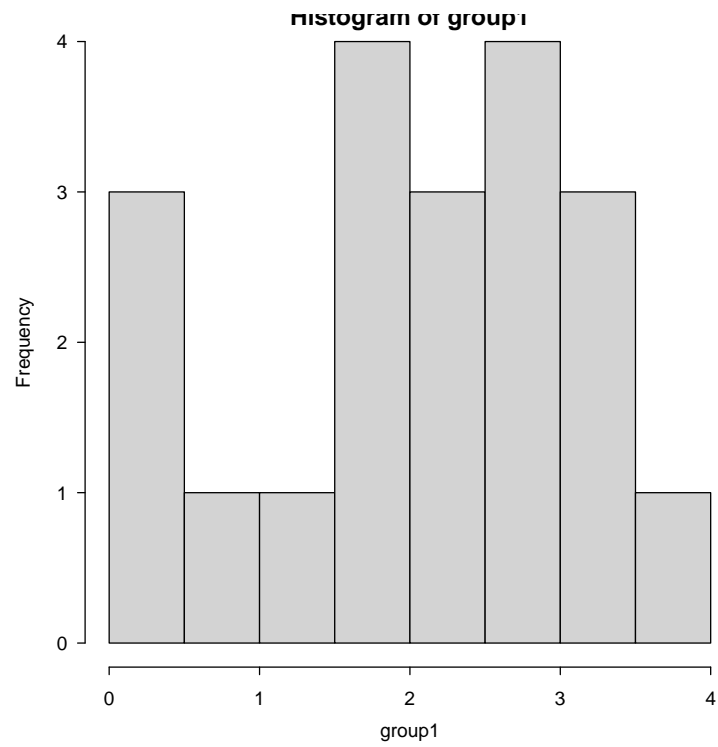


March 16, 2022

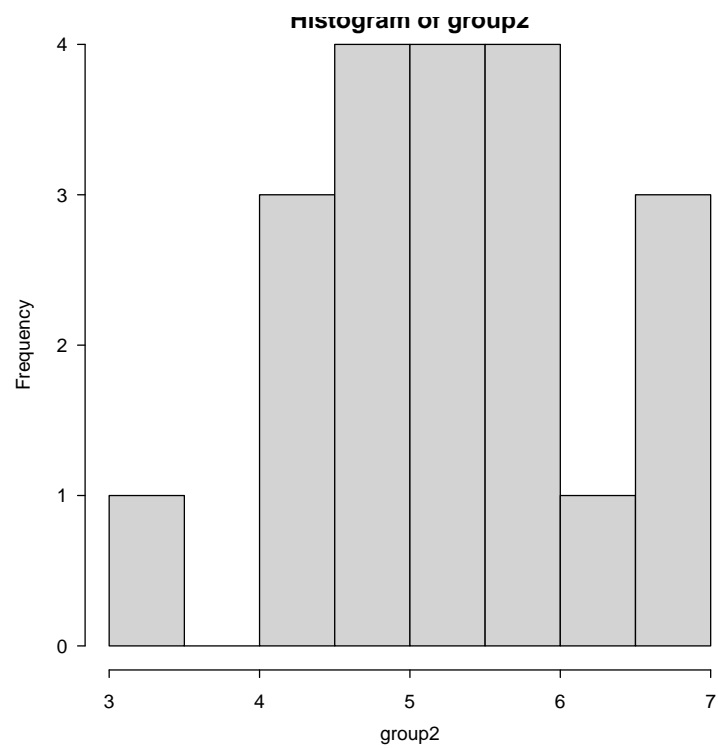
The results below are generated from an R script.

```
#####  
# Bio 381, HW 8: Simulating Data #  
# 14 Mar 2022 #  
# E.M. Beasley #  
#####  
  
library(boot)  
  
# Why sim data?  
# Saves time- write parts of your code before you have data  
# Baseline for comparisons: check assumptions in your data  
# Test new stats techniques (this is less common)  
  
# Part 1: Normally distributed data -----  
# Start with groups of normally distributed data  
# For t-tests or ANOVA  
  
# simulate groups of 20 observations  
group1 <- rnorm(n = 20, mean = 2, sd = 1)  
hist(group1) # look at distribution
```

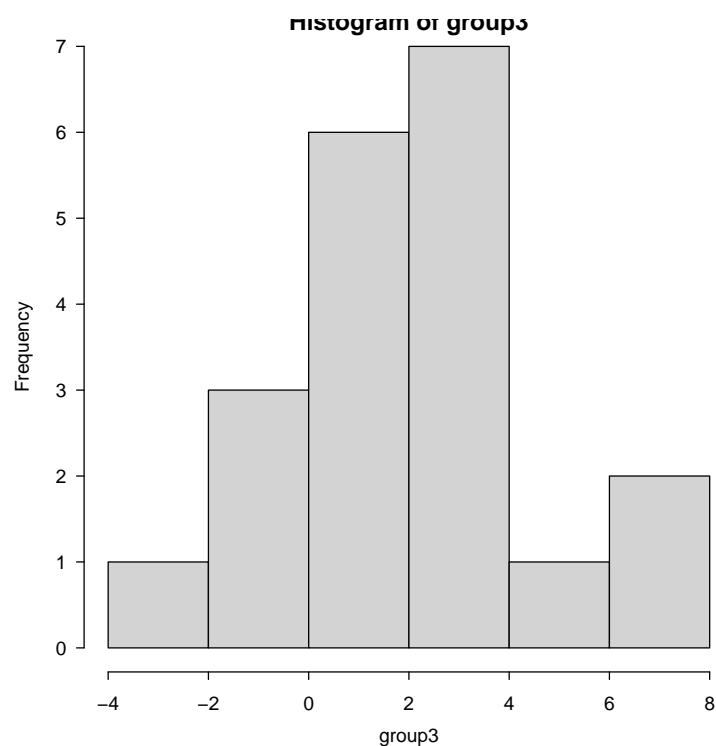


```
# change up some parameters
group2 <- rnorm(n = 20, mean = 5, sd = 1)
group3 <- rnorm(n = 20, mean = 2, sd = 3)

hist(group2)
```



```
hist(group3)
```



```
# You will work more with grouped data on the homework
```

```
# Data sim for simple linear regression  
# Assume slope of 0, so y = beta1*x  
# where beta1 is your slope  
# and x is your environmental covariate
```

```
# slope will be constant:
```

```
beta1 <- 1
```

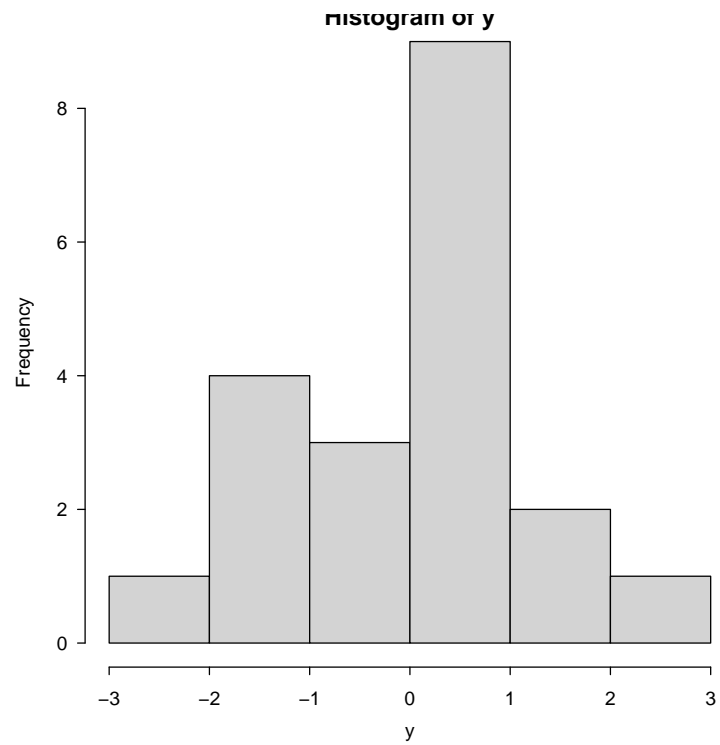
```
# sim the covariate:
```

```
x <- rnorm(n = 20)
```

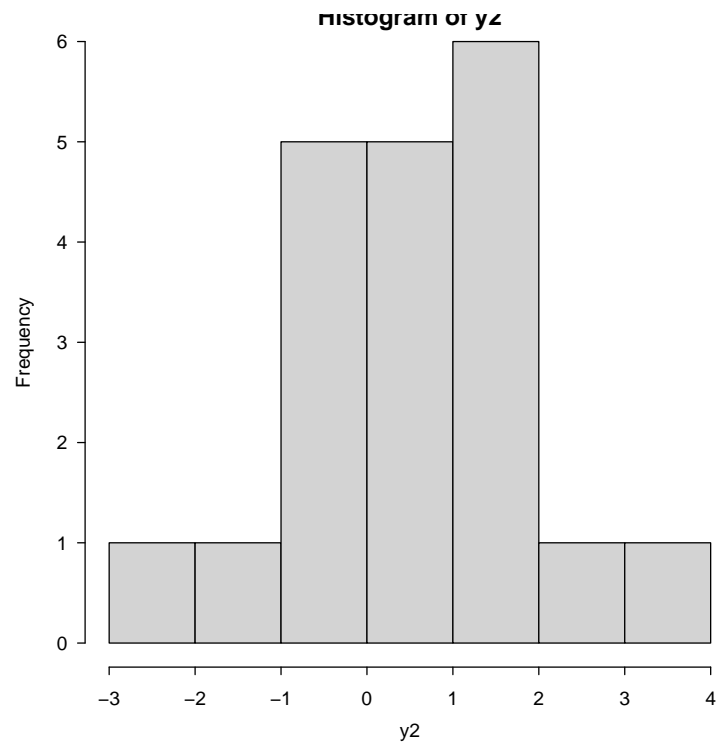
```
# now use the above to create a response variable:
```

```
y <- beta1*x
```

```
hist(y)
```

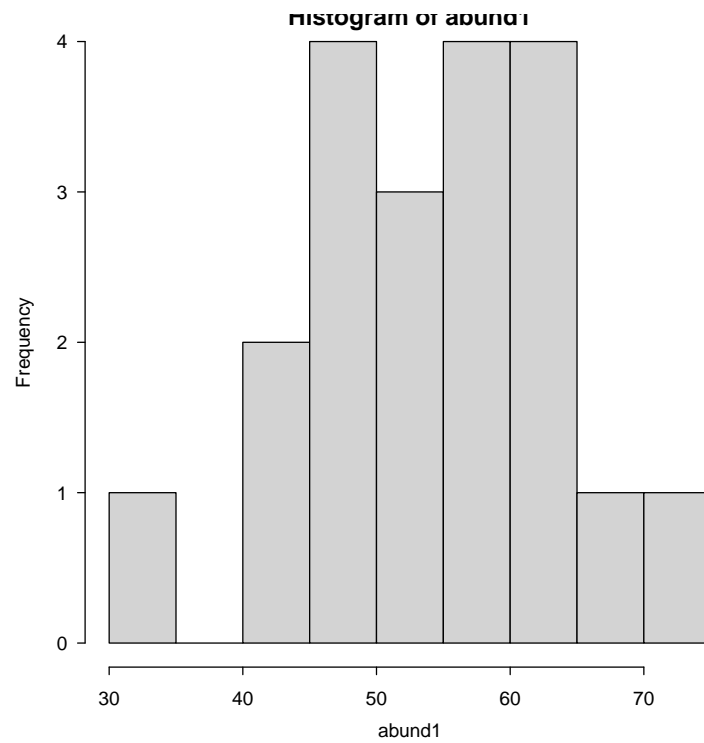


```
# you can add complexity by adding intercepts or more covariates:  
beta0 <- 0.5  
  
# add intercept beta0  
y2 <- beta0 + beta1*x  
hist(y2)
```

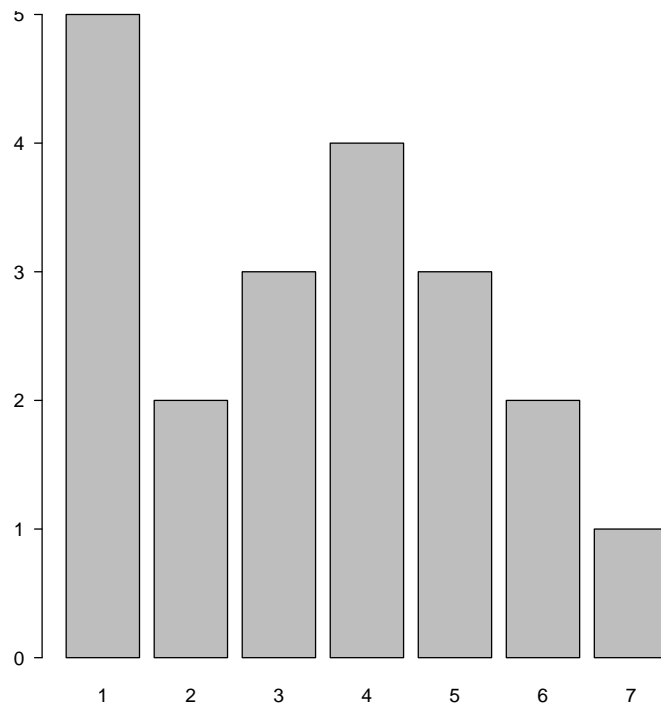


```
# You can also play with different slopes
# or different distributions for the intercept/covariates

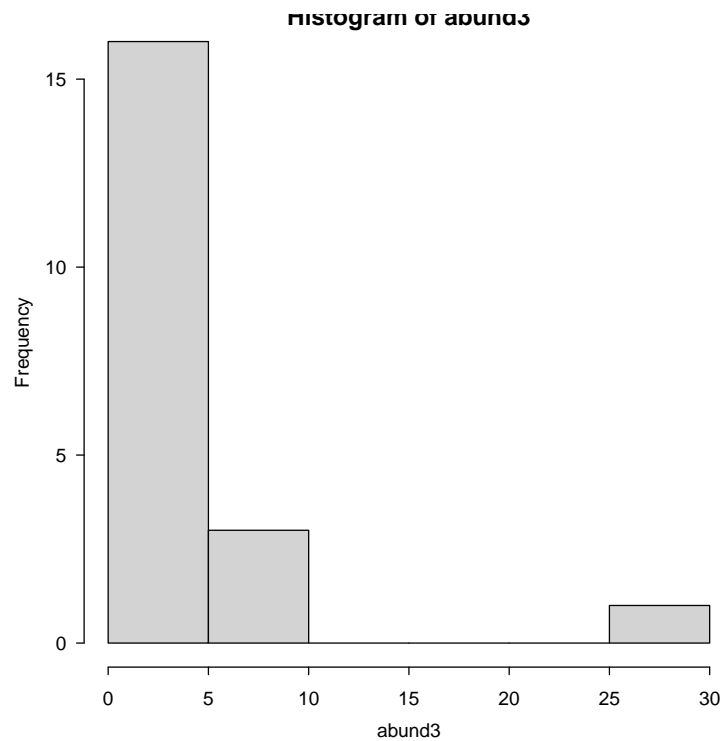
# Part 2: Abundance/count data -----
# Option 1: data are normal-ish
# use round() to get whole numbers
abund1 <- round(rnorm(n = 20, mean = 50, sd = 10))
hist(abund1)
```



```
# this only works if sd is sufficiently large  
# and rnorm unlikely to get negative numbers  
  
# A better way: use Poisson distribution  
# Simulate counts from the same distribution  
# where lambda = typical abundance  
abund2 <- rpois(n = 20, lambda = 3)  
barplot(table(abund2))
```



```
# Sometimes the environment affects abundance/counts  
# When that happens, first generate lambdas  
# then use those to get abundances  
  
# use regression to get initial values  
pre.lambda <- beta0+beta1*x  
# use inverse log to make lambdas positive  
lambda <- exp(pre.lambda)  
  
# use these lambda values to get abundances/counts:  
abund3 <- rpois(n = 20, lambda = lambda)  
hist(abund3)
```

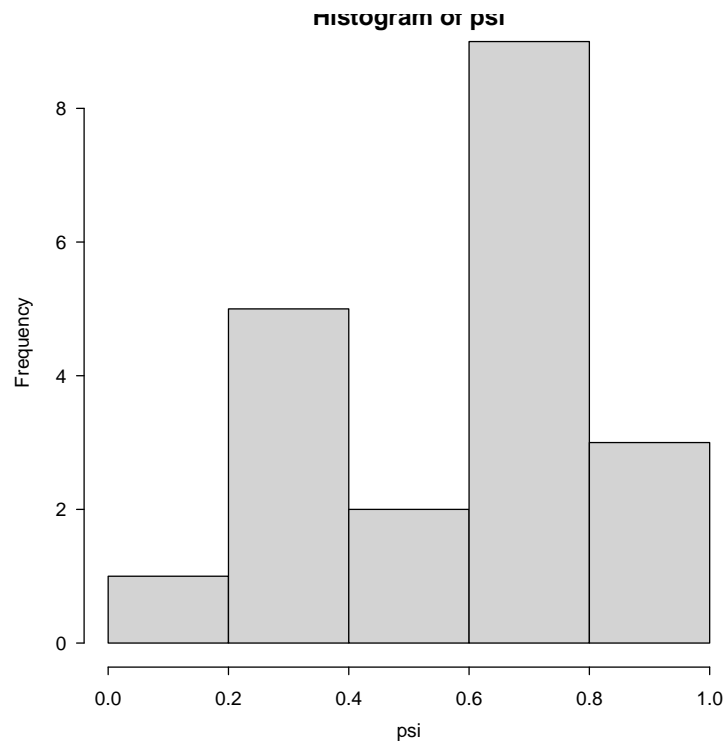


```
# Part 3: Occupancy/presence-absence data -----
# Option 1: getting probabilities from a beta distribution
probs <- rbeta(n = 20, shape1 = 1, shape2 = 1)
occ1 <- rbinom(n = 20, size = 1, prob = probs)
print(occ1)

## [1] 0 0 0 1 0 0 1 1 1 0 1 0 0 1 0 1 0 0 0 1

# Occupancy depends on environment too
# We can do something similar to above
# Except we're calculating probabilities, not lambdas

pre.probs <- beta0 + beta1*x
psi <- inv.logit(pre.probs) # inverse logit to put on 0-1 scale
hist(psi) # good distribution of probs
```

```
# use rbinom again to get occupancy data
occ2 <- rbinom(n = 20, size = 1, prob = psi)
print(occ2)

## [1] 1 1 0 1 1 1 0 0 0 0 0 1 0 1 1 1 1 0 1 0
```

The R session information (including the OS info, R version and all packages used):

```
sessionInfo()

## R version 4.1.2 (2021-11-01)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19044)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252 LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252 LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods    base
##
## other attached packages:
## [1] boot_1.3-28 knitr_1.36
##
## loaded via a namespace (and not attached):
## [1] compiler_4.1.2 magrittr_2.0.1 tools_4.1.2    tinytex_0.35  stringi_1.7.6
## [6] highr_0.9       stringr_1.4.0 xfun_0.27      evaluate_0.14
```

```
Sys.time()
```

```
## [1] "2022-03-16 10:50:28 EDT"
```