



# HOW TO LEARN R?

IN CHATGPT ERA

Ahmad-Reza Ehyaei



# Why R?

# R Advantages



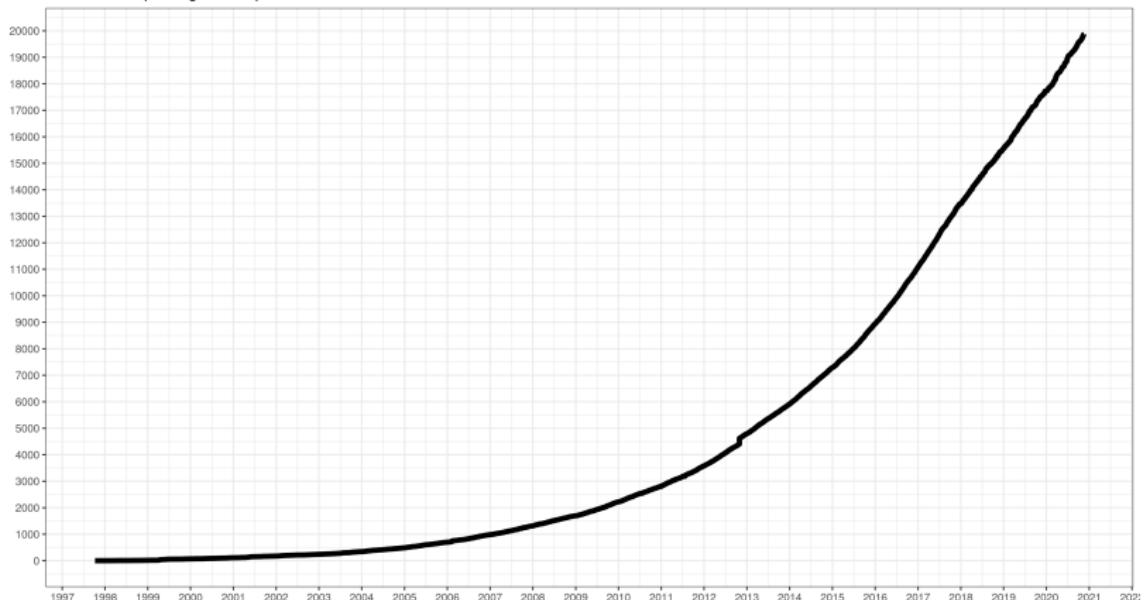
- ▶ Designed for statistics
- ▶ Strong data visualization capabilities
- ▶ A rich environment for dynamic documents
- ▶ Smart Developers!

Tool	Similar superhero	Similar powers in common
		<ul style="list-style-type: none"><li>• Detective work</li><li>• Intelligence</li><li>• Cunning</li><li>• Usage of tools</li><li>• More brain than muscle</li></ul>
		<ul style="list-style-type: none"><li>• Muscle power</li><li>• Super strength</li><li>• Elegance</li><li>• Wide range</li><li>• More muscle than brain</li></ul>

# Frontiers of Statistics



Number of R packages ever published on CRAN



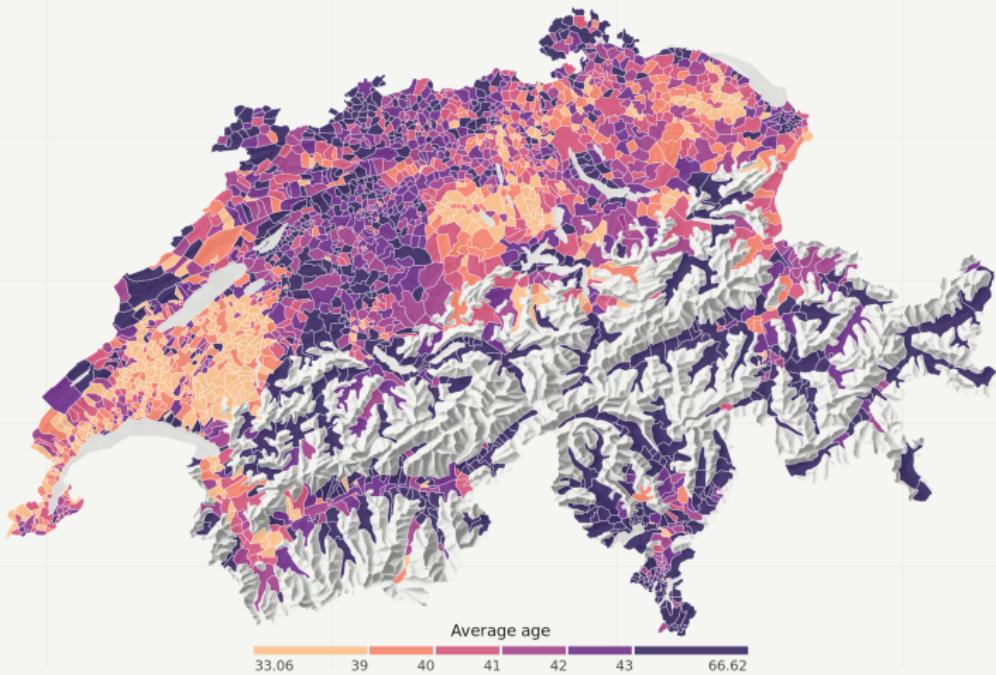
# Visualization



# Visualization



Switzerland's regional demographics  
Average age in Swiss municipalities, 2015

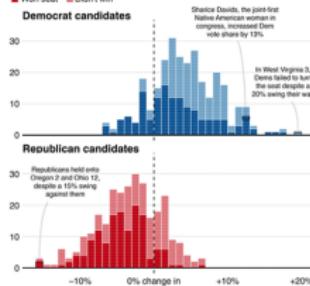


Map CC-BY-SA; Author: Timo Grossenbacher (@grssnbchr), Geometries: ThemenKart, BFS; Data: BFS, 2016; Relief: swisstopo, 2016

# Visualization

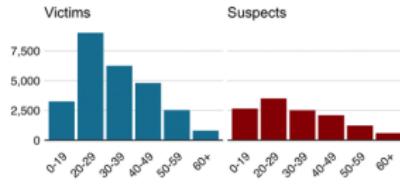


**Blue wave**  
■ Won seat ■ Didn't win  
**Democrat candidates**



## Homophobic hate crimes are mainly committed by young people on young people

Number in each age group 2014 - 2017



NB. Analysis excludes 3,474 victims and 2,003 suspects whose ages are unknown  
Source: BBC Freedom of Information requests to UK police forces

## Results of the 2018 election



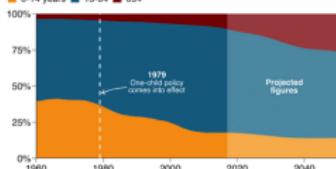
Source: Reuters

BBC

## Breakdown of China's population by age group

Proportion of total population (1960-2050)

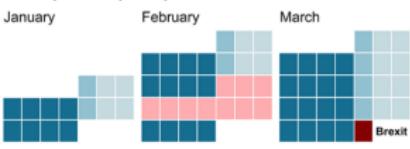
■ 0-14 years ■ 15-64 ■ 65+



Source: The World Bank

## The Commons has 36 normal working days until Brexit

■ Monday to Thursday ■ Friday ■ Weekend ■ Recess



Note: The House of Commons sometimes sits on Fridays to debate individual MPs' bills

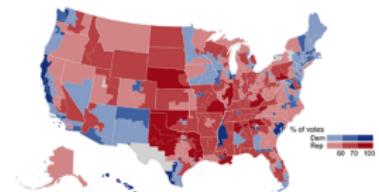
Source: Parliament

## Democrats take the House

Dem 232

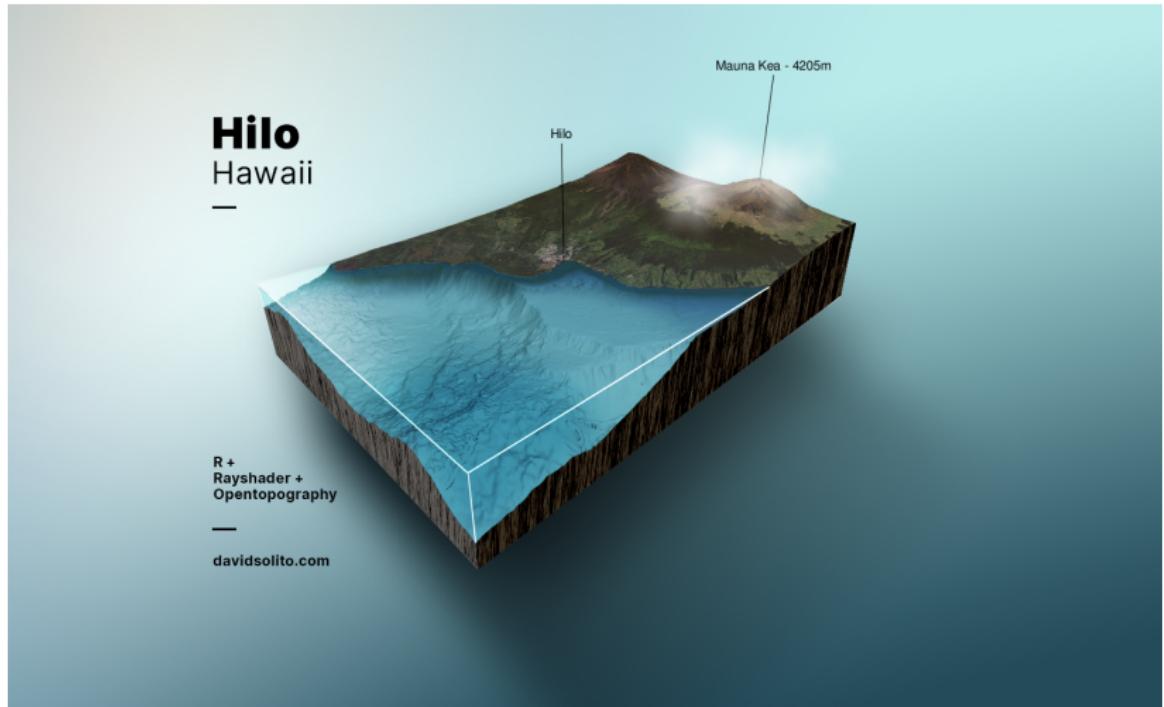
218 to win

Rep 198



BBC

# Visualization



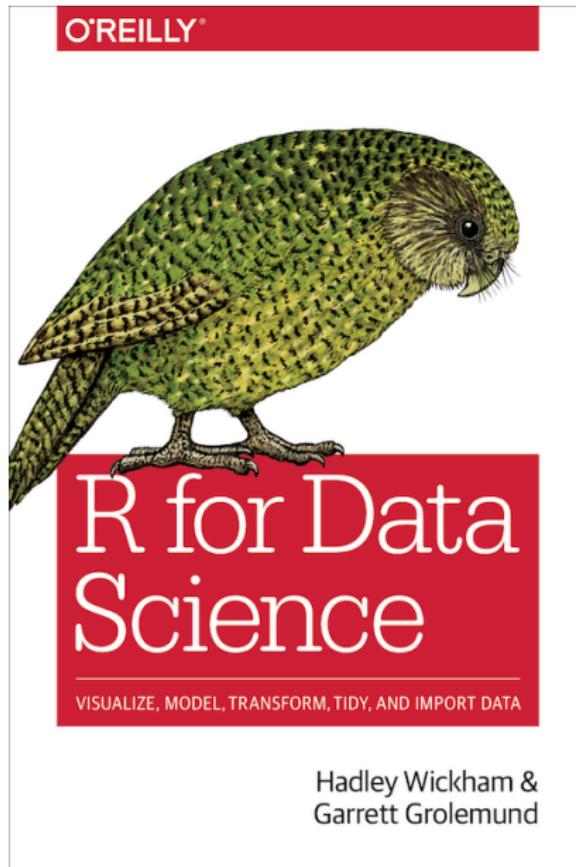




## R Studio<sup>®</sup> Team

RStudio's recommended  
solution for every professional  
data science team.

# How To Learn R





# Cheatsheets

## Data visualization with ggplot2 :: CHEAT SHEET

### Basics

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



To display values, map variables in the data to visual properties of the geom (**aesthetics**) like **size**, **color**, and **x** and **y** locations.



Complete the template below to build a graph.

```
ggplot(data = DATA) +  
  GEOGRAPHICAL_FUNCTIONS mapping = aes(MAPPINGS),  
  stat = STAT, position = POSITIONS) +  
  COORDINATE_FUNCTIONS +  
  FACET_FUNCTIONS +  
  SCALE_FUNCTIONS +  
  THEME_FUNCTIONS
```

Not required, some functions have defaults supplied

ggplot(data = mtcars, aes(x = cyl, y = hwy)) begins a plot that you finish by adding layers to. Add one geom function per layer.

last\_plot() Returns the last plot.

ggname("plot.png", width = 5, height = 5) Saves last plot as 5x5 file type "plot.png" in working directory. Matches file type to file extension.

### Aes

Common aesthetic values.

```
color + fill + string ("red", "#RRGGBB")  
linetype + integer or string (0 = "blank", 1 = "solid",  
2 = "dashed", 3 = "dotted", 4 = "dashdot", 5 = "longdash",  
6 = "twodash")  
lineend + string ("round", "butt", "square")  
linejoin + string ("round", "miter", "bevel")  
size + integer (line width in mm)  
shape + integer (shape name or  
a single character (""))
```



### Geoms

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

#### GRAPHICAL PRIMITIVES

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

a + geom\_blank() and a + expand\_limits()  
Ensure limits include values across all plots.

b + geom\_curve(aes(yend = lat + 1,  
xend = long + 1), curvature = 1) -> send, y, yend,  
alpha, angle, curve, curvature, color, linetype, size

a + geom\_diamond(aes(method = "lrc"),  
lineend = "round", linearmethod = 1)

a + geom\_polygon(aes(alpha = 50)) -> x, y, alpha,  
color, fill, group, linetype, size

b + geom\_rect(aes(xmin = long, ymin = lat,  
xmax = long + 1, ymax = lat + 1)) -> xmin, xmax,  
ymin, ymax, alpha, color, fill, linetype, size

b + geom\_raster(aes(xmin = long, ymin = lat,  
xmax = long + 1, ymax = lat + 1)) -> xmin, xmax,  
ymin, ymax, alpha, color, fill, linetype, size

a + geom\_ribbons(aes(ymin = unemploy - 90,  
ymax = unemploy + 90)) -> x, ymin, ymax,  
alpha, color, fill, group, linetype, size

a + geom\_text(aes(label = cyl), nudge\_x = 1,  
nudge\_y = 1) -> x, label, alpha, angle, color,

family, fontface, fontweight, size, yshift

#### LINE SEGMENTS

common aesthetics: x, y, alpha, color, linetype, size

b + geom\_abline(aes(intercept = 0, slope = 1))  
b + geom\_hline(aes(yintercept = lat))

b + geom\_vline(aes(xintercept = long))

b + geom\_segment(aes(yend = 0, x = 1, send = 1))

b + geom\_spoke(aes(cx = 1, cy = 1))

#### ONE VARIABLE

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

c + geom\_area(aes(fill = "blue"))  
c + geom\_alpha(aes(alpha = 0.5))

c + geom\_density(aes(fill = "gaussian"))  
c + geom\_dotplot(aes(alpha = 0.5))

c + geom\_freqpoly(aes(alpha = 0.5))

c + geom\_histogram(aes(binwidth = 5))  
c + geom\_ribbon(aes(sample = hwy))

c + geom\_qq(aes(sample = hwy))

#### discrete

d <- ggplot(mpg, aes(factor(cyl)))

d + geom\_bar()

d + geom\_bar(stat = "fill")

#### continuous

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

e + geom\_point()

e + geom\_quantile()

e + geom\_rug()

e + geom\_smooth(method = "loess")

e + geom\_text(aes(label = cyl), nudge\_x = 1,  
nudge\_y = 1) -> x, label, alpha, angle, color,

family, fontface, fontweight, size, yshift

#### TWO VARIABLES

both continuous

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

e + geom\_label(aes(label = cyl), nudge\_x = 1,  
nudge\_y = 1) -> x, label, alpha, angle, color,

family, fontface, fontweight, size, yshift

e + geom\_point()

e + geom\_hex()

e + geom\_hexbin()

e + geom\_hexgrid()

e + geom\_hexstroke()

e + geom\_hexbin()

e + geom\_hexgrid()

e + geom\_hexstroke()

#### one discrete, one continuous

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

f + geom\_col()

f + geom\_bar()

f + geom\_boxplot()

f + geom\_dotplot()

f + geom\_violin()

f + geom\_hex()

f + geom\_hexbin()

f + geom\_hexgrid()

#### both discrete

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

g + geom\_count()

g + geom\_hex()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

g + geom\_hexbin()

g + geom\_hexgrid()

g + geom\_hexstroke()

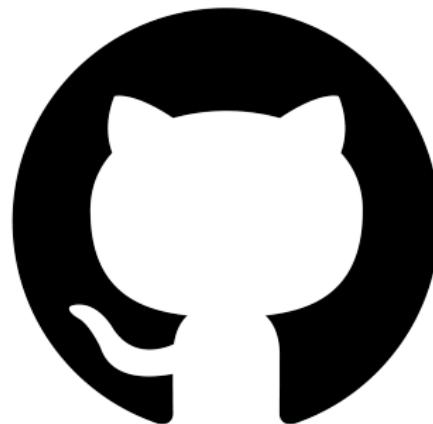
g + geom\_hexbin()

g + geom\_hexgrid()





# R Pubs



# GitHub



# Visual Studio Extension





