

ggplot2 ecosystem & designing visualizations

Lecture 10

Dr. Colin Rundel

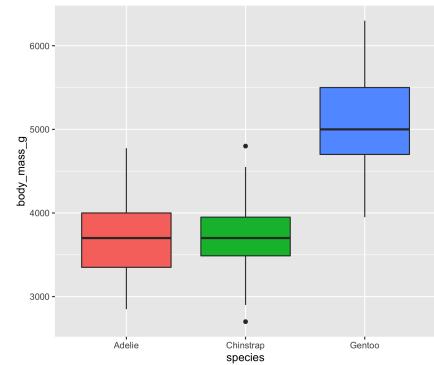
The wider ggplot2 ecosystem

ggthemes

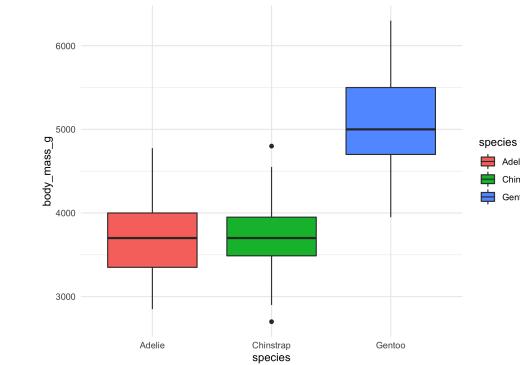
ggplot2 themes

```
1 g = ggplot( palmerpenguins::penguins,
2             aes(x=species, y=body_mass_g, fill=species)) +
3   geom_boxplot()
```

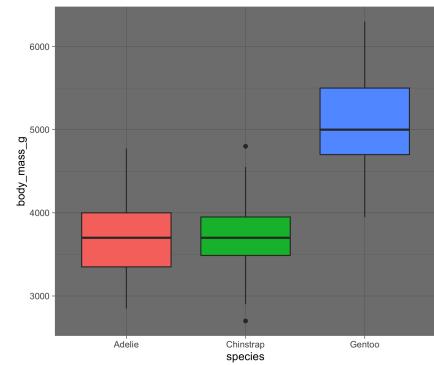
```
1 g
```



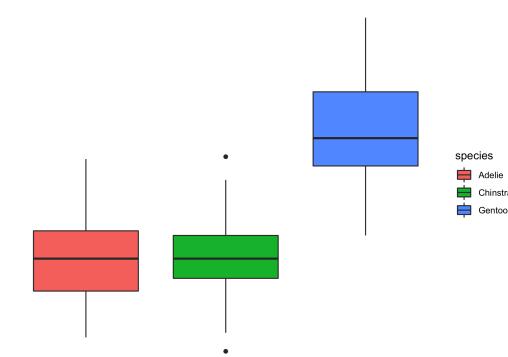
```
1 g + theme_minimal()
```



```
1 g + theme_dark()
```

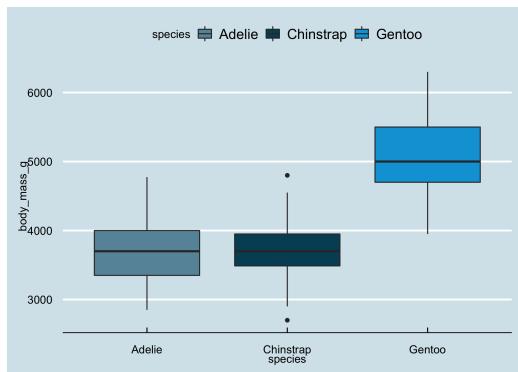


```
1 g + theme_void()
```

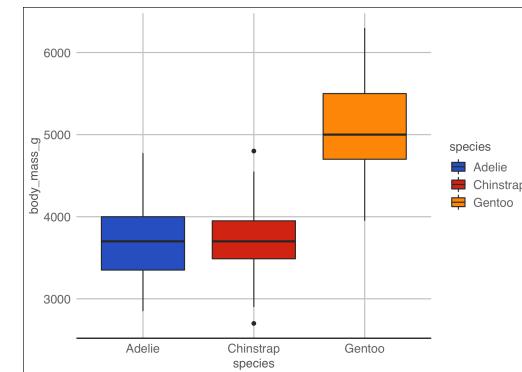


ggthemes

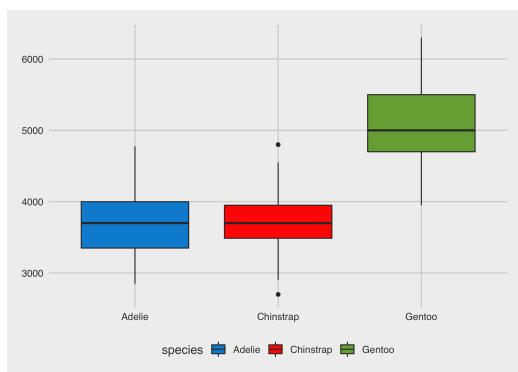
```
1 g + ggthemes::theme_economist() +  
2 ggthemes::scale_fill_economist()
```



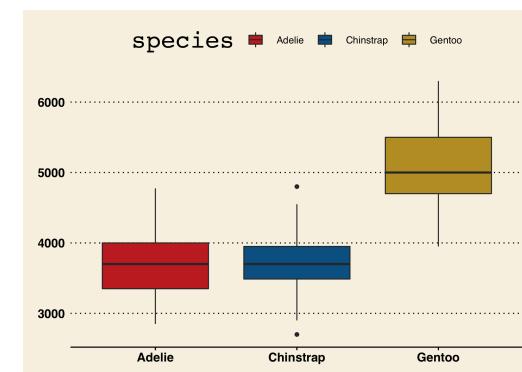
```
1 g + ggthemes::theme_gdocs() +  
2 ggthemes::scale_fill_gdocs()
```



```
1 g + ggthemes::theme_fivethirtyeight() +  
2 ggthemes::scale_fill_fivethirtyeight()
```

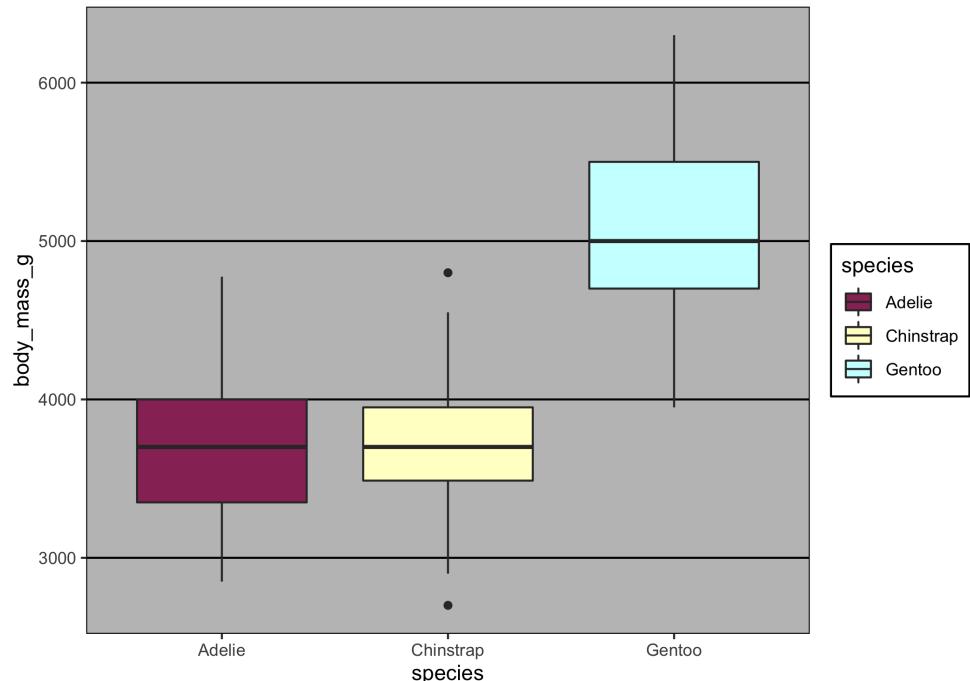


```
1 g + ggthemes::theme_wsj() +  
2 ggthemes::scale_fill_wsj()
```

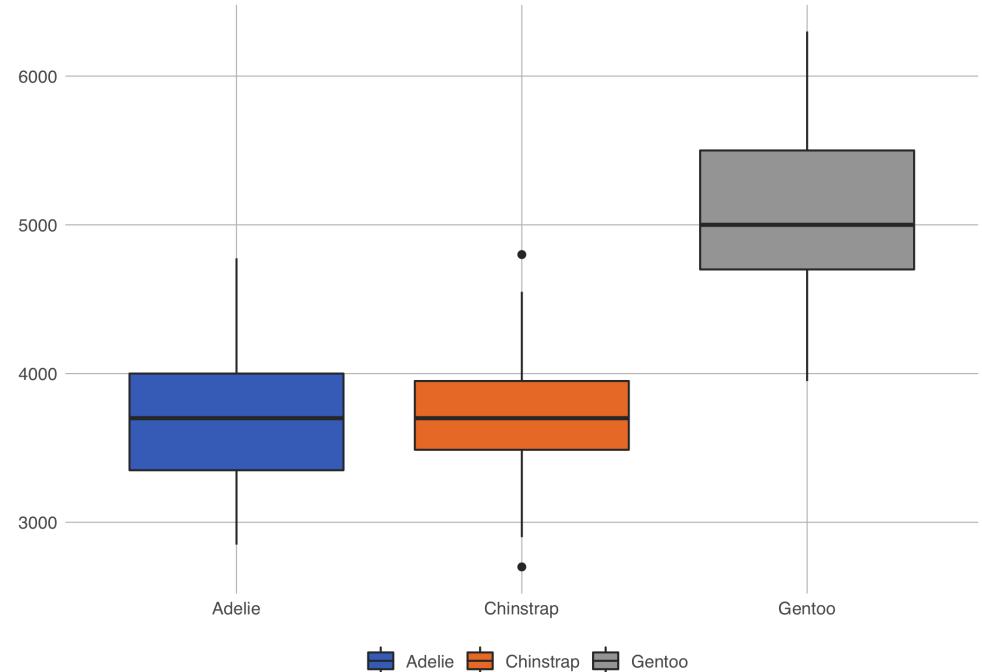


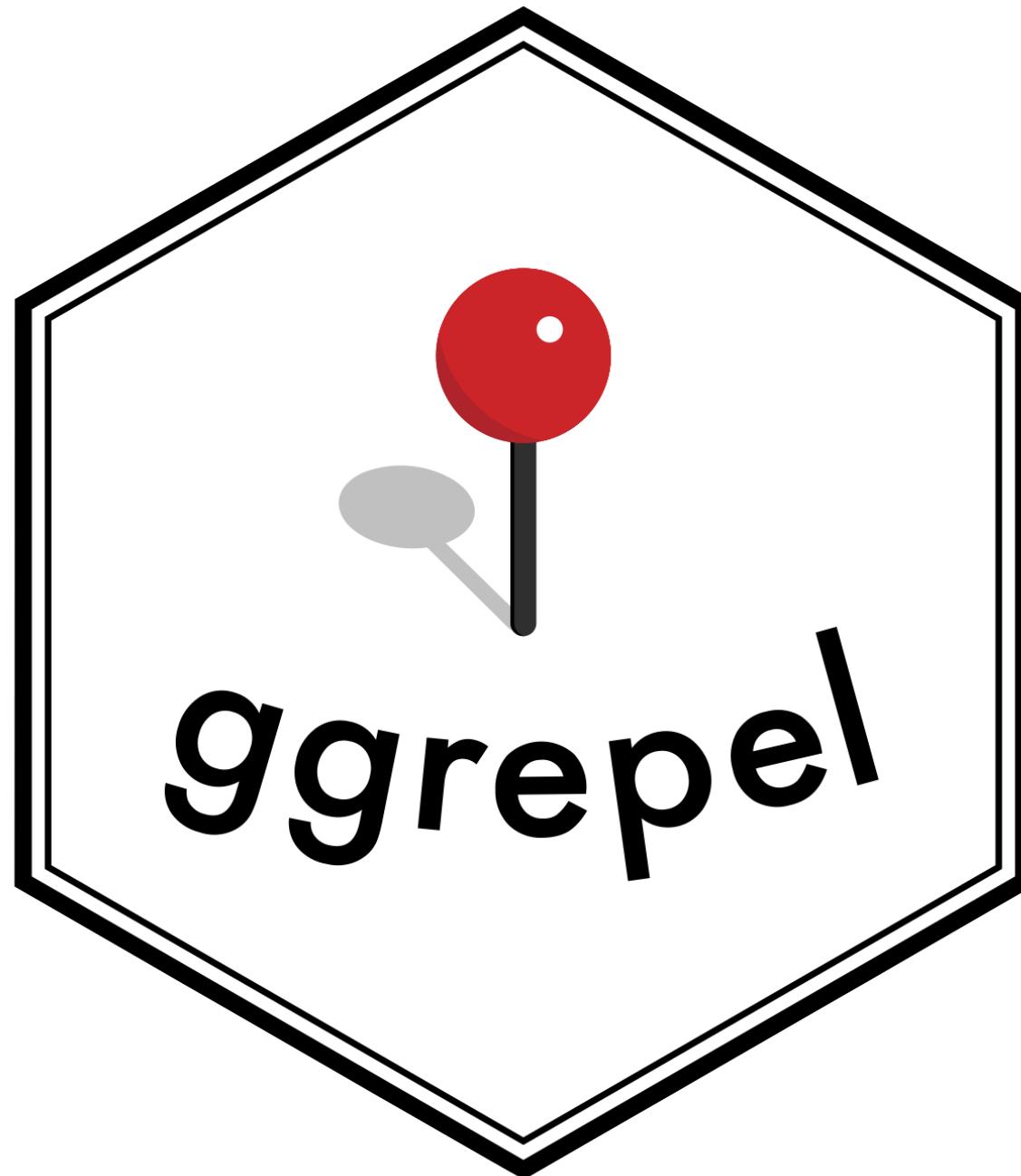
And for those who miss Excel

```
1 g + ggthemes::theme_excel() +  
2 ggthemes::scale_fill_excel()
```



```
1 g + ggthemes::theme_excel_new() +  
2 ggthemes::scale_fill_excel_new()
```





```

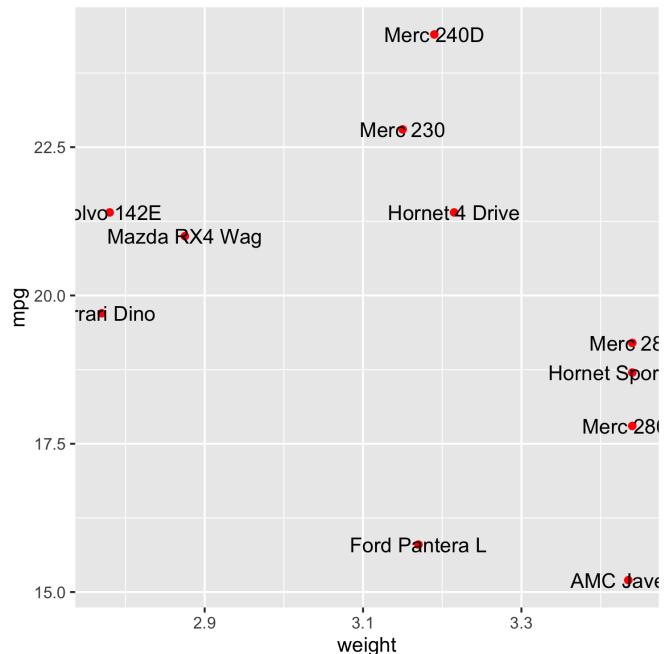
1 d = tibble(
2   car = rownames(mtcars),
3   weight = mtcars$wt,
4   mpg = mtcars$mpg
5 ) %>%
6 filter(weight > 2.75, weight < 3.45)

```

```

1 ggplot(d, aes(x=weight, y=mpg)) +
2   geom_point(color="red") +
3   geom_text(
4     aes(label = car)
5 )

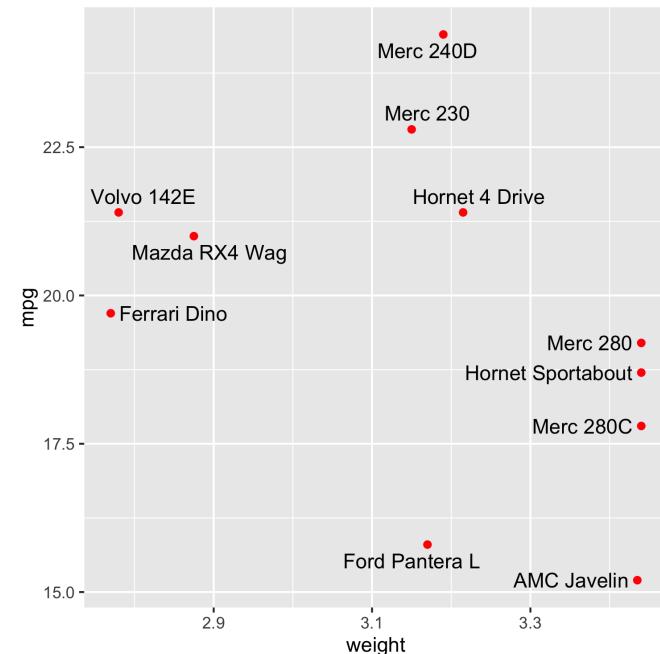
```



```

1 ggplot(d, aes(x=weight, y=mpg)) +
2   geom_point(color="red") +
3   ggrepel::geom_text_repel(
4     aes(label = car)
5 )

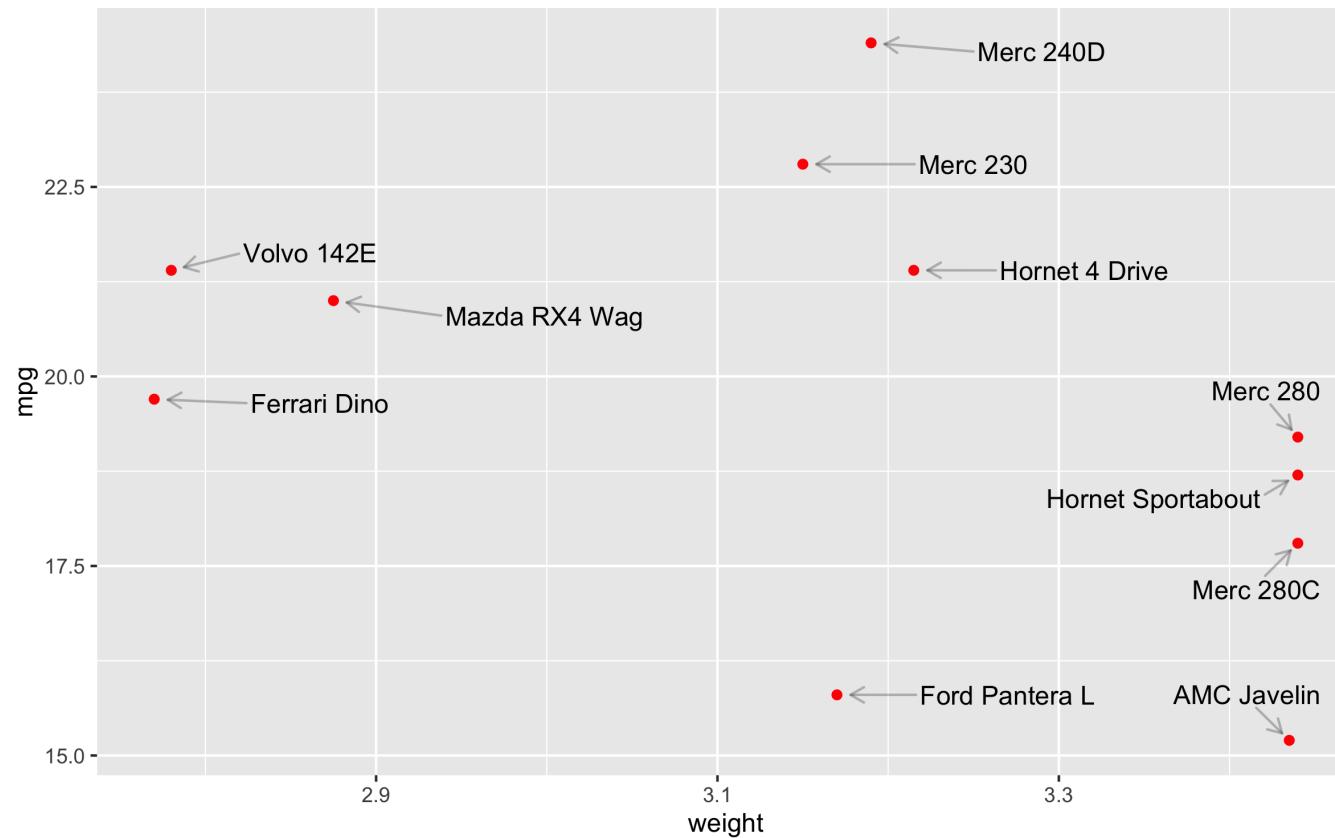
```



```

1 ggplot(d, aes(x=weight, y=mpg)) +
2   geom_point(color="red") +
3   ggrepel::geom_text_repel(
4     aes(label = car),
5     nudge_x = .1, box.padding = 1, point.padding = 0.6,
6     arrow = arrow(length = unit(0.02, "npc")), segment.alpha = 0.25
7 )

```





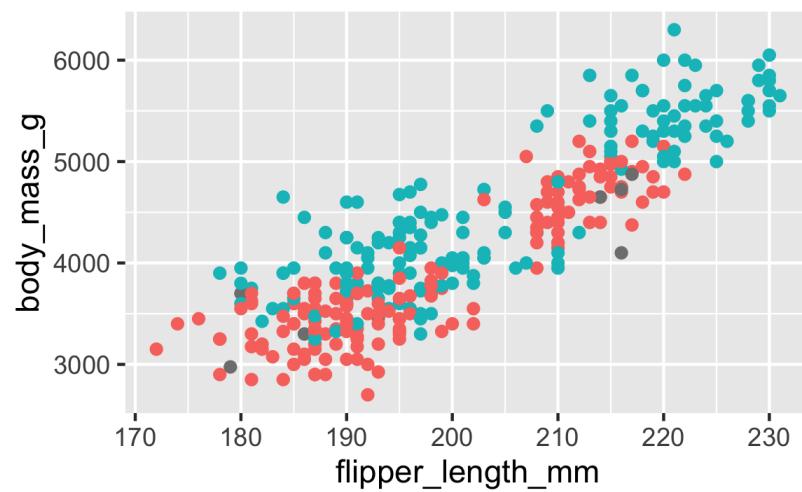
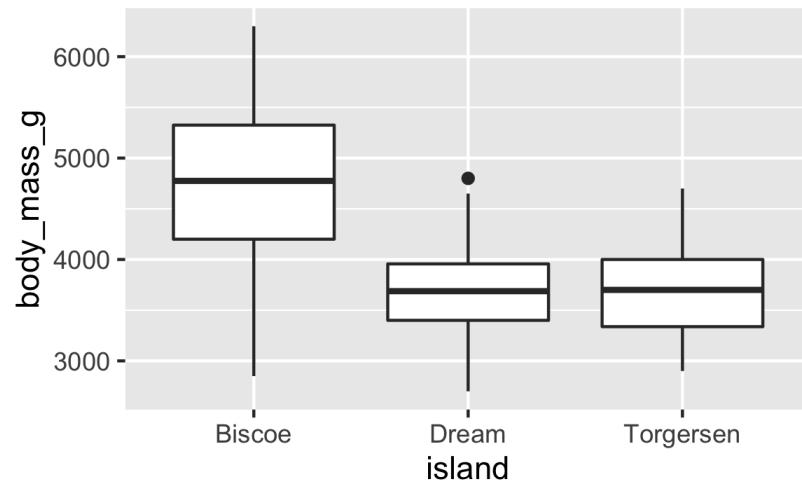
Sta 523 - Fall 2022

ggplot objects

```
1 library(patchwork)
2
3 p1 = ggplot(palmerpenguins::penguins) +
4   geom_boxplot(aes(x = island, y = body_mass_g))
5
6 p2 = ggplot(palmerpenguins::penguins) +
7   geom_boxplot(aes(x = species, y = body_mass_g))
8
9 p3 = ggplot(palmerpenguins::penguins) +
10  geom_point(aes(x = flipper_length_mm, y = body_mass_g, color = sex))
11
12 p4 = ggplot(palmerpenguins::penguins) +
13  geom_point(aes(x = bill_length_mm, y = body_mass_g, color = sex))
```

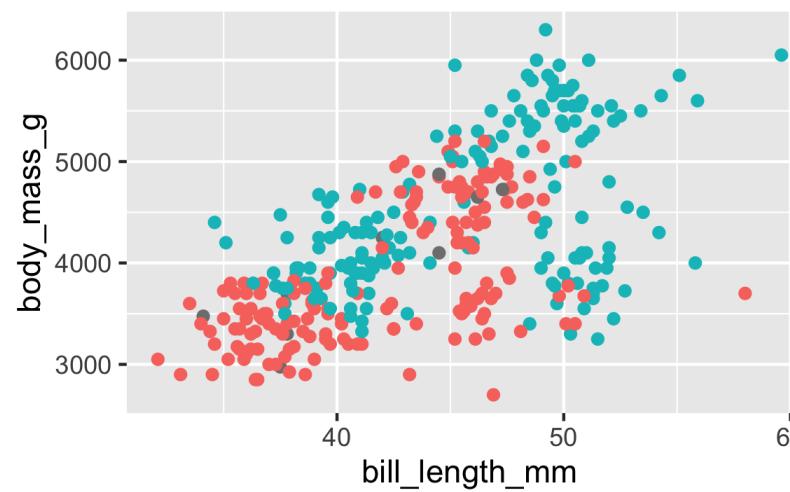
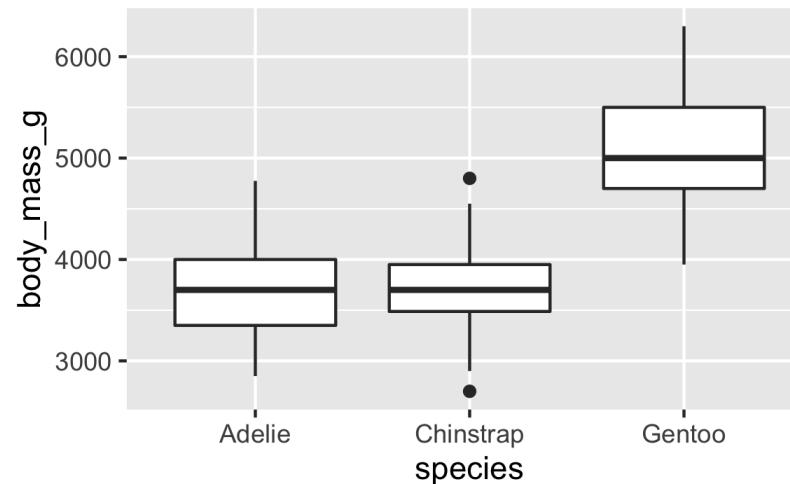
```
1 class(p1)
[1] "gg"     "ggplot"
```

1 p1 + p2 + p3 + p4



sex

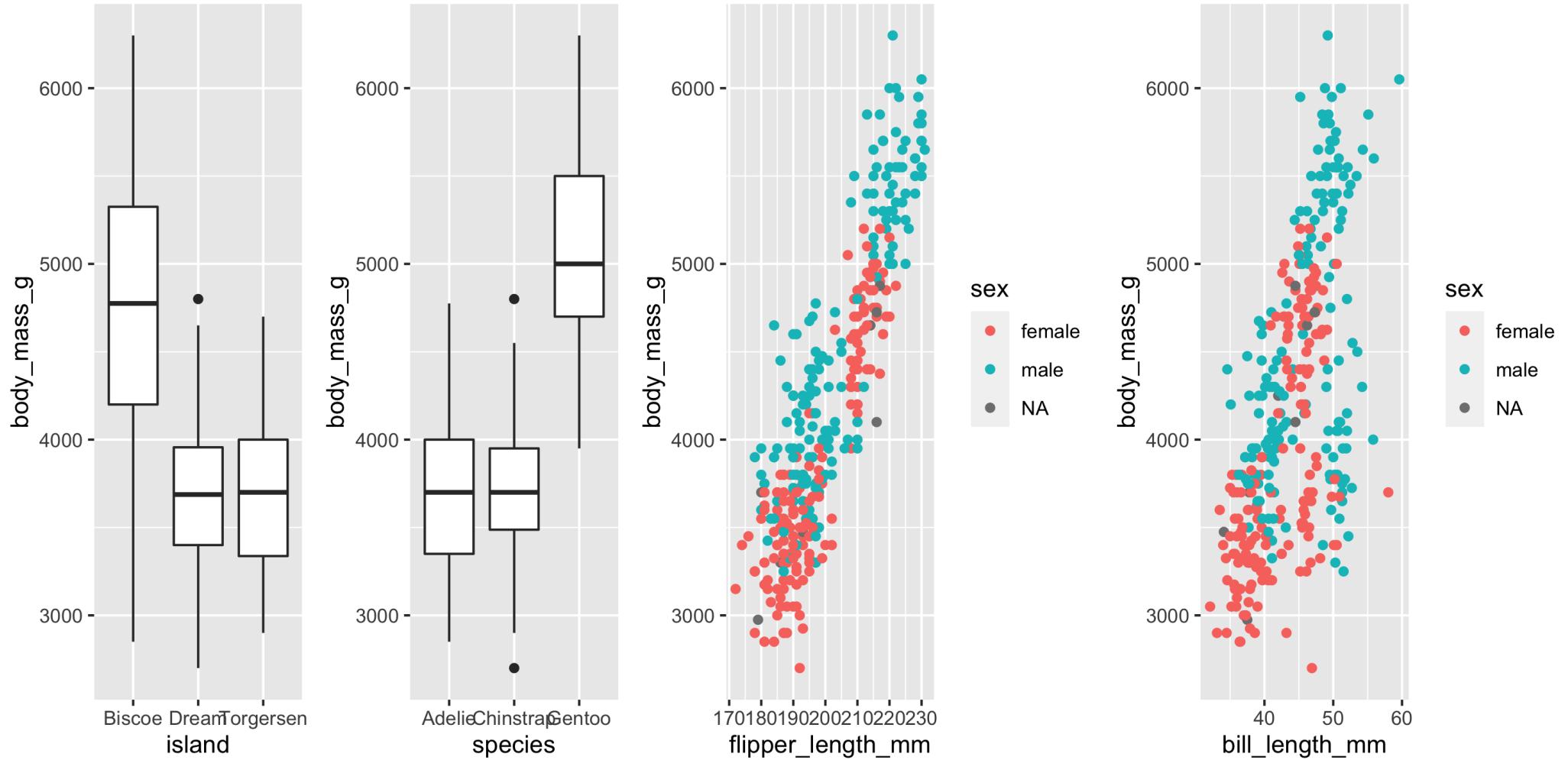
- female
- male
- NA



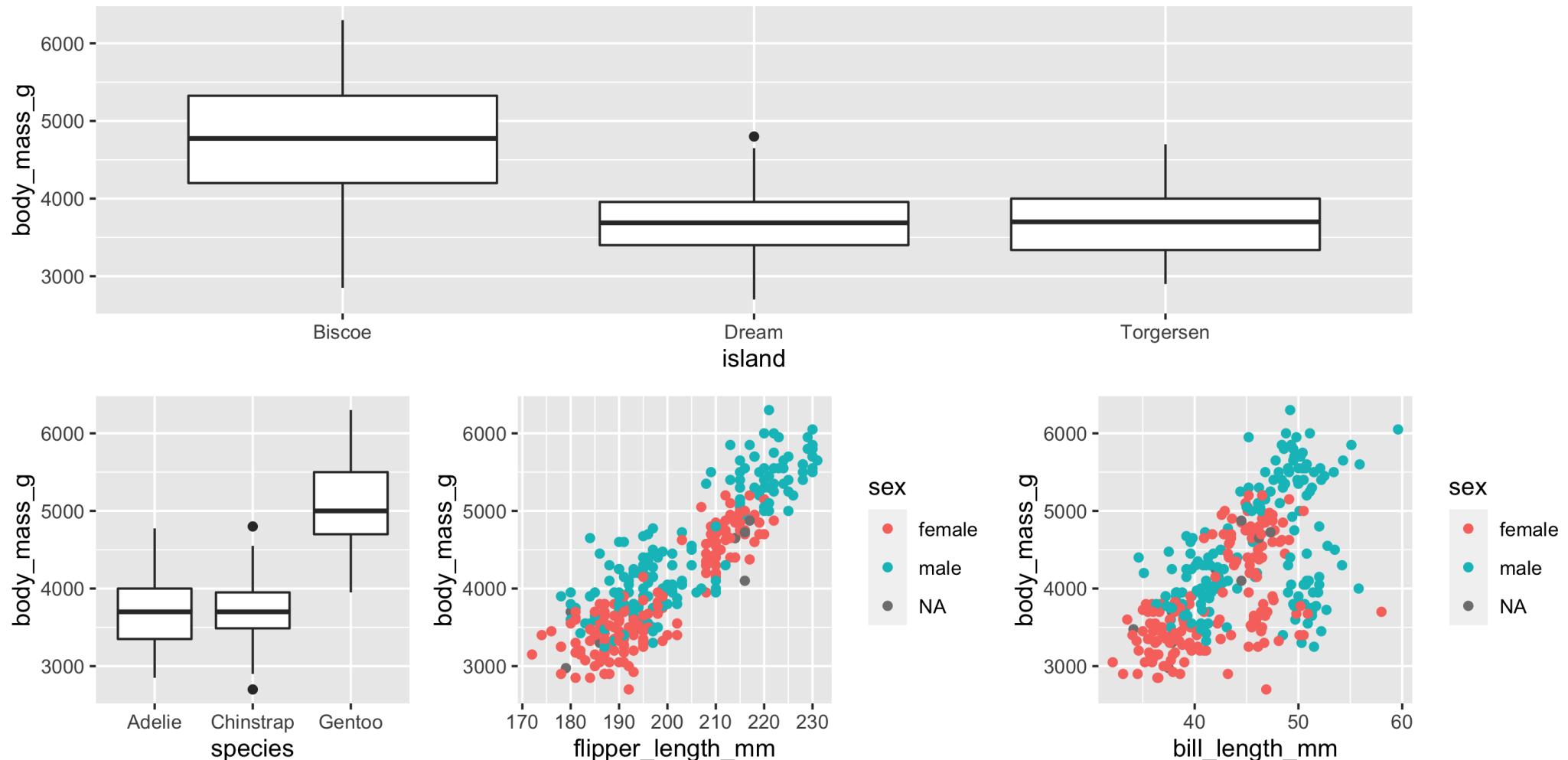
sex

- female
- male
- NA

```
1 p1 + p2 + p3 + p4 + plot_layout(nrow=1)
```



$1 \ p1 \ / \ (p2 + p3 + p4)$



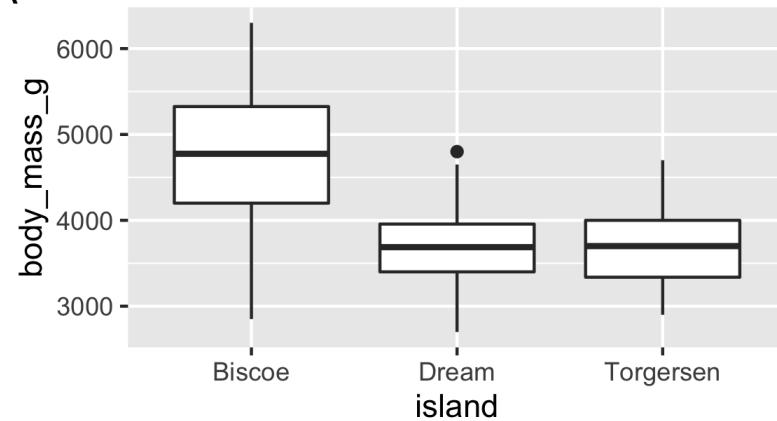
```

1 p1 + p2 + p3 + p4 +
2 plot_annotation(title = "Palmer Penguins", tag_levels = c("A"))

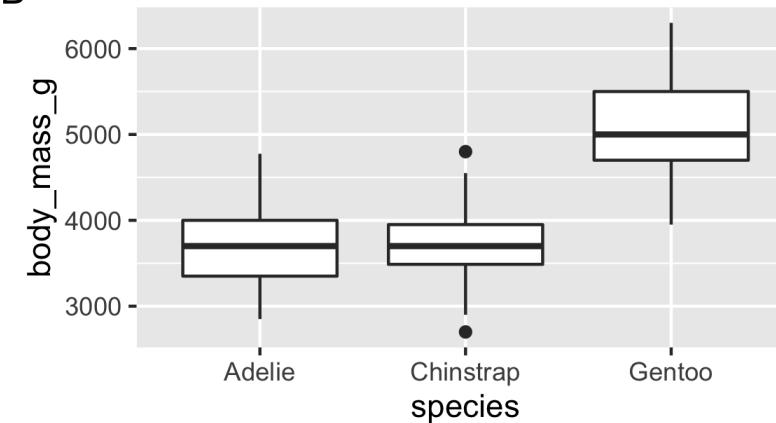
```

Palmer Penguins

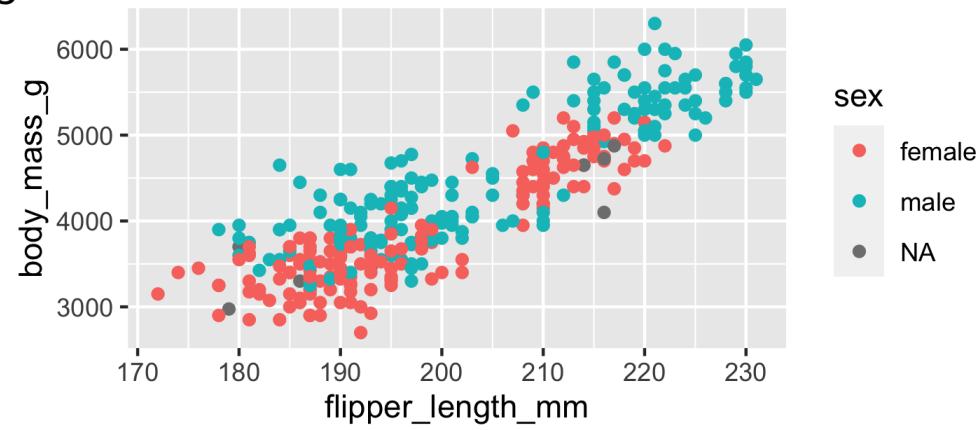
A



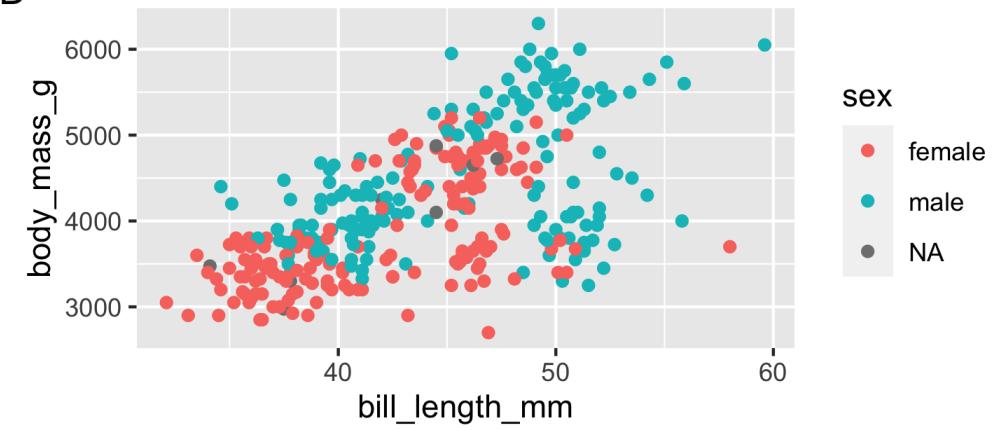
B



C



D



```

1 p1 + {
2   p2 + {
3     p3 + p4 + plot_layout(ncol = 1) + plot_layout(tag_level = 'new')
4   }
5 } +
6 plot_layout(ncol = 1) +
7 plot_annotation(tag_levels = c("1","a"), tag_prefix = "Fig ")

```

Fig 1

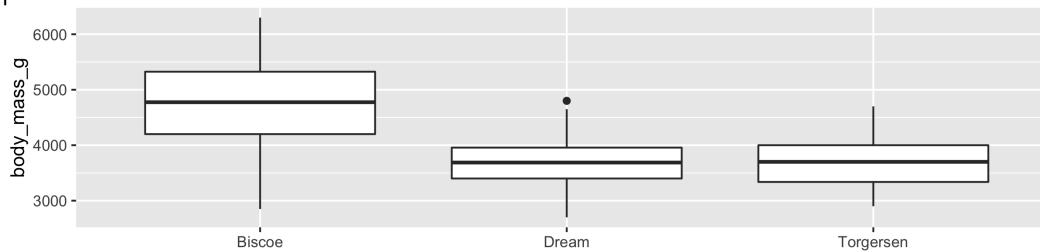


Fig 2

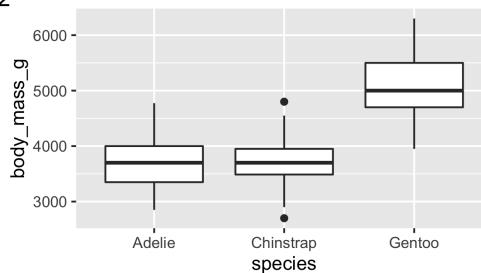


Fig 3a

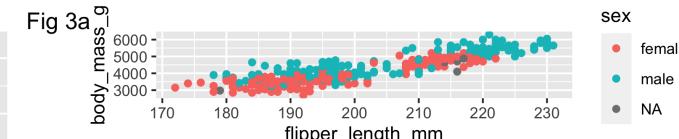
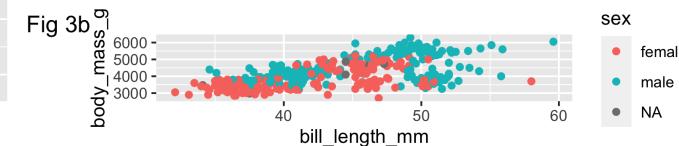
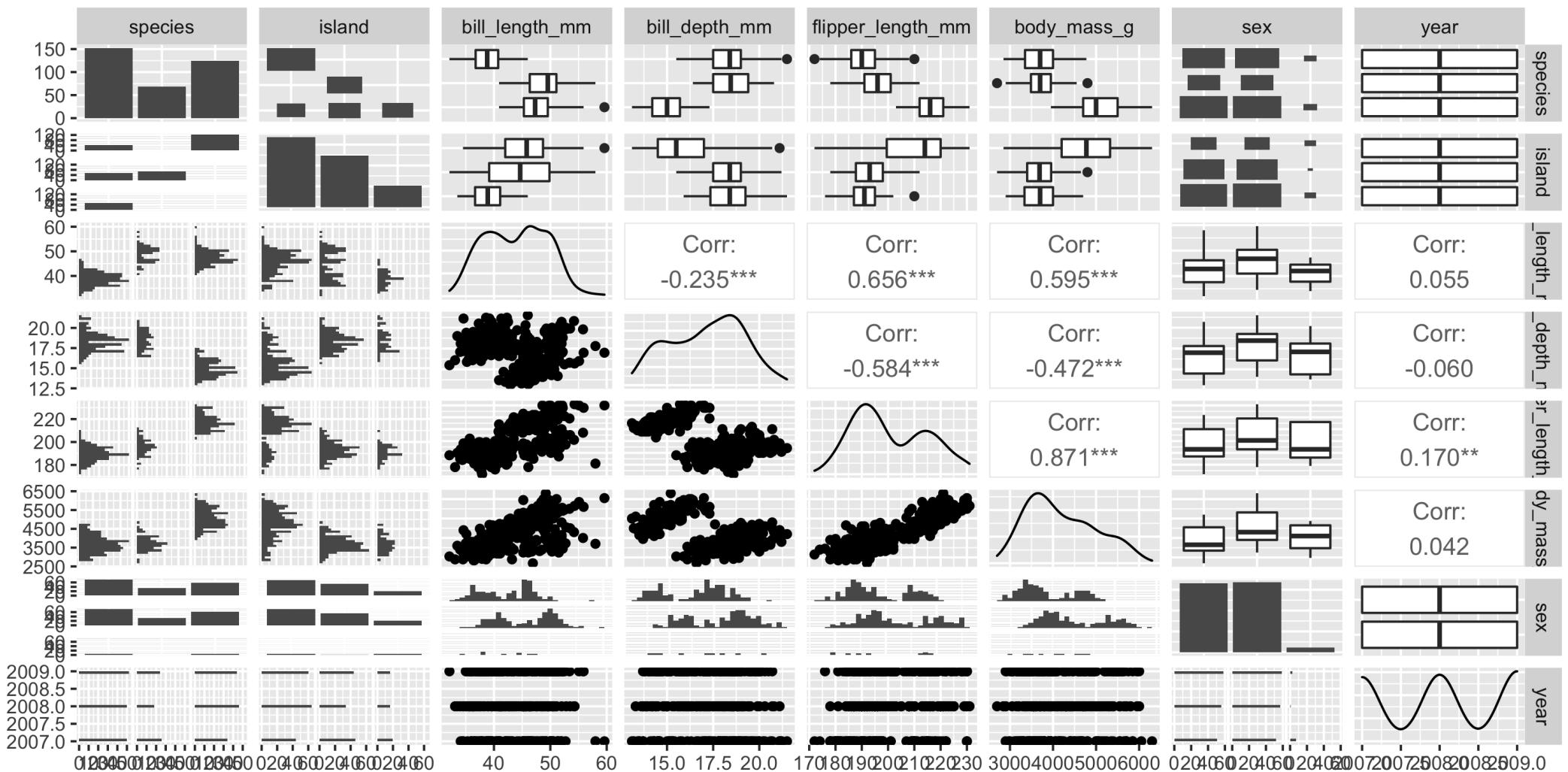


Fig 3b

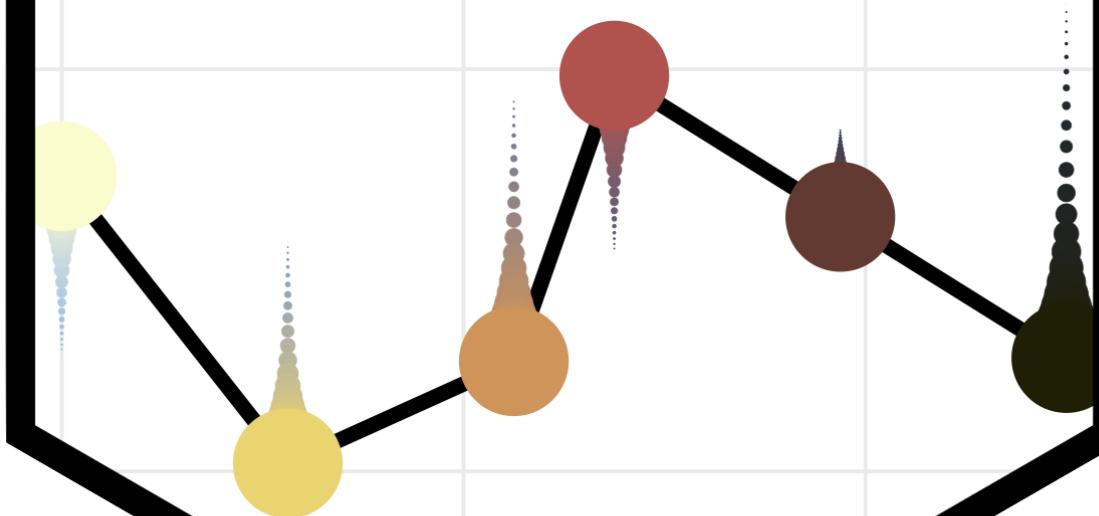


GGally

1 GGallery::ggpairs(palmerpenguins::penguins)



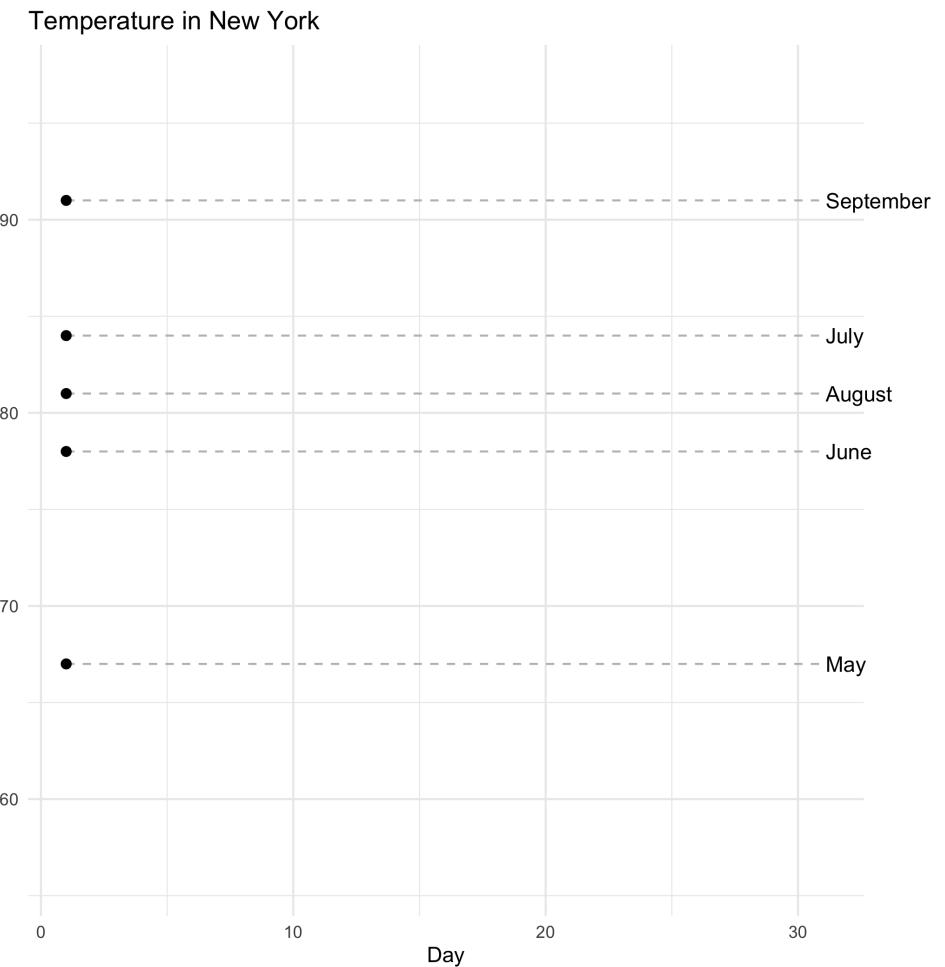
ganimate



```

1 airq = airquality
2 airq$Month = month.name[airq$Month]
3
4 ggplot(
5   airq,
6   aes(Day, Temp, group = Month)
7 ) +
8   geom_line() +
9   geom_segment(
10    aes(xend = 31, yend = Temp),
11    linetype = 2,
12    colour = 'grey'
13 ) +
14   geom_point(size = 2) +
15   geom_text(
16    aes(x = 31.1, label = Month),
17    hjust = 0
18 ) +
19   gganimate::transition_reveal(Day) +
20   coord_cartesian(clip = 'off') +
21   labs(
22     title = 'Temperature in New York',
23     y = 'Temperature (°F)'

```



More extensions

exts.ggplot2.tidyverse.org/gallery/

ggplot2 extensions - gallery Add Your Extension! exts.ggplot2.tidyverse.org

Sort Text Filter Author Filter Tag Filter CRAN Only

Github stars ▼ search name, autho ▼

Showing 86 of 101

The screenshot shows the 'ggplot2 extensions' gallery interface. At the top, there are filters for 'Sort' (set to 'Github stars'), 'Text Filter' (empty), 'Author Filter' (empty), 'Tag Filter' (empty), and a 'CRAN Only' toggle switch (on). Below these are dropdown menus for 'Github stars' (set to 'search name, autho') and 'Tag Filter' (empty). The main area displays 86 of 101 registered extensions. Three examples are shown in detail:

- patchwork** (Star 1932): A package for easy composition of ggplot plots using arithmetic operators. It includes a histogram of 'carb' values, a scatter plot of 'disp' vs 'mpg', and a box plot of 'gear'.
- gganimate** (Star 1709): A Grammar of Animated Graphics. It shows a time series plot of GDP per capita for different regions from 1958 to 2015.
- ggstatsplot** (Star 1283): Provides a collection of functions to enhance ggplot2 plots with results from statistical tests. It includes a violin plot of Sepal Length across Iris species with statistical annotations.

Why do we visualize?

Anscombe's Quartet

```
1 datasets::anscombe %>% as_tibble()
```

```
# A tibble: 11 × 8
```

	x1	x2	x3	x4	y1	y2	y3	y4
1	10	10	10	8	8.04	9.14	7.46	6.58
2	8	8	8	8	6.95	8.14	6.77	5.76
3	13	13	13	8	7.58	8.74	12.7	7.71
4	9	9	9	8	8.81	8.77	7.11	8.84
5	11	11	11	8	8.33	9.26	7.81	8.47
6	14	14	14	8	9.96	8.1	8.84	7.04
7	6	6	6	8	7.24	6.13	6.08	5.25
8	4	4	4	19	4.26	3.1	5.39	12.5
9	12	12	12	8	10.8	9.13	8.15	5.56
10	7	7	7	0	4.92	7.26	6.82	7.01

Tidy anscombe

```
1 tidy_anscombe = datasets::anscombe %>%
2   pivot_longer(everything(), names_sep = 1, names_to = c("var", "group")) %>%
3   pivot_wider(id_cols = group, names_from = var,
4               values_from = value, values_fn = list(value = list)) %>%
5   unnest(cols = c(x,y)))
```

```
# A tibble: 44 × 3
```

```
  group     x     y
  <chr> <dbl> <dbl>
1 1       10  8.04
2 1       8   6.95
3 1       13  7.58
4 1       9   8.81
5 1       11  8.33
6 1       14  9.96
7 1       6   7.24
8 1       4   4.26
9 1       12  10.8
10 1      7   4.82
# ... with 34 more rows
```

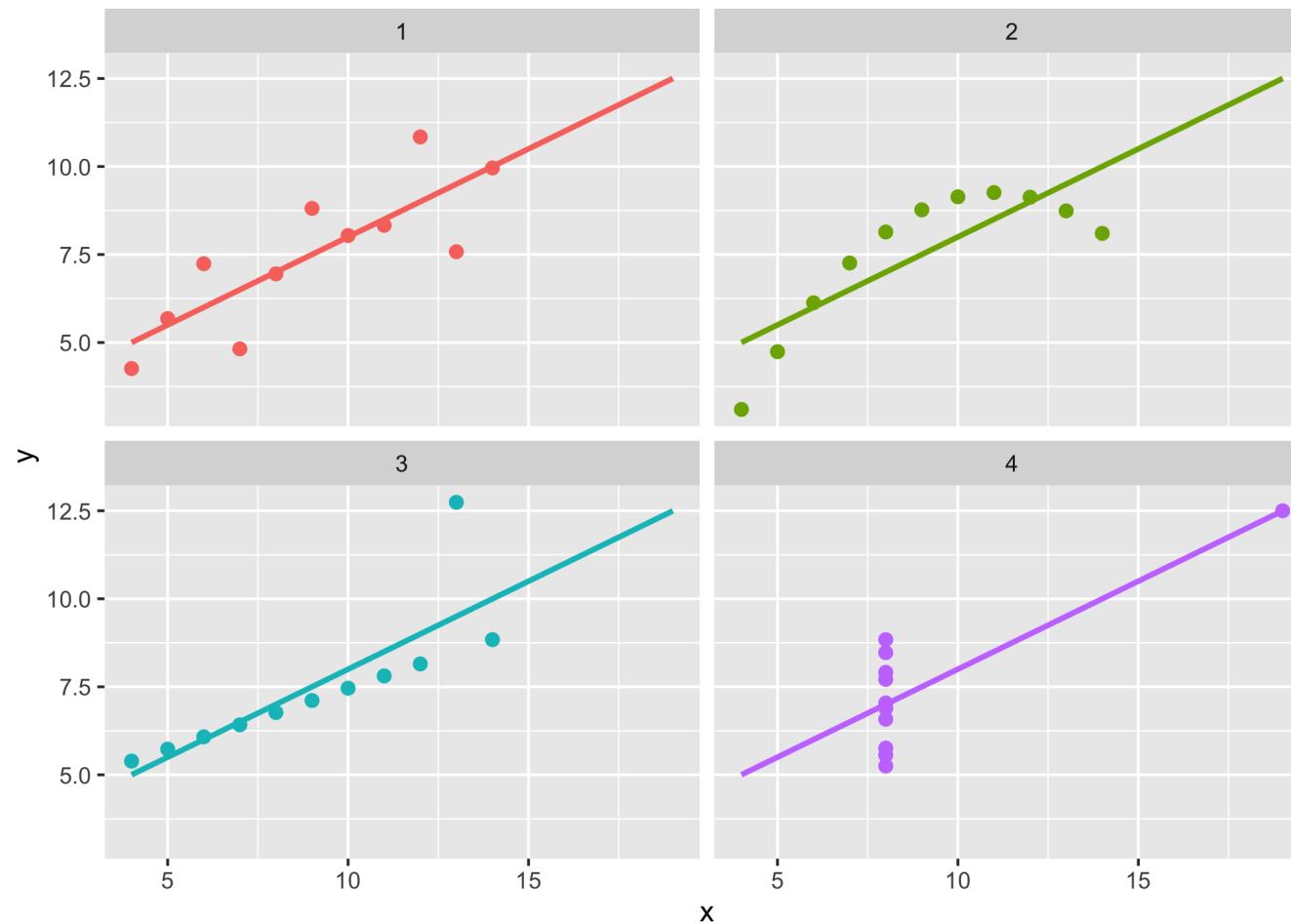
```
1 tidy_anscombe %>%
2   group_by(group) %>%
3   summarize(
4     mean_x = mean(x), mean_y = mean(y),
5     sd_x = sd(x), sd_y = sd(y),
6     cor = cor(x,y), .groups = "drop"
7   )
```

```
# A tibble: 4 × 6
  group mean_x mean_y  sd_x  sd_y    cor
  <chr>  <dbl>  <dbl> <dbl> <dbl> <dbl>
1 1        9    7.50  3.32  2.03  0.816
2 2        9    7.50  3.32  2.03  0.816
3 3        9    7.5    3.32  2.03  0.816
4 4        9    7.50  3.32  2.03  0.817
```

```

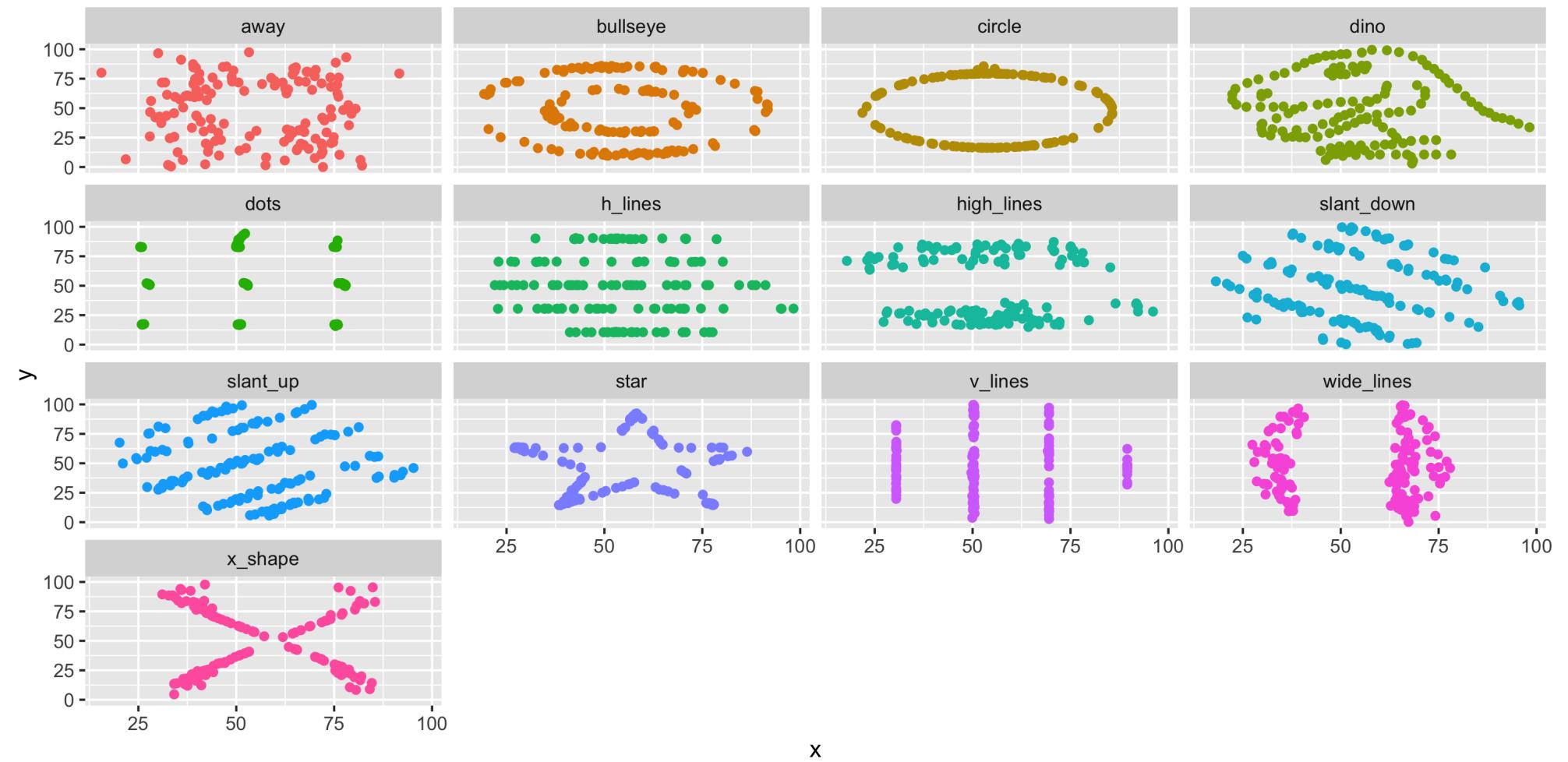
1 ggplot(tidy_anscombe, aes(x = x, y = y, color = as.factor(group))) +
2   geom_point(size=2) +
3   facet_wrap(~group) +
4   geom_smooth(method="lm", se=FALSE, fullrange=TRUE, formula = y~x) +
5   guides(color="none")

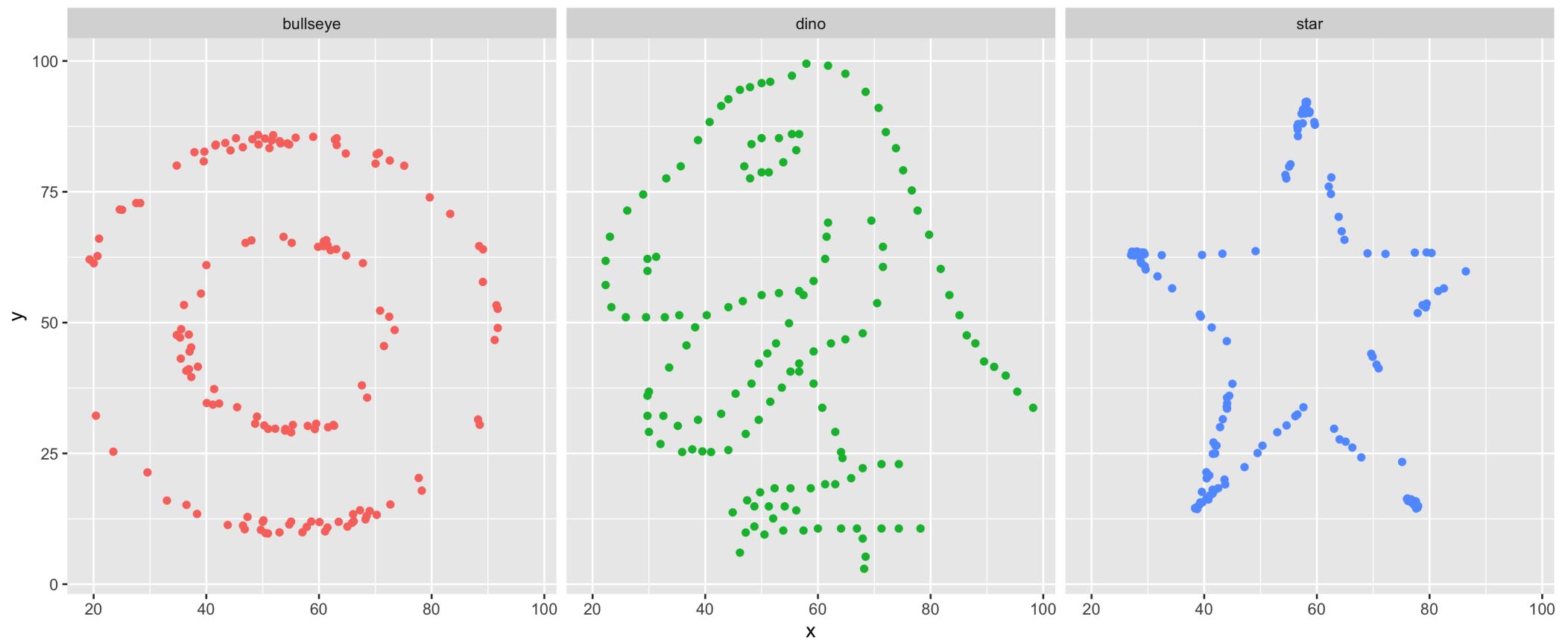
```



DatasauRus

```
1 ggplot(  
2   datasauRus::datasaurus_dozen,  
3   aes(x = x, y = y, color = dataset)  
4 ) +  
5   geom_point() +  
6   facet_wrap(~dataset) +  
7   guides(color="none")
```





```
1 datasauRus::datasaurus_dzen
```

```
# A tibble: 1,846 × 3
```

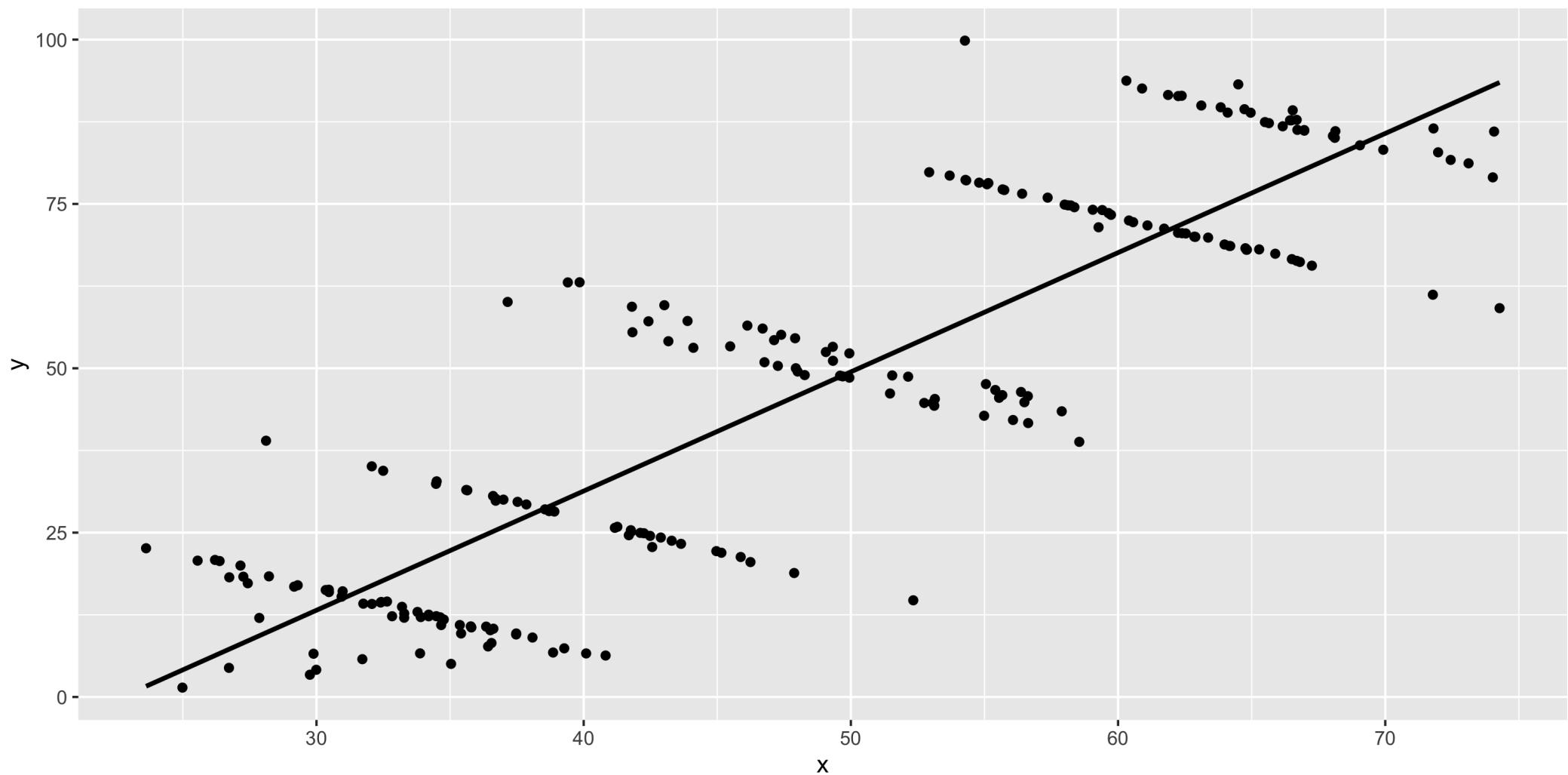
dataset	x	y
<chr>	<dbl>	<dbl>
1 dino	55.4	97.2
2 dino	51.5	96.0
3 dino	46.2	94.5
4 dino	42.8	91.4
5 dino	40.8	88.3
6 dino	38.7	84.9
7 dino	35.6	79.9
8 dino	33.1	77.6
9 dino	29.0	74.5
10 dino	26.2	71.4
# ... with 1,836 more rows		

```
1 datasauRus::datasaurus_dzen %>%
 2   group_by(dataset) %>%
 3   summarize(mean_x = mean(x), mean_y = mean(y),
 4             sd_x = sd(x), sd_y = sd(y),
 5             cor = cor(x,y), .groups = "drop")
```

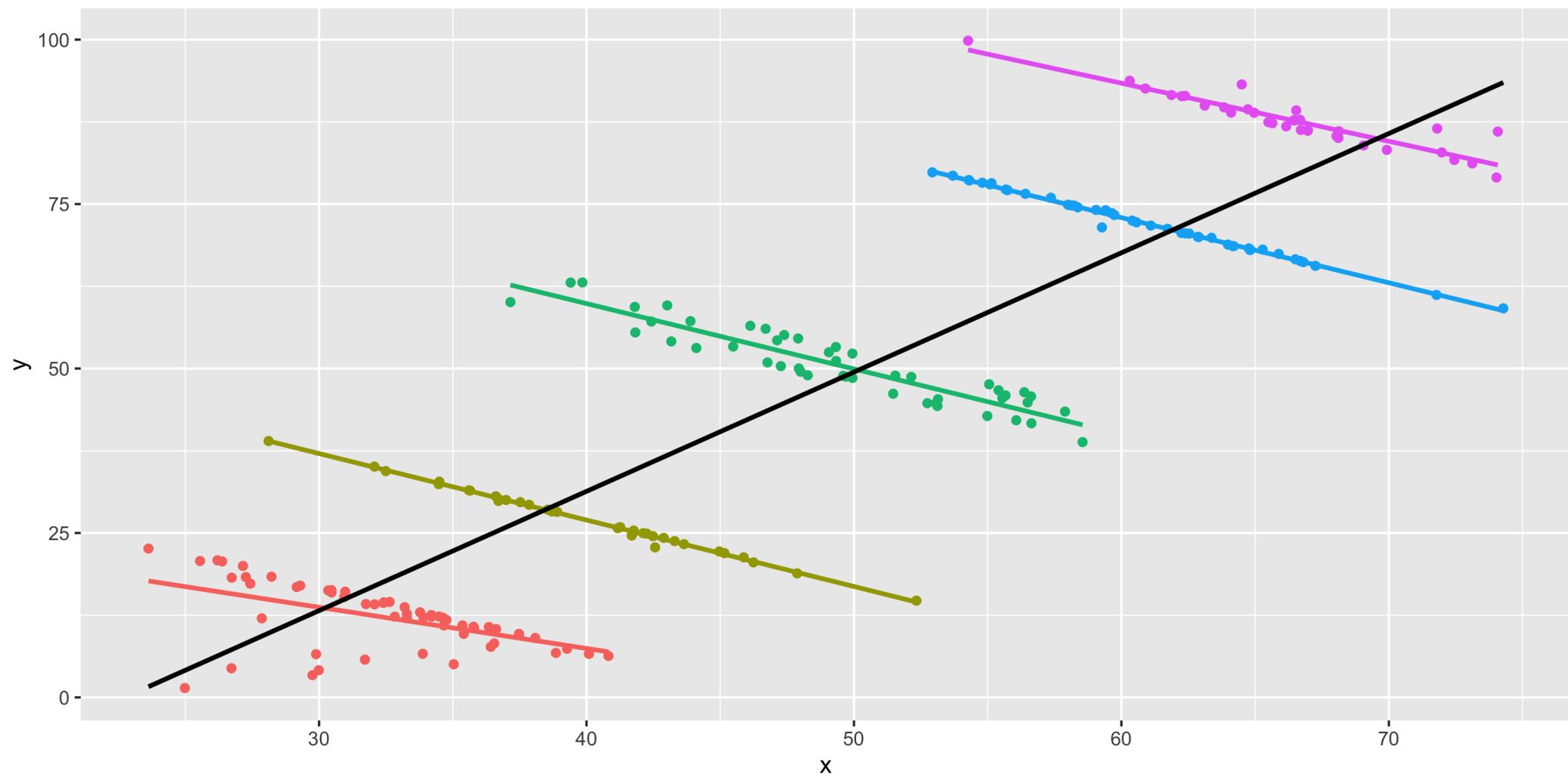
```
# A tibble: 13 × 6
```

dataset	mean_x	mean_y	sd_x	sd_y	cor
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1 away	54.3	47.8	16.8	26.9	-0.0641
2 bullseye	54.3	47.8	16.8	26.9	-0.0686
3 circle	54.3	47.8	16.8	26.9	-0.0683
4 dino	54.3	47.8	16.8	26.9	-0.0645
5 dots	54.3	47.8	16.8	26.9	-0.0603
6 h_lines	54.3	47.8	16.8	26.9	-0.0617
7 high_lines	54.3	47.8	16.8	26.9	-0.0685
8 slant_down	54.3	47.8	16.8	26.9	-0.0690
9 slant_up	54.3	47.8	16.8	26.9	-0.0686
10 star	54.3	47.8	16.8	26.9	-0.0630
11 v_lines	54.3	47.8	16.8	26.9	-0.0694
12 wide_lines	54.3	47.8	16.8	26.9	-0.0666
13 x_shape	54.3	47.8	16.8	26.9	-0.0656

Simpson's Paradox

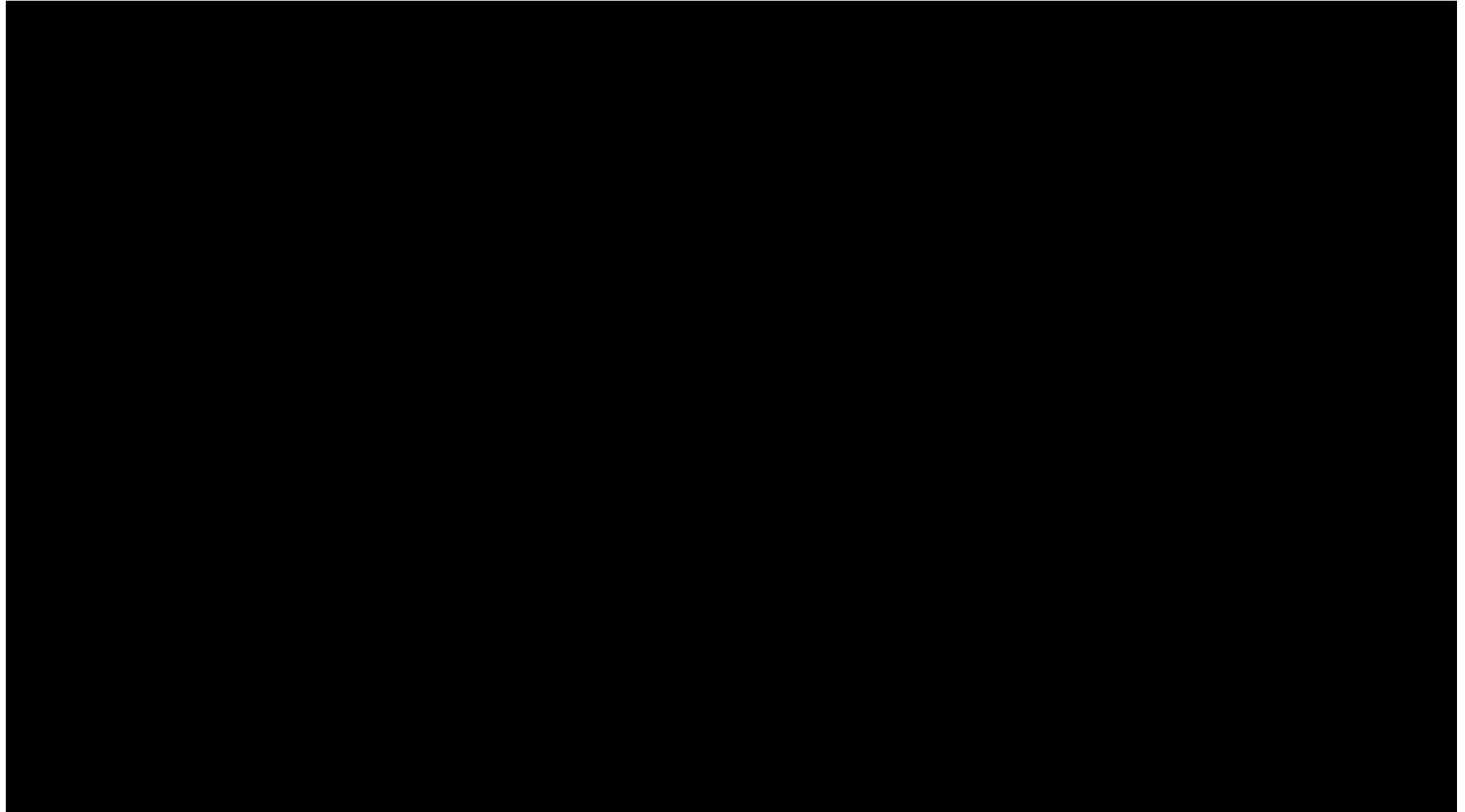


Simpson's Paradox

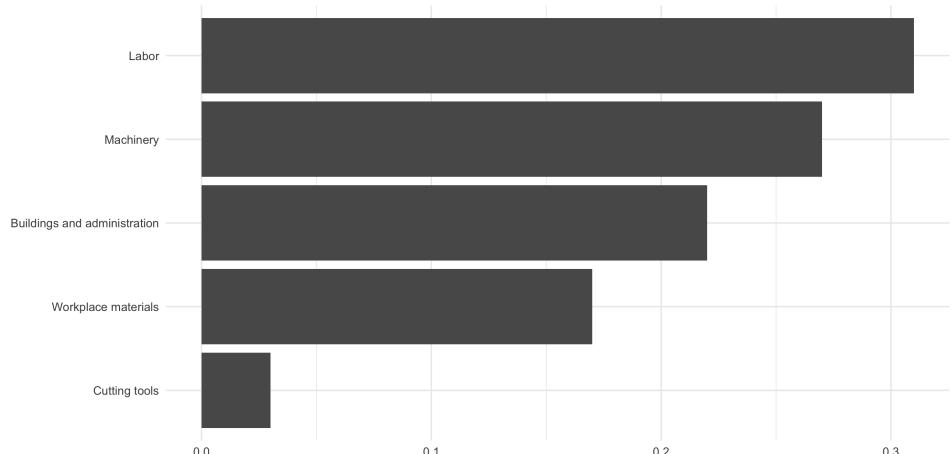
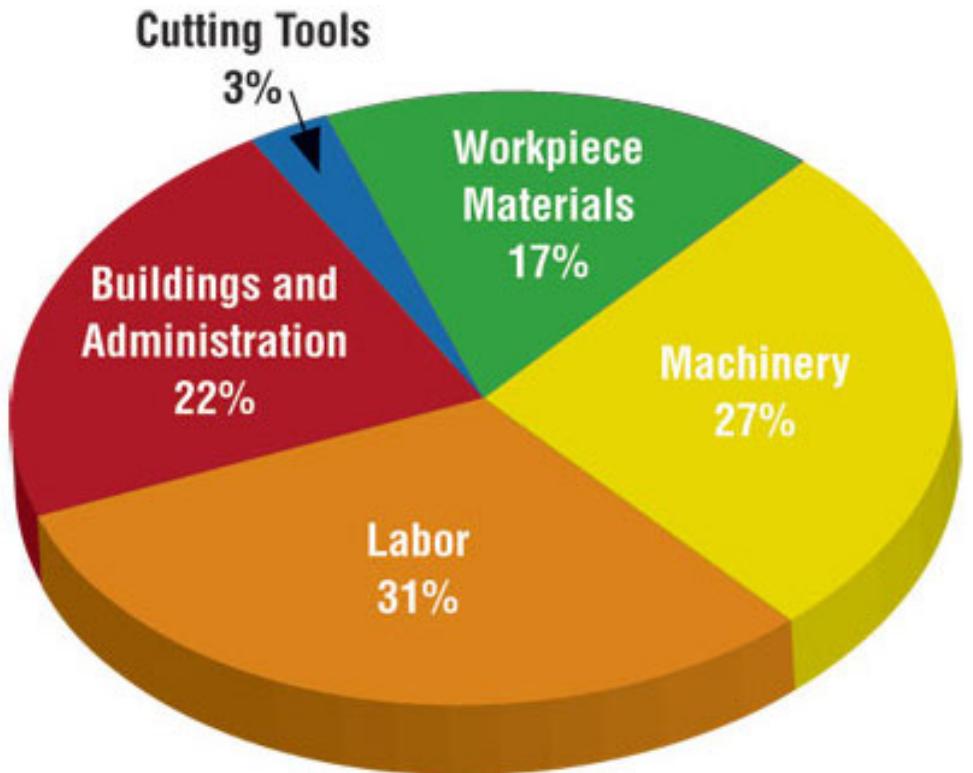


Designing effective visualizations

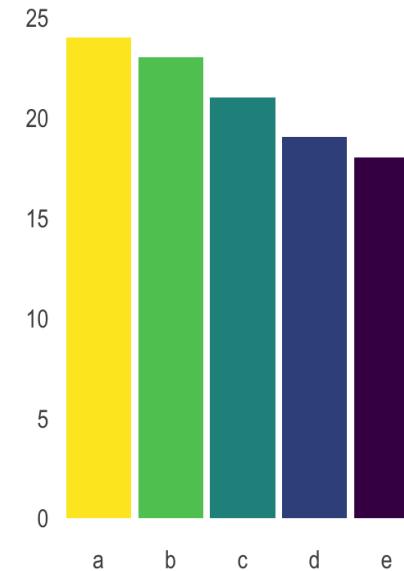
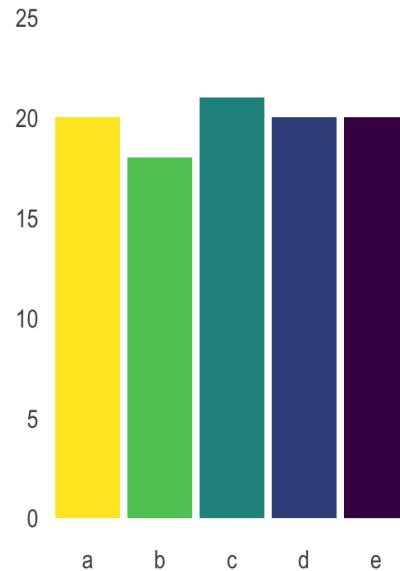
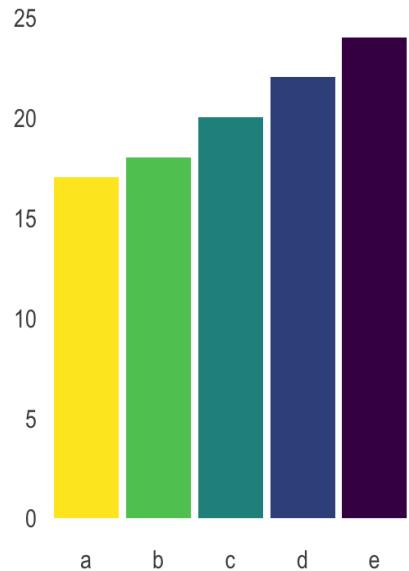
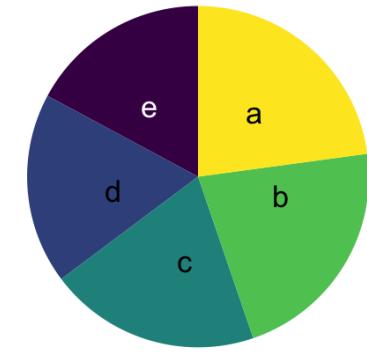
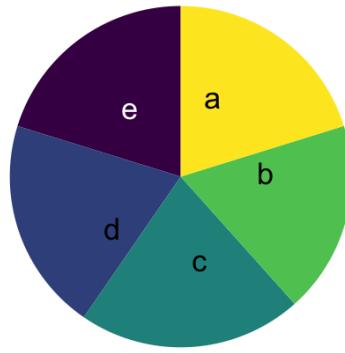
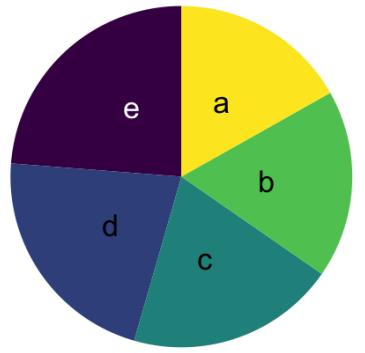
Gapminder



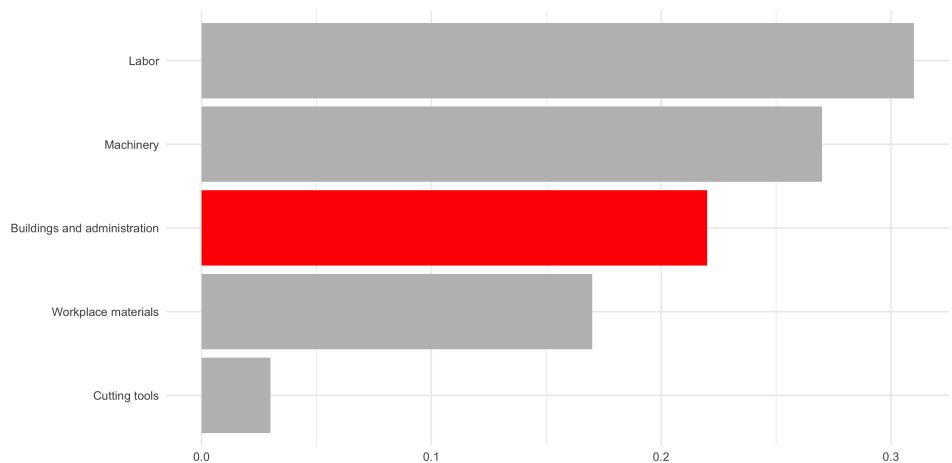
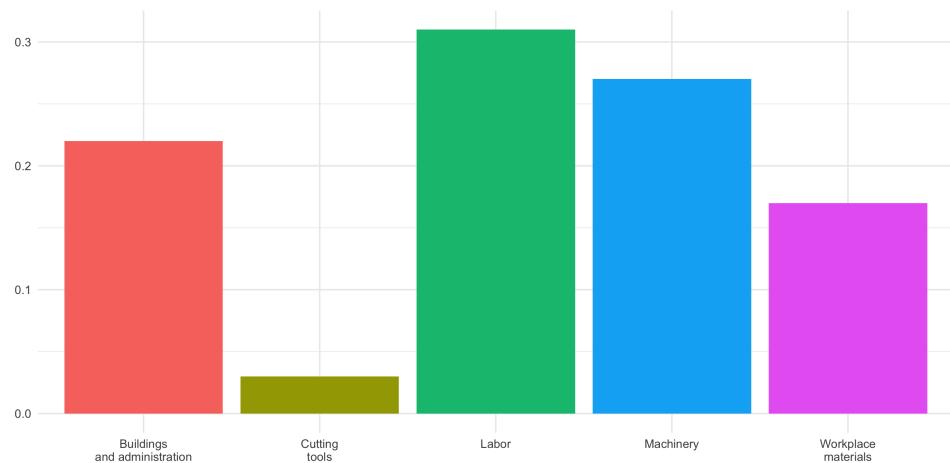
Keep it simple



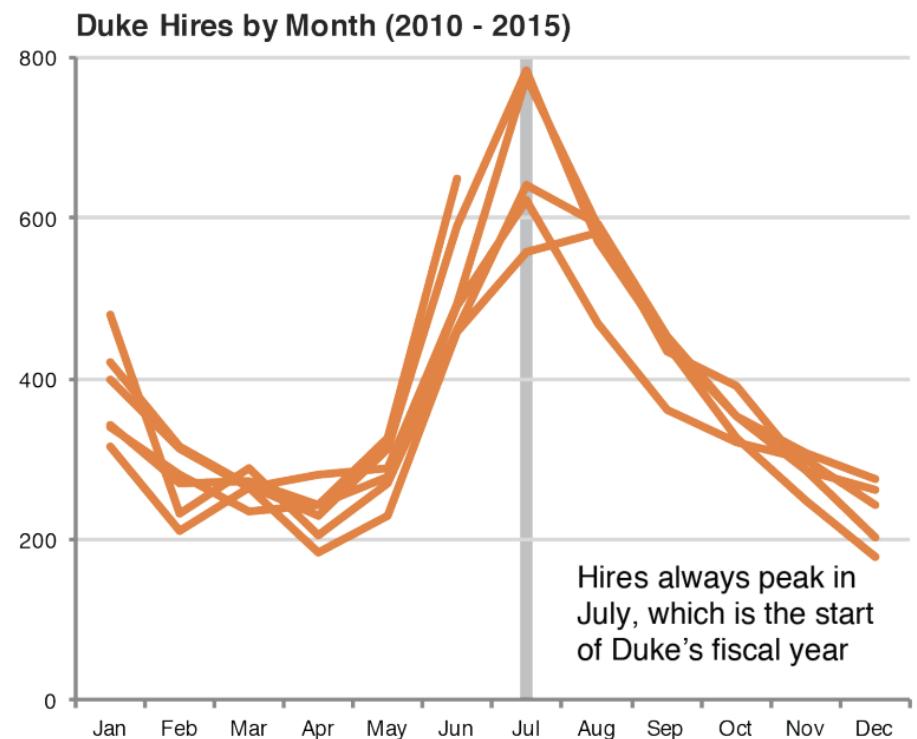
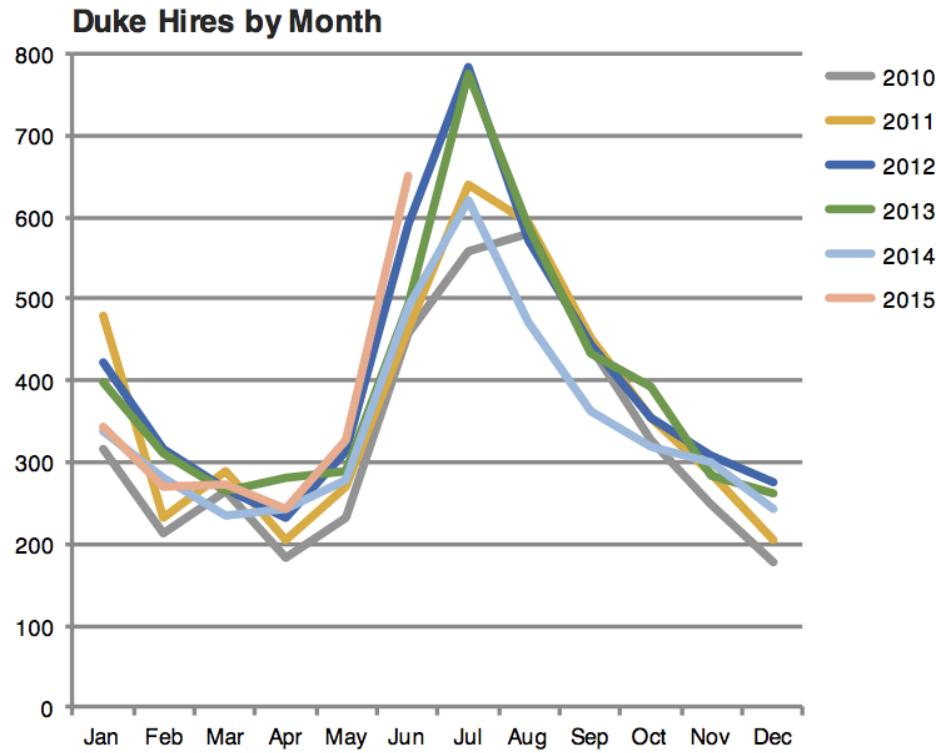
Judging relative area



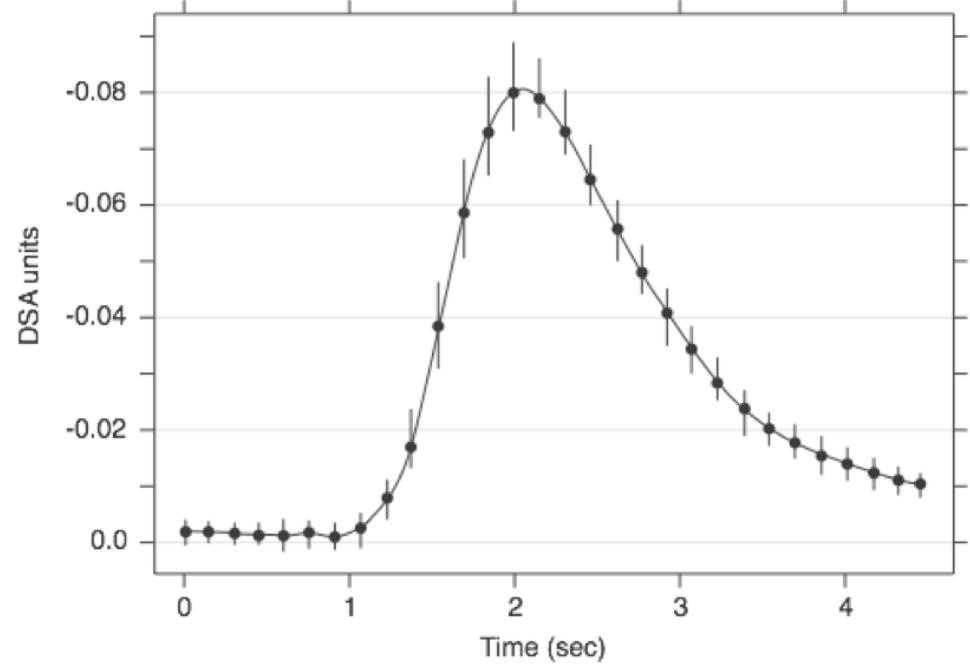
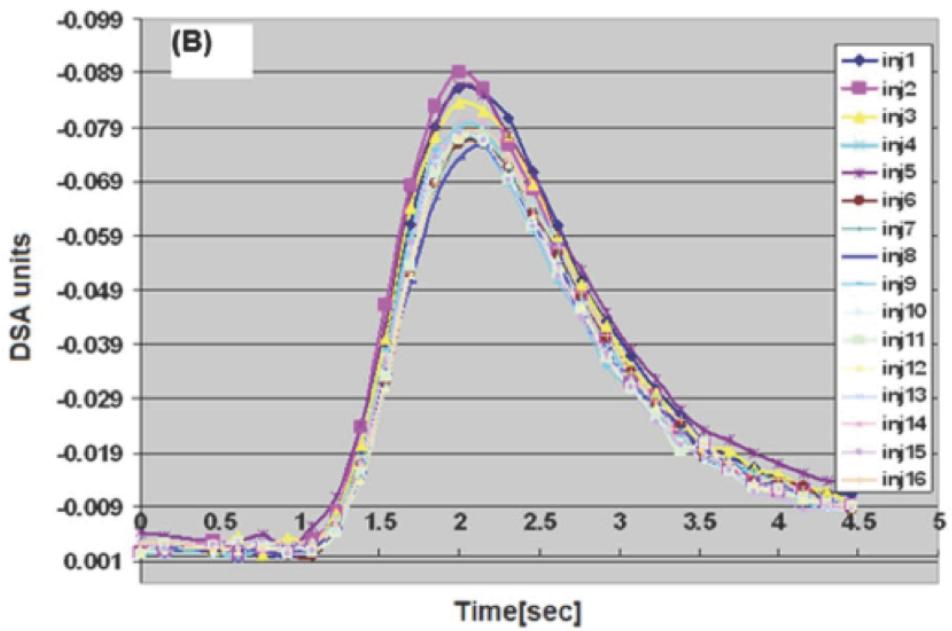
Use color to draw attention



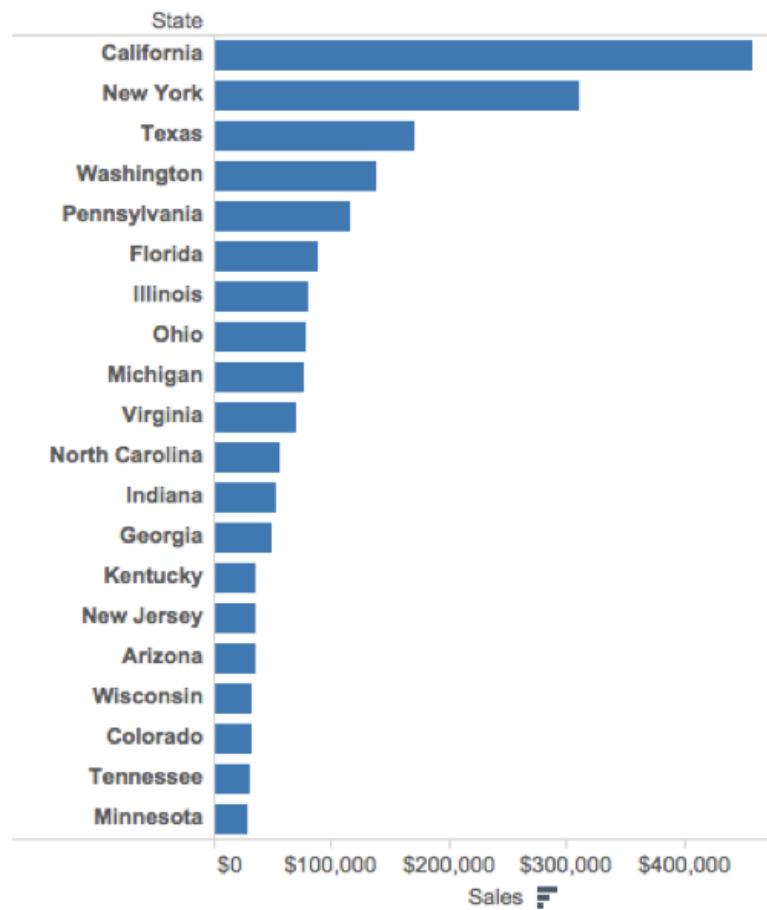
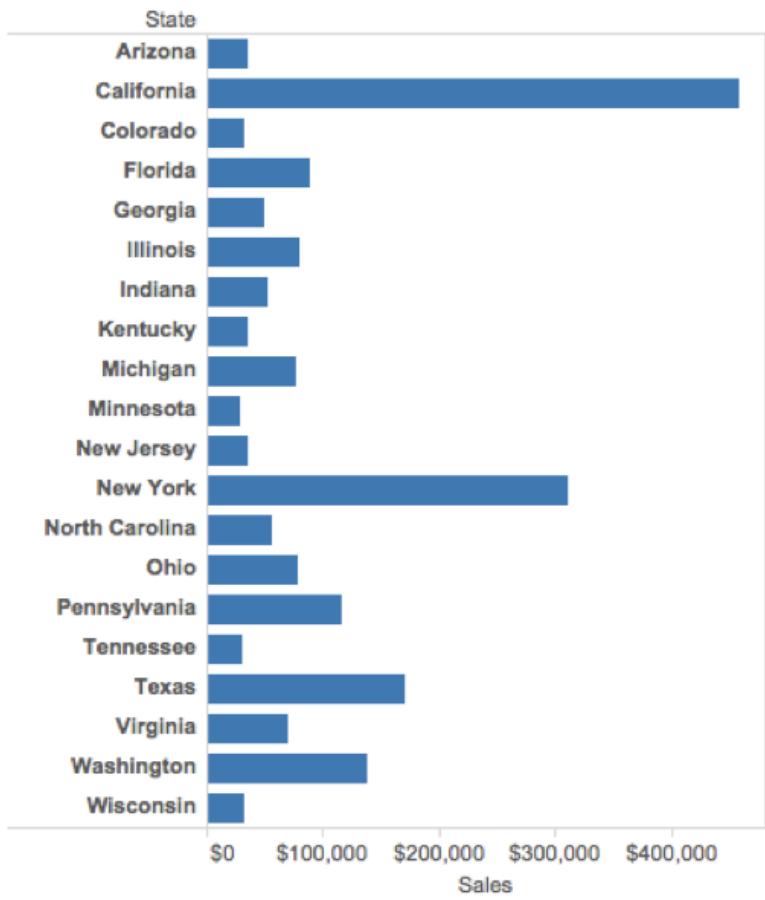
Tell a story



Leave out non-story details



Ordering matter



Clearly indicate missing data

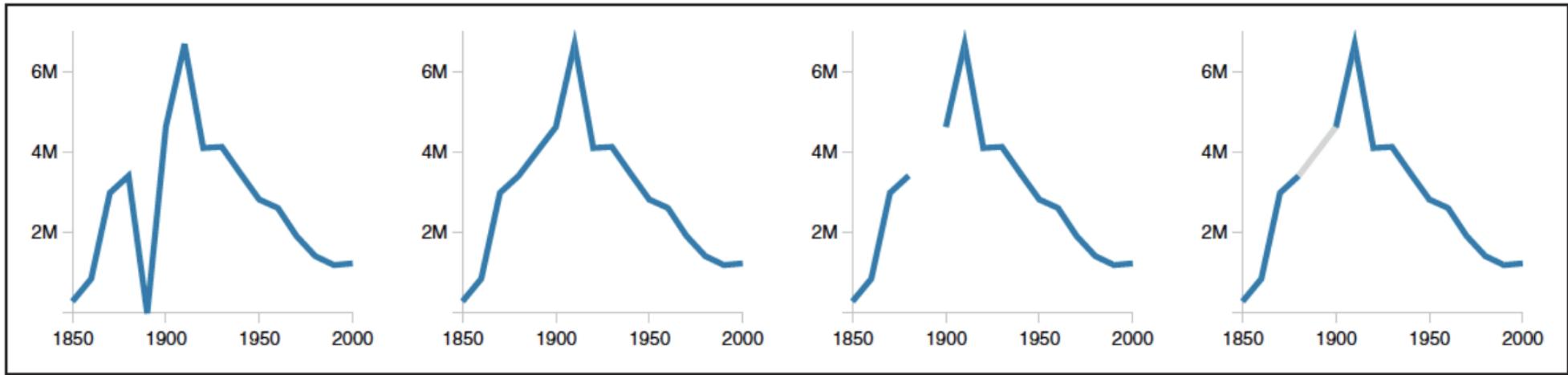
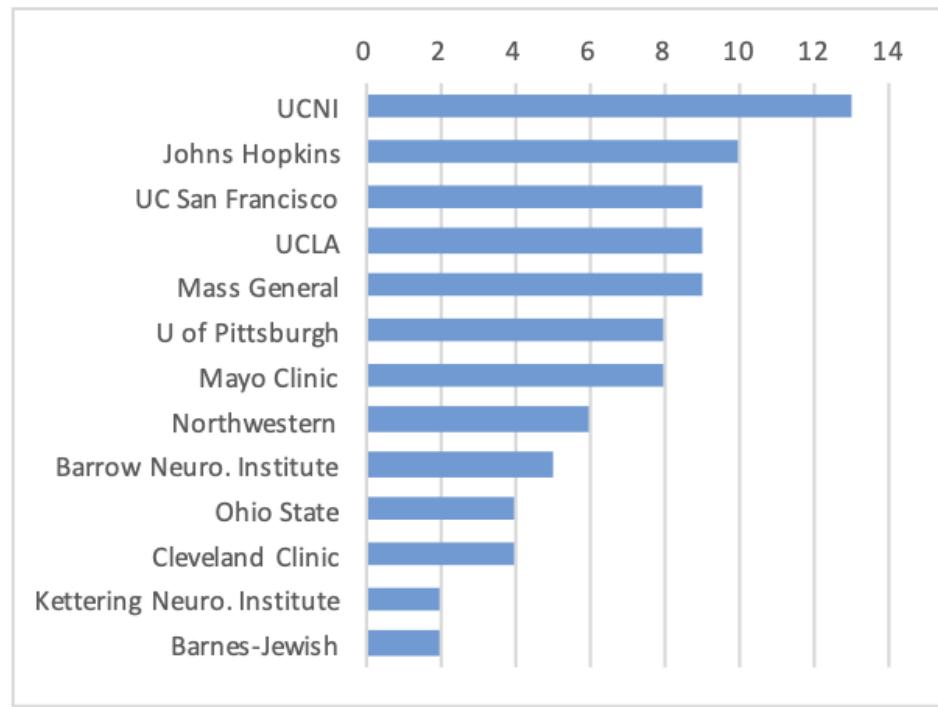
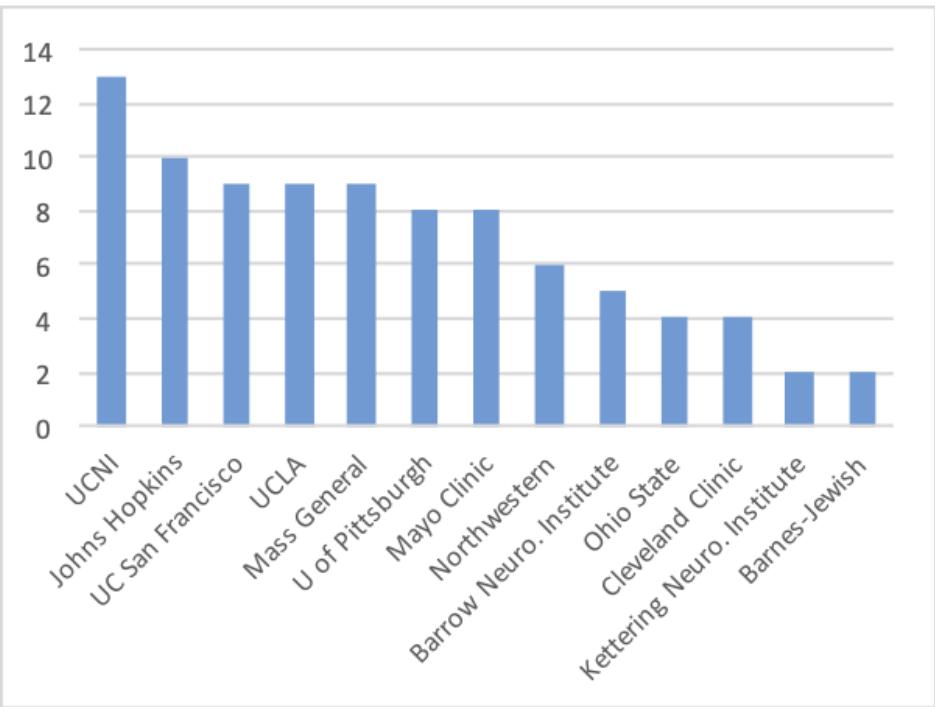


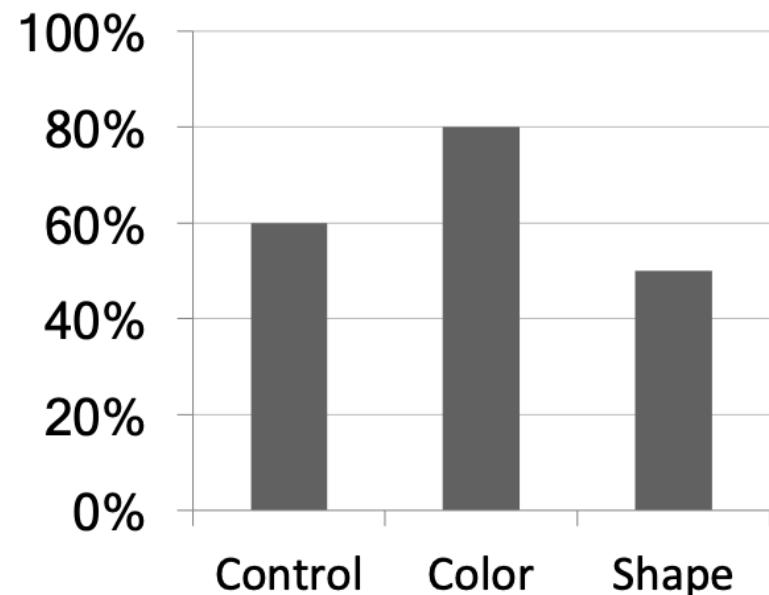
Figure 4. Alternative representations of missing data in a line chart. The data are U.S. census counts of people working as 'Farm Laborers'; values from 1890 are missing due to records being burned in a fire. (a) Missing data is treated as a zero value. (b) Missing data is ignored, resulting in a line segment that interpolates the missing value. (c) Missing data is omitted from the chart. (d) Missing data is explicitly interpolated and rendered in gray.

Reduce cognitive load

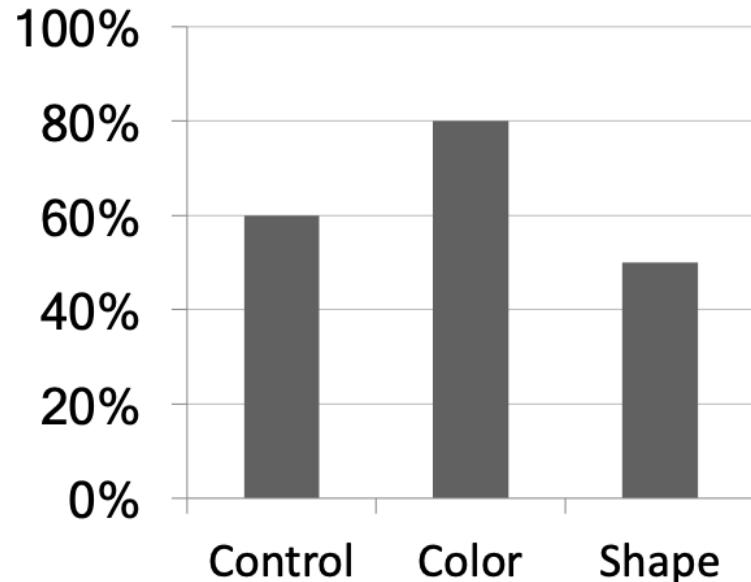


Use descriptive titles

Accuracy versus Color and Shape



Accuracy Improved by Color, not Shape

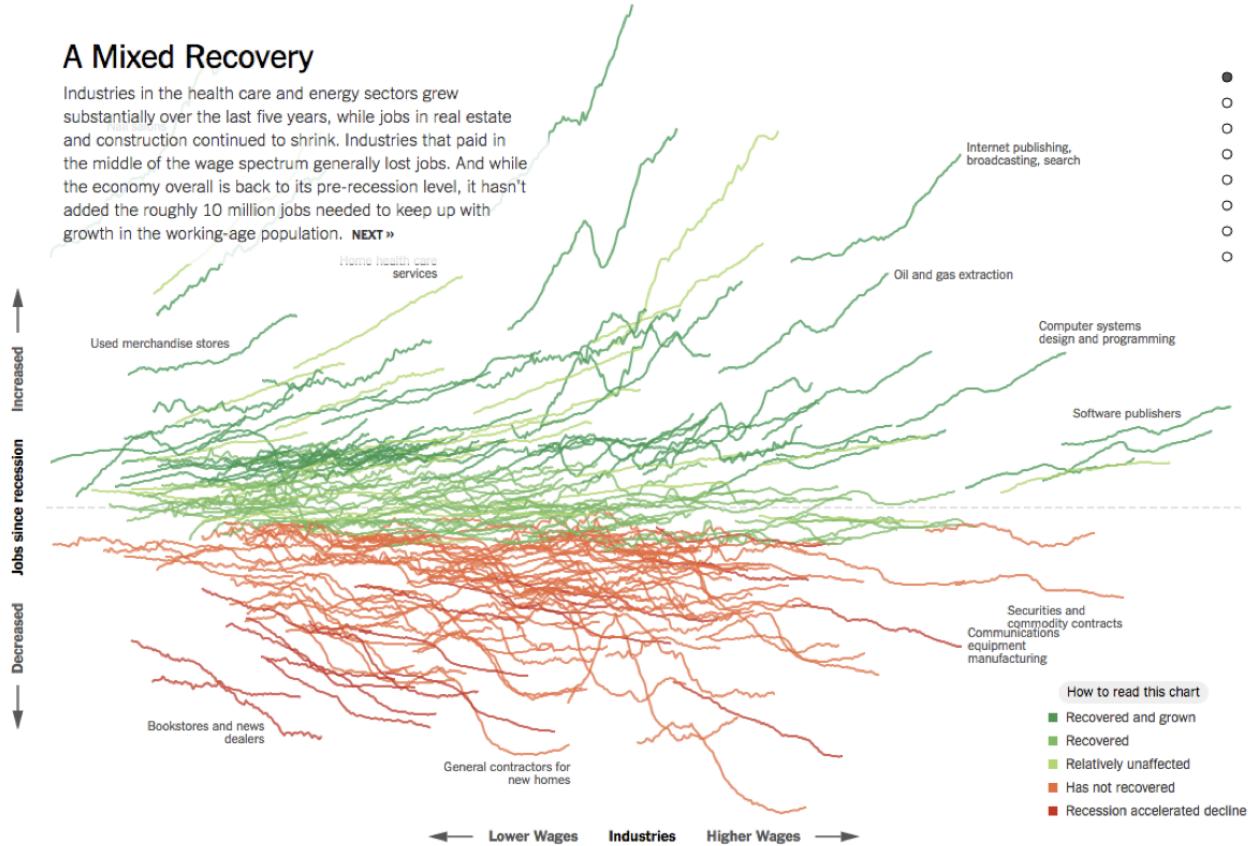


Annotate figures

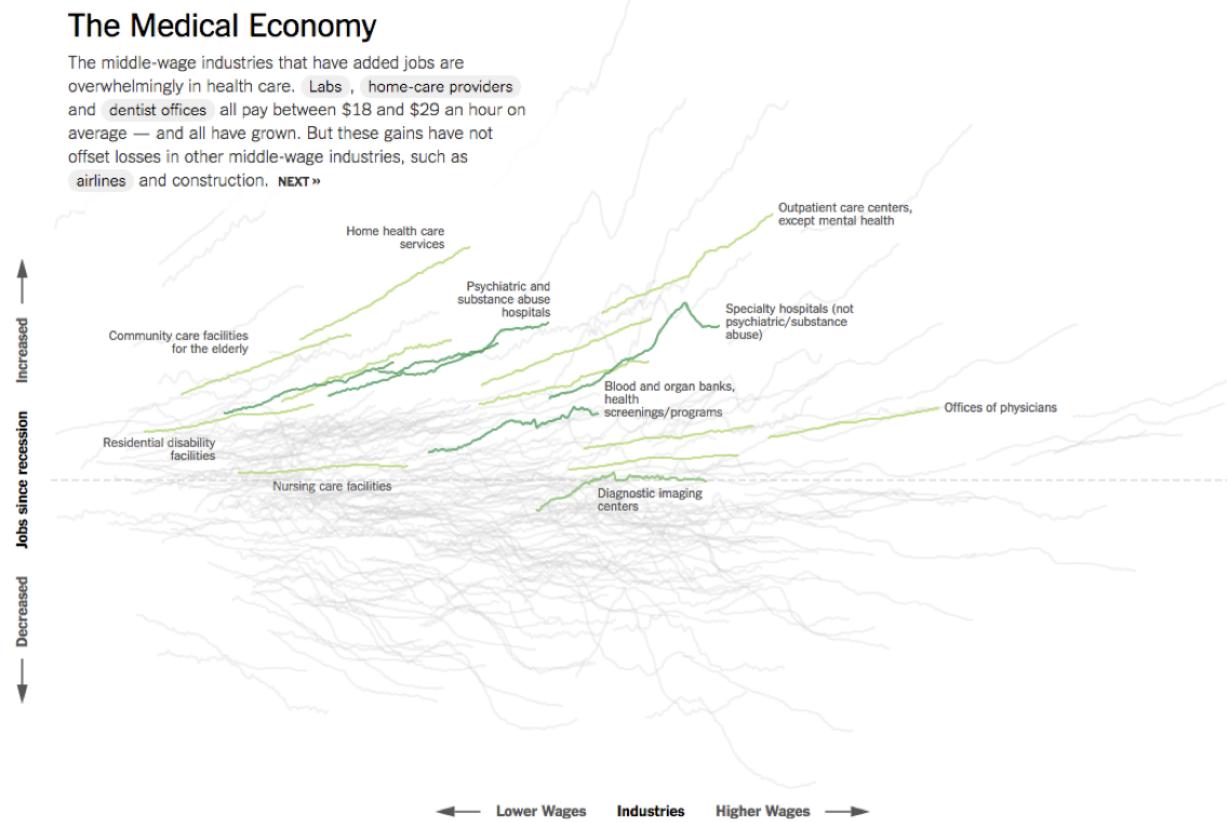
AAPL stock example



All of the data doesn't tell a story



All of the data doesn't tell a story



All of the data doesn't tell a story

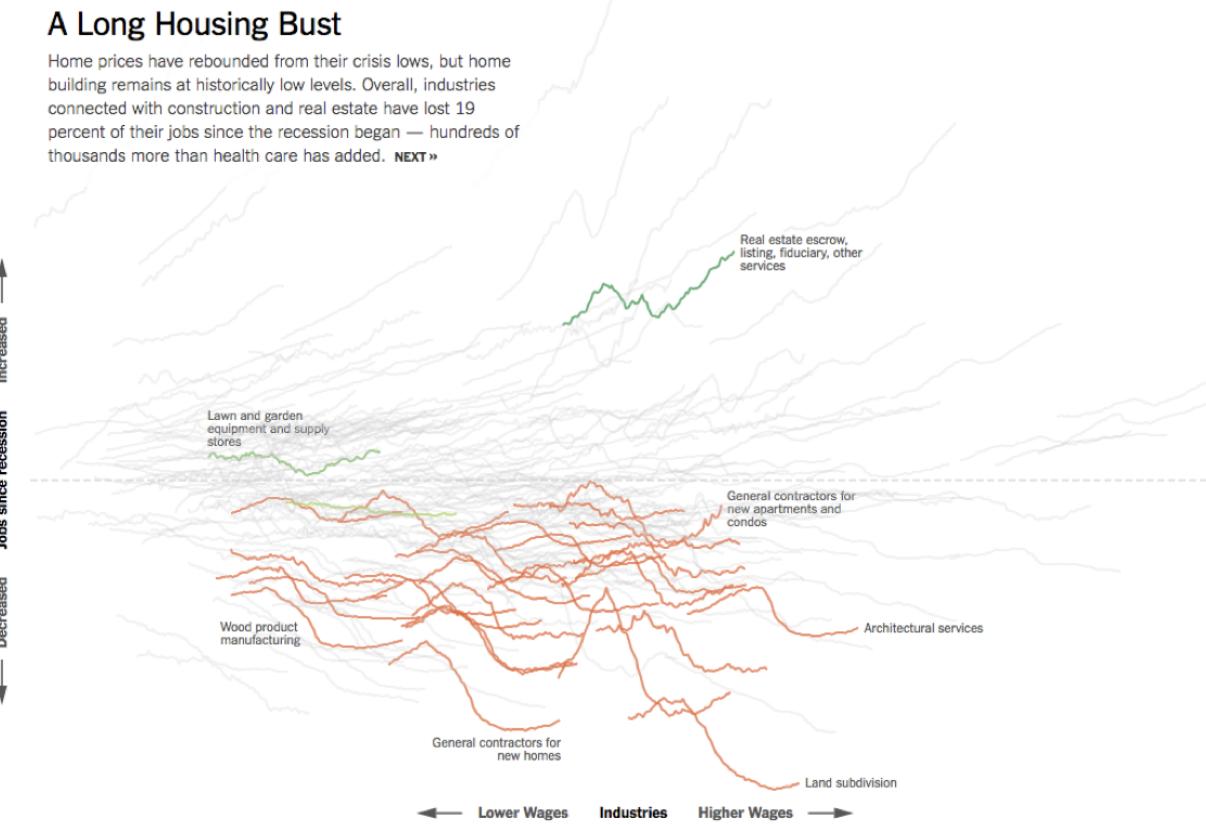
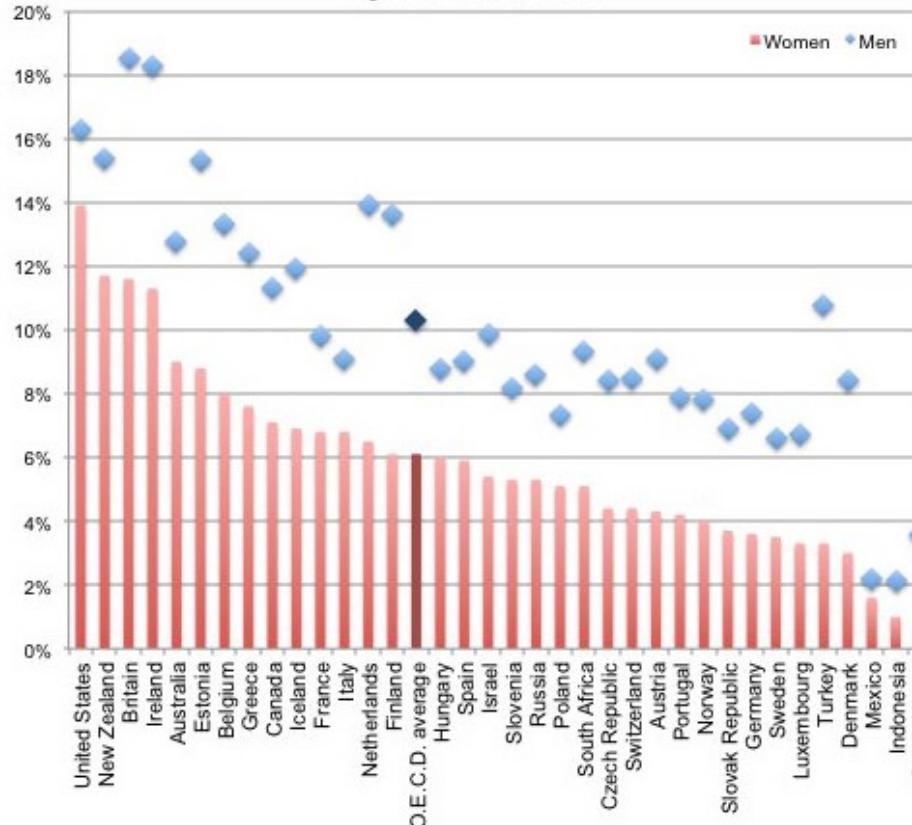


Chart Remakes / Makeovers

The Why Axis - Gender Gap

Percentage of Employed Who Are Senior Managers,
by Gender, 2008



The Why Axis - BLS

Job openings in November 2012

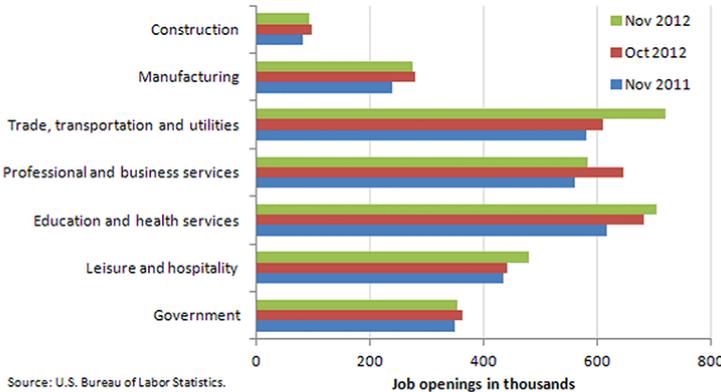
JANUARY 11, 2013

There were 3.7 million job openings on the last business day of November 2012, unchanged from October 2012. In November 2011 there were 3.3 million job openings.

CHART IMAGE

CHART DATA

Job openings by industry, November 2011, October 2012 and November 2012, seasonally adjusted



Source: U.S. Bureau of Labor Statistics.

From November 2011 to November 2012, job openings increased most in retail trade (144,000, within the trade, transportation and utilities industry) and health care and social assistance (91,000, within the education and health services industry).

Government job openings increased the least, by 6,000.

These data are from the [Job Openings and Labor Turnover Survey](#). Data for the most recent month are preliminary and subject to revision. For additional information, see Job Openings and Labor Turnover — November 2012" ([HTML](#)) ([PDF](#)), news release USDL-13-0015. More charts featuring data on job openings, hires, and employment separations can be found in [Job Openings and Labor Turnover Survey Highlights: November 2012](#) ([PDF](#)).

Other Resources

- Duke Library - Center for Data and Visualization Sciences -
<https://library.duke.edu/data/>
- Tidy tuesday - <https://github.com/rfordatascience/tidytuesday>
- Flowing data - <https://flowingdata.com/>
- Twitter - #dataviz, #tidytuesday
- Books:
 - Wickham, Navarro, Pedersen. *ggplot2: Elegant Graphics for Data Analysis.* (in progress) 3rd edition. Springer, 2021.
 - Wilke. *Fundamentals of Data Visualization.* O'Reilly Media, 2019.
 - Healy. *Data Visualization: A Practical Introduction.* Princeton University Press, 2018.
 - Tufte. *The visual display of quantitative information.* 2nd edition. Connecticut Graphics Press, 2015.

Acknowledgments

Above materials are derived in part from the following sources:

- Visualization training materials developed by Angela Zoss and Eric Monson, Duke DVS