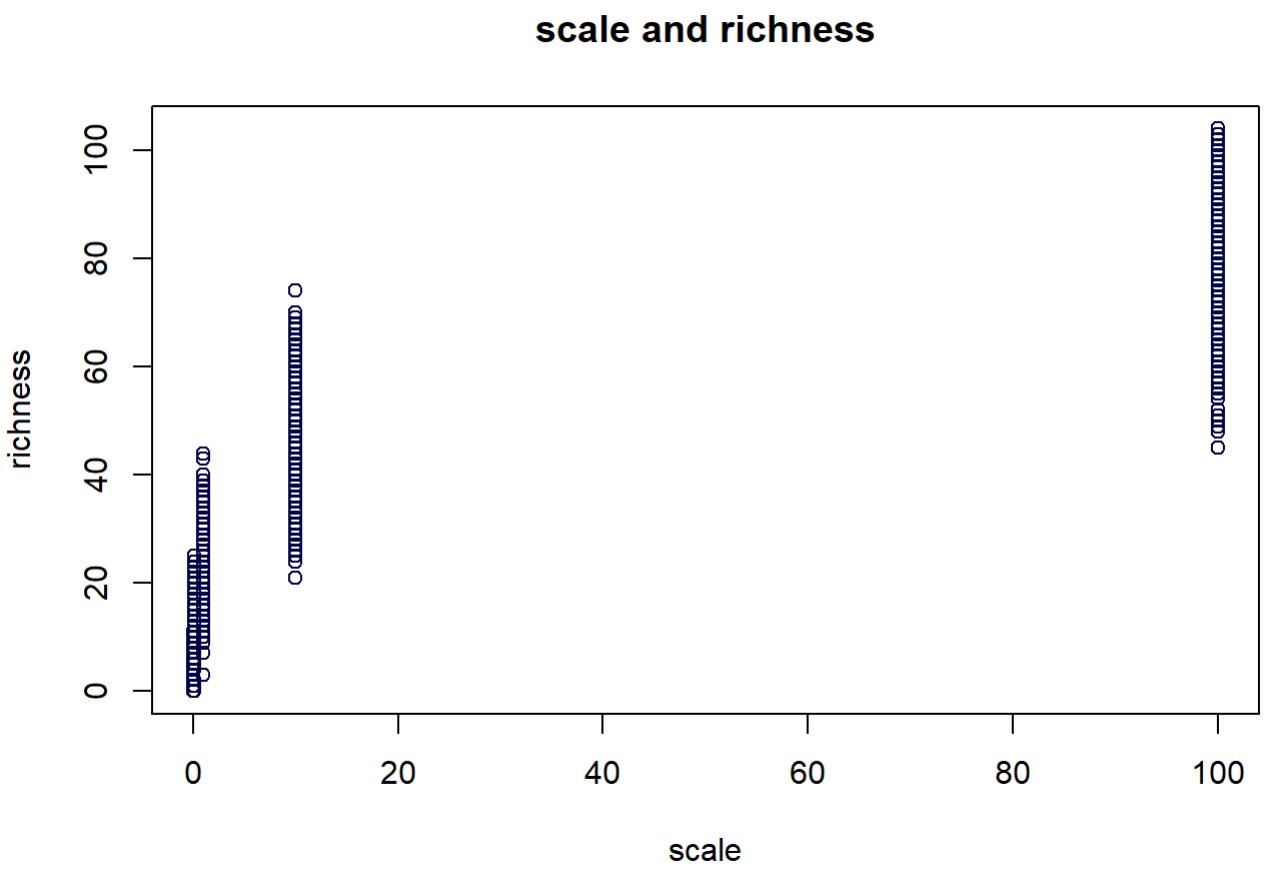
```
tgpp[8,10]
```

```
## [1] 6.9
```

See output above. Answers found through subsetting the data from the established dataframe

5. Create a pdf of the relationship between the variables “scale” and “richness”. Scale is the area in square meters of the quadrat in which richness was recorded. Be sure to label your axes clearly, and choose a color you find pleasing for the points. To get a list of available stock colors use the function colors(). Also see this link:

```
x <- tgpp$scale
y <- tgpp$richness
graph <- data.frame(x,y)
plot(graph$x, graph$y, main= "scale and richness", xlab='scale', ylab='richness', col="#000444")
```



1. What happens when you set the plot argument to . . . log = “xy”

```
graph <- data.frame(x,y)
head(graph)
```

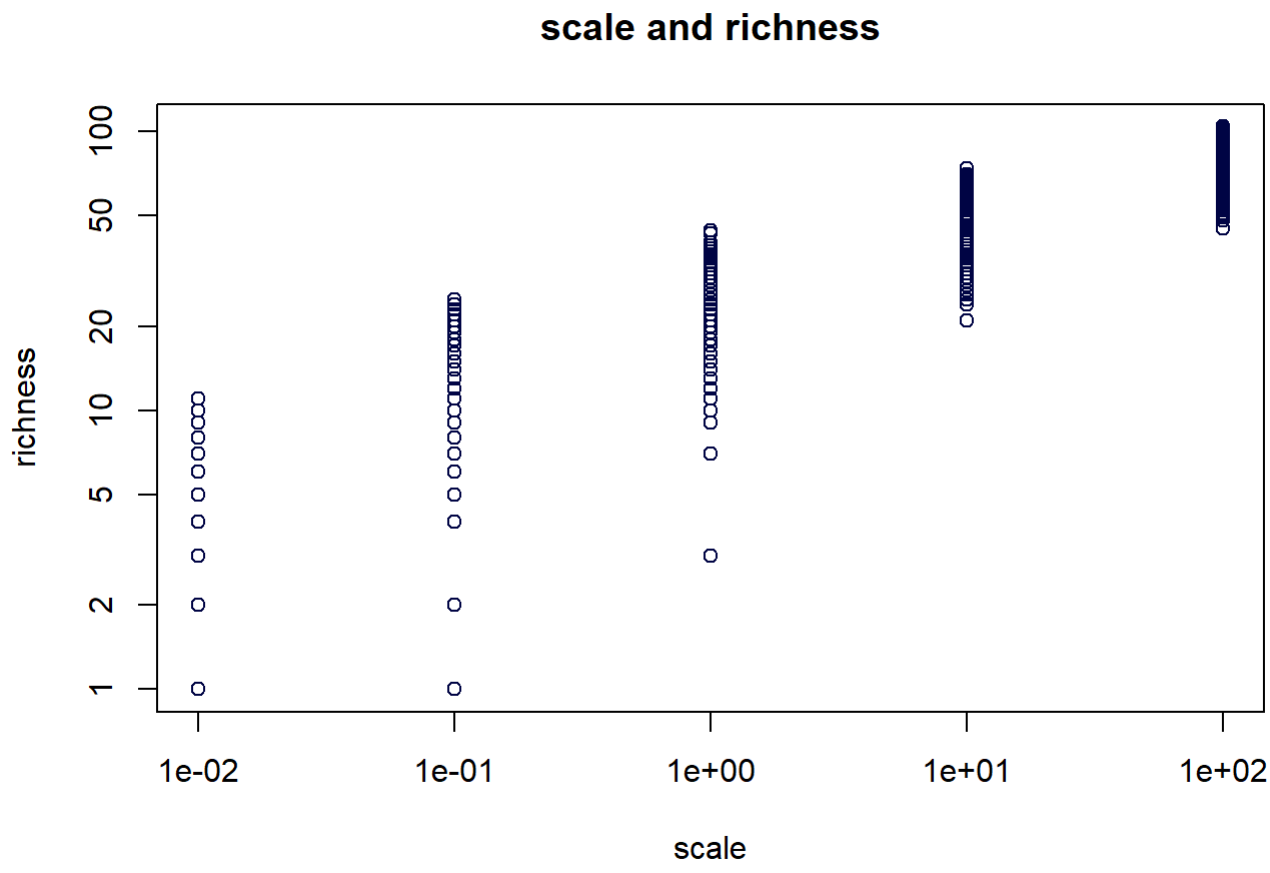
	x	y
	<dbl>	<int>

	x <dbl>	y <int>
1	100	60
2	10	36
3	10	34
4	10	37
5	10	33
6	1	21

6 rows

```
plot(graph$y~graph$x, main= "scale and richness", xlab='scale', ylab='richness', col="#000444",
 log="xy")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 4 y values <= 0 omitted
## from logarithmic plot
```



```
pdf('./PDFs/inflammation_fig1.pdf')
x <- 1:length(tgpp$scale)
y <- tgpp$richness
plot(x, y, main= "scale and richness", xlab='scale', ylab='richness', col="#000444")

pdf('./PDFs/inflammation_fig2.pdf')
x <- 1:length(tgpp$scale)
y <- tgpp$richness
plot(x, y, main= "scale and richness", xlab='scale', ylab='richness', col="#000444", log="xy")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 4 y values <= 0 omitted
## from logarithmic plot
```

```
tgpp <-read.csv("C:/Users/Emily/Desktop/Stats Class/Data/tgpp.csv", header = TRUE)
```