SDM4 in R: Probability Models (Chapter 16)

Nicholas Horton (nhorton@amherst.edu)

June 6, 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 16: Probability Models

Section 16.1: Bernoulli models

We can replicate the calculation on page 413:

```
library(mosaic)
options(digits = 3)
dbinom(0, size = 1, prob = 0.2)  # Probability that you don't get Hope's picture
```

```
## [1] 0.8
```

```
dbinom(1, size = 1, prob = 0.2)  # Probability that you do
```

```
## [1] 0.2
```

```
dgeom(1, prob = 0.2)  # Probability that you get it on the second try P(X=1)
```

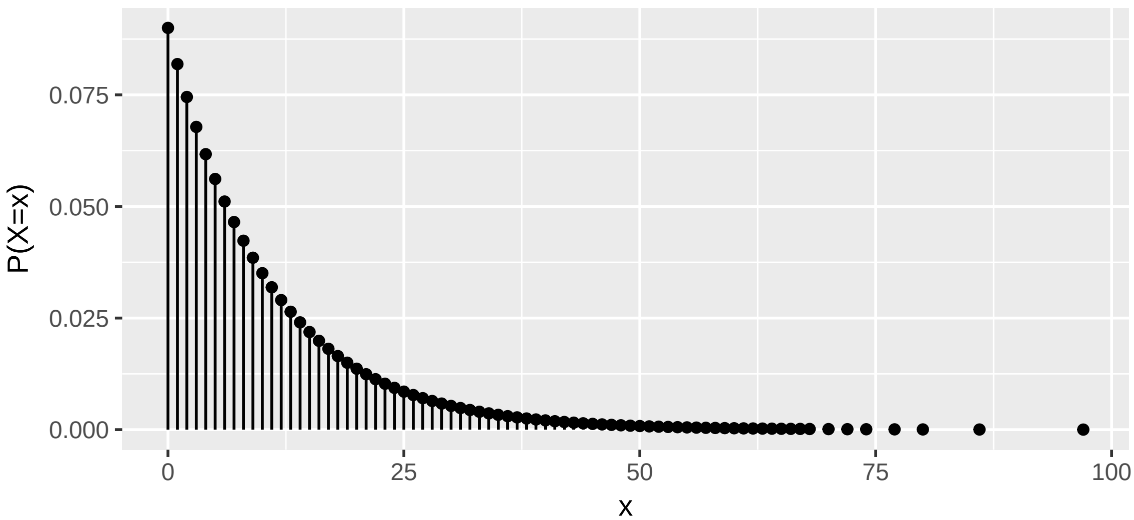
```
## [1] 0.16
```

```
# Note that the geometric in R doesn't include the final success!
```

Section 16.2: The Geometric model

We can replicate the example from page 414:

```
p <- 0.09
gf_dist("geom", prob = p,
  main = "Geometric distribution", ylab = "P(X=x)")
```




```
1/p   # true expected value
```

```
## [1] 11.1
```

```
mean(~ rgeom(20000, prob = p)) #  simulation to check
```

```
## [1] 10.1
```

along with the Think/Show/Tell on pages 415-416:

```
p <- 0.06
dgeom(0:3, prob = p)
```

```
## [1] 0.0600 0.0564 0.0530 0.0498
```

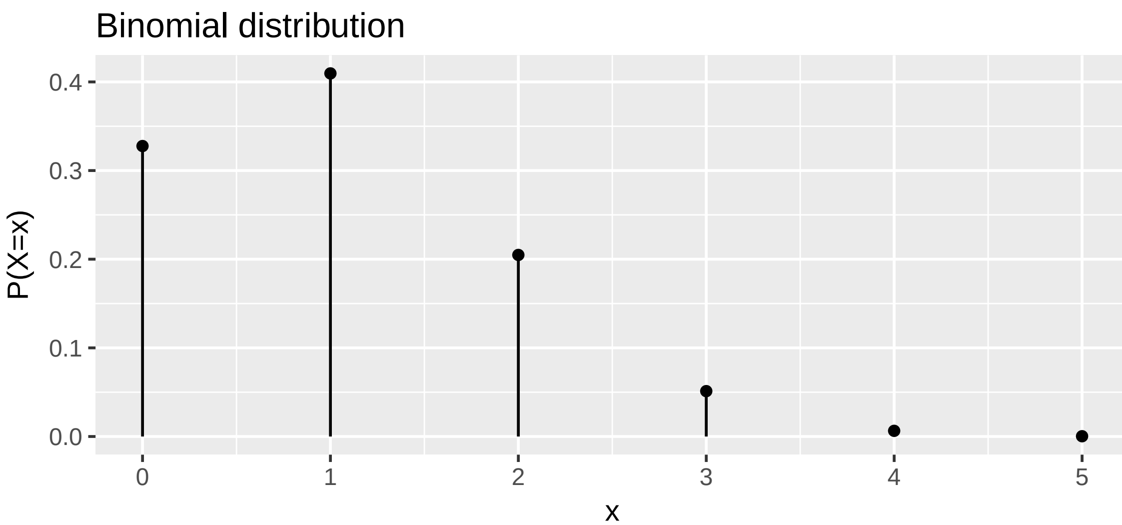
```
pgeom(3, prob = p)
```

```
## [1] 0.219
```

Section 16.3: The Binomial model

Let's replicate the example on page 417:

```
p <- 0.2    # back to Hope Solo example
n <- 5
gf_dist("binom", size = n, prob = p, xlab = "x", ylab = "P(X=x)",
  title = "Binomial distribution")
```




```
n*p   # true expected value
```

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

```
n*p*(1-p)   # true variance
```

```
## [1] 0.8
```

```
mysamp <- rbinom(20000, size = n, prob = p) #  simulation to check
mean(~ mysamp)
```

```
## [1] 0.991
```

```
var(~ mysamp)
```

```
## [1] 0.812
```

```
dbinom(2, size = 5, prob = p)
```

```
## [1] 0.205
```

or the SPAM example on page 418:

```
dbinom(1, size = 25, prob=0.09)
```

```
## [1] 0.234
```

```
dbinom(2, size = 25, prob = 0.09)
```

```
## [1] 0.278
```

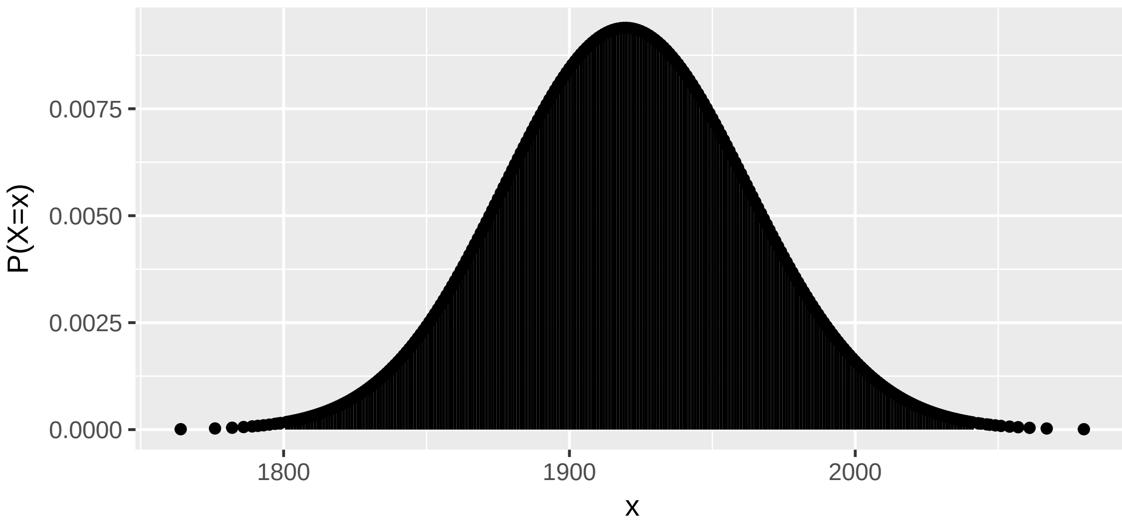
```r
sum(dbinom(1:2, size = 25, prob = 0.09))
```

```
## [1] 0.512
```

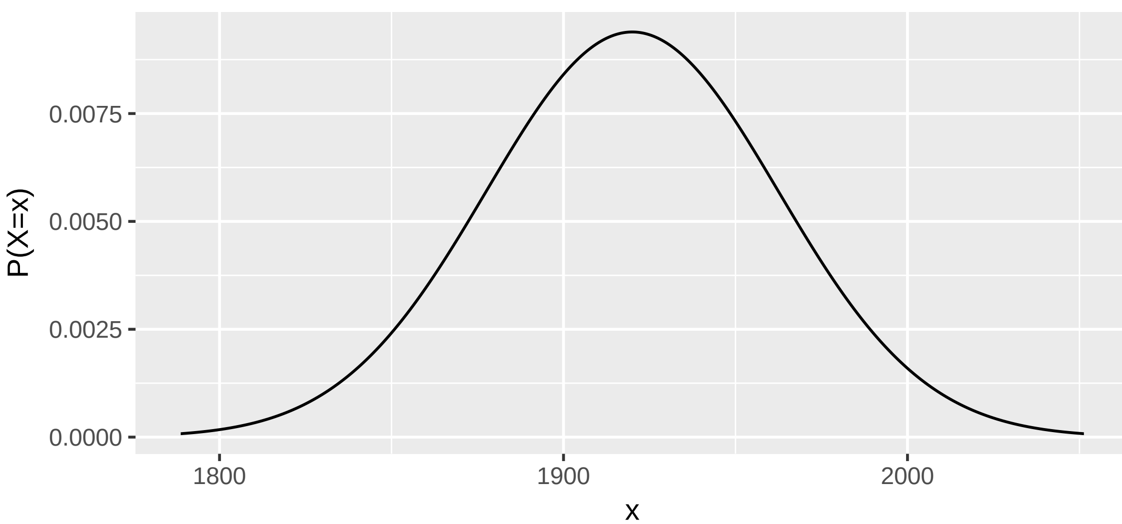
Section 16.4: Normal approximation to the binomial

Let's replicate the example on page 419:

```r
p <- 0.06; n <- 32000
gf_dist("binom", size = n, prob = p, xlab = "x", ylab = "P(X=x)")
```




```r
ex <- n*p
varx <- n*p*(1-p)
gf_dist("norm", mean = ex, sd = sqrt(varx), xlab = "x", ylab = "P(X=x)")
```




```r
pbinom(1849, n, p)
```

```
## [1] 0.0479
```

```r
pnorm(1849, ex, sqrt(varx))
```

```
## [1] 0.0473
```

Your parents had to use the normal approximation. :-) The good news is that you don't have to if you use R (or other modern software).

Section 16.5: The continuity correction

Let's repeat the example on page 421 to demonstrate that we don't need the continuity correction if we calculate the exact distribution of the binomial.

```r
p <- 0.2
n <- 50
dbinom(10, n, p)
```

```
## [1] 0.14
```

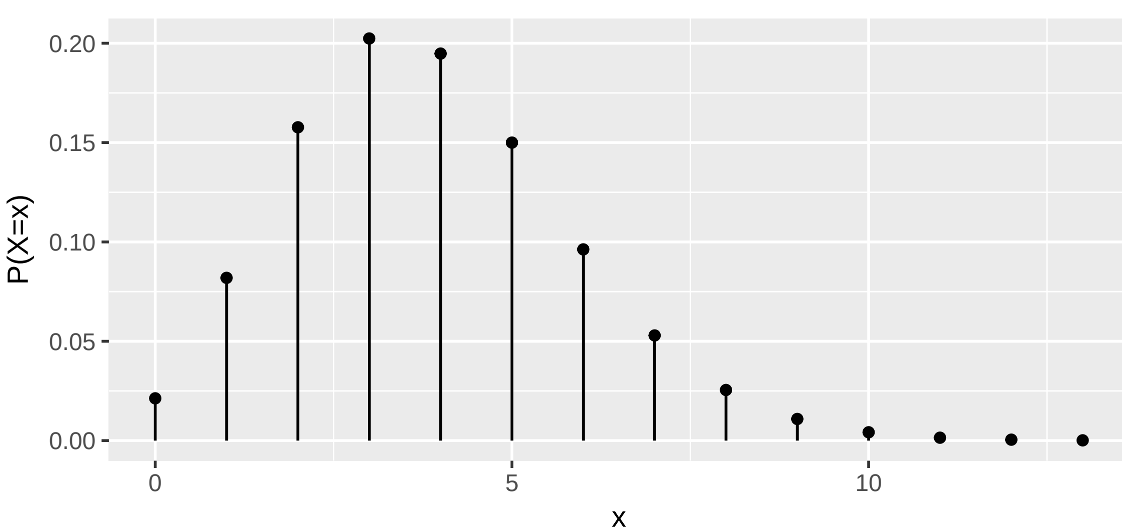
```r
pnorm(10.5, n * p, sqrt(n * p * (1 - p))) -
  pnorm(9.5, n * p, sqrt(n * p *(1 - p)))  # continuity correction
```

```
## [1] 0.14
```

Section 16.6: The Poisson model

Let's replicate the example on page 424:

```r
lambda <- 35000 * 0.00011
gf_dist("pois", lambda = lambda, xlab = "x", ylab = "P(X=x)")
```



```r
1 - ppois(7, lambda)
```

```
## [1] 0.0427
```

Section 16.7: Uniform and exponential

Let's replicate the example on page 426 for the uniform distribution:

```r
x <- runif(20000, min = 0, max = 20)
mean(~ x)
```

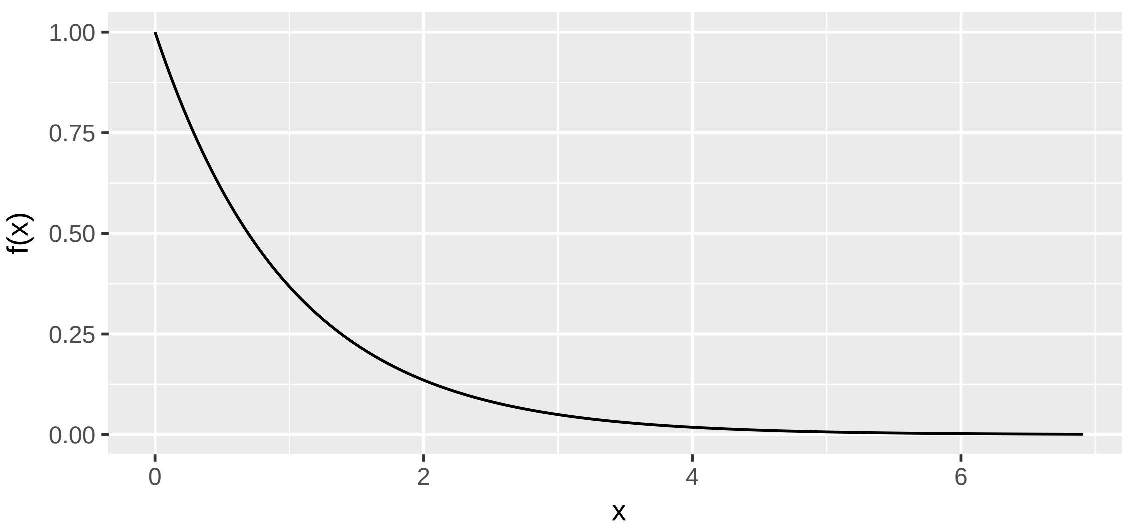
```
## [1] 9.99
```

```r
sd(~ x)
```

```
## [1] 5.8
```

and on page 427 for the exponential distribution:

```r
gf_dist("exp", rate = 1, xlab = "x", ylab = "f(x)")
```




```r
pexp(1/3, rate = 4)    # rate is 1/E[X]
```

```
## [1] 0.736
```