IS4 in R: The Standard Deviation as a Ruler and the Normal Model (Chapter 5)

Patrick Frenett, Vickie Ip, and Nicholas Horton (nhorton@amherst.edu)

June 19, 2018

Introduction and background

This document is intended to help describe how to undertake analyses introduced as examples in the Fourth Edition of *Intro Stats* (2013) 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 https://nhorton.people.amherst.edu/is4.

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 5: The Standard Deviation as a Ruler and the Normal Model

Section 5.1: Standardizing with z-scores

From page 111

```
library(mosaic)
library(readr)
library(ggformula)
options(na.rm = TRUE)
options(digits = 3)
(6.54 - 5.91)/0.56  # Dobrynska's jump was 2.18 SD's greater than the mean
```

```
## [1] 1.12
```

```
twohund <- as.vector(c(23.2, 23.3, 23.3, 23.6, 23.9, 23.9, 24.2, 24.2, 24.3,
                       24.3, 24.3, 24.3, 24.3, 24.4, 24.5, 24.5, 24.6, 24.6,
                       24.6, 24.7, 24.7, 24.9, 24.9, 24.9, 25.0, 25.0, 25.0,
                       25.2, 25.3, 25.4, 25.4, 25.4, 25.4, 25.5, 25.9, 25.9, 26.1))
twohund <- data.frame(twohund)
df_stats(~ ., data = twohund)
```

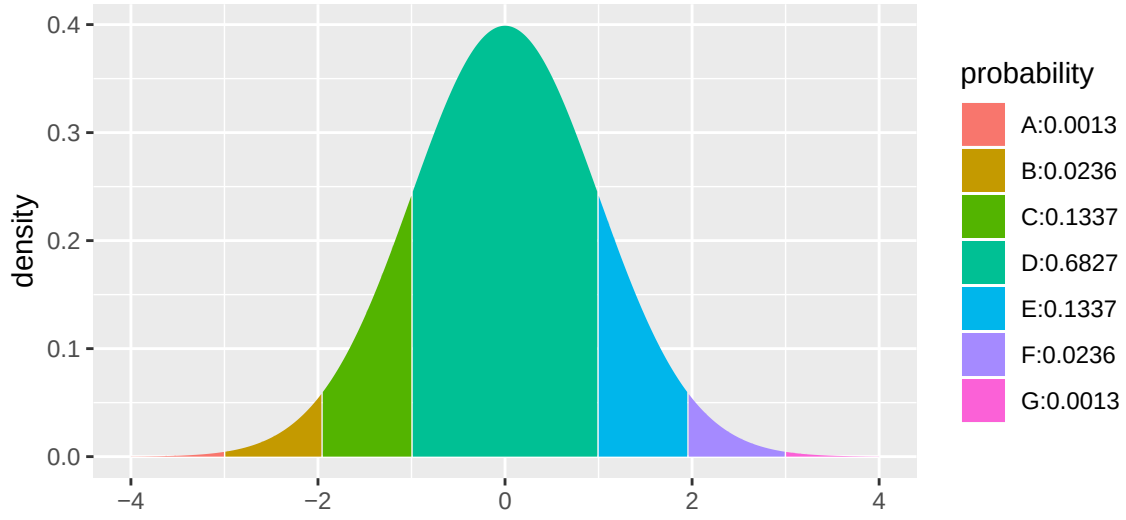
```
##    min   Q1 median   Q3  max mean    sd  n missing
## 1 23.2 24.3   24.6 25.2 26.1 24.7 0.718 37       0
```

Section 5.2: Shifting and Scaling

Section 5.3: Normal Models

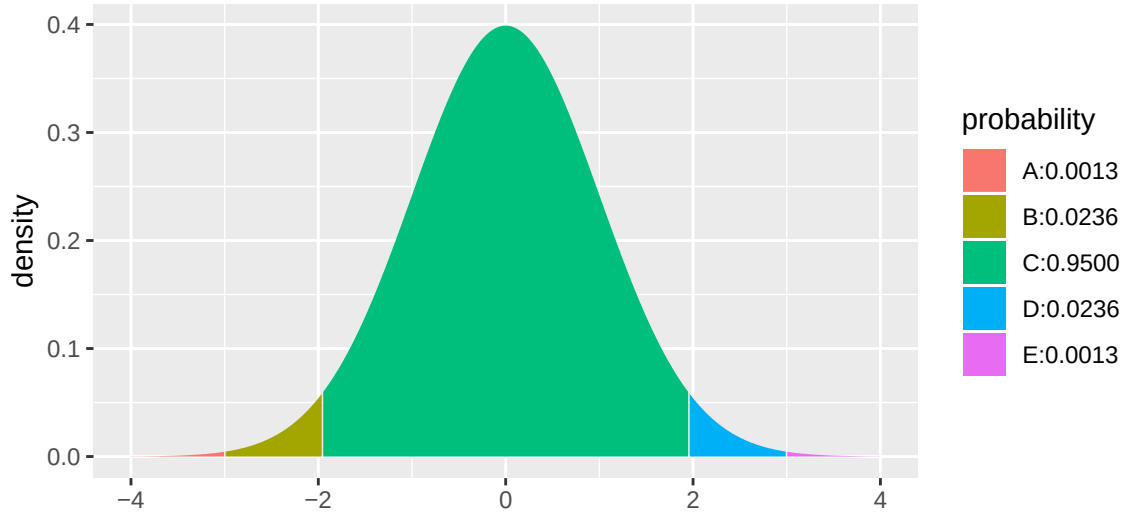
The 68-95-99.7 rule

```r
xpnorm(c(-3, -1.96, -1, 1, 1.96, 3), mean = 0, sd = 1, verbose = FALSE)
```



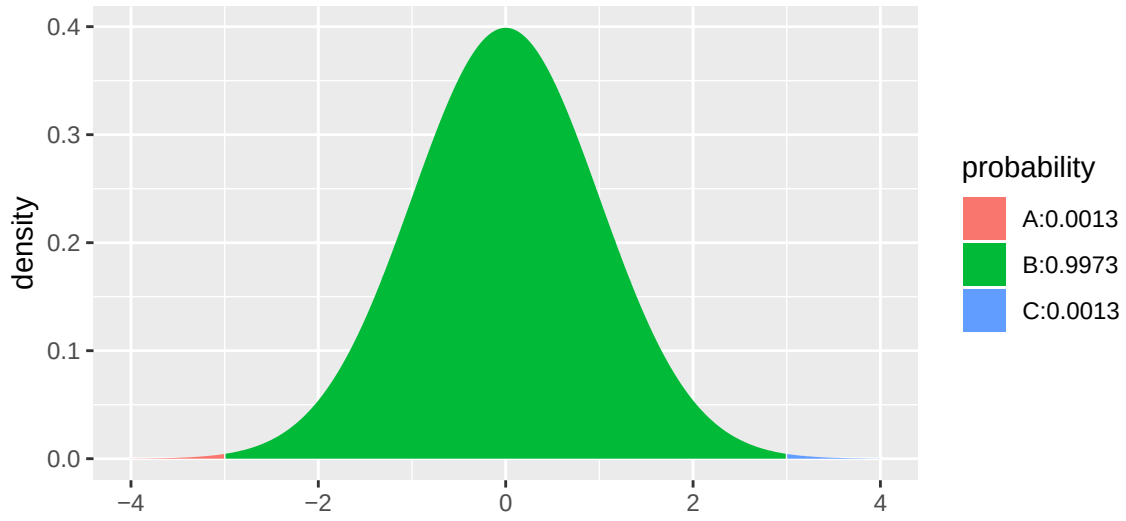

```
## [1] 0.00135 0.02500 0.15866 0.84134 0.97500 0.99865
```

```r
xpnorm(c(-3, -1.96, 1.96, 3), mean = 0, sd = 1, verbose = FALSE)
```




```
## [1] 0.00135 0.02500 0.97500 0.99865
```

```r
xpnorm(c(-3, 3), mean = 0, sd = 1, verbose = FALSE)
```




```
## [1] 0.00135 0.99865
```

Step-by-step (page 120)

```
xpnorm(600, mean = 500, sd = 100)
```

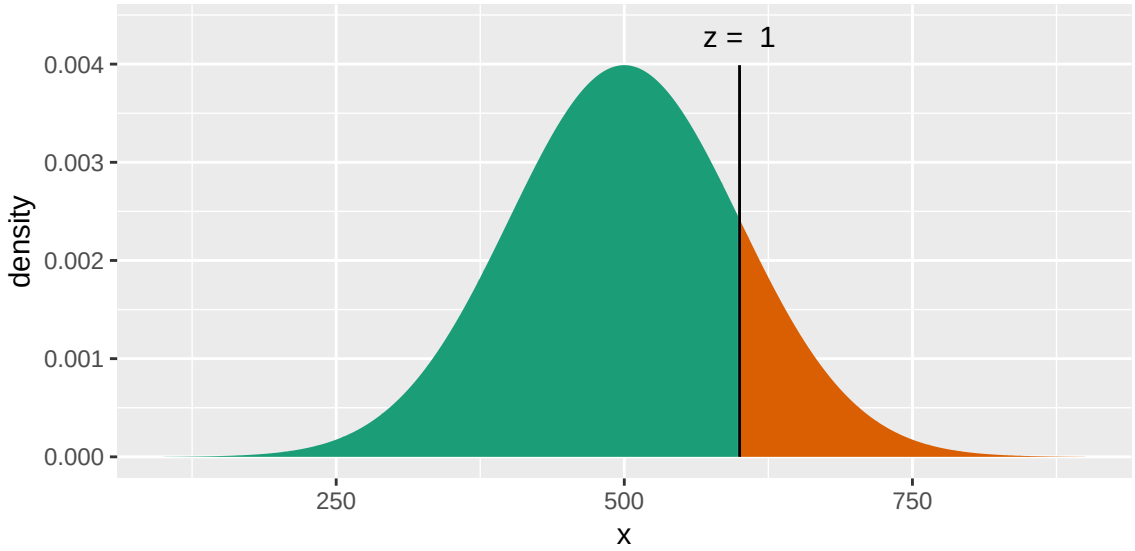
```
##
```

```
## If X ~ N(500, 100), then
```

```
##  P(X <= 600) = P(Z <= 1) = 0.8413
```

```
##  P(X >  600) = P(Z >  1) = 0.1587
```

```
##
```



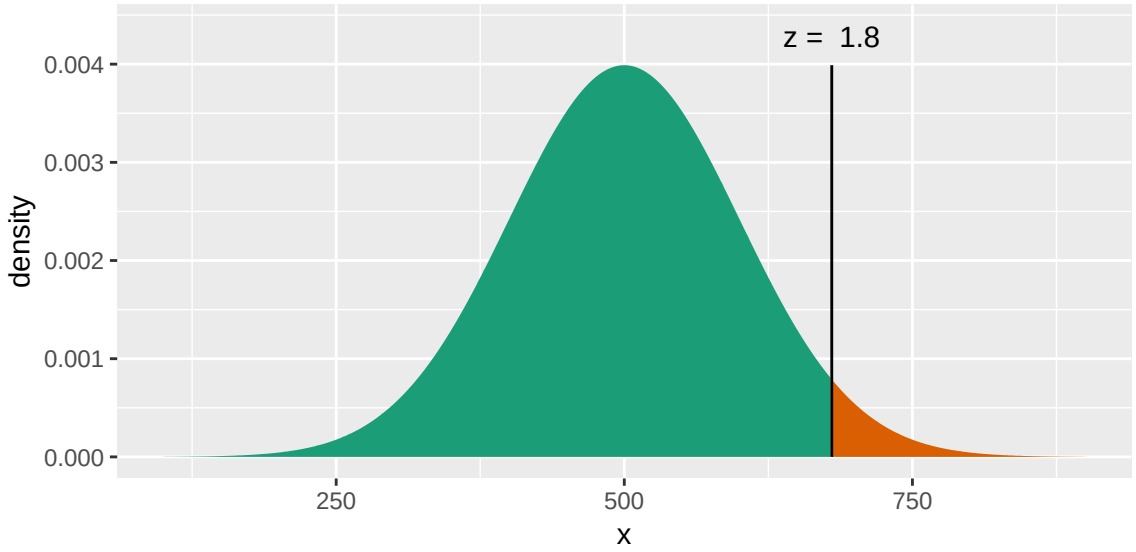

```
## [1] 0.841
```

Section 5.4: Finding normal percentiles

as on page 121

```
xpnorm(680, mean = 500, sd = 100)
```

```
##
## If X ~ N(500, 100), then
##  P(X <= 680) = P(Z <= 1.8) = 0.9641
##  P(X >  680) = P(Z >  1.8) = 0.03593
##
```




```
## [1] 0.964
```

```
qnorm(0.964, mean = 500, sd = 100)   # inverse of pnorm()
```

```
## [1] 680
```

```
qnorm(0.964, mean = 0, sd = 1)   # what is the z-score?
```

```
## [1] 1.8
```

or on page 122

```
xpnorm(450, mean = 500, sd = 100)
```

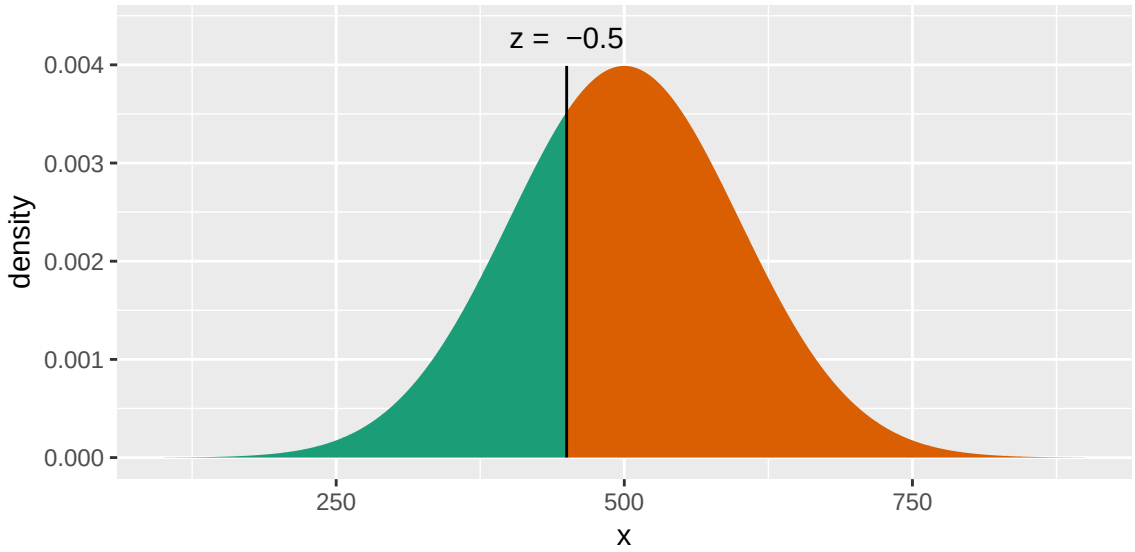
```
##
```

```
## If X ~ N(500, 100), then

##  P(X <= 450) = P(Z <= -0.5) = 0.3085

##  P(X >  450) = P(Z >  -0.5) = 0.6915

##
```




```
## [1] 0.309
```

and page 123

```
qnorm(.9, mean = 500, sd = 100)
```

```
## [1] 628
```

```
qnorm(.9, mean = 0, sd = 1)    # or as a Z-score
```

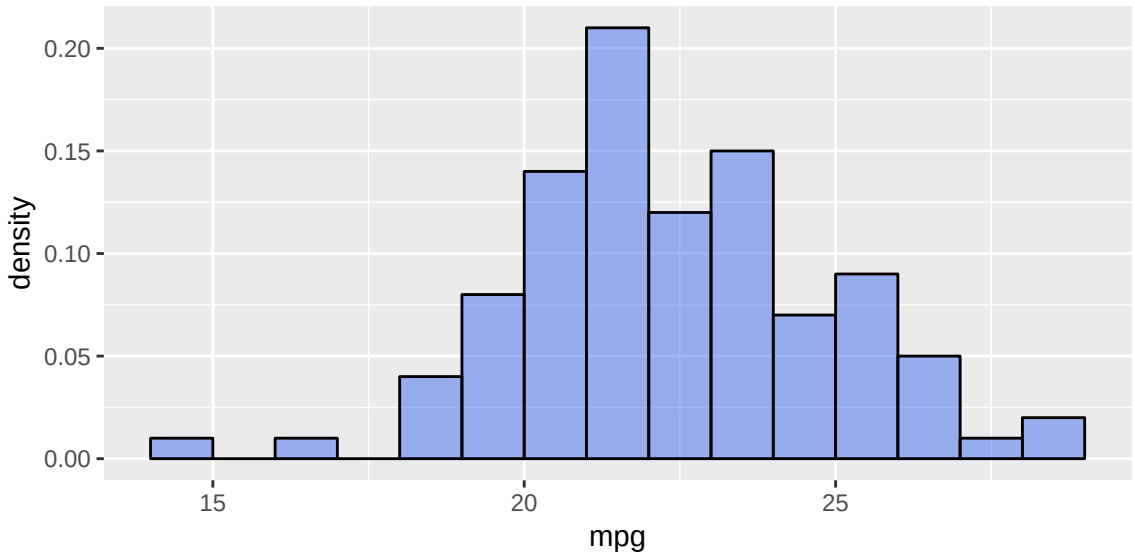
```
## [1] 1.28
```

Section 5.5: Normal Probability Plots

See Figure 5.8 on page 127

```
Nissan <-
read_delim("http://nhorton.people.amherst.edu/sdm4/data/Nissan.txt",
  delim = "\t")
```

```
gf_histogram(..density.. ~ mpg, binwidth = 1, center = 0.5,
             data = Nissan, fill = "royalblue2", col = TRUE)
```




```
gf_qq(~ mpg, data = Nissan) %>%
  gf_labs(x = "qnorm", y = "mpg")
```