Data visualization and a grammar of graphics

Overview

A brief history of data visualization

The grammar of graphics and ggplot

Joining data tables

A very brief history of data visualization

The Golden Age of Statistical Graphics, Friendly 2008

Data visualization

What are some reasons we visualize data rather than just reporting statistics?

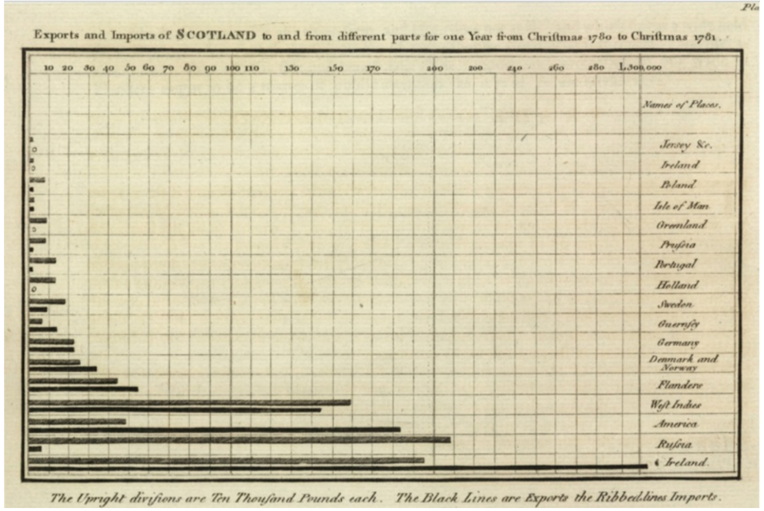
Whatever relates to extent and quantity may be represented by geometrical figures. Statistical projections which speak to the senses without fatiguing the mind, possess the advantage of fixing the attention on a great number of important facts.

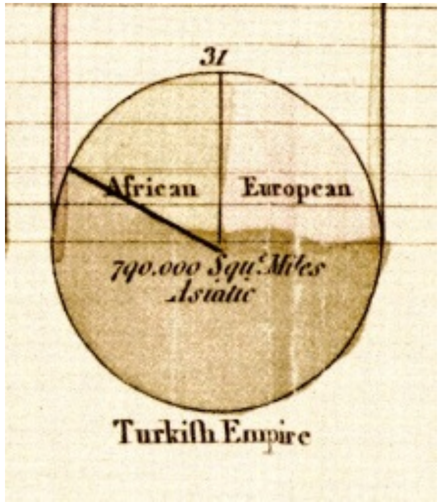
—Alexander von Humboldt, 1811

A very brief history of data visualization

The age of modern statistical graphs began around the beginning of the 19th century

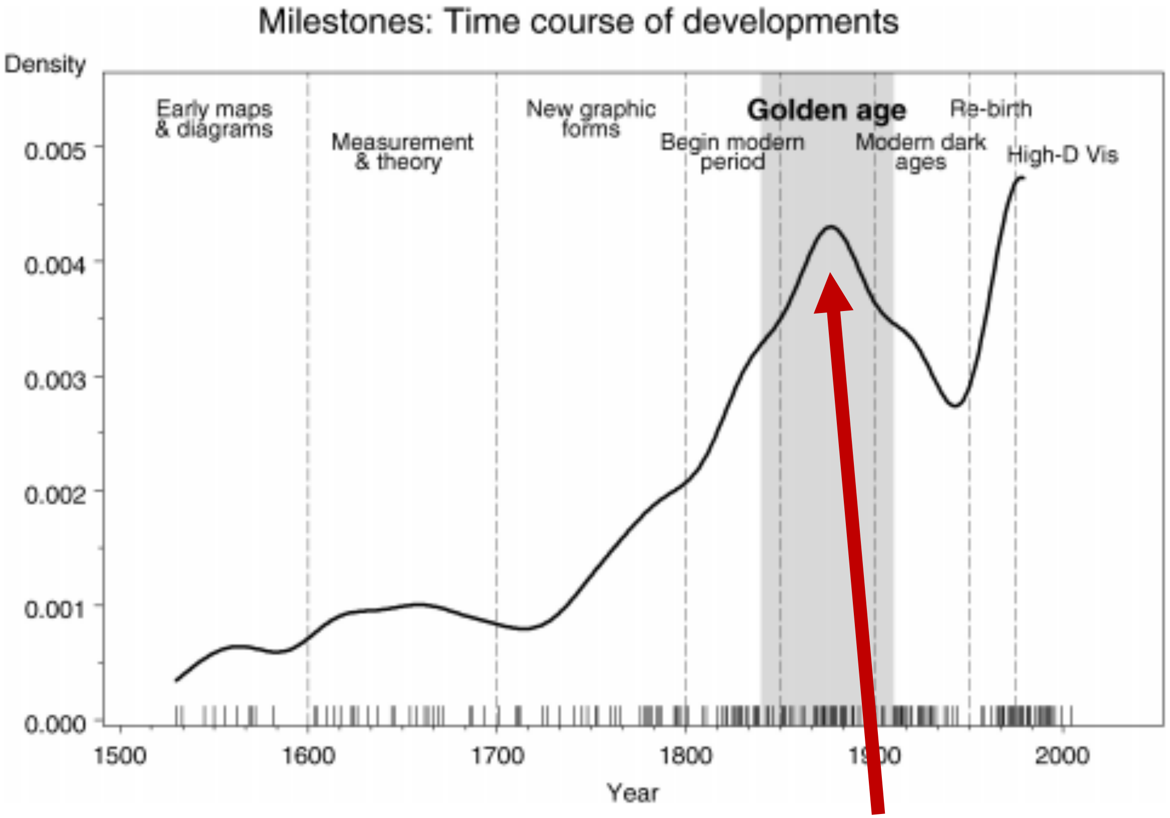
William Playfair (1759-1823) credited with inventing the line graph, bar chart and pie chart





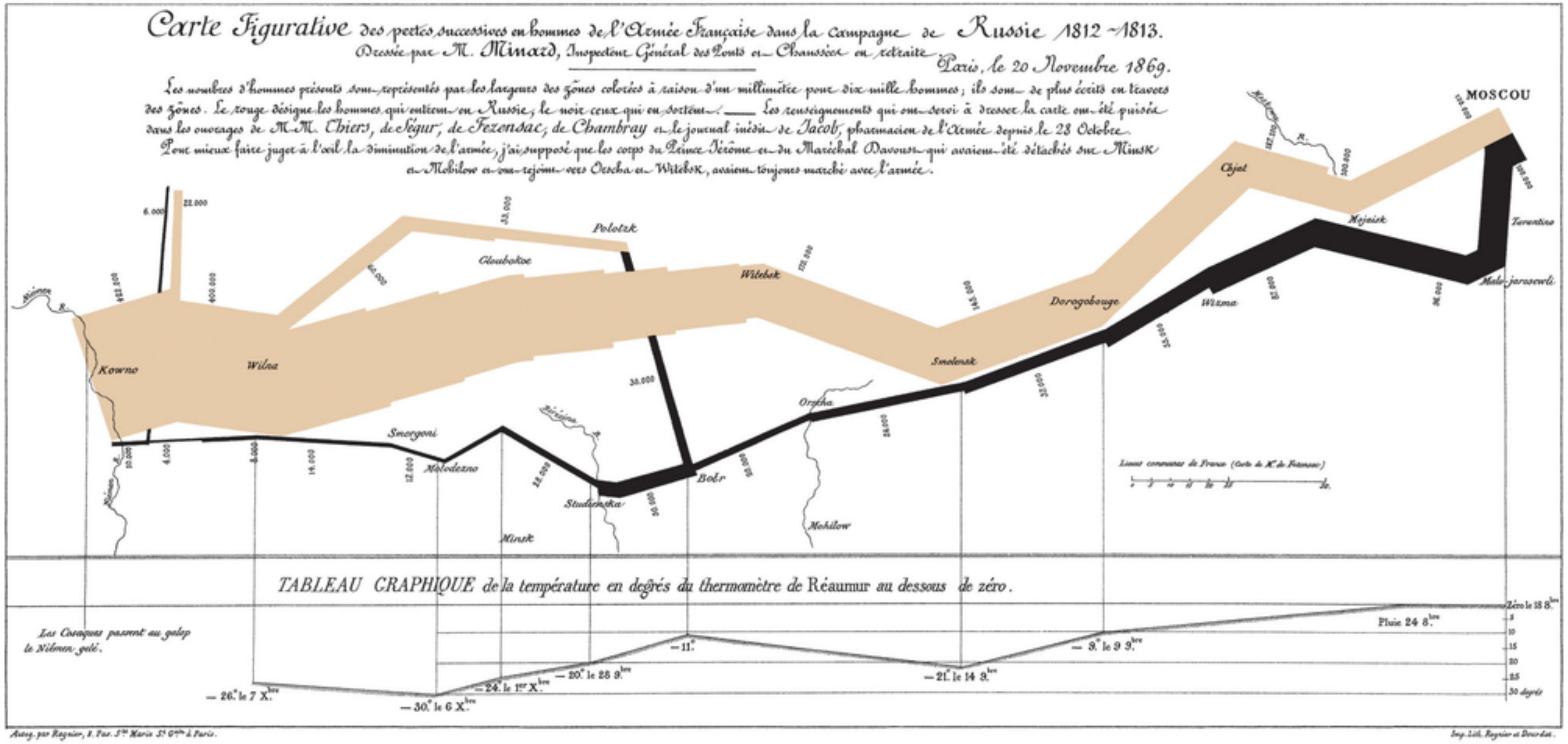
A very brief history of data visualization

According to Friendly, statistical graphics researched its golden age between 1850-1900



A very brief history of data visualization

Joseph Minard (1781-1870)



Carte Figurative des pertes successives en hommes de l'Armée Française dans la campagne de Russie 1812-1813.
Dressée par M. Minard, Inspecteur Général des Ponts et Chaussées en retraite.
Paris, le 20 Novembre 1869.

Les nombres d'hommes présents sont représentés par les largeurs des zones colorées à raison d'un millimètre pour dix mille hommes; ils sont de plus écrits en travers des zones. Le rouge désigne les hommes qui entrent en Russie, le noir ceux qui en sortent. —— Les renseignements qui ont servi à dresser la carte ont été puisés dans les ouvrages de M.M. Thiers, de Ségur, de Fezensac, de Chambray et le journal inédit de Jacob, pharmacien de l'Armée depuis le 28 Octobre.

Pour mieux faire juger à l'œil la diminution de l'armée, j'ai supposé que les corps du Prince Jérôme et du Maréchal Davoust qui avaient été détachés sur Minsk et Mobilow et ont rejoint vers Orscha et Witebsk, avaient toujours marché avec l'armée.

TABLEAU GRAPHIQUE de la température en degrés du thermomètre de Réaumur au dessous de zéro.

Les Cosaques passent au galop le Niémen gelé.

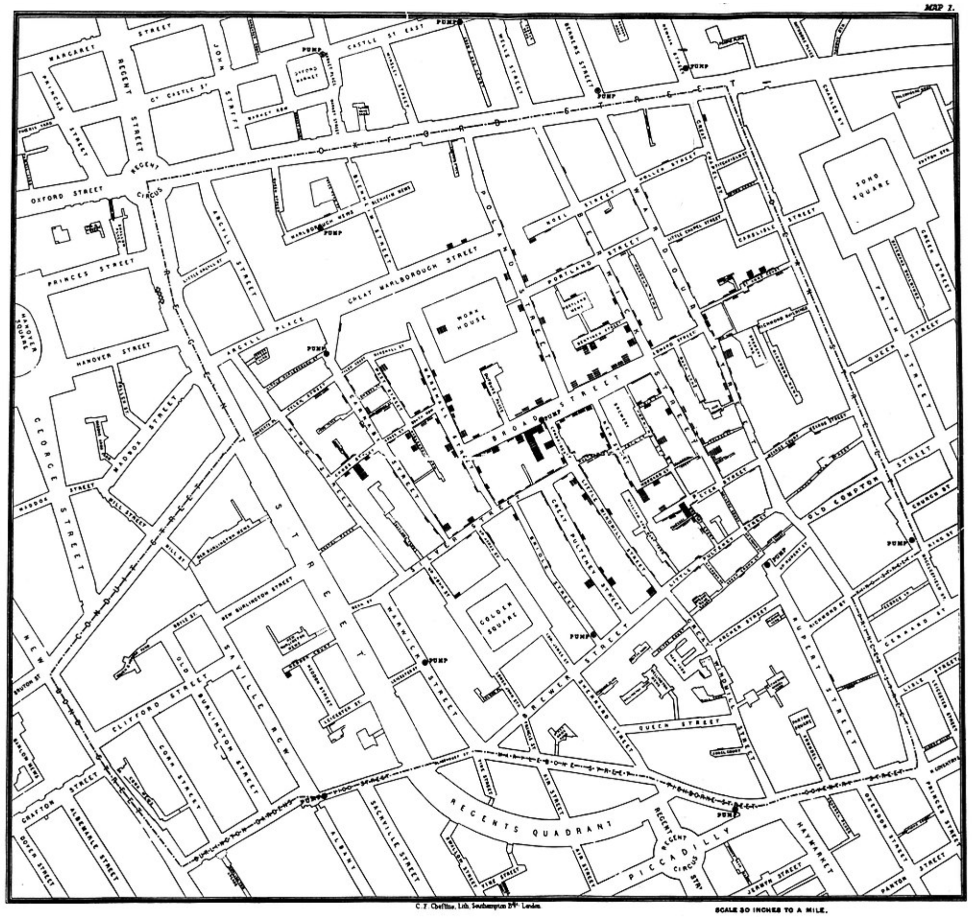
Autog. par Regnier, 8. Pas. Ste Marie St Gain à Paris.

Imp. Lith. Regnier et Dourdet.

Map of Napoleon's march on Russia

A very brief history of data visualization

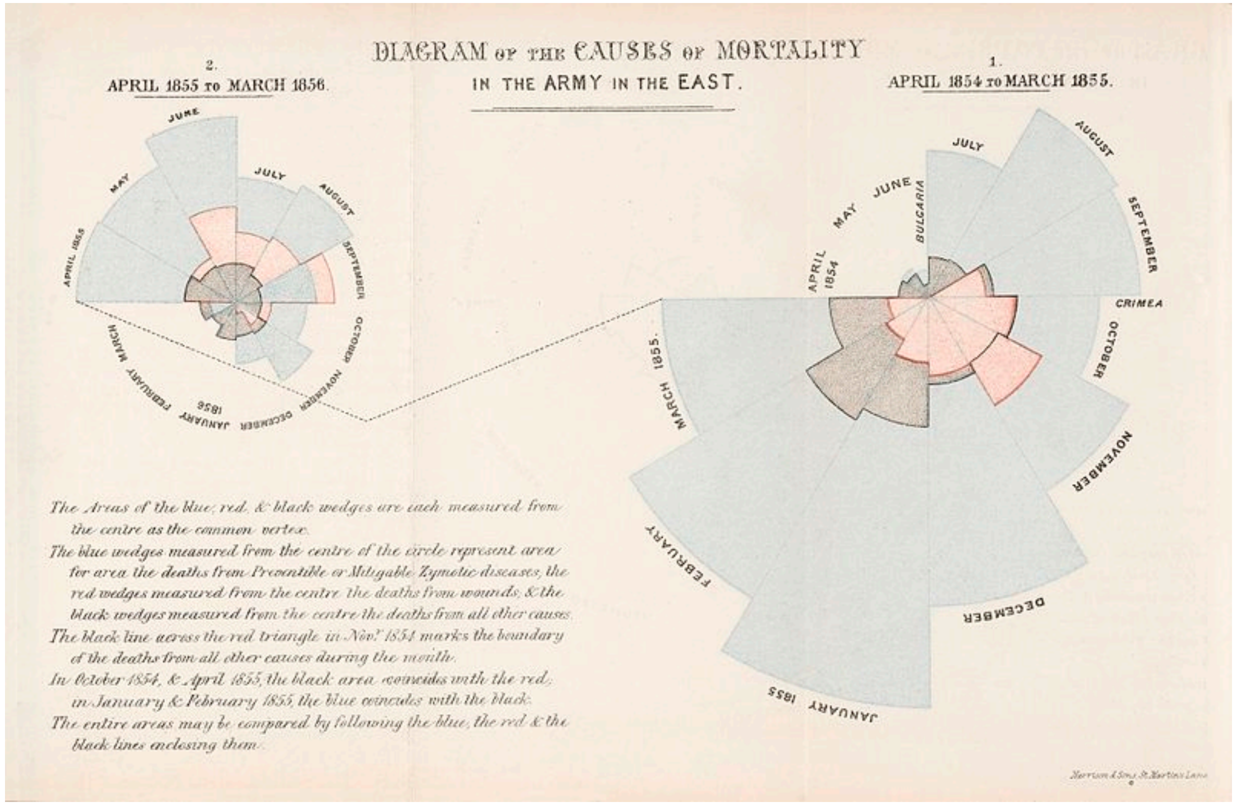
John Snow (1813-1858)



Clusters of cholera cases in London epidemic of 1854

A very brief history of data visualization

Florence Nightingale (1820-1910)



The Areas of the blue, red, & black wedges are each measured from the centre as the common vertex.

The blue wedges measured from the centre of the circle represent area for area the deaths from Preventible or Mitigable Zymotic diseases, the red wedges measured from the centre the deaths from wounds, & the black wedges measured from the centre the deaths from all other causes.

The black line across the red triangle in Nov.r 1854 marks the boundary of the deaths from all other causes during the month.

In October 1854, & April 1855, the black area coincides with the red; in January & February 1855, the blue coincides with the black.

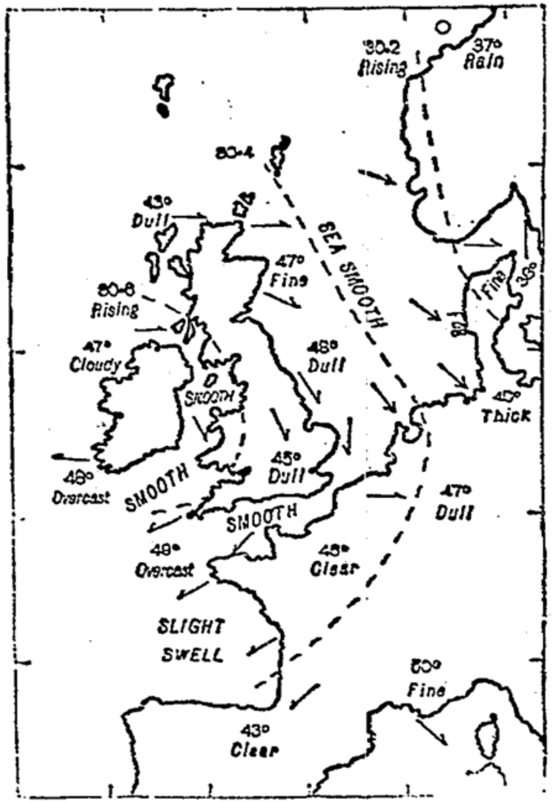
The entire areas may be compared by following the blue, the red & the black lines enclosing them.

Diagram of the causes of mortality in the army in the east

A very brief history of data visualization

Francis Galton (1822-1911)

WEATHER CHART, MARCH 31, 1875.

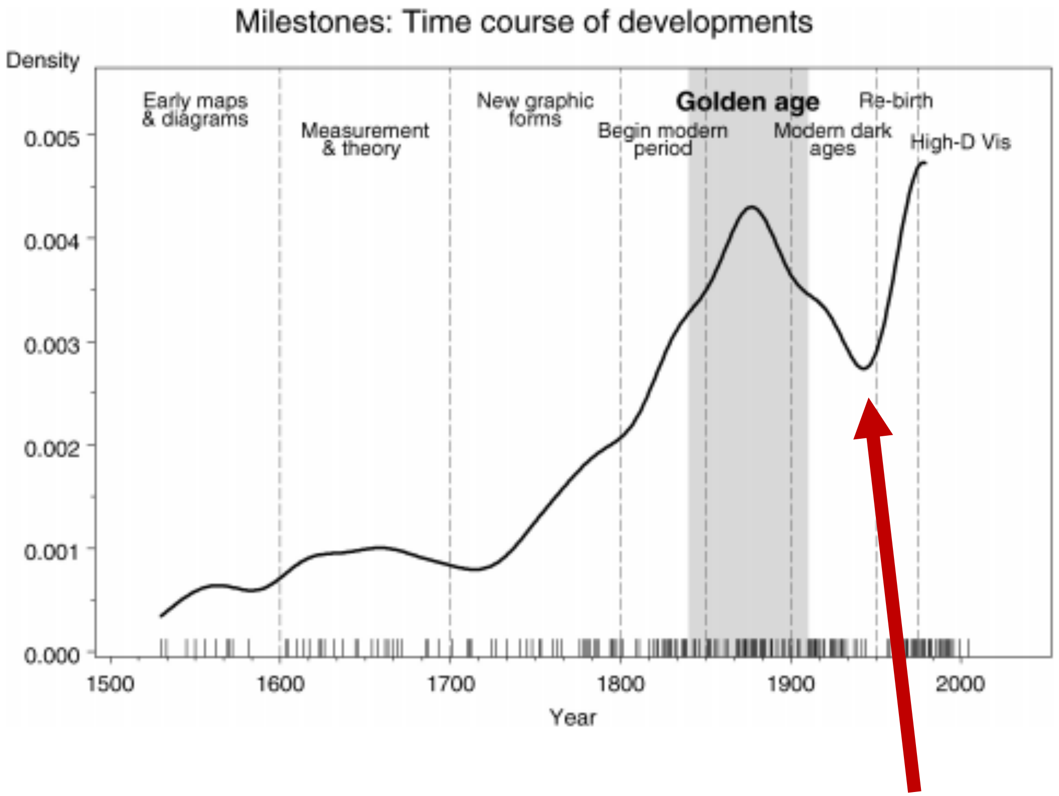


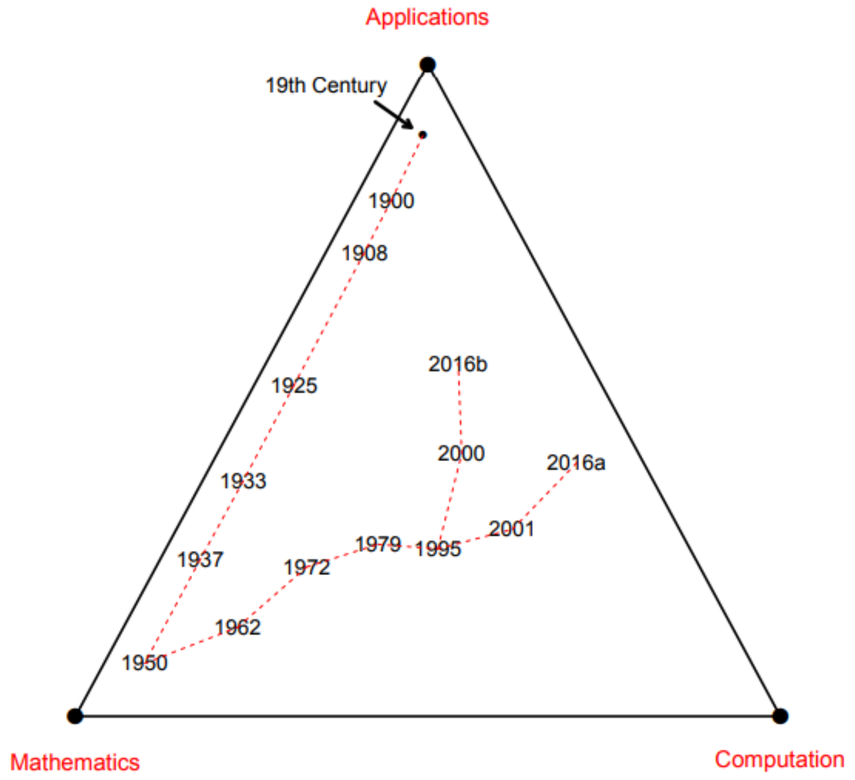
The dotted lines indicate the gradations of barometric pressure The variations of the temperature are marked by figures, the state of the sea and sky by descriptive words, and the direction of the wind by arrows—barbed and feathered according to its force. ⊙ denotes calm.

First weather map published in a newspaper (1875)

A very brief history of data visualization

“Graphical dark ages” around 1950

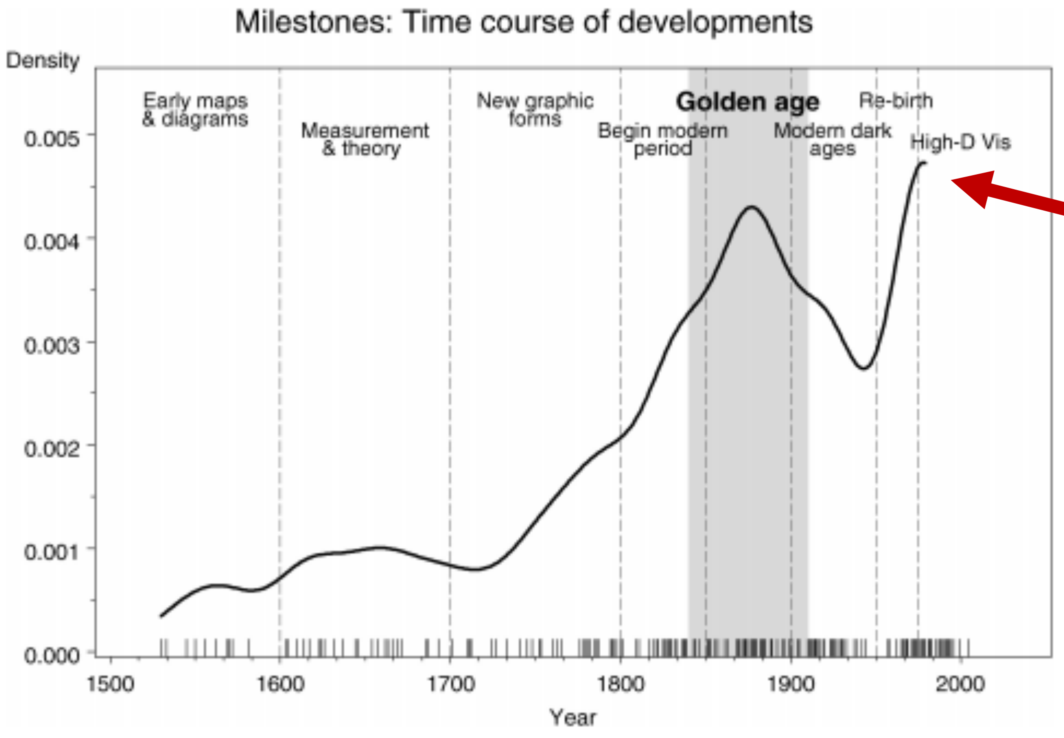


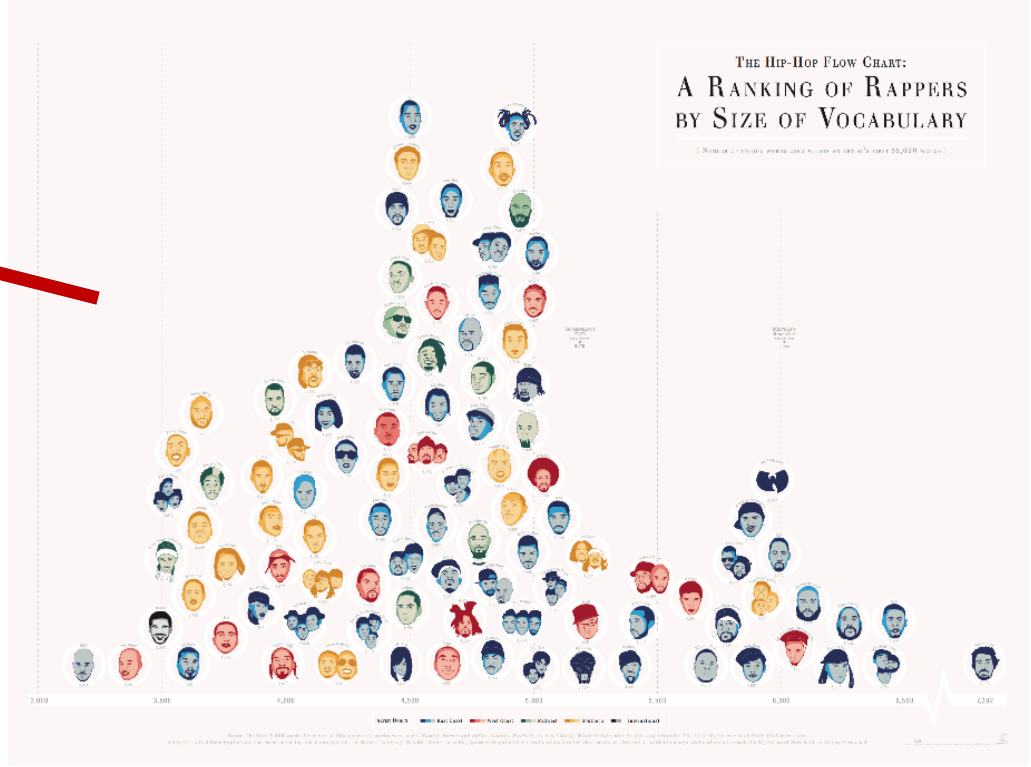


Computer Age Statistical Inference, Efron and Hastie

A very brief history of data visualization

Currently undergoing a “Graphical re-birth”





Survey question 1

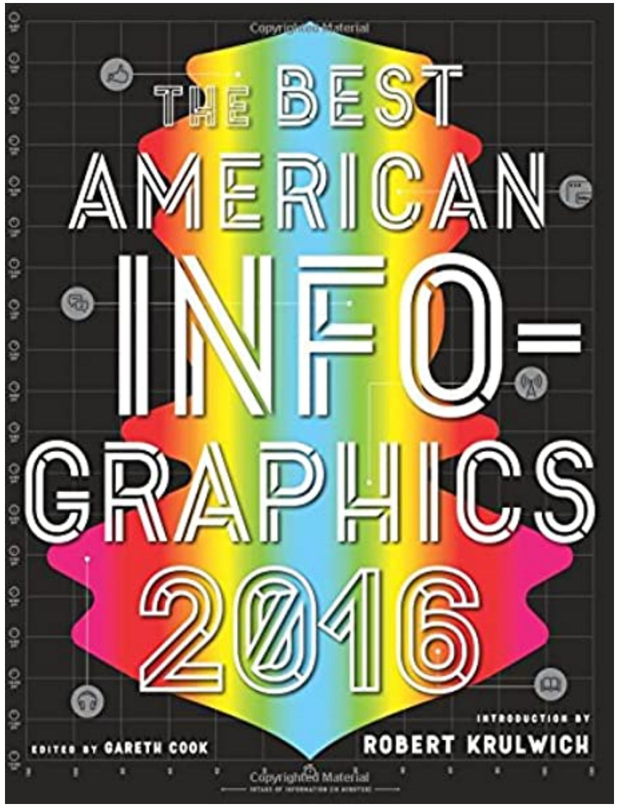
Find an interesting data visualization on the web:

1. Write down the URL link to the image
2. Explain why you think it is interesting

Brief class share on Thursday

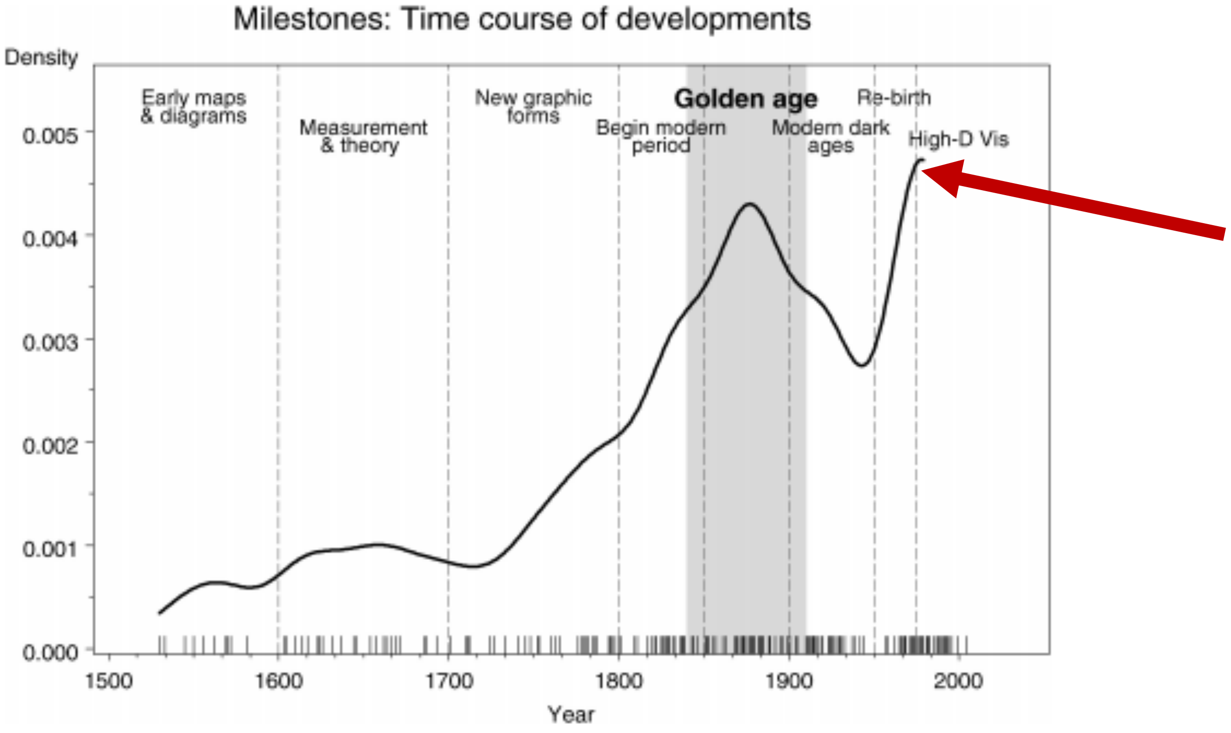
https://www.reddit.com/r/dataisbeautiful/

https://flowingdata.com/



A very brief history of data visualization

Currently undergoing a “Graphical re-birth”



Hans Rosling’s gapminder

- Simple version
- TV special effects
- Ted Talk

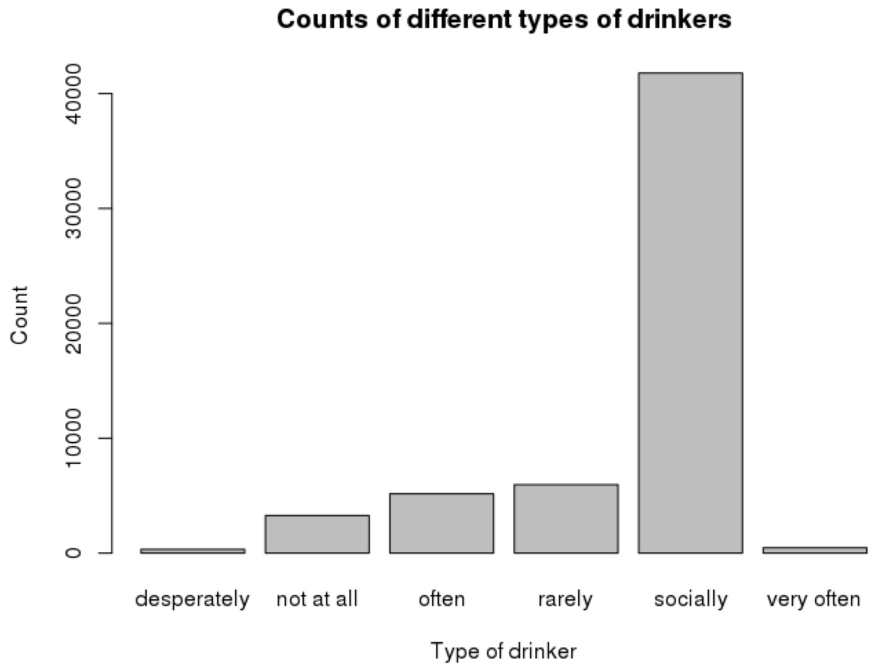
Gapminder tools:
https://www.gapminder.org/tools

```
> library('gapminder')
```

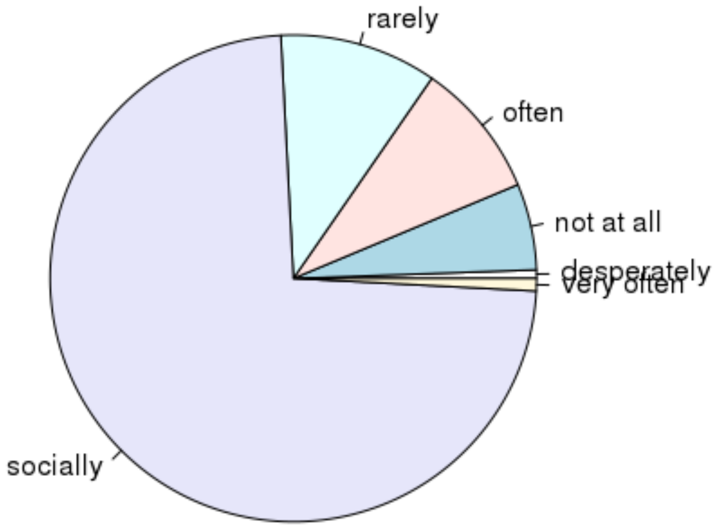
A grammar of graphics and ggplot

Review: plots of categorical data

Bar plot

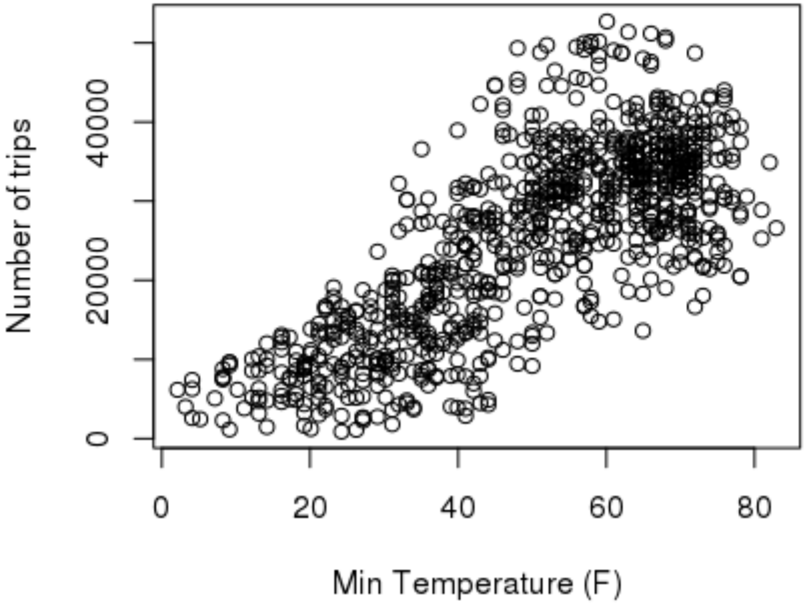


Pie chart

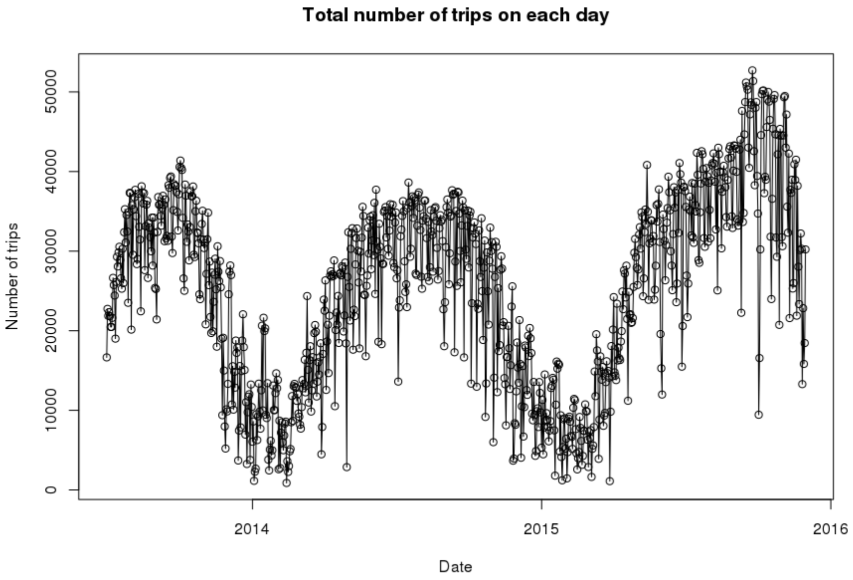


Review: plots of quantitative data

Scatter plots

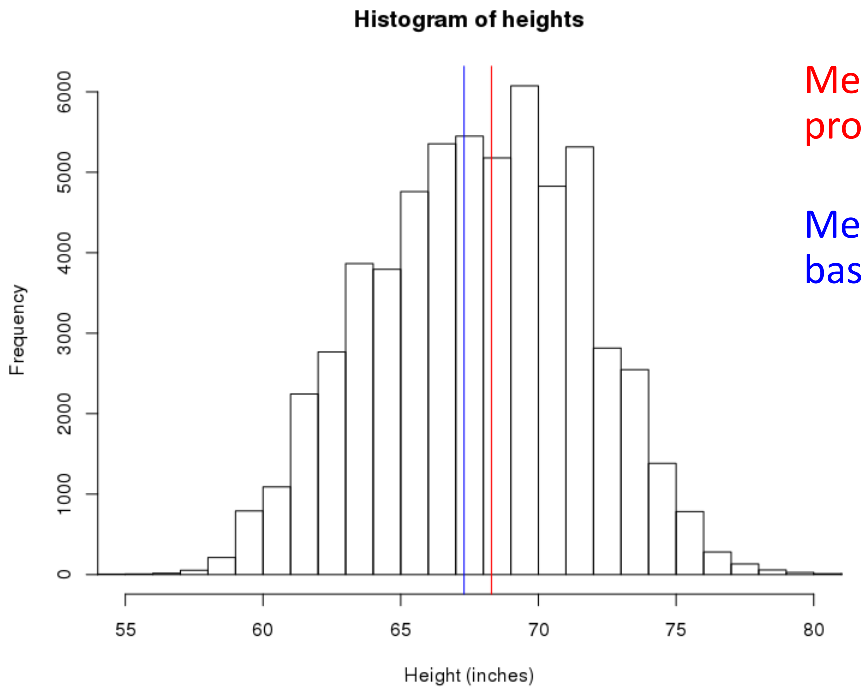


Line chart



Review: plots of quantitative data

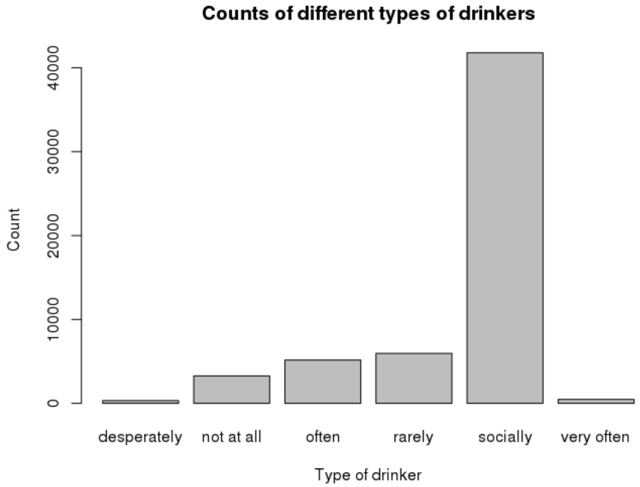
Histograms

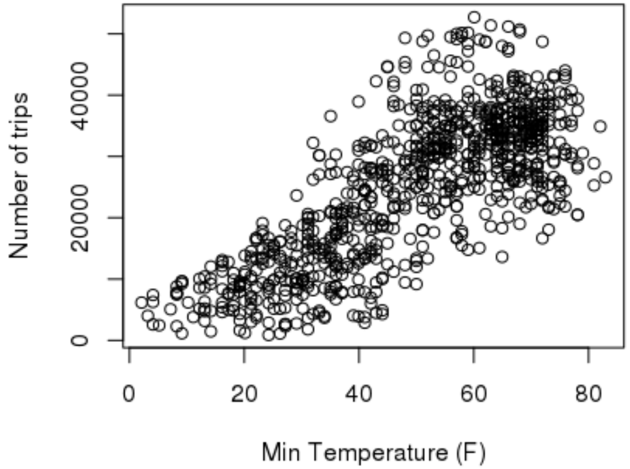


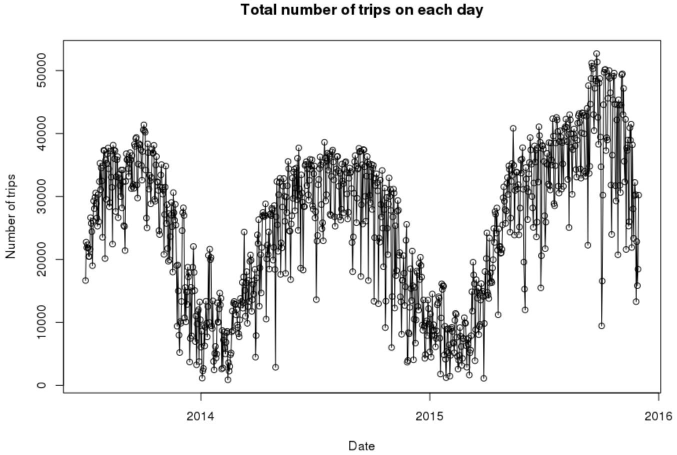
Mean height in the profiles data

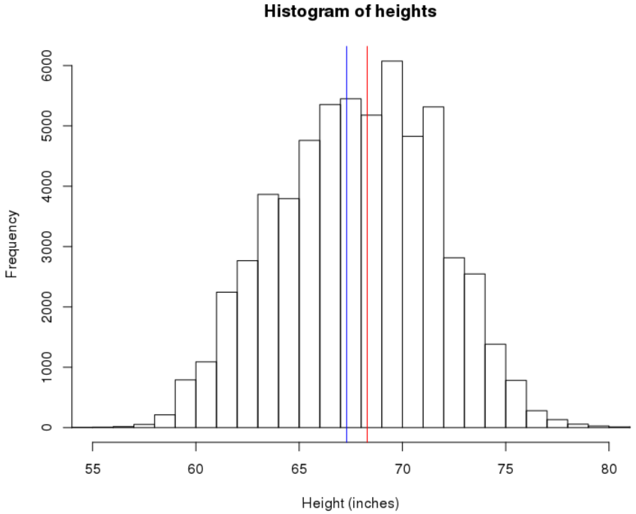
Mean height expected based on US population

Survey question 2: What are some similarities between these graphs?







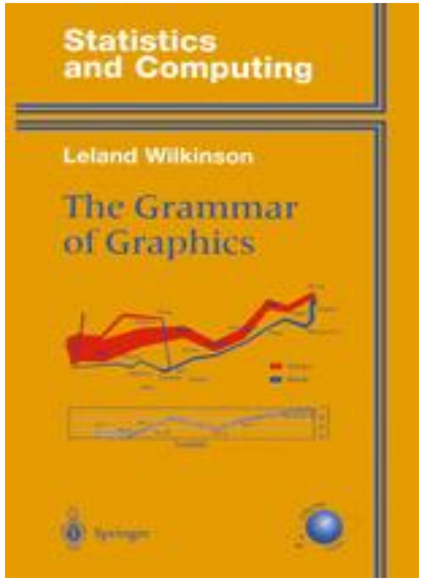


The grammar of graphics

Leland Wilkinson noticed similarities between many graphs and tried to generate a ‘grammar’ that could be used to express a graph

- i.e., a list elements that can be combined together to create a graph

First edition



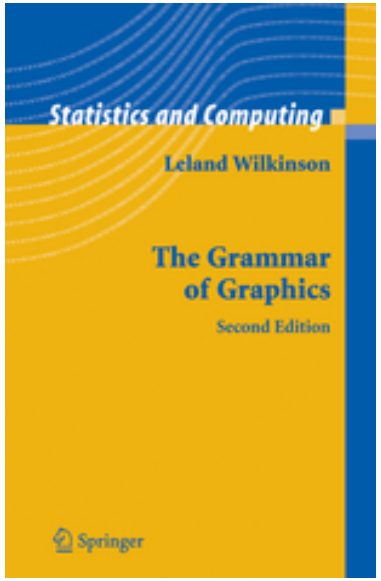
Statistics and Computing

Leland Wilkinson

The Grammar of Graphics

Springer

Second edition



Statistics and Computing

Leland Wilkinson

The Grammar of Graphics

Second Edition

Springer

Graphs are composed of…

A Frame: Coordinate system on which data is placed

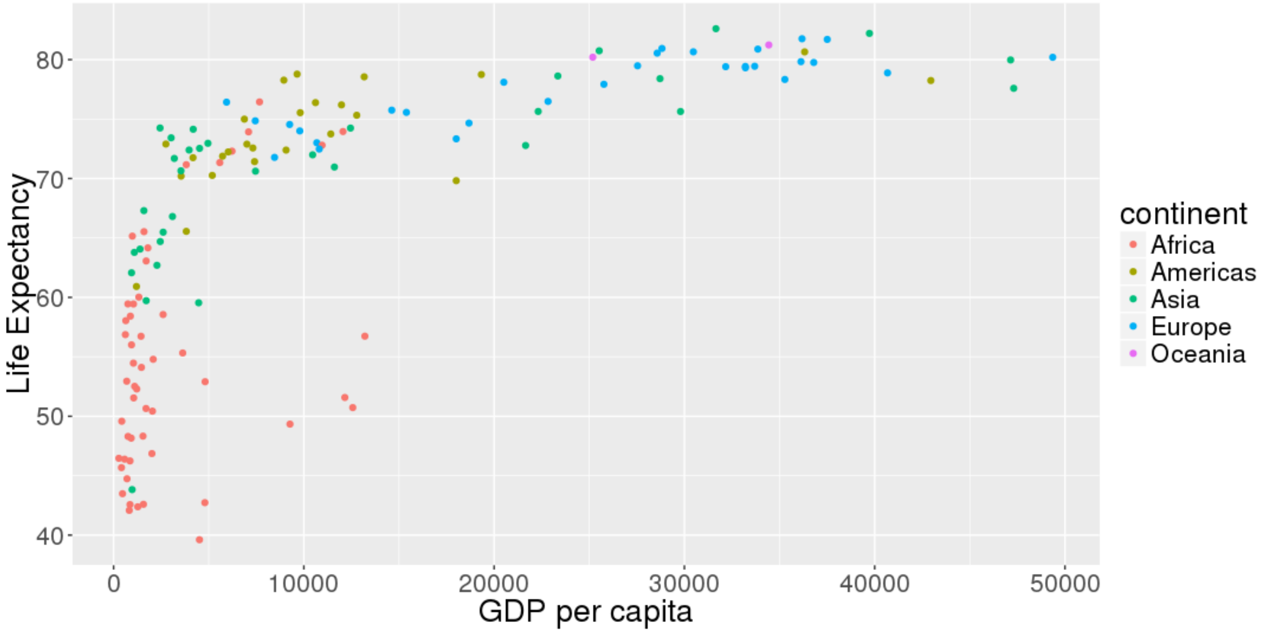
- E.g., Cartesian coordinate system, polar coordinates, etc.

Glyphs: basic graphic unit representing cases or statistics

- Contains visual properties (aesthetics) such as: shape, color, size, etc.
- Need to specify how properties of the data are **mapped** onto these aesthetics

Scales and guides: shows how to interpret axes and other properties of the glyphs

- i.e., gives information about how the data values were mapped into glyph properties



Plots can also contain…

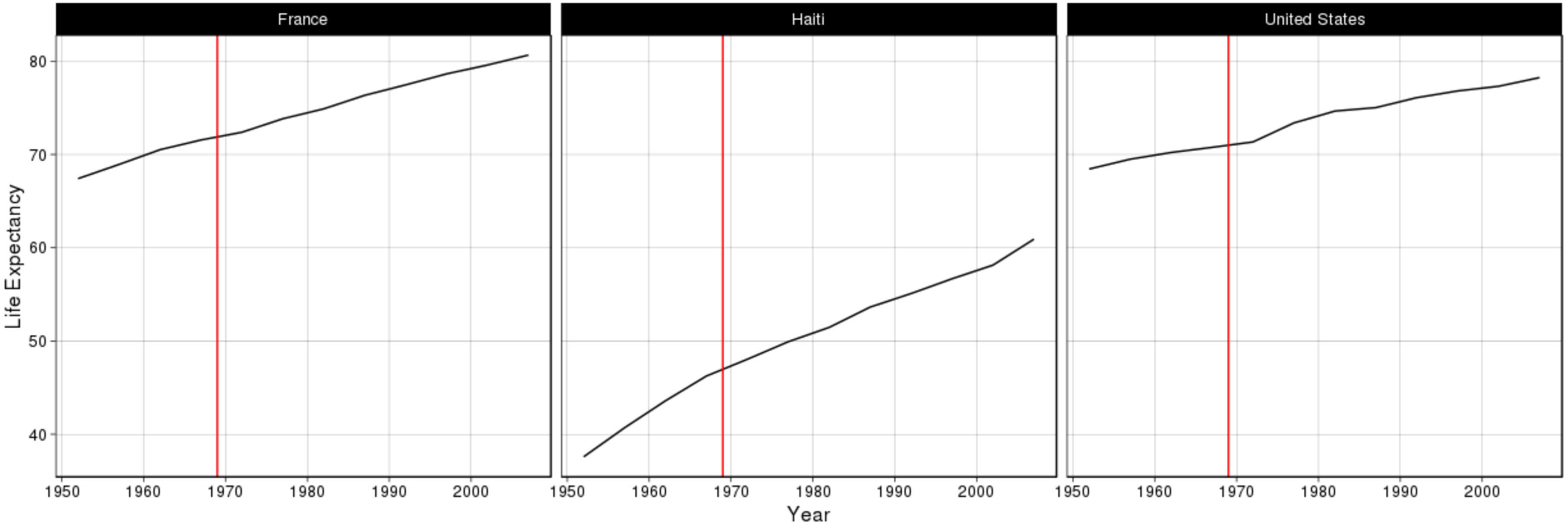
Facets: allows for multiple side-by-side graphs based on a categorical variable

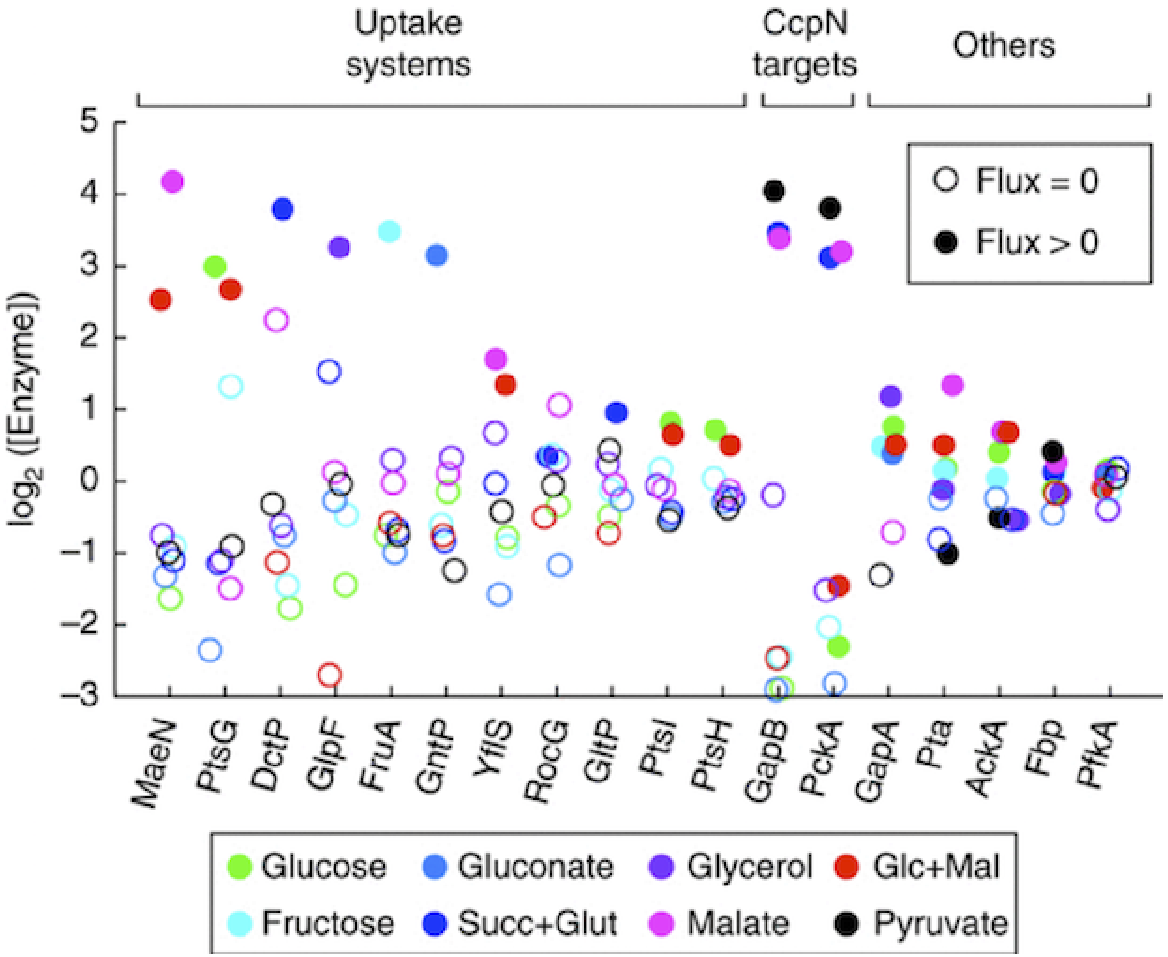
- Makes it easier to compare different conditions

Layers: allows for more than one types of data to be mapped onto the same figure

Theme: contains finer points of display

- E.g., font size, background color, etc.





The variables are:

- Log enzyme concentration
  - -3 to 5
- Target
  - CcpN, Uptake,…
- Flux
  - Zero or positive
- Gene
  - MaeN, PtsG, …
- Molecule:
  - Glocose, Fructose, …

Survey question 3: What all the mapping between variables and visual attributes?

- i.e., see if you can list the mappings from all variables to visual attributes.

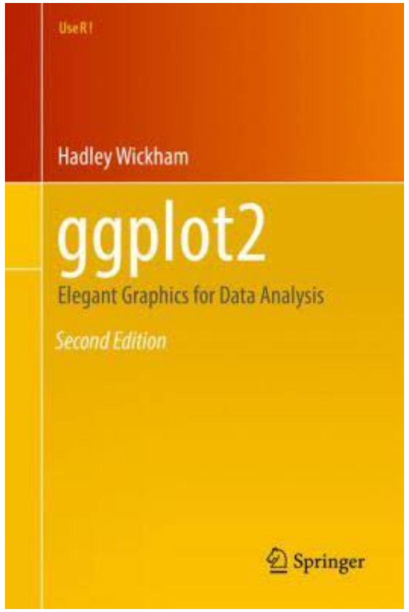
Also see if you can sketch out the data frame that underlies this figure on a piece of paper

ggplot

ggplot2 is an R package that implements the grammar of graphics

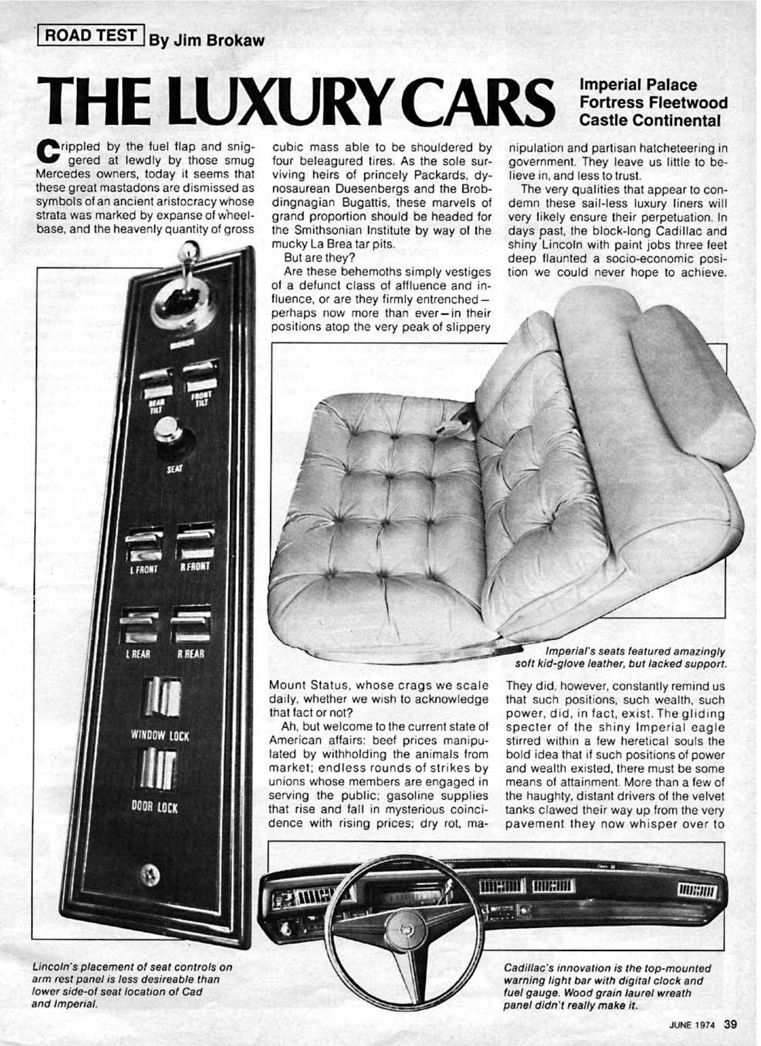
- It builds up graphics by starting with a frame, adding glyphs, etc.

```
# load the ggplot2 library
> library('ggplot2')
```



Get the book on GitHub

Example data: mtcars


REAR TILT
FRONT TILT
SEAT
L FRONT
R FRONT
L REAR
R REAR
WINDOW LOCK
DOOR LOCK

ROAD TEST By Jim Brokaw

THE LUXURY CARS

Imperial Palace
Fortress Fleetwood
Castle Continental

Crippled by the fuel flap and sniggered at lewdly by those smug Mercedes owners, today it seems that these great mastadons are dismissed as symbols of an ancient aristocracy whose strata was marked by expanse of wheelbase, and the heavenly quantity of gross cubic mass able to be shouldered by four beleagured tires. As the sole surviving heirs of princely Packards, dynosaurean Duesenbergs and the Brobdingnagian Bugattis, these marvels of grand proportion should be headed for the Smithsonian Institute by way of the mucky La Brea tar pits.

But are they?

Are these behemoths simply vestiges of a defunct class of affluence and influence, or are they firmly entrenched—perhaps now more than ever—in their positions atop the very peak of slippery Mount Status, whose crags we scale daily, whether we wish to acknowledge that fact or not?

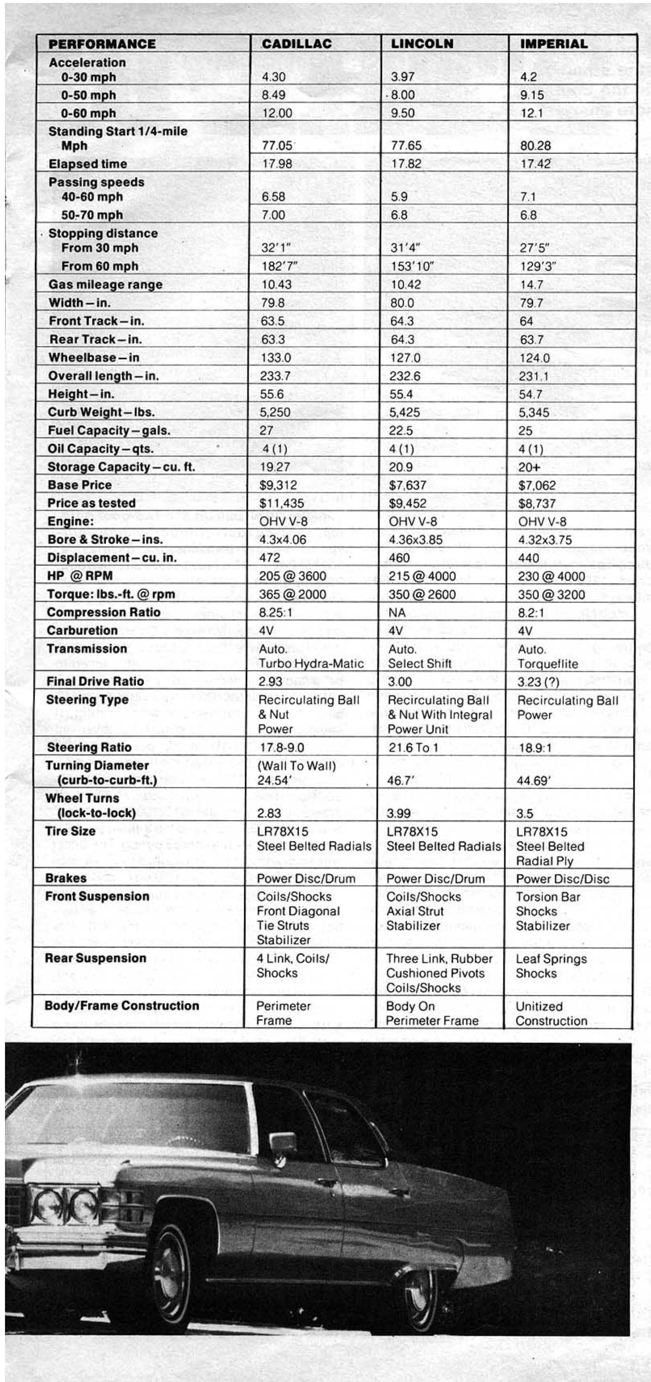
Ah, but welcome to the current state of American affairs: beef prices manipulated by withholding the animals from market; endless rounds of strikes by unions whose members are engaged in serving the public; gasoline supplies that rise and fall in mysterious coincidence with rising prices; dry rot, manipulation and partisan hatcheteering in government. They leave us little to believe in, and less to trust.

The very qualities that appear to condemn these sail-less luxury liners will very likely ensure their perpetuation. In days past, the block-long Cadillac and shiny Lincoln with paint jobs three feet deep flaunted a socio-economic position we could never hope to achieve. They did, however, constantly remind us that such positions, such wealth, such power, did, in fact, exist. The gliding specter of the shiny Imperial eagle stirred within a few heretical souls the bold idea that if such positions of power and wealth existed, there must be some means of attainment. More than a few of the haughty, distant drivers of the velvet tanks clawed their way up from the very pavement they now whisper over to

Imperial's seats featured amazingly soft kid-glove leather, but lacked support.

Lincoln's placement of seat controls on arm rest panel is less desireable than lower side-of seat location of Cad and Imperial.

Cadillac's innovation is the top-mounted warning light bar with digital clock and fuel gauge. Wood grain laurel wreath panel didn't really make it.



PERFORMANCE	CADILLAC	LINCOLN	IMPERIAL
Acceleration 0-30 mph	4.30	3.97	4.2
0-50 mph	8.49	8.00	9.15
0-60 mph	12.00	9.50	12.1
Standing Start 1/4-mile Mph	77.05	77.65	80.28
Elapsed time	17.98	17.82	17.42
Passing speeds 40-60 mph	6.58	5.9	7.1
50-70 mph	7.00	6.8	6.8
Stopping distance From 30 mph	32′1″	31′4″	27′5″
From 60 mph	182′7″	153′10″	129′3″
Gas mileage range	10.43	10.42	14.7
Width—in.	79.8	80.0	79.7
Front Track—in.	63.5	64.3	64
Rear Track—in.	63.3	64.3	63.7
Wheelbase—in	133.0	127.0	124.0
Overall length—in.	233.7	232.6	231.1
Height—in.	55.6	55.4	54.7
Curb Weight—lbs.	5,250	5,425	5,345
Fuel Capacity—gals.	27	22.5	25
Oil Capacity—qts.	4 (1)	4 (1)	4 (1)
Storage Capacity—cu. ft.	19.27	20.9	20+
Base Price	$9,312	$7,637	$7,062
Price as tested	$11,435	$9,452	$8,737
Engine:	OHV V-8	OHV V-8	OHV V-8
Bore & Stroke—ins.	4.3x4.06	4.36x3.85	4.32x3.75
Displacement—cu. in.	472	460	440
HP @ RPM	205 @ 3600	215 @ 4000	230 @ 4000
Torque: lbs.-ft. @ rpm	365 @ 2000	350 @ 2600	350 @ 3200
Compression Ratio	8.25:1	NA	8.2:1
Carburetion	4V	4V	4V
Transmission	Auto. Turbo Hydra-Matic	Auto. Select Shift	Auto. Torqueflite
Final Drive Ratio	2.93	3.00	3.23 (?)
Steering Type	Recirculating Ball & Nut Power	Recirculating Ball & Nut With Integral Power Unit	Recirculating Ball Power
Steering Ratio	17.8-9.0	21.6 To 1	18.9:1
Turning Diameter (curb-to-curb-ft.)	(Wall To Wall) 24.54′	46.7′	44.69′
Wheel Turns (lock-to-lock)	2.83	3.99	3.5
Tire Size	LR78X15 Steel Belted Radials	LR78X15 Steel Belted Radials	LR78X15 Steel Belted Radial Ply
Brakes	Power Disc/Drum	Power Disc/Drum	Power Disc/Disc
Front Suspension	Coils/Shocks Front Diagonal Tie Struts Stabilizer	Coils/Shocks Axial Strut Stabilizer	Torsion Bar Shocks Stabilizer
Rear Suspension	4 Link, Coils/ Shocks	Three Link, Rubber Cushioned Pivots Coils/Shocks	Leaf Springs Shocks
Body/Frame Construction	Perimeter Frame	Body On Perimeter Frame	Unitized Construction

mtcars data frame

How can you determine what variables are in a data frame?

```
> View(mtcars)      # only works in Rstudio, not in Markdown
> glimpse(mtcars)
> ? mtcars          # this data frame as a code book
```

```
[, 1]     mpg    Miles/(US) gallon
[, 2]     cyl    Number of cylinders
[, 4]     hp     Gross horsepower
[, 6]     wt     Weight (1000 lbs)
[, 9]     am     Transmission (0 = automatic, 1 = manual)
```

Do cars that weigh more use more fuel?

Question: do cars that weigh more use more fuel?

What variables in the mtcars data frame are of interest?

- mpg
- wt

We can create a scatter plot using base graphics…

```
> plot(mtcars$wt, mtcars$mpg)
```

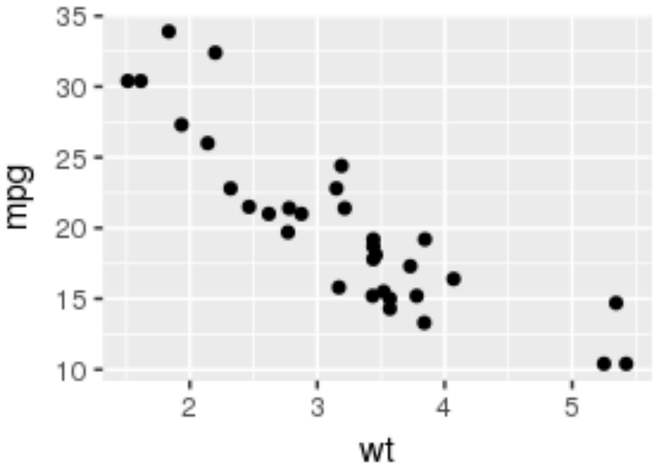
Creating a scatter plot in ggplot

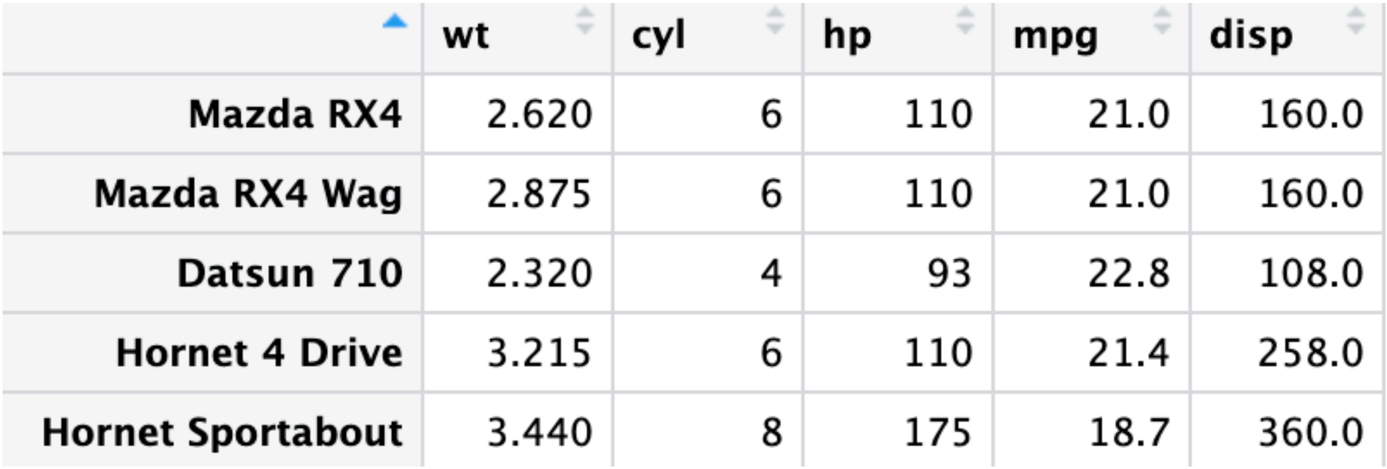
Data frame to be used

Aesthetic mapping

```
> ggplot(data = mtcars, mapping = aes(x = wt, y = mpg)) +
      geom_point()
```

Adds a layer with glyphs





	wt	cyl	hp	mpg	disp
Mazda RX4	2.620	6	110	21.0	160.0
Mazda RX4 Wag	2.875	6	110	21.0	160.0
Datsun 710	2.320	4	93	22.8	108.0
Hornet 4 Drive	3.215	6	110	21.4	258.0
Hornet Sportabout	3.440	8	175	18.7	360.0

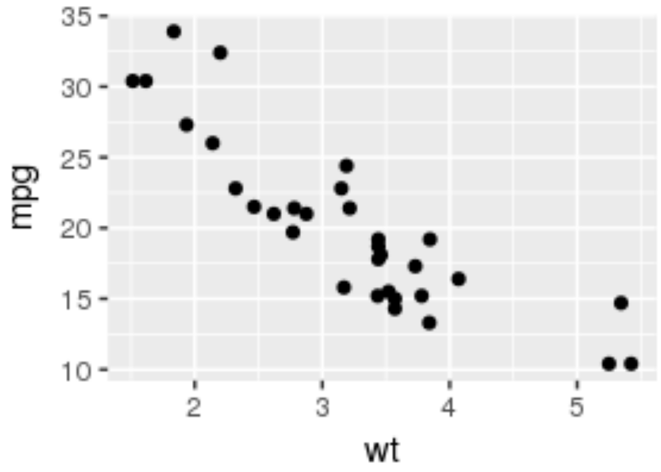
Creating a scatter plot in ggplot

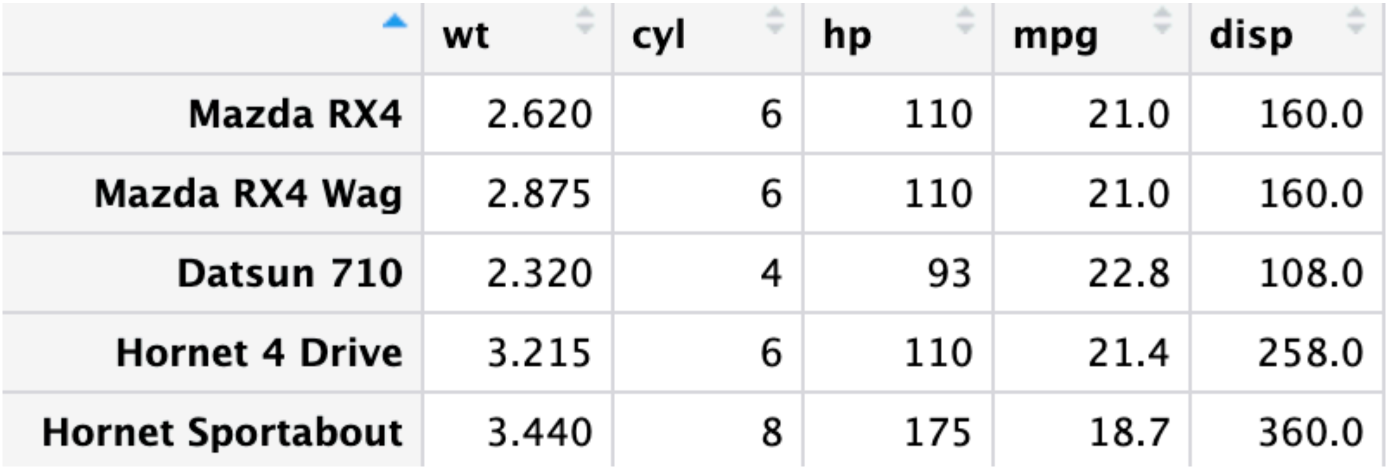
Data frame to be used

Aesthetic mapping

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +
        geom_point()
```

Adds a layer with glyphs

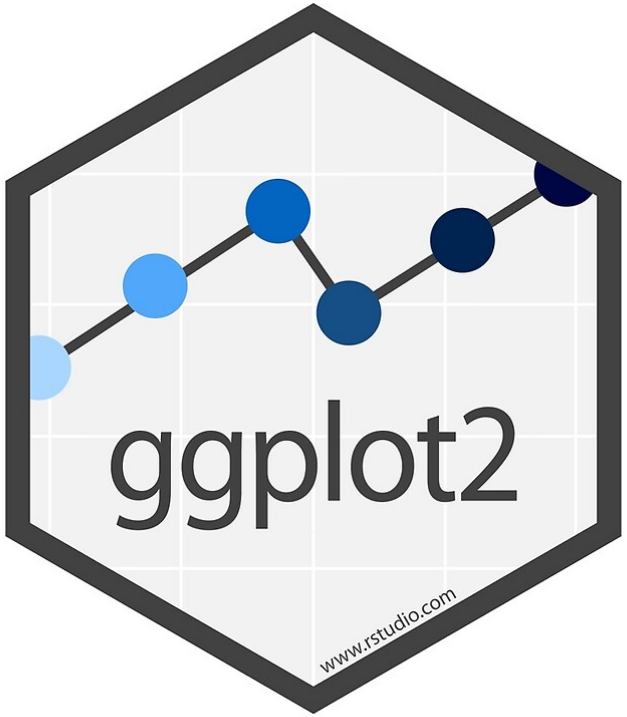




	wt	cyl	hp	mpg	disp
Mazda RX4	2.620	6	110	21.0	160.0
Mazda RX4 Wag	2.875	6	110	21.0	160.0
Datsun 710	2.320	4	93	22.8	108.0
Hornet 4 Drive	3.215	6	110	21.4	258.0
Hornet Sportabout	3.440	8	175	18.7	360.0

A lot more that ggplot can do!

- More aesthetic mapping
- Multiple glyphs/layers
- Axis labels
- Facets
- Visual themes
- Different coordinate systems
- Etc.



The R Graph Gallery

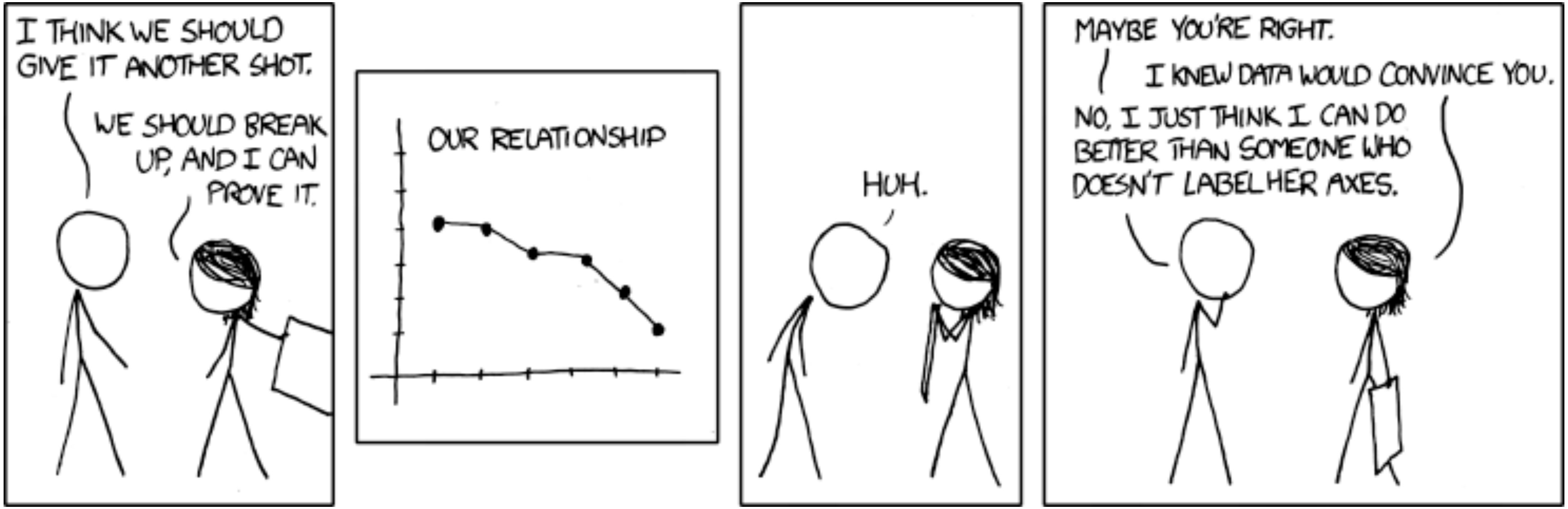
Let’s try the rest in R!

Adding labels to plots

We can add labels to the plots using the xlab("label1") and ylab("label2") functions

Add labels to your last plot

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +
        geom_point()  +
        xlab("Weight") +
        ylab("Miles per Gallon")
```


I THINK WE SHOULD GIVE IT ANOTHER SHOT.
WE SHOULD BREAK UP, AND I CAN PROVE IT.
OUR RELATIONSHIP
HUH.
MAYBE YOU'RE RIGHT.
I KNEW DATA WOULD CONVINCE YOU.
NO, I JUST THINK I CAN DO BETTER THAN SOMEONE WHO DOESN'T LABEL HER AXES.

If you don’t want an ex, label you axes!

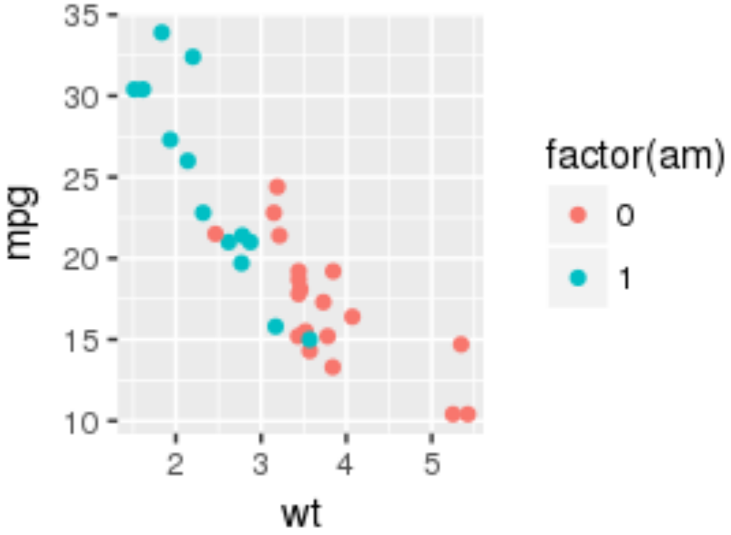
More aesthetic mappings

Let's look at the relationship between weight, miles per gallon and transmission type on the same graph by plotting… (?)

```
> ggplot(mtcars, aes(x = wt, y = mpg, col = am)) +
            geom_point()
```

```
# It is better if we make am a categorical variable
> ggplot(mtcars, aes(x = wt, y = mpg, col = factor(am))) +  geom_point()
```

Notice the guides!!!



Try mapping am on to shape using:

1. *shape = am*
2. size using: *size = am*

Which is better to use color or shape or size?

Attributes vs. Aesthetics

Setting **aesthetics** map a variable to a glyph property

Setting **attributes** set a glyph property to a fixed value

```
# setting a aesthetic
> ggplot(mtcars) +
        geom_point(aes(x = wt, y = mpg, col = factor(am)))
```

```
# setting an attribute
> ggplot(mtcars) +
        geom_point(aes(x = wt, y = mpg), col = "red")
```

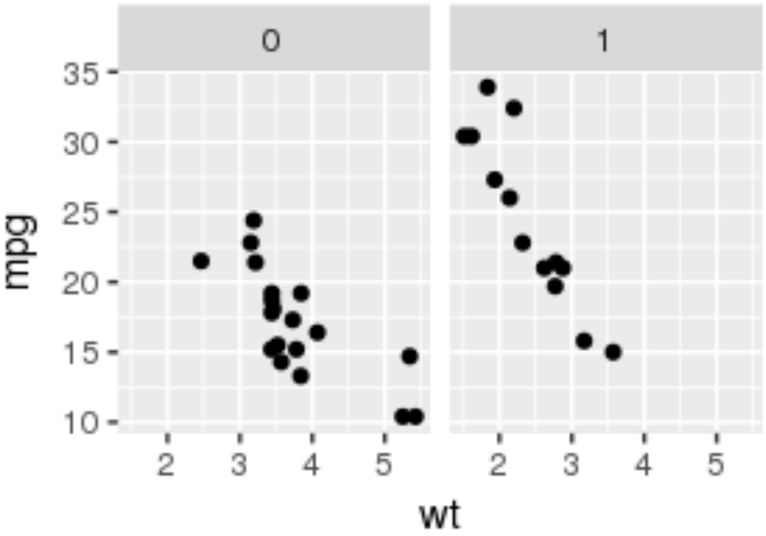
Outside the aesthetic mapping!

Facets

Beyond comparing variables based on aesthetics you can compare categorical variables by splitting a plot into subplots (called facets) using facet_wrap

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +   geom_point() +
        facet_wrap(~am)
```

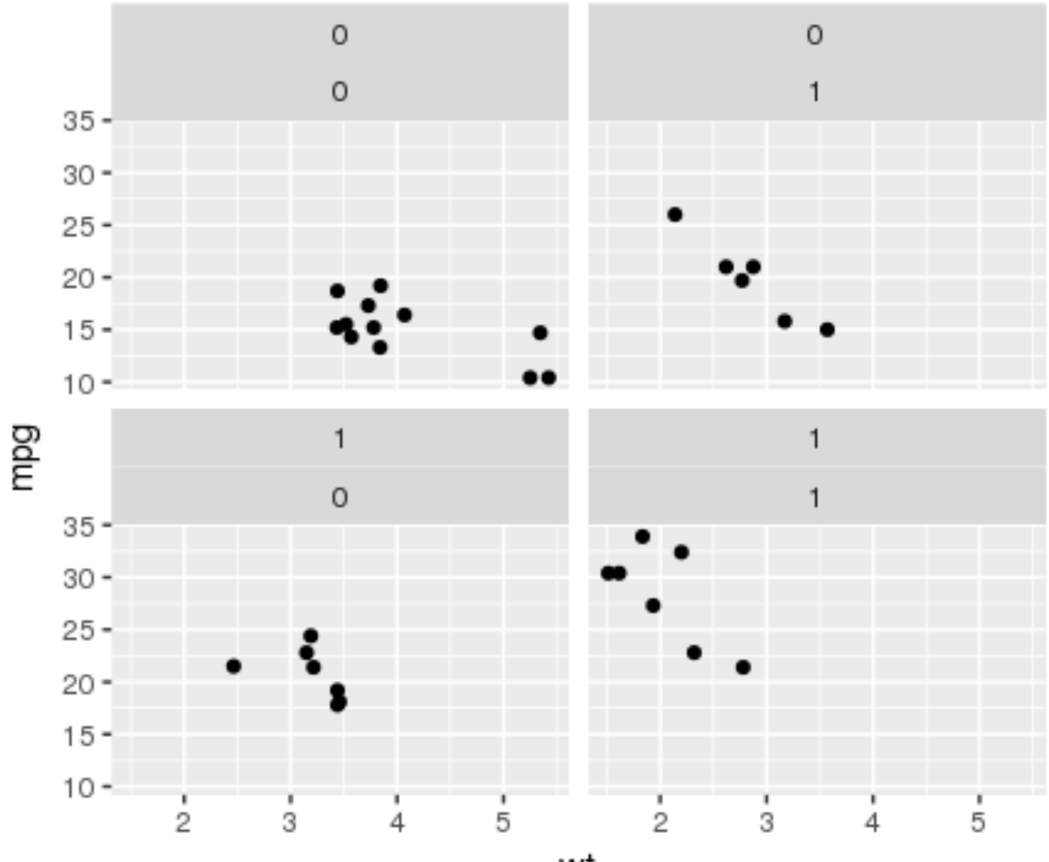
What do facets make it easy to see on this graph?



Facets along two dimensions

One can also do facets in two dimensions

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +
        geom_point() +
        facet_wrap(vs ~am)
```



Overplotting

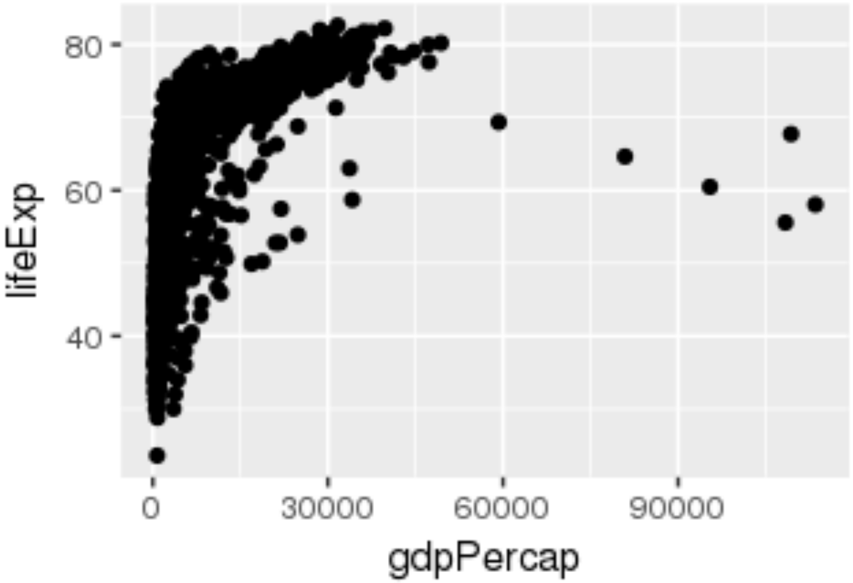
Sometimes points overlap making it hard to estimate the number of points at a particular range of values

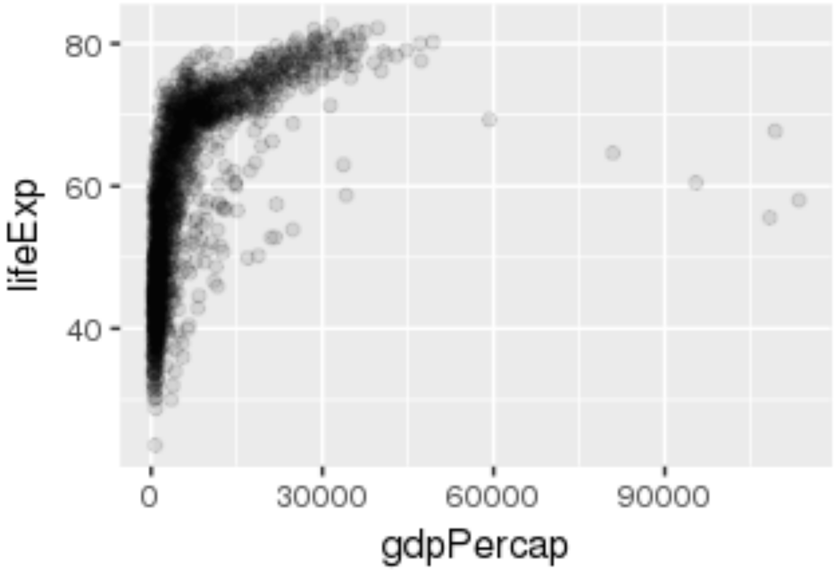
We can control the transparency of points by changing their alpha values

```
# compare these two plots
> ggplot(gapminder, aes(x = gdpPercap, y = lifeExp)) +
        geom_point()

> ggplot(gapminder, aes(x = gdpPercap, y = lifeExp)) +
        geom_point(alpha = .1)
```

Overplotting





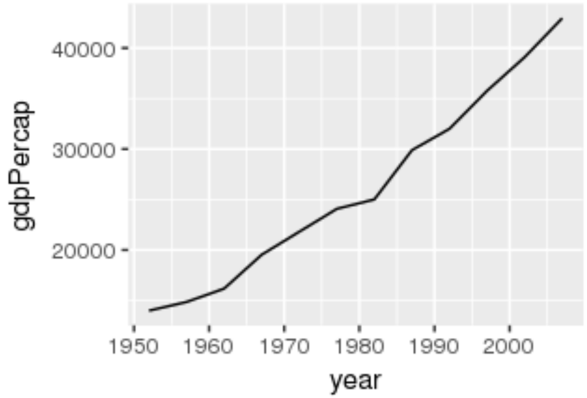
Geometries: line plot

So far we’ve only created scatter plots, but we can use different geoms to create other types of plots

Create a plot that shows the GDP in the United States as a function of the year using the geom *geom_line()*

- Hint: filter the gapminder data first…

```
>  gapminder %>% filter(country == 'United States') %>%
        ggplot(aes(x = year, y = gdpPercap)) +
        geom_line()
```



Geometries: histograms

We can also make histograms using the geom_histogram() function.

Plot a histogram of the weights of cars

```
> ggplot(mtcars, aes(x = wt)) +
        geom_histogram()
```

Note the histogram geom only has an x aesthetic, and does not have a y aesthetic value.

Geometries: boxplot

There are many other geom as well, including geom_boxplot()

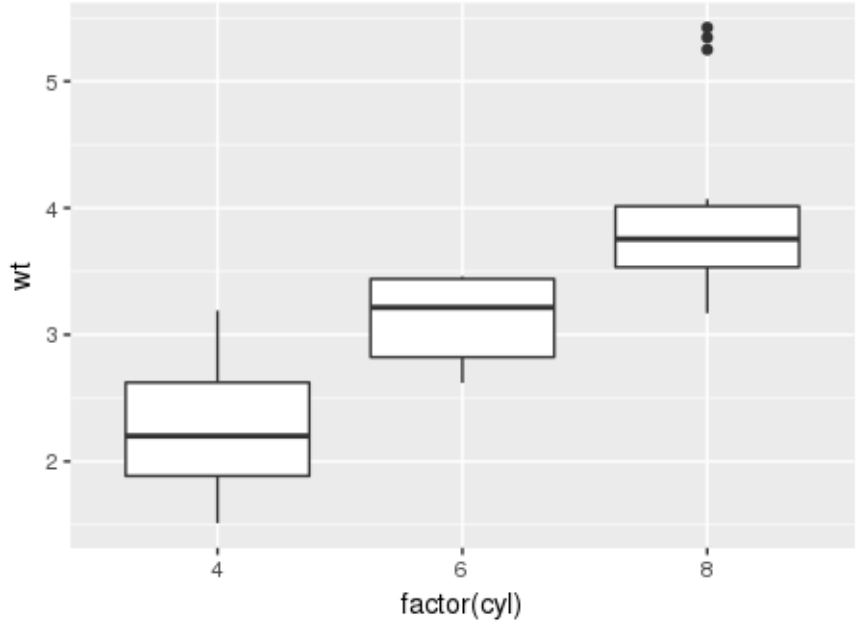
Plot a boxplot of the weights of cars

```
> ggplot(mtcars, aes(x = "", y = wt)) +
        geom_boxplot()
```

Side-by-side boxplots

Often it is useful to compare boxplots across different groups

```
> ggplot(mtcars, aes(x = factor(cyl), y = wt)) +
        geom_boxplot()
```



Violin and Joy plots

Violin and Joy plots are other ways to view distributions of data

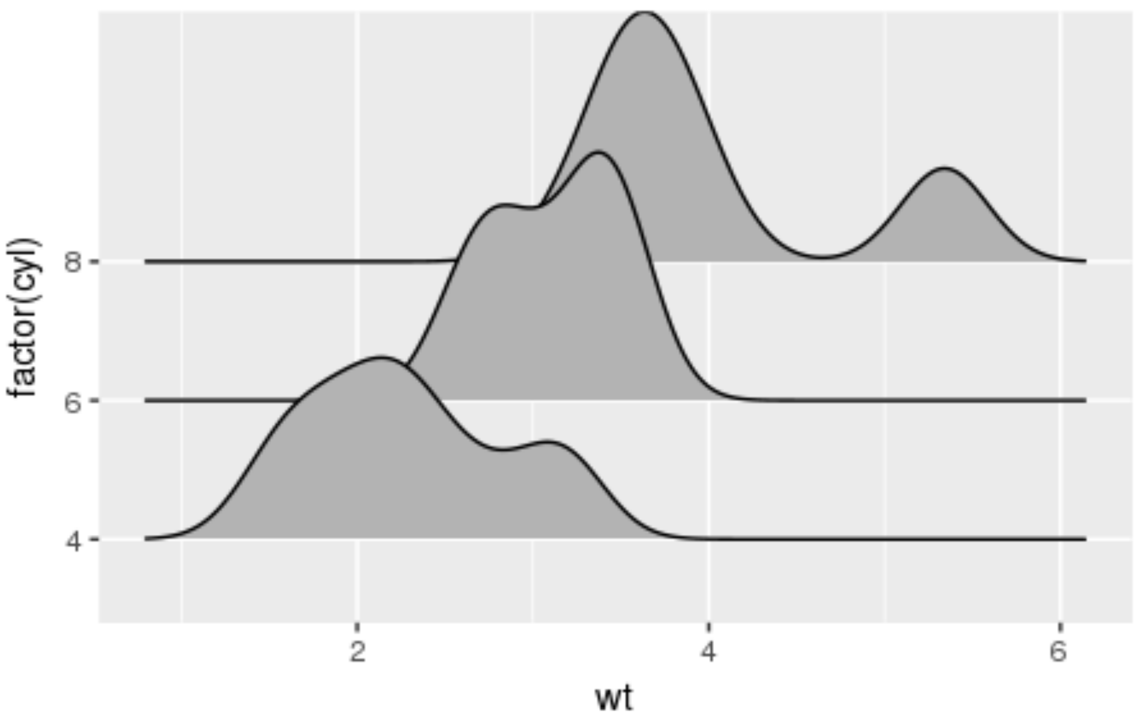
```
> ggplot(mtcars, aes(x = factor(cyl), y = wt)) +
        geom_violin()
```

Note x and y are reversed

```
> library("ggridges")
> ggplot(mtcars, aes(y = factor(cyl), x = wt)) +
        geom_density_ridges()
```

Violin and Joy plots

Any ideas why they are called joy plots?



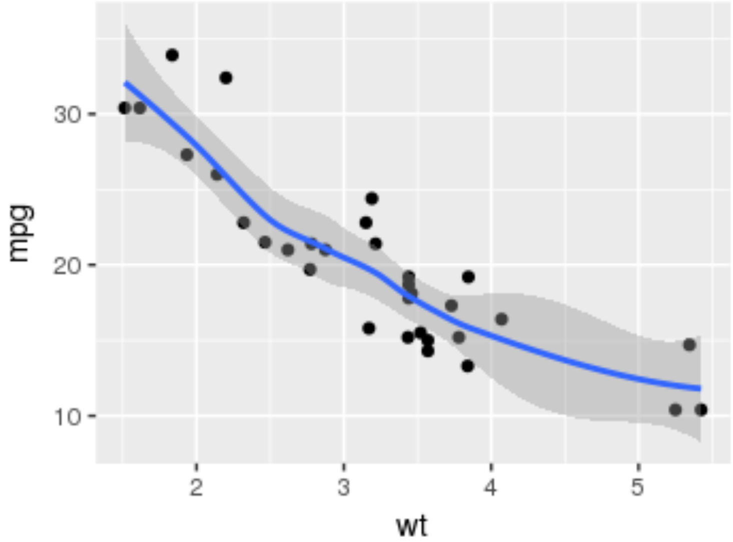
Multiple layers

We can also have multiple geom layers on a single graph by using the + symbol

- E.g ggplot(…) + geom_type1() + geom_type2()

Create a scatter plot of miles per gallon as a function of weight and then add a smoothed line using geom_smooth()

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +
        geom_point() +
        geom_smooth()
```



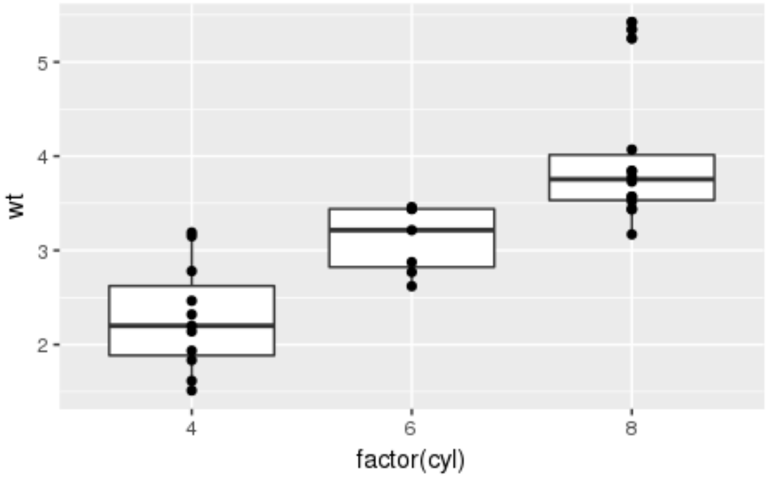
Multiple layers

We can also have multiple geom layers on a single graph by using the + symbol

- E.g ggplot(…) + geom_type1() + geom_type2()

Recreate a boxplot of weight (wt) grouped by the factor of cylinders (cyl), and then add points using geom_point()

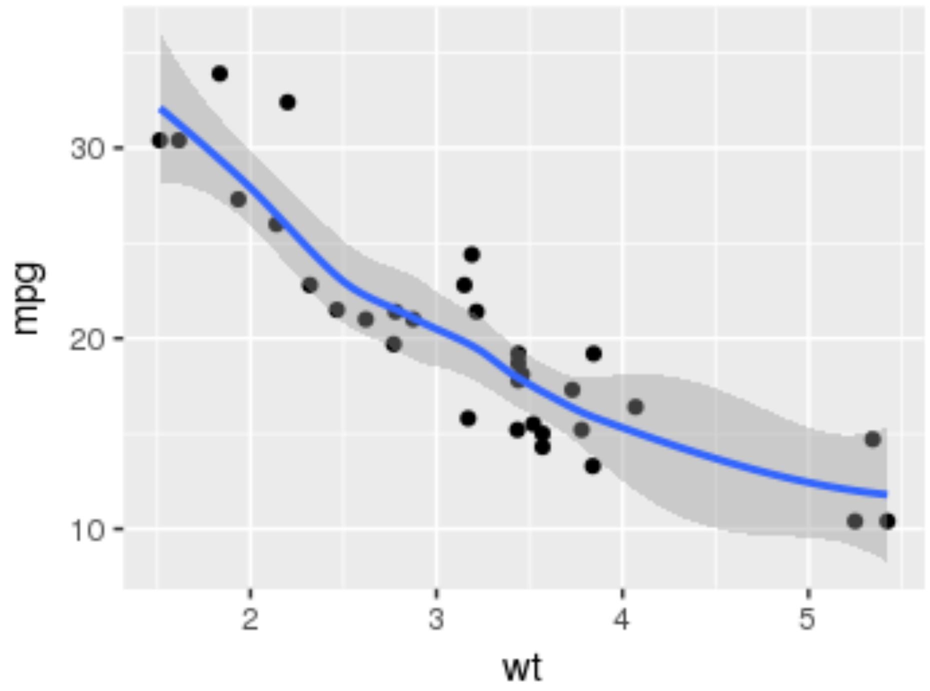
```
> ggplot(mtcars, aes(x = factor(cyl), y = wt)) +
        geom_boxplot() +
        geom_point()
```



Multiple layers

Create a scatter plot of miles per gallon (mpg) as a function of weight (wt) and then add a smoothed line using geom_smooth()

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +
        geom_point() +
        geom_smooth()
```



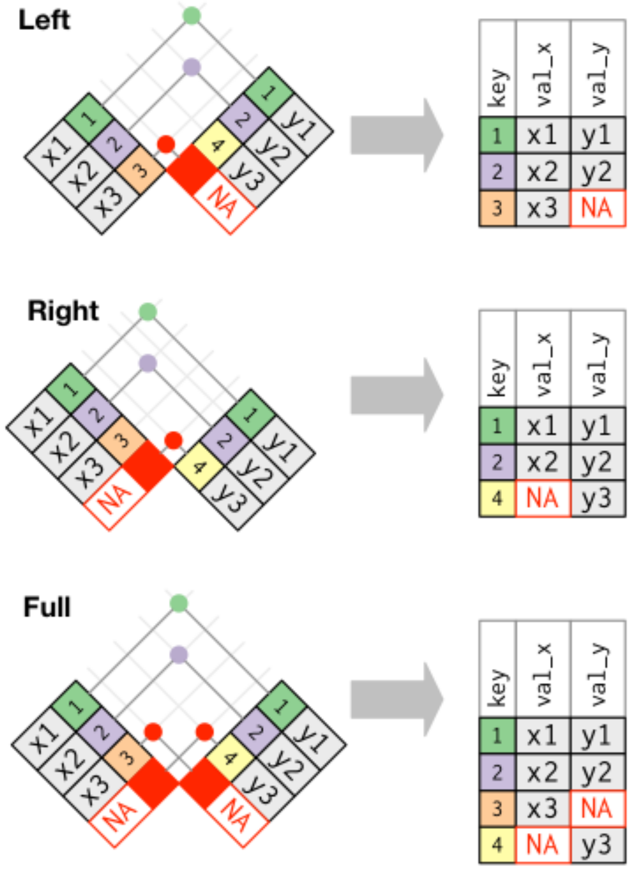
Themes

We can also use different types to change the appearance of our plot

Add theme_classic() to your plot

```
> ggplot(mtcars, aes(x = wt, y = mpg)) +
        geom_point()  +
        xlab("Weigth") +
        ylab("Miles per Gallon") +
        theme_classic()
```

Joining data frames



key	val_x	val_y
1	x1	y1
2	x2	y2
3	x3	NA

key	val_x	val_y
1	x1	y1
2	x2	y2
4	NA	y3

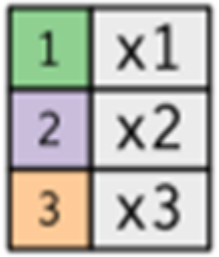
key	val_x	val_y
1	x1	y1
2	x2	y2
3	x3	NA
4	NA	y3

Left and right tables

Suppose we have two data frames called x and y

- x have two variables called key_x, and val_x
- y has two variables called key_y and val_y

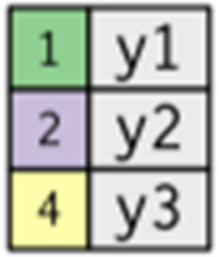
x$key_x x$val_x



1	x1
2	x2
3	x3

Data frame x

y$key_y y$val_y



1	y1
2	y2
4	y3

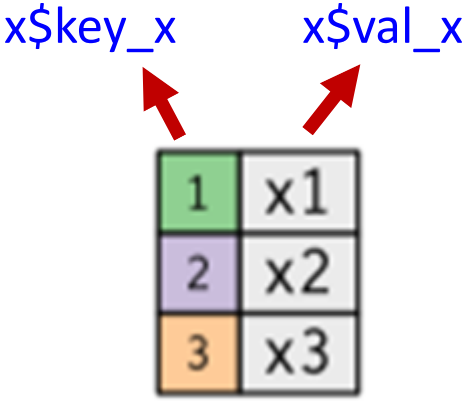
Data frame y

SDS230:download_data('x_y_join.rda')

Left and right tables

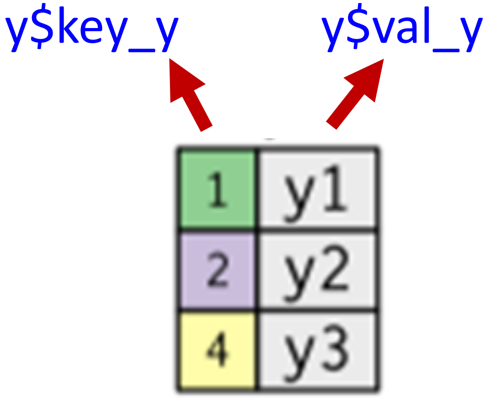
Suppose we have two data frames called x and y

- x have two variables called key_x, and val_x
- y has two variables called key_y and val_y



x$key_x	x$val_x
1	x1
2	x2
3	x3

Data frame x



y$key_y	y$val_y
1	y1
2	y2
4	y3

Data frame y

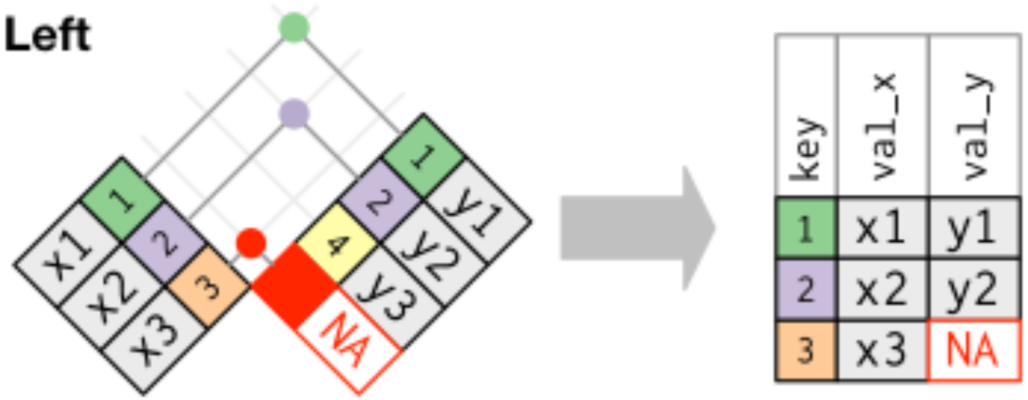
Joins have the general form:

```
join(x, y, by = c("key_x" = "key_y"))
```

Left joins

Left joins keep all rows in the left table.

Data from right table is added when there is a matching key, otherwise NA as added.



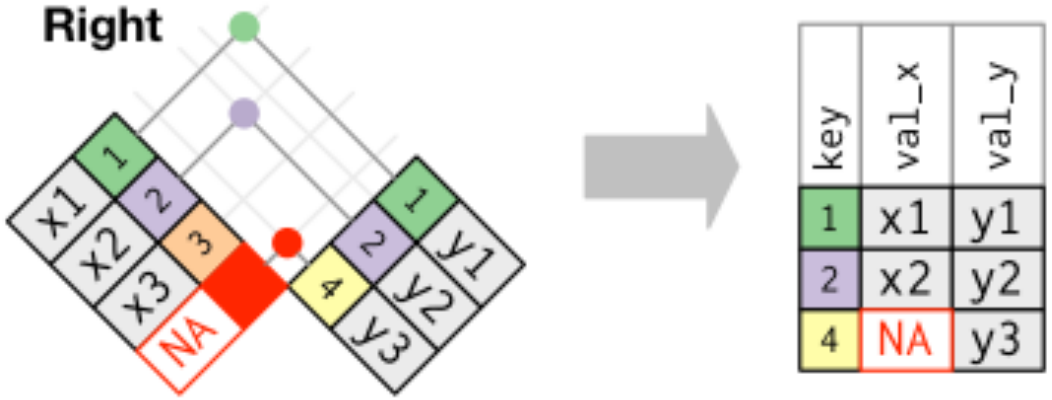
key	val_x	val_y
1	x1	y1
2	x2	y2
3	x3	NA

```
> left_join(x, y, by = c("key_x" = "key_y"))
```

Right joins

Right joins keep all rows in the right table.

Data from left table added when there is a matching key, otherwise NA as added.

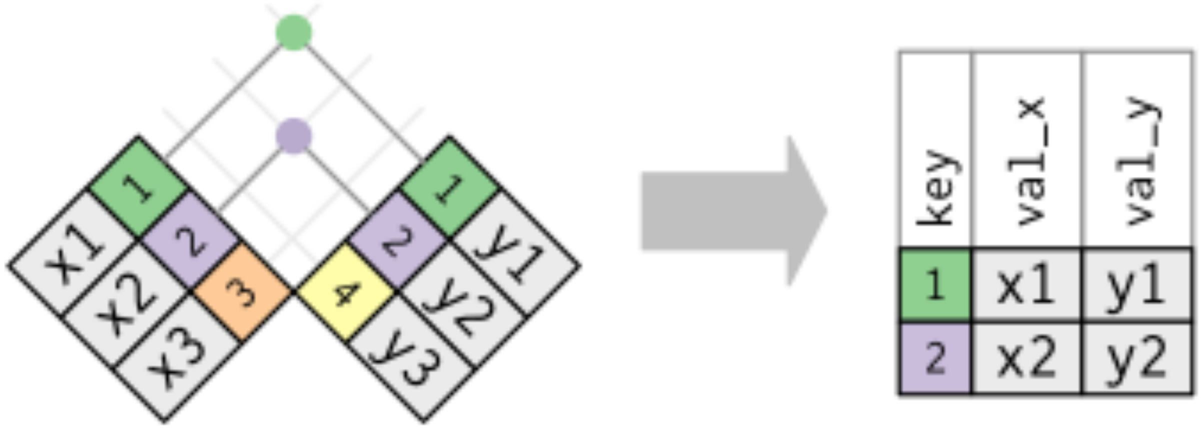


key	val_x	val_y
1	x1	y1
2	x2	y2
4	NA	y3

```
> right_join(x, y, by = c("key_x" = "key_y"))
```

Inner joins

Inner joins only keep rows in which there are matches between the keys in both tables.



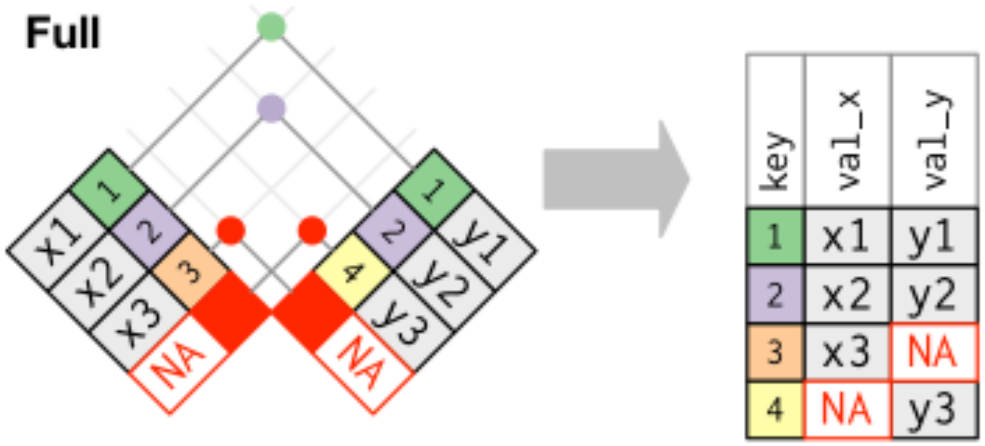
key	val_x	val_y
1	x1	y1
2	x2	y2

```
> inner_join(x, y, by = c("key_x" = "key_y"))
```

Full joins

Full joins keep all rows in both table.

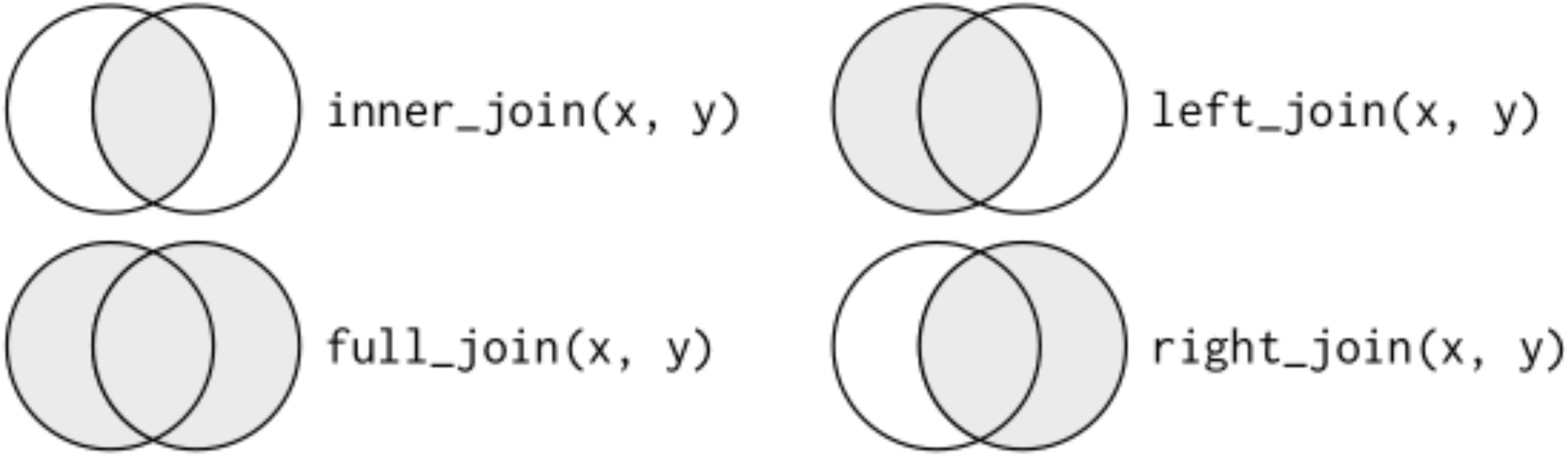
NAs are added where there are no matches.



key	val_x	val_y
1	x1	y1
2	x2	y2
3	x3	NA
4	NA	y3

```
> full_join(x, y, by = c("key_x" = "key_y"))
```

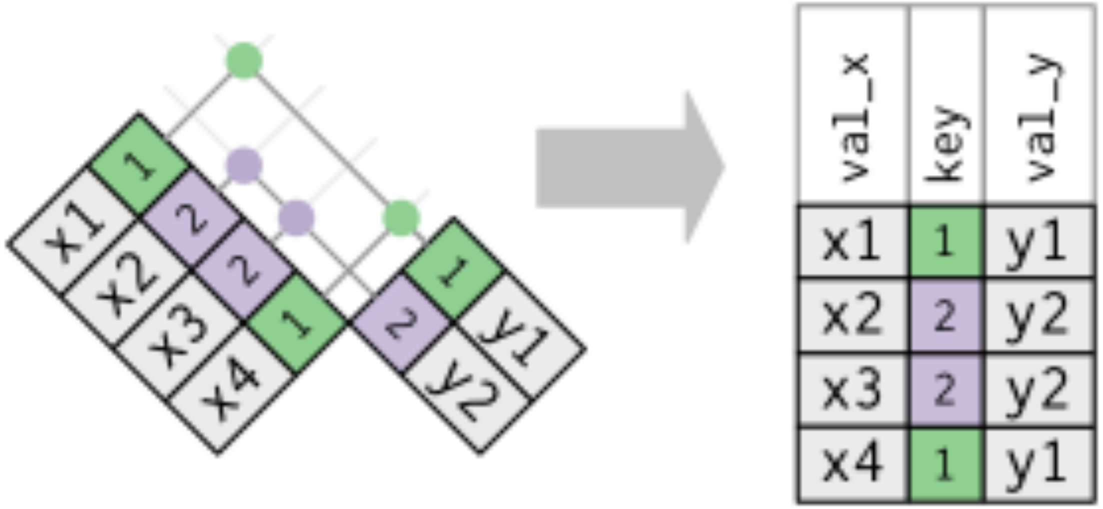
Summary



Duplicate keys

Duplicate keys are useful if there is a one-to-many relationship

- Duplicates are usually in the left table.

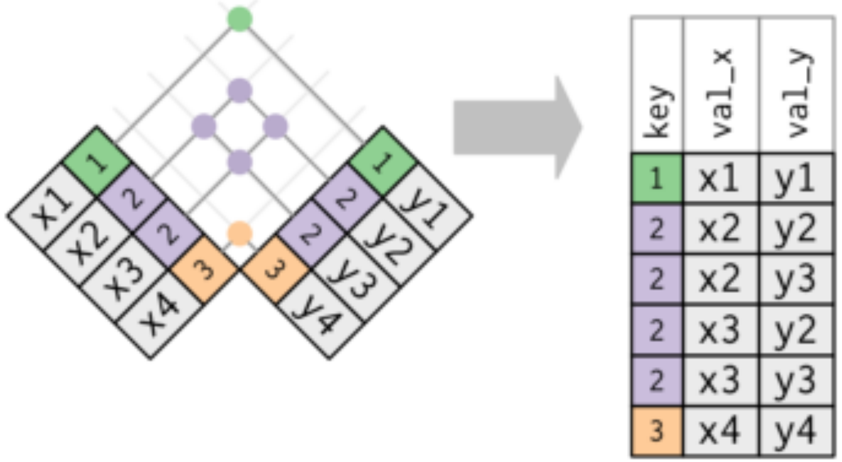


val_x	key	val_y
x1	1	y1
x2	2	y2
x3	2	y2
x4	1	y1

Duplicate keys

If both tables have duplicate keys you get all possible combinations of joined values (Cartesian product).

- **This is usually an error!**



key	val_x	val_y
1	x1	y1
2	x2	y2
2	x2	y3
2	x3	y2
2	x3	y3
3	x4	y4

Always check the output after you join a table because even if there is not a syntax error you might not get the table you are expecting!

- You can check how many rows a data frame has using the nrow() function

Duplicate keys

To deal with duplicate keys in both tables, we can join the tables using multiple keys in order to make sure that each row is uniquely specified.

We can do this using the syntax:

```
join(x2, y2, by = c("key1_x" = "key1_y", "key2_x" = "key2_y"))
```

Duplicate keys

```
>  x2 <- data.frame(key1_x = c(1, 2, 2),
                    key2_x = c("a", "a", "b"),
                    val_x = c("y1", "y2", "y3"))

>  y2 <- y2 <- data.frame(key1_y = c(1, 2, 2, 3, 3),
                    key2_y = c("a", "a", "b", "a", "b"),
                    val_y = c("y1", "y2", "y3", "y4", "y5"))
```

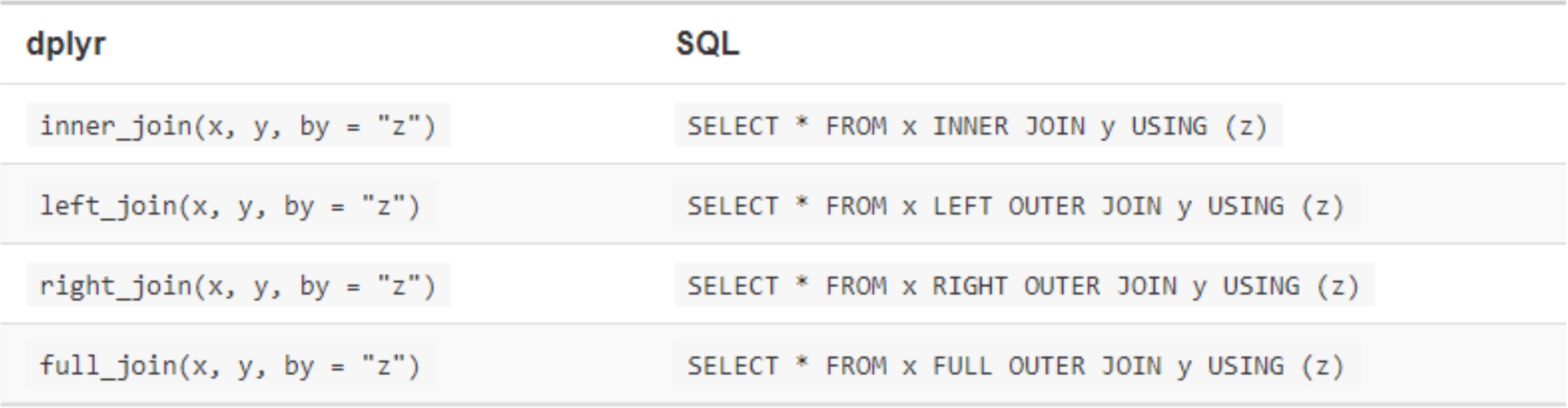
```
> left_join(x2, y2, c("key1_x" = "key1_y"))

> left_join(x2, y2, c("key1_x" = "key1_y", "key2_x" = "key2_y"))
```

Structured Query Language

Having multiple tables that can be joined together is common in Relational Database Systems (RDBS).

- A common language used by RDBS is Structured Query Language (SQL)



dplyr	SQL
`inner_join(x, y, by = "z")`	`SELECT * FROM x INNER JOIN y USING (z)`
`left_join(x, y, by = "z")`	`SELECT * FROM x LEFT OUTER JOIN y USING (z)`
`right_join(x, y, by = "z")`	`SELECT * FROM x RIGHT OUTER JOIN y USING (z)`
`full_join(x, y, by = "z")`	`SELECT * FROM x FULL OUTER JOIN y USING (z)`