Some dos and don'ts"

Daniel Anderson

Remember

- Get R installed: https://cran.r-project.org
- Get RStudio installed: https://www.rstudio.com/products/rstudio/download/

NOTE: If you need help with either of the above, please contact me. I'd like everybody to be ready to go **before** we need to use it. Best to get it installed now and make sure it's working so we can troubleshoot if not.

> “Above all else, show the data”

Edward Tufte

Hypothetical example

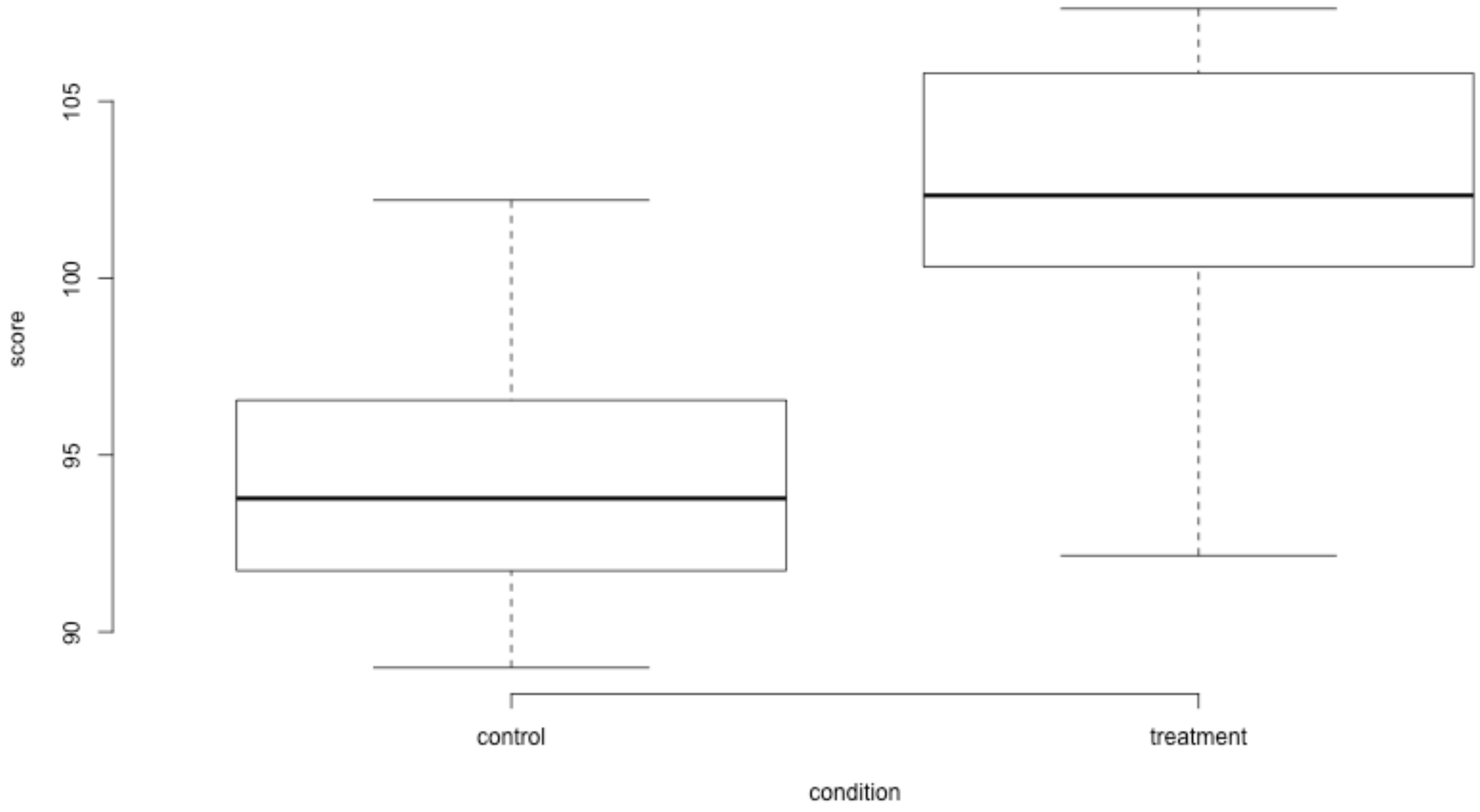
- School wants to try out a new reading intervention
- Work with researchers at the UO to design a study
- Kindergarten students who are behind their peers in literacy are selected
- Randomly assign half the students to the intervention, the rest continue with "typical" instruction
- Now the study is over - how do we tell if it worked? Visualize it! (and other stuff too)

Barplots

(tried and true)

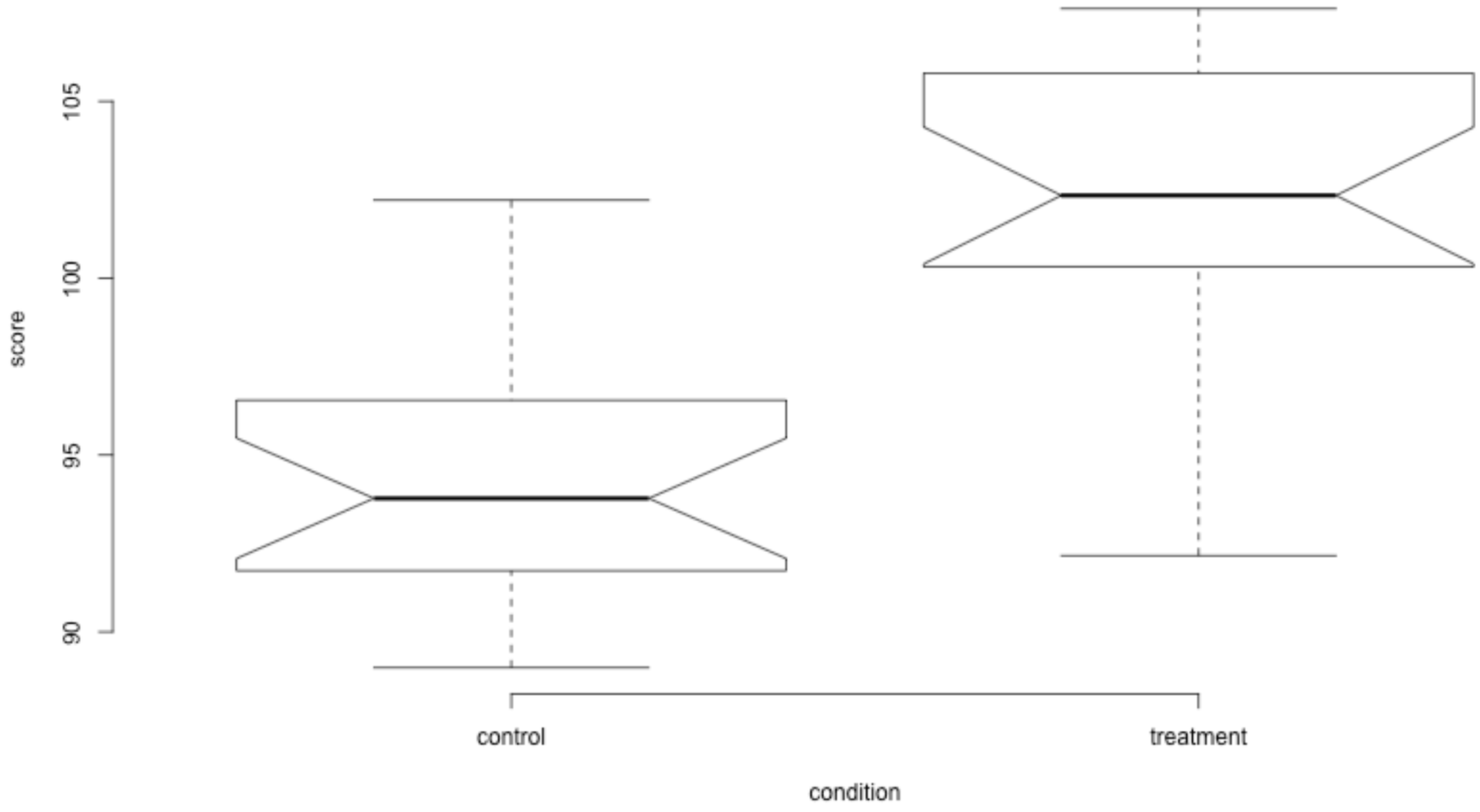
Boxplots

(tried and true)



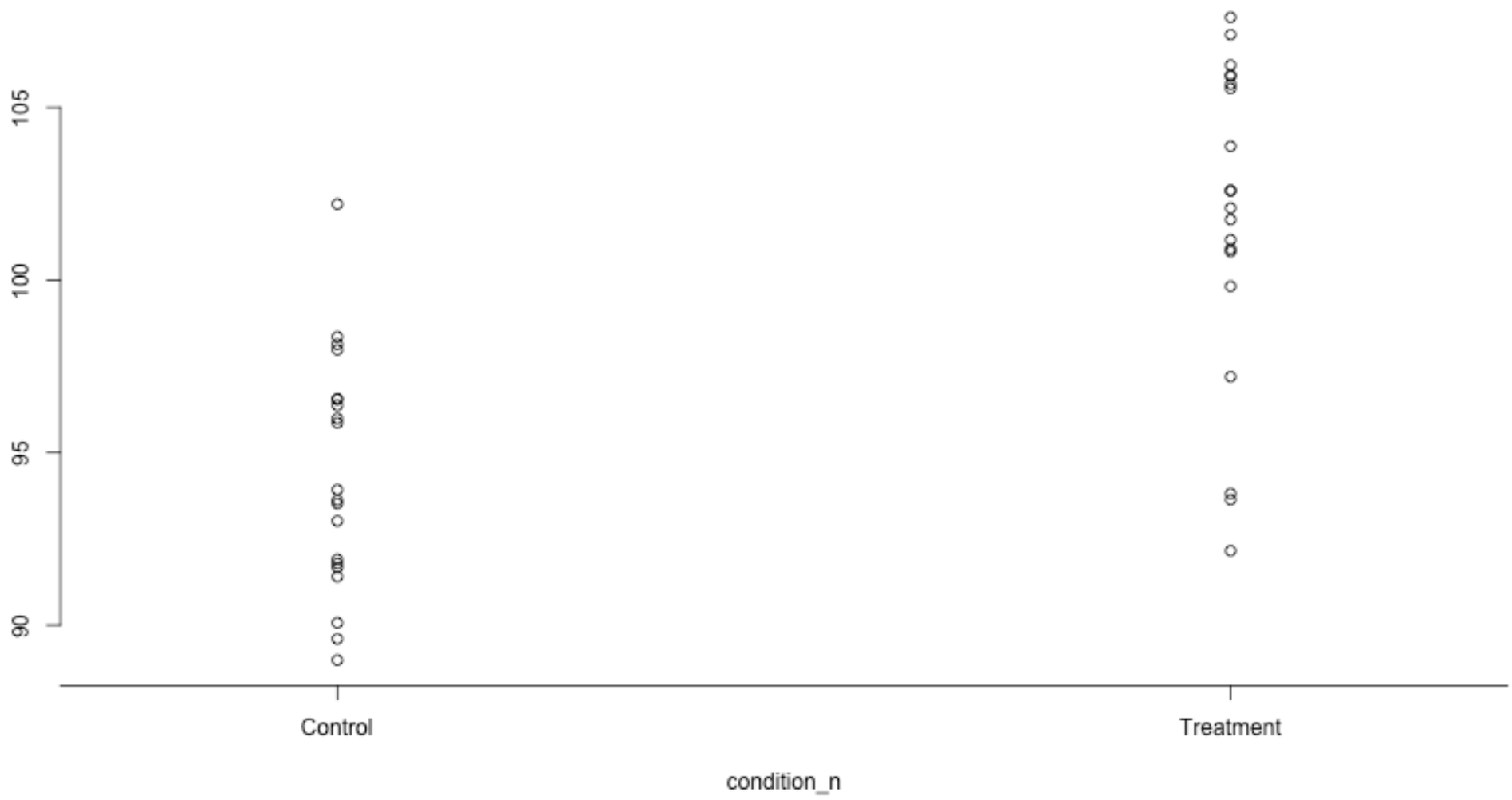
Notched boxplots

(slightly better)



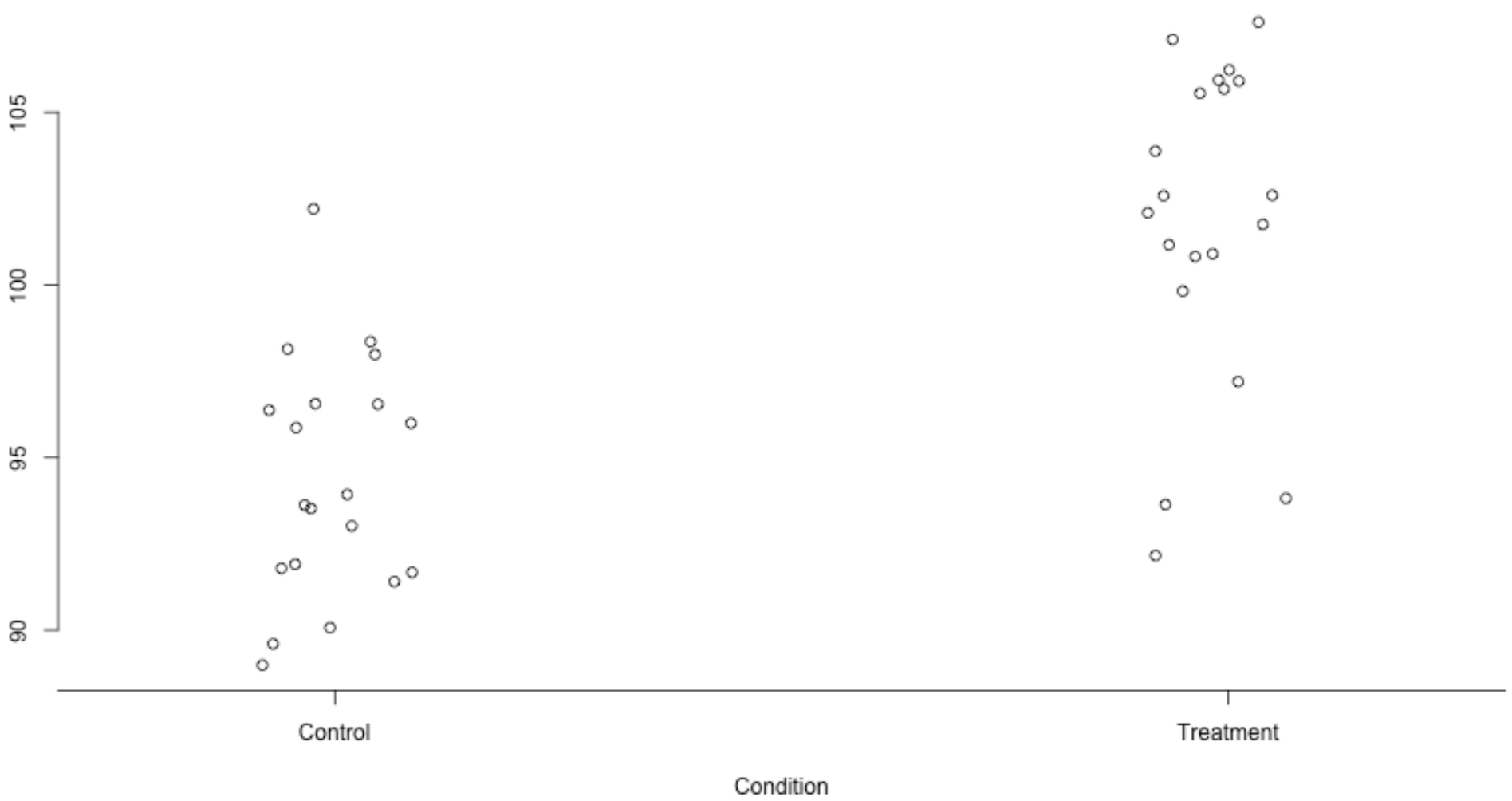
Stripcharts

Show the data!

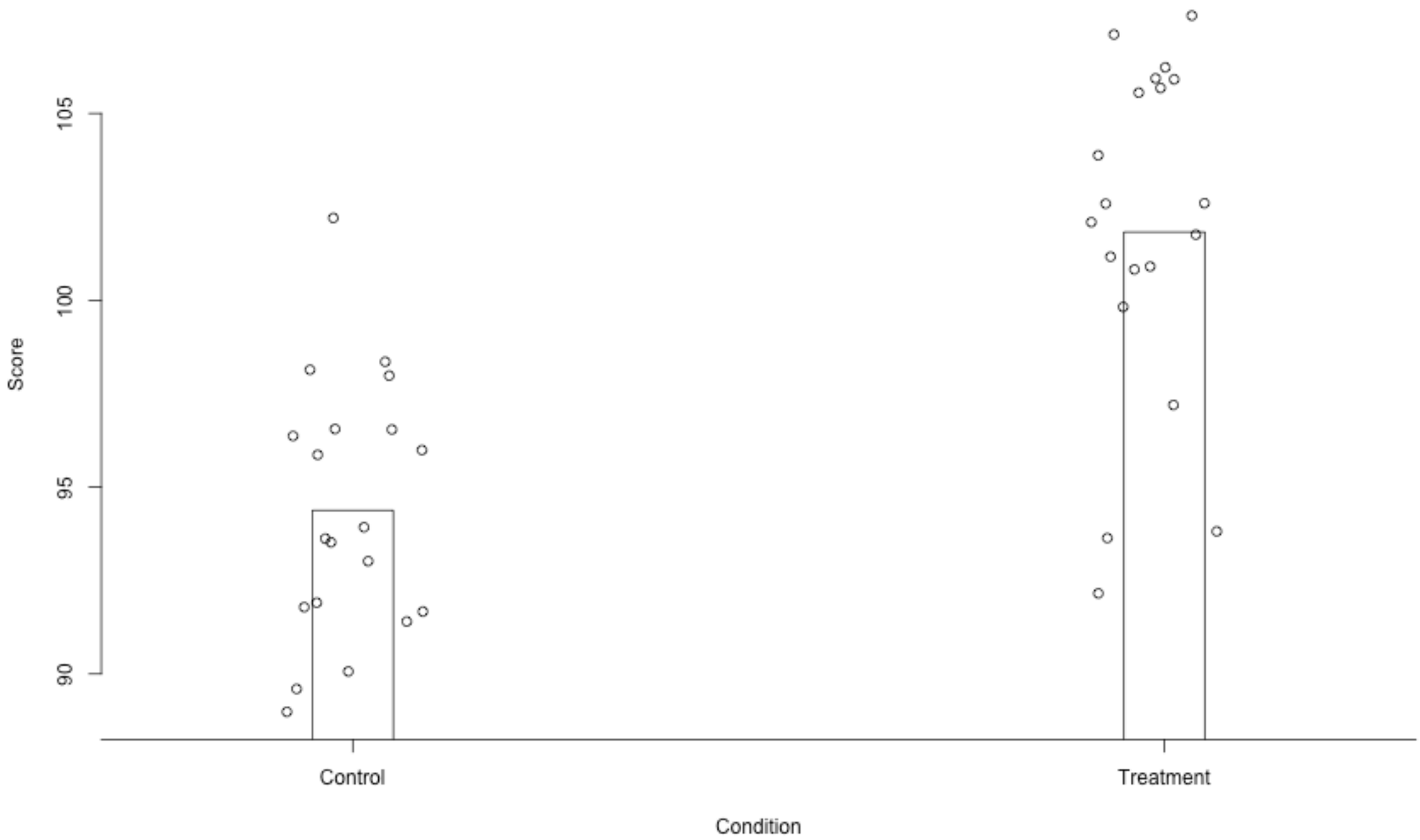


Jittered stripcharts

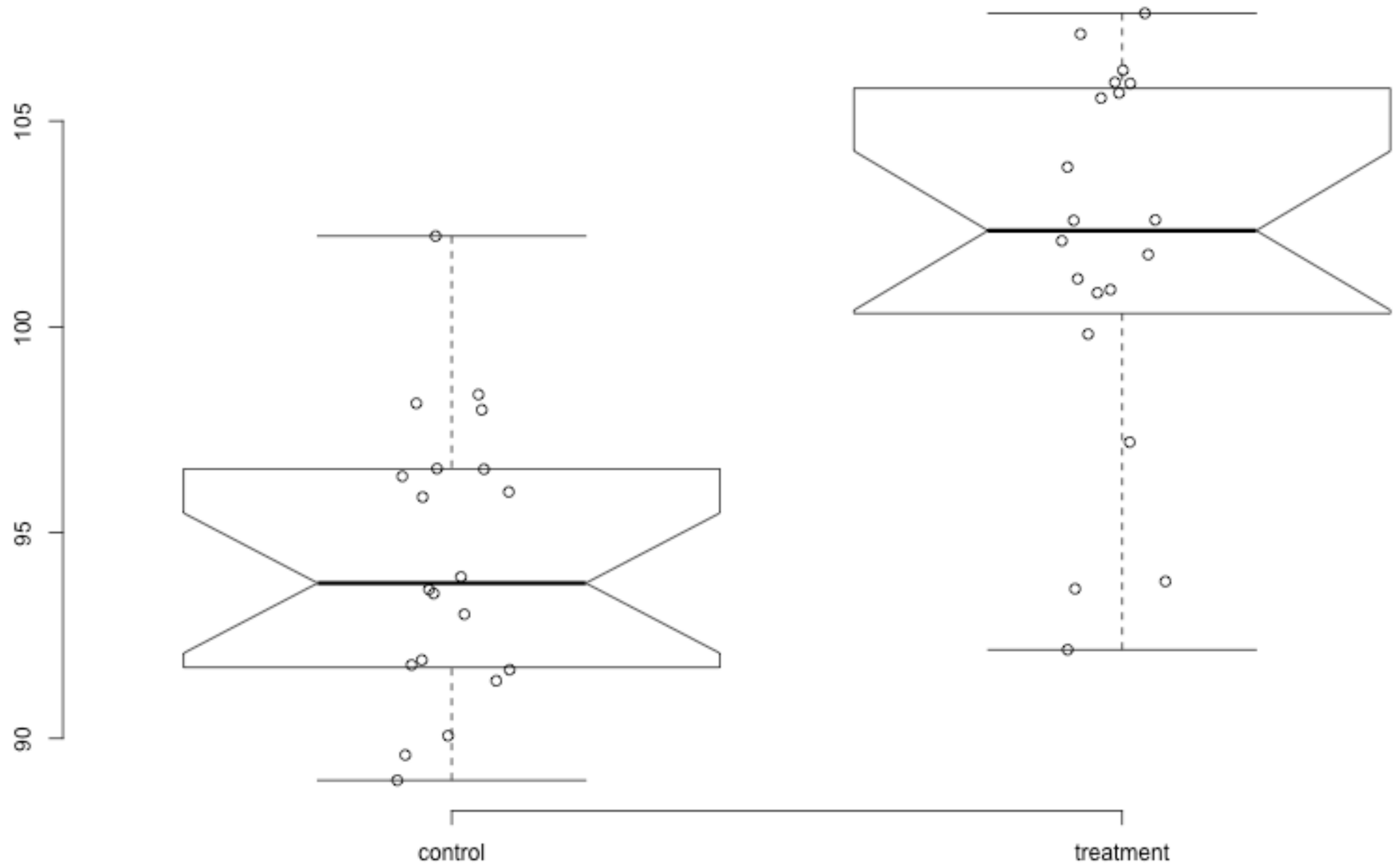
Show the data!



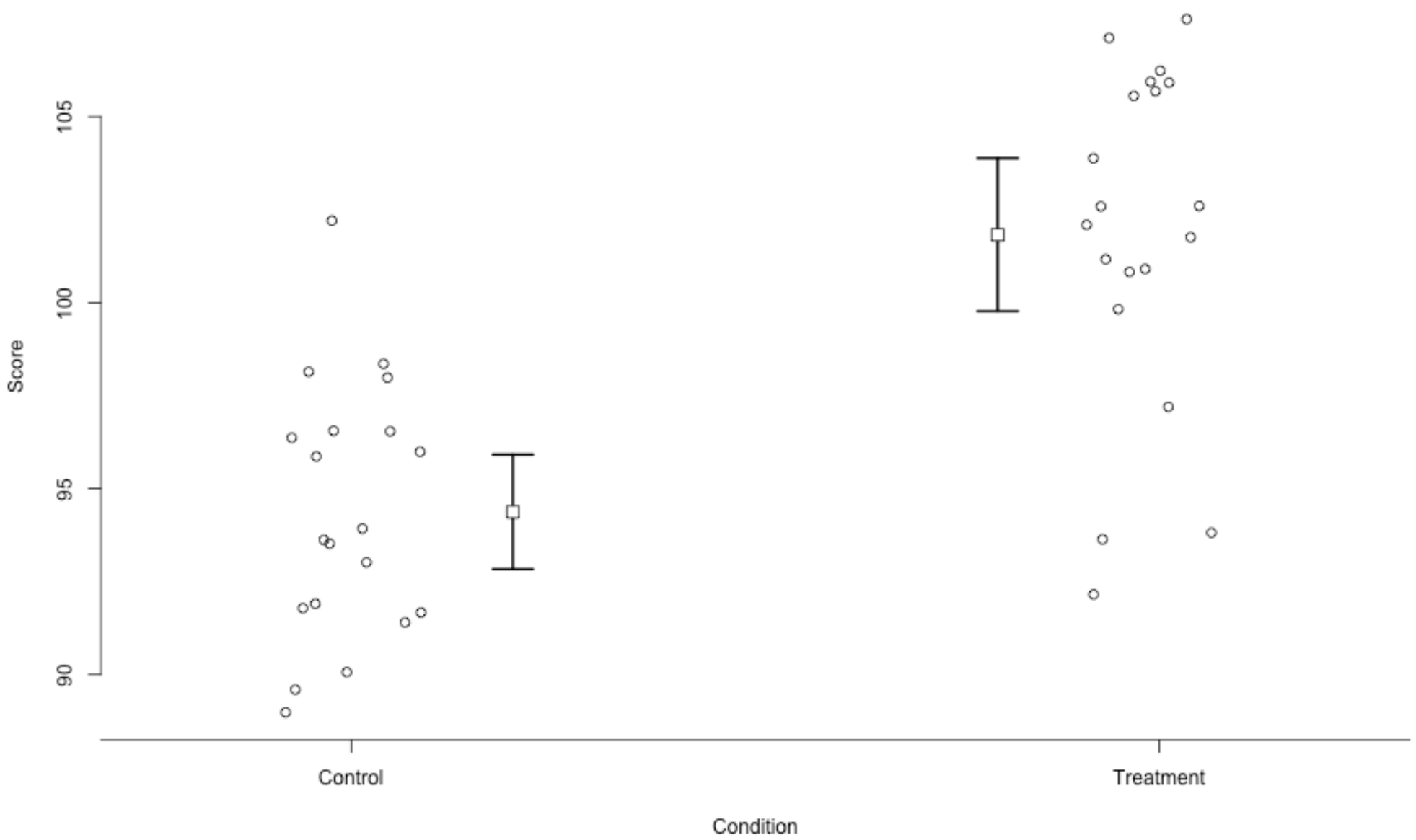
Combine barplots and jittered stripcharts



Combine boxplots and jittered stripcharts



Best?

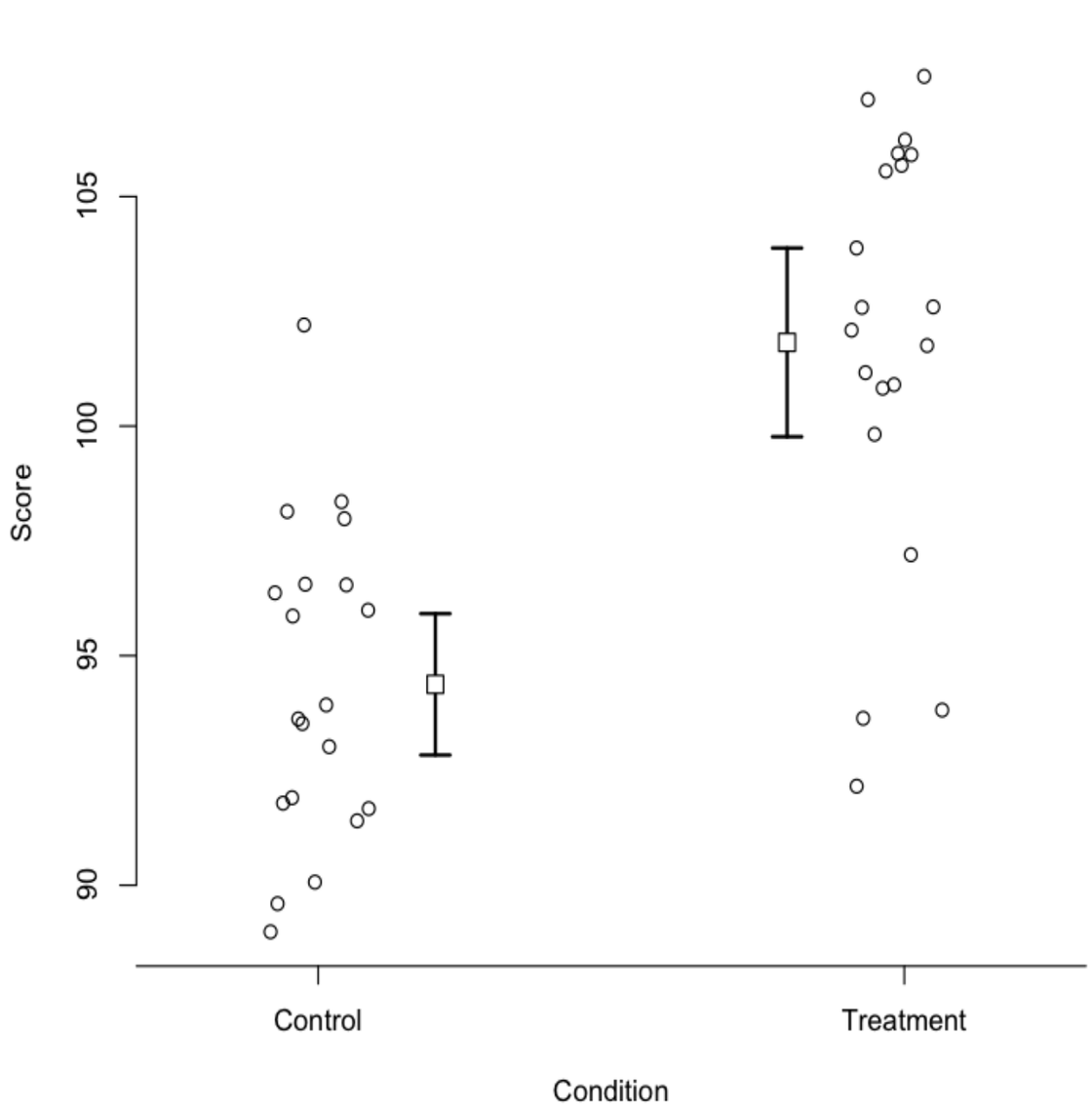


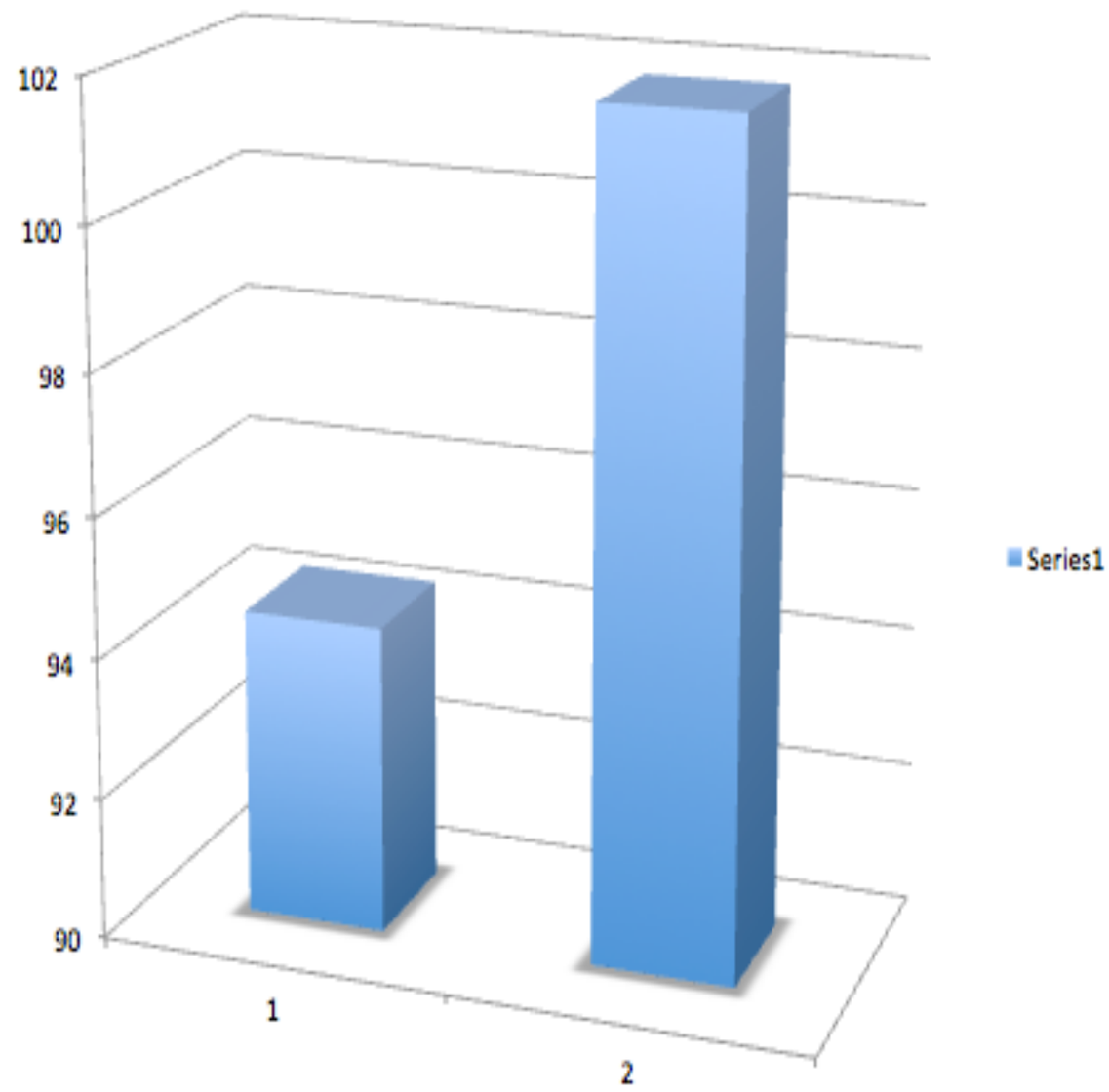
Some things to avoid

- 3D plots
- Pie charts
- Dual axes
- Restricted axes
- Unnecessary frills (colors, etc)
    - Show the data as plainly as possible. Let the data speak!

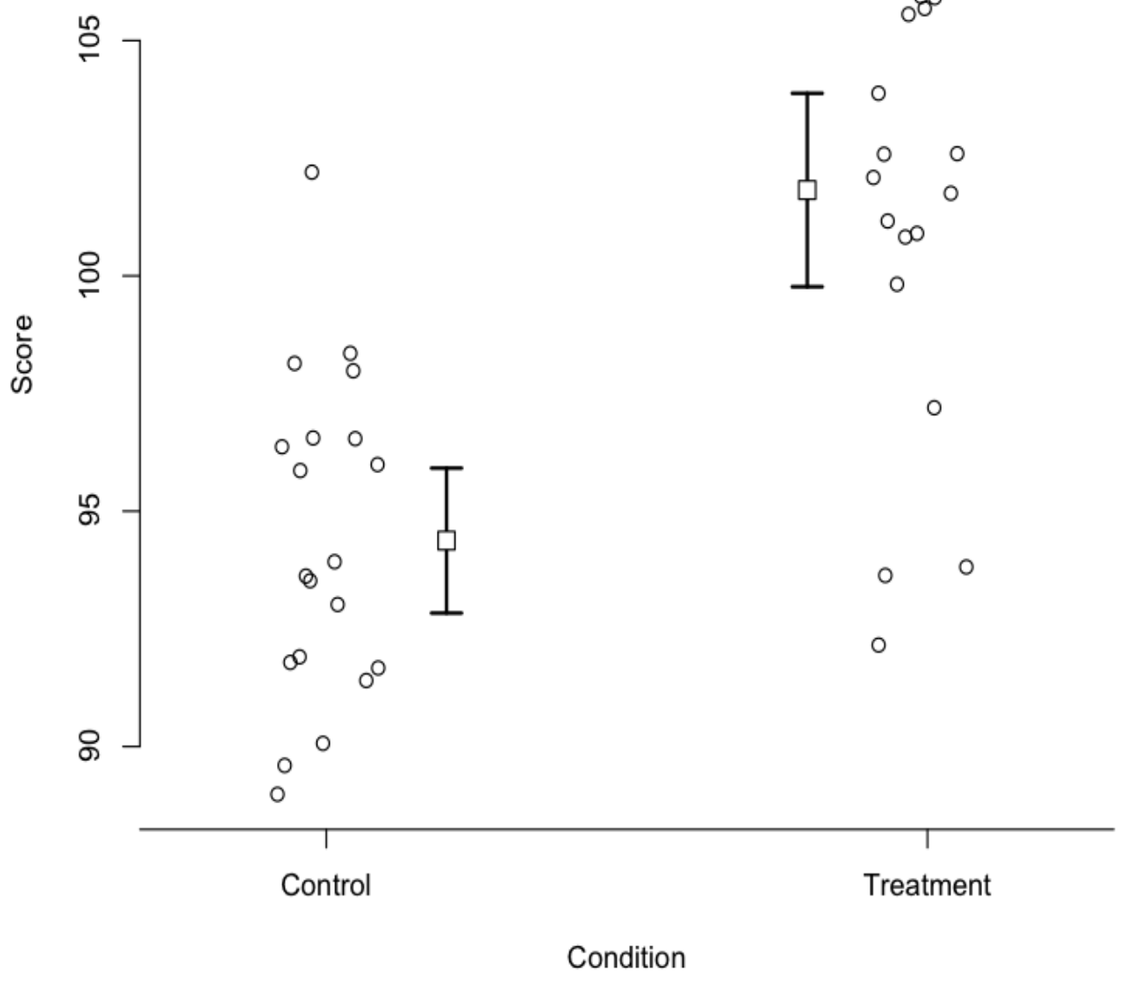
NOTE: The following 10 slides (and the previous plot) inspired/taken from Karl Broman's presentation on graphs (see here)

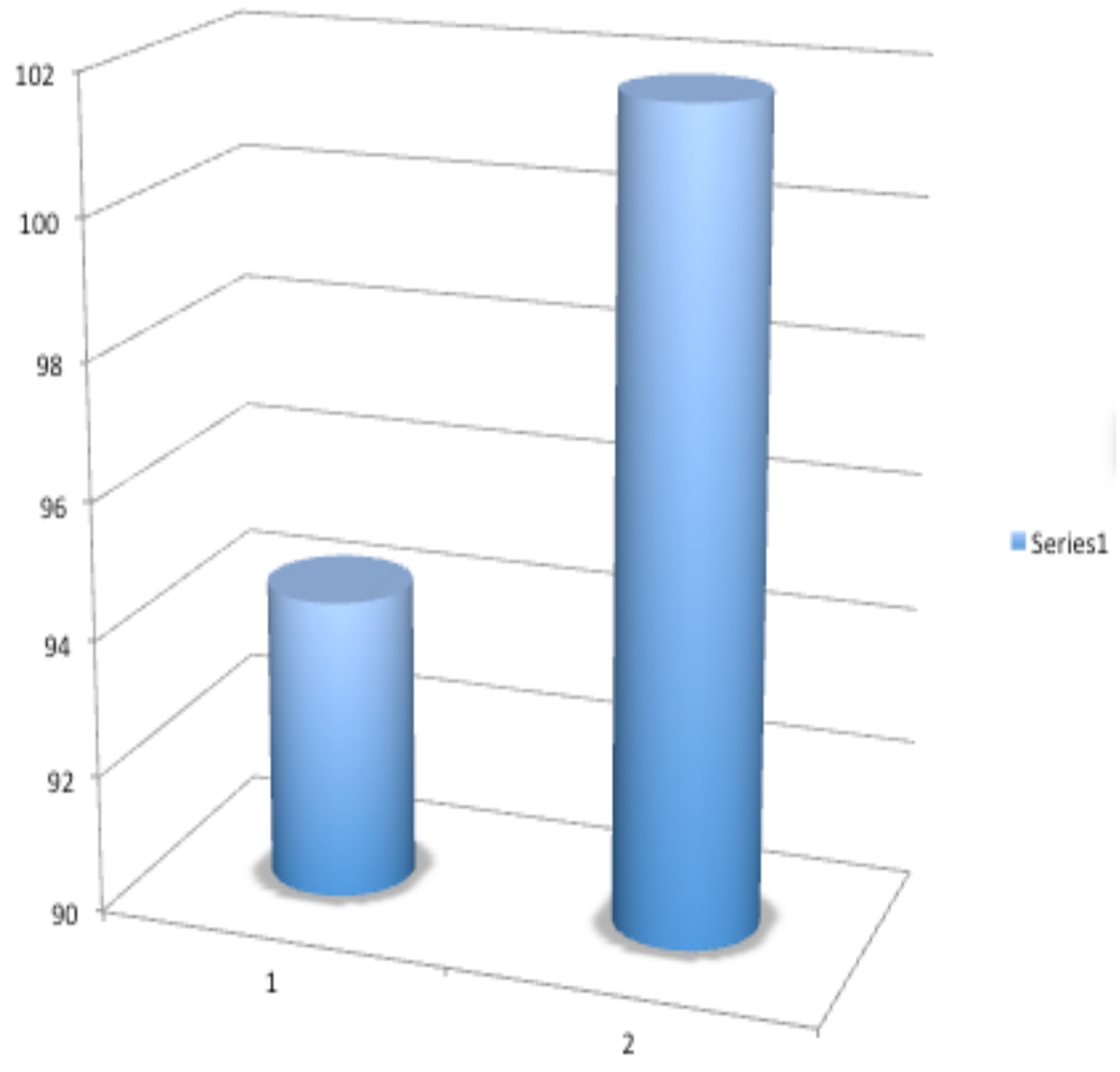
Examples


Score
105
100
95
90
Control
Treatment
Condition

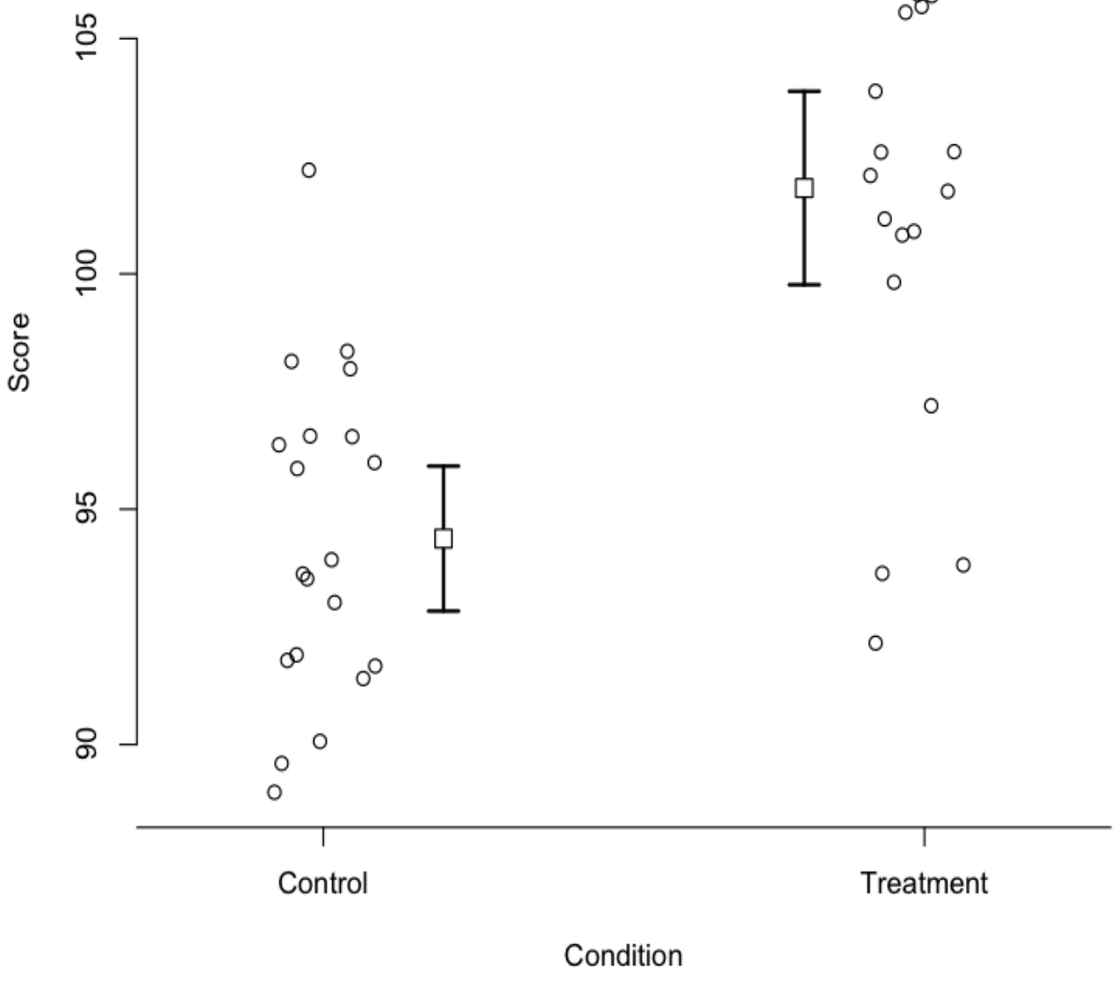

102
100
98
96
94
92
90
1
2
Series1

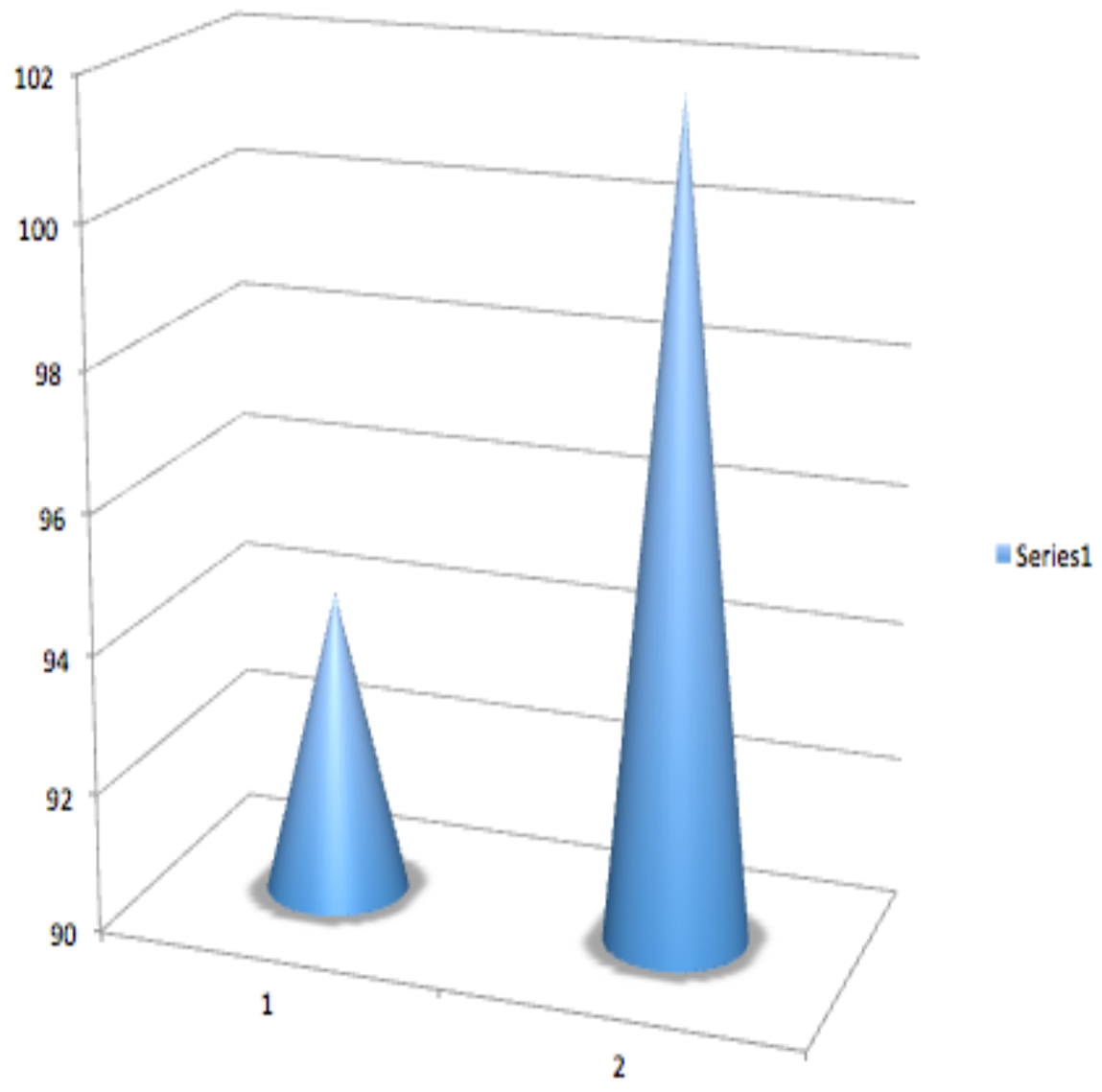
Examples


Score
105
100
95
90
Control
Treatment
Condition

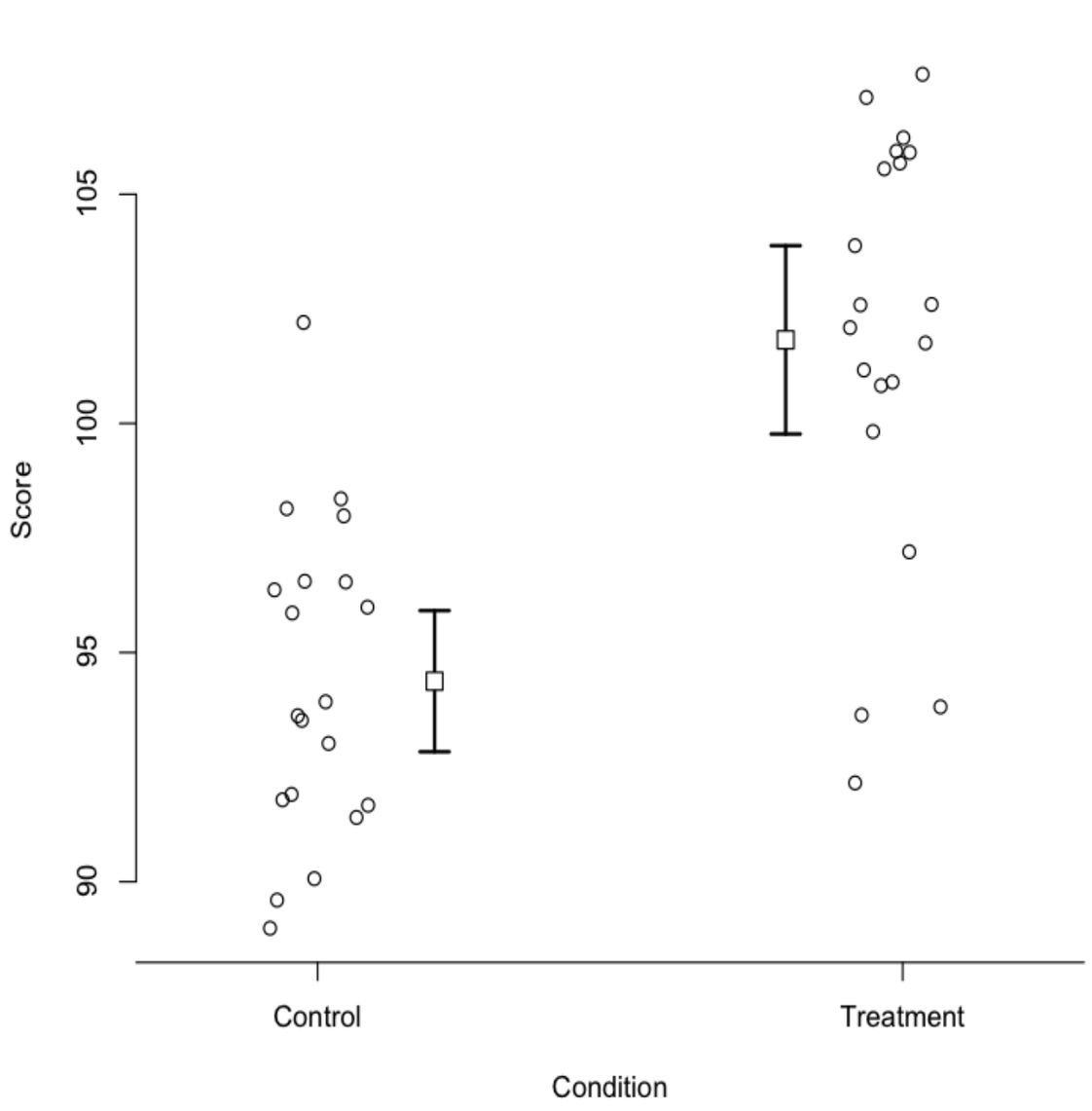

102
100
98
96
94
92
90
1
2
Series1

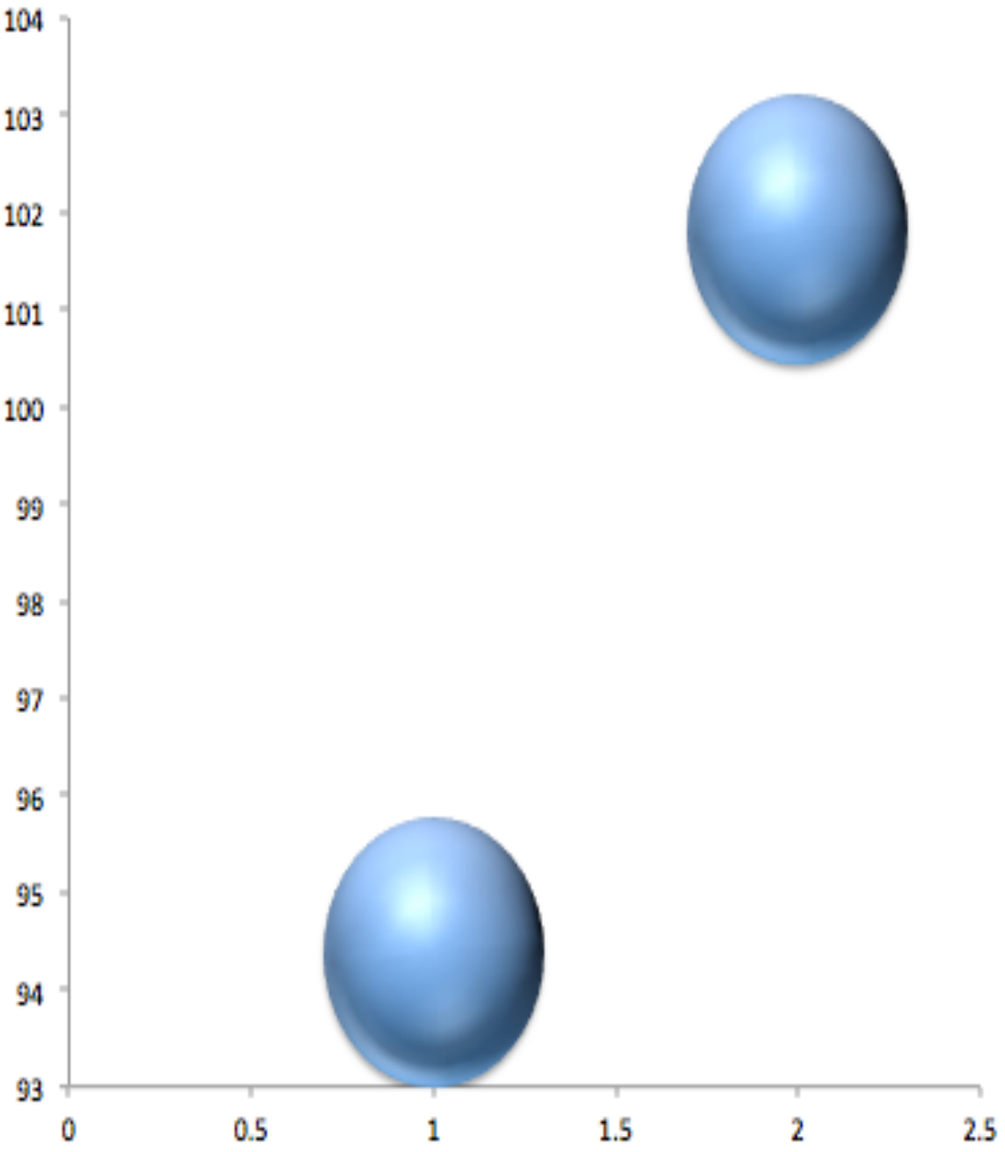
Examples


Score
105
100
95
90
Control
Treatment
Condition

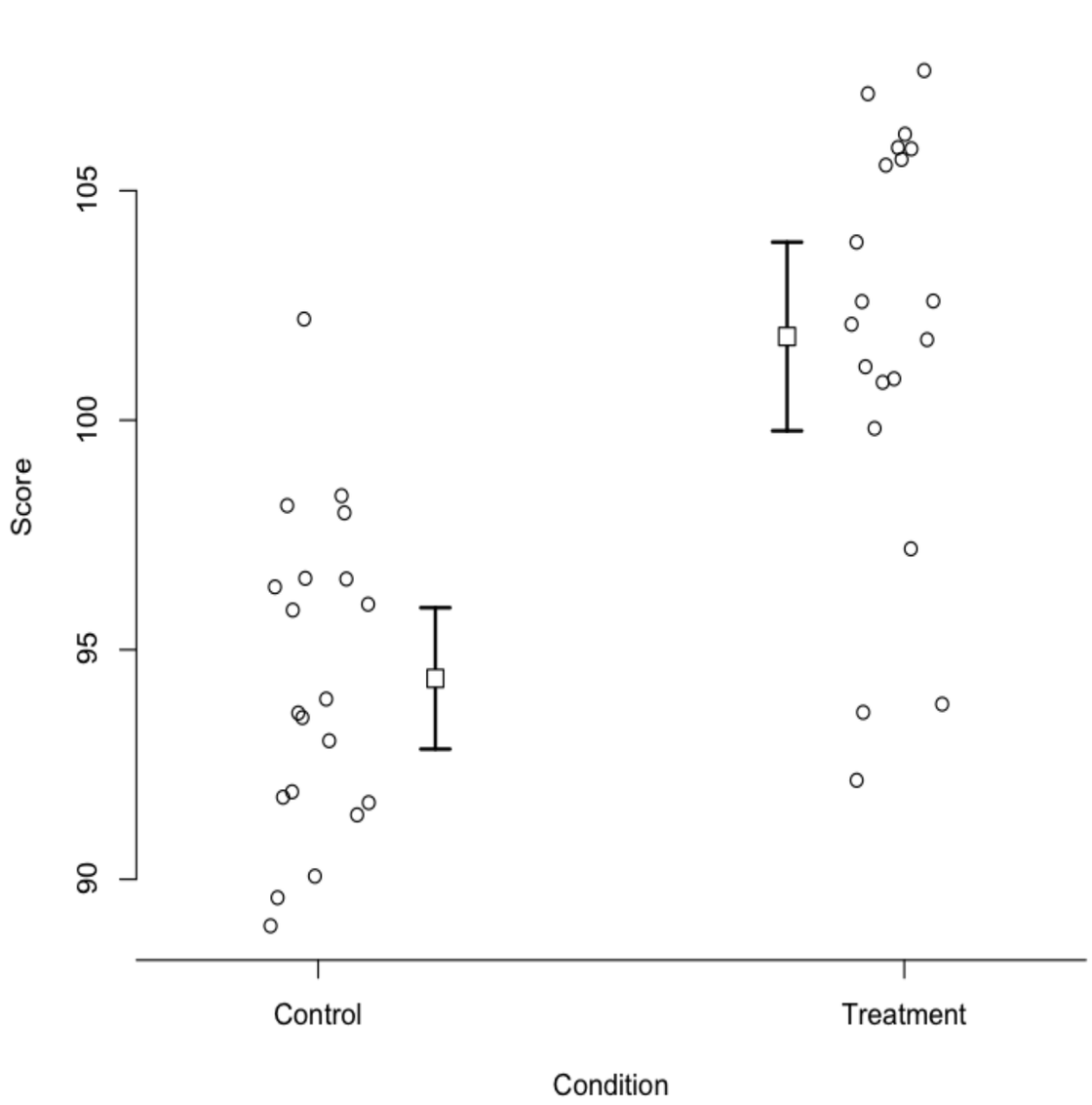

102
100
98
96
94
92
90
1
2
Series1

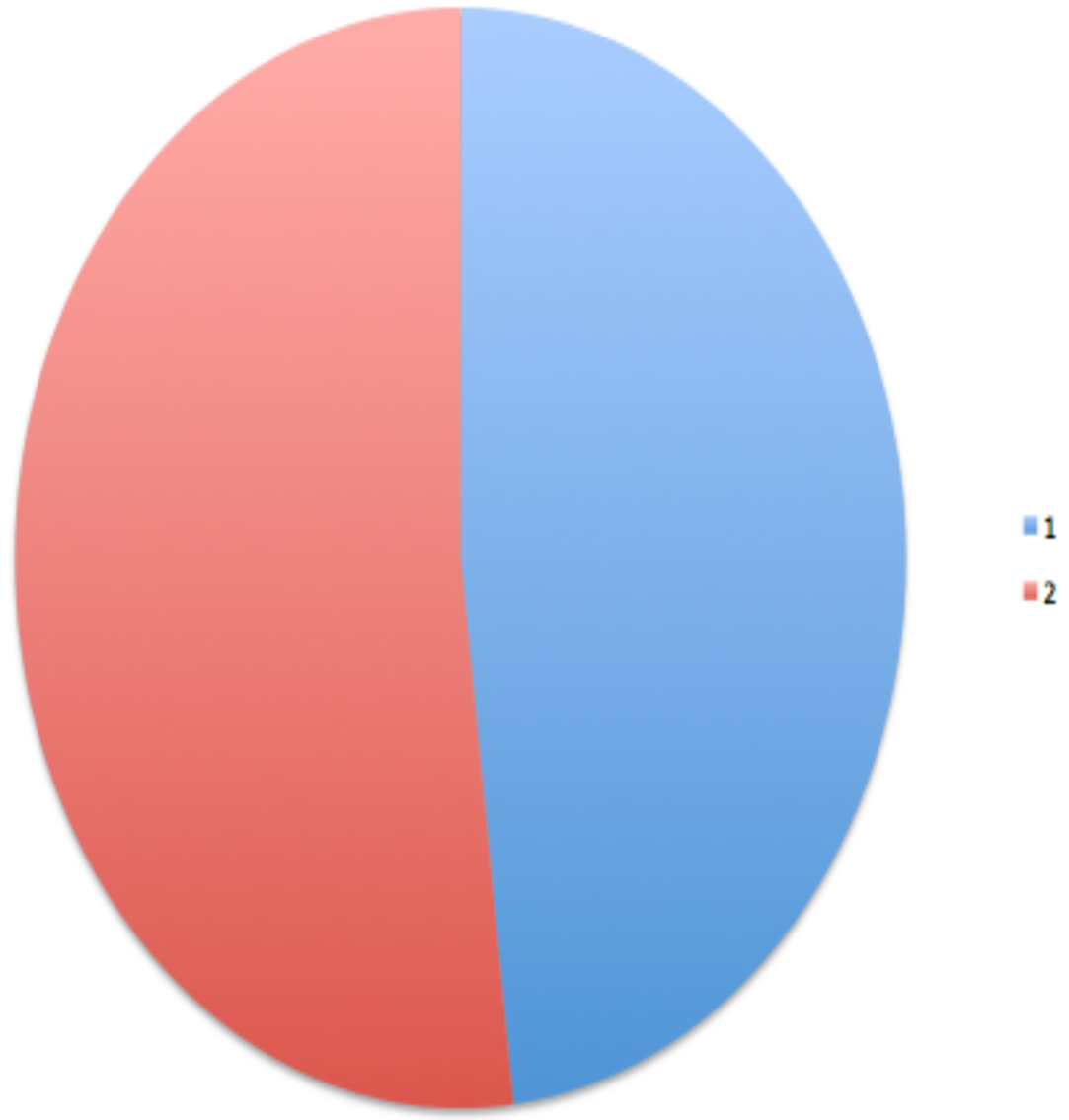
Examples


Score
105
100
95
90
Control
Treatment
Condition

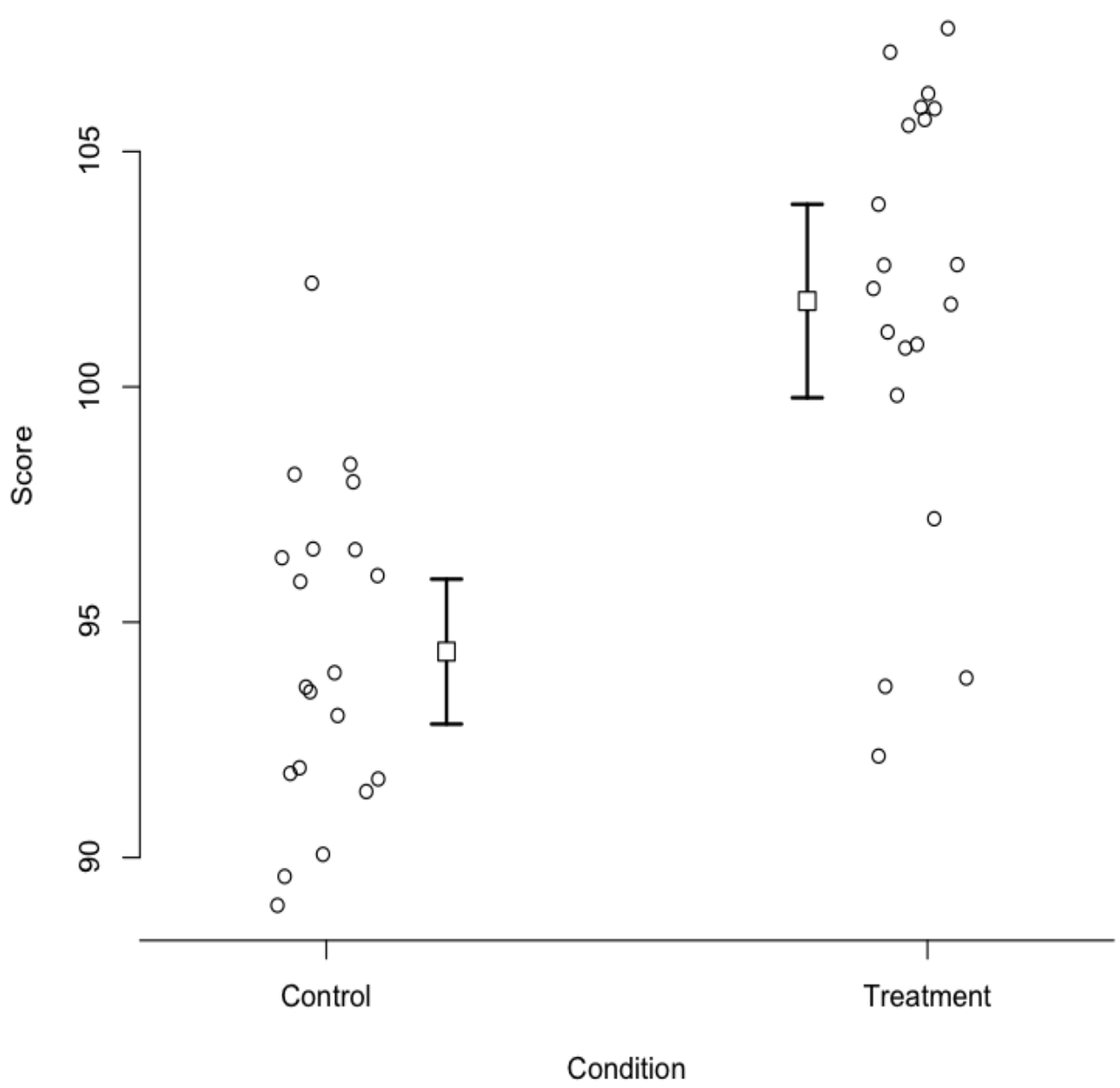

104
103
102
101
100
99
98
97
96
95
94
93
0
0.5
1
1.5
2
2.5
Series1

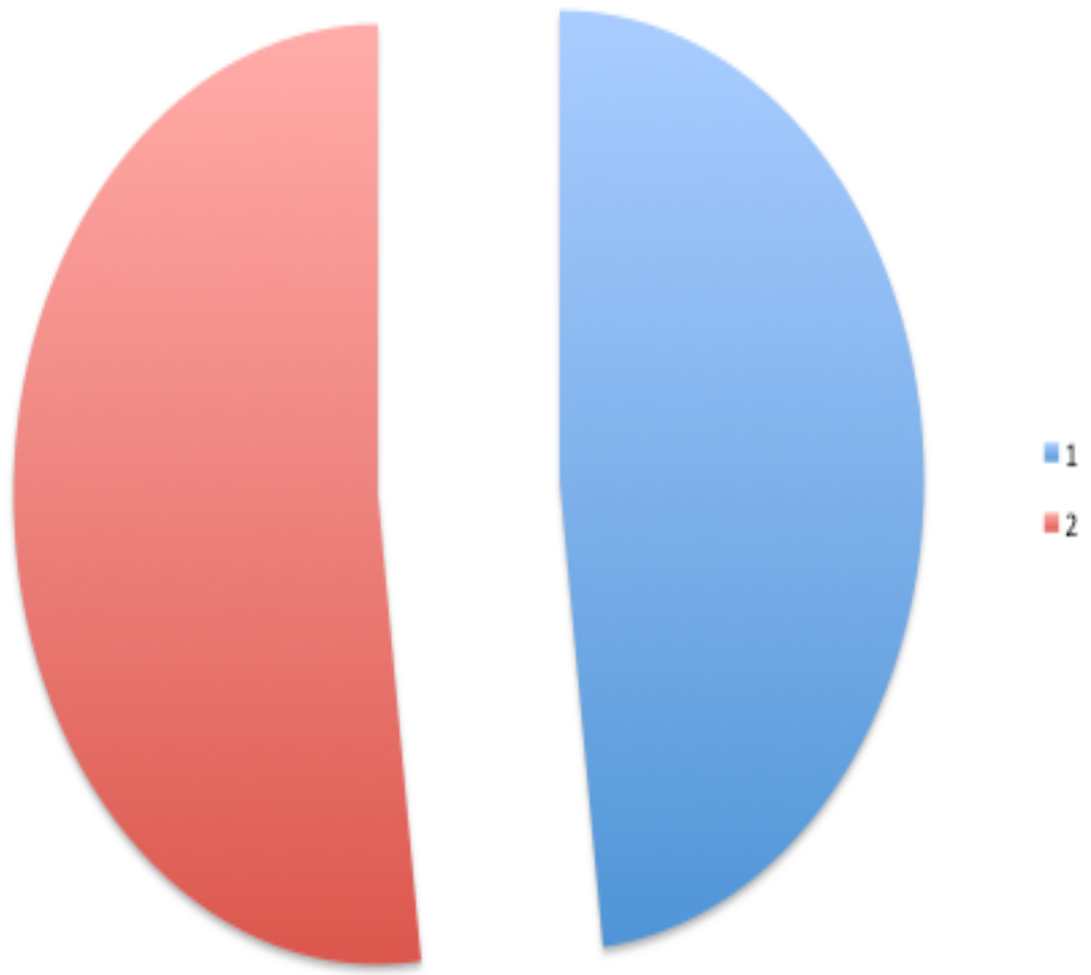
Examples



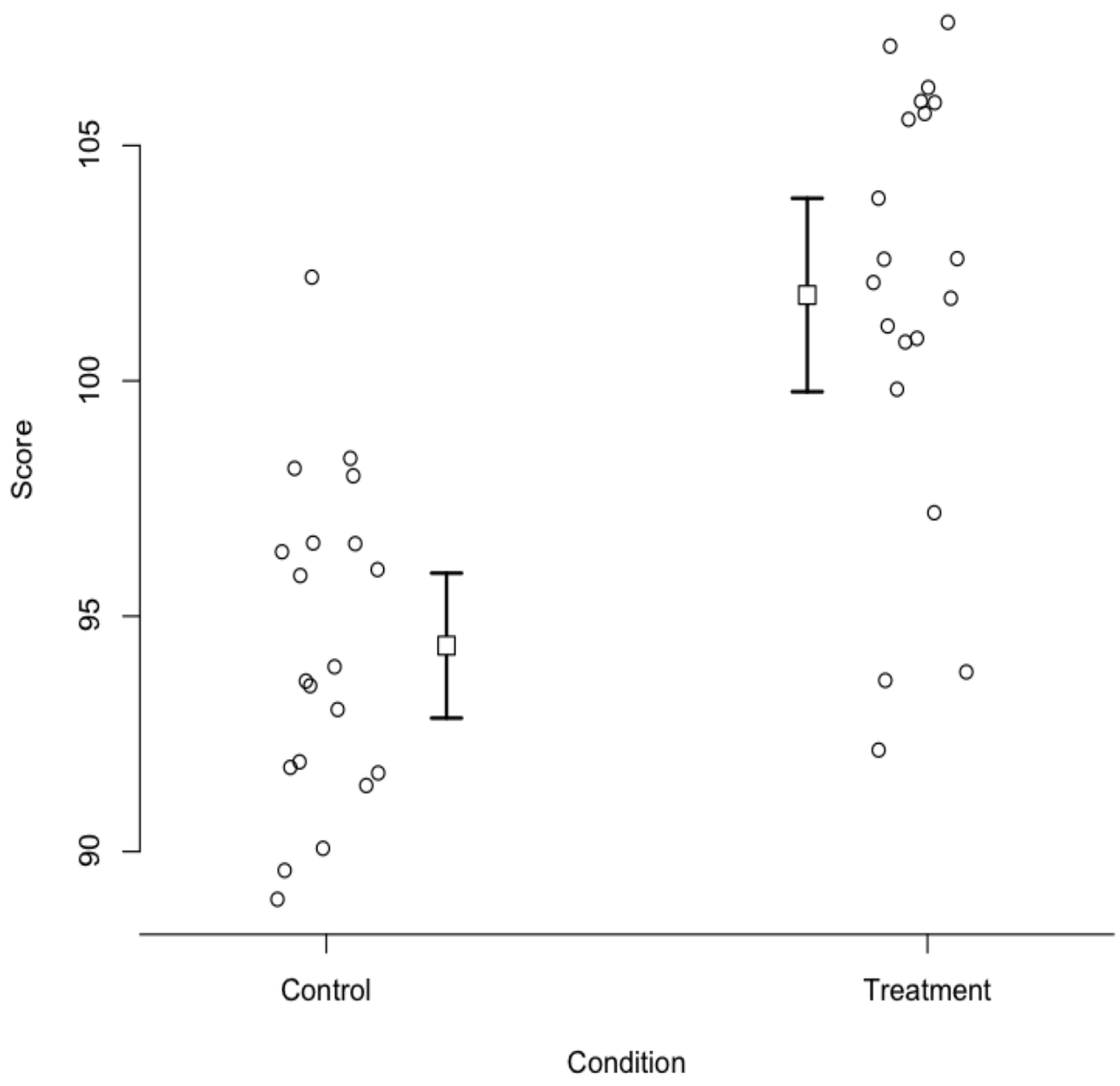


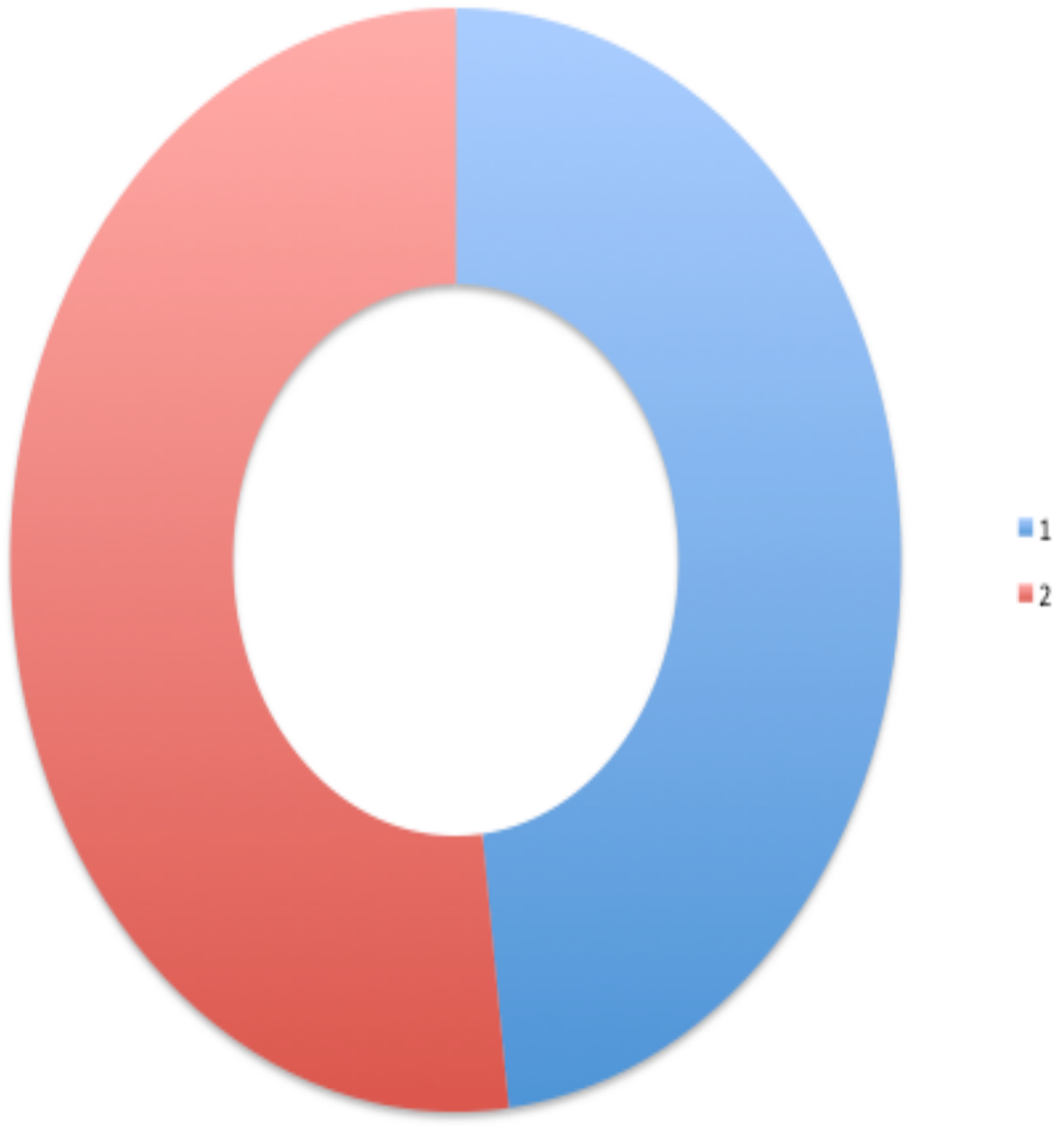
Examples



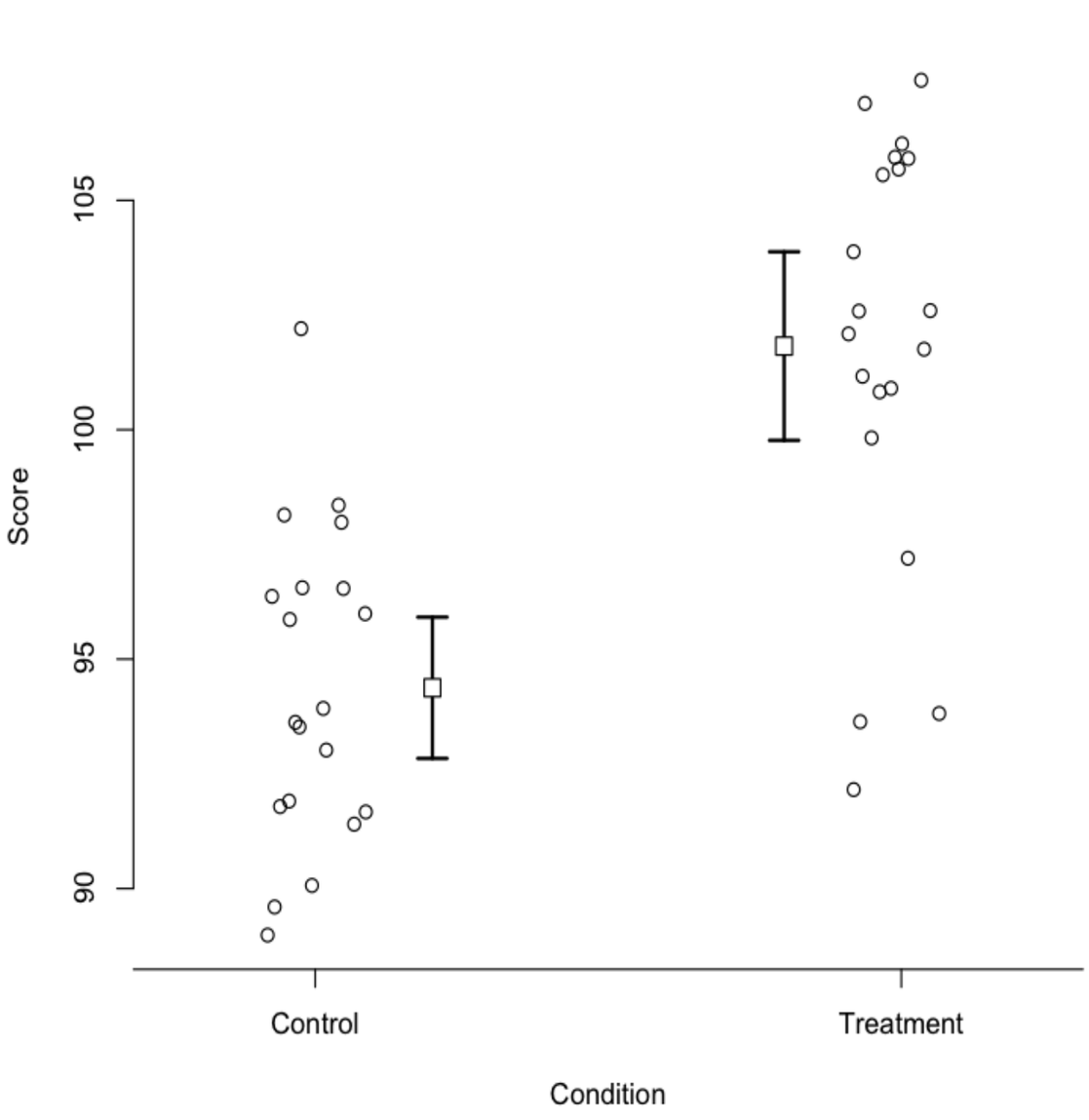


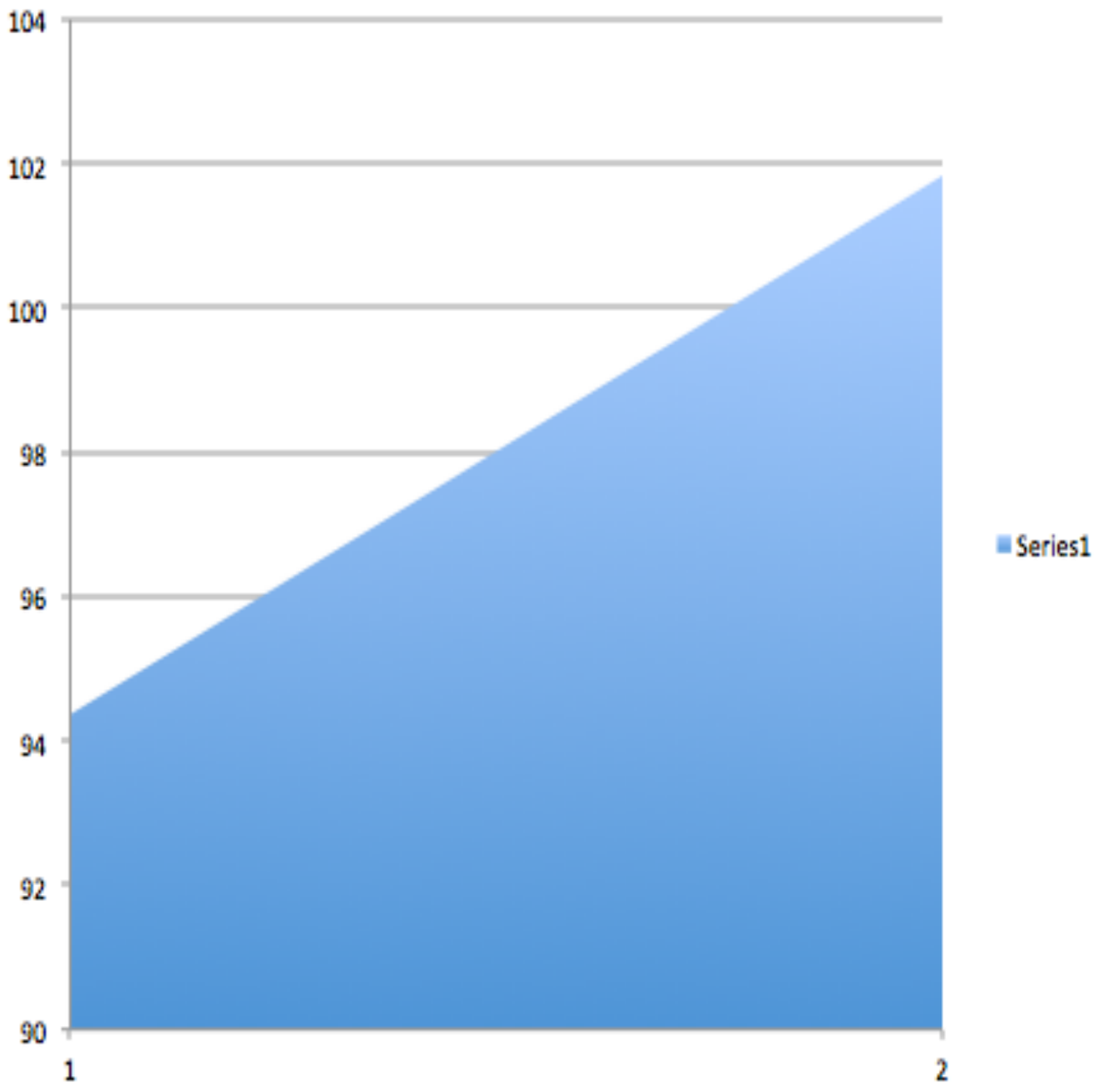
Examples



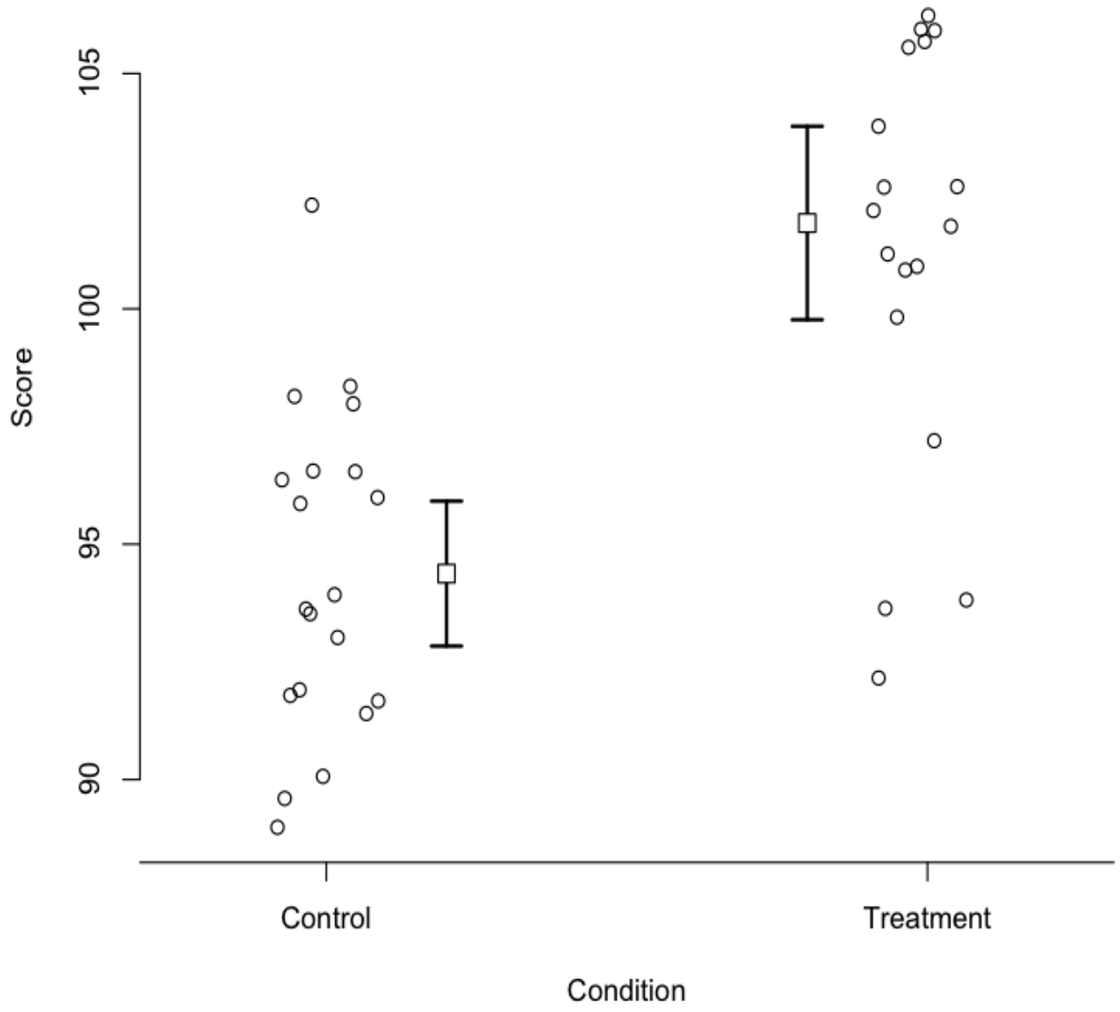


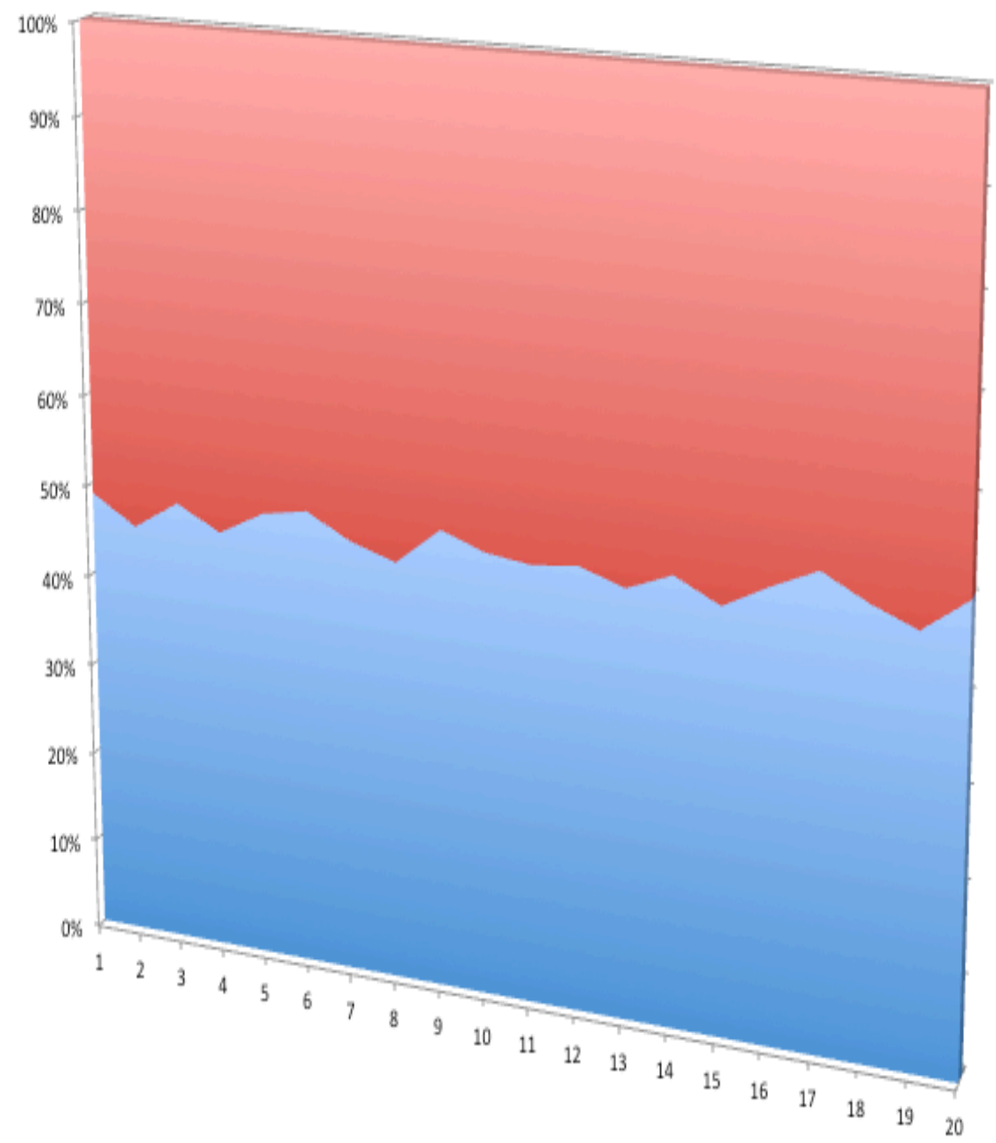
Examples


Score
105
100
95
90
Control
Treatment
Condition

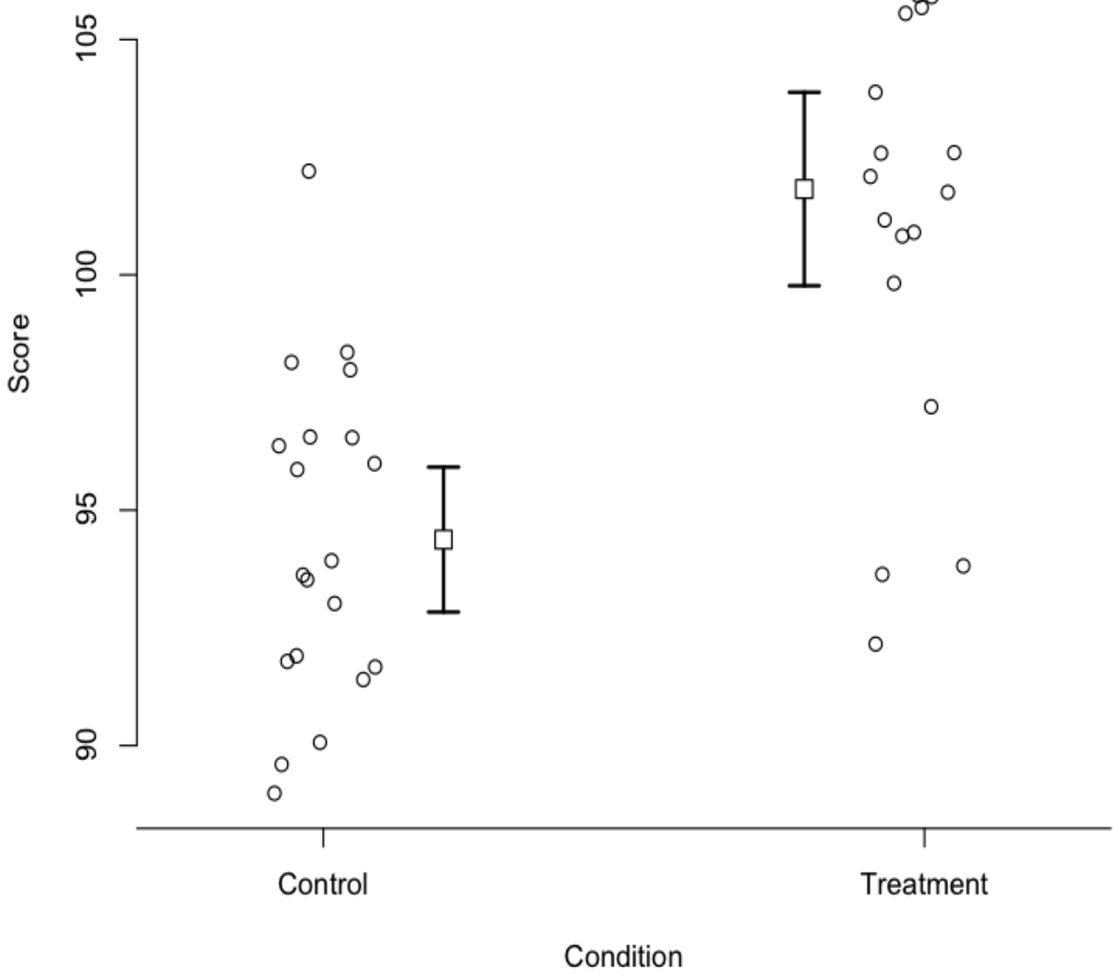

104
102
100
98
96
94
92
90
1
2
Series1

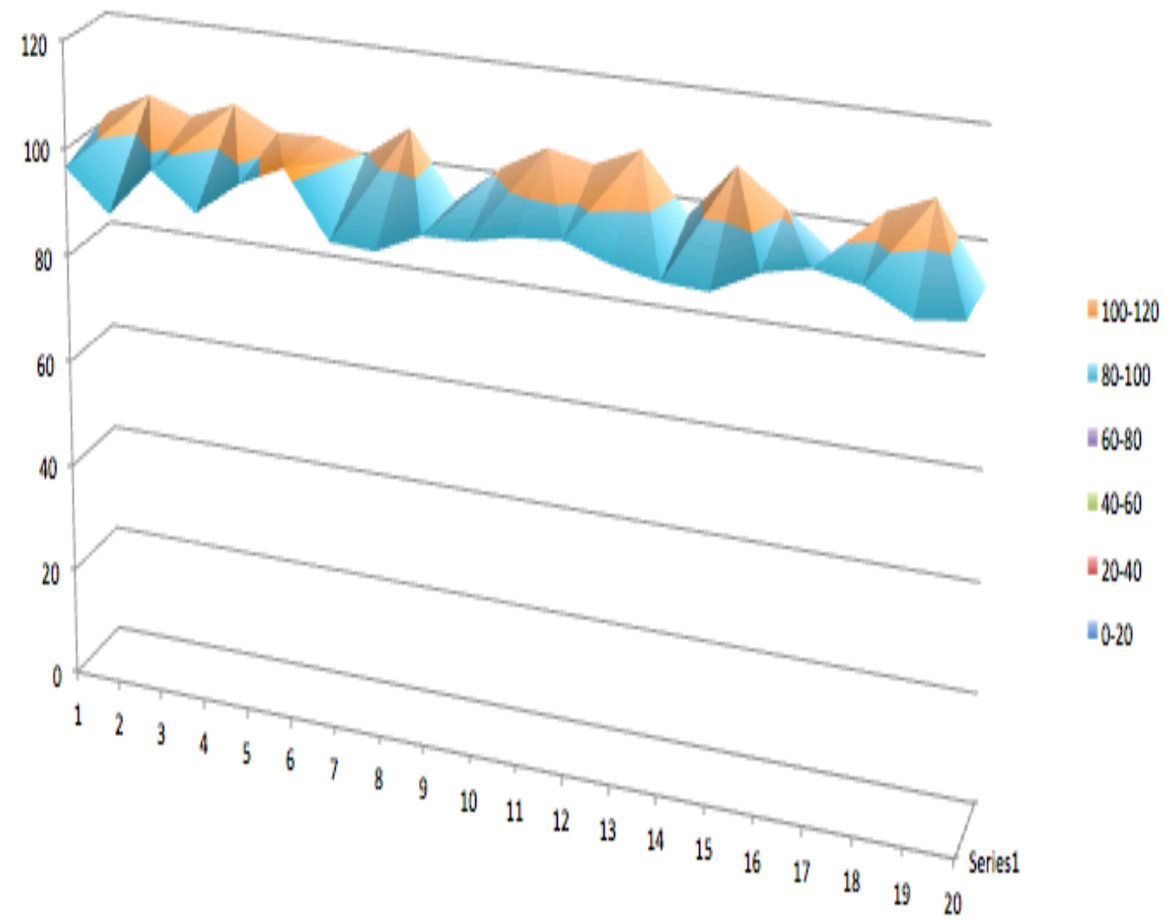
Examples


Score
105
100
95
90
Control
Treatment
Condition


100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Series2
Series1

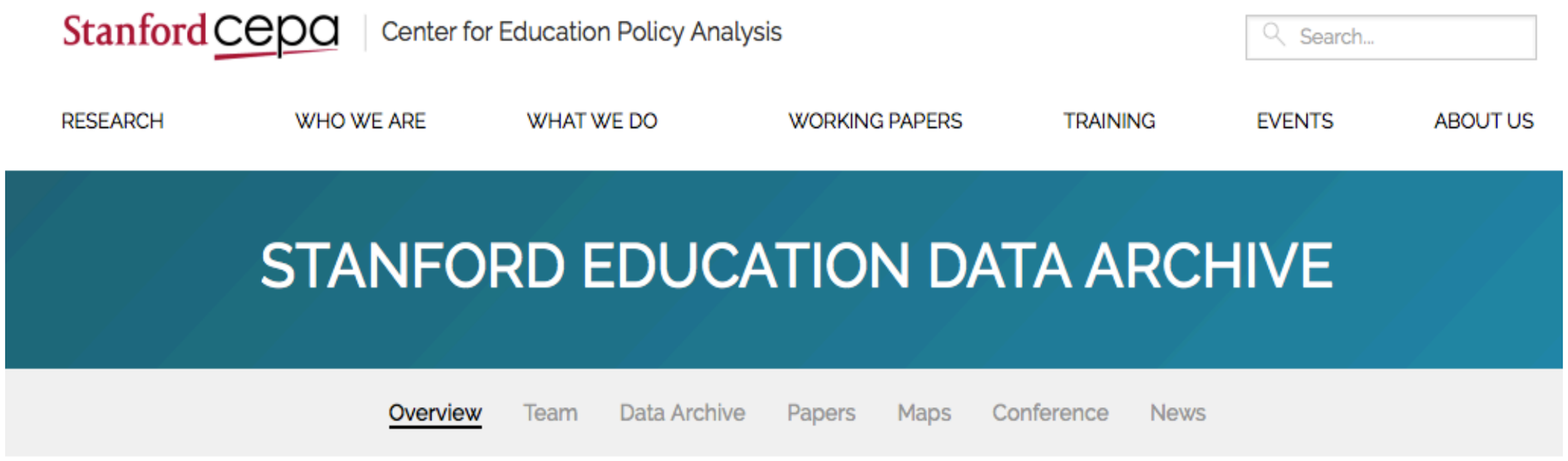
Examples


Score
105
100
95
90
Control
Treatment
Condition


120
100
80
60
40
20
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Series1
100-120
80-100
60-80
40-60
20-40
0-20

Some great examples: SEDA

Sean Reardon: https://cepa.stanford.edu/seda/overview

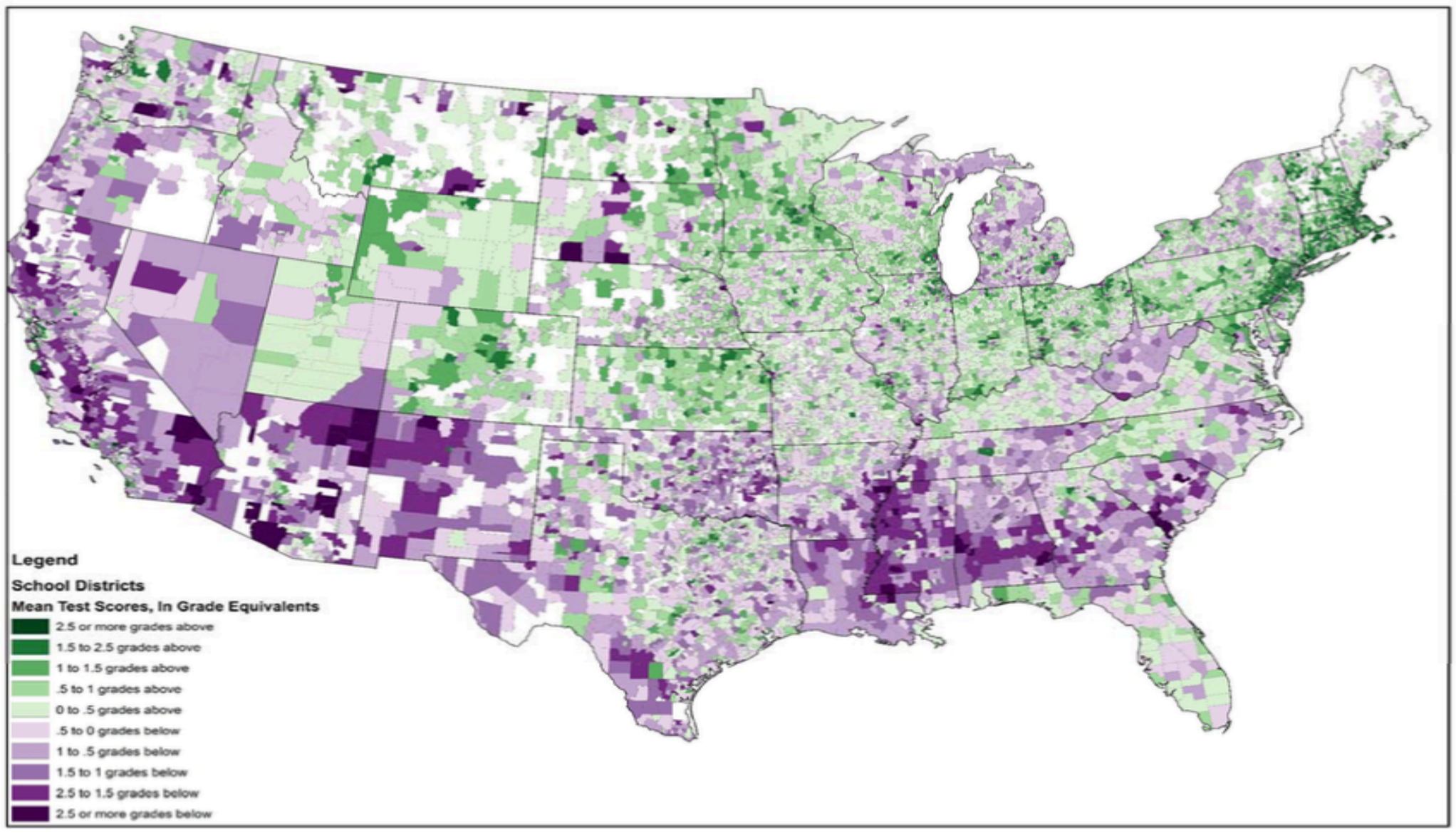


Racial, socioeconomic, and gender disparities in academic performance and educational attainment are stubborn features of the U.S. educational system. These disparities are neither inevitable nor immutable, however. They have been produced by—and so may also be reduced by—a welter of social and economic policies, social norms and patterns of interaction, and the organization of American schooling.

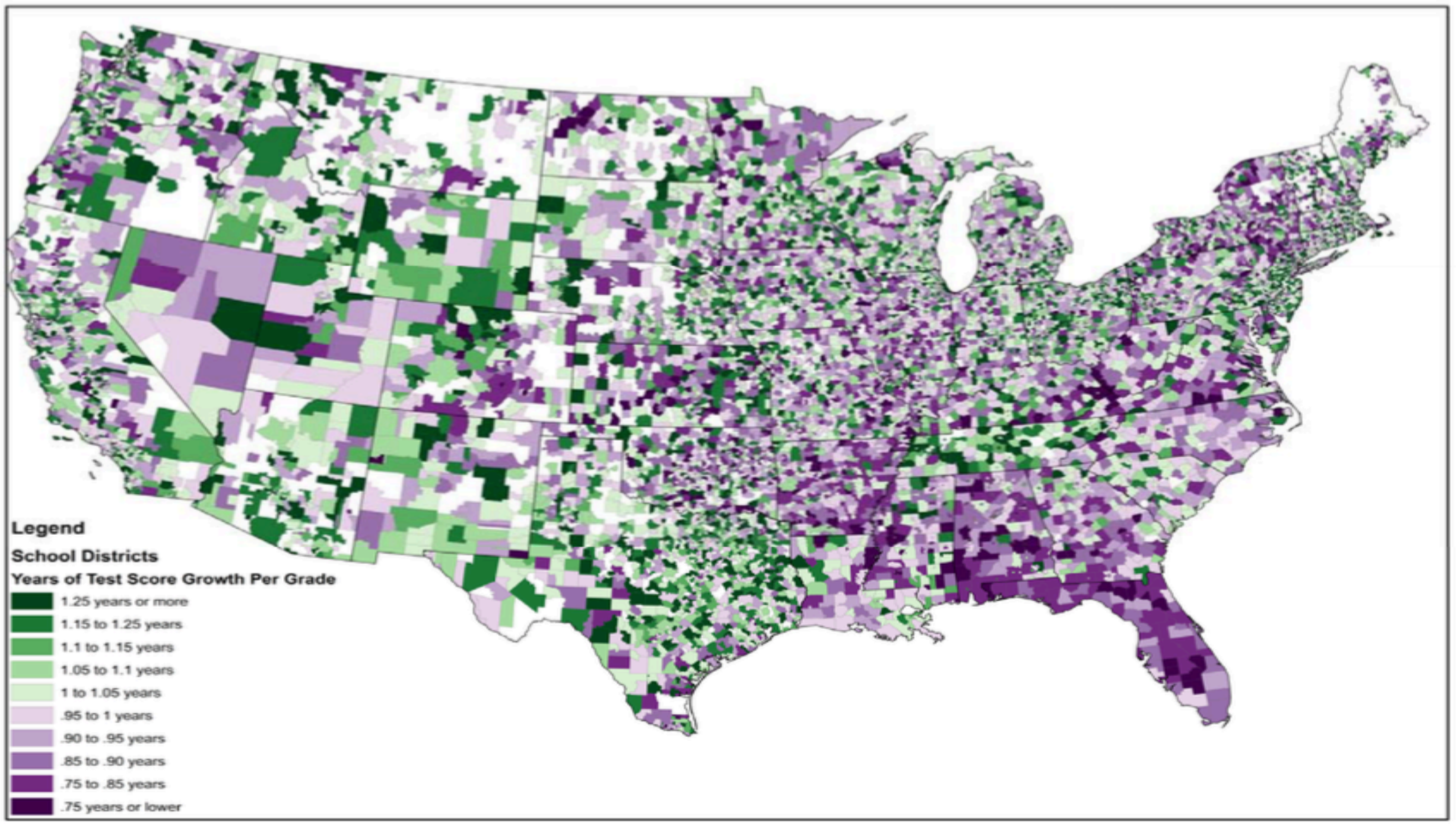
The Stanford Education Data Archive (SEDA) is an initiative aimed at harnessing data to help us—scholars, policymakers, educators, parents—learn how to improve educational opportunity for all children. We are making the data files public so that anyone who is interested can obtain detailed information about

Sean F. Reardon (Stanford University). *The Landscape of U.S. Educational Inequality.* Download presentation

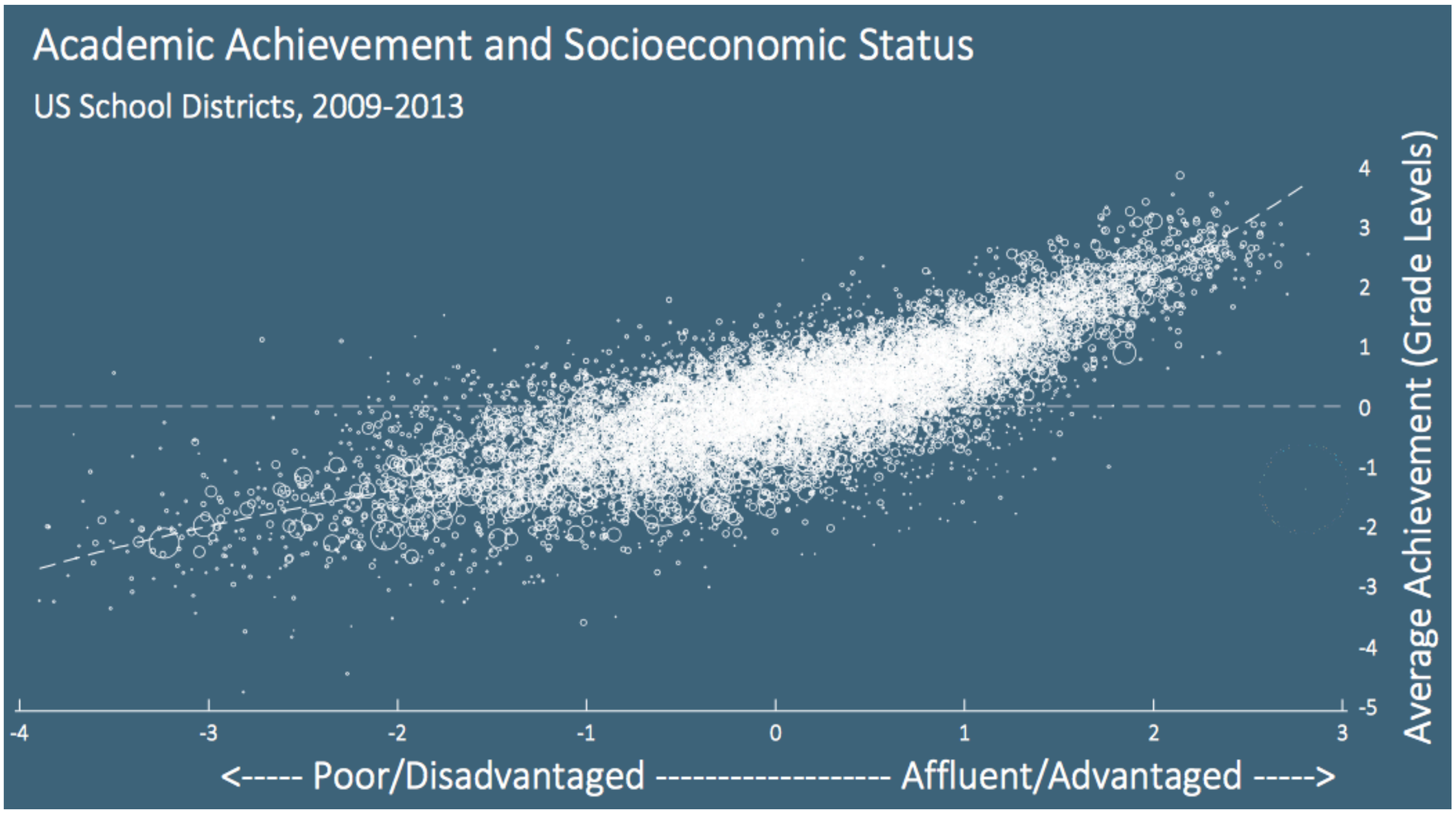
Means by district



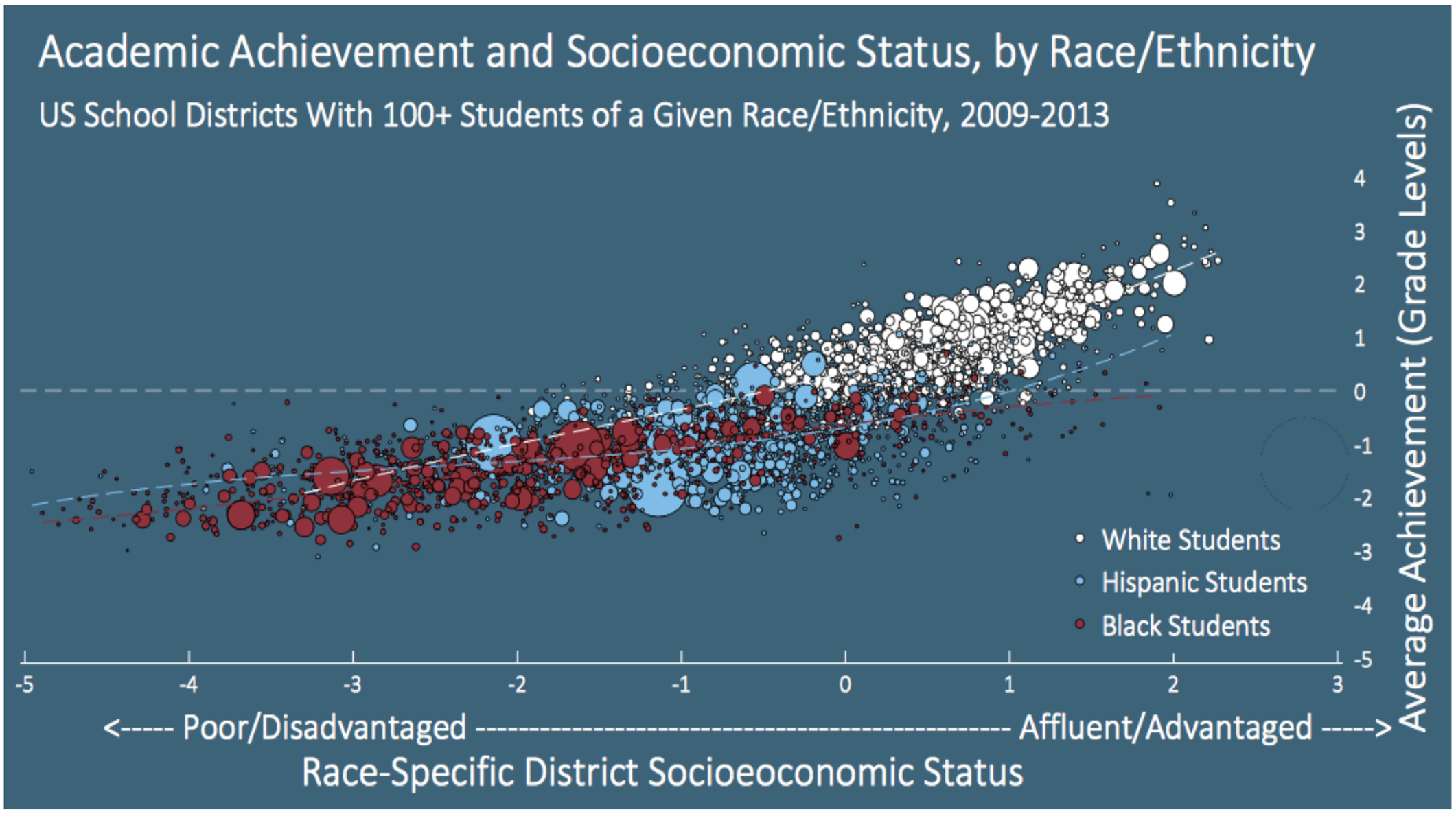
Average gains by district



Mean scores and SES



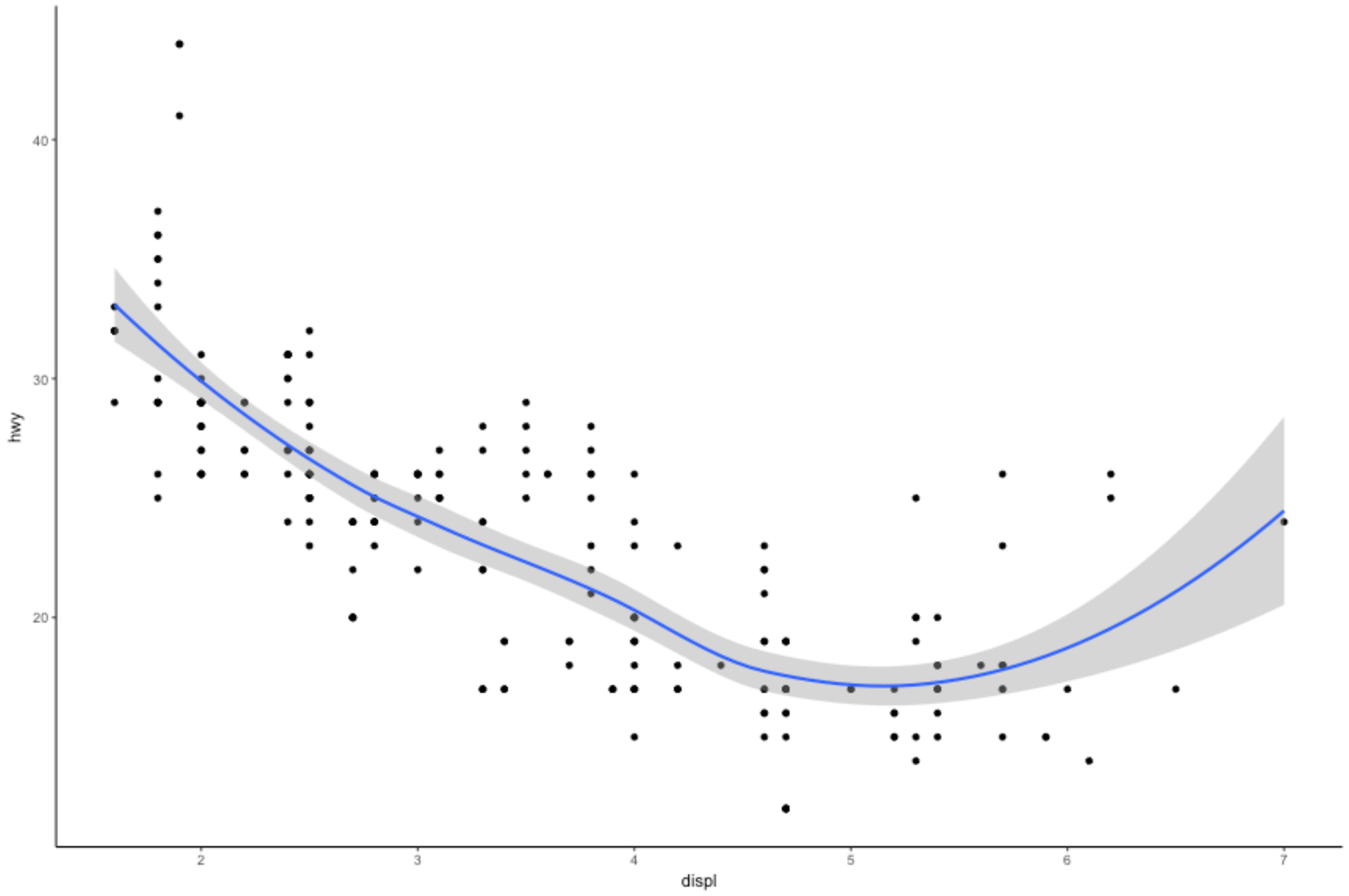
Mean scores and SES by Race/Ethniciy



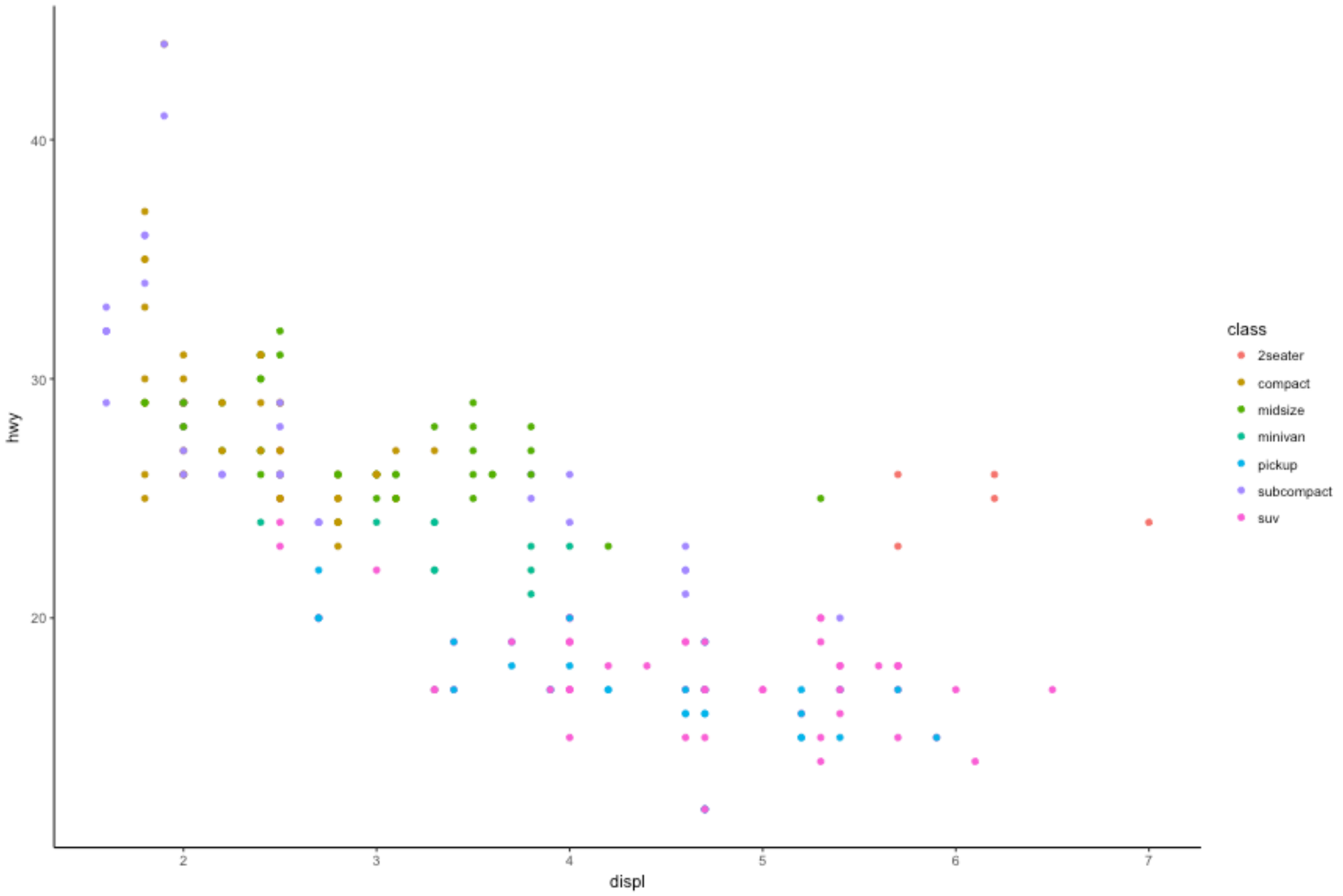
Other examples: Visualizing scale

- Space stuff: http://imgur.com/a/lGabv
- Time: http://www.sciencealert.com/watch-this-3-minute-animation-will-change-your-perception-of-time

Some *ggplot* examples

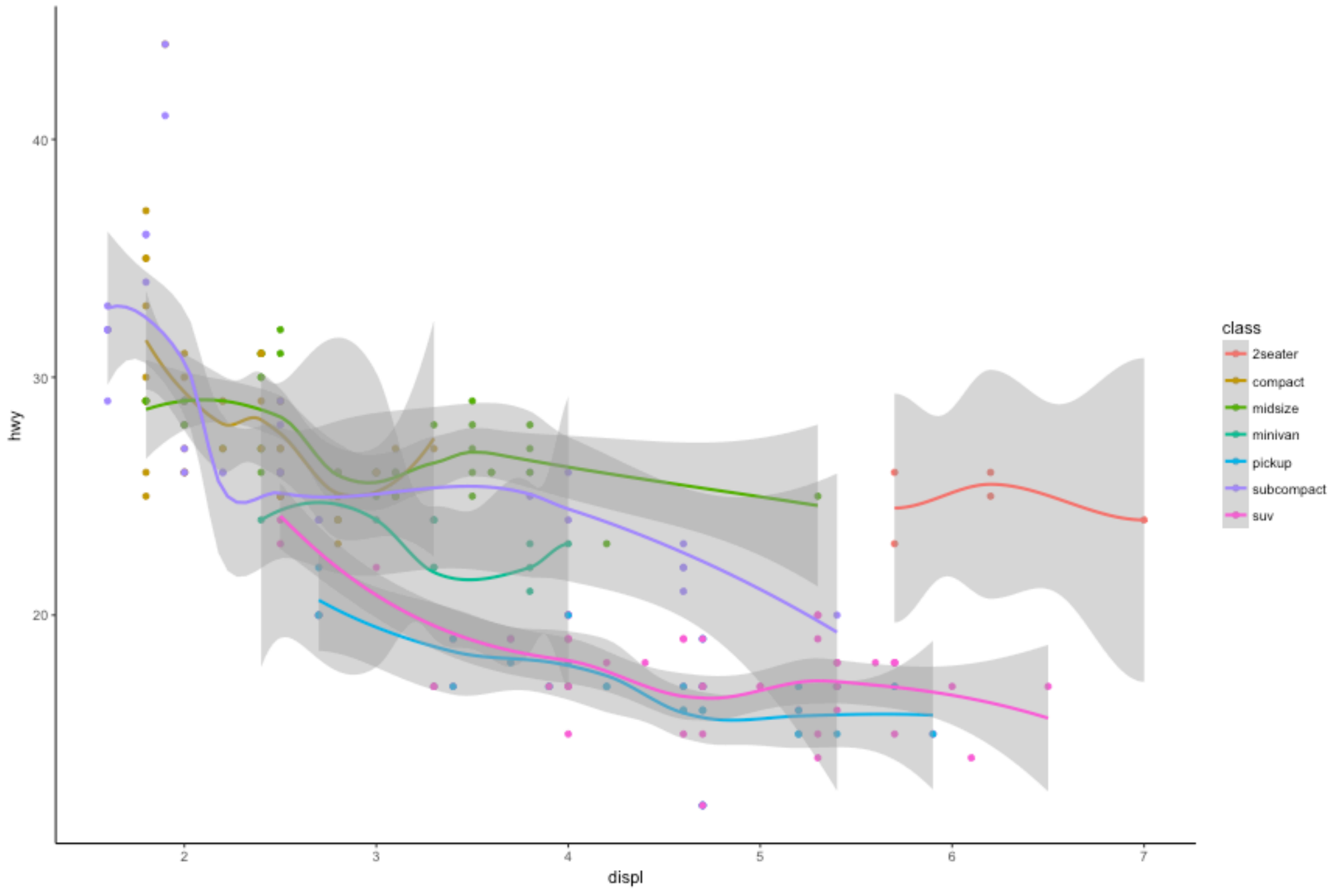


Add an additional aesthetic

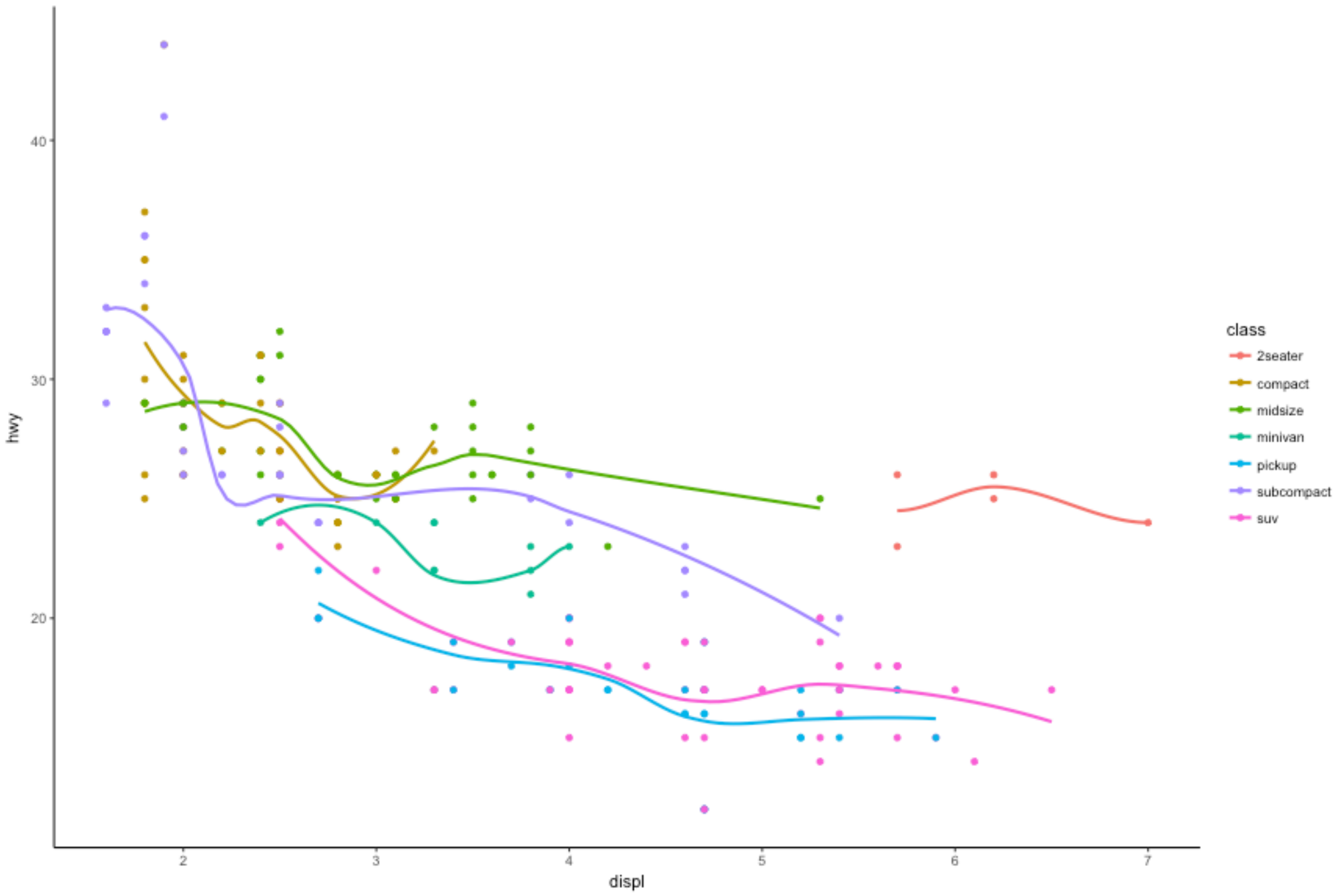


Add smooth line for each class

Too busy

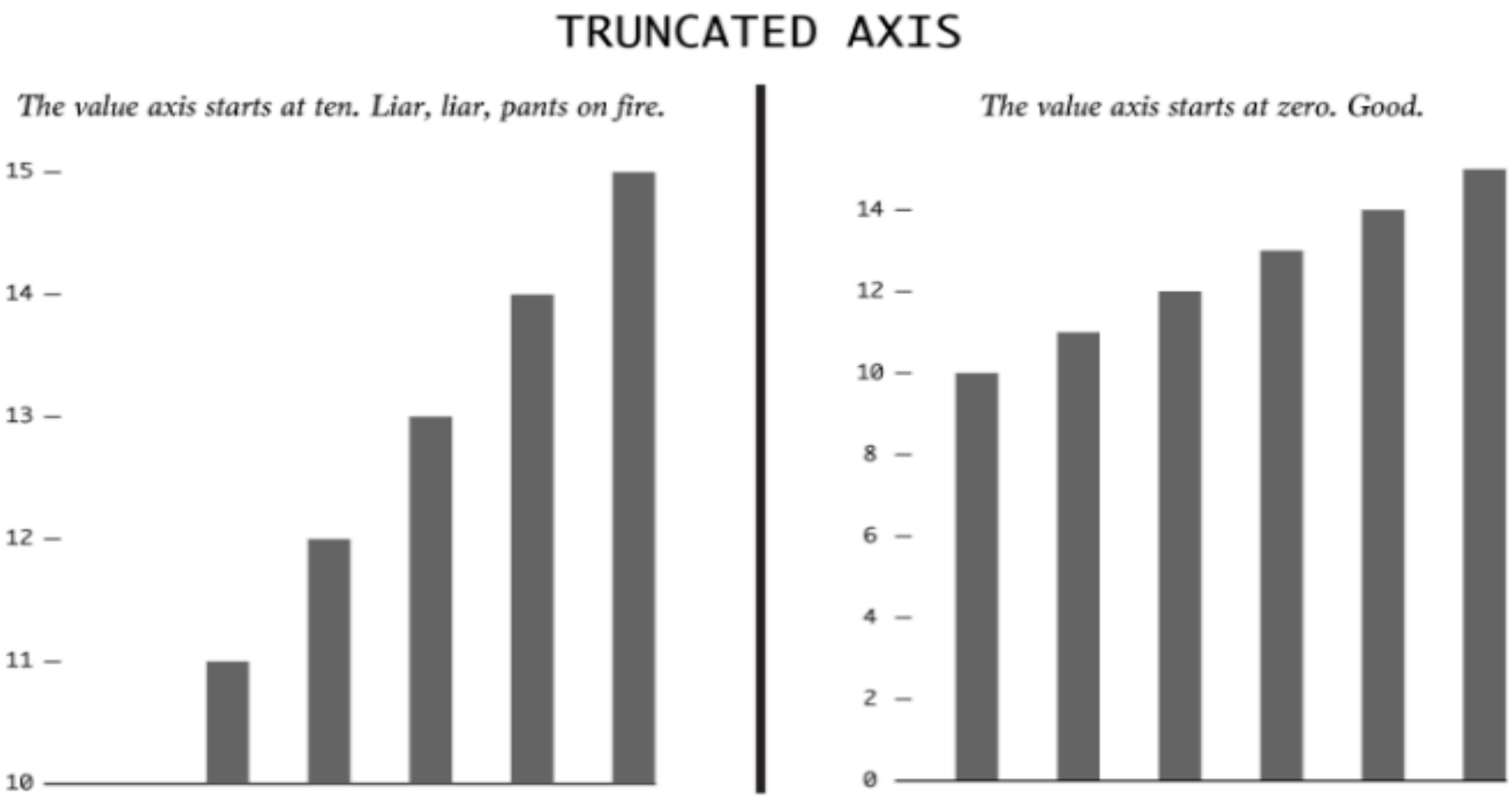


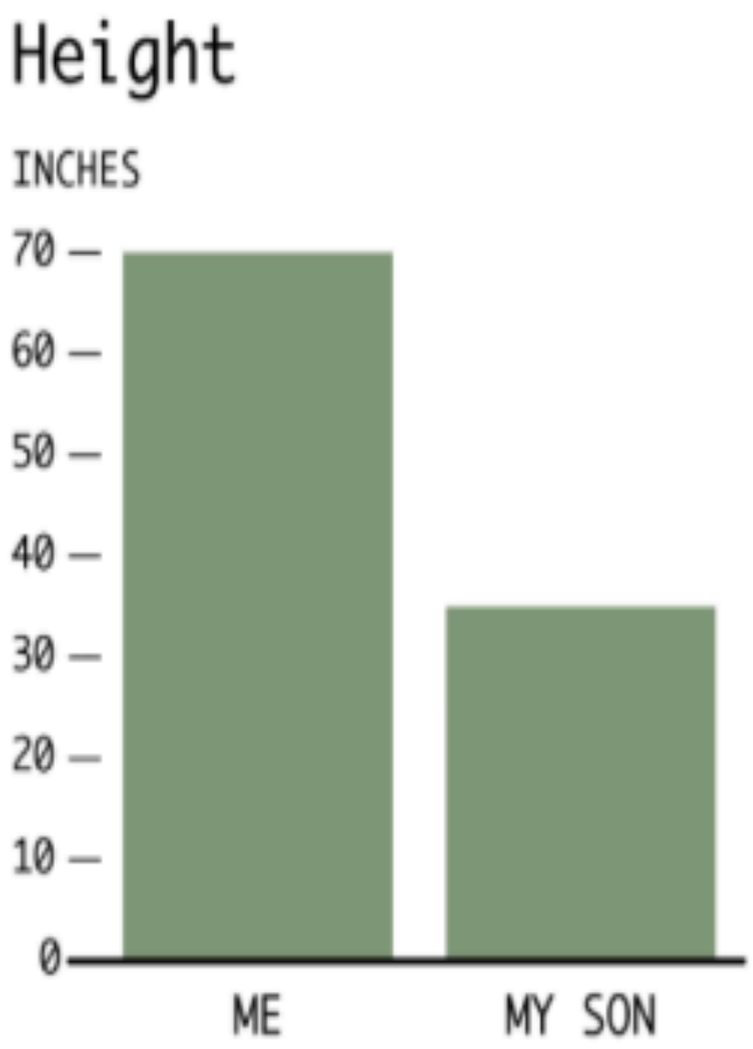
Remove SE


hwy
40
30
20
2
3
4
5
6
7
displ
class
2seater
compact
midsize
minivan
pickup
subcompact
suv

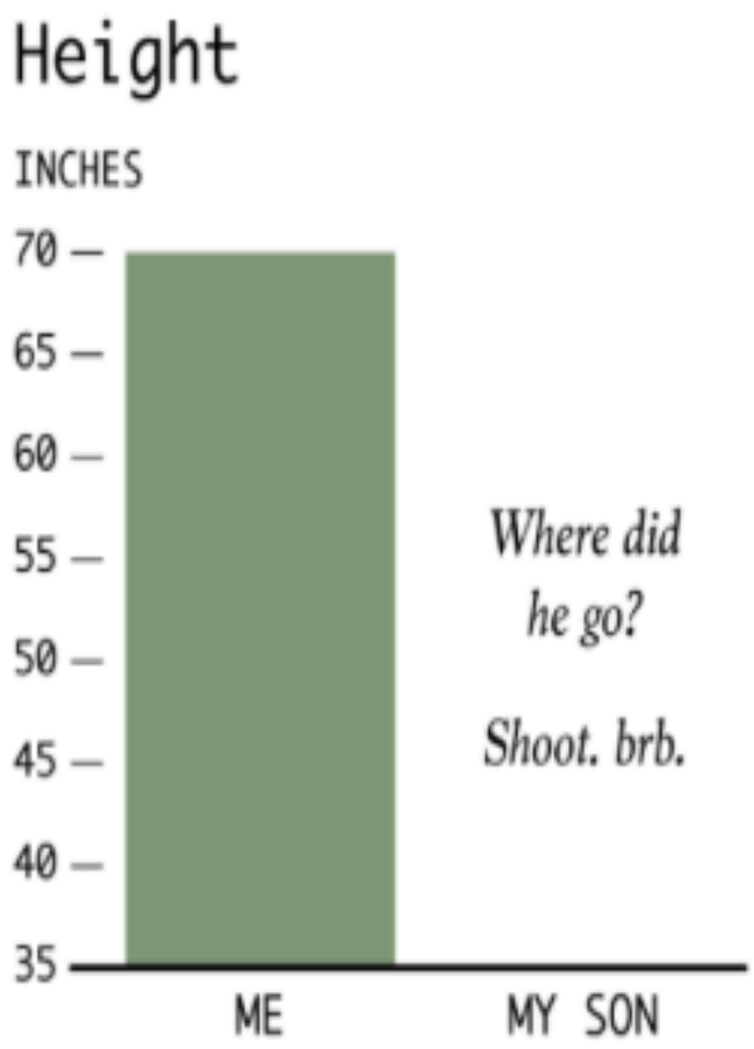
Some things to avoid

Truncated axes





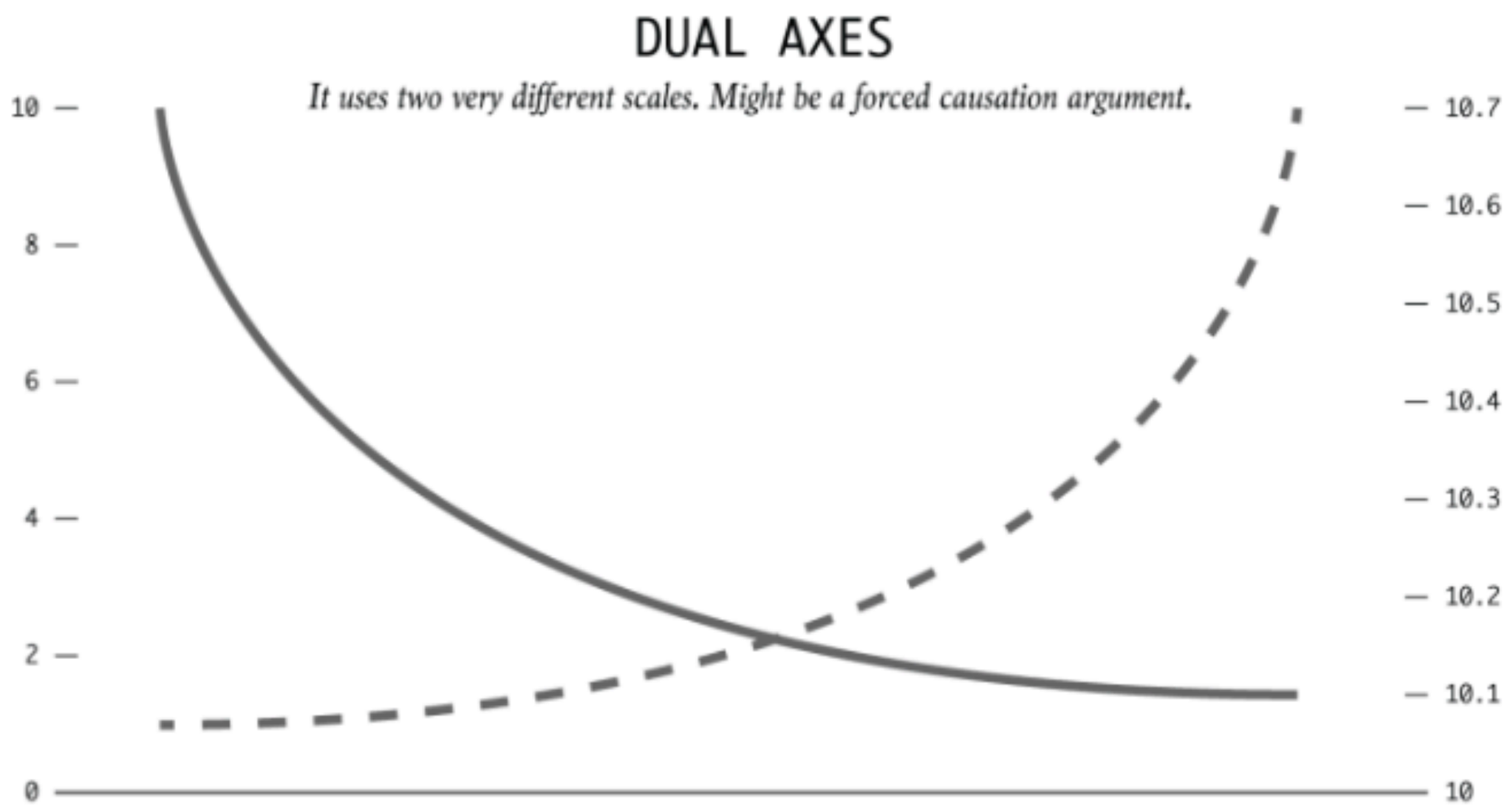
vs.



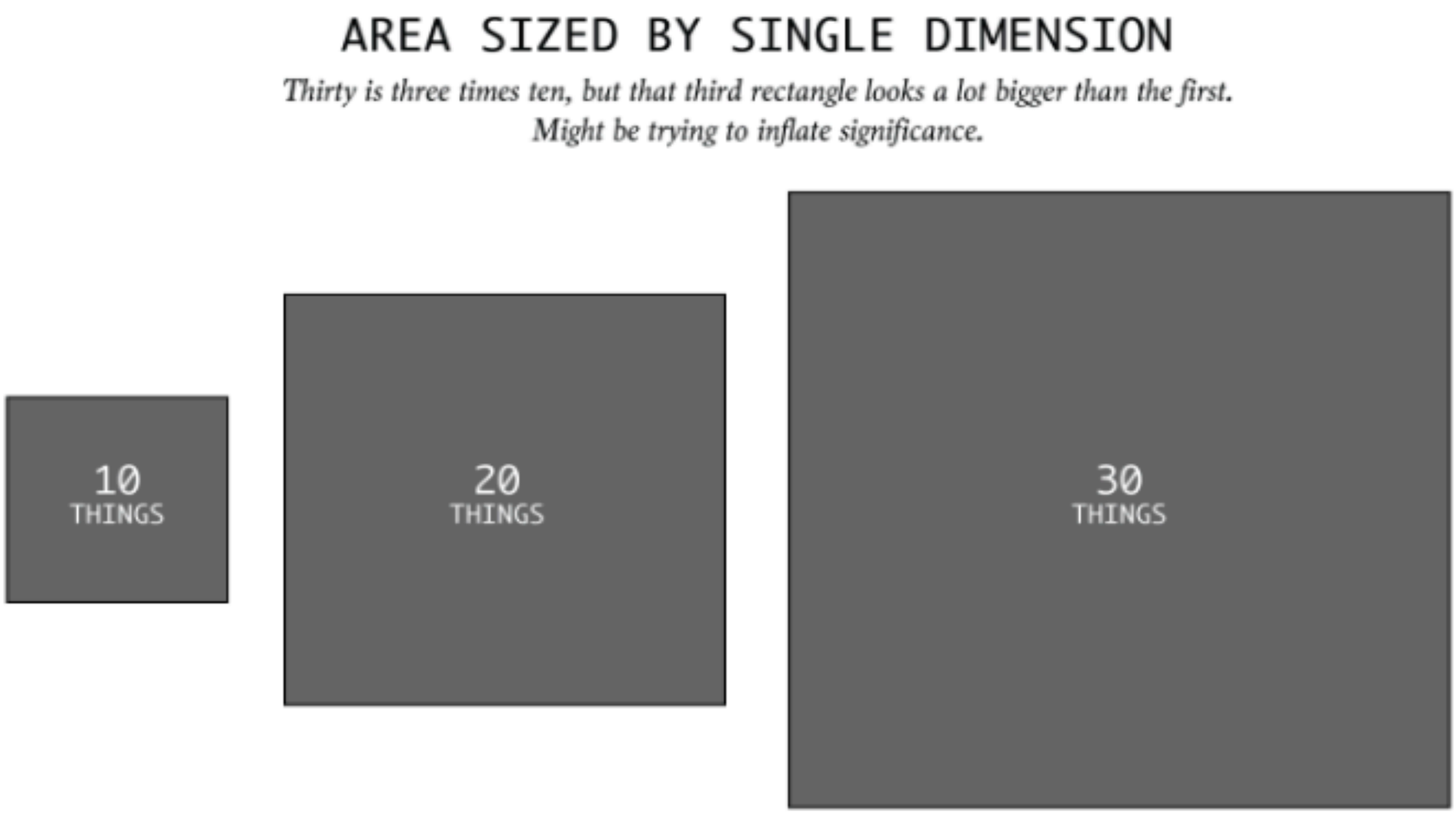
Where did he go?

Shoot. brb.

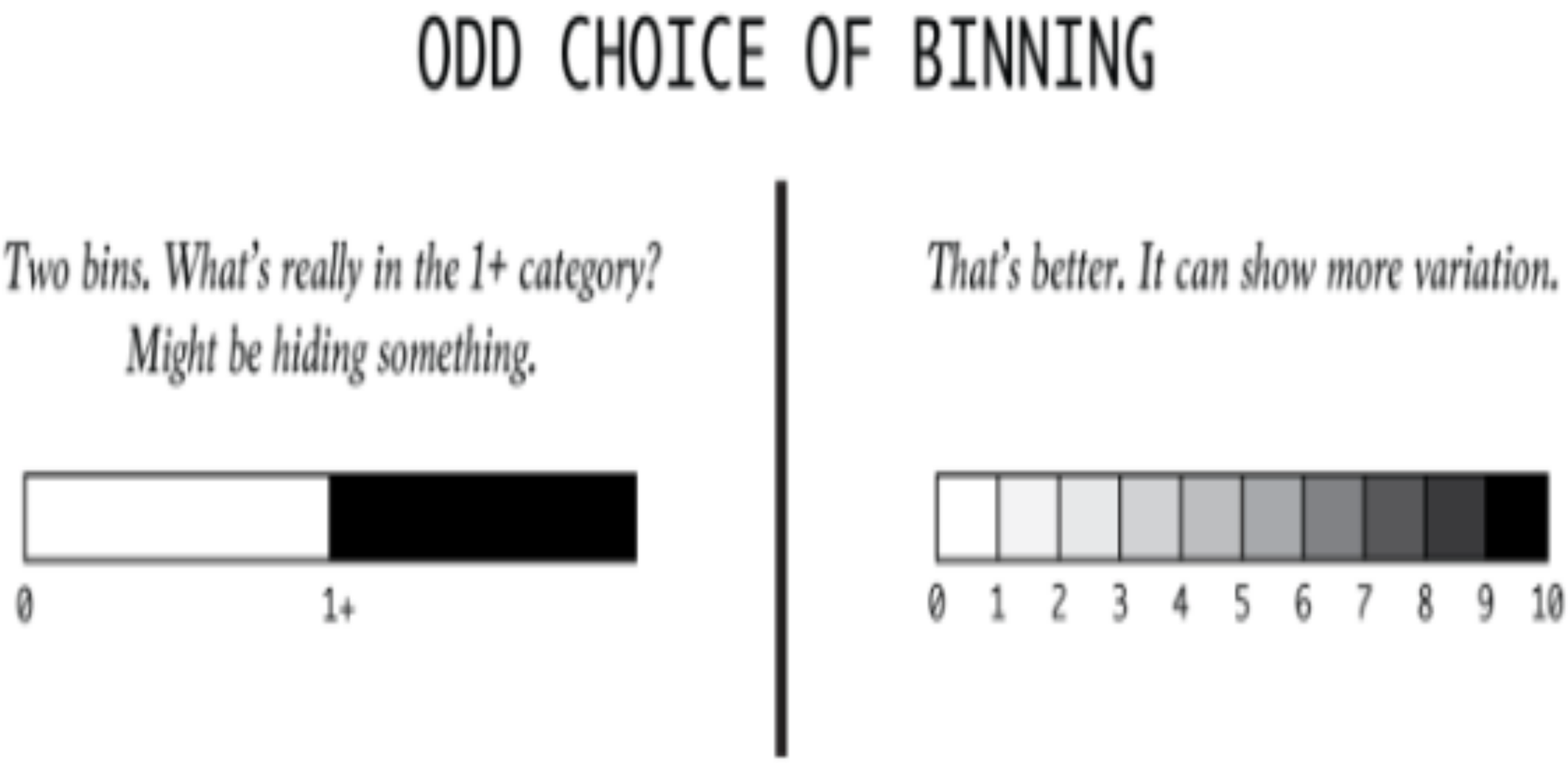
Dual axes


DUAL AXES
It uses two very different scales. Might be a forced causation argument.
10
8
6
4
2
0
10.7
10.6
10.5
10.4
10.3
10.2
10.1
10

Scaling issues



Poor binning choices

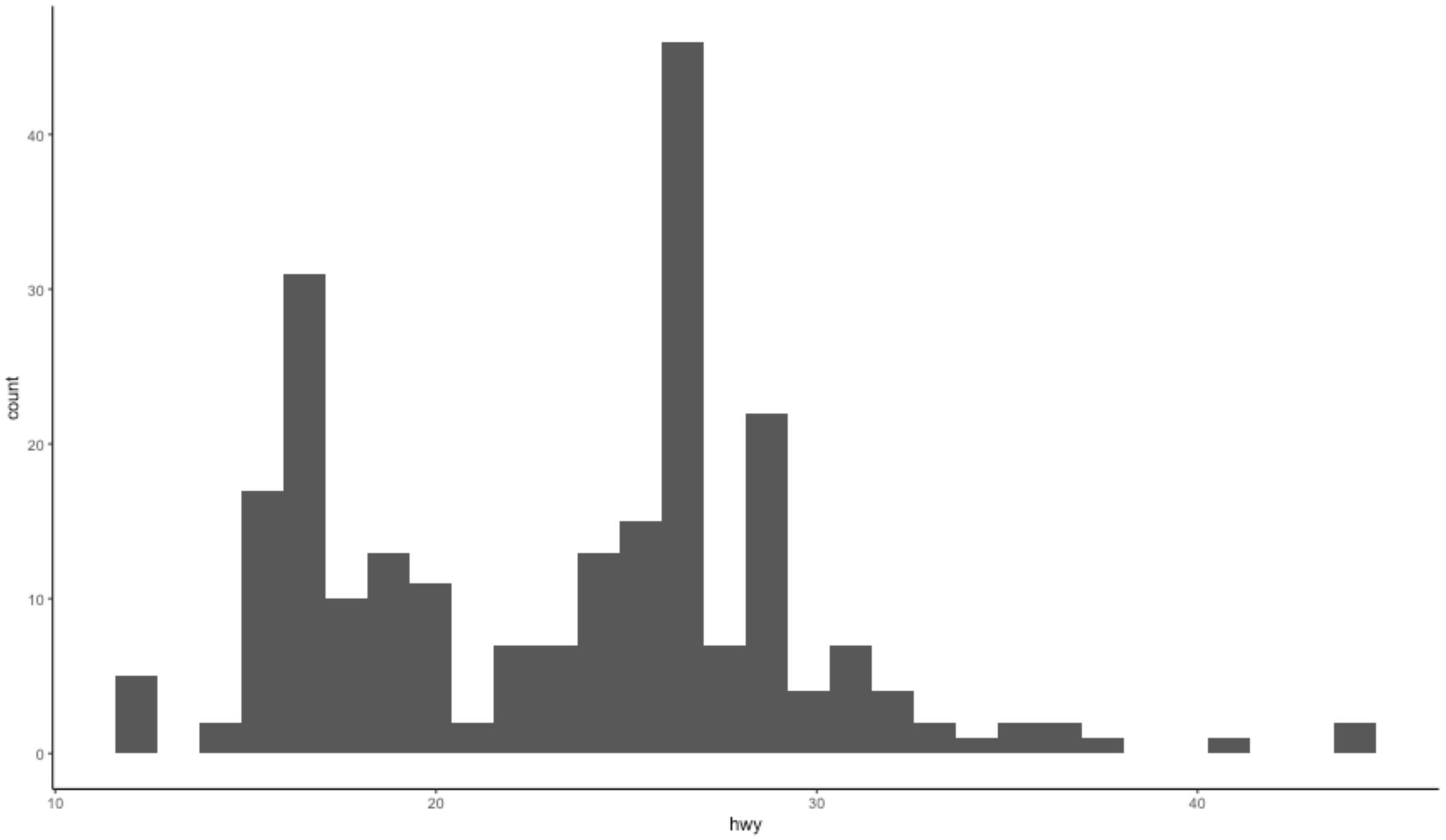


Some general advice

- Consider the purpose of the plot.
    - Relation? Scatterplots
    - Distribution? Histogram or density plot
    - Trend? Line plot, scatterplot with smoother, etc.
- How many variables? What type?
    - One continuous variable: histogram, density plot, or similar
    - Two continuous: Scatterplot (if you have lots of data, consider binning)
    - One categorical one continuous: boxplots, violin plots, bar plots
    - Two categorical variable? Mosaic plot

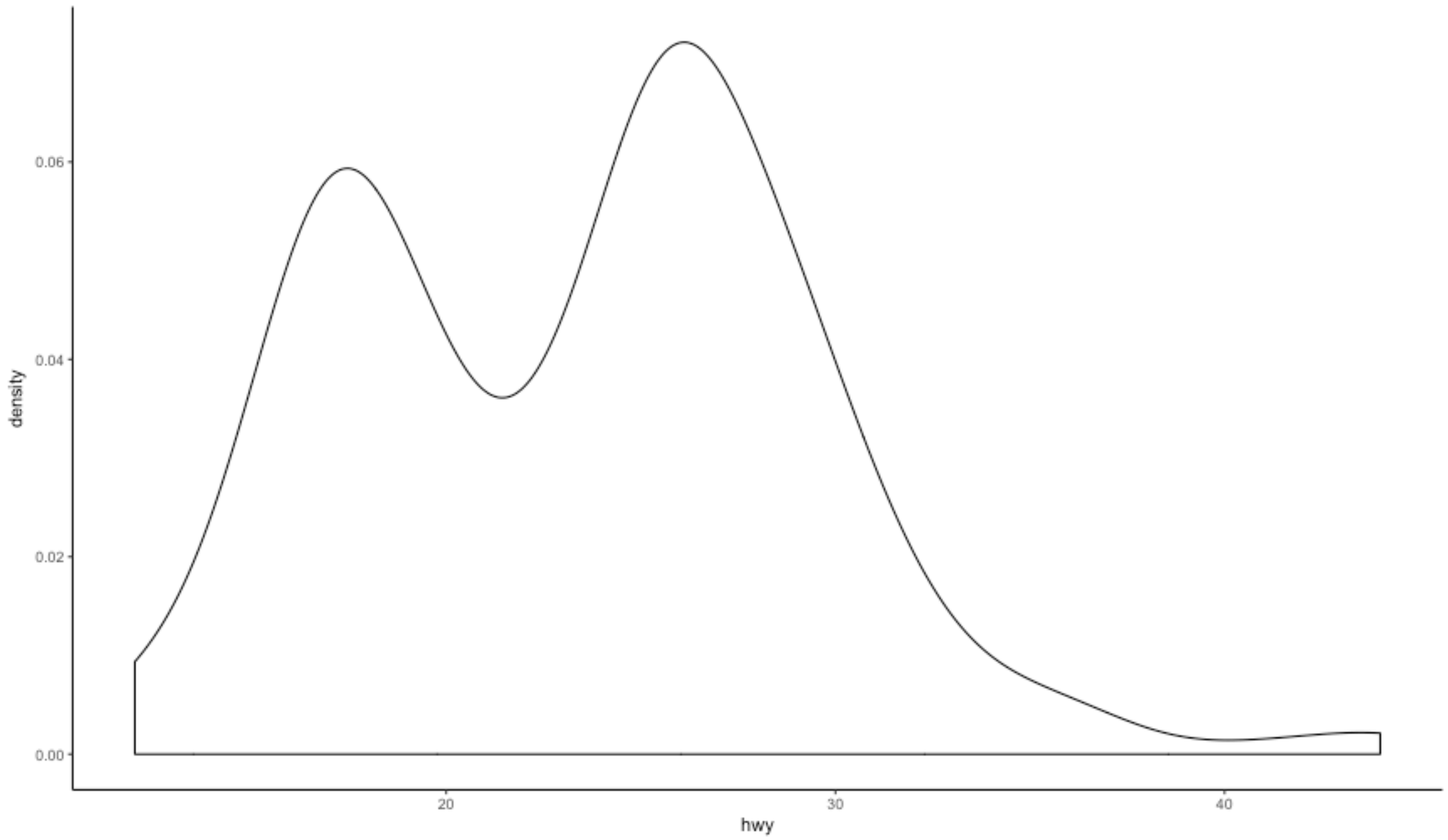
One continuous variable

Histogram



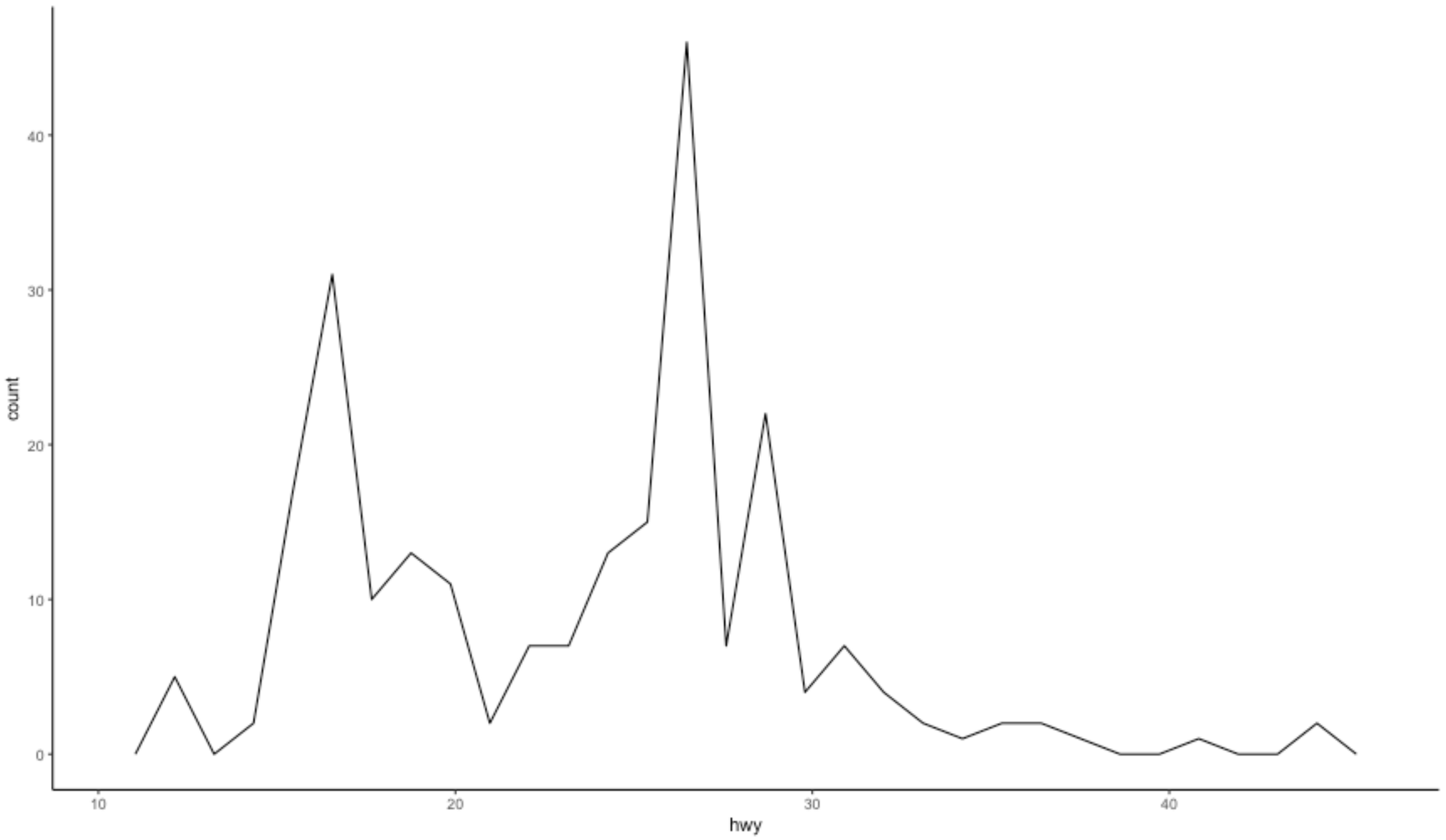
One continuous variable

Density plot

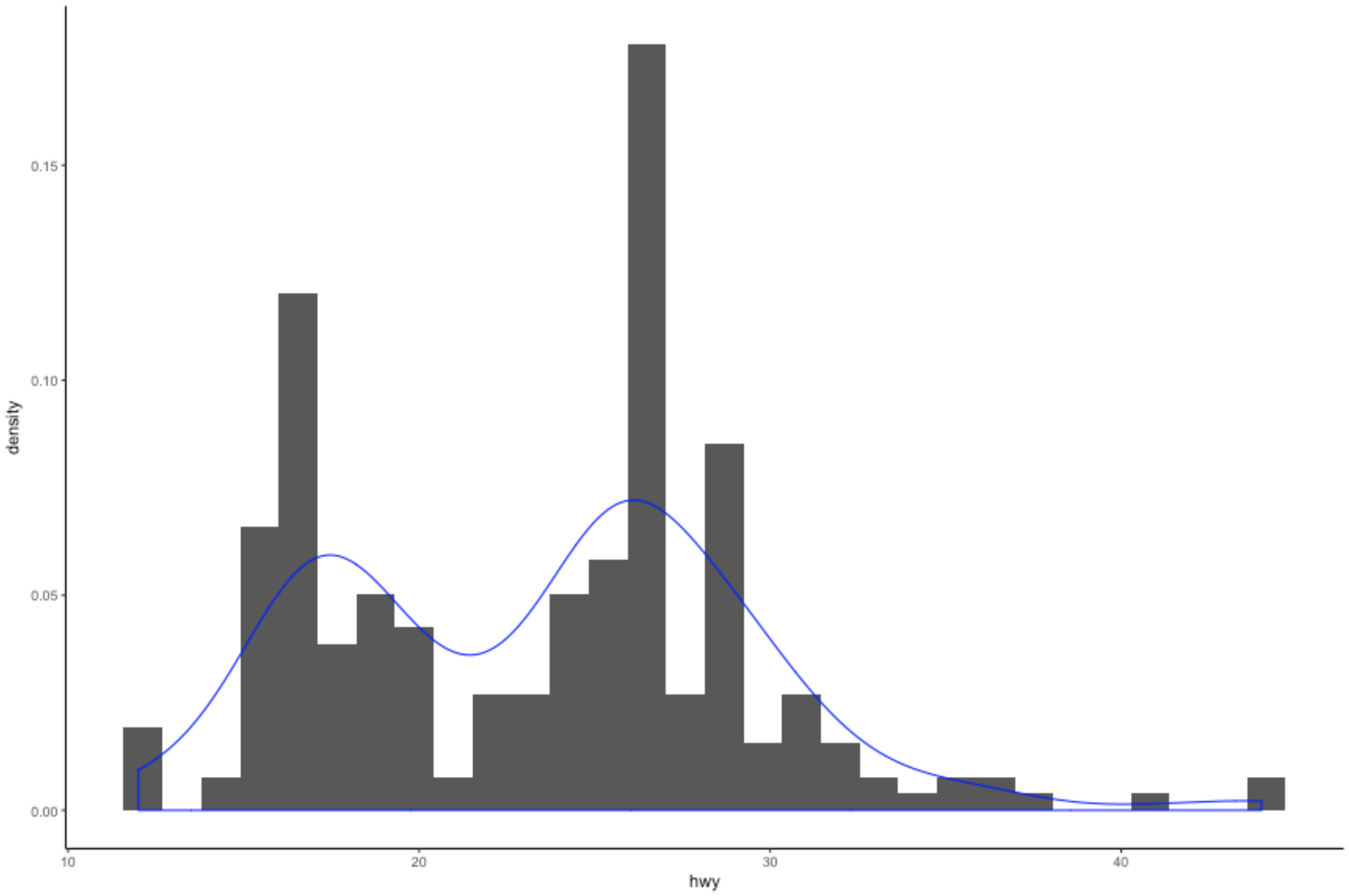


One continuous variable

Frequency polygon

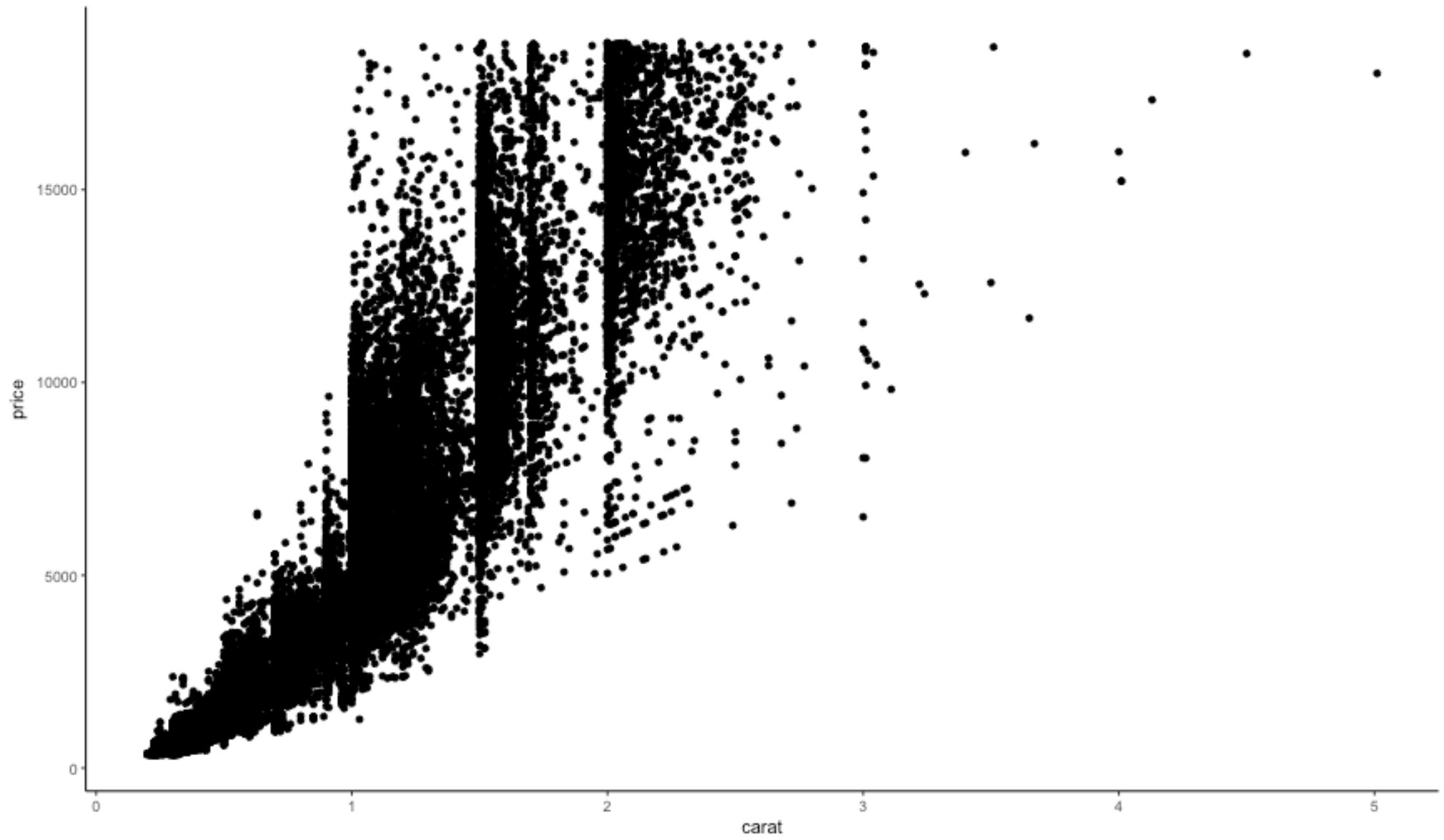


Consider overlays



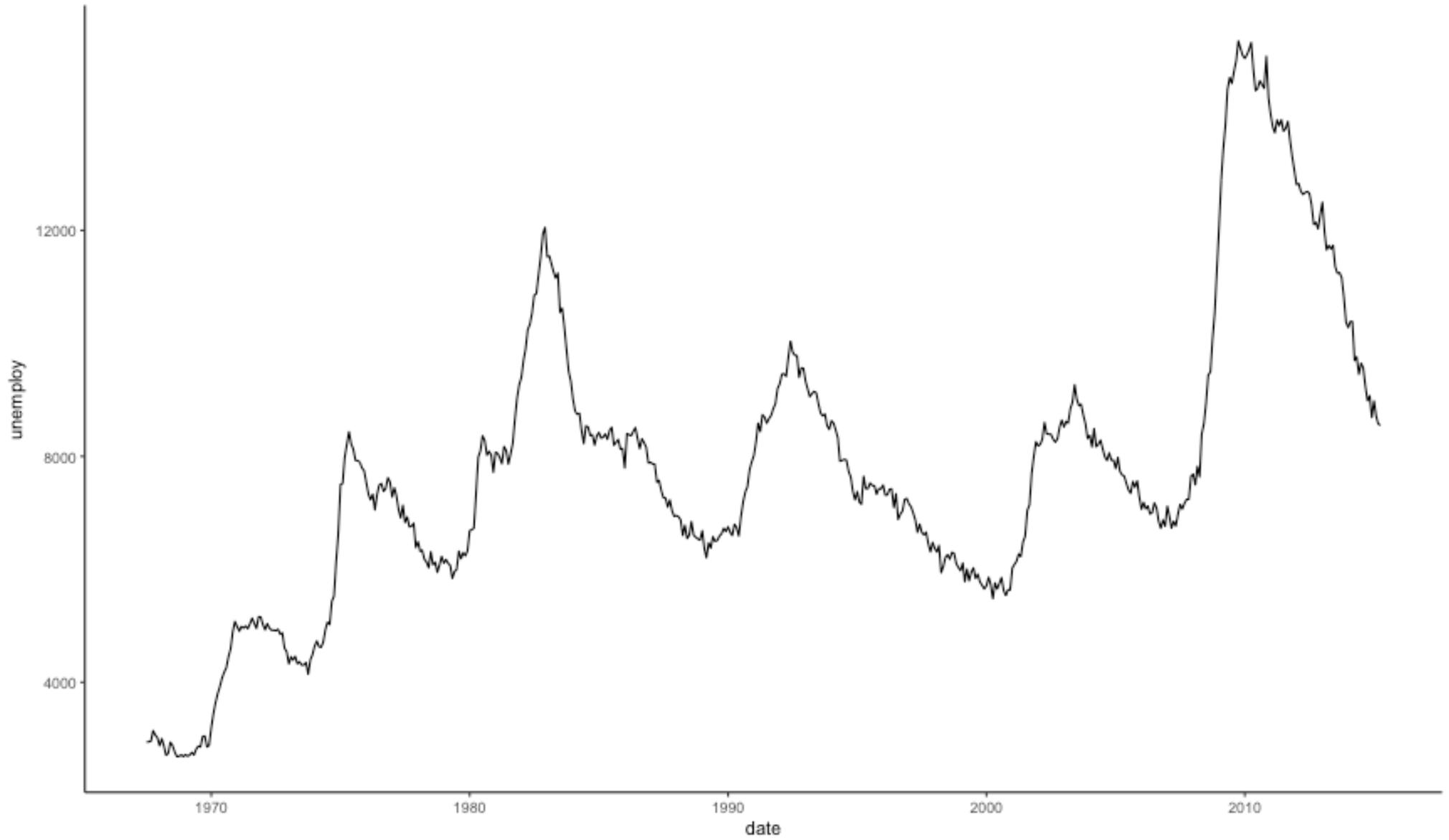
Two continuous variables

Scatterplot

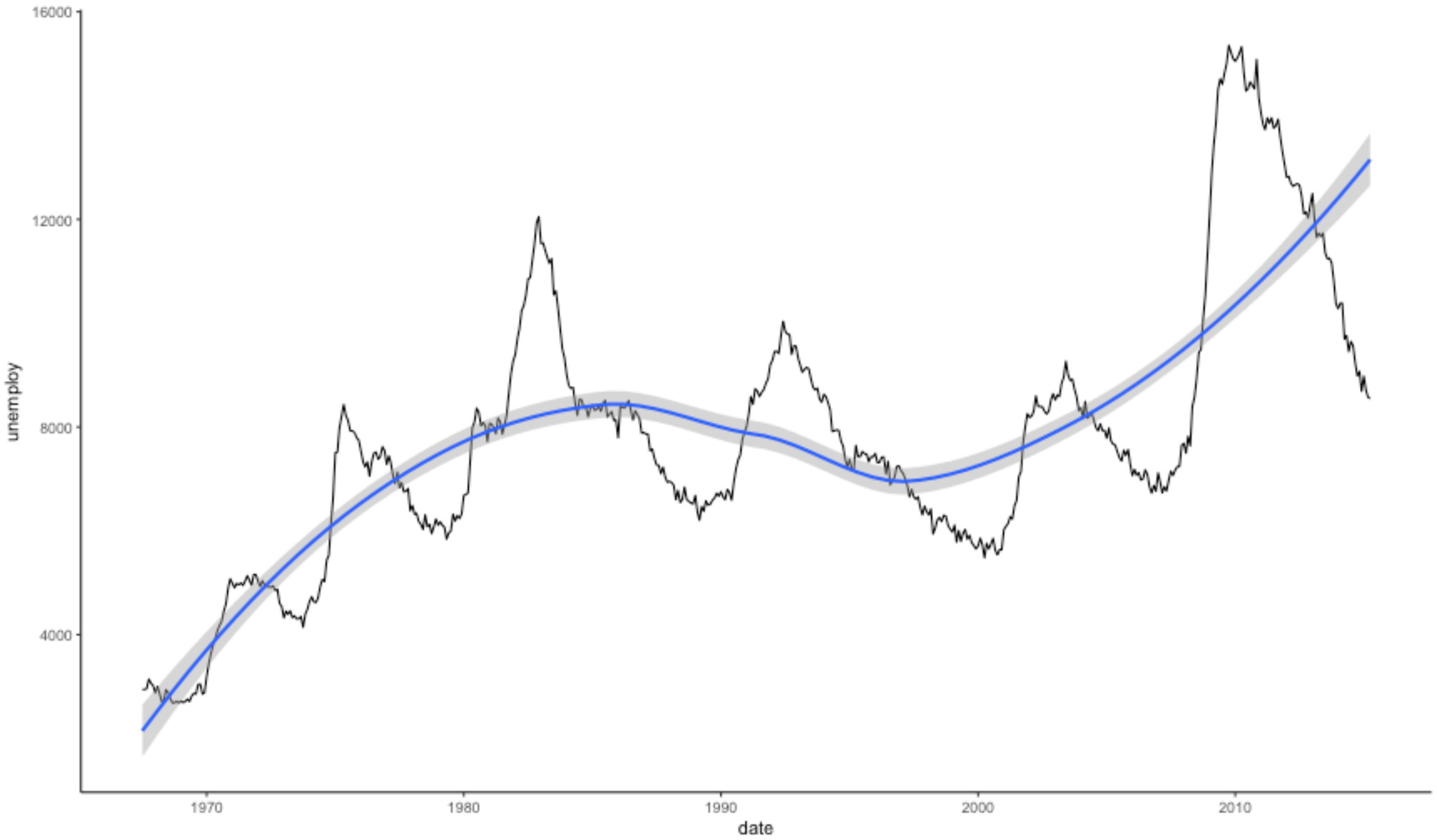


Trend

Line plot (often with date or time on x-axis)

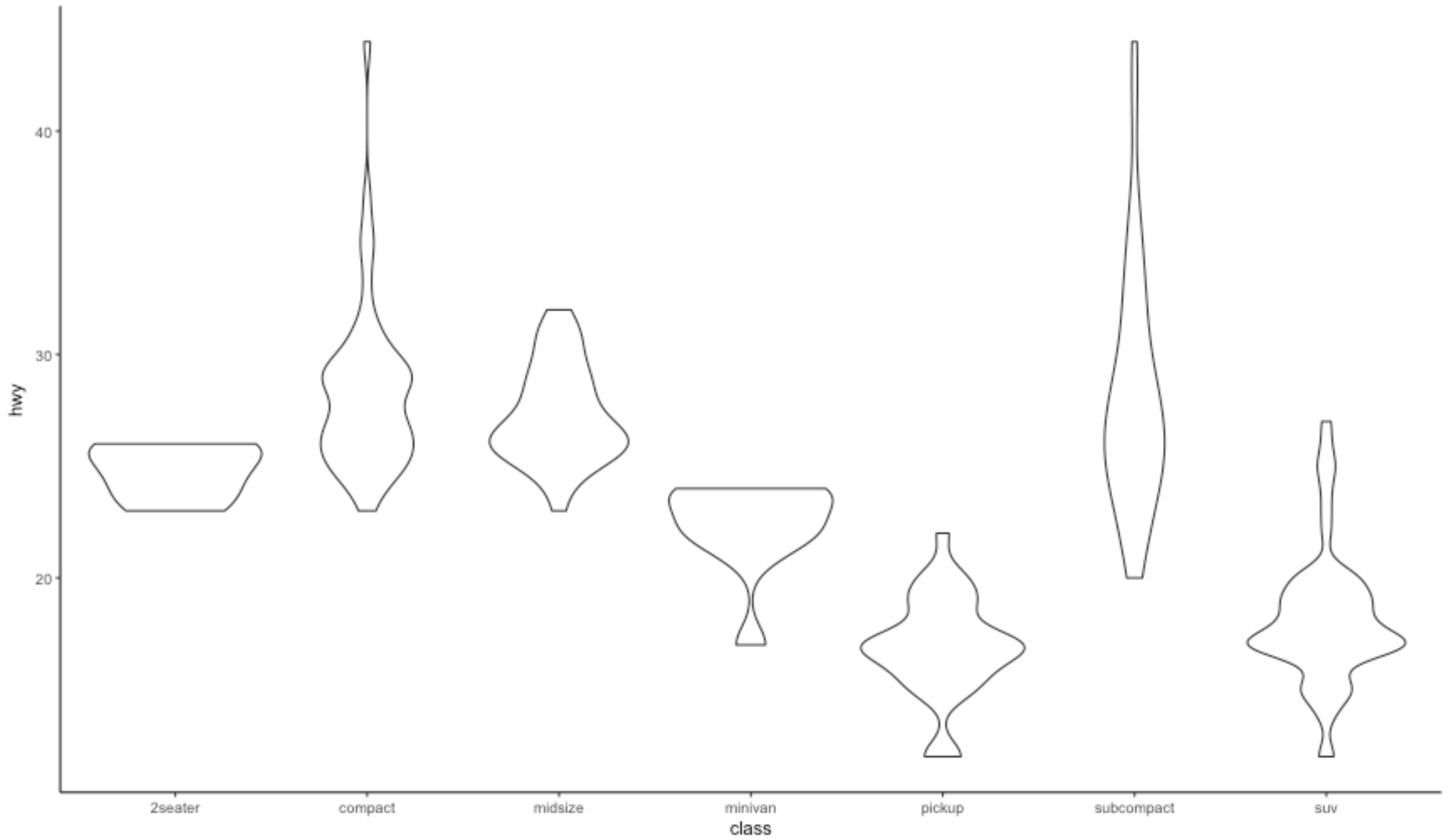


Trend w/smoother

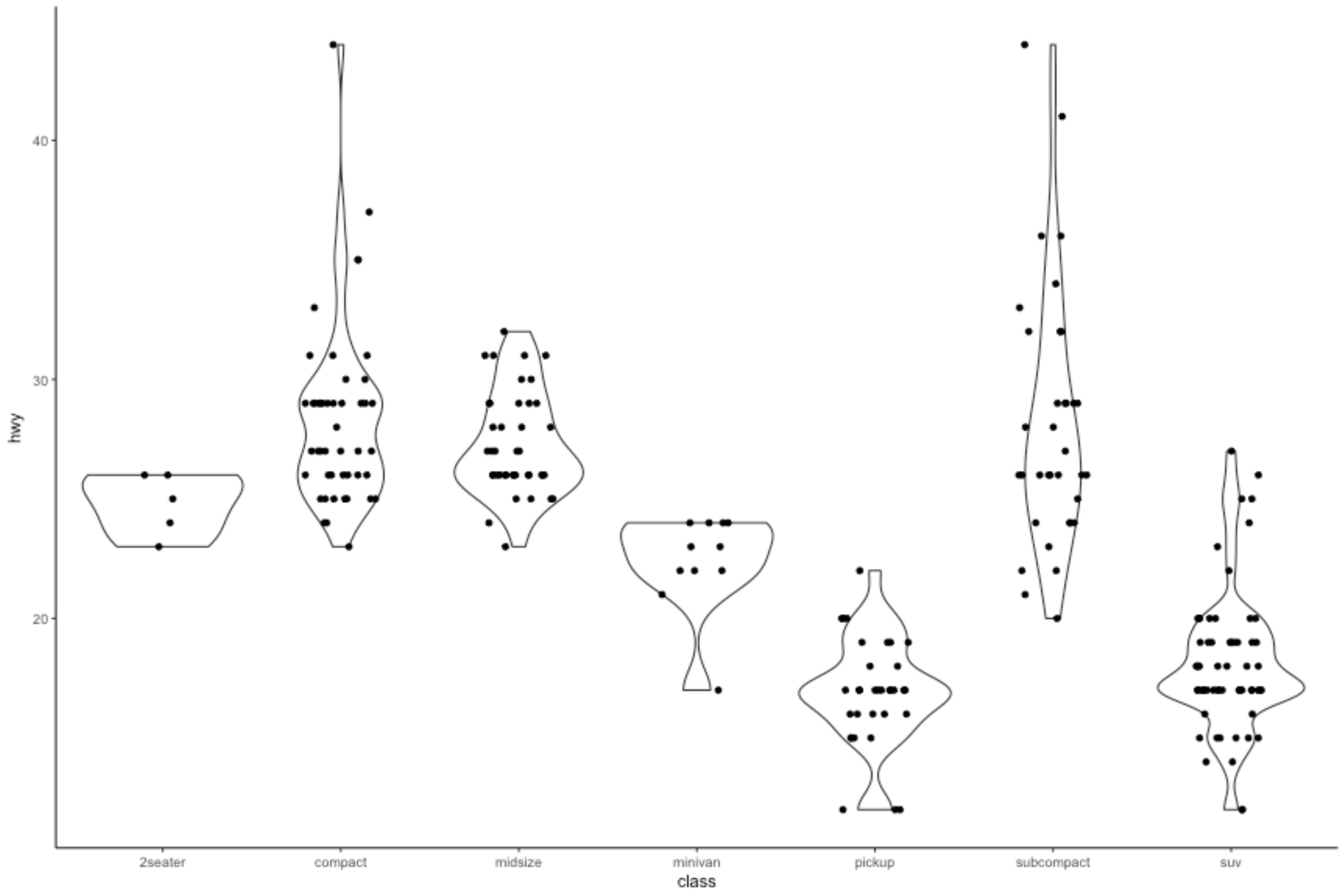


Categorical & Continuous

Violin plots

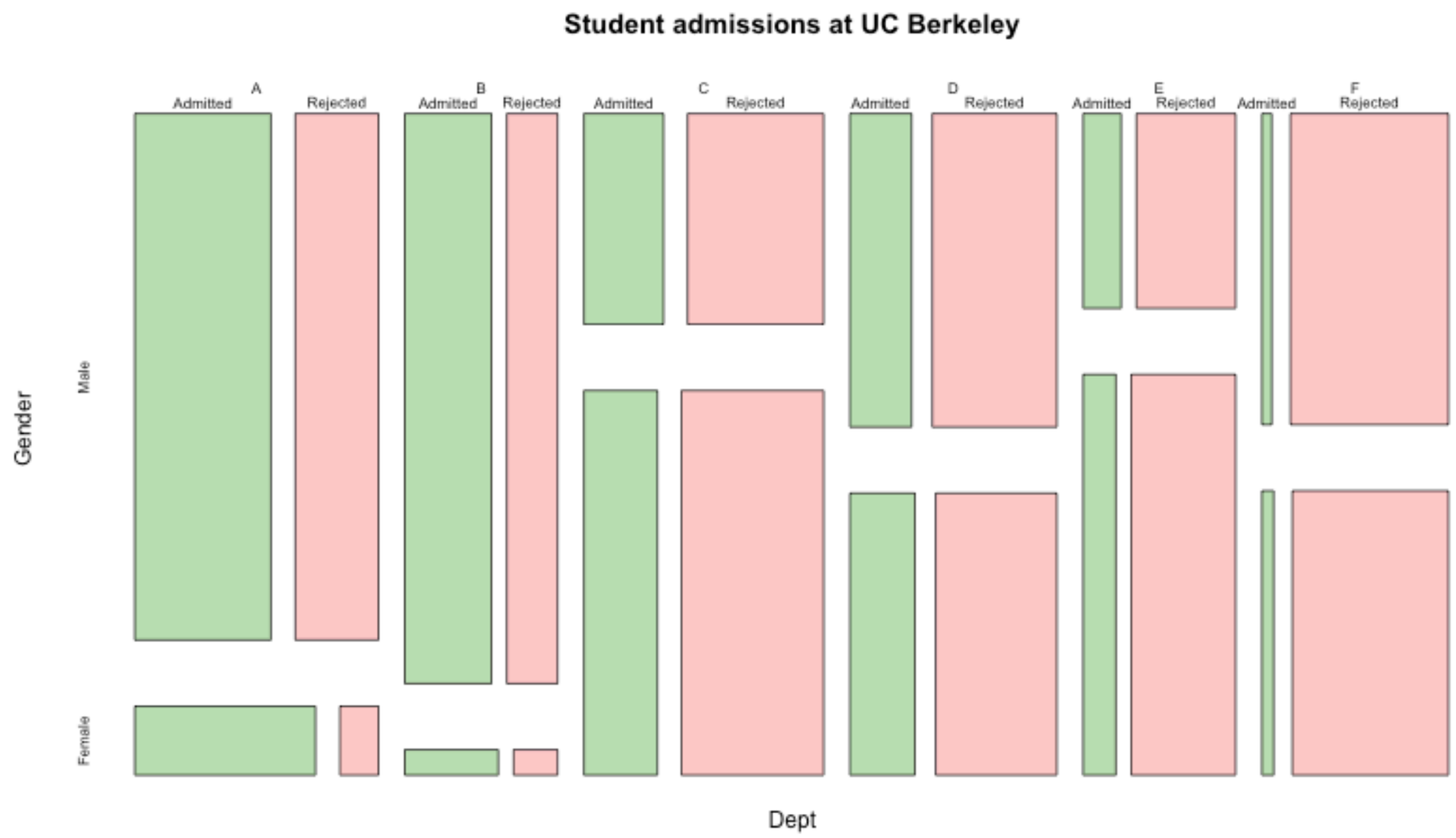


Overlay data



Two categorical variables

Mosaic plot



Don't end up in a blog for wrong reasons

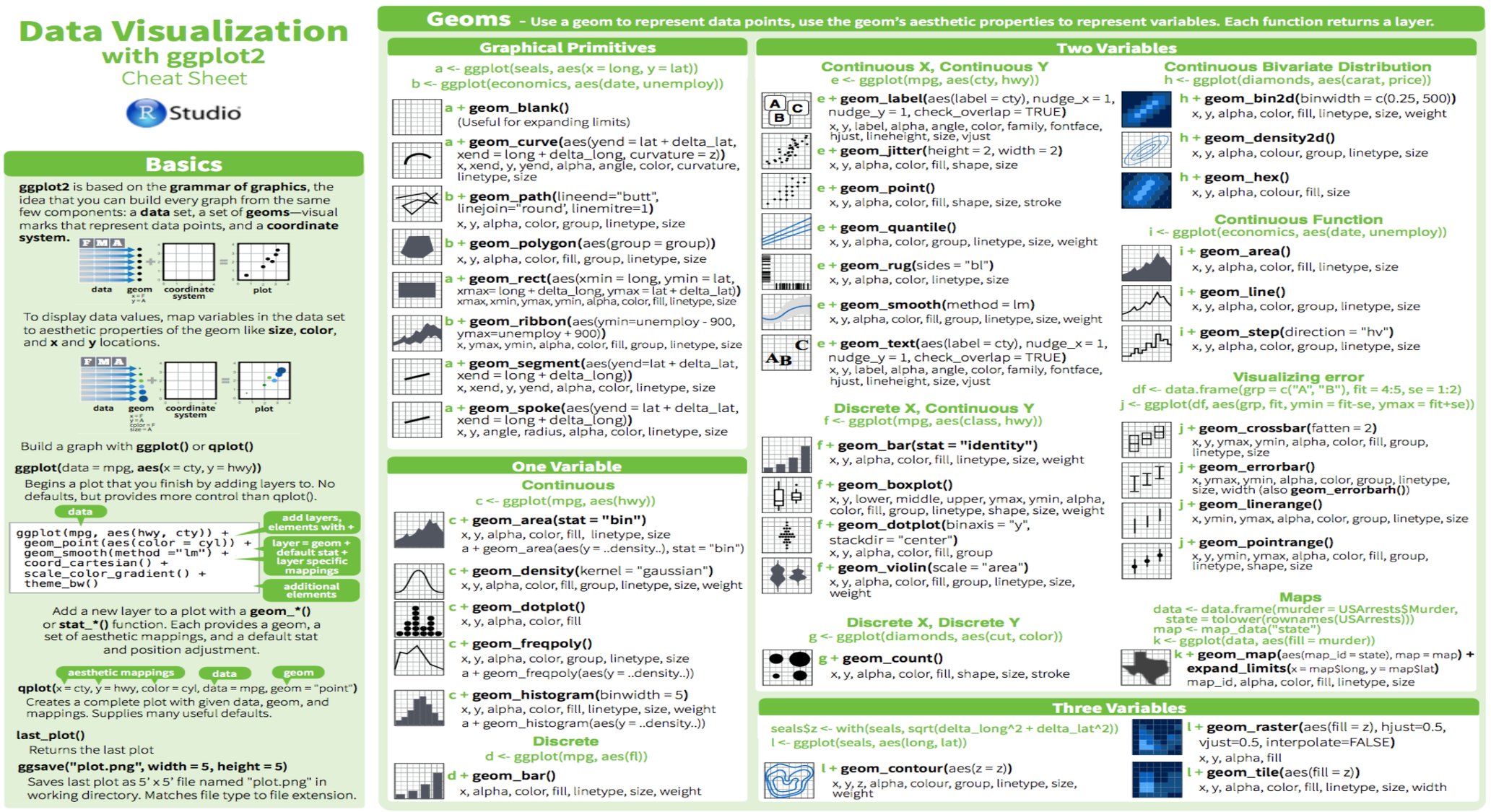
- https://flowingdata.com/2010/05/14/wait-something-isnt-right-here/
- https://flowingdata.com/2009/11/26/fox-news-makes-the-best-pie-chart-ever/

Conclusions

- Essentially never
  - Use pie charts (use bar charts instead)
  - Use dual axes (produce separate plots instead)
  - Truncate axes
  - Use 3D unnecessarily
  - Add color for color's sake (this isn't sales)
- Do
  - Show the data
  - Be as clear as possible
  - Let the data tell the story

Next time

Apply what we've talked about today with R and ggplot!



Data Visualization with ggplot2 Cheat Sheet

RStudio

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same few components: a **data** set, a set of **geoms**—visual marks that represent data points, and a **coordinate system.**

To display data values, map variables in the data set to aesthetic properties of the geom like **size**, **color**, and **x** and **y** locations.

Build a graph with **ggplot()** or **qplot()**

ggplot(data = mpg, **aes(**x = cty, y = hwy**))**
Begins a plot that you finish by adding layers to. No defaults, but provides more control than qplot().

data

```
ggplot(mpg, aes(hwy, cty)) +
  geom_point(aes(color = cyl)) +
  geom_smooth(method ="lm") +
  coord_cartesian() +
  scale_color_gradient() +
  theme_bw()
```

add layers, elements with +
layer = geom + default stat + layer specific mappings
additional elements

Add a new layer to a plot with a **geom_*()** or **stat_*()** function. Each provides a geom, a set of aesthetic mappings, and a default stat and position adjustment.

aesthetic mappings | data | geom

qplot(x = cty, y = hwy, color = cyl, data = mpg, geom = "point"**)**
Creates a complete plot with given data, geom, and mappings. Supplies many useful defaults.

last_plot()
Returns the last plot

ggsave("plot.png", width = 5, height = 5)
Saves last plot as 5' x 5' file named "plot.png" in working directory. Matches file type to file extension.

Geoms - Use a geom to represent data points, use the geom's aesthetic properties to represent variables. Each function returns a layer.

Graphical Primitives

a <- ggplot(seals, aes(x = long, y = lat))
b <- ggplot(economics, aes(date, unemploy))

- a + **geom_blank()** (Useful for expanding limits)
- a + **geom_curve(**aes(yend = lat + delta_lat, xend = long + delta_long, curvature = z**))** x, xend, y, yend, alpha, angle, color, curvature, linetype, size
- b + **geom_path(**lineend="butt", linejoin="round', linemitre=1**)** x, y, alpha, color, group, linetype, size
- b + **geom_polygon(**aes(group = group**))** x, y, alpha, color, fill, group, linetype, size
- a + **geom_rect(**aes(xmin = long, ymin = lat, xmax= long + delta_long, ymax = lat + delta_lat**))** xmax, xmin, ymax, ymin, alpha, color, fill, linetype, size
- b + **geom_ribbon(**aes(ymin=unemploy - 900, ymax=unemploy + 900**))** x, ymax, ymin, alpha, color, fill, group, linetype, size
- a + **geom_segment(**aes(yend=lat + delta_lat, xend = long + delta_long**))** x, xend, y, yend, alpha, color, linetype, size
- a + **geom_spoke(**aes(yend = lat + delta_lat, xend = long + delta_long**))** x, y, angle, radius, alpha, color, linetype, size

One Variable

Continuous

c <- ggplot(mpg, aes(hwy))

- c + **geom_area(stat = "bin")** x, y, alpha, color, fill, linetype, size a + geom_area(aes(y = ..density..), stat = "bin")
- c + **geom_density(**kernel = "gaussian"**)** x, y, alpha, color, fill, group, linetype, size, weight
- c + **geom_dotplot()** x, y, alpha, color, fill
- c + **geom_freqpoly()** x, y, alpha, color, group, linetype, size a + geom_freqpoly(aes(y = ..density..))
- c + **geom_histogram(**binwidth = 5**)** x, y, alpha, color, fill, linetype, size, weight a + geom_histogram(aes(y = ..density..))

Discrete

d <- ggplot(mpg, aes(fl))

- d + **geom_bar()** x, alpha, color, fill, linetype, size, weight

Two Variables

Continuous X, Continuous Y

e <- ggplot(mpg, aes(cty, hwy))

- e + **geom_label(**aes(label = cty), nudge_x = 1, nudge_y = 1, check_overlap = TRUE**)** x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust
- e + **geom_jitter(**height = 2, width = 2**)** x, y, alpha, color, fill, shape, size
- e + **geom_point()** x, y, alpha, color, fill, shape, size, stroke
- e + **geom_quantile()** x, y, alpha, color, group, linetype, size, weight
- e + **geom_rug(**sides = "bl"**)** x, y, alpha, color, linetype, size
- e + **geom_smooth(**method = lm**)** x, y, alpha, color, fill, group, linetype, size, weight
- e + **geom_text(**aes(label = cty), nudge_x = 1, nudge_y = 1, check_overlap = TRUE**)** x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust

Discrete X, Continuous Y

f <- ggplot(mpg, aes(class, hwy))

- f + **geom_bar(stat = "identity")** x, y, alpha, color, fill, linetype, size, weight
- f + **geom_boxplot()** x, y, lower, middle, upper, ymax, ymin, alpha, color, fill, group, linetype, shape, size, weight
- f + **geom_dotplot(**binaxis = "y", stackdir = "center"**)** x, y, alpha, color, fill, group
- f + **geom_violin(**scale = "area"**)** x, y, alpha, color, fill, group, linetype, size, weight

Discrete X, Discrete Y

g <- ggplot(diamonds, aes(cut, color))

- g + **geom_count()** x, y, alpha, color, fill, shape, size, stroke

Continuous Bivariate Distribution

h <- ggplot(diamonds, aes(carat, price))

- h + **geom_bin2d(**binwidth = c(0.25, 500)**)** x, y, alpha, color, fill, linetype, size, weight
- h + **geom_density2d()** x, y, alpha, colour, group, linetype, size
- h + **geom_hex()** x, y, alpha, colour, fill, size

Continuous Function

i <- ggplot(economics, aes(date, unemploy))

- i + **geom_area()** x, y, alpha, color, fill, linetype, size
- i + **geom_line()** x, y, alpha, color, group, linetype, size
- i + **geom_step(**direction = "hv"**)** x, y, alpha, color, group, linetype, size

Visualizing error

df <- data.frame(grp = c("A", "B"), fit = 4:5, se = 1:2)
j <- ggplot(df, aes(grp, fit, ymin = fit-se, ymax = fit+se))

- j + **geom_crossbar(**fatten = 2**)** x, y, ymax, ymin, alpha, color, fill, group, linetype, size
- j + **geom_errorbar()** x, ymax, ymin, alpha, color, group, linetype, size, width (also **geom_errorbarh()**)
- j + **geom_linerange()** x, ymin, ymax, alpha, color, group, linetype, size
- j + **geom_pointrange()** x, y, ymin, ymax, alpha, color, fill, group, linetype, shape, size

Maps

data <- data.frame(murder = USArrests$Murder, state = tolower(rownames(USArrests)))
map <- map_data("state")
k <- ggplot(data, aes(fill = murder))

- k + **geom_map(**aes(map_id = state), map = map**)** + **expand_limits(**x = map$long, y = map$lat**)** map_id, alpha, color, fill, linetype, size

Three Variables

seals$z <- with(seals, sqrt(delta_long^2 + delta_lat^2))
l <- ggplot(seals, aes(long, lat))

- l + **geom_contour(**aes(z = z)) x, y, z, alpha, colour, group, linetype, size, weight
- l + **geom_raster(**aes(fill = z), hjust=0.5, vjust=0.5, interpolate=FALSE**)** x, y, alpha, fill
- l + **geom_tile(**aes(fill = z)**)** x, y, alpha, color, fill, linetype, size, width

RStudio® is a trademark of RStudio, Inc. • CC BY RStudio • info@rstudio.com • 844-448-1212 • rstudio.com

Learn more at **docs.ggplot2.org** • ggplot2 2.0.0 • Updated: 12/15