# SDM4 in R: Sampling Distribution Models (Chapter 17)

*Nicholas Horton (nhorton@amherst.edu) and Sarah McDonald*

*June 5, 2018*

## Introduction and background

This document is intended to help describe how to undertake analyses introduced as examples in the Fourth Edition of *Stats: Data and Models* (2014) by De Veaux, Velleman, and Bock. More information about the book can be found at http://wps.aw.com/aw_deveaux_stats_series. This file as well as the associated R Markdown reproducible analysis source file used to create it can be found at http://nhorton.people.amherst.edu/sdm4.

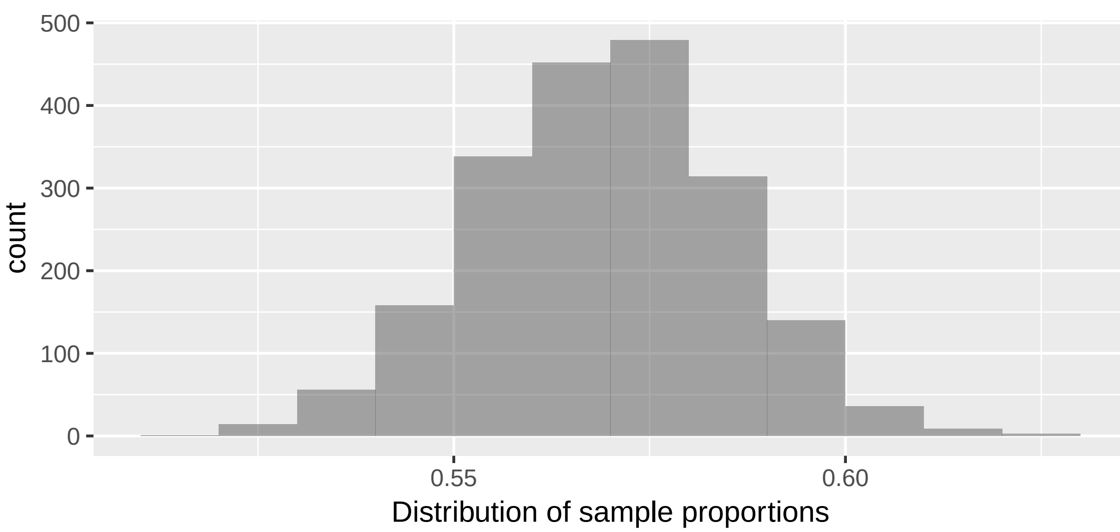
This work leverages initiatives undertaken by Project MOSAIC (http://www.mosaic-web.org), an NSF-funded effort to improve the teaching of statistics, calculus, science and computing in the undergraduate curriculum. In particular, we utilize the `mosaic` package, which was written to simplify the use of R for introductory statistics courses. A short summary of the R needed to teach introductory statistics can be found in the mosaic package vignettes (http://cran.r-project.org/web/packages/mosaic). A paper describing the mosaic approach was published in the *R Journal*: https://journal.r-project.org/archive/2017/RJ-2017-024.

## Chapter 17: Sampling Distribution Models

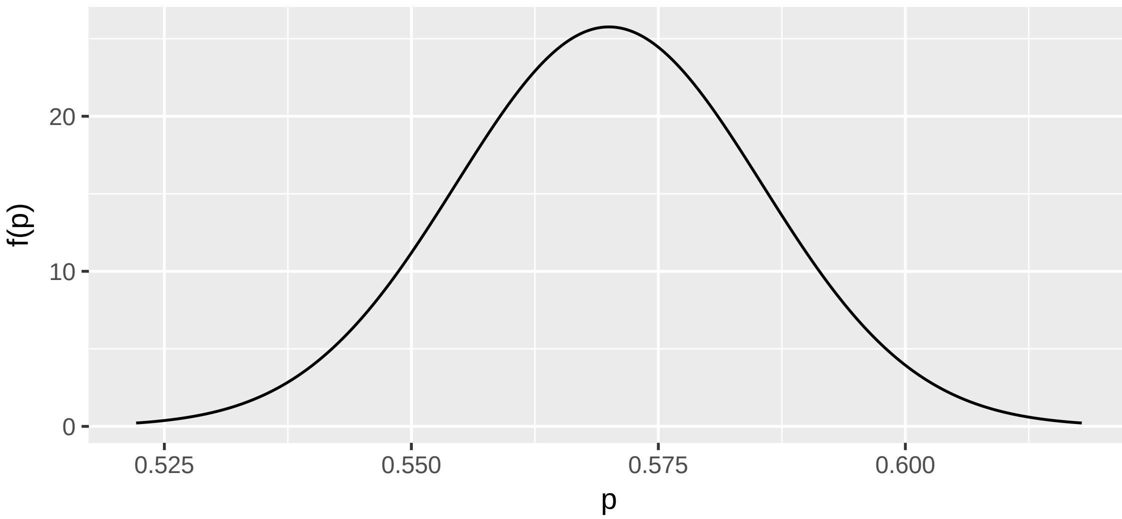
### Section 17.1: Sampling distribution of a proportion

Let's regenerate Figure 17.1 (page 444).

```
library(mosaic)
options(digits = 3)
numsim <- 2000
n <- 1022
p <- 0.57
samples <- rbinom(numsim, size = n, prob = p)/n
gf_histogram(~ samples, xlab = "Distribution of sample proportions",
          binwidth = 0.01, center = 0.01/2, type = "count")
```



```r
gf_dist("norm", params = list(p, sqrt(p*(1-p)/n)), xlab = "p", ylab = "f(p)")
```




**Section 17.2: When does the normal model work?**

We can replicate the example from page 449:

```r
p <- 0.22
n <- 200
pnorm(.155, mean = p, sd = sqrt(p*(1-p)/n))   # normal approximation
```

```
## [1] 0.0132
```

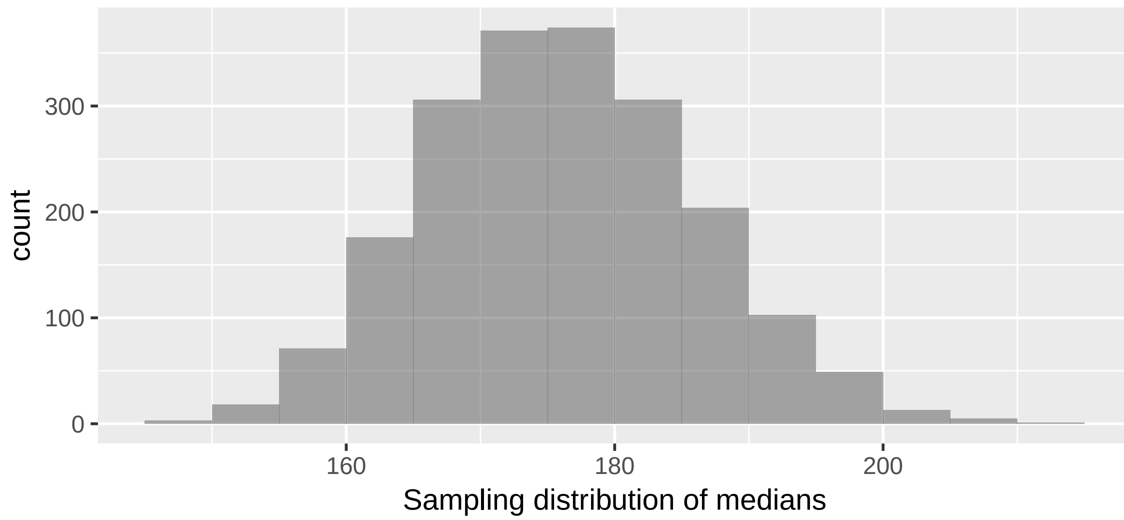
```r
pbinom(31, size = n, prob = p)    # exact value
```

```
## [1] 0.0139
```

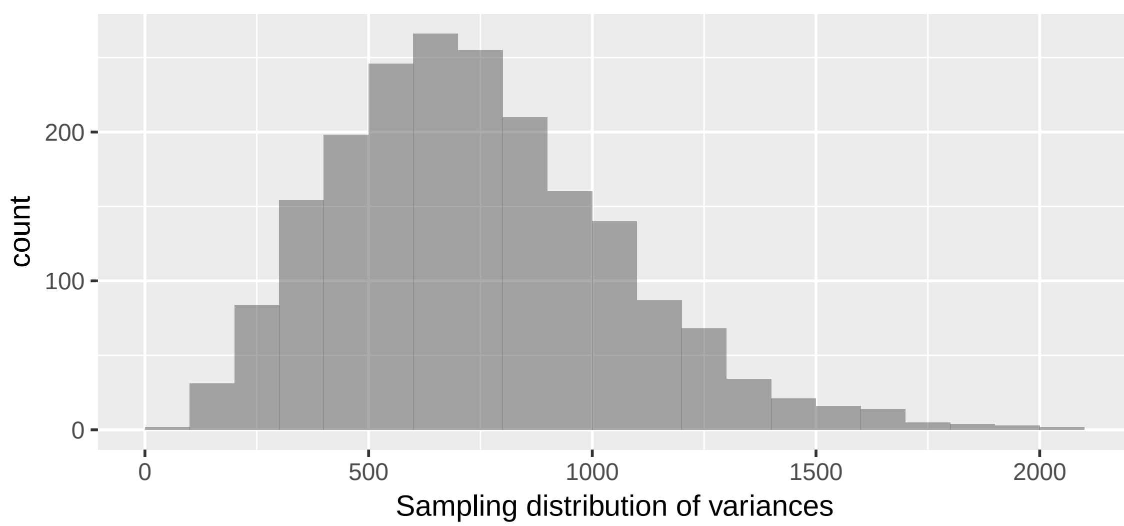
**Section 17.3: The sampling distribution of other statistics**

Let's replicate the display on page 451:

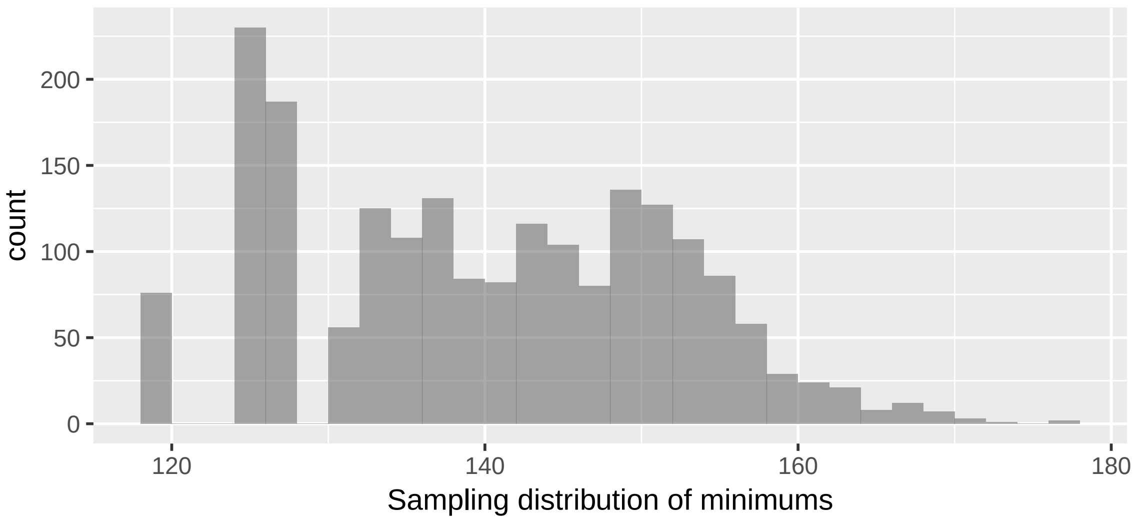
```r
BodyFat <- read.csv("http://nhorton.people.amherst.edu/sdm4/data/Body_fat_complete.csv")
medians <- do(2000)*median(~ Weight, data = sample(BodyFat, 10, replace = FALSE))
gf_histogram(~ median, xlab = "Sampling distribution of medians",
          binwidth = 5, center = 5/2, data = medians)
```




```
variances <- do(2000)*var(~ Weight, data = sample(BodyFat, 10, replace = FALSE))
gf_histogram(~ var, xlab = "Sampling distribution of variances",
          binwidth = 100, center = 100/2, data = variances)
```




```
minimums <- do(2000)*min(~ Weight, data = sample(BodyFat, 10, replace = FALSE))
gf_histogram(~ min, xlab = "Sampling distribution of minimums",
          binwidth = 2, center = 2/2, data = minimums)
```




Neither of the sampling distributions of the variance or the minimums are normally distributed.

**Section 17.4: Central Limit Theorem**

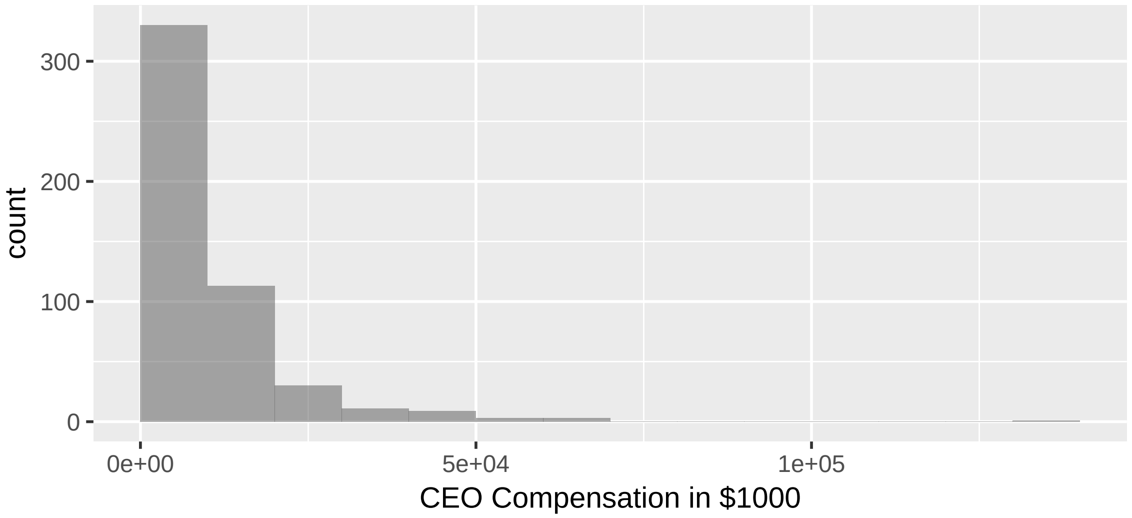
Let's replicate the displays on pages 453-454:

```
require(readr)
CEO <- read_delim("http://nhorton.people.amherst.edu/sdm4/data/CEO_Salary_2012.txt", delim = "\t")
CEO <- mutate(CEO, Pay = One_Year_Pay*1000)
favstats(~ Pay, data = CEO)
```

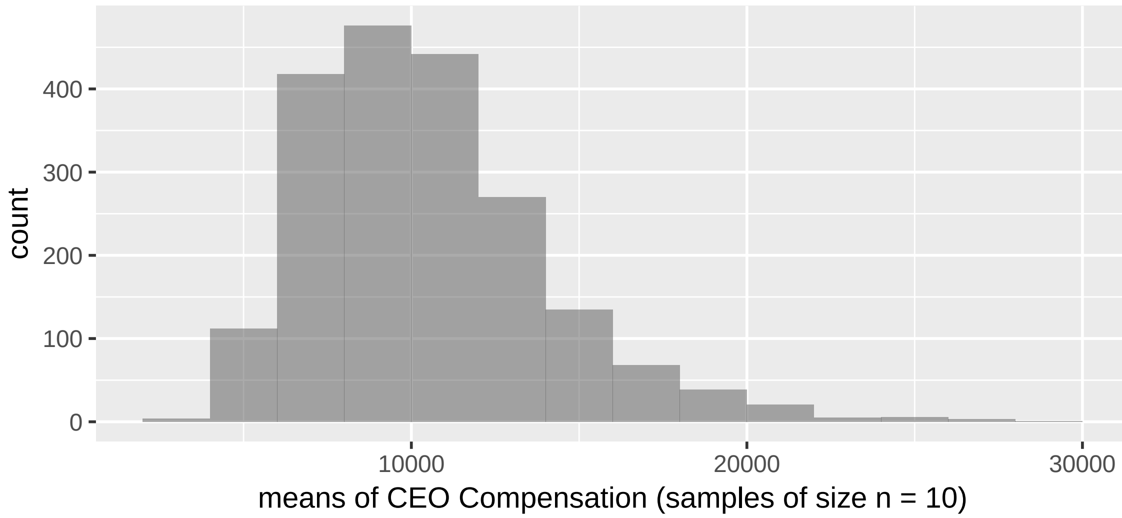
```
##  min   Q1 median    Q3    max  mean    sd   n missing
##    0 3885   6968 13361 131190 10476 11462 500       0
```

Note that Figure 17.11 seems to be off by a factor of 10!

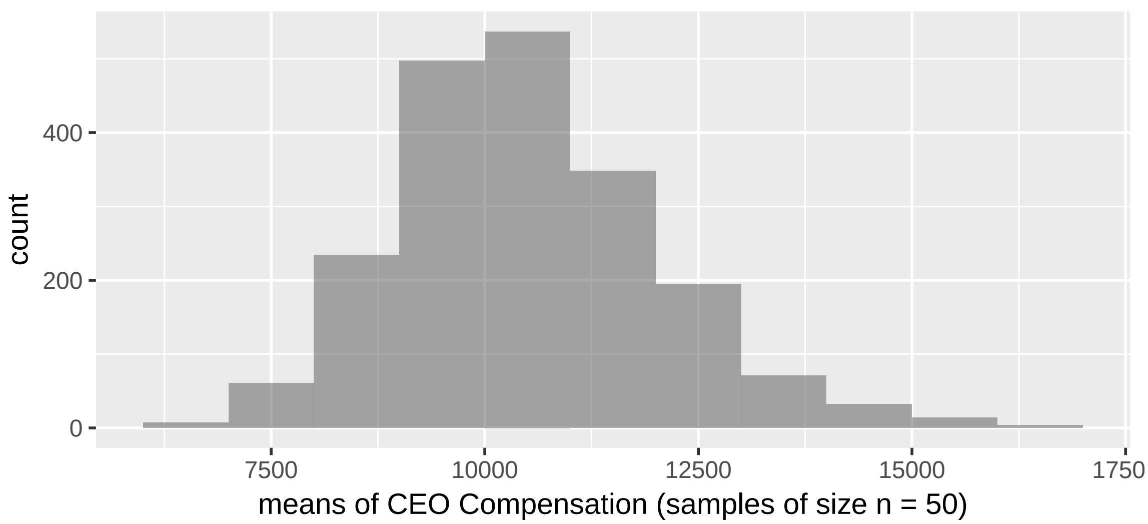
```
gf_histogram(~ Pay, xlab ="CEO Compensation in $1000", binwidth = 10000, center = 10000/2-.01, data = C
```



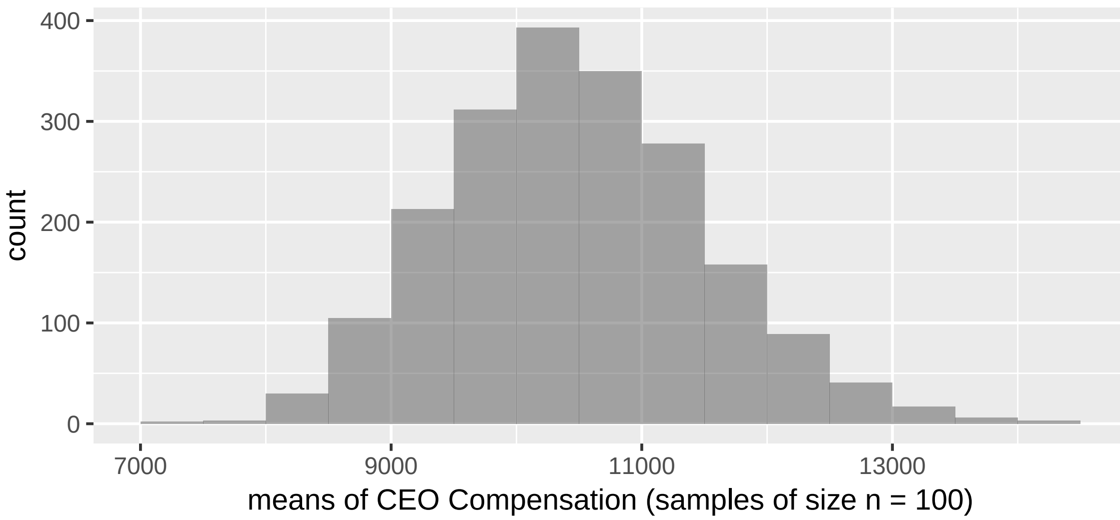
```
samp10 <- do(2000)*mean(~ Pay, data = sample(CEO, 10))
gf_histogram(~ mean, xlab = "means of CEO Compensation (samples of size n = 10)",
  binwidth = 2000, center = 2000/2, data = samp10)
```



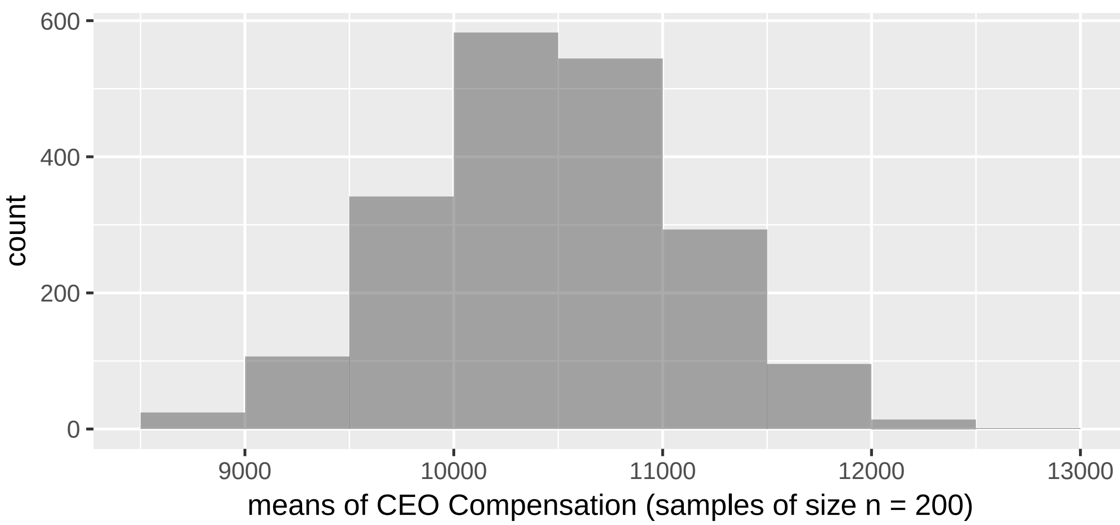

```
samp50 <- do(2000)*mean(~ Pay, data = sample(CEO, 50))
gf_histogram(~ mean, xlab = "means of CEO Compensation (samples of size n = 50)",
  binwidth = 1000, center = 1000/2, data = samp50)
```




```
samp100 <- do(2000)*mean(~ Pay, data = sample(CEO, 100))
gf_histogram(~ mean, xlab = "means of CEO Compensation (samples of size n = 100)",
  binwidth = 500, center = 500/2, data = samp100)
```




```
samp200<- do(2000)*mean(~ Pay, data = sample(CEO, 200))
gf_histogram(~ mean, xlab = "means of CEO Compensation (samples of size n = 200)",
  binwidth = 500, center = 500/2, data = samp200)
```




Note how the axis limits get narrower as the sample size increases (since the means are less variable when the sample size increases:

```
mysd <- sd(~ Pay, data = CEO)
mysd
```

```
## [1] 11462
```

```
sd(~ mean, data = samp10)  # what we observed
```

```
## [1] 3529
```

```r
mysd/sqrt(10)    # what we would expect
```

```
## [1] 3625
```

We can repeat this comparison for each of the sets of samples.

```r
sd(~ mean, data = samp50)
```

```
## [1] 1529
```

```r
sd(~ mean, data = samp100)
```

```
## [1] 1040
```

```r
sd(~ mean, data = samp200)
```

```
## [1] 641
```