Informatics 143

Information Visualization

Lecture 7

These course materials are based on books from Claus O. Wilke, Kieran Healy, Edward R. Tufte, Alberto Cairo, Colin Ware, Tamara Munzner, and others.
Powerpoint theme by Prof. André van der Hoek.

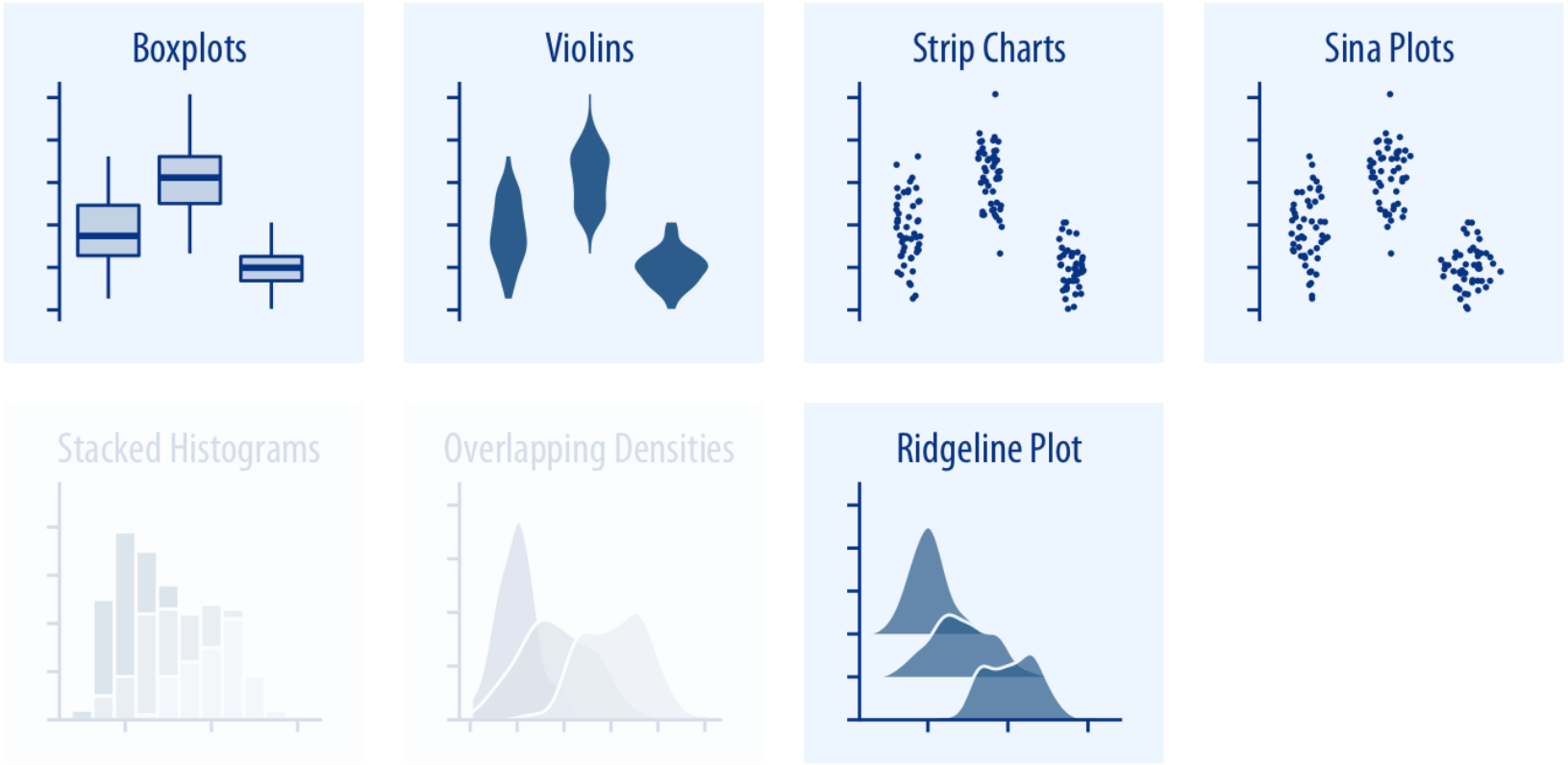
Visualization of Distributions: multiple distributions

- What if you need to visualize more than one distribution at once, where **now** *more* is a large number?

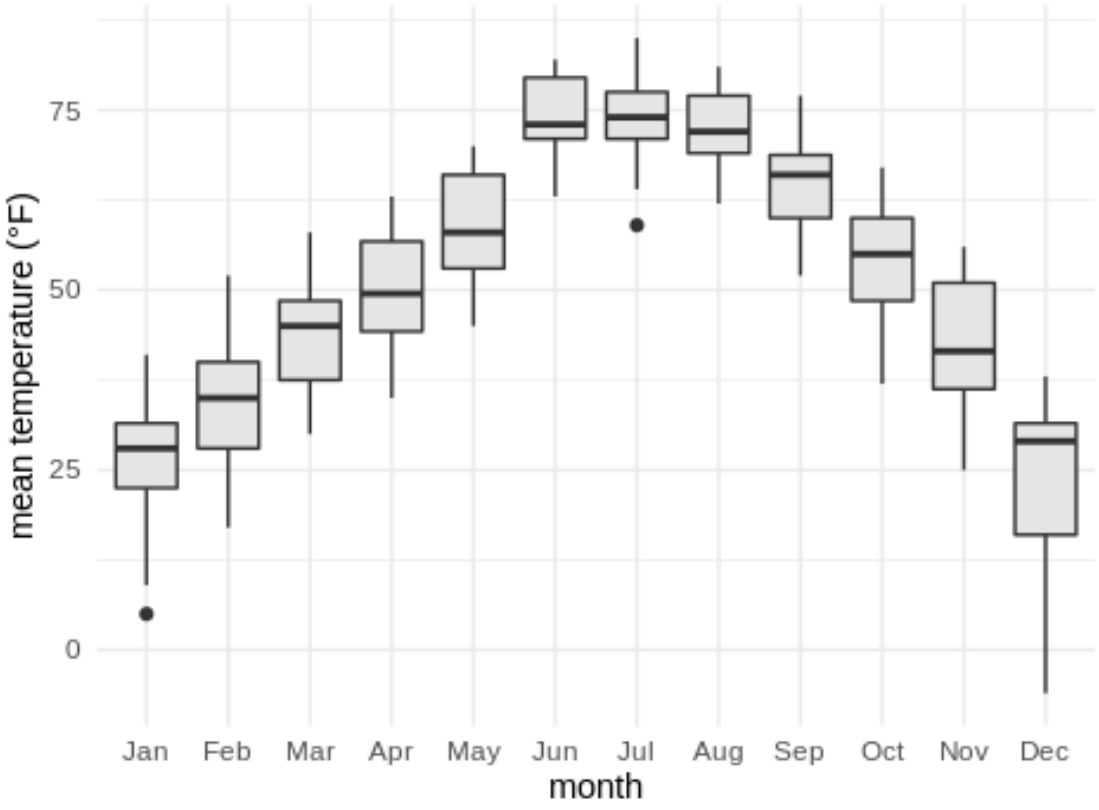
- What to do in these cases?
  - E.g. how temperature varies along different months *and* within each month?

Visualization of Distributions: multiple distributions

- Some options...



Visualization of Distributions: Boxplot

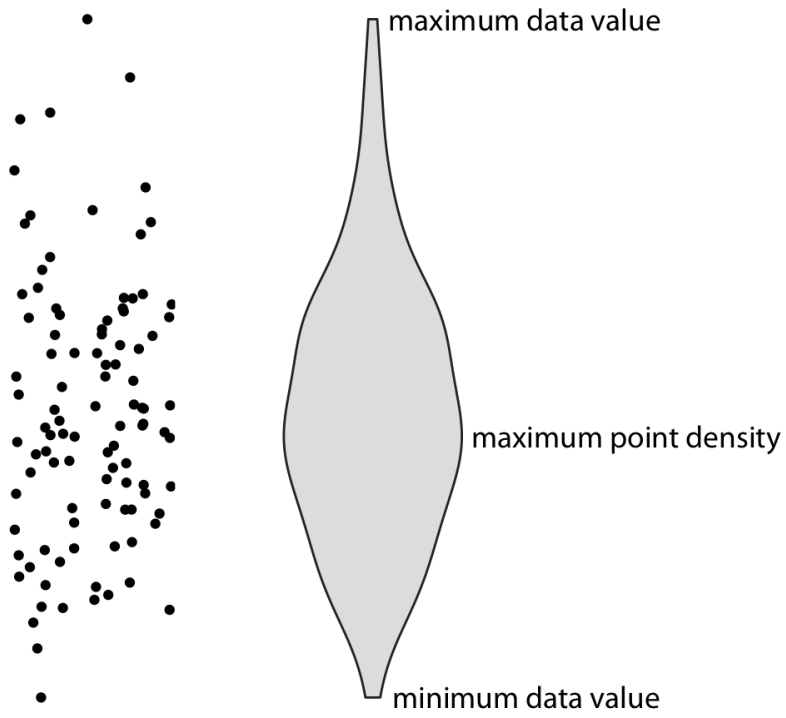


Visualization of Distributions: Boxplot

- Boxplots
  - Invented betweeen 50-70's by Mary Eleanor Spear and John Tukey
  - Very simple to read
  - Very simple to draw **by hand**
    - Very important aspect until recent times...
  - **But they are still *hiding* information.**

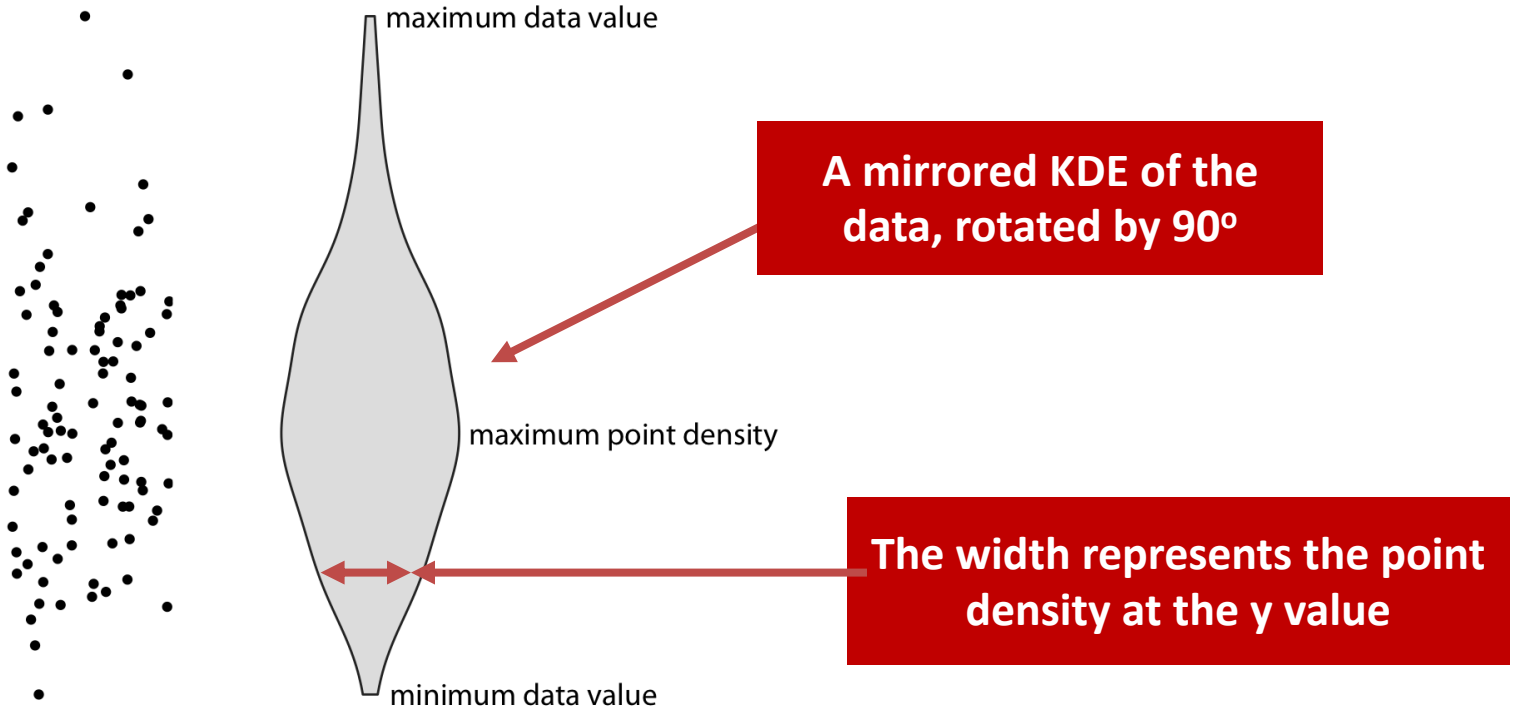
Visualization of Distributions: Violin plot

- Violin plots
  - Thanks to the appearance of computers, now we can draw the entire distribution!



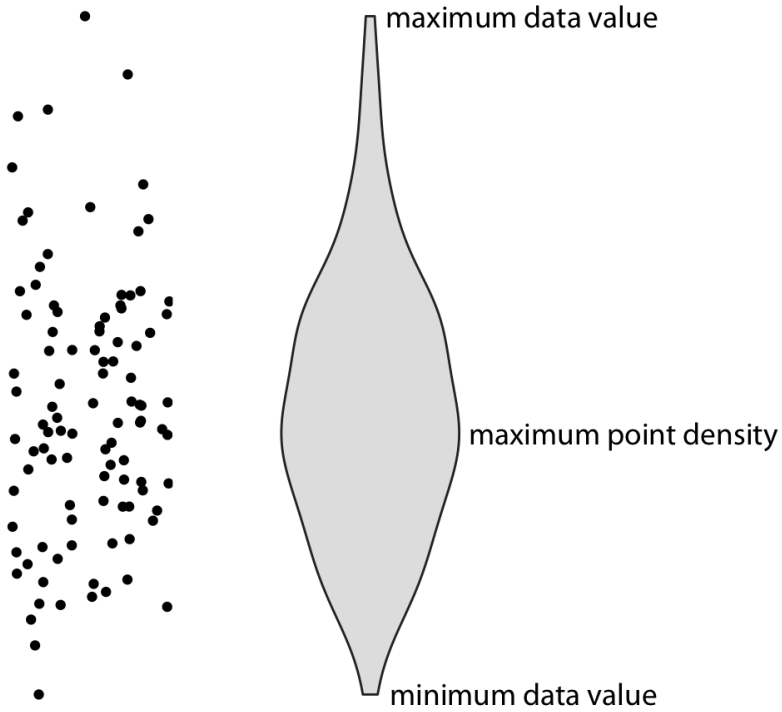
Visualization of Distributions: Violin plot

- Violin plots
  - Thanks to the appearance of computers, now we can draw the entire distribution!



Visualization of Distributions: Violin plot

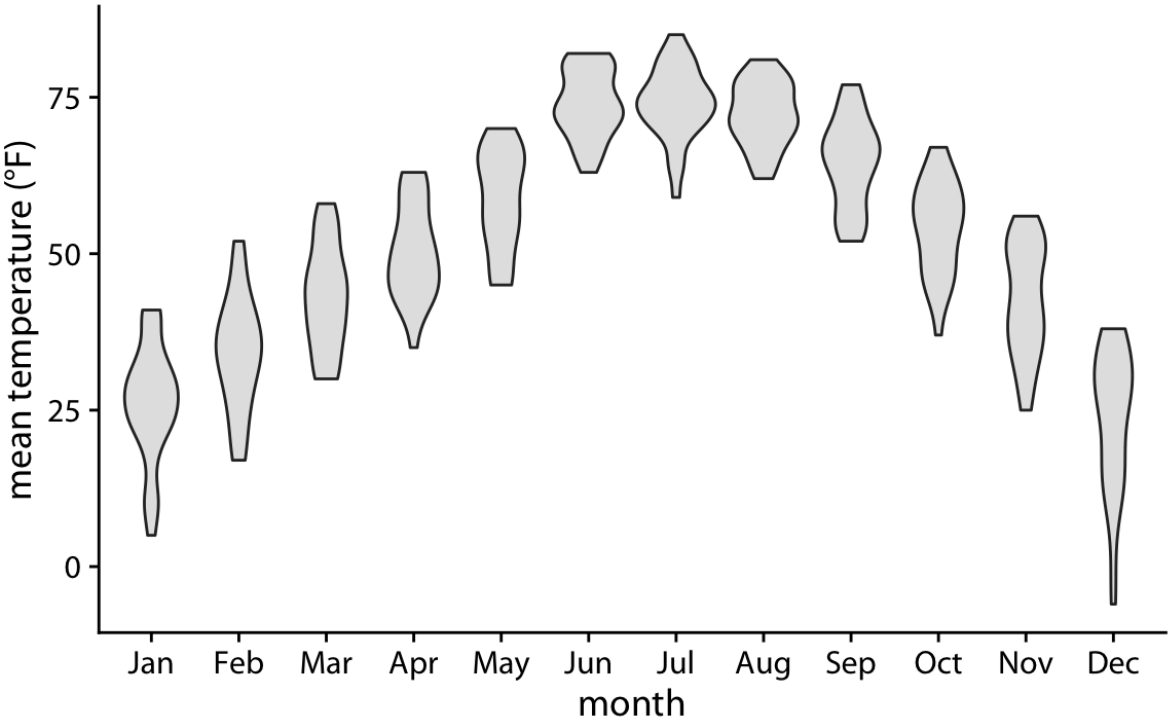
- Violin plots
  - Thanks to the appearance of computers, now we can draw the entire distribution!



Warning: check if you have enough data points in each group.

Visualization of Distributions: Violin plot

- Violin plot of the temperature dataset



Data source: Weather Underground

Visualization of Distributions: Violin plot

- How to do it in ggplot2?
  - Use geom_violin()

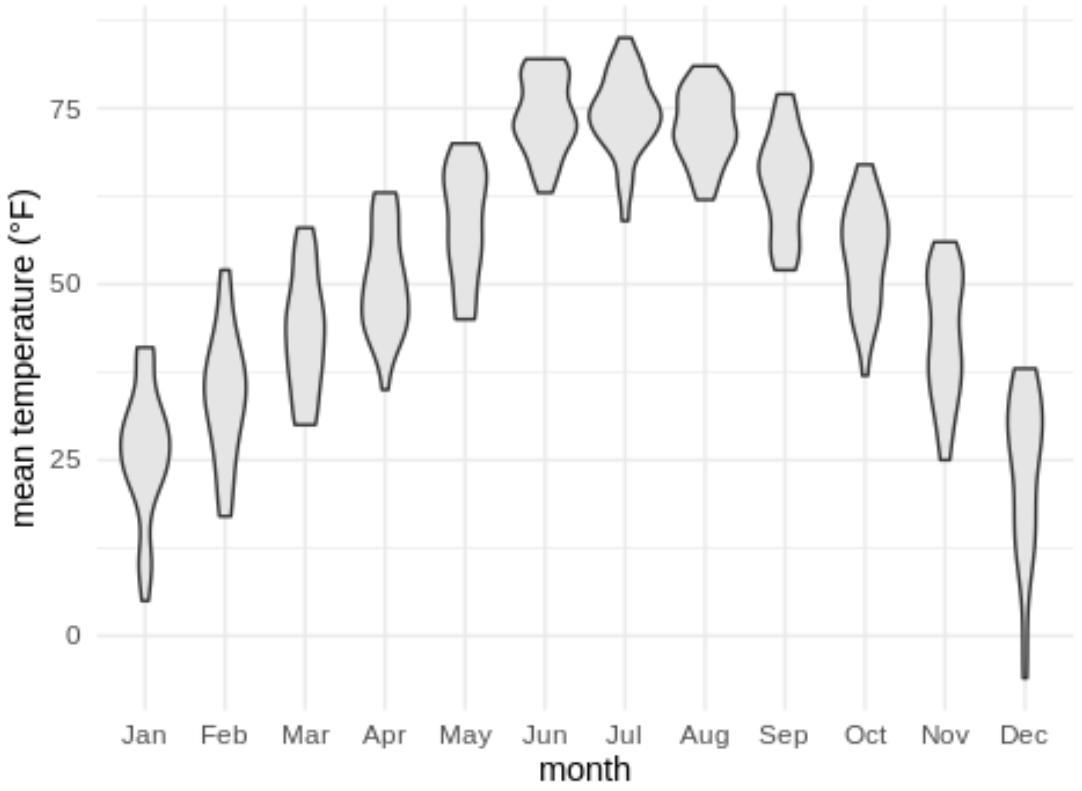
```
lincoln_df <-read.csv(
"https://www.ics.uci.edu/~algol/teaching/informatics143w2021/lincoln_df.csv")

lincoln_df$month_short <- factor(lincoln_df$month_short,
                                 levels = unique(lincoln_df$month_short))

ggplot(lincoln_df, aes(x = month_short, y = Mean.Temperature..F.)) +
  geom_violin(fill = 'grey90') +
  xlab("month") +
  ylab("mean temperature (°F)") +
  theme_minimal()
```

Visualization of Distributions: Violin plot

- How to do it in ggplot2?
  - Use geom_violin()

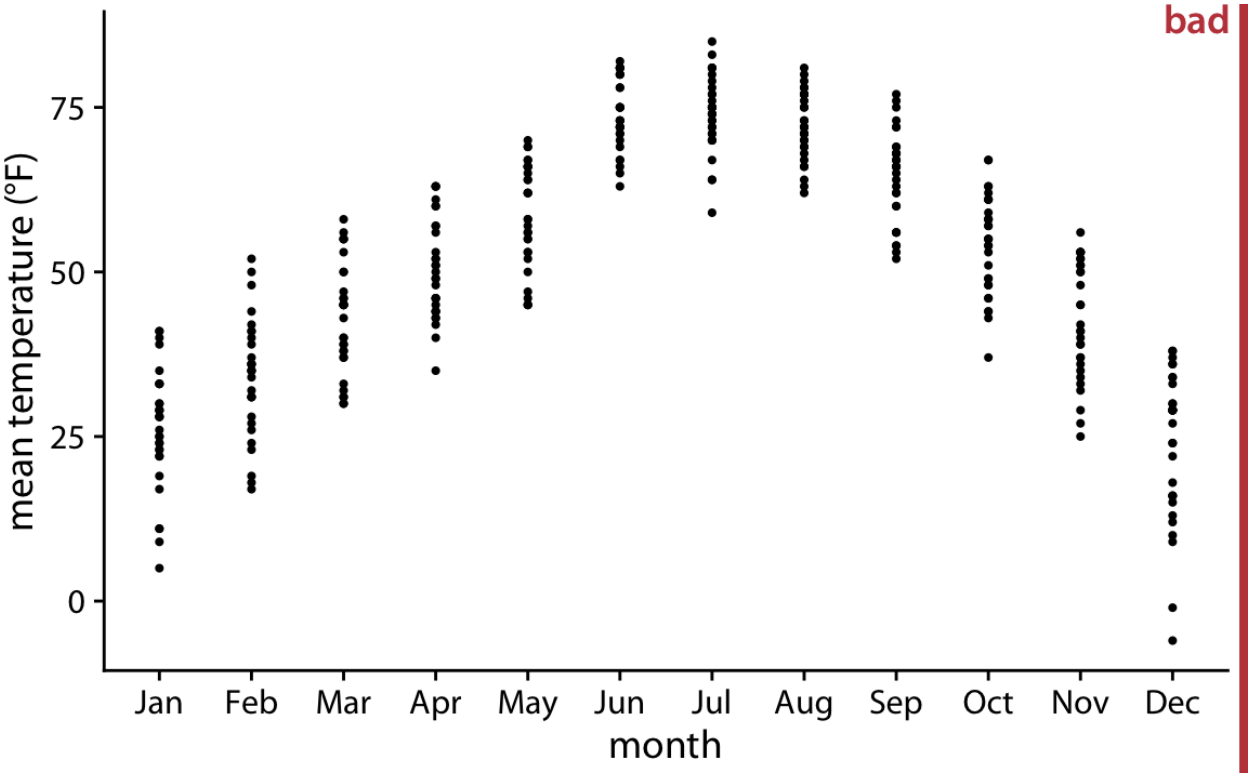


Visualization of Distributions: Violin plot

- Violin plots
  - Invented in 1998 by Hintze and Nelson
  - Very simple to read
  - Require computers to produce correctly
  - Can be misleading if you have *small* and *sparse* datasets

Visualization of Distributions: Strip charts

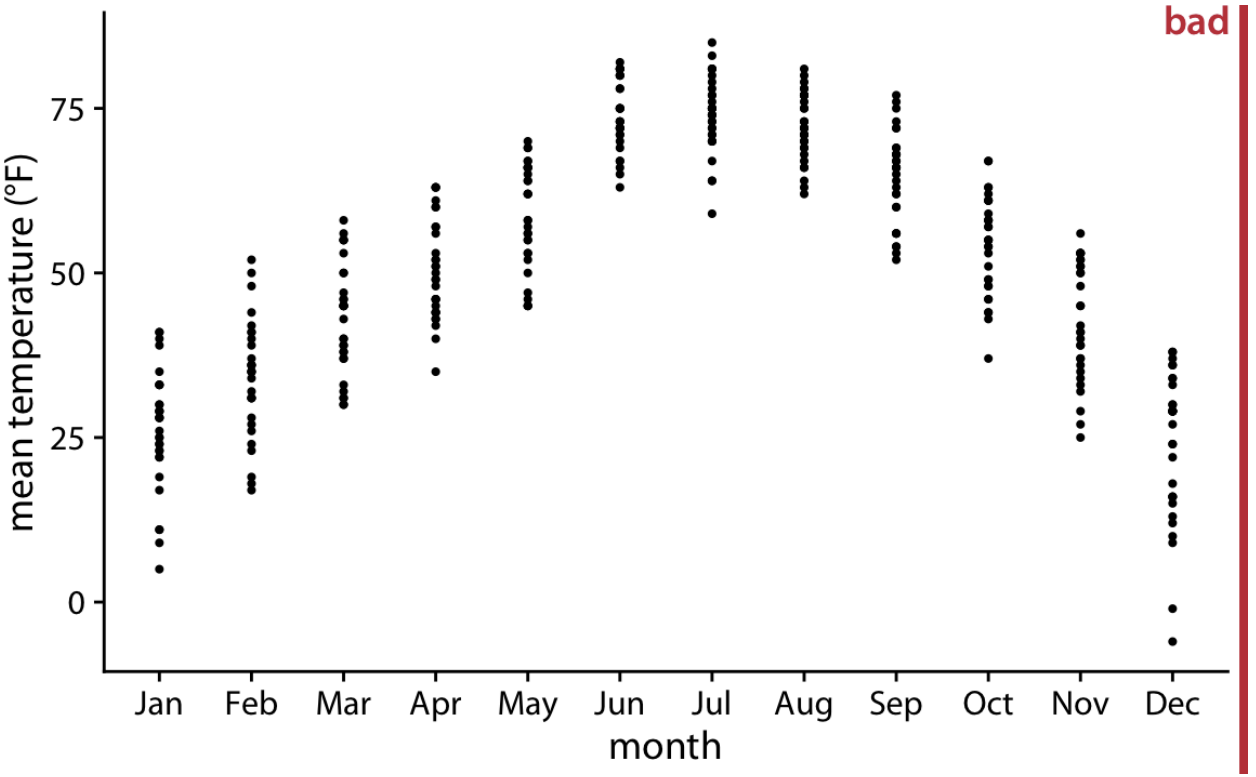
- Can't we just plot all the individual points?



Data source: Weather Underground

Visualization of Distributions: Strip charts

- Can’t we just plot all the individual points?

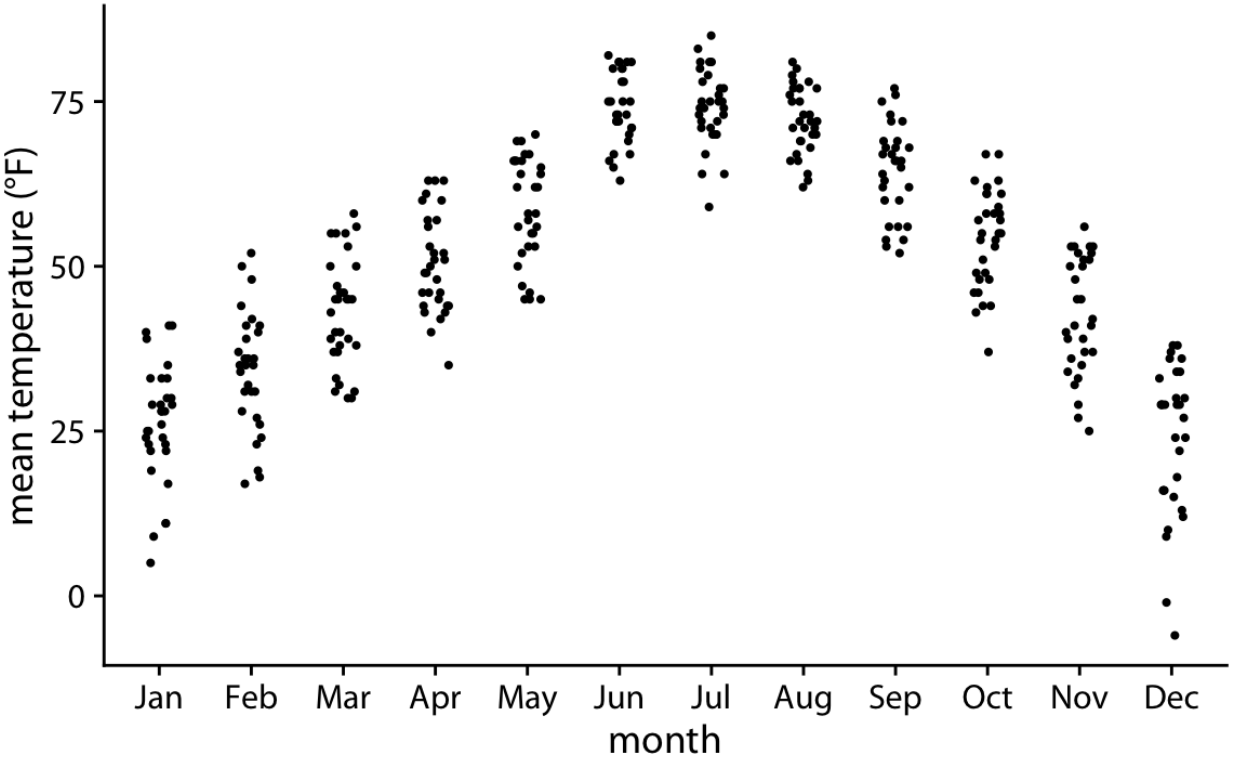


Which temperatures are the most common in each month?

Data source: Weather Underground

Visualization of Distributions: Strip charts

- Use jittering to avoid overplotting



Data source: Weather Underground

Visualization of Distributions: Strip charts

- How to do it in ggplot2?
  - Use geom_point() and add some jitter

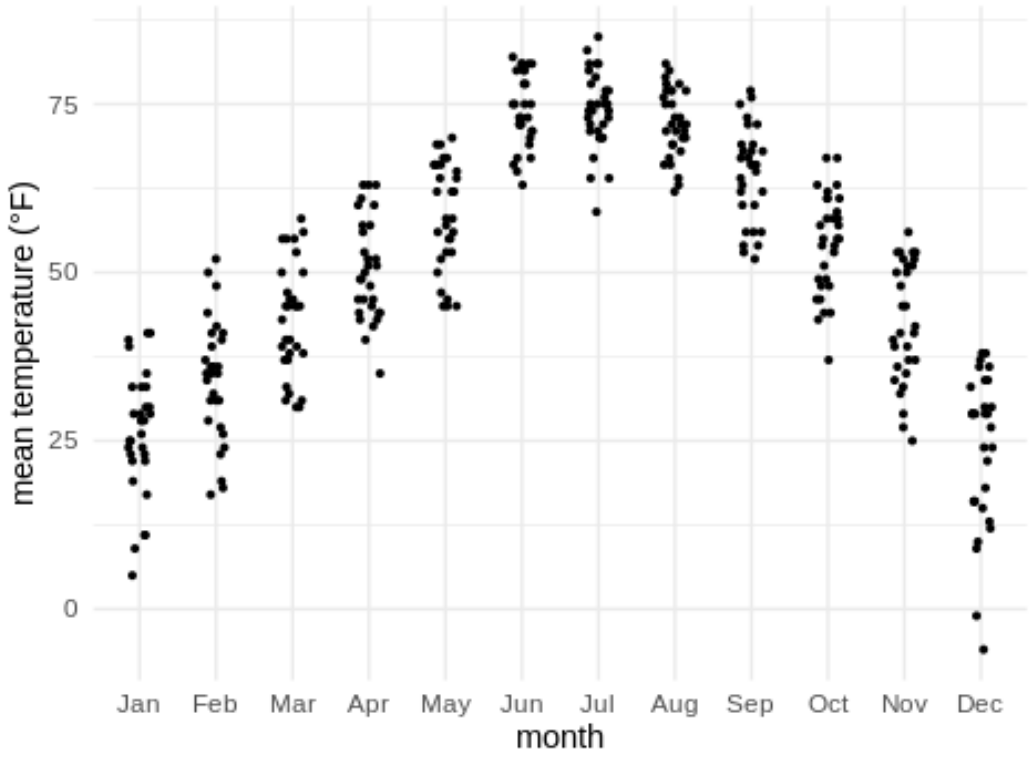
```
lincoln_df <-read.csv(
"https://www.ics.uci.edu/~algol/teaching/informatics143w2021/lincoln_df.csv")

lincoln_df$month_short <- factor(lincoln_df$month_short,
                                 levels = unique(lincoln_df$month_short))

ggplot(lincoln_df, aes(x = month_short, y = Mean.Temperature..F.)) +
  geom_point(
         position = position_jitter(width = .15, height = 0, seed = 320),
         size = 0.75) +
  xlab("month") +
  ylab("mean temperature (°F)") +
  theme_minimal()
```

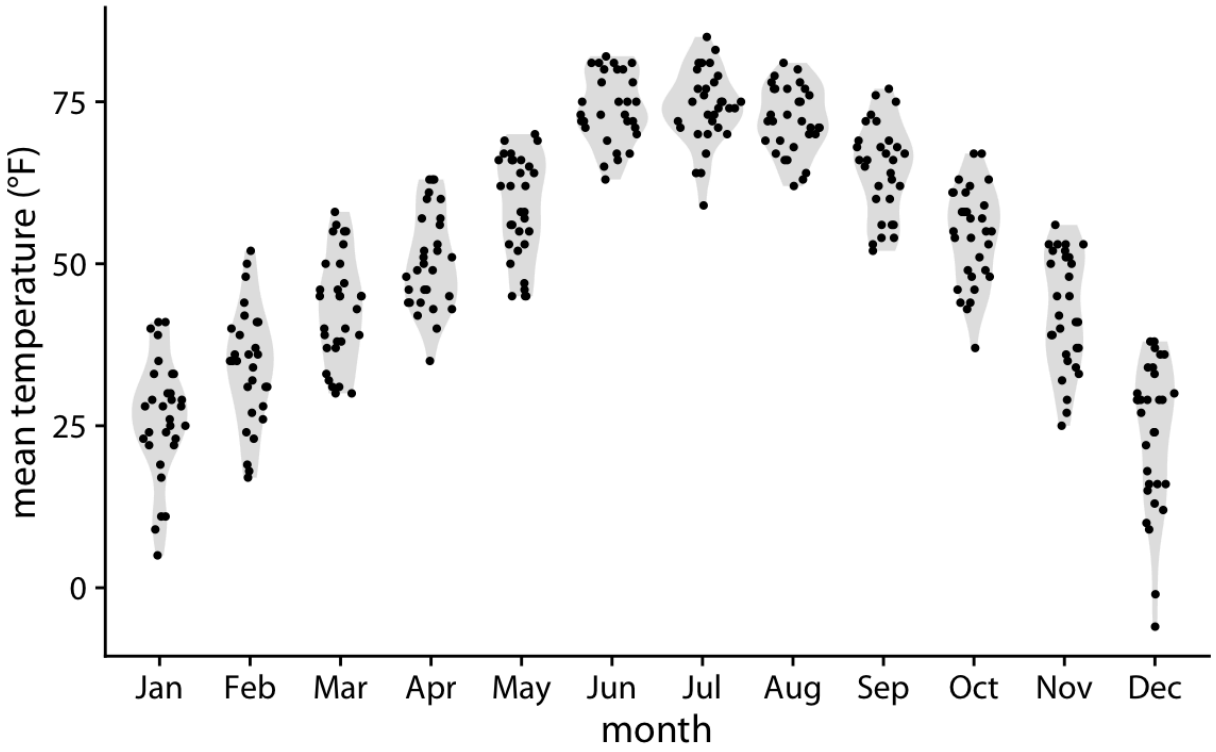
Visualization of Distributions: Strip charts

- How to do it in ggplot2?
  - Use geom_point() and add some jitter



Visualization of Distributions: Sina plot

- Can't we combine both?
  - Sina plot. Very recently invented (Sidiropoulos et al. 2018)



Data source: Weather Underground

Visualization of Distributions: Sina plots

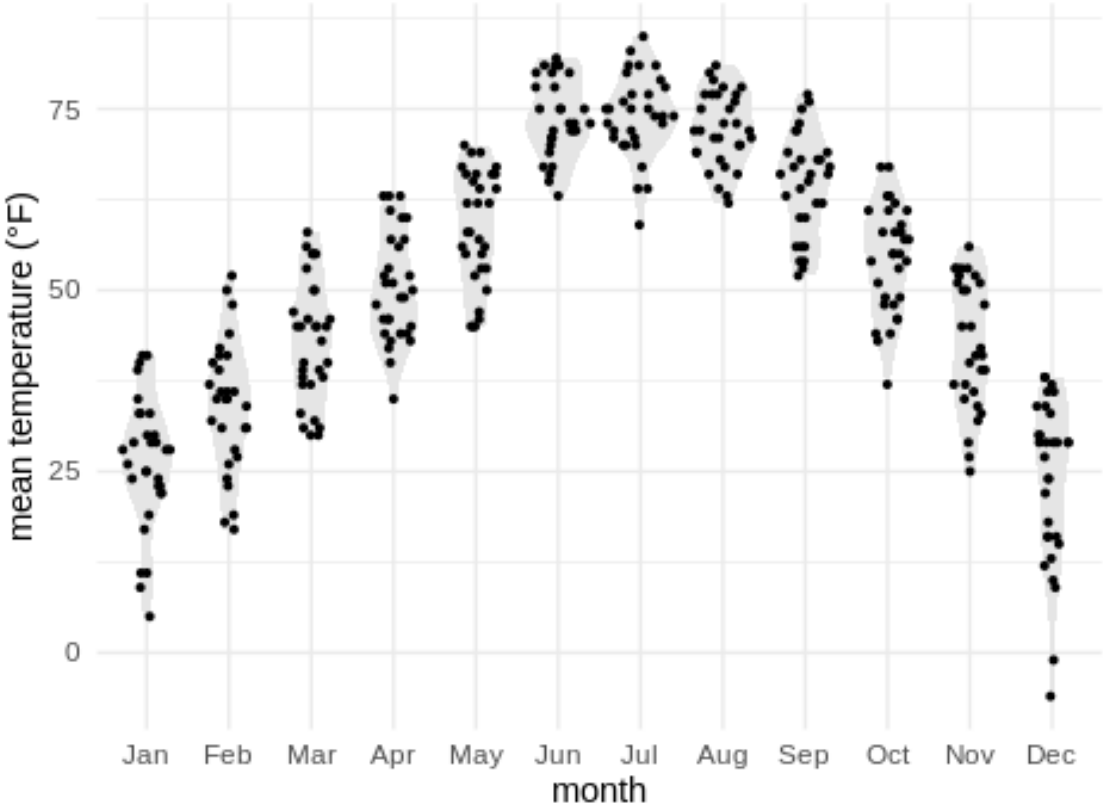
- How to do it in ggplot2?
  - It is not yet in ggplot2... Load Wilke's implementation of stat_sina()
  - Overplot the points over a normal violin plot

```
source("https://www.ics.uci.edu/~algol/teaching/informatics143w2021/claus_sina_stat.R")

ggplot(lincoln_df, aes(x = month_short, y = Mean.Temperature..F.)) +
  geom_violin(color = "transparent", fill = "gray90") +
  stat_sina(size = 0.75) +
  xlab("month") +
  ylab("mean temperature (°F)") +
  theme_minimal()
```

Visualization of Distributions: Sina plots

- How to do it in ggplot2?
  - It is not yet in ggplot2... Load Claus' implementation of stat_sina()
  - Overplot the points over a normal violin plot

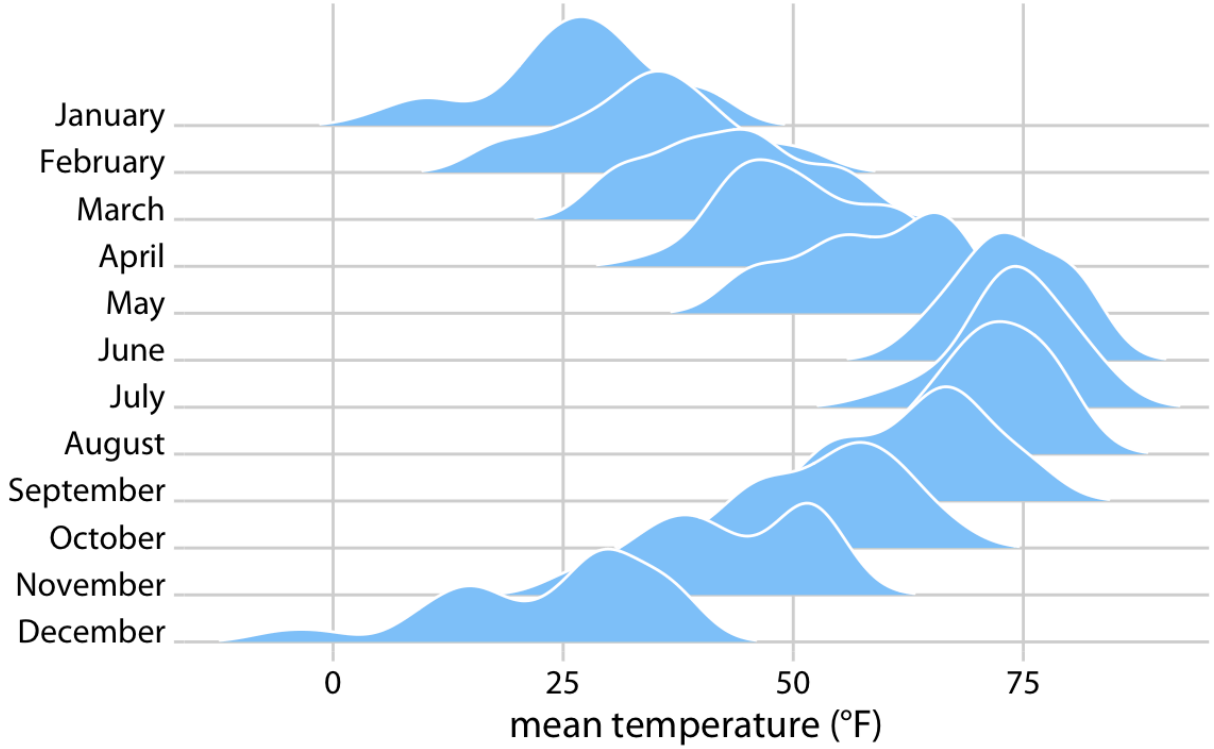


Visualization of Distributions: Ridgelines

- Ridgelines
  - Just multiple density plots shifted along the y-axis
  - Very effective to represent trends along the time (that runs over the y-axis!)
  - Evokes a more intuitive understanding of the data than violins

Visualization of Distributions: Ridgelines

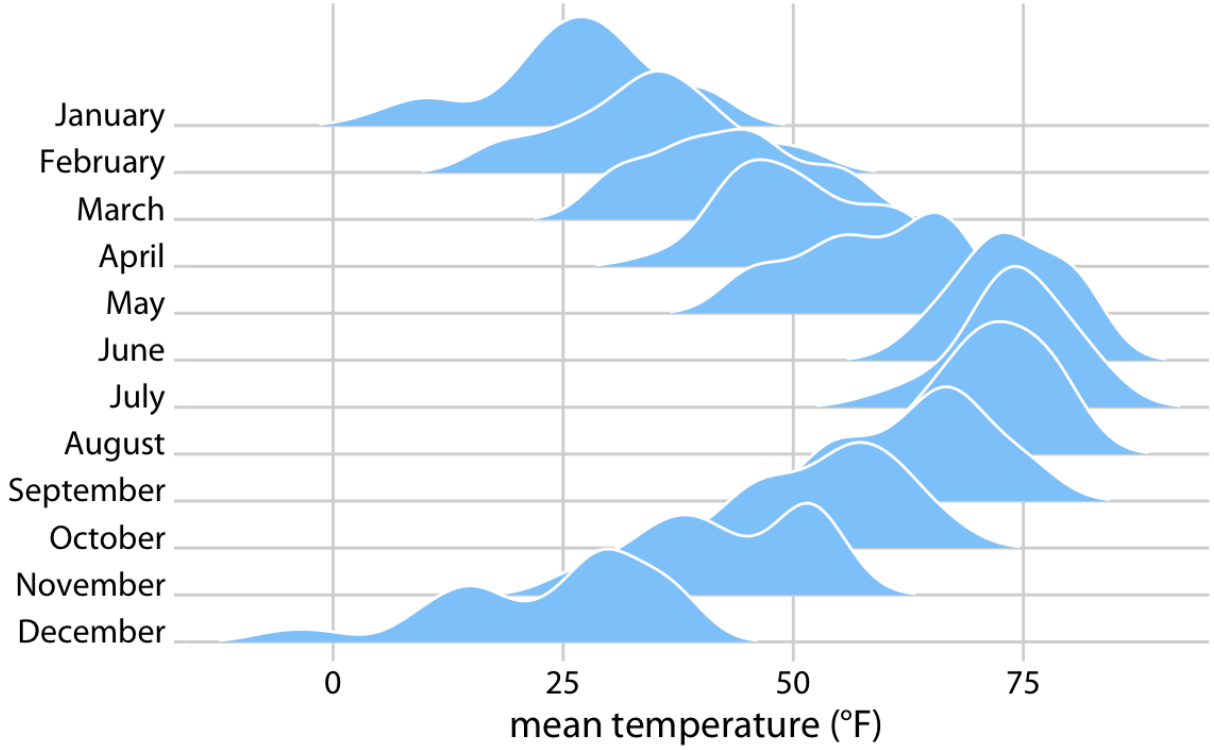
- Ridgelines



Data source: Weather Underground

Visualization of Distributions: Ridgelines

- Ridgelines

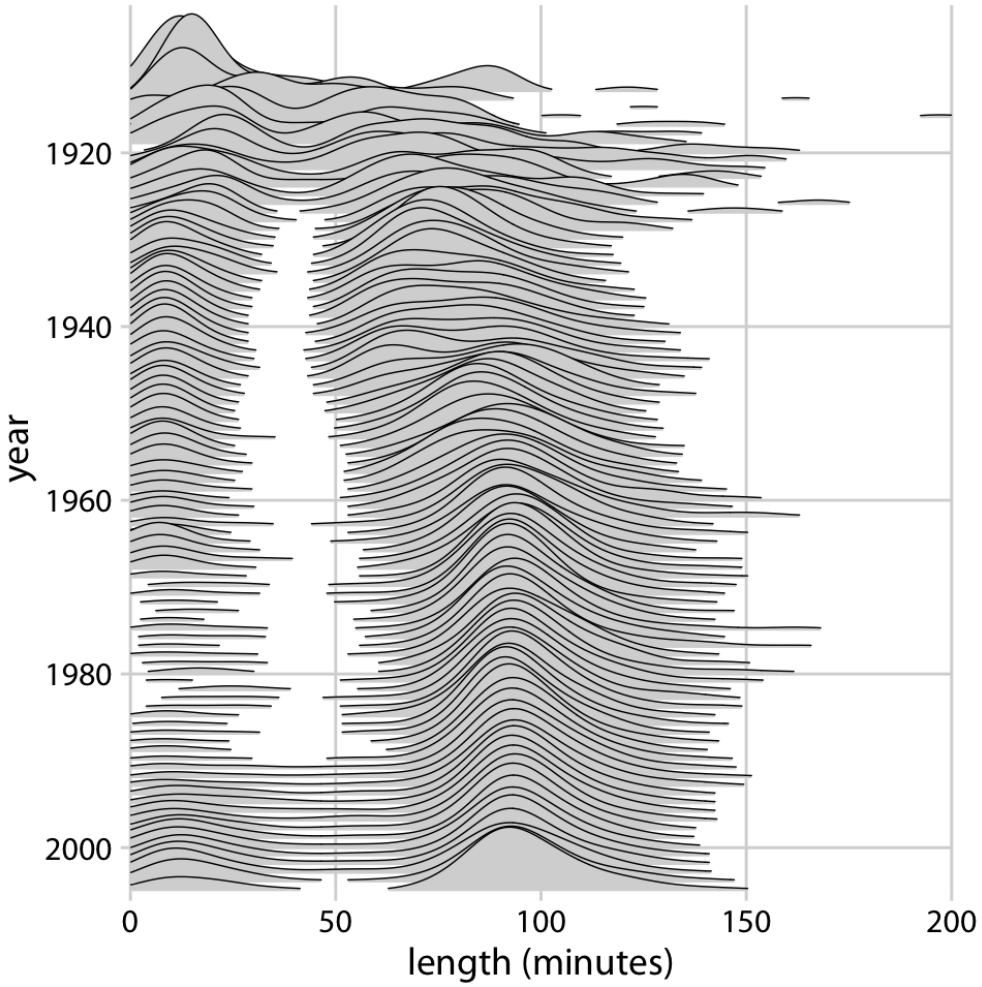


No separate axis for the density values.
The purpose is to allow easy qualitative comparison.

Data source: Weather Underground

Visualization of Distributions: Ridgelines

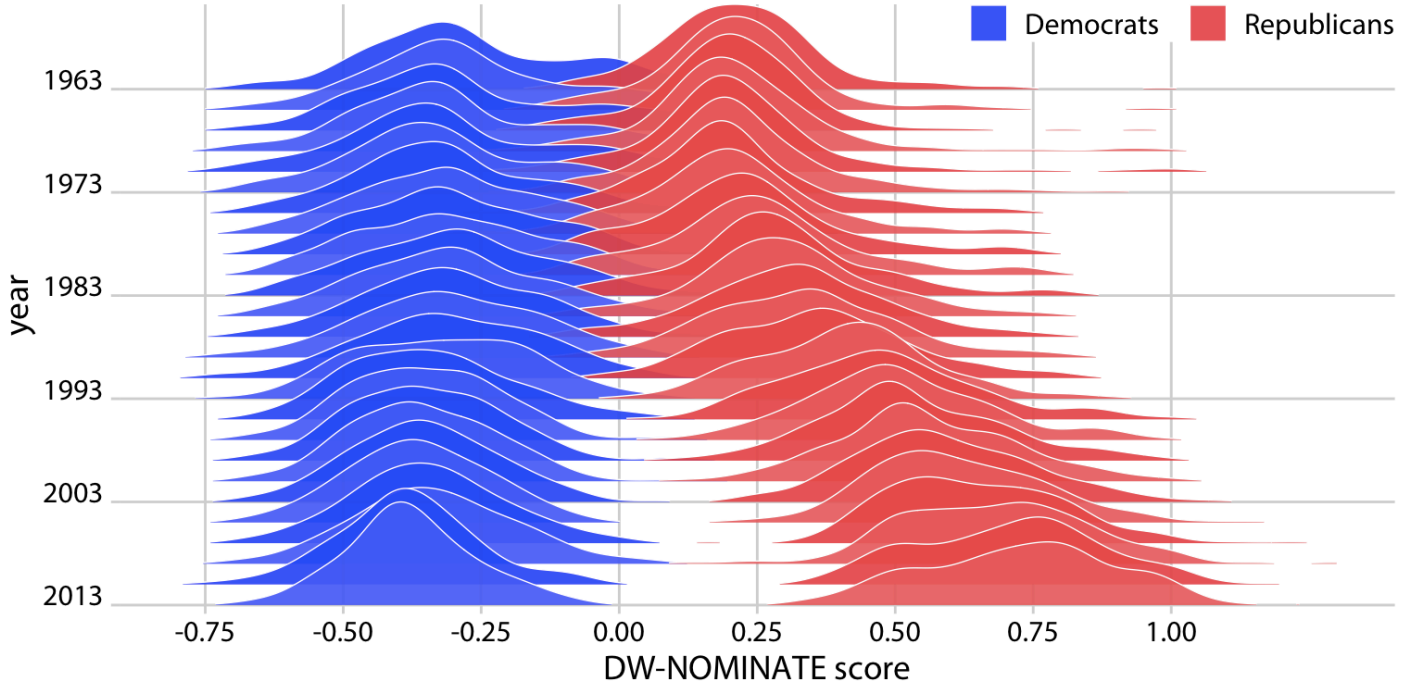
- They support a **very high** number of distributions



Data source: Internet Movie Database, IMDB

Visualization of Distributions: Ridgelines

- They support a very high number of distributions and multiple trends, such as voting patterns of representatives



Data source: McDonald, Ian. 2017

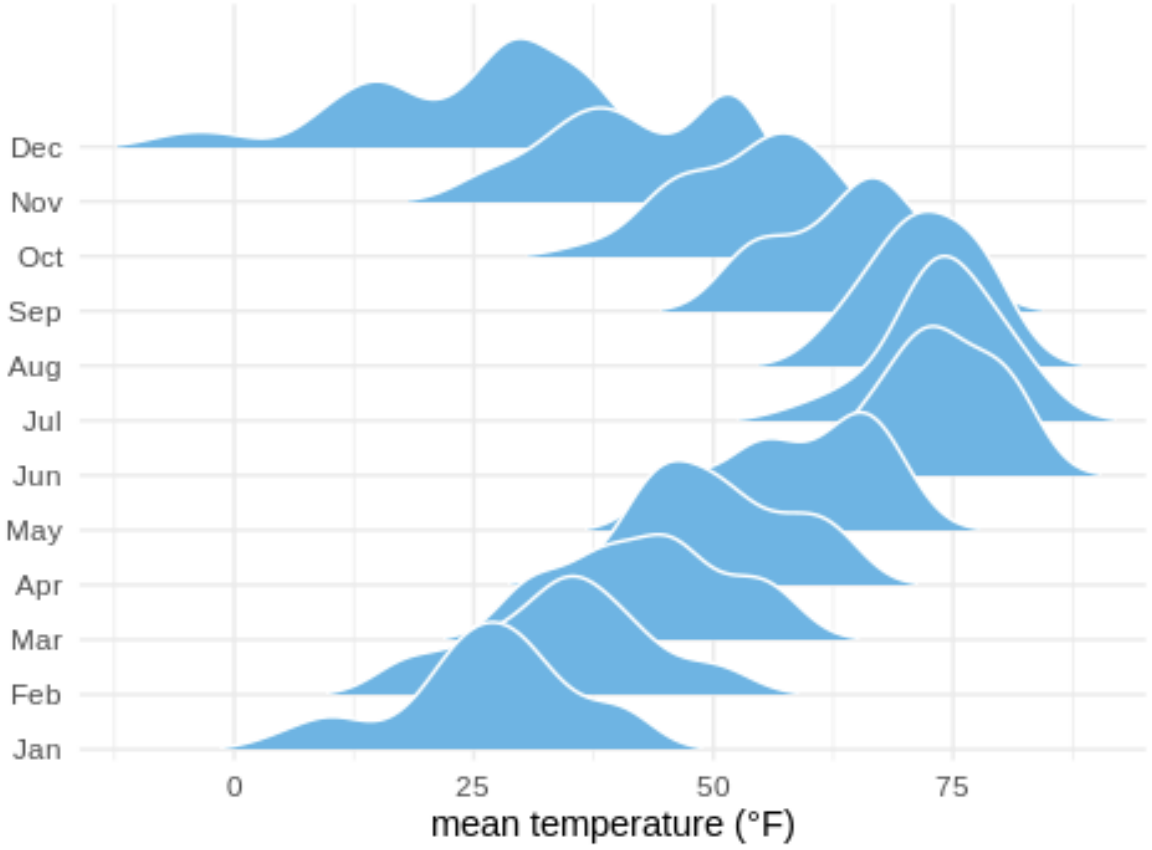
Visualization of Distributions: Ridgelines plots

- How to do ridgelines in ggplot2?
  - Use geom_density_ridges()

```
ggplot(lincoln_df, aes(x = Mean.Temperature..F., y = month_short)) +
  geom_density_ridges(
    scale = 3, rel_min_height = 0.01,
    fill = "#56B4E9", color = "white") +
  scale_x_continuous(
    name = "mean temperature (°F)",
    expand = c(0, 0), breaks = c(0, 25, 50, 75)) +
  scale_y_discrete(name = NULL, expand = c(0, .2, 0, 2.6)) +
  theme_minimal()
```

Visualization of Distributions: Ridgelines plots

- How to do ridgelines in ggplot2?
  - Use geom_density_ridges()



Visualization of Distributions: Ridgelines plots

- How to reproduce the movies plots?
  - Use geom_density_ridges(). There is no difference at all to support large number of distributions!

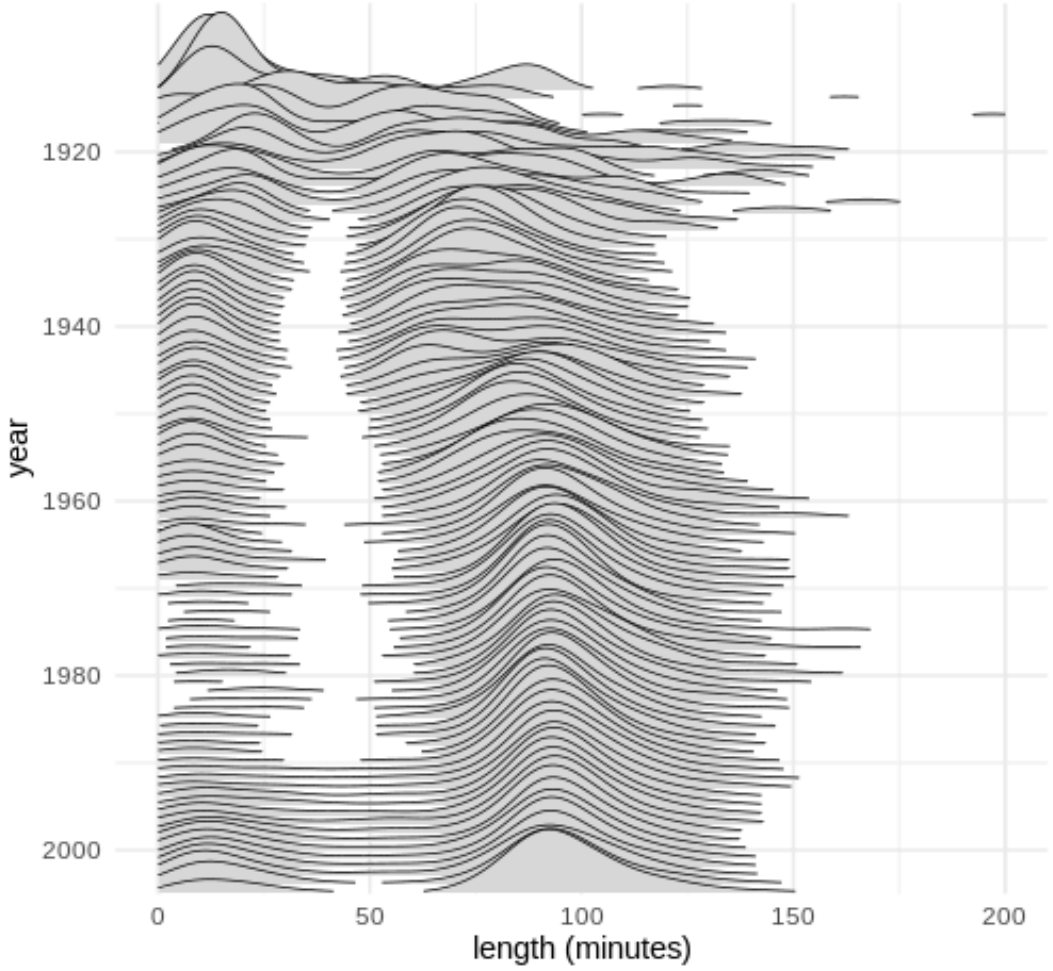
```
require(ggplot2movies) # to load the data; you may need to install first.
require(ggridges)

movie_lengths <- movies[which(movies$year>1912), c("length", "year")]

ggplot(movie_lengths, aes(x = length, y = year, group = year)) +
  geom_density_ridges(scale = 10, size = 0.25,
                      rel_min_height = 0.03, fill = "grey85", na.rm = TRUE) +
  scale_x_continuous(limits = c(0, 200),
                     name = "length (minutes)") +
  scale_y_reverse(breaks = c(2000, 1980, 1960, 1940, 1920),
                  limits = c(2005, 1903)) +
  coord_cartesian(clip = "off") +
  theme_minimal()
```

Visualization of Distributions: Ridgelines plots

- How to reproduce the movies plots?



Visualization of Distributions: Ridgelines plots

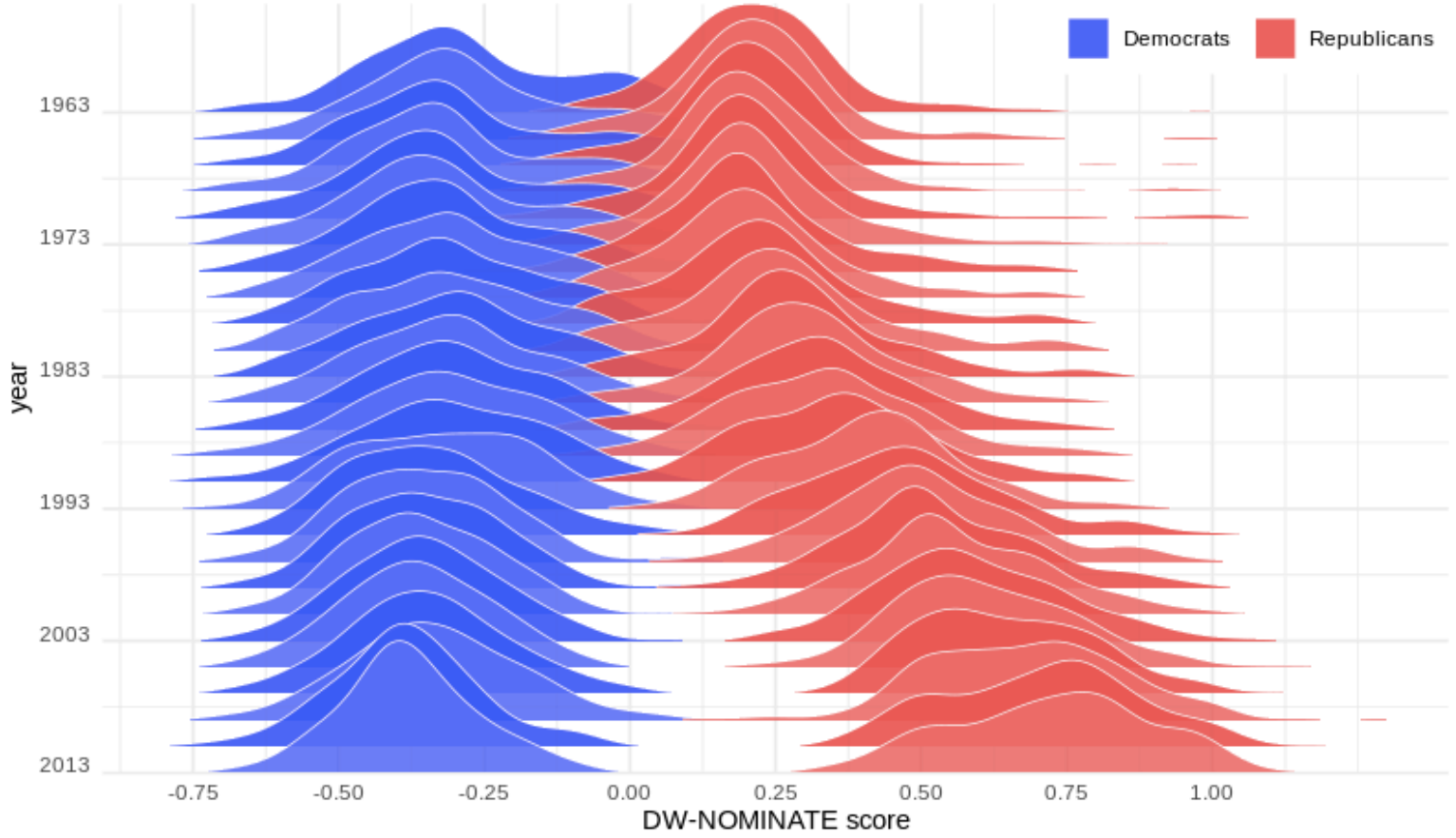
- How to reproduce the voting pattern evolution plot?

```
house_data <-
read.csv("https://www.ics.uci.edu/~algol/teaching/informatics143w2021/house_data.csv")

ggplot(house_data, aes(x = dim_1, y = year1,
                       group = interaction(party_code, factor(year1)),
                       fill = interaction(party_code, factor(year1)))) +
      geom_density_ridges(scale = 5, size = 0.25,
                          rel_min_height = 0.01, alpha=0.9, color = "white") +
      scale_x_continuous(name = "DW-NOMINATE score", limits = c(-.8, 1.3),
                         breaks = c(-1,-.75,-.5,-.25,0,.25,.5,.75,1)) +
      scale_y_reverse(name = "year", expand = c(0, 0),
                      breaks=c(seq(2013, 1963, -10))) +
      scale_fill_cyclical(
            breaks = c("100.1963", "200.1963"),
            labels = c(`100.1963` = "Democrats  ", `200.1963` = "Republicans"),
            values = c("#4040ff", "#ff4040", "#6060ff", "#ff6060"),
            name = NULL, guide = "legend") +
      theme_minimal() +
      theme(axis.text.y = element_text(vjust = 0),
            legend.position = c(1, 1),
            legend.justification = c(1, 1),
            legend.direction = "horizontal",
            legend.background = element_rect(fill = "white",color = "white"))
```

Visualization of Distributions: Ridgelines plots

- How to reproduce the voting pattern evolution plot?



Visualization of Distributions