Continuous Variables (Chapter 3)

Prof. Joyce Robbins

Continuous Variables

We’re looking for features such as:

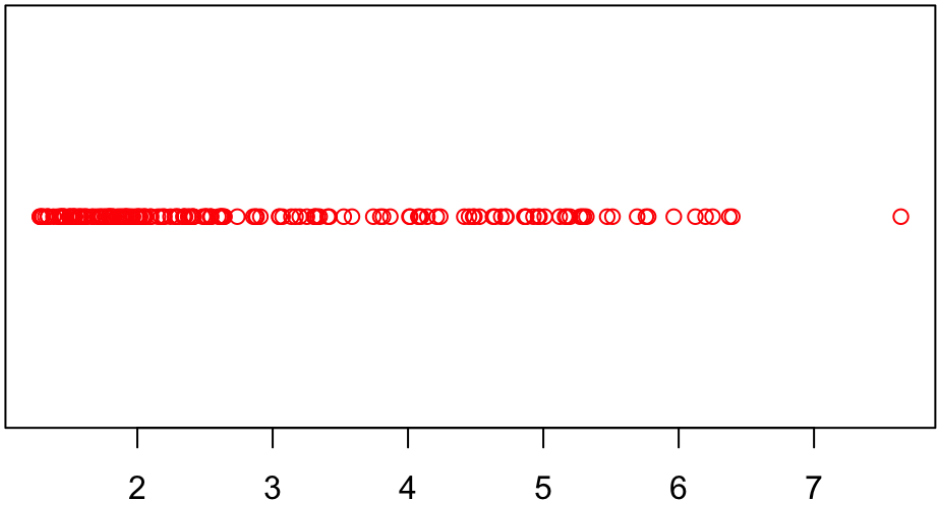
- Asymmetry
- Outliers
- Multimodality
- Gaps
- Heaping
- Rounding
- Impossibilities / Errors

Basic Options

- Stripcharts / rug plot
- Stem and leaf plot
- Dotplots
- Histogram / density curve
- Boxplot

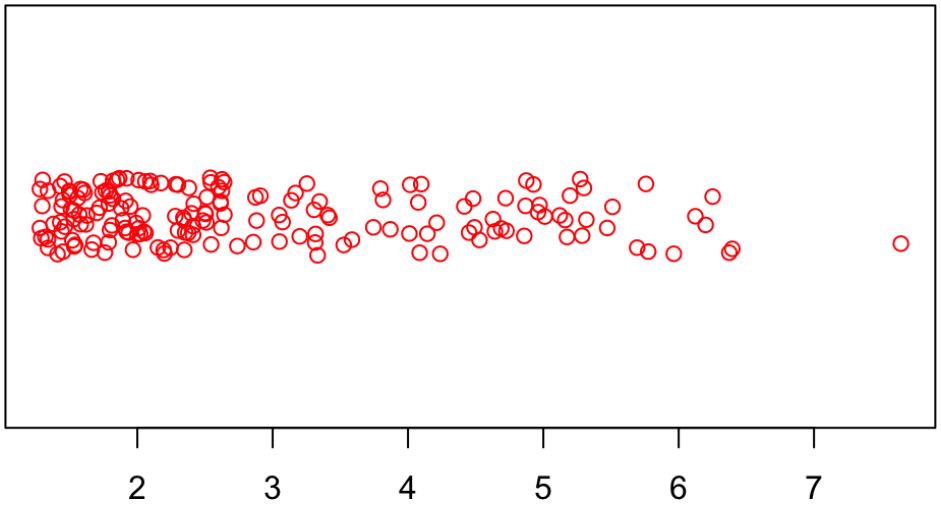
Strip charts

```r
par(las = 1) # for all chunks since global.par set to TRUE above
world <- read.csv("countries2012.csv")
stripchart(world$TFR, col = "red", pch = 21)
```



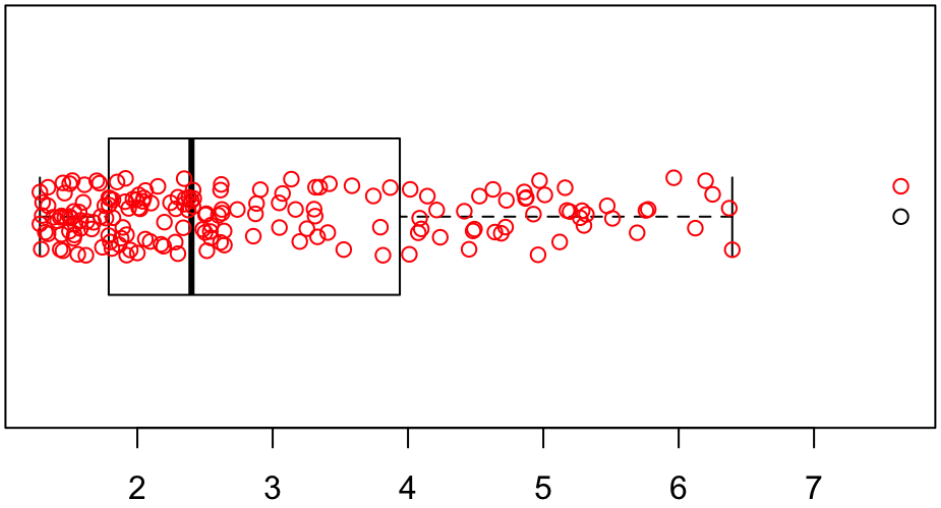
Strip charts

```
stripchart(world$TFR, col = "red", pch = 21,
           method = "jitter")
```



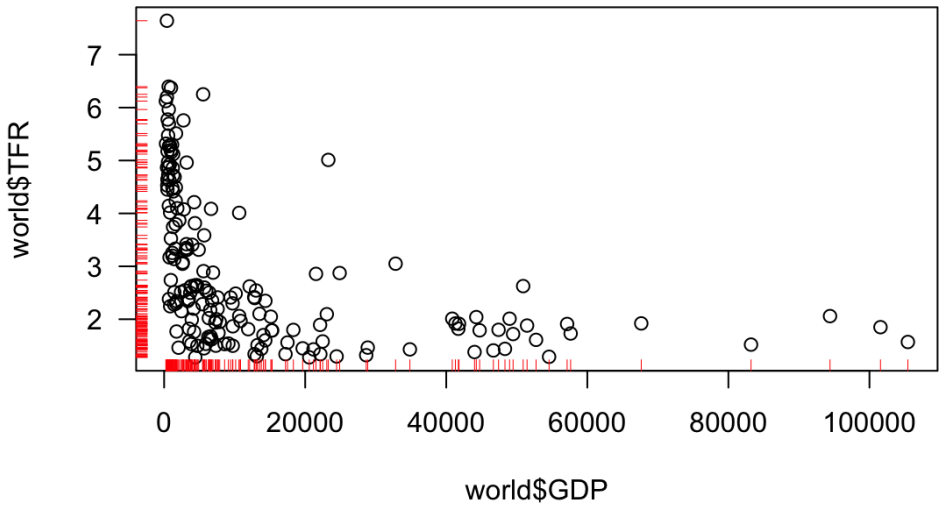
Strip charts w/ boxplot

```
boxplot(world$TFR, horizontal = TRUE)
stripchart(world$TFR, col = "red", pch = 21, add = TRUE, method = "jitter")
```



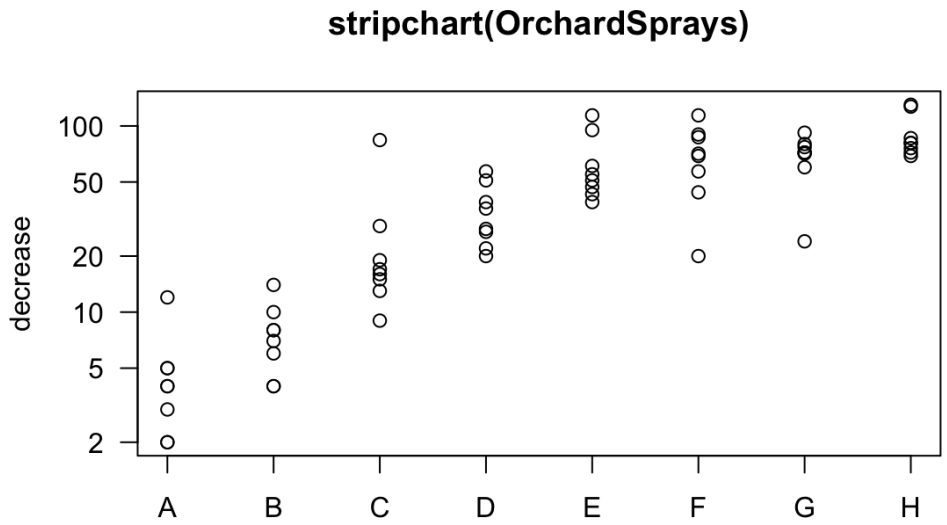
Rug plot

```
plot(world$GDP, world$TFR)
rug(world$GDP, col = "red")
rug(world$TFR, col = "red", side = 2)
```



Strip charts

```
stripchart(decrease ~ treatment,
    main = "stripchart(OrchardSprays)",
    vertical = TRUE, log = "y",
    data = OrchardSprays, pch = 21)
```



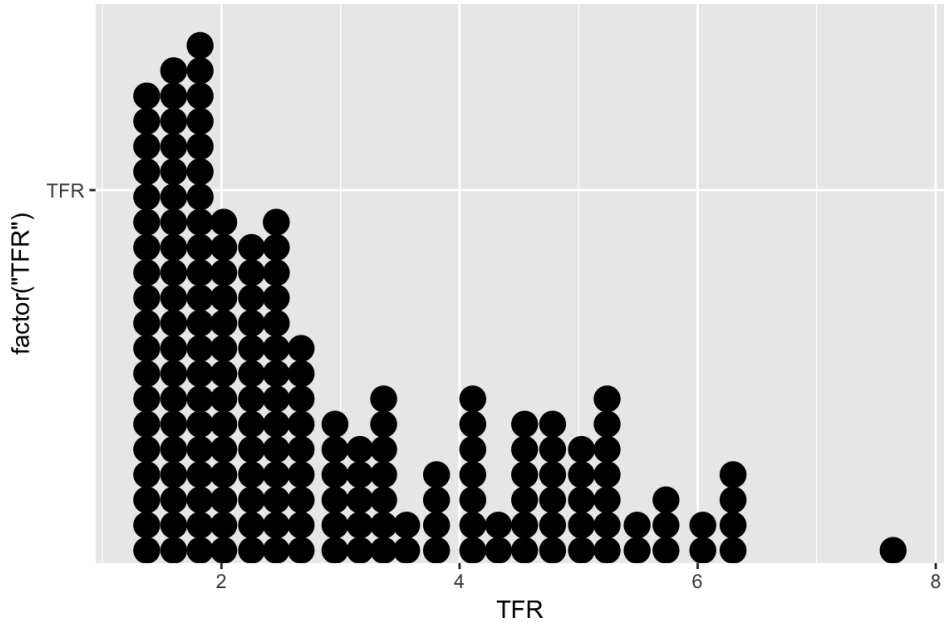
Stem and leaf plot

```r
prices <- c(379, 425, 450, 450, 499, 529, 535, 535, 545, 599, 665, 675, 699, 699, 725, 725, 745, 799)
stem(prices)
```

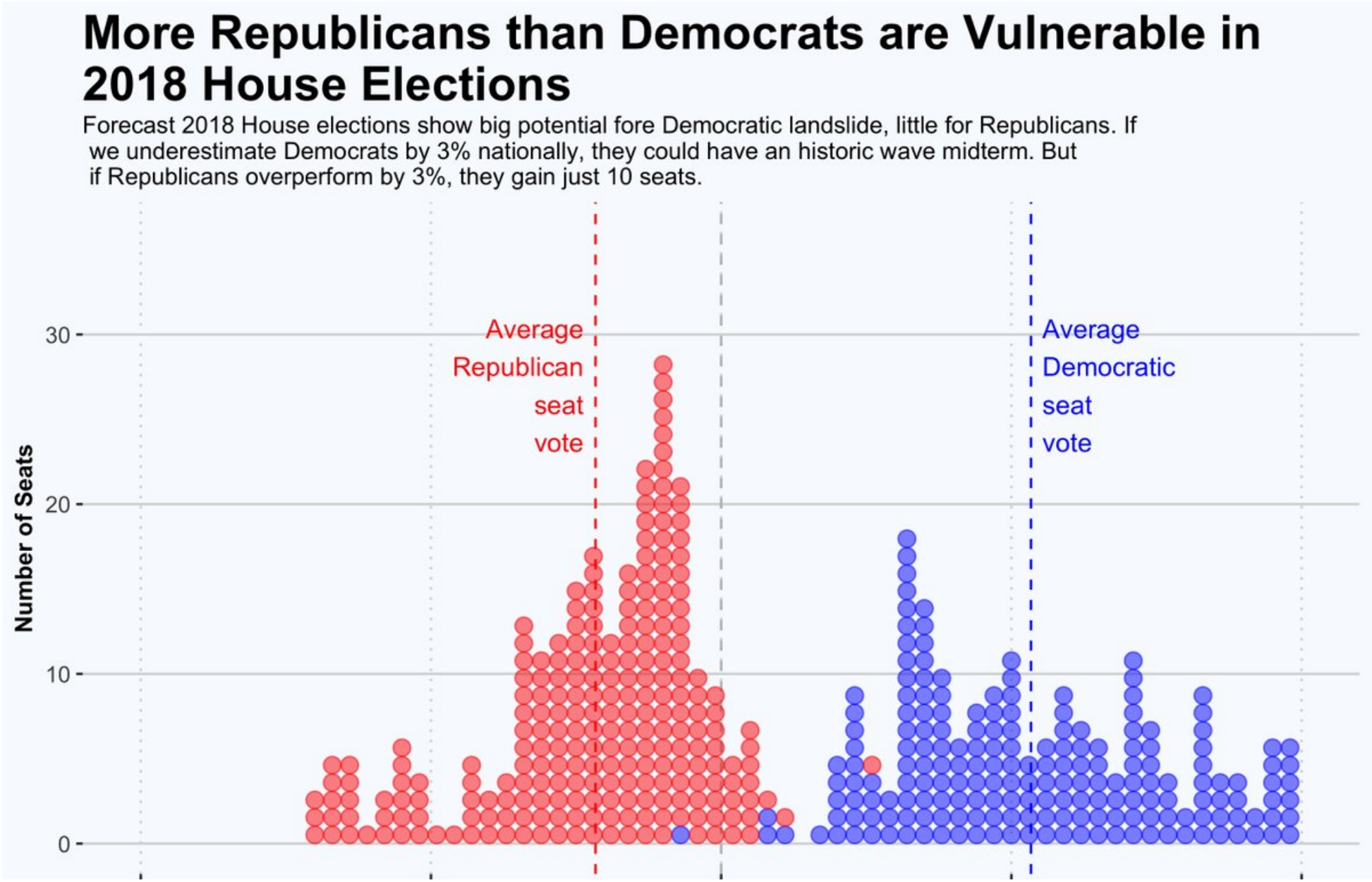
```
##
##   The decimal point is 2 digit(s) to the right of the |
##
##   3 | 8
##   4 | 355
##   5 | 03445
##   6 | 078
##   7 | 00335
##   8 | 0
```

Dot plot

```
library(ggplot2)
ggplot(world, aes(TFR, y = factor("TFR"))) +
    geom_dotplot()
```



2018 Congressional Race




0
25
50
75
100
Forecast 2018 Democrat Vote Share (%)
*Forecast comes from thecrosstab.com/2018-midterms-forecast/
@GElliottMorris | TheCrosstab.com | George Elliott Morris

Histograms

- primary tool for continuous data
- boundary issues
- count / relative frequency / density histograms
- unequal binwidth histograms
- importance of binwidth
- using `ggvis` to interactively adjust binwidths

How are histograms created?

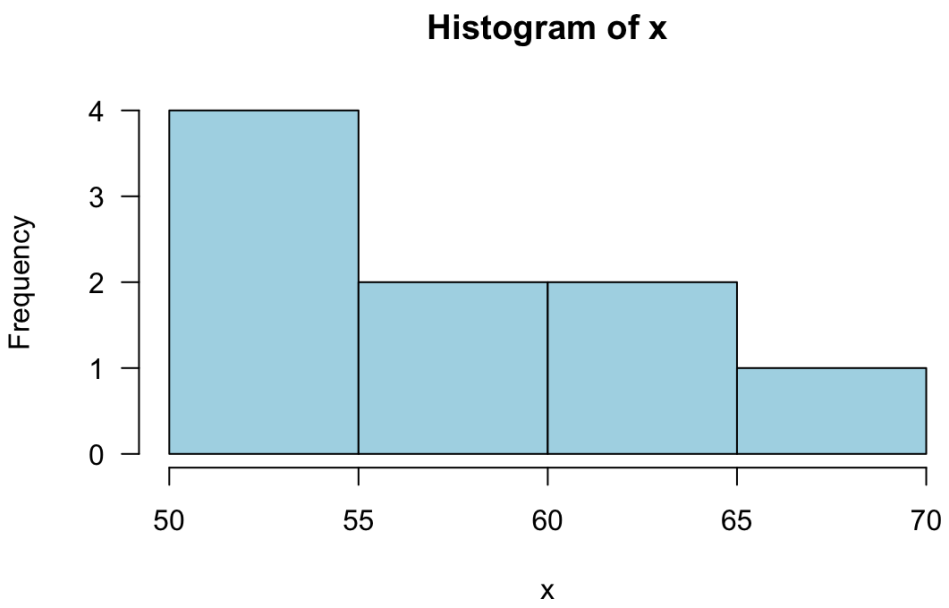
Draw a histogram on paper of the following data.

(use binwidth = 5)

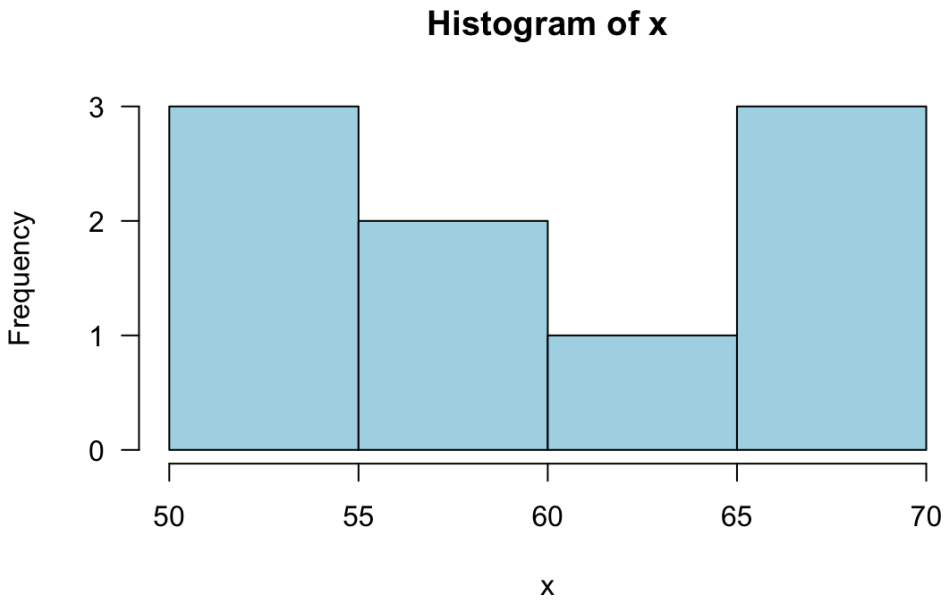
50, 51, 53, 55, 56, 60, 65, 65, 68

How are histograms created?

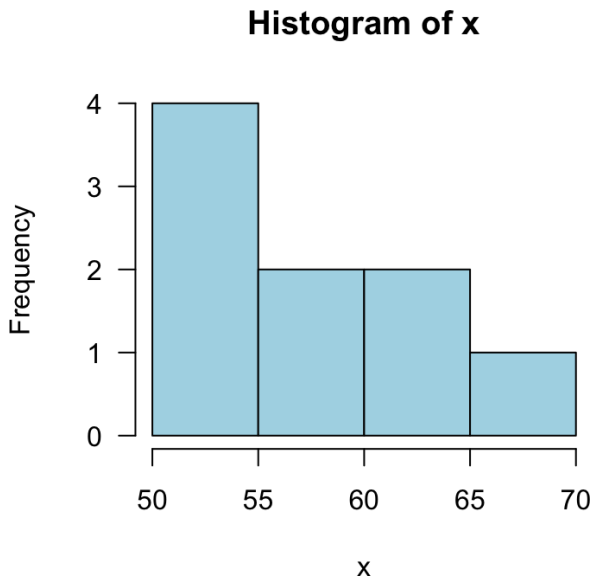
```r
par(las = 1)  # opts_knit$set(global.par = TRUE) above
x <- c(50, 51, 53, 55, 56, 60, 65, 65, 68)
hist(x, col = "lightblue")
```

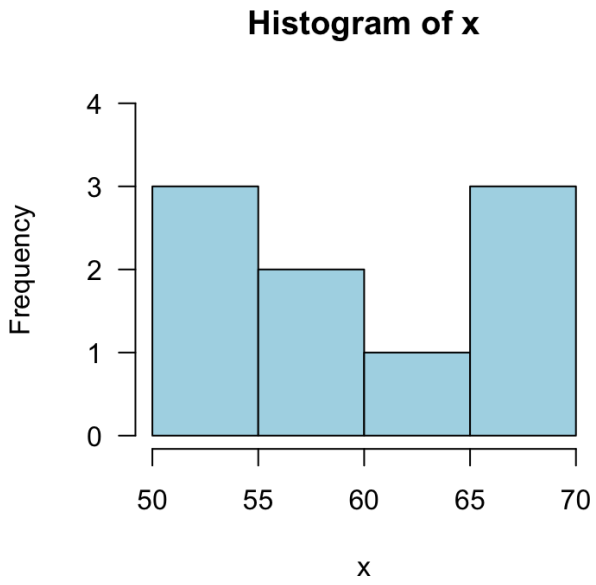


How are histograms created?



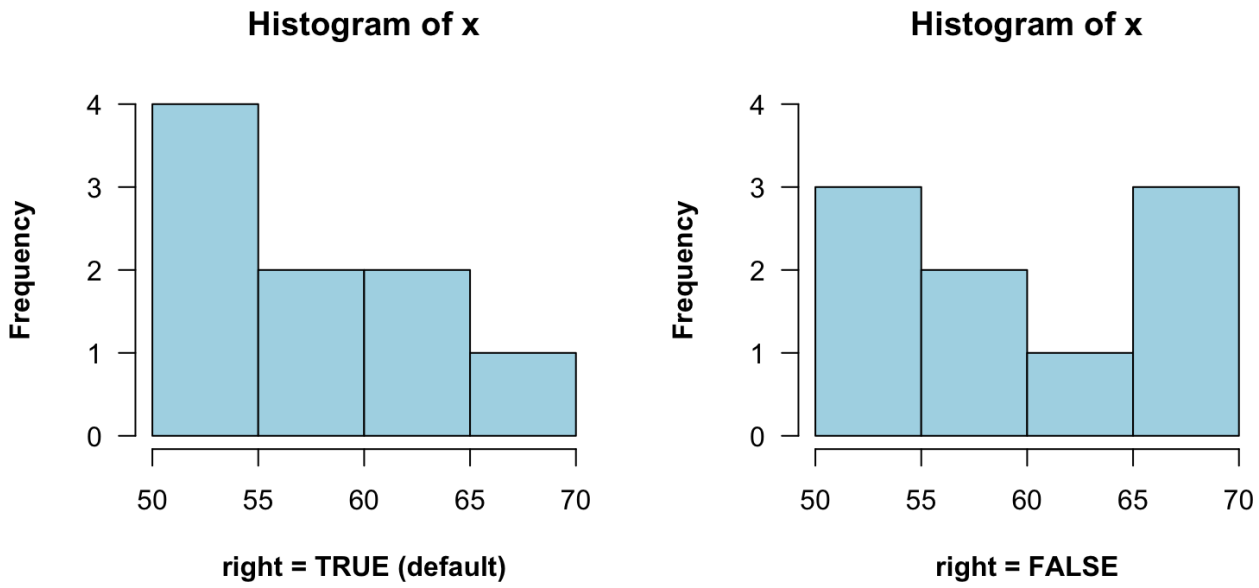
What is causing the difference?





Bin boundaries

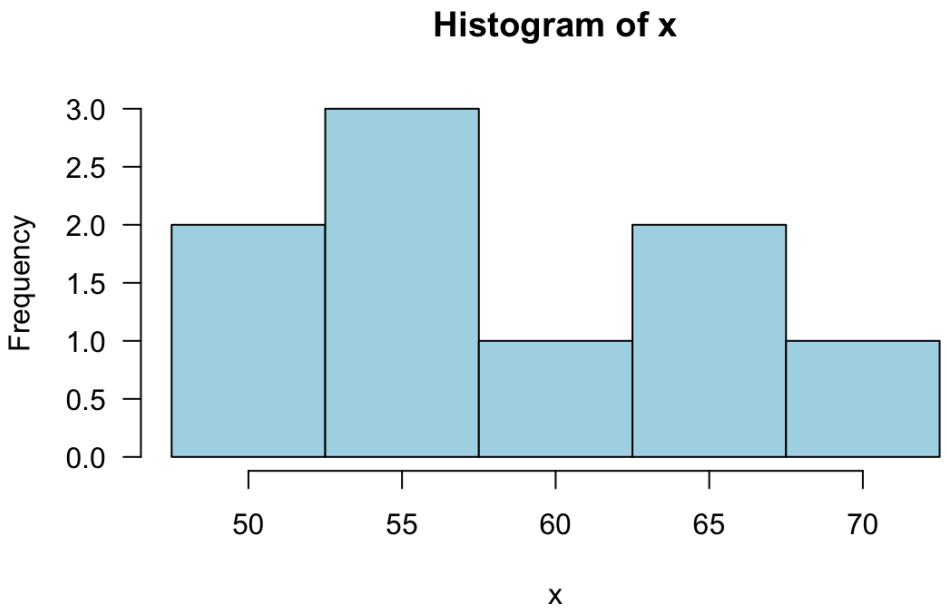
```
op <- par(mfrow = c(1, 2), las = 1)
hist(x, col = "lightblue", ylim = c(0, 4),
     xlab = "right = TRUE (default)", font.lab = 2)
hist(x, col = "lightblue", right = FALSE, ylim = c(0, 4),
     xlab = "right = FALSE", font.lab = 2)
```




```
par(op)
```

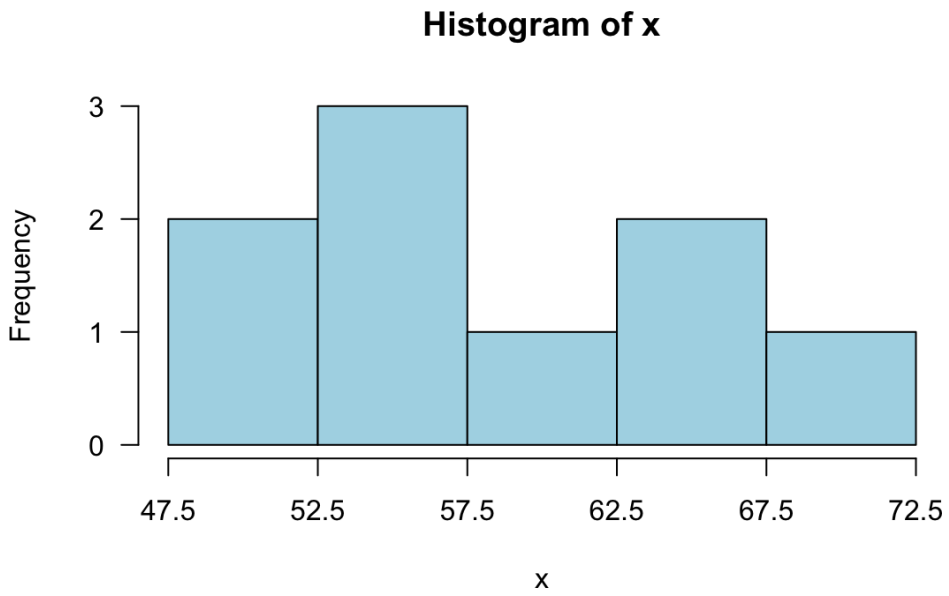
Bin boundaries

```
hist(x, breaks = seq(47.5, 72.5, 5), col = "lightblue")
```



Bin boundaries

```
# presentation issues
hist(x, breaks = seq(47.5, 72.5, 5), col = "lightblue",
     axes = FALSE)
axis(1, at = seq(47.5, 72.5, 5))
axis(2, at = 0:3)
```



Frequency, Relative Frequency, Density

mids	freq	relfreq	density
350	1	0.05556	0.00056
450	4	0.22222	0.00222
550	5	0.27778	0.00278
650	4	0.22222	0.00222
750	4	0.22222	0.00222

- the sum of relative frequencies is 1
- the sum of densities x binwidth is 1

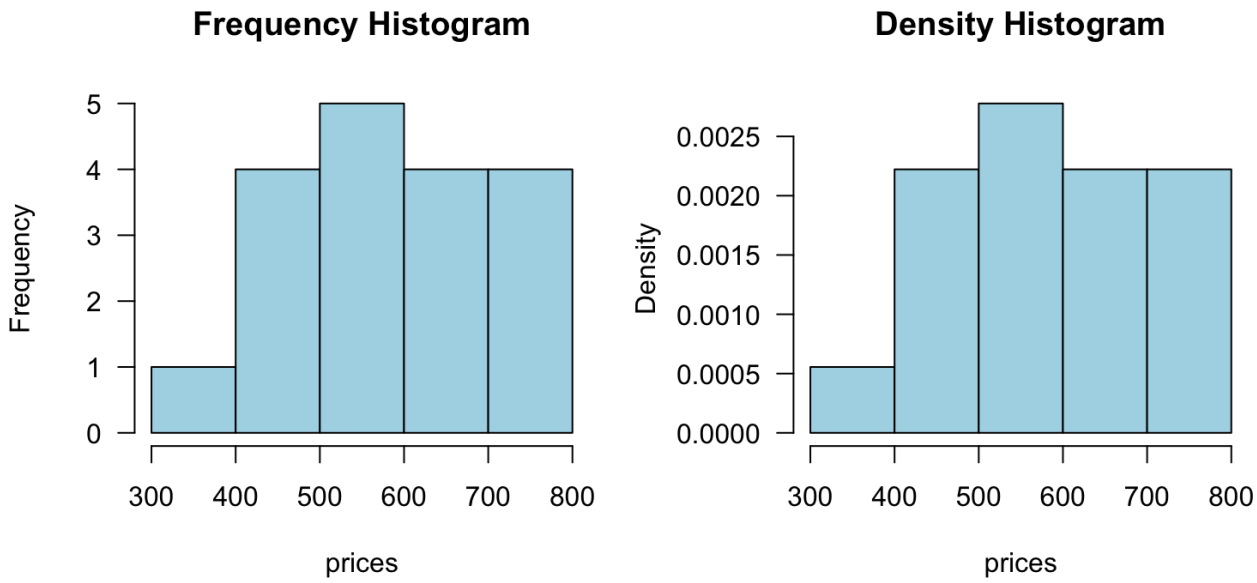
Frequency, Relative Frequency, Density

```r
x <- hist(prices, breaks = seq(300, 800, 100), plot = FALSE)
x
```

```
## $breaks
## [1] 300 400 500 600 700 800
##
## $counts
## [1] 1 4 5 4 4
##
## $density
## [1] 0.00055556 0.00222222 0.00277778 0.00222222 0.00222222
##
## $mids
## [1] 350 450 550 650 750
##
## $xname
## [1] "prices"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

Frequency vs. Density Histogram (`freq = FALSE`)

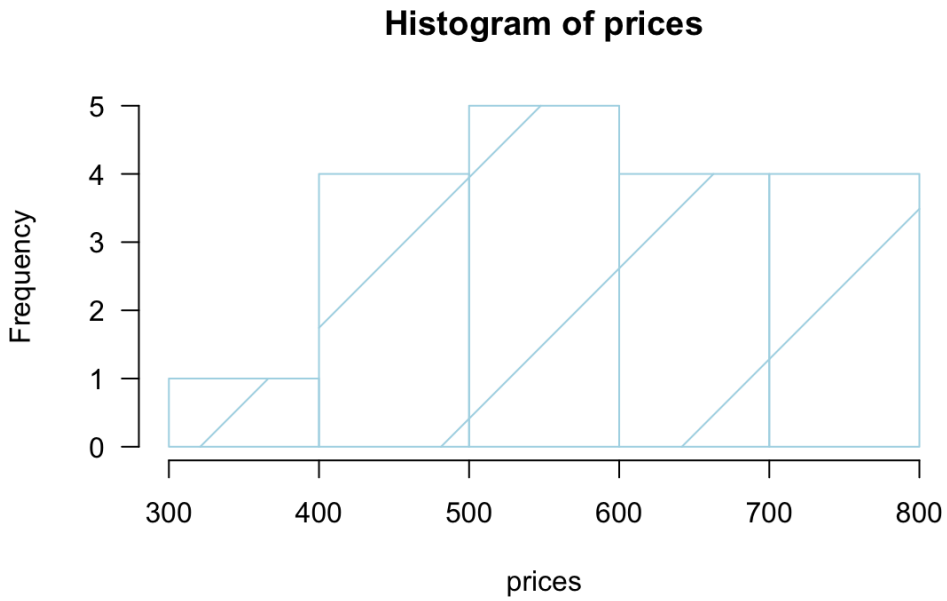
```
oldpar <- par(mfrow = c(1, 2), las = 1)
hist(prices, breaks = c(300, 400, 500, 600, 700, 800),
     col = "lightblue", main = "Frequency Histogram")
hist(prices, breaks = c(300, 400, 500, 600, 700, 800),
     freq = FALSE, col = "lightblue", ylab = "",
     main = "Density Histogram")
mtext("Density", side = 2, line = 4, las = 3)
```




```
par(oldpar)
```

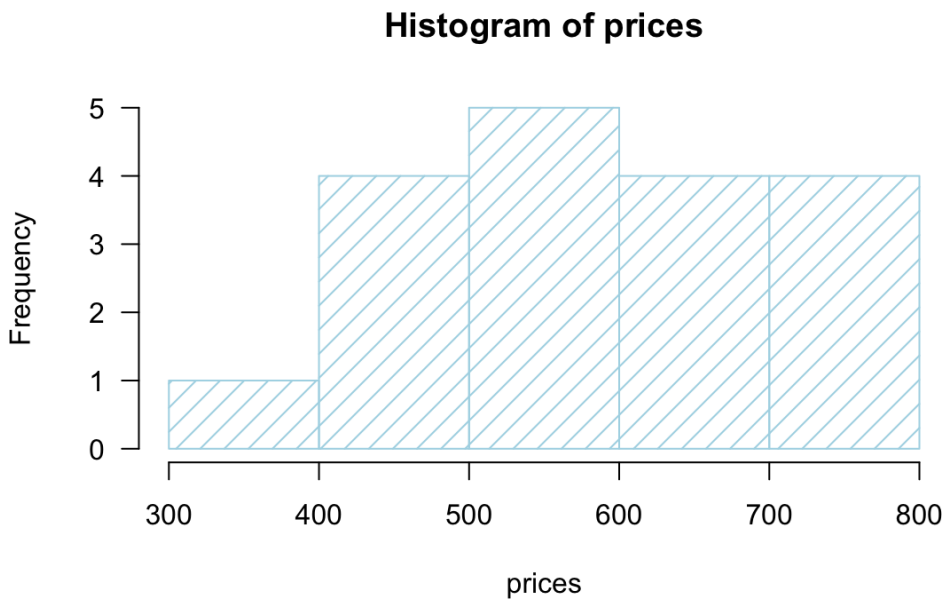
Don't use `density = TRUE`

```
hist(prices, breaks = seq(300, 800, 100), col = "lightblue",
     density = TRUE)
```



Don't use `density = TRUE`

```
hist(prices, breaks = seq(300, 800, 100), col = "lightblue",
     density = 10)
```

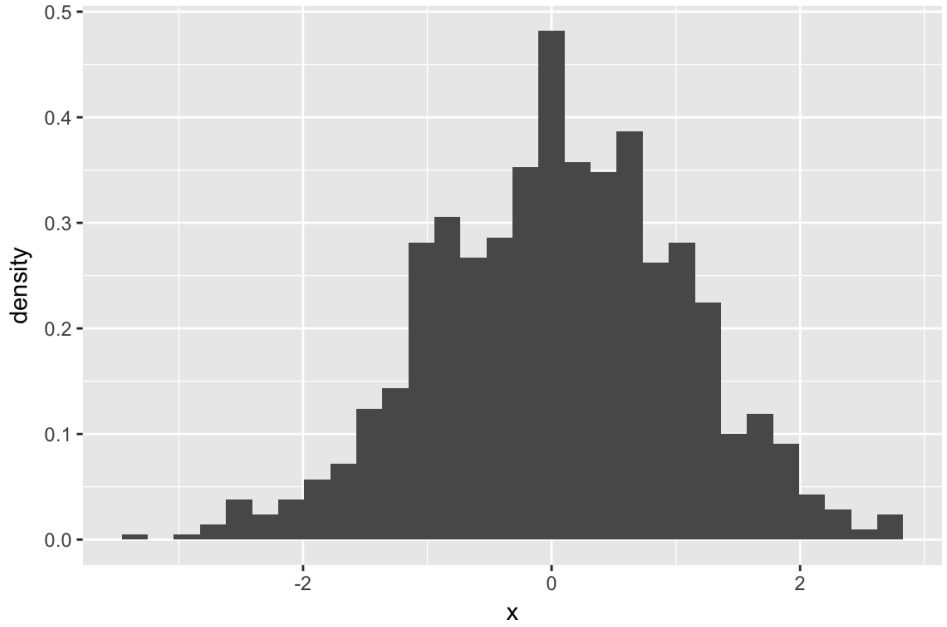



```
sum(TRUE)
```

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

Density histogram `ggplot2`

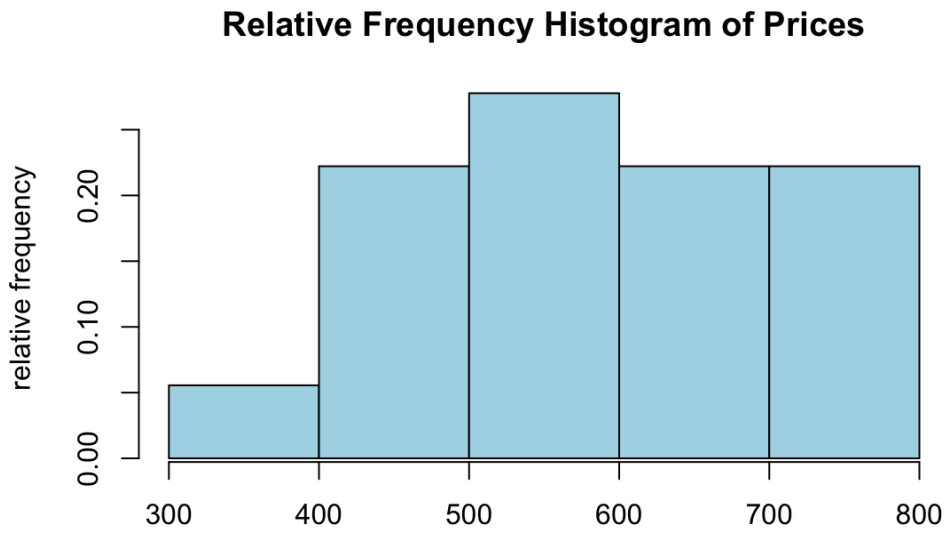
```
df2 <- data.frame(x = rnorm(1000))
ggplot(df2, aes(x, y = ..density..)) + geom_histogram()
```



Relative frequency histogram

Method # 1 Use `barplot()`

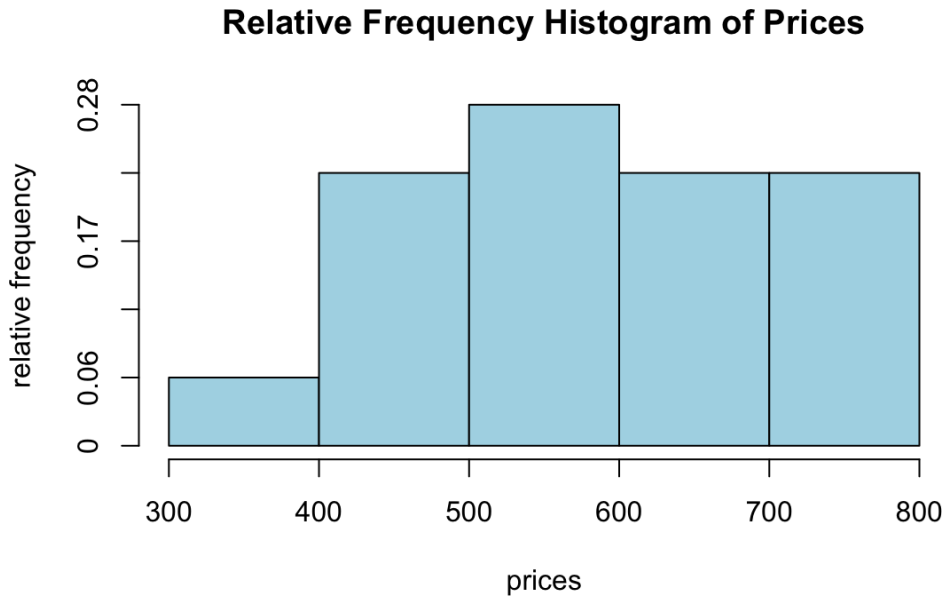
```
x <- barplot(df$relfreq, space = 0, col = "lightblue",
             ylab = "relative frequency")
# axis(1) to see the scale on the x-axis
axis(1, at = 0:5, labels = seq(300, 800, 100))
title("Relative Frequency Histogram of Prices")
```



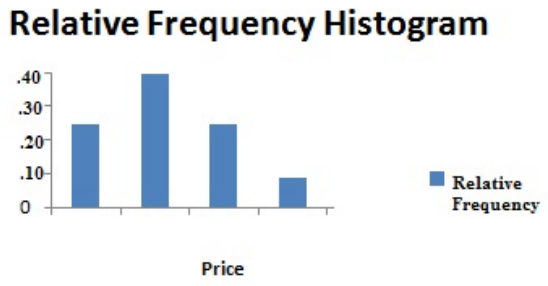
Relative frequency histogram

Method # 2 Use `hist()` and change the y-axis tick mark labels… but be careful!!

```
hist(prices, breaks = c(300, 400, 500, 600, 700, 800),
     col = "lightblue", yaxt = "n",
     ylab = "relative frequency",
     main = "Relative Frequency Histogram of Prices")
axis(2, at = 0:5, labels = round((0:5)/18,2))
```



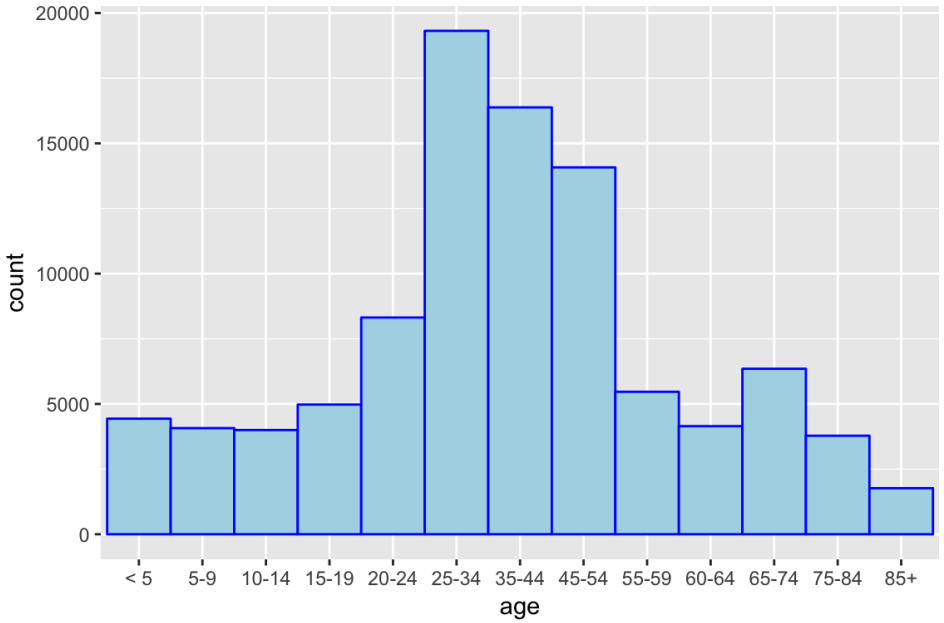
Example from the web



Source: http://www.statisticshowto.com/relative-frequency-histogram-2/

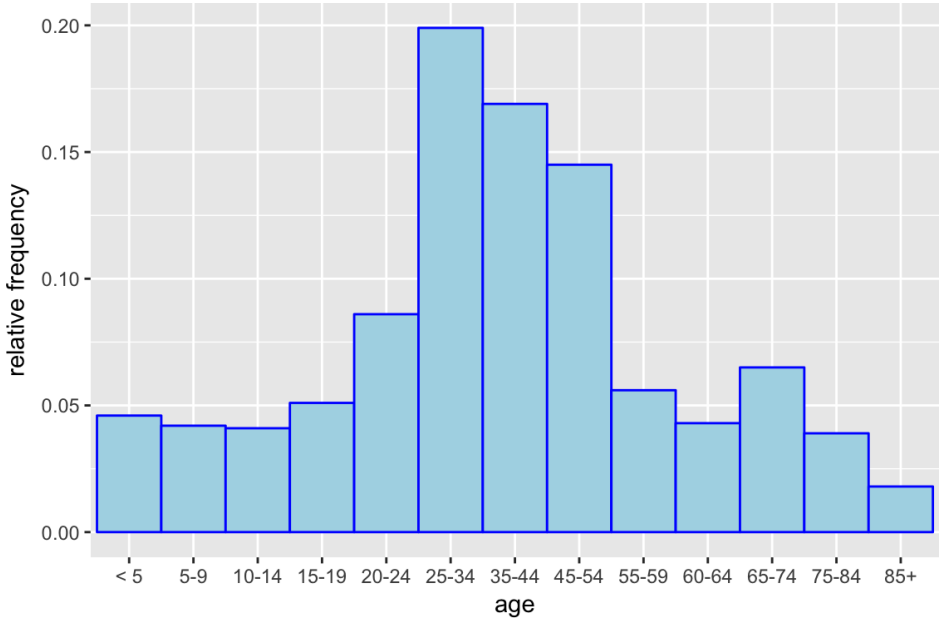
What's wrong with this histogram?

```r
# Use geom_col since we already have frequency counts
# This is an example of what not to do
df <- read.csv("zip10027census2000.csv")
df$age <- factor(df$age, levels = df$age)
g0 <- ggplot(df, aes(x = age, y = pop)) +
    geom_col(width = 1, color = "blue", fill = "lightblue") +
    ylab("count")
g0
```



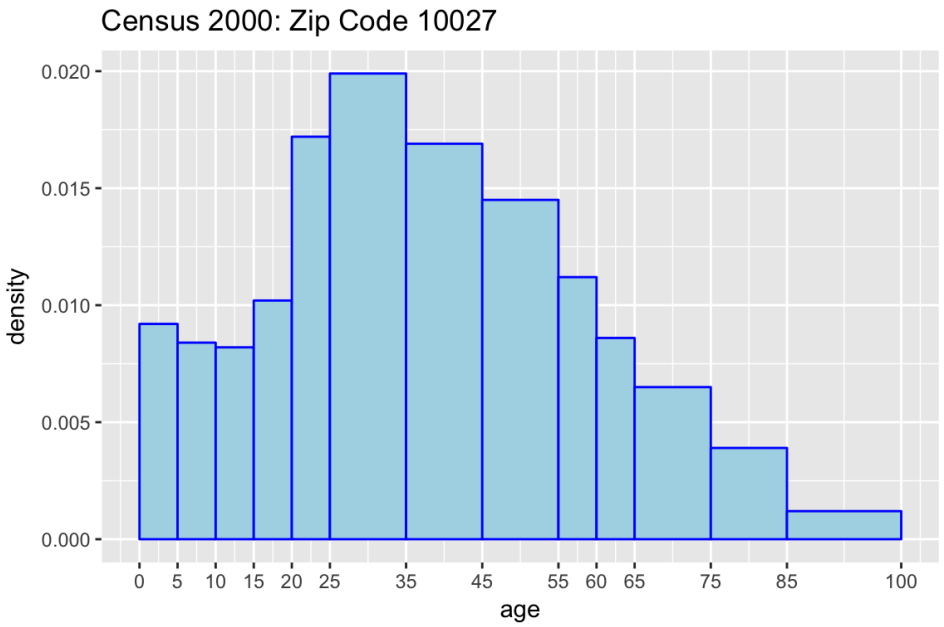
Relative frequency histogram

```
# Doesn't fix the problem
ggplot(df, aes(x = age, y = percent/100)) +
    geom_col(width = 1, color = "blue", fill = "lightblue") +
    ylab("relative frequency")
```



Density histogram with unequal bin (or class) widths

```
g2 <- ggplot(df, aes(x = center, y = percent/(100*binwidth),
                width = binwidth)) +
    geom_col(color = "blue", fill = "lightblue") +
    ylab("density") + xlab("age") +
    scale_x_continuous(breaks = c(0, df$breaks)) +
    ggtitle("Census 2000: Zip Code 10027")
g2
```



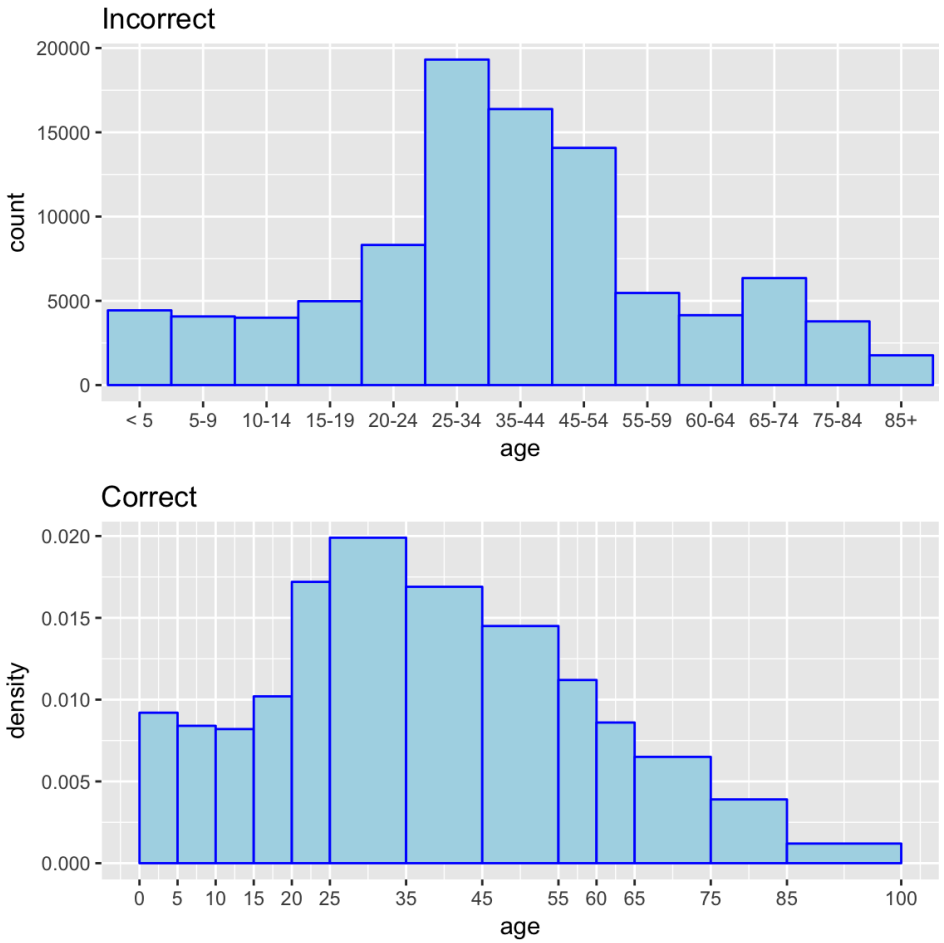
Density = RelFreq / Binwidth

```
library(dplyr)
kdf <- df %>% transmute(Class = age, Frequency = pop,
                        RelFreq = round(pop/sum(pop),3),
                        ClassWidth = binwidth,
                        Density = round(RelFreq/ClassWidth,3))
knitr::kable(kdf)
```

Class	Frequency	RelFreq	ClassWidth	Density
< 5	4435	0.046	5	0.009
5-9	4072	0.042	5	0.008
10-14	3999	0.041	5	0.008
15-19	4977	0.051	5	0.010
20-24	8316	0.086	5	0.017
25-34	19317	0.199	10	0.020
35-44	16380	0.169	10	0.017
45-54	14077	0.145	10	0.014
55-59	5467	0.056	5	0.011
60-64	4148	0.043	5	0.009
65-74	6350	0.065	10	0.007
75-84	3781	0.039	10	0.004
85+	1767	0.018	15	0.001

Compare the histograms

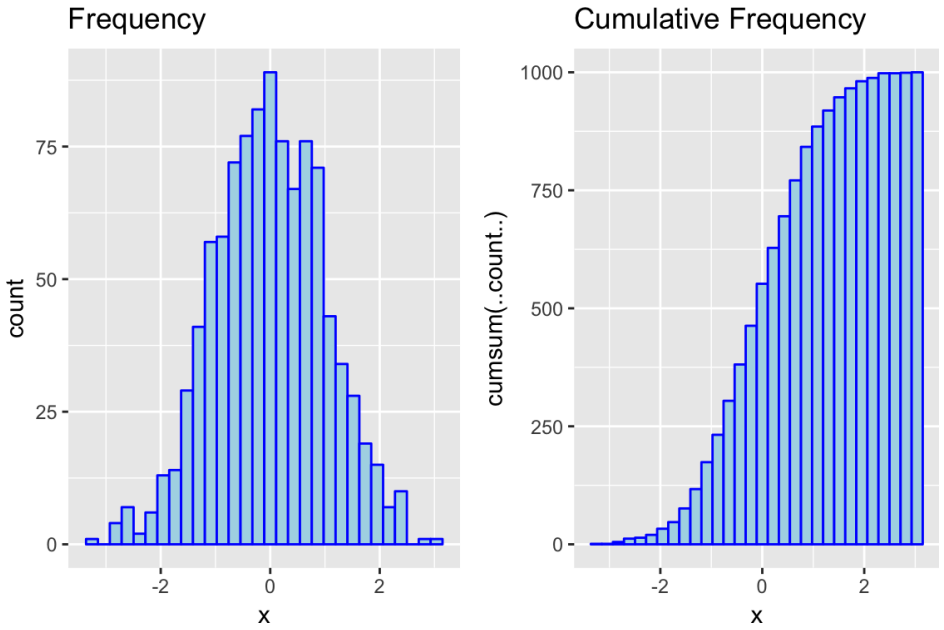
```
library(gridExtra)
grid.arrange(g0 + ggtitle ("Incorrect"),
             g2 + ggtitle ("Correct"))
```



Source: https://factfinder.census.gov/

Cumulative frequency histogram

```
df <- data.frame(x = rnorm(1000))
g1 <- ggplot(df, aes(x = x)) +
    geom_histogram(color = "blue", fill = "lightblue") +
    ggtitle ("Frequency")
g2 <- ggplot(df, aes(x = x)) +
    geom_histogram(aes(y = cumsum(..count..)),
                   color = "blue", fill = "lightblue") +
    ggtitle("Cumulative Frequency")
grid.arrange(g1, g2, nrow = 1)
```



Binwidth

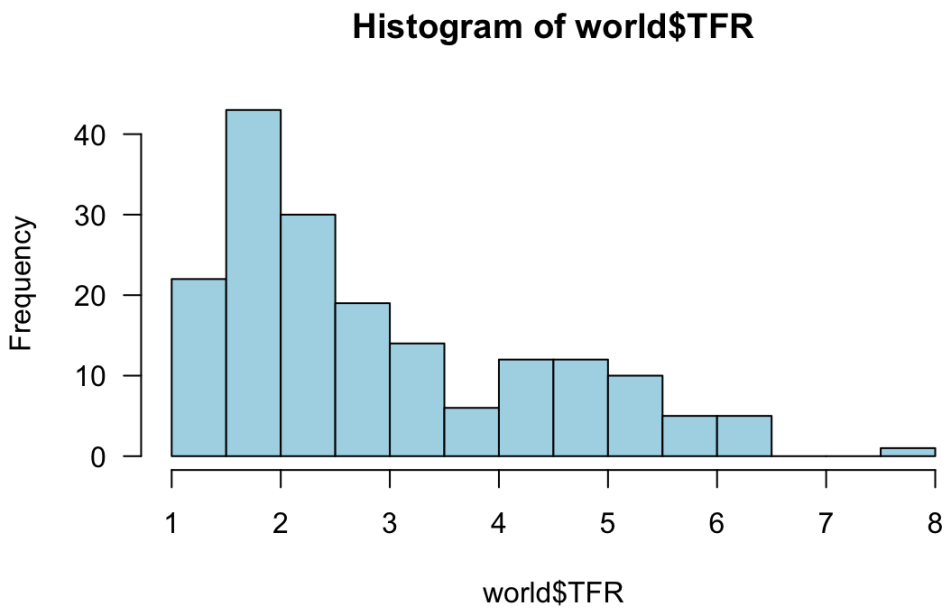
```
'stat_bin()' using 'bins = 30'. Pick better
value with 'binwidth'.
```

Histograms

```
hist(world$TFR, col = "lightblue")
```

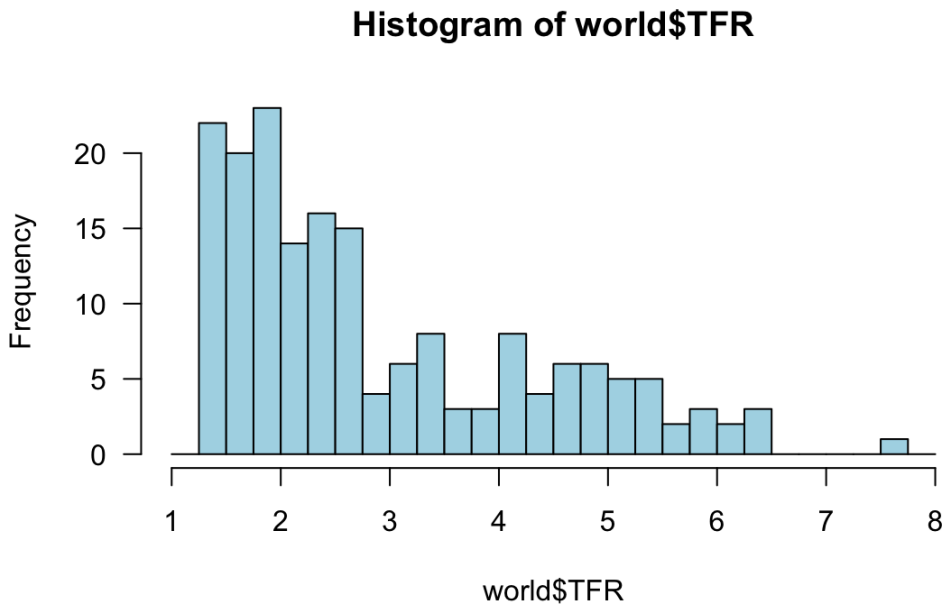
Histograms

```
hist(world$TFR, col = "lightblue", breaks = 10)
```



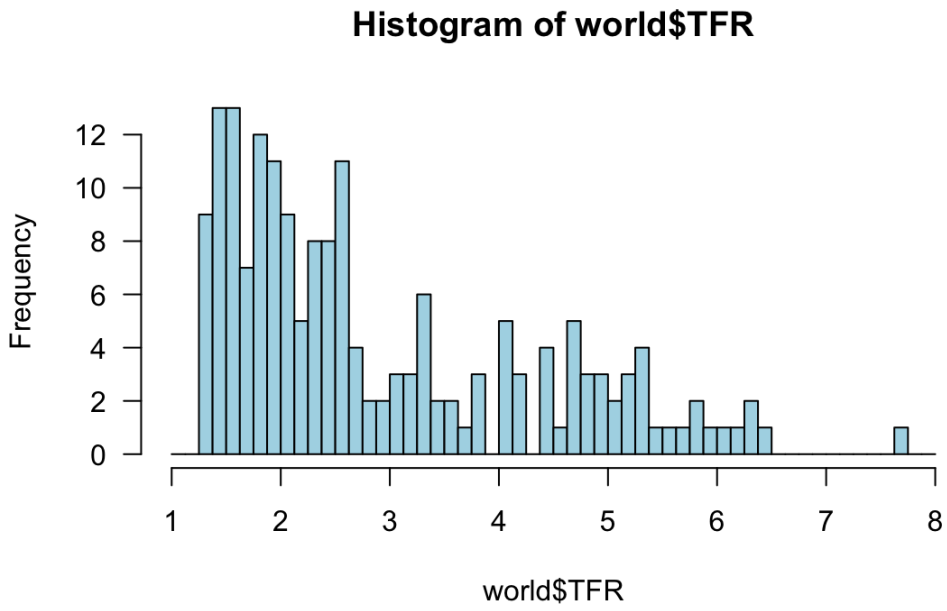
Histograms

```
hist(world$TFR, col = "lightblue",
     breaks = seq(1 , 8, .25))
```



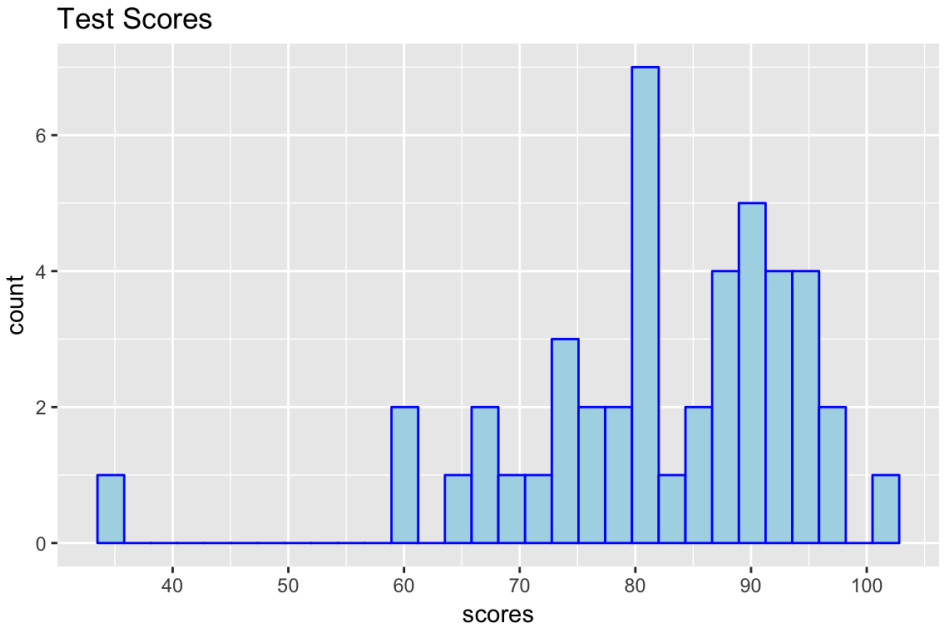
Histograms

```
hist(world$TFR, col = "lightblue",
     breaks = seq(1 , 8, .125))
```



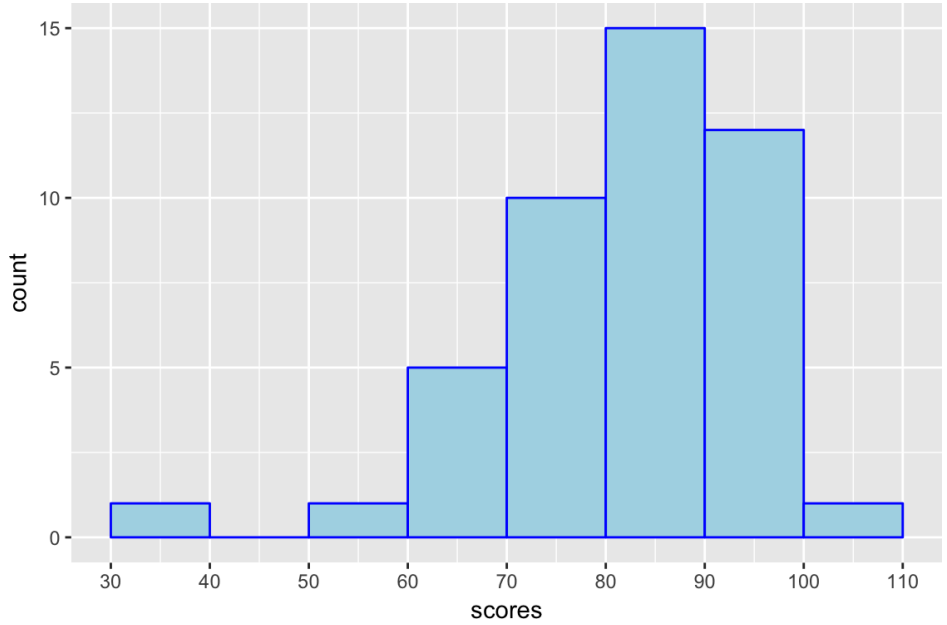
Histograms

```
df <- data.frame(scores = c(35, 59, 61, 64, 66, 66, 70, 72, 73, 74,
                            75, 76, 76, 78, 79, 80, 80, 81, 81, 82,
                            82, 82, 84, 86, 86, 88, 88, 88, 88, 89,
                            89, 90, 91,91, 92, 92, 92, 92, 94, 94,
                            94, 94, 96, 98, 102))
ggplot(df, aes(x = scores)) +
    geom_histogram(color = "blue", fill = "lightblue") +
    scale_x_continuous(breaks = seq(30, 100, 10)) +
    ggtitle("Test Scores")
```



Fewer bins

```
ggplot(df, aes(x = scores)) +
    geom_histogram(color = "blue", fill = "lightblue",
                   breaks = seq(30, 110, 10)) +
        scale_x_continuous(breaks = seq(30, 110, 10))
```



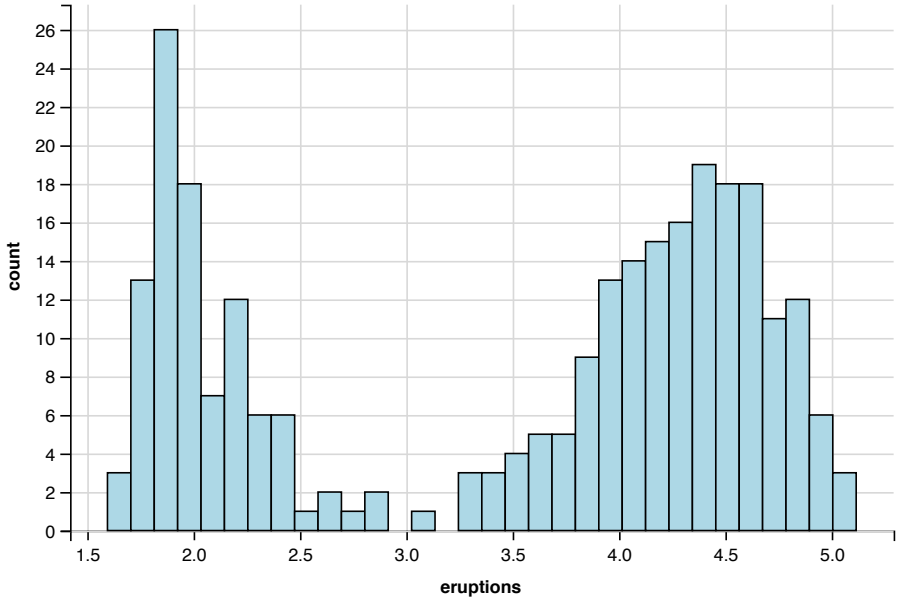
ggvis

- built w/ Vega, Shiny
- Vega <– D3 + …
- Shiny <– R + web (HTML, CSS, SVG, JavaScript)
- code looks like `ggplot2` + `dplyr`
- best use: EDA
- More info, tutorials: https://ggvis.rstudio.com/

ggvis

```
library(ggvis)
faithful %>% ggvis(~eruptions) %>%
    layer_histograms(fill := "lightblue",
                     width = input_slider(0.01, 1,
                                          value = .1,
                                          step = .1,
                                          label = "width"))
```

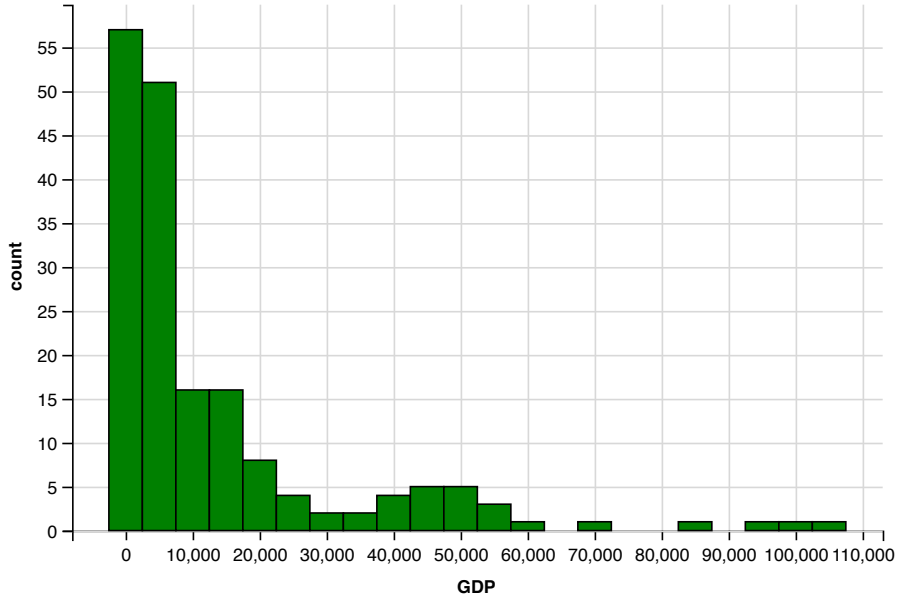
```
## Warning: Can't output dynamic/interactive ggvis plots in a knitr document.
## Generating a static (non-dynamic, non-interactive) version of the plot.
```



GDP

```
df <-read.csv("countries2012.csv")
df %>% ggvis(~GDP) %>%
    layer_histograms(fill := "green",
                     width = input_slider(500, 10000,
                                          value = 5000,
                                          step = 500,
                                          label = "width"))
```

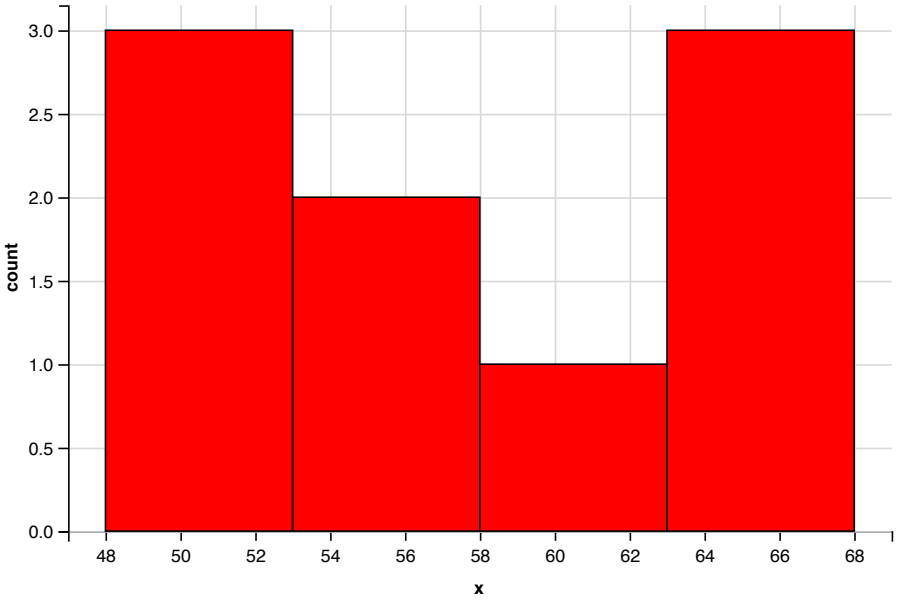
```
## Warning: Can't output dynamic/interactive ggvis plots in a knitr document.
## Generating a static (non-dynamic, non-interactive) version of the plot.
```



Center

```
df <- data.frame(x = c(50, 51, 53, 55, 56, 60, 65, 65, 68))
df %>% ggvis(~x) %>%
    layer_histograms(fill := "red",
                     width = input_slider(1, 10,
                                          value = 5,
                                          step = 1,
                                          label = "width"),
                     center = input_slider(0, 1,
                                           value = .5,
                                           step = .5,
                                           label = "center"))
```

```
## Warning: Can't output dynamic/interactive ggvis plots in a knitr document.
## Generating a static (non-dynamic, non-interactive) version of the plot.
```



Boundary

```
df %>% ggvis(~x) %>%
    layer_histograms(fill := "red",
                     width = input_slider(1, 10,
                                          value = 5,
                                          step = 1,
                                          label = "width"),
                     boundary = input_slider(47.5, 50,
                                           value = 50,
                                           step = .5,
                                           label = "boundary"))
```

```
## Warning: Can't output dynamic/interactive ggvis plots in a knitr document.
## Generating a static (non-dynamic, non-interactive) version of the plot.
```