Social Data Science

Data Visualization

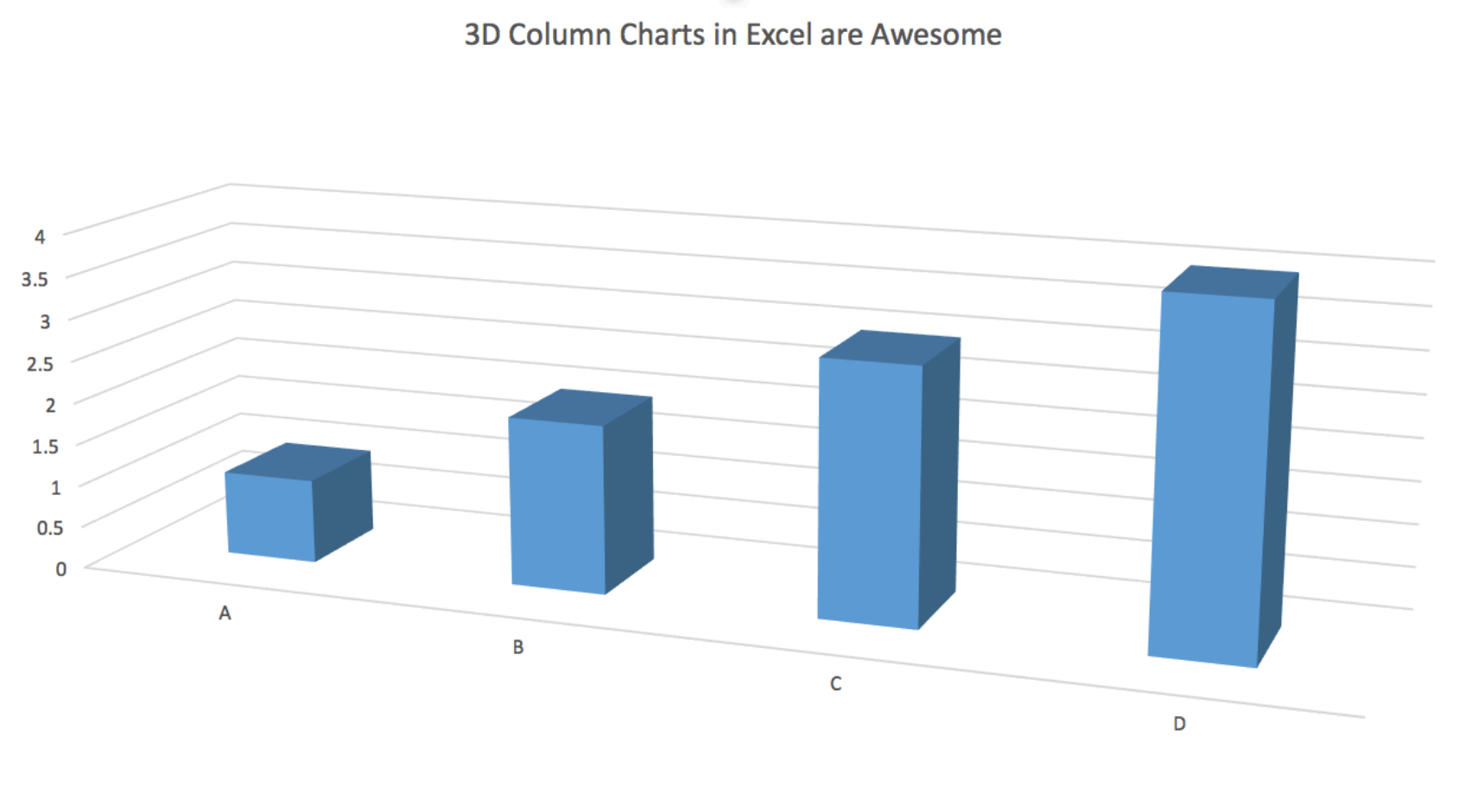
Sebastian Barfort

August 04, 2016

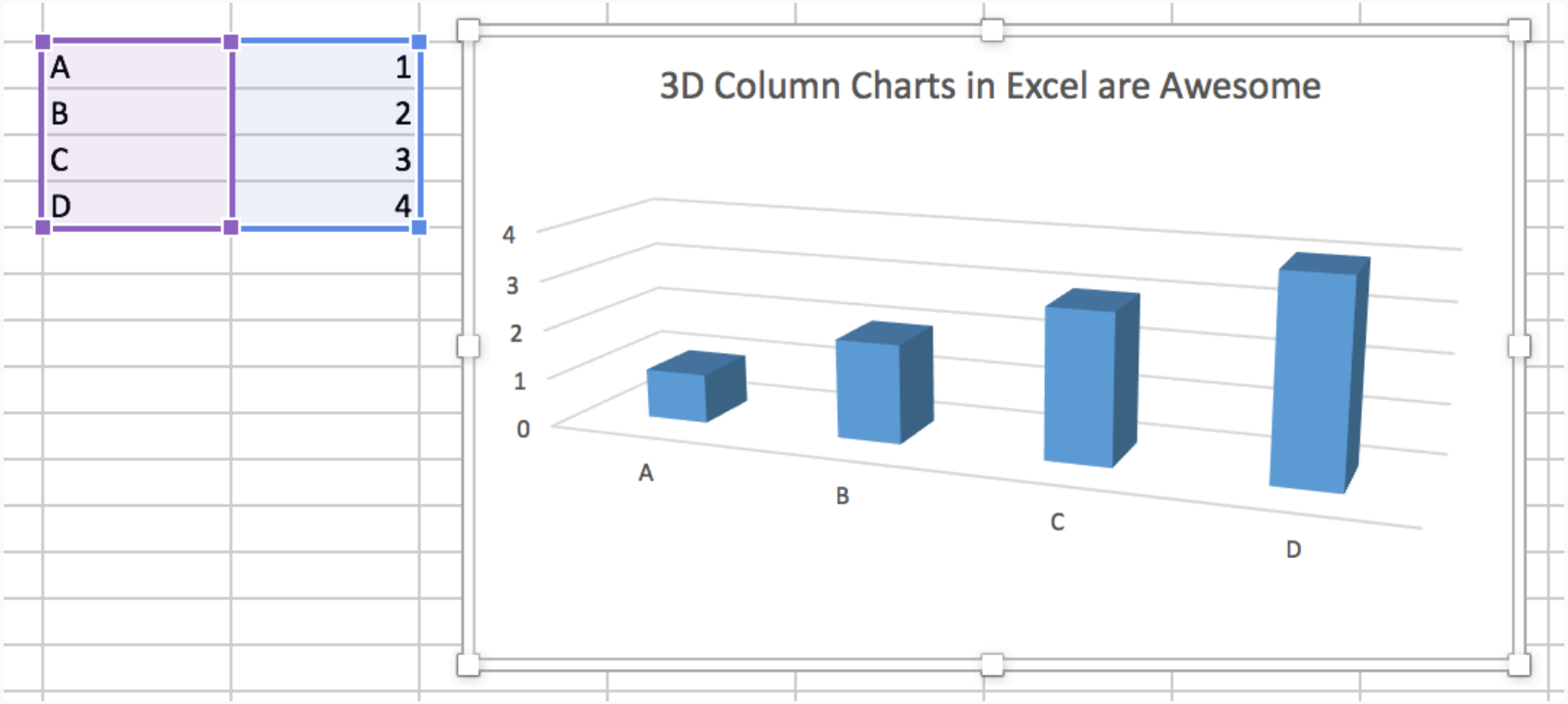
University of Copenhagen
Department of Economics

Who’s ahead in the polls?

What values are displayed in this chart?



The answer may surprise you



What are you trying to accomplish?

1. Who's the audience?

   - Exploratory (use defaults) vs. explanatory (customize)
   - Raw data vs. model results

2. Graphs should be self explanatory
3. A graph is a narrative - should convey key point(s)

Schwabish (2014)

1. Show the data (many graphs show too much)
2. Reduce the clutter
3. Integrate text and graph

Discuss in groups

1. Look at the eight transformed graphs in Schwabish (2014). What do you think about the transformations?
2. Are there other objectives of data visualization not mentioned by Schwabish?
3. If you were to improve on one of the eight transformed graphs, which one would you choose and how would you change it?

OTHER CONSIDERATIONS

Color palette

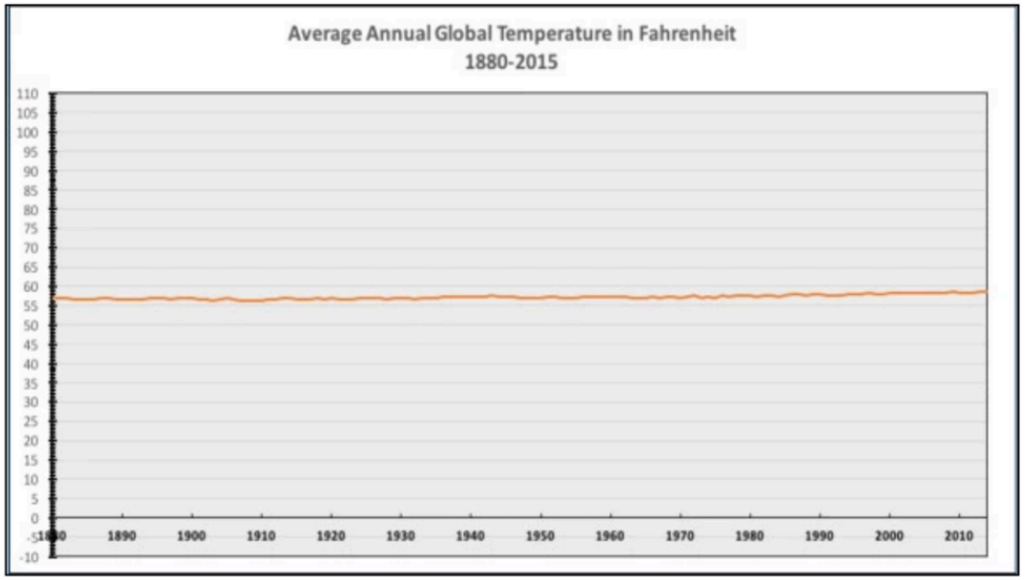
Is your data numeric, binary, catetorical, text?

Should you truncate your y axis?

Remember

Visualizing data is not just a matter of good taste

Basic perceptual processes play a very strong role



National Review
@NRO

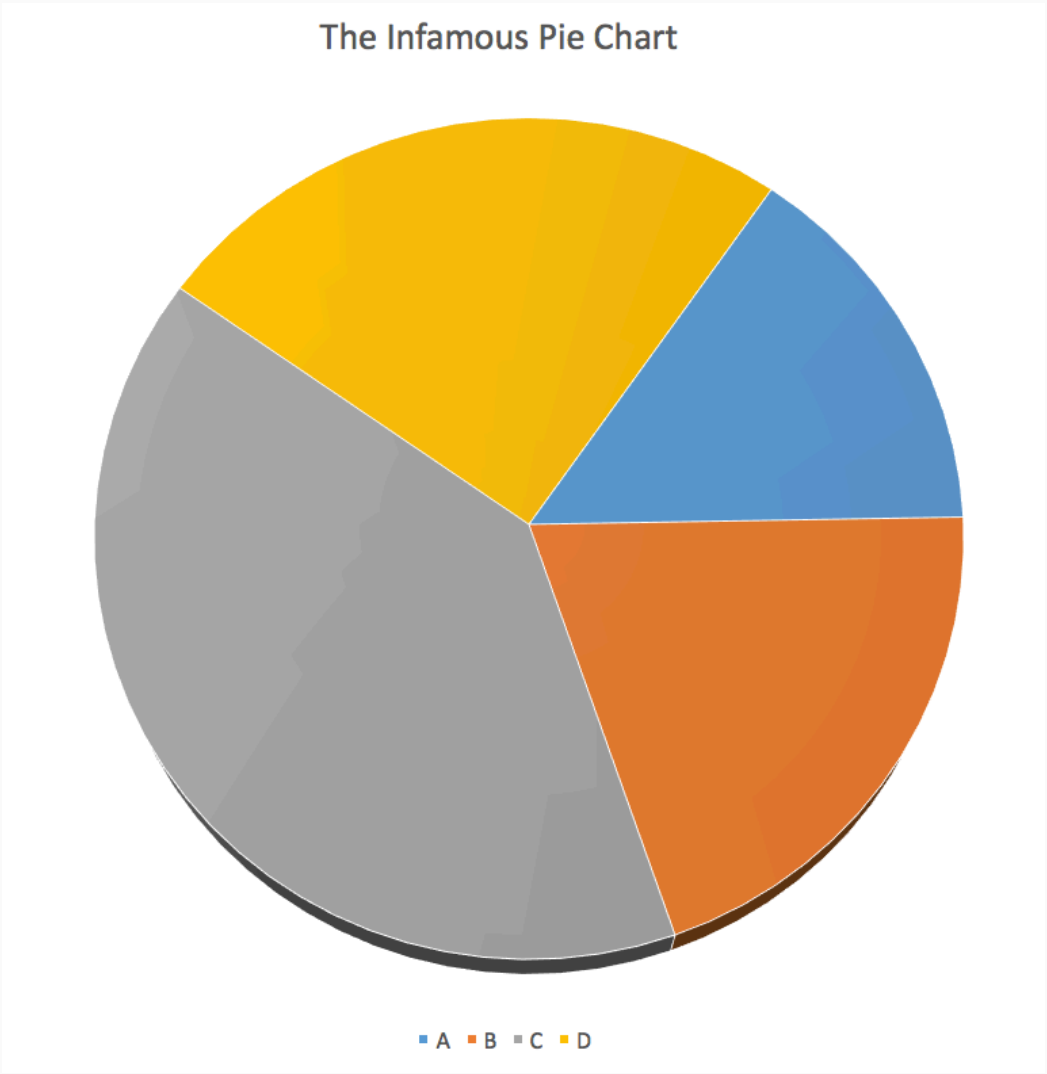
Follow

The only #climatechange chart you need to see. natl.re/wPKpro

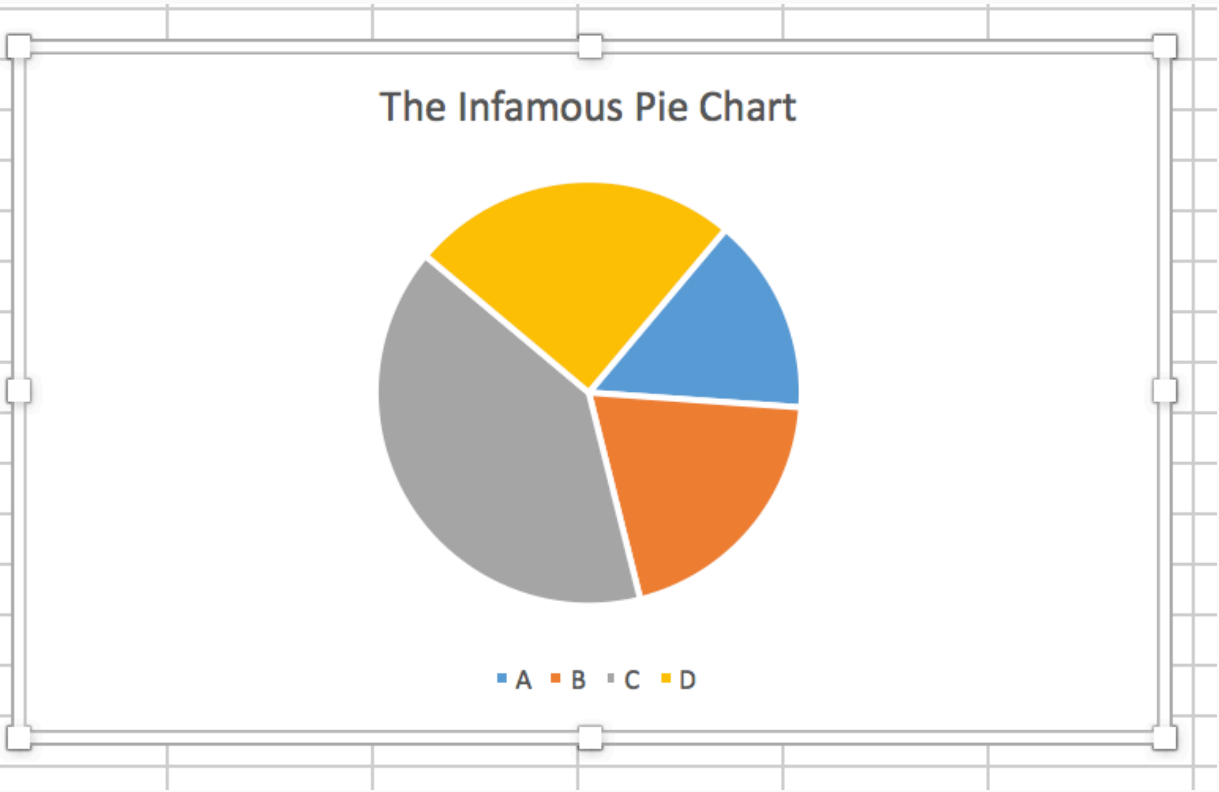
(h/t @powerlineUS)

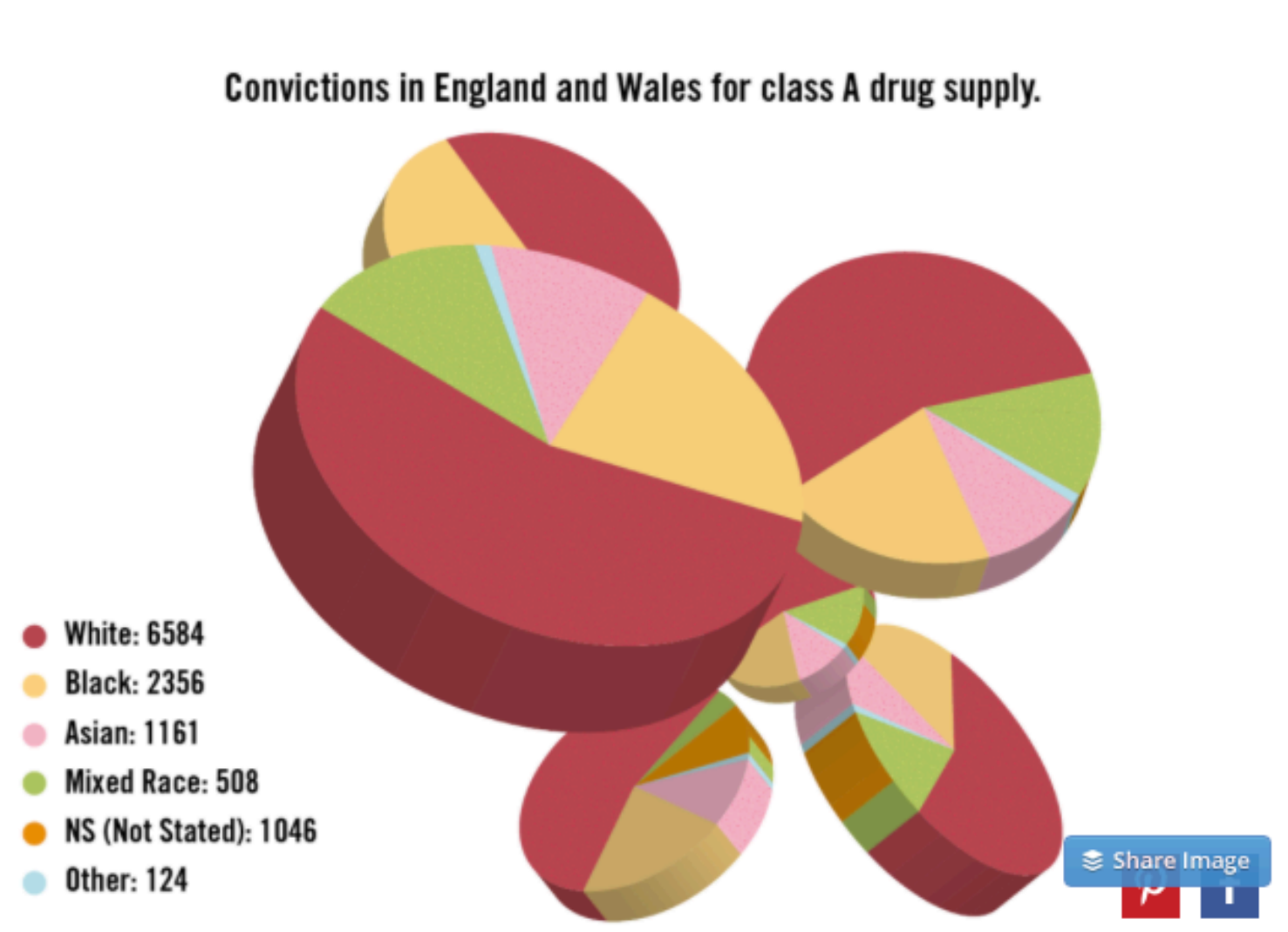
3:36 PM - 14 Dec 2015

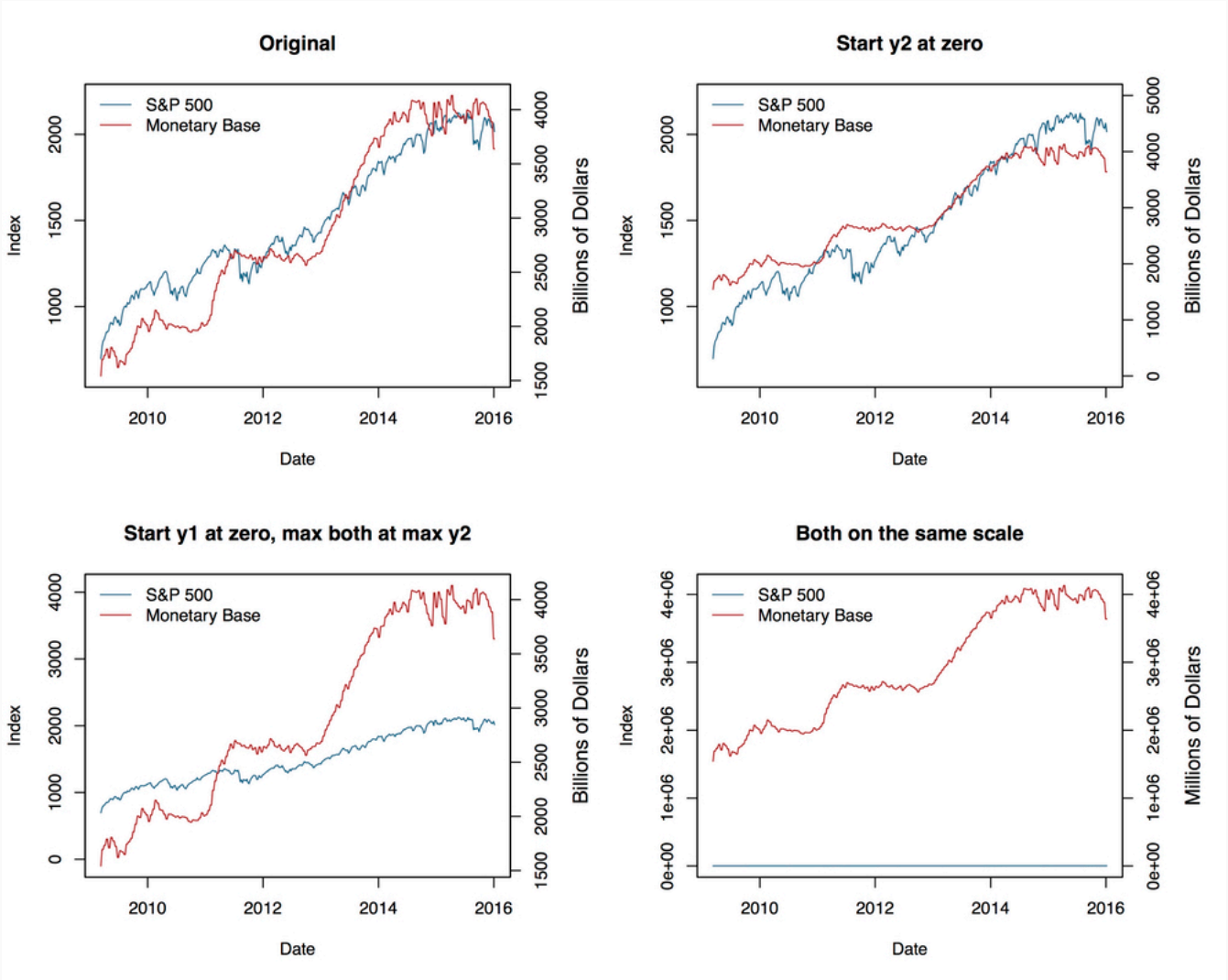
413 318


The Infamous Pie Chart
A
B
C
D

A	15
B	20
C	40
D	25
	100



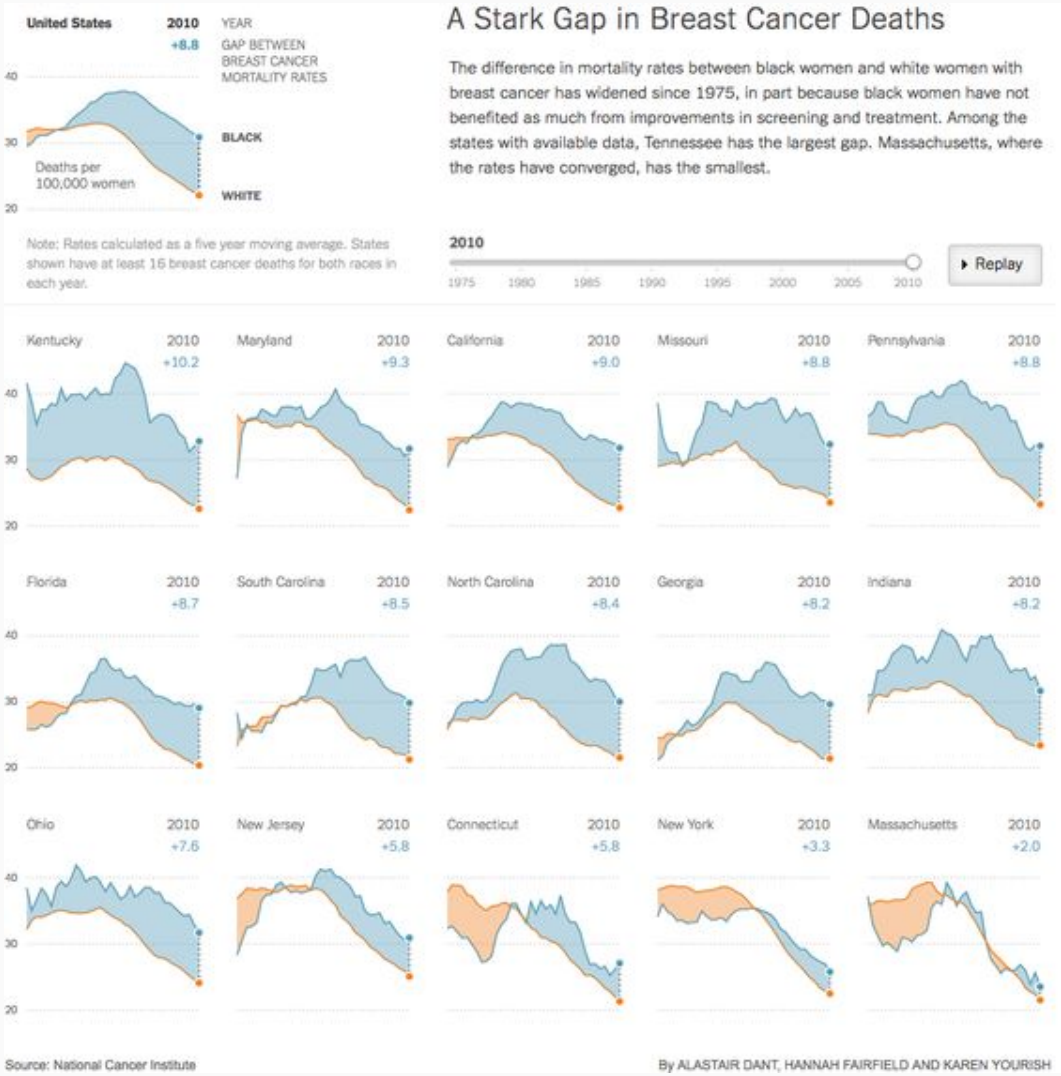

Convictions in England and Wales for class A drug supply.
White: 6584
Black: 2356
Asian: 1161
Mixed Race: 508
NS (Not Stated): 1046
Other: 124
Share Image


Original
S&P 500
Monetary Base
Index
Billions of Dollars
Date
Start y2 at zero
S&P 500
Monetary Base
Index
Billions of Dollars
Date
Start y1 at zero, max both at max y2
S&P 500
Monetary Base
Index
Billions of Dollars
Date
Both on the same scale
S&P 500
Monetary Base
Index
Millions of Dollars
Date

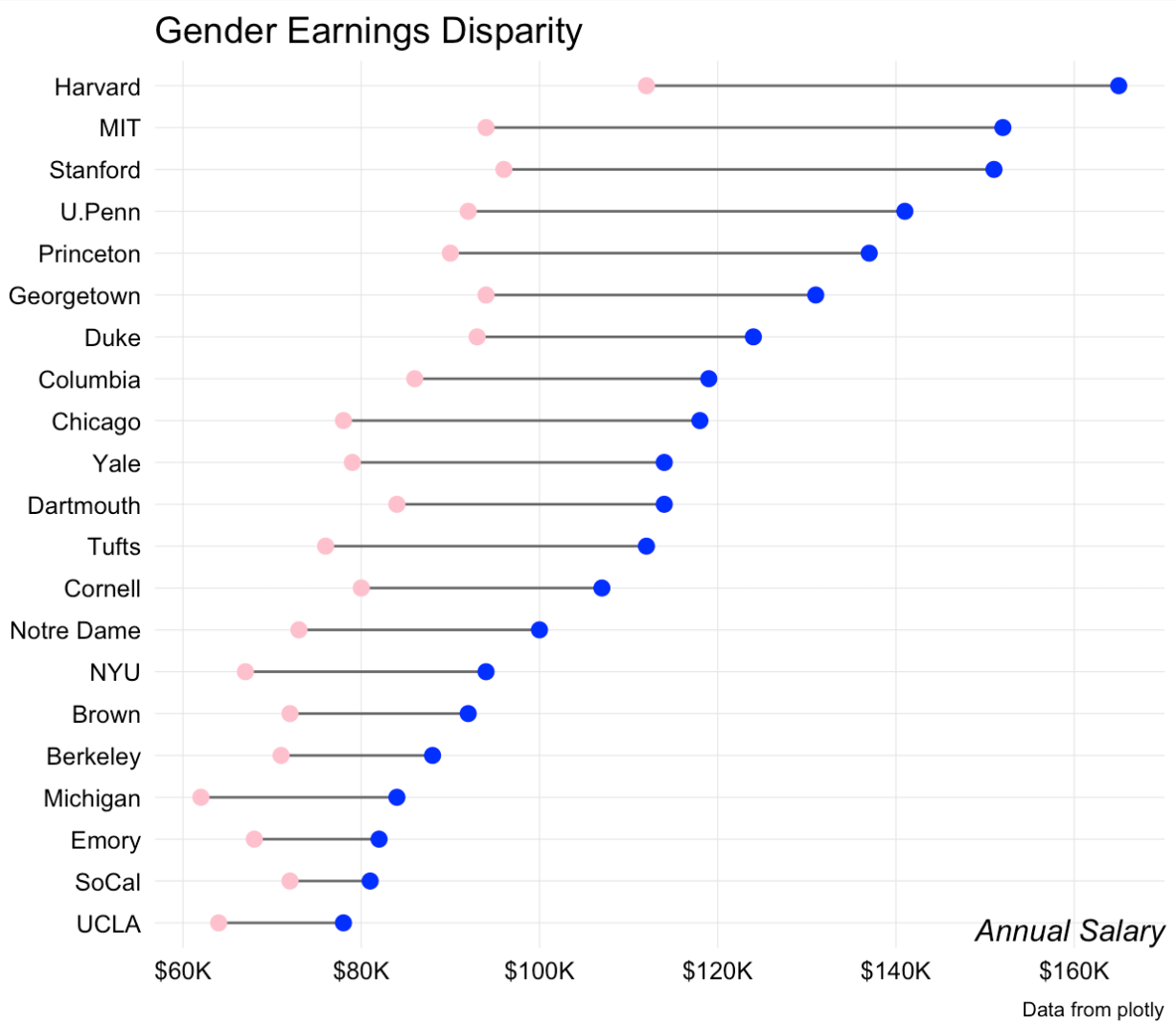
Small multiples

Tufte

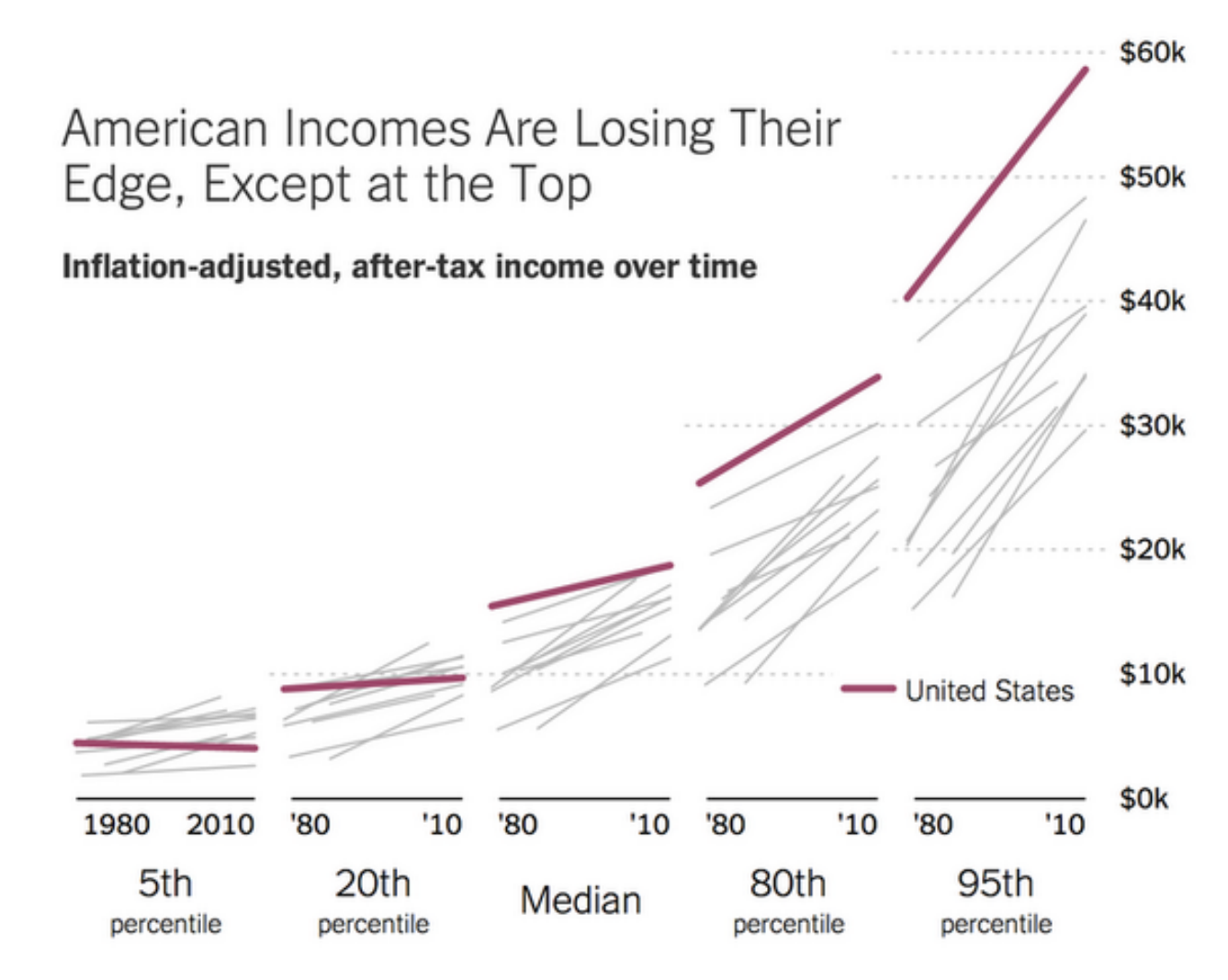
> *"Illustrations of postage-stamp size are indexed by category or a label, sequenced over time like the frames of a movie, or ordered by a quantitative variable not used in the single image itself."*


United States
2010
+8.8
YEAR
GAP BETWEEN BREAST CANCER MORTALITY RATES
BLACK
WHITE
Deaths per 100,000 women
40
30
20
A Stark Gap in Breast Cancer Deaths
The difference in mortality rates between black women and white women with breast cancer has widened since 1975, in part because black women have not benefited as much from improvements in screening and treatment. Among the states with available data, Tennessee has the largest gap. Massachusetts, where the rates have converged, has the smallest.
Note: Rates calculated as a five year moving average. States shown have at least 16 breast cancer deaths for both races in each year.
2010
1975 1980 1985 1990 1995 2000 2005 2010
Replay
Kentucky 2010 +10.2
Maryland 2010 +9.3
California 2010 +9.0
Missouri 2010 +8.8
Pennsylvania 2010 +8.8
Florida 2010 +8.7
South Carolina 2010 +8.5
North Carolina 2010 +8.4
Georgia 2010 +8.2
Indiana 2010 +8.2
Ohio 2010 +7.6
New Jersey 2010 +5.8
Connecticut 2010 +5.8
New York 2010 +3.3
Massachusetts 2010 +2.0
Source: National Cancer Institute
By ALASTAIR DANT, HANNAH FAIRFIELD AND KAREN YOURISH

“LOLLIPOPS” AND “DUMBBELLS”



SLOPEGRAPHS



Static

1. Base R
2. `lattice`
3. `ggplot2`

Interactive

1. D3 (javascript)
2. htmlwidgets (R)
3. Tableau (commercial)

ggplot2

WHY?

Positives

- Great defaults
- Intuitive
- Extremely well documented

Negatives

- Slow
- Limited functionality

GRAMMAR OF GRAPHICS

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

You start with your data, and then you assign a geometry to elements of that data, such as circle size to population, then you draw those geometries based upon some scaling of your data. When you think about visualization this way it helps you develop a better understanding of the data itself and think of proper ways to visualize it.

A PEW VISUALIZATION OF PARTISANSHIP

The Pew Research Center did an interesting visualization of political polarization in the U.S.

Polarization

> *The new survey finds that as ideological consistency has become more common, it has become increasingly aligned with partisanship. Looking at 10 political values questions tracked since 1994, more Democrats now give uniformly liberal responses, and more Republicans give uniformly conservative responses than at any point in the last 20 years*

Bob Rudis recreated their plot in `ggplot2`. Let's see how.

Political Polarization, 1994-2014

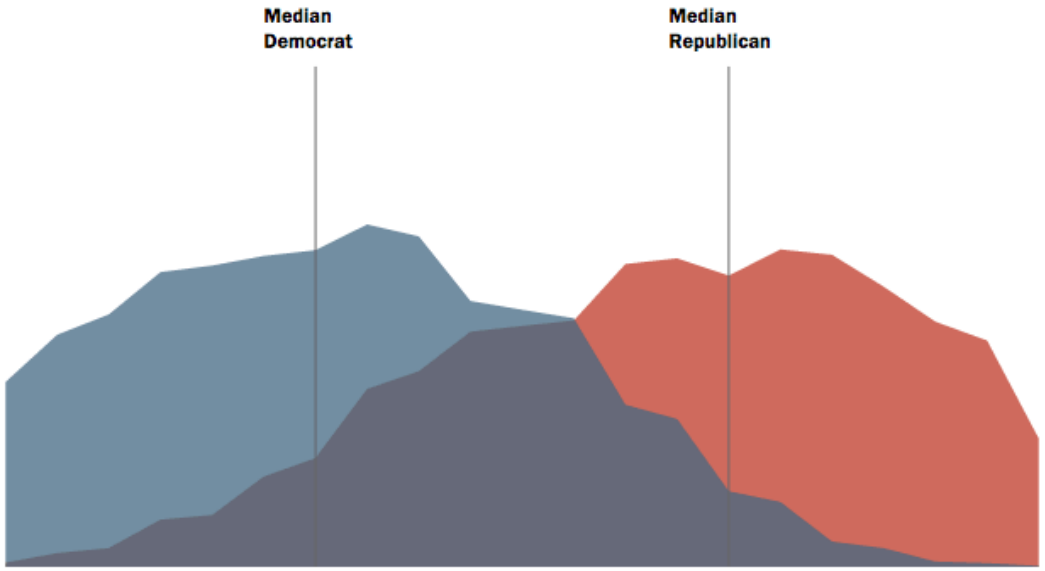
The interactive below illustrates the shift since 1994, using data from five Pew Research Center surveys. In addition to viewing results both for the total population and by party, they can be filtered on just the share of Americans (about one-third of the public) who are most politically active.

See results for: GENERAL POPULATION | POLITICALLY ACTIVE — OVERALL | BY PARTY

Animate data from 1994-2014

Year shown:
2014

View an individual year:
2014



`ggplot` works by building your plot piece by piece

First we need to load some data into R (link here)

```
library("readr")
gh.link = "https://raw.githubusercontent.com/"
user.repo = "sebastianbarfort/sds_summer/"
branch = "gh-pages/"
link = "data/polarization.csv"
data.link = paste0(gh.link, user.repo, branch, link)
df = read_csv(data.link)
names(df)
```

```
## [1] "x"     "year"  "party" "pct"
```

```
## [1] "x"     "year"  "party" "pct"
```

x	year	party	pct
-10	1994	Dem	0.57
-9	1994	Dem	1.60
-8	1994	Dem	1.89
-7	1994	Dem	3.49
-6	1994	Dem	3.96
-5	1994	Dem	6.56

ggplot2 BASICS

`ggplot` works by building your plot piece by piece

Then we tell ggplot what pieces of the data frame we are interested in

We create an object called `p` containing this information

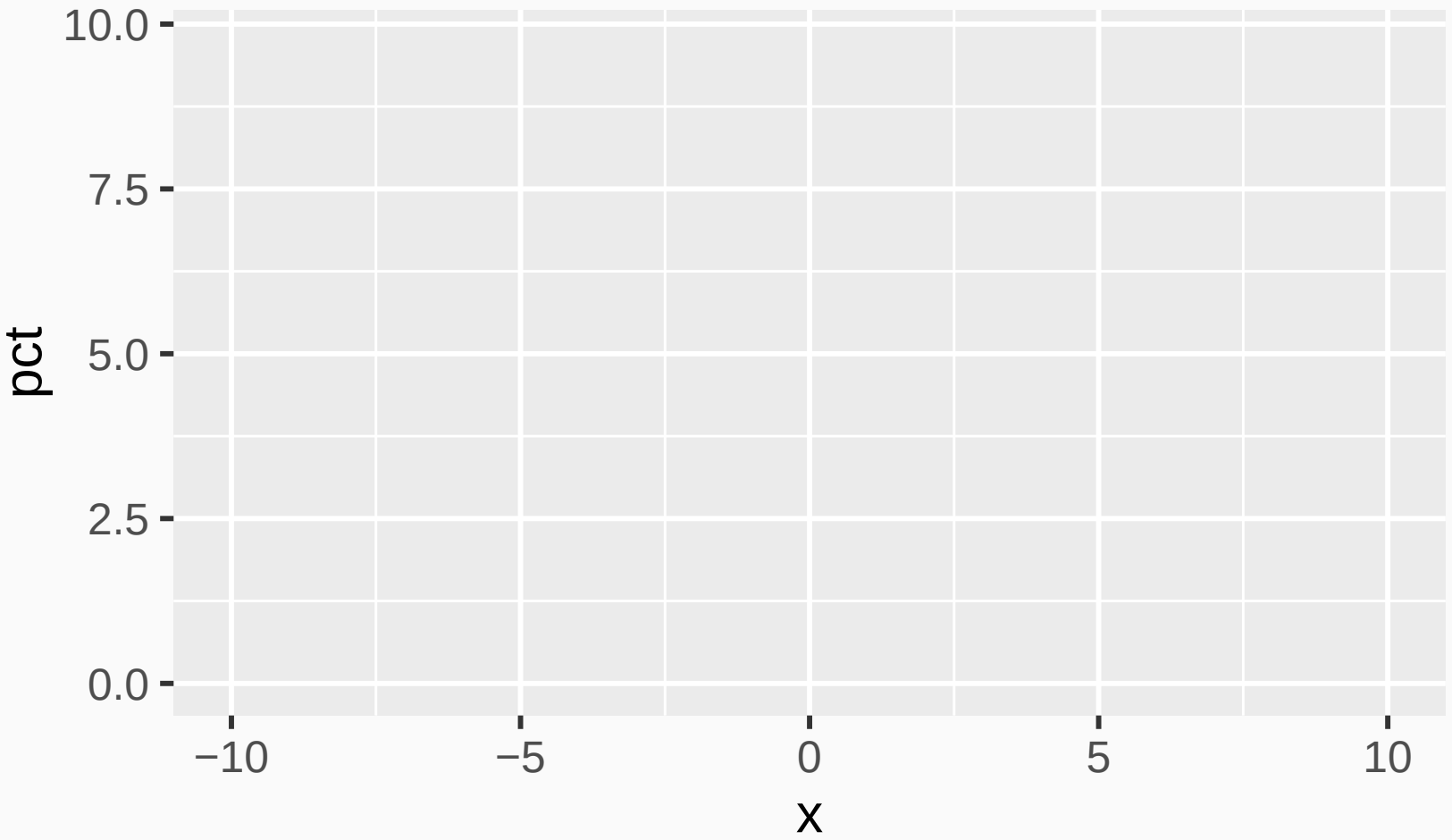
Here, `x = x` and `y = pct` say what will go on the *x* and the *y* axes

These are aesthetic mappings that connect pieces of the data to things we can actually see on a plot.

```
library("ggplot2")
p = ggplot(data = df, aes(x = x, y = pct))
```

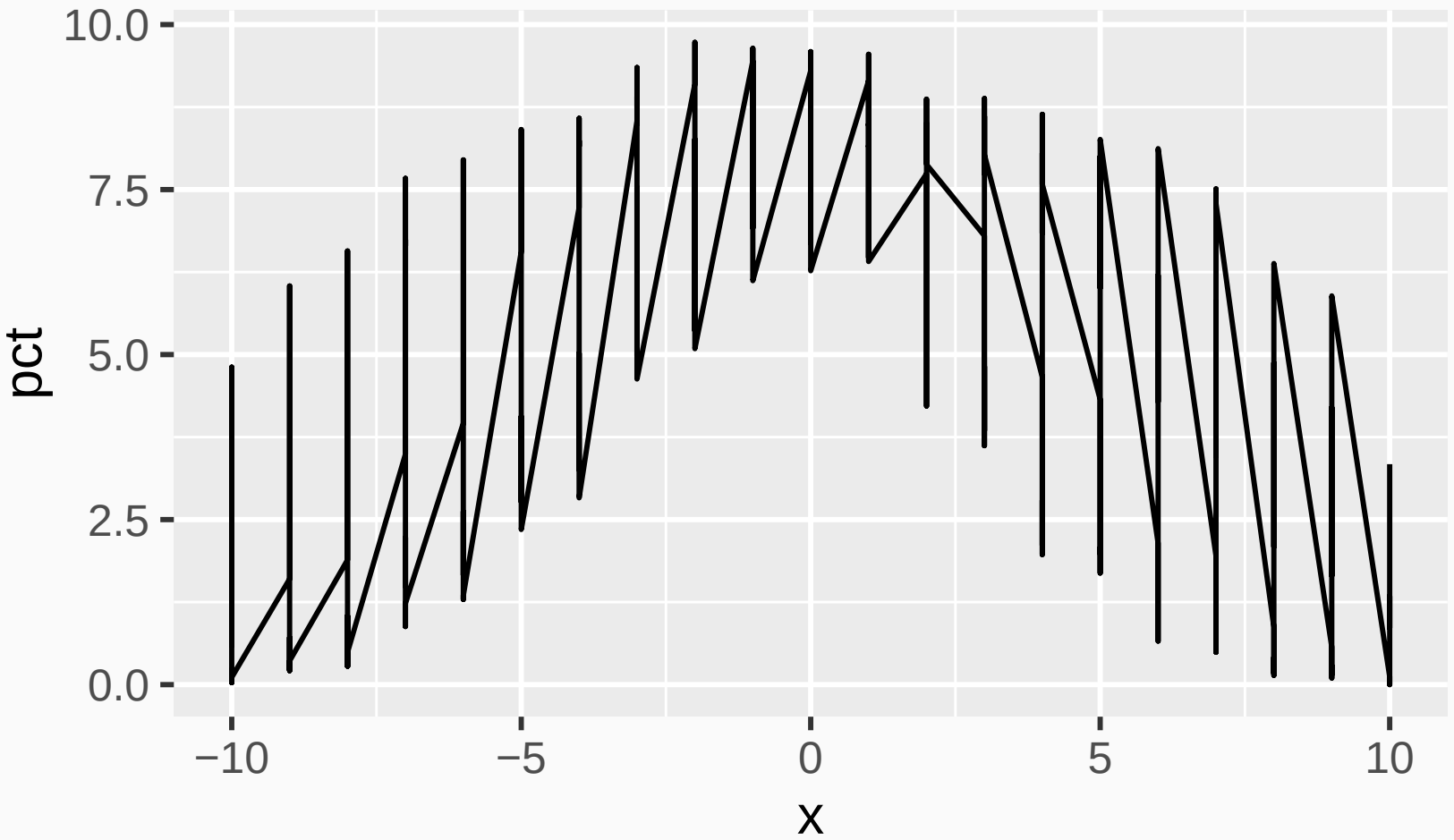
AESTHETICS, BUT NO GEOMS

The plot is so far just a frame with no actual information



LET'S ADD A LINE

```
p + geom_line()
```



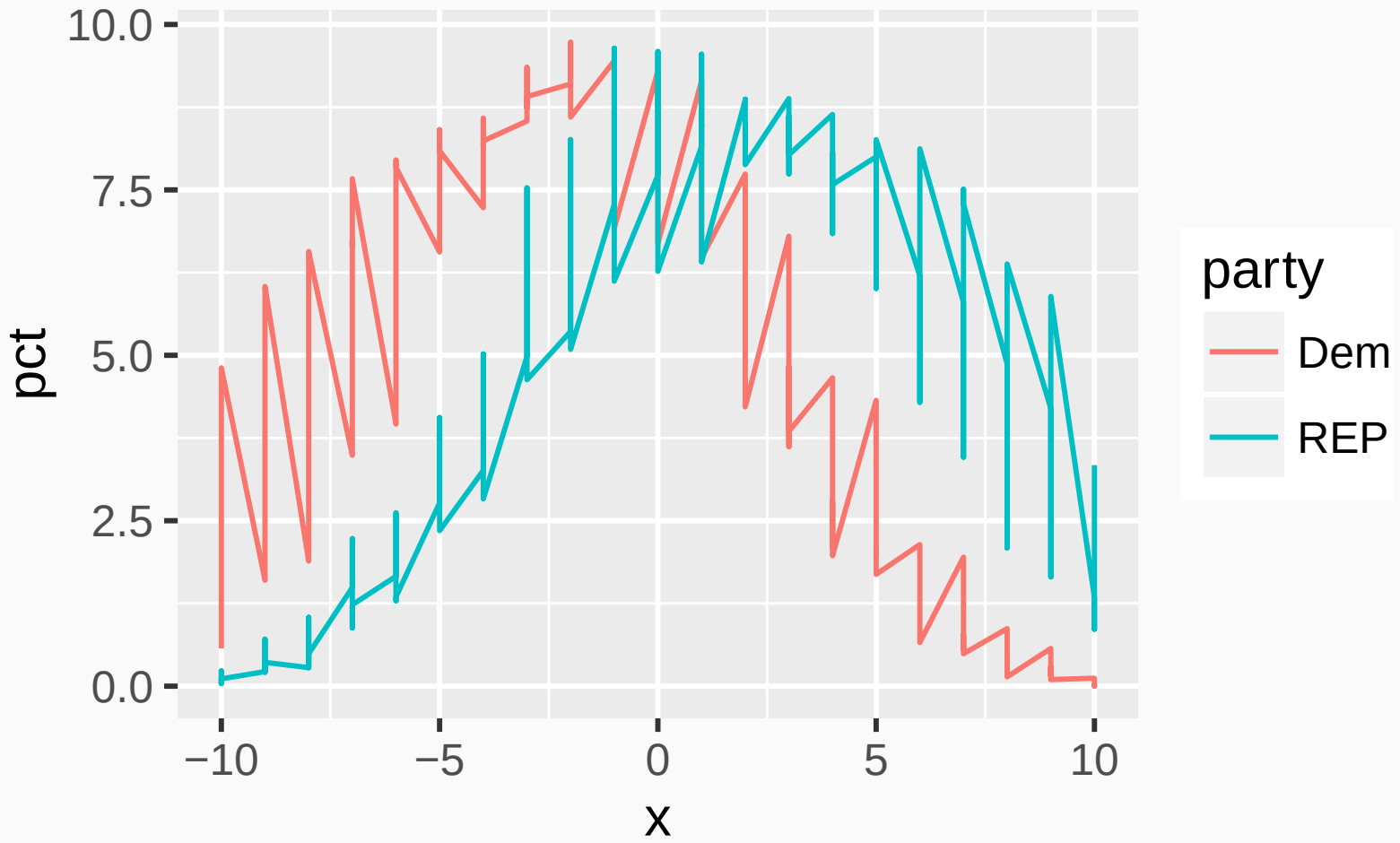
We need to inform **ggplot** that Democrats and Republicans are two separate groups.

This can be done using **fill**, **groups**, **shape**, **size** or **color**.

```
p = ggplot(data = df,
           aes(x = x, y = pct,
               fill = party, color = party))
```

TRY AGAIN

```
p + geom_line()
```

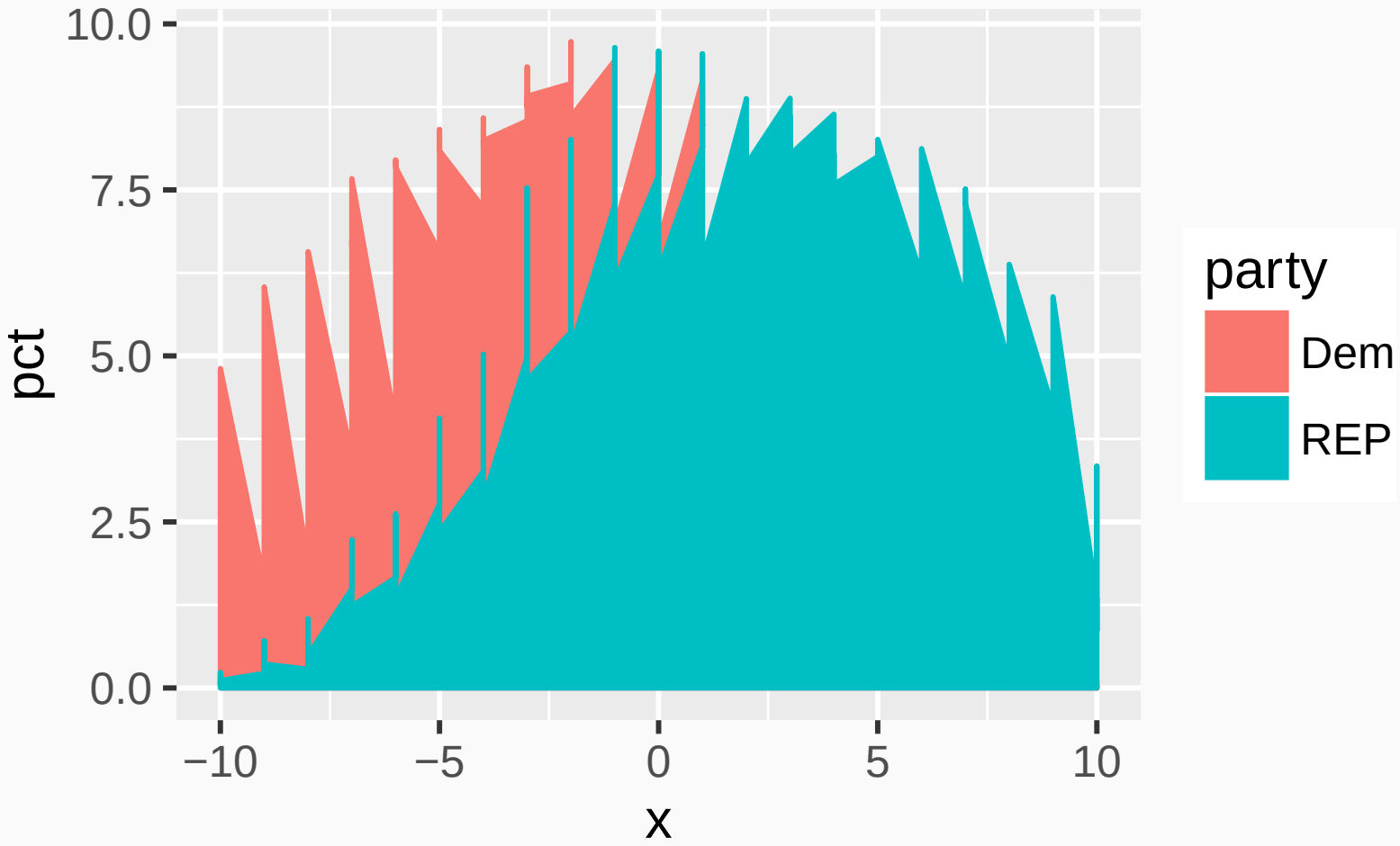


THOUGHTS

1. Shading below the line
2. Filling should be transparent
3. Custom filling
4. Axes should be modified
5. Subset by year
6. Axis title
7. Background
8. Legend

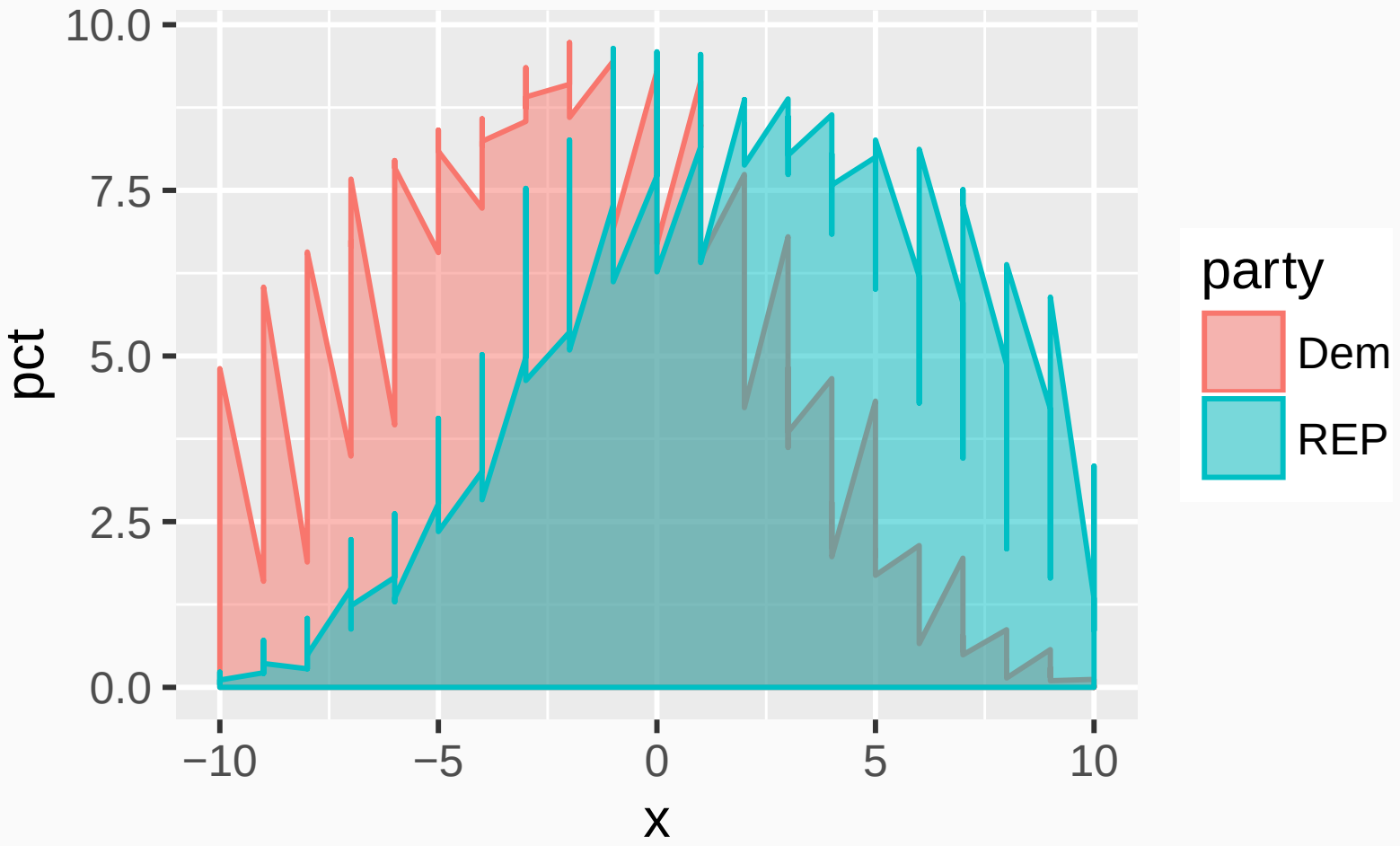
1. SHADING BELOW THE LINE

```
p + geom_ribbon(aes(ymin = 0, ymax = pct))
```

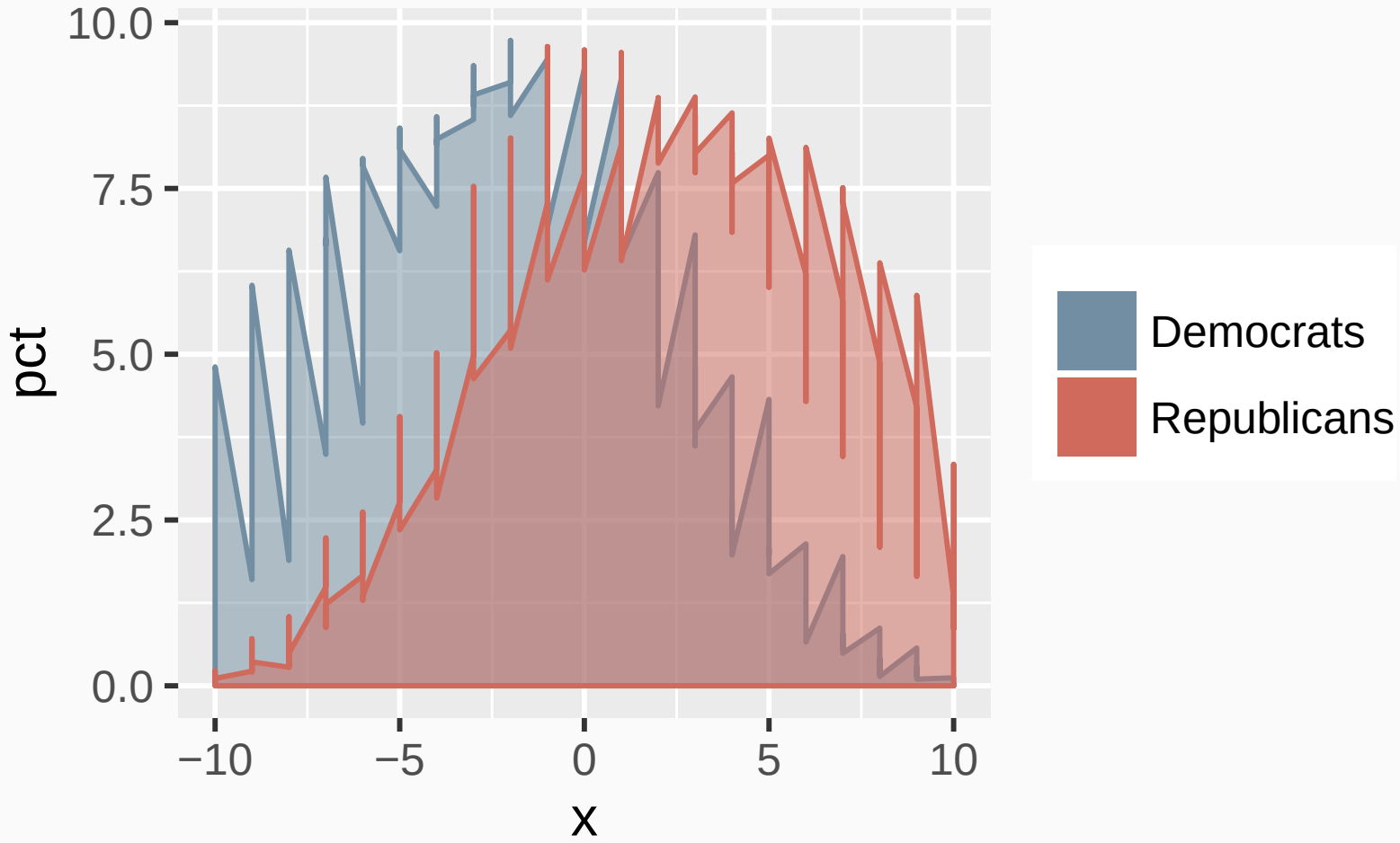


2. FILLING SHOULD BE TRANSPARENT

```
p = p + geom_ribbon(aes(ymin = 0, ymax = pct),
                    alpha = .5)
p
```

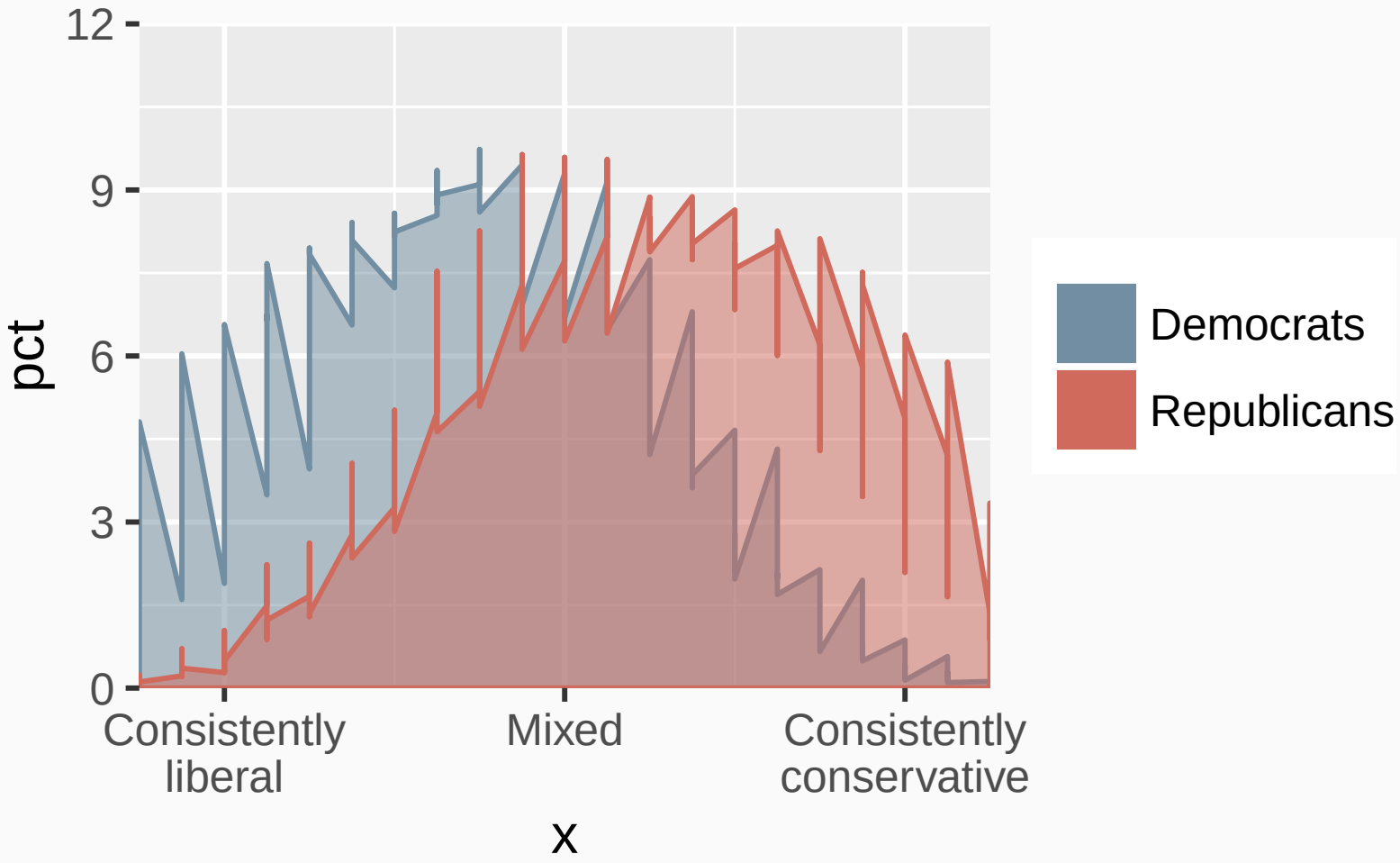


```
p = p + scale_color_manual(
  name=NULL,
  values=c(Dem="#728ea2",REP="#cf6a5d"),
  labels=c(Dem="Democrats", REP="Republicans")) +
  scale_fill_manual(
    name=NULL,
    values=c(Dem="#728ea2", REP="#cf6a5d"),
    labels=c(Dem="Democrats", REP="Republicans")) +
  guides(color="none",
         fill=guide_legend(override.aes=list(alpha=1)))
```


10.0
7.5
5.0
2.5
0.0
pct
−10
−5
0
5
10
x
Democrats
Republicans

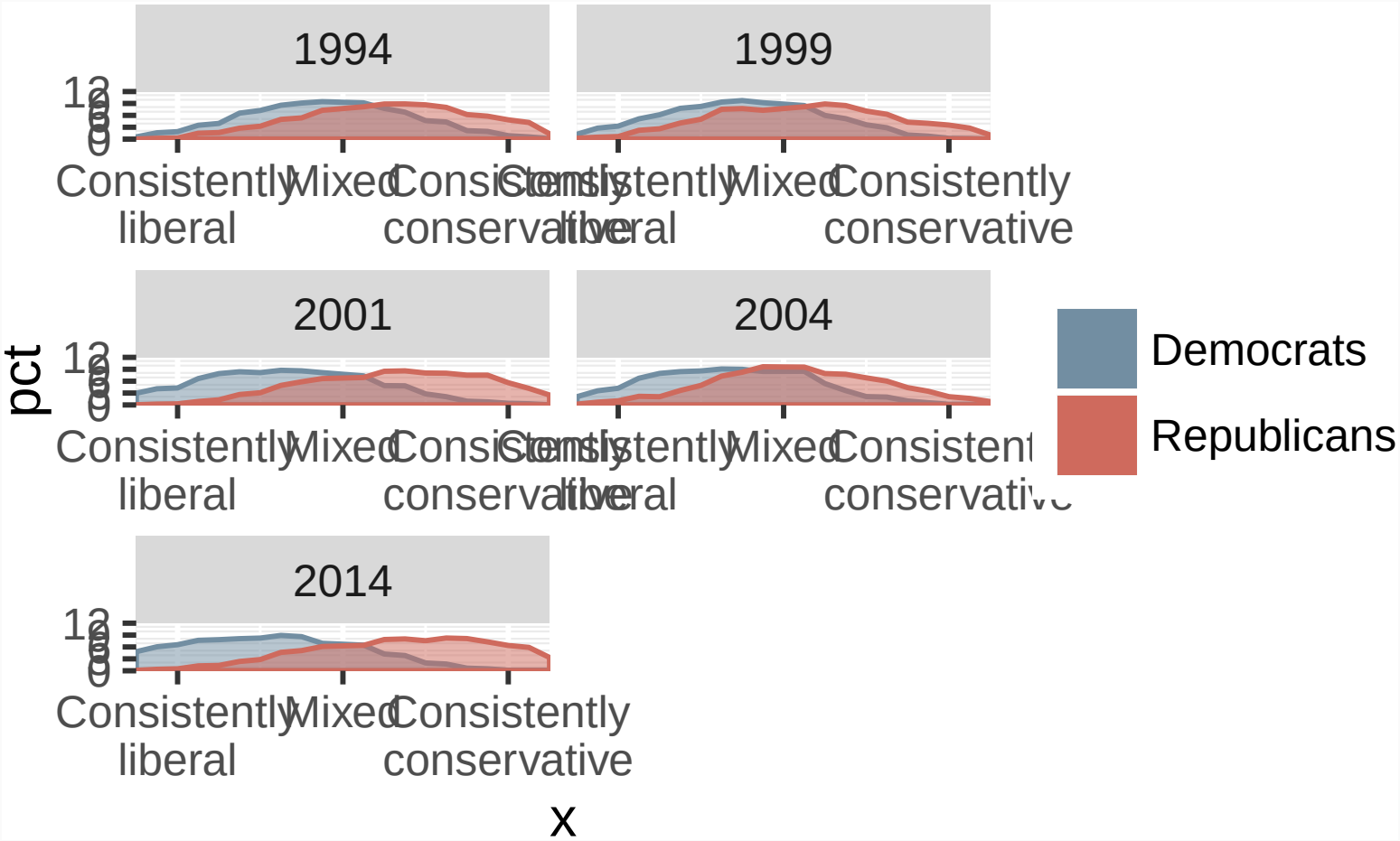
4. AXES SHOULD BE MODIFIED

```
p = p +
  scale_x_continuous(
    expand = c(0,0), breaks = c(-8, 0, 8),
    labels= c("Consistently\nliberal",
              "Mixed",
              "Consistently\nconservative")) +
  scale_y_continuous(
    expand = c(0,0), limits = c(0, 12))
```


12
9
6
3
0
pct
Consistently liberal
Mixed
Consistently conservative
x
Democrats
Republicans

5. SUBSET BY YEAR

```
p = p + facet_wrap(~ year, ncol = 2,
                   scales = "free_x")
```


1994
1999
2001
2004
2014
pct
x
Consistently liberal
Mixed
Consistently conservative
Democrats
Republicans

6. AXIS TITLE

```
p = p + labs(x = NULL,
             y = NULL,
             title = "Polarization, 1994-2014")
```

```
p = p + theme_minimal()
```

8. LEGEND

```
p = p +
  theme(legend.position = c(0.75, 0.1)) +
  theme(legend.direction = "horizontal") +
  theme(axis.text.y = element_blank())
```

OUTPUT

The output can be seen here.

VISUALIZING POLLS

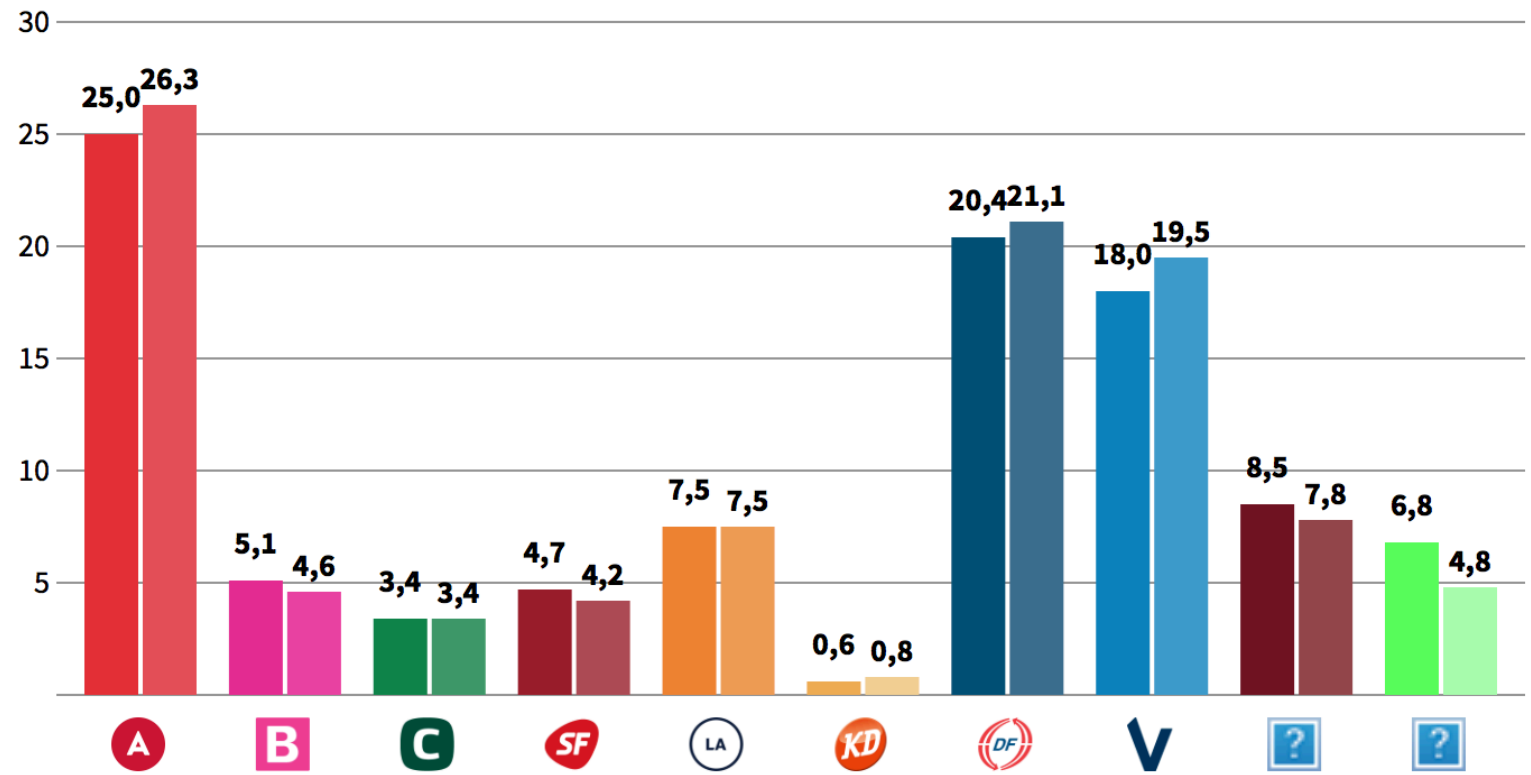
How do you visualize polls?

New polls generate a lot of news stories

But much of it is probably noise

How to deal with this visually?

VÆLG MÅLING **VÆGTET GNS. 7. jul. 2016**SAMMENLIGNET MED **VALGET 18. jun. 2015**



ALTERNATIVE

A poll is not really a number

It's a distribution

Usually, that distribution is normal

So let's draw the distribution

READ DATA

```
library("readr")
gh.link = "https://raw.githubusercontent.com/"
user.repo = "sebastianbarfort/sds_summer/"
branch = "gh-pages/"
link = "data/polls_tidy.csv"
data.link = paste0(gh.link, user.repo, branch, link)
df = read_csv(data.link)
```

First five rows/columns

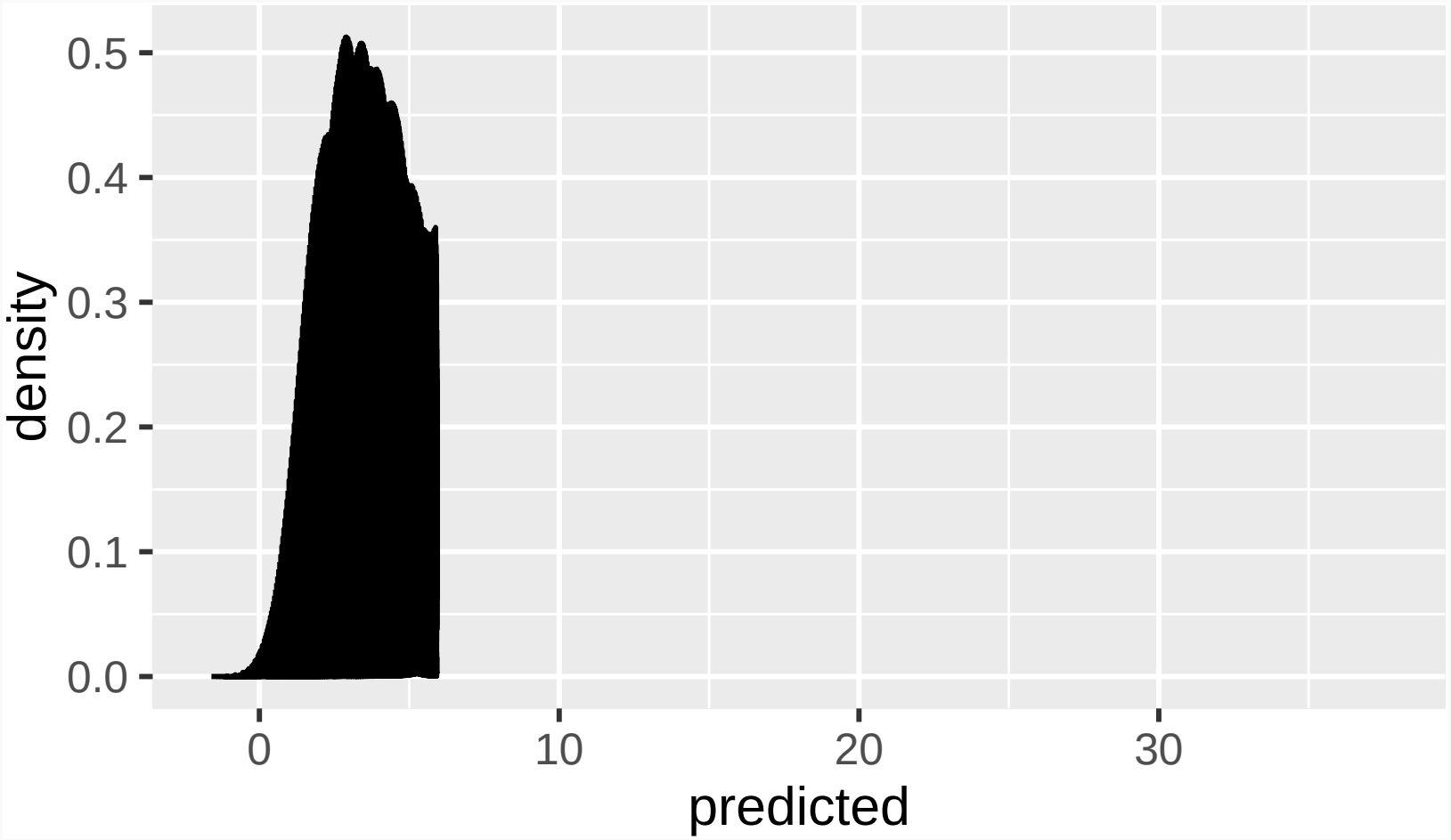
predicted	density	estimate	ci	Parti
12.83318	6.60e-06	23.6	2.39159	Socialdemokraterne
12.92999	7.90e-06	23.6	2.39159	Socialdemokraterne
13.02681	9.50e-06	23.6	2.39159	Socialdemokraterne
13.12363	1.14e-05	23.6	2.39159	Socialdemokraterne
13.22045	1.36e-05	23.6	2.39159	Socialdemokraterne

Inspect the dataset

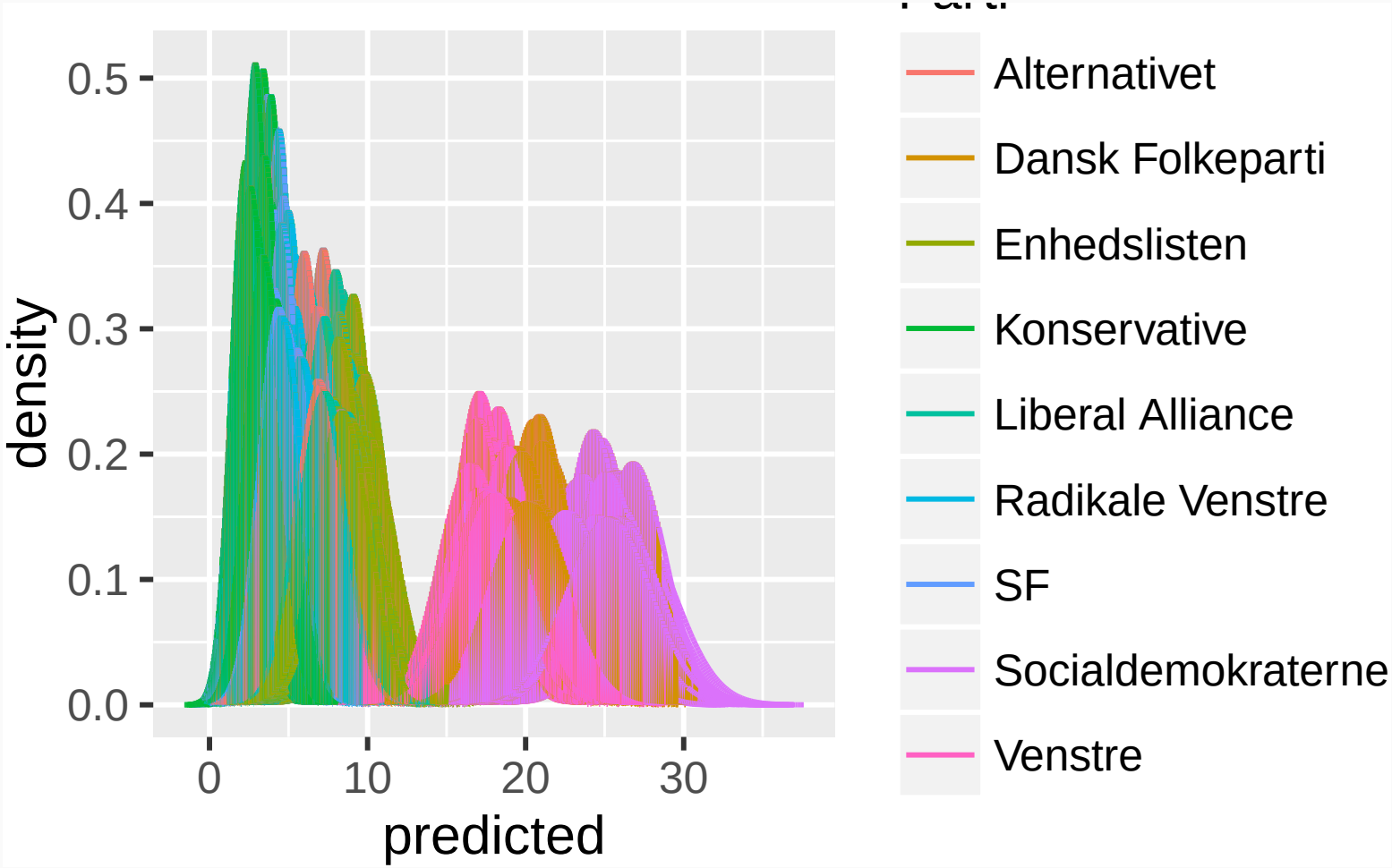
What do you think the variables `predicted` and `density` mean?

```
p = ggplot(df, aes(x = predicted, y = density))
```

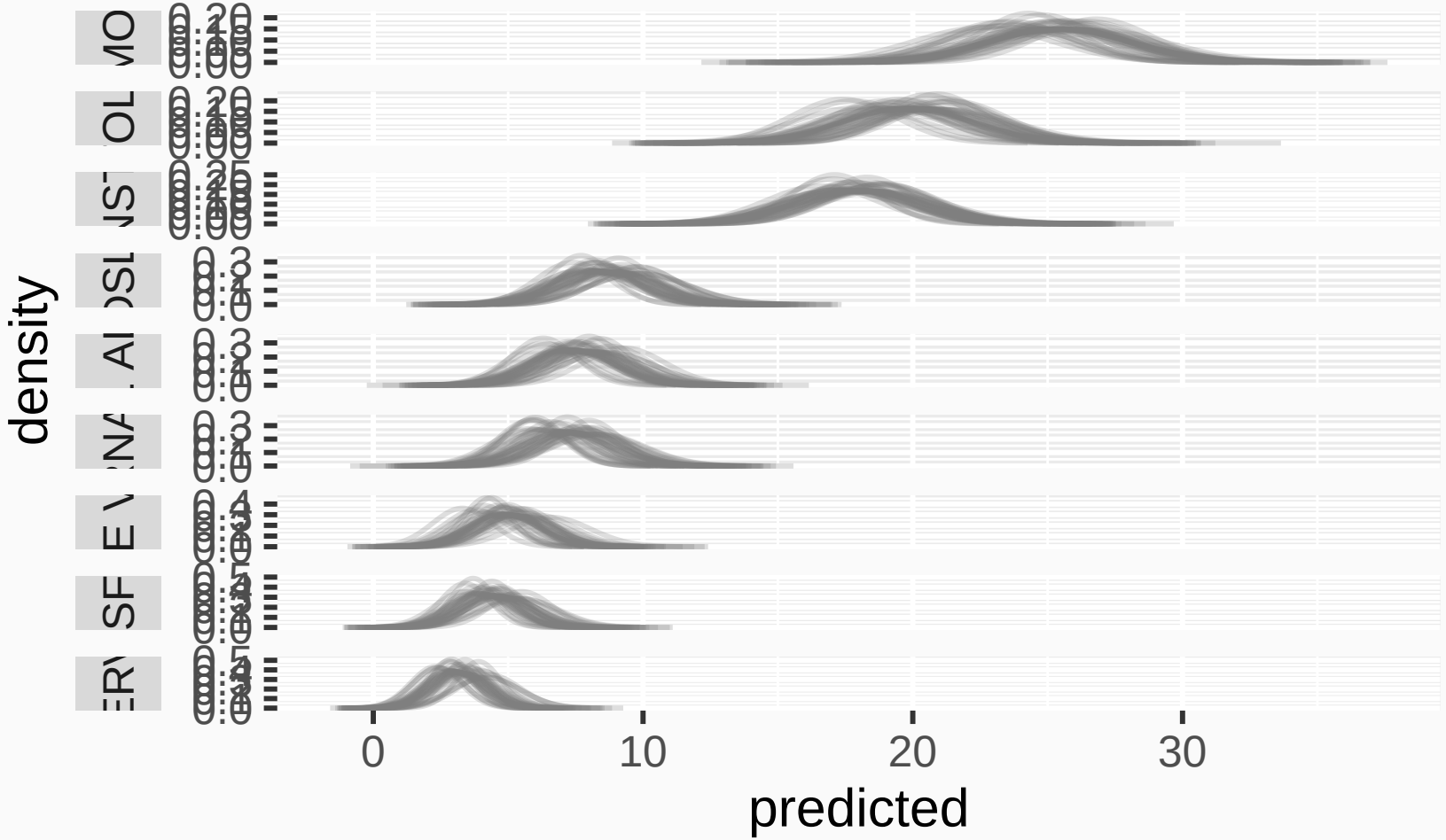
```
p + geom_line()
```



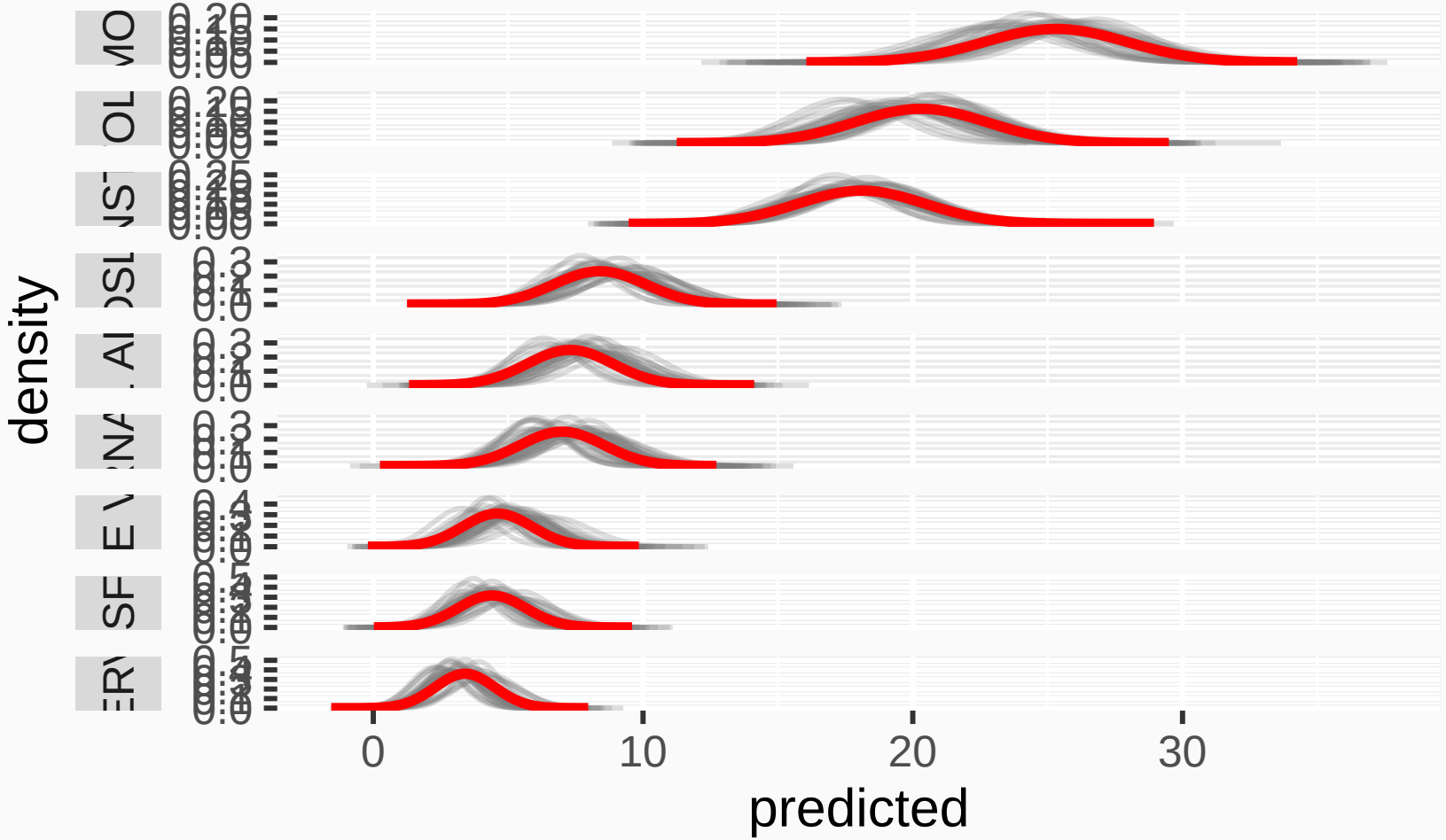
```
p = ggplot(df, aes(x = predicted, y = density,
                   group = group, color = Parti))
p + geom_line()
```



```
p = ggplot(df, aes(x = predicted, y = density,
                   group = group, color = Parti))
p = p + geom_line() +
  facet_grid(party.ord ~ ., scales = "free_y", switch =
```

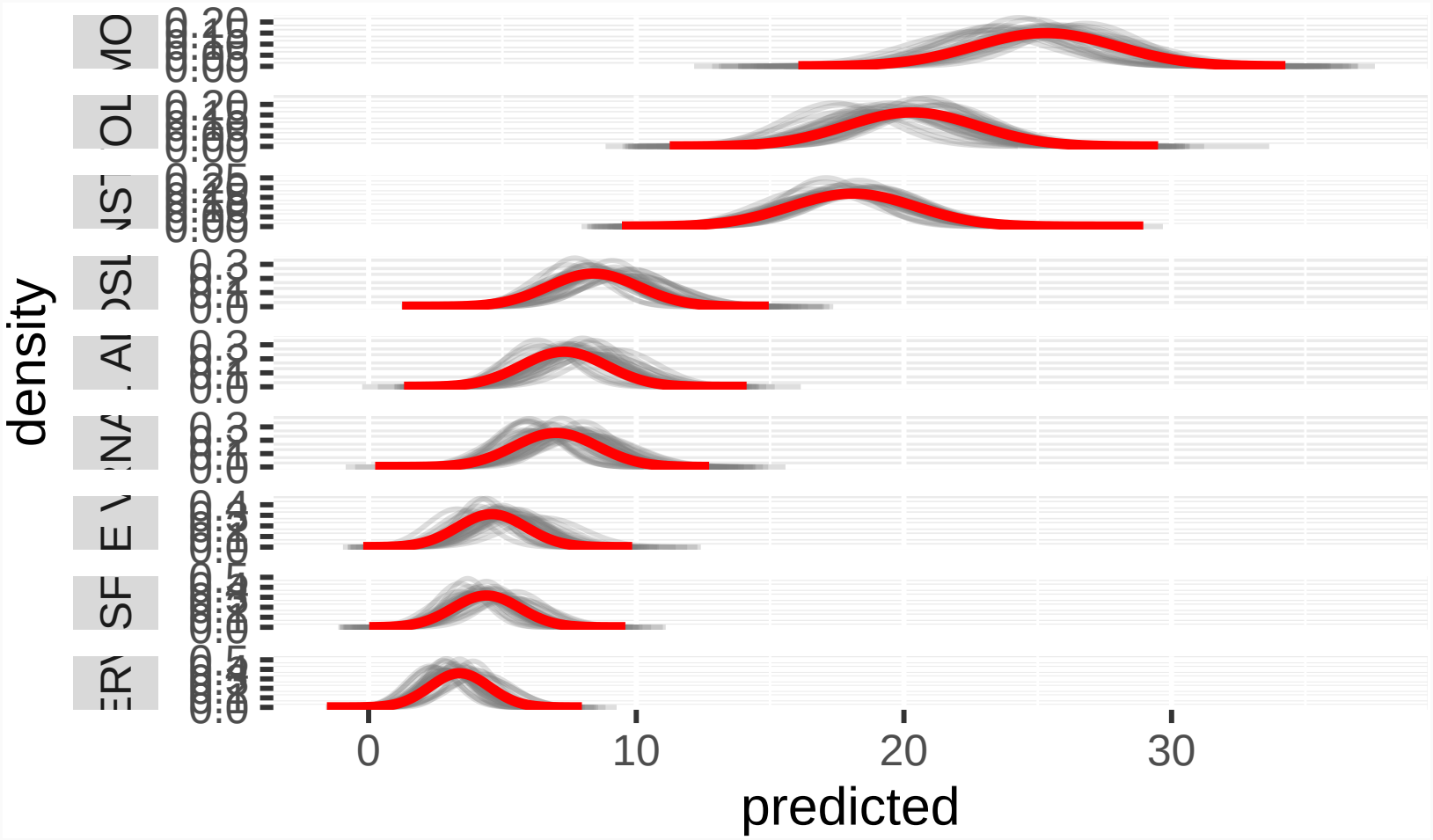

density
predicted
0
10
20
30

```
df.1 = subset(df, date != max(date))
df.2 = subset(df, date == max(date))
p = ggplot()
p = p + geom_line(data = df.1,
                  aes(x = predicted, y = density,
                      group = group),
                  color = "grey50", alpha = .25) +
  facet_grid(party.ord ~ .,
                 scales = "free_y", switch = "y")
```


density
0
10
20
30
predicted

ADD LATEST POLL

```
p = p + geom_line(data = df.2,
                  aes(x = predicted, y = density,
                      group = group),
                  colour = "red", size = 1)
```


density
0
10
20
30
predicted

DETAILS I

```
p = p + scale_x_continuous(
  breaks = scales::pretty_breaks(10)) +
  theme_minimal() +
  theme(strip.text.y = element_text(angle = 180,
                                    face = "bold",
                                    vjust = 0.75,
                                    hjust = 1),
        axis.text.y = element_blank()) +
  labs(y = NULL, x = NULL)
```

OUTPUT

The output can be seen here.

Stephen Curry's 3-Point Record in Context: Off the Charts

FIREARM PRICES

Look at this visualization from Bob Rudis.

Data here

LOAD DATA

```
library("readr")
gh.link = "https://raw.githubusercontent.com/"
user.repo = "sebastianbarfort/sds_summer/"
branch = "gh-pages/"
link = "data/armslist.csv"
data.link = paste0(gh.link, user.repo, branch, link)
df = read_csv(data.link)
names(df)
```

```
##  [1] "url"          "post_id"      "title"          "li
##  [5] "price"        "location"     "city"           "st
##  [9] "description"  "registered"   "category"       "ma
## [13] "caliber"      "action"       "firearm_type"   "pa
## [17] "img"
```

CREATE GGPLOT

```
p = ggplot(data = df, aes(x = price))
```

YOUR TURN

Discuss in groups

1. What is interesting about these data?
2. How would you visualize the distribution of the `price` variables
3. Search the `ggplot2` documentation here and create a visualization
4. Discuss how you would improve your visualization
5. Look at Bob Rudis' plots. Would you change anything? Do you think they work?

Maps

MAKING MAPS

There are many ways to make maps in R

1. **easy**: use an existing package
2. **hard**: learn how to work with shapefiles (we don't have time for this today, but I strongly recommend reading these notes on the topic)

MAP PACKAGES

There are many useful packages for making maps in R

- `maps`: all kinds of maps
- `ggcounty`: generate United States county maps
- `ggmap`: extends `ggplot2` for maps
- `mapDK`: maps of Denmark