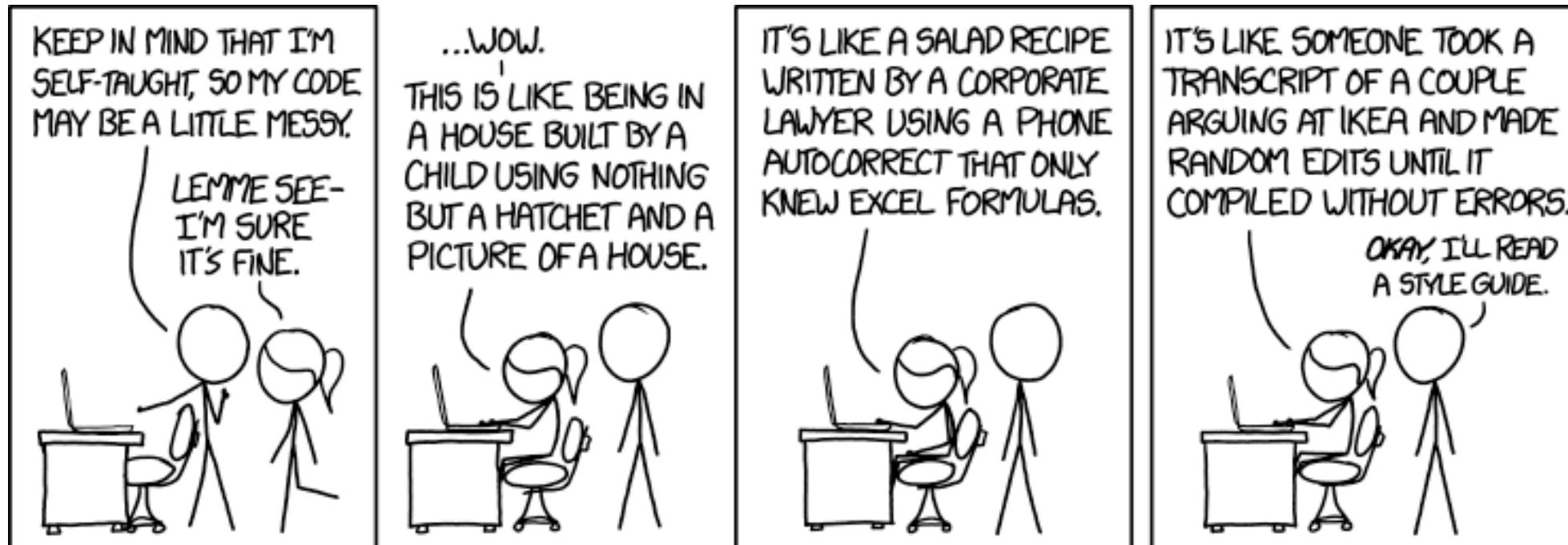


# Session 1

Lunch R workshop 2018.03.07

Max Lindmark & Philip Jacobson

# lunchR

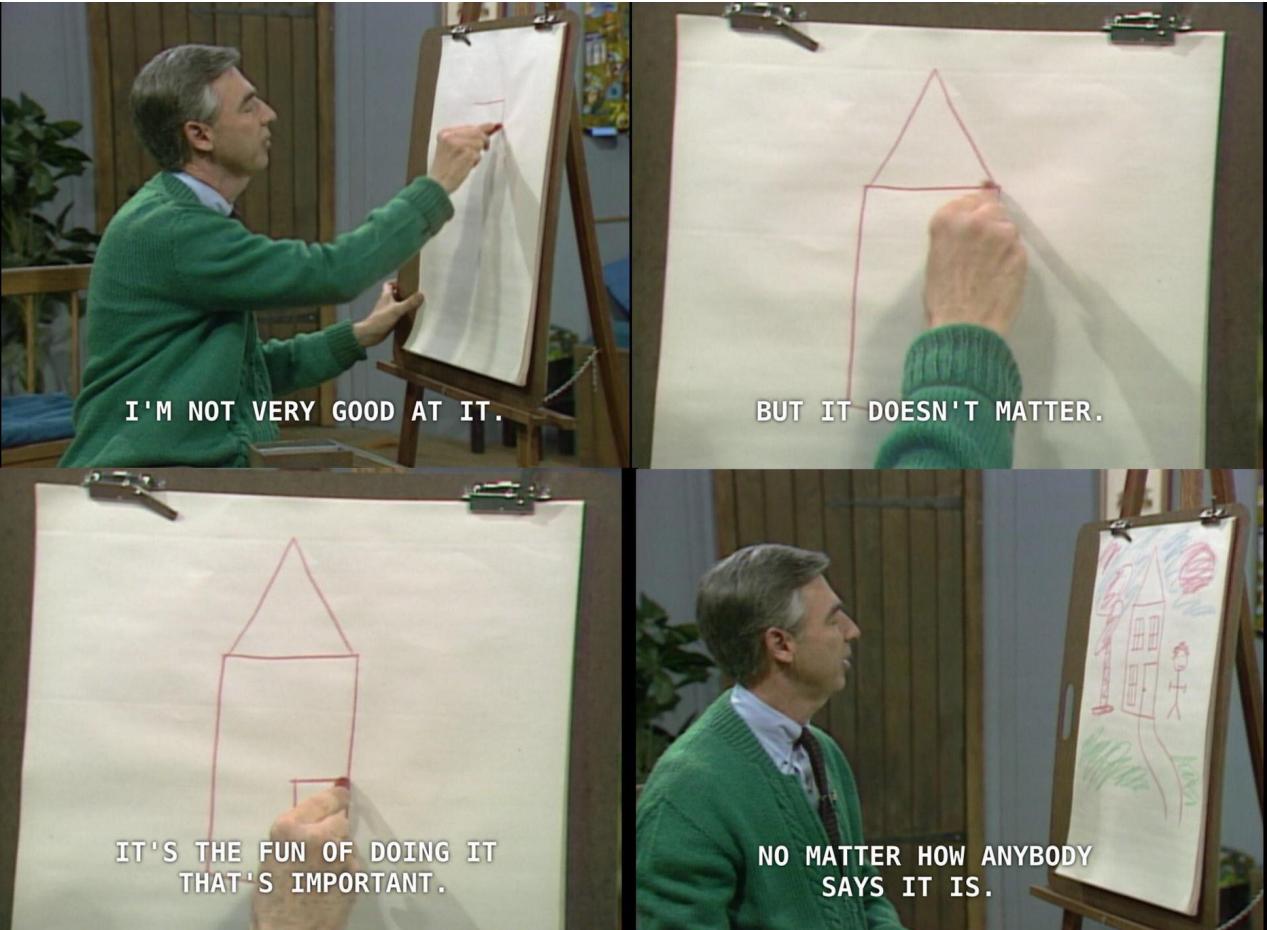




Welcome!

Sorry to disappoint but  
we are not experts...

LunchR



... we only enjoy doing it!  
(and we hope to learn  
from you too!)

## Learning outcomes

- What is R and benefits of using it
- Efficient data exploration
- Publishable figures
- Tips and tricks in R – how to learn more
- Products: shared documents (e.g. google docs / “Rstuga”?)

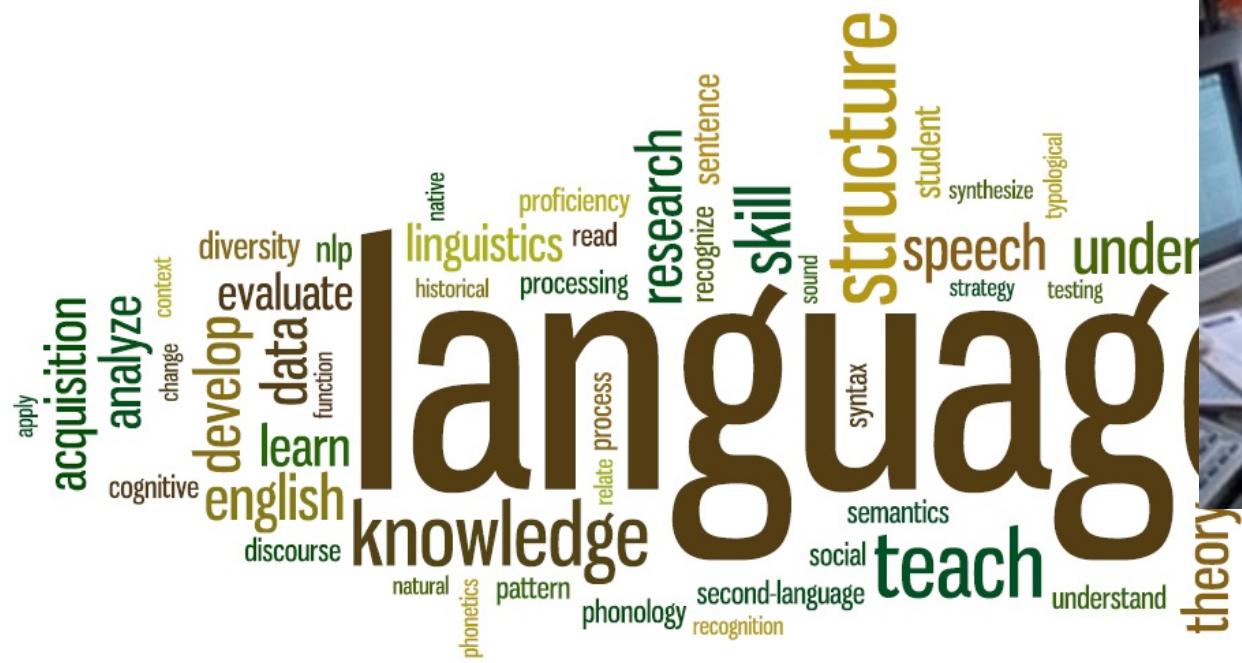
## Sessions

1. Intro to why R, data formats and plotting styles in R
2. Digging deeper into data exploration in R using base R and ggplot; focus on different graph styles
3. Tips and tricks in R – format your data for easy plotting
4. Creating publish-quality figures

## Structure

- Each session will start with a short presentation from us
- After the presentation, we will have exercises
- Ask us!

# What is R?



# New to R

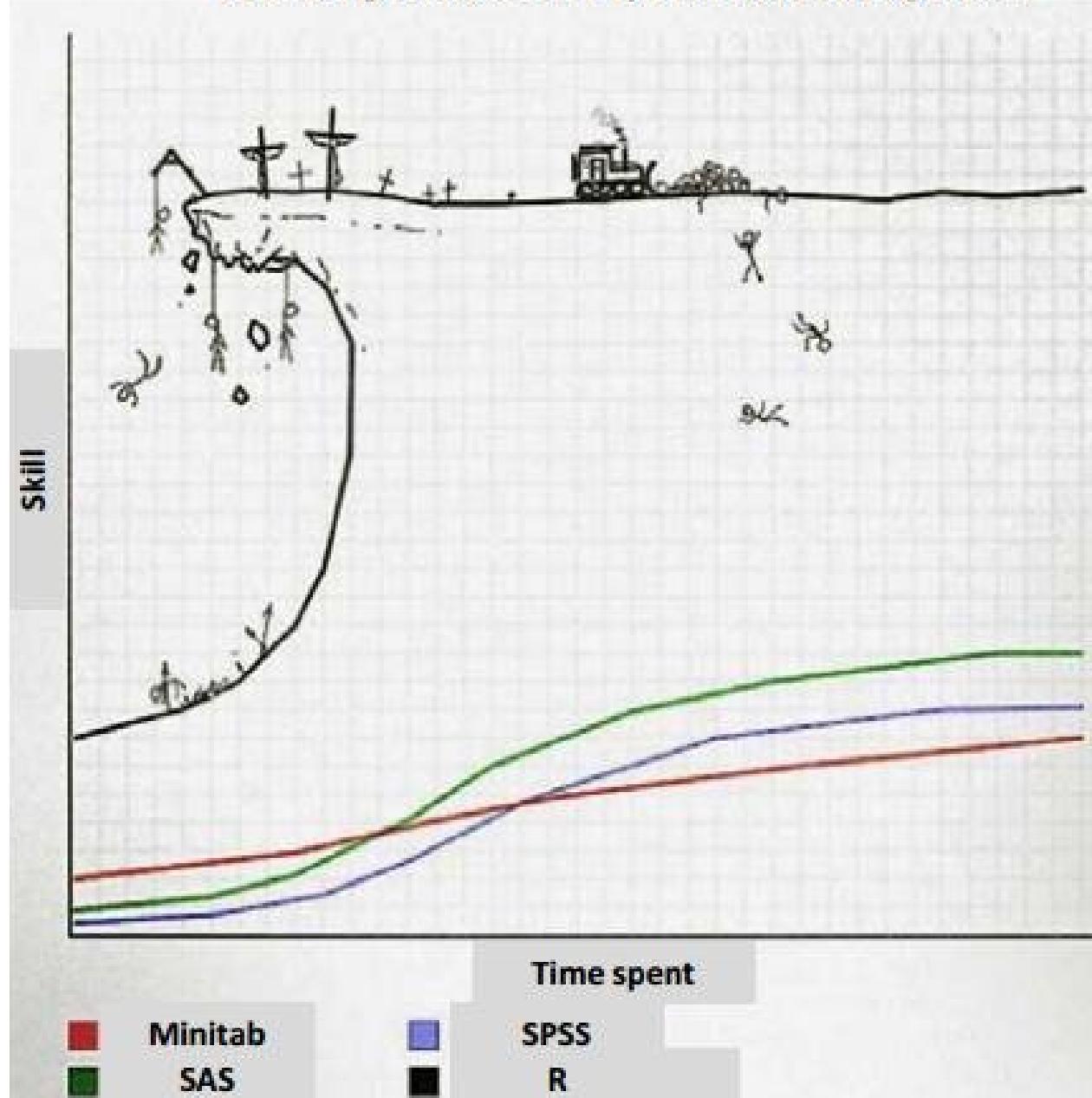
Error messages seem obscure

Help page – hard to read

Other people seem to have had not quite the same issue as you.

<http://socviz.co/index.html#preface>

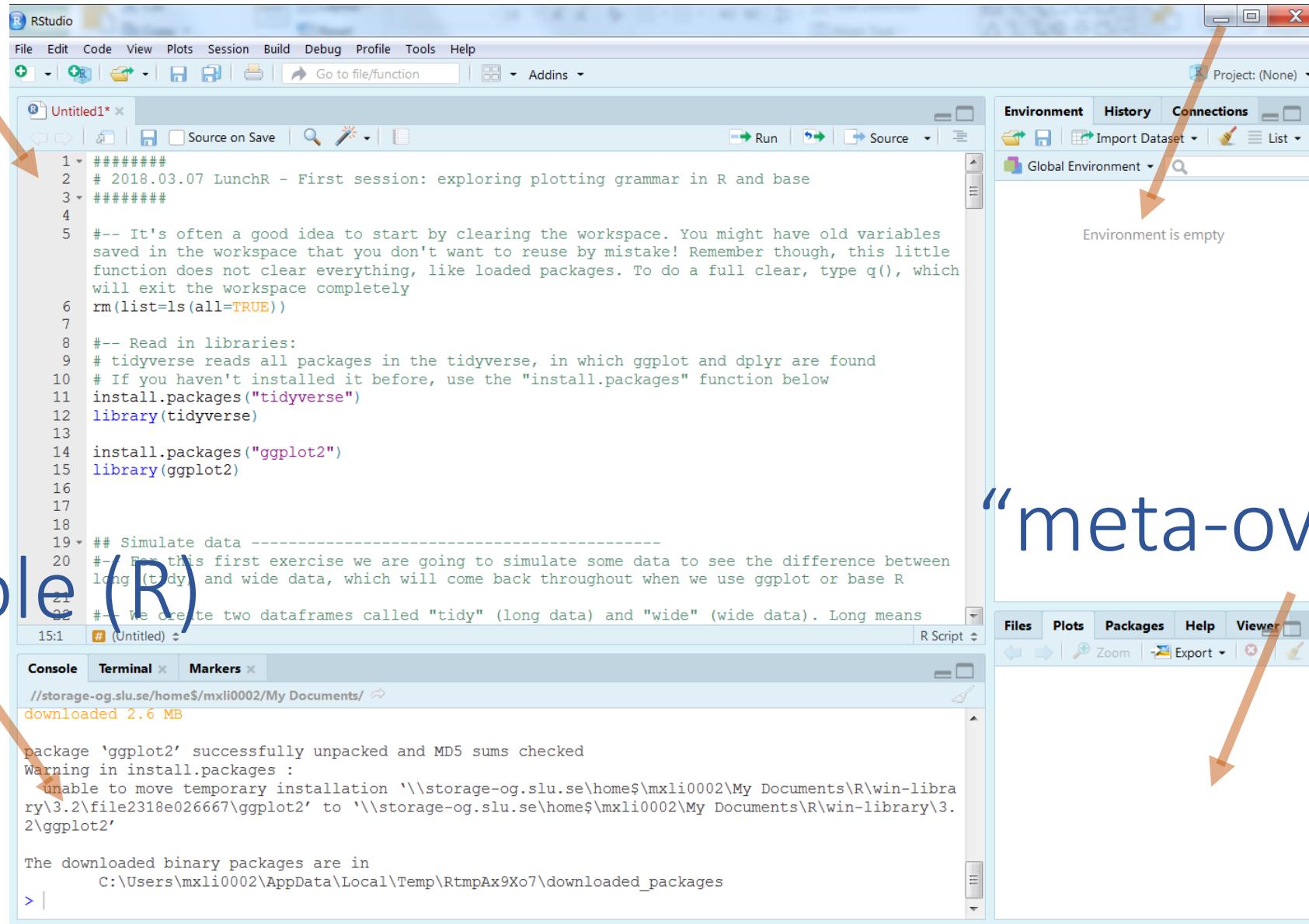
Learning Curves of Popular Stats Programs



# Script/text editor

# R studio “workspace overview”

## Console (R)

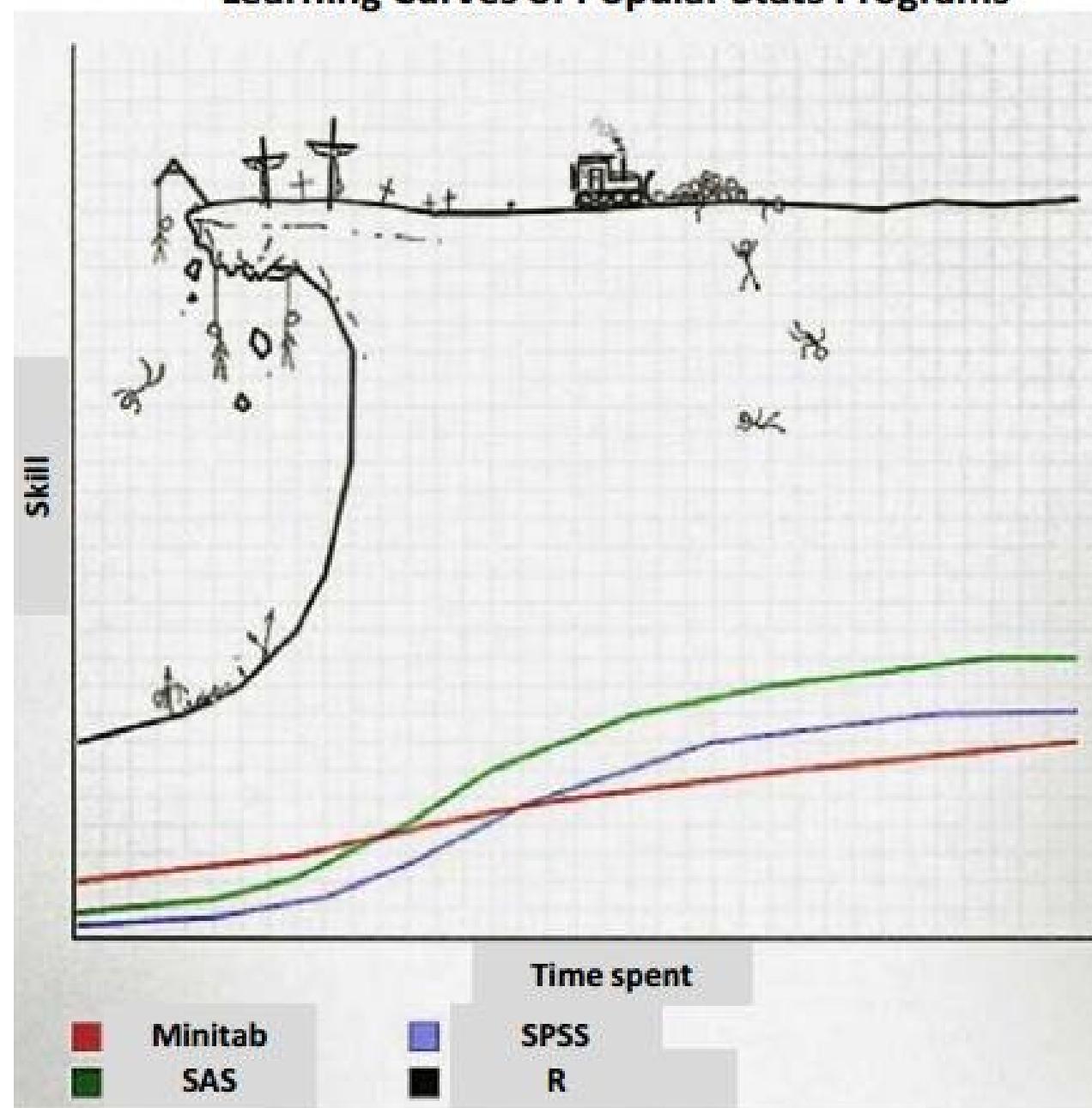


“meta-overview” 😊

# New to R

Is it worth it?

Learning Curves of Popular Stats Programs



# Why R?

Avoid manual data manipulation

Journals have started to request raw data and scripts

Easier to share your work with colleagues and reviewers

Transparent data handling and reproducible analysis

Easy to go back to data and analysis in the future compared to in other programs

Have fun (solve problems, get better every time you use it)!

# Why R?

Avoid manual data manipulation

- *Automatically in R*

Journals have started to request raw data and scripts

- *Automatically in R*

Easier to share your work with colleagues, reviewers and within groups (co-authors, students).

- *Automatically in R*

Transparent data handling and reproducible analysis

- *Automatically in R*

Easy to go back to data and analysis in the future compared to in other programs

- *Automatically in R*

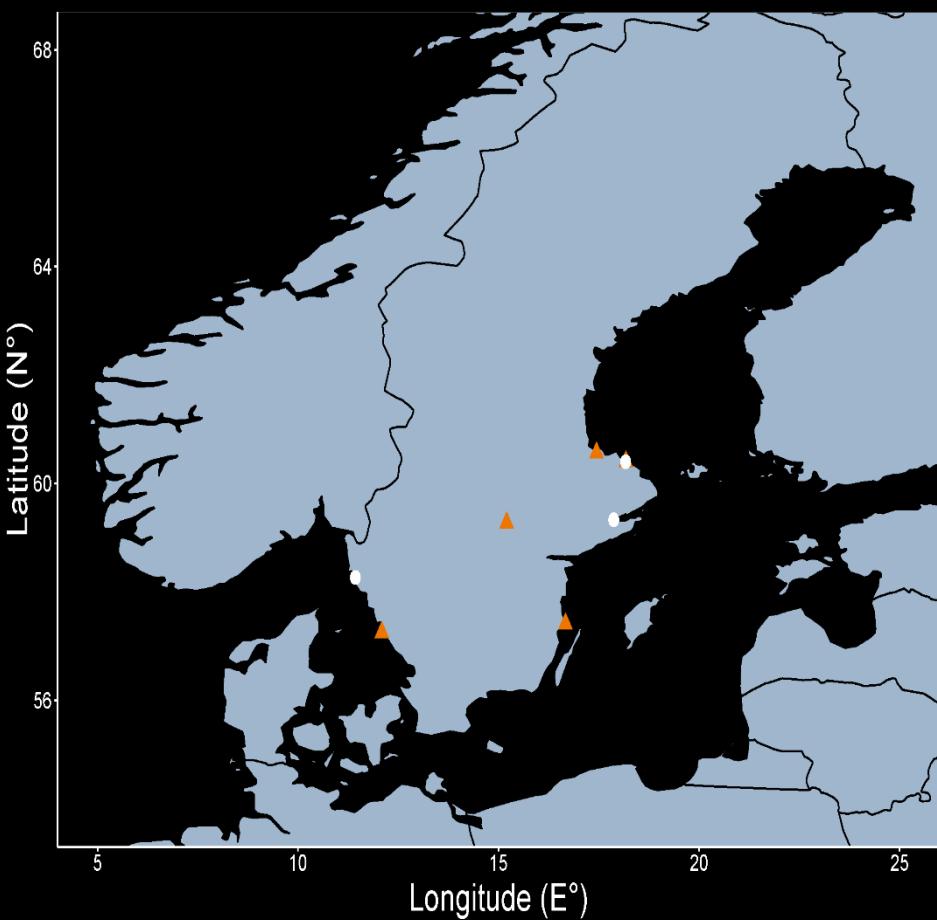
Read this excellent paper for more details:

Sandve GK, Nekrutenko A, Taylor J, Hovig E (2013) Ten Simple Rules for Reproducible Computational Research. *PLoS Comput Biol* 9(10): e1003285. doi:10.1371/journal.pcbi.1003285

<http://journals.plos.org/ploscompbiol/article/file?id=10.1371/journal.pcbi.1003285&type=printable>

# Producing figures in R

# Hello to the labs



```
x11()  
ggplot(data = map1, aes(x=long, y=lat, group = group))+  
  geom_polygon(data = map1, aes(x=long, y=lat, group = group), colour="Black", fill="slategray3") +  
  coord_cartesian(xlim = c(5,25), ylim = c(54,68)) +  
  scale_x_continuous(name="Longitude (E°)") +  
  scale_y_continuous(name="Latitude (N°)") +  
  geom_point(aes(x=17.873962, y=59.333397), size=4, shape=16, colour="white") + # Drottningholm  
  geom_point(aes(x=11.43098, y=58.26827), size=4, shape=16, colour="white") + # Lysekil  
  geom_point(aes(x=16.67141, y=57.416626), size=4, shape=17, colour="darkorange2") + # Simpevarp  
  geom_point(aes(x=18.17482, y=60.404104), size=4, shape=17, colour="darkorange2") + # Forsmark  
  geom_point(aes(x=12.089295, y=57.255269), size=4, shape=17, colour="darkorange2") + # Ringhals/väröbacka  
  geom_point(aes(x=17.444894, y=60.566823), size=4, shape=17, colour="darkorange2") + # Ä-by  
  geom_point(aes(x=15.203705, y=59.27781), size=4, shape=17, colour="darkorange2") + # Örebro  
  geom_point(aes(x=18.17, y=60.402875), size=4, shape=16, colour="white") + # Öregrund  
  theme_bw() +  
  theme(panel.background = element_rect(fill = "Black", colour = "white")) +  
  theme(panel.grid.major = element_blank()) +  
  theme(panel.grid.minor = element_blank()) +  
  theme(axis.title.x = element_text(vjust=-0.05, size=25, colour="white")) +  
  theme(axis.title.y = element_text(vjust=-0.05, size=25, colour="white")) +  
  theme(axis.text = element_text(size=15, colour="white")) +  
  theme(plot.background=element_rect(fill="black", colour=NA)) +  
  theme(axis.line = element_line(colour = "white", linetype = "solid")) +  
  theme(axis.ticks = element_line(colour="white"))
```

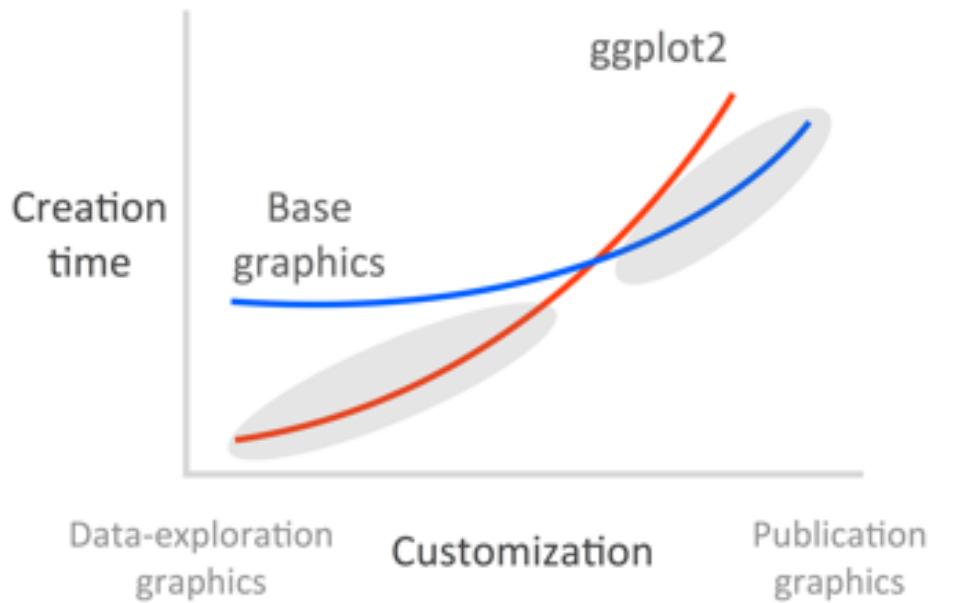
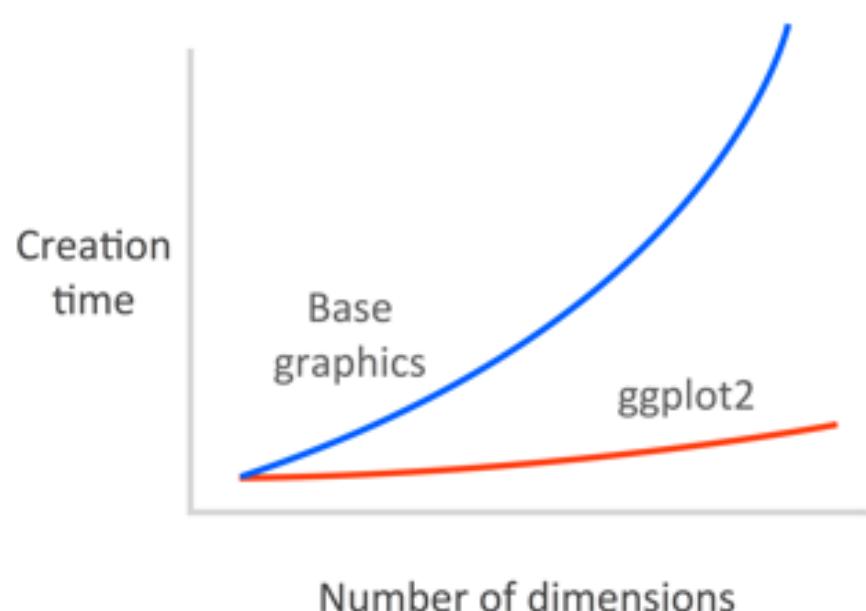
# Session 1

## ggplot vs base R



**ggplot** = simple and clean exploration of multidimensional data

**base R** = control everything (and sometimes you will need to, i.e. time consuming, figure-specific)



# Session 1

## ggplot

ggplot + tidy data = super fast and easy plotting

part of bigger tidyverse ecosystem with  
shared philosophy (see session 3!)



Hadley Wickham



## What are tidy data?

“tidy” data

	x	y	group
1	1	0.497868052	a
2	2	0.691927672	a
3	3	0.760313282	a
4	4	0.155401223	a
5	5	0.849457093	a
6	6	0.946817819	a
7	7	0.588419190	a
8	8	0.502250815	a
9	9	0.189779918	a
10	10	0.001836858	a
11	1	1.077578062	b
12	2	0.334111338	b
13	3	0.222741224	b
14	4	1.139136706	b
15	5	0.492948723	b
16	6	0.364326574	b
17	7	0.599102556	b
18	8	0.659575412	b
19	9	0.634030849	b
20	10	0.717009826	b

“wide data”

	x	a	b
1	1	0.497868052	1.0775781
2	2	0.691927672	0.3341113
3	3	0.760313282	0.2227412
4	4	0.155401223	1.1391367
5	5	0.849457093	0.4929487
6	6	0.946817819	0.3643266
7	7	0.588419190	0.5991026
8	8	0.502250815	0.6595754
9	9	0.189779918	0.6340308
10	10	0.001836858	0.7170098

Several ways to switch between tidy and wide data!

`gather()` [tidyverse]  
`melt()` [reshape2]

See session 3!

# Session 1

## ggplot – basic grammar

ggplot – this is the function. Tell it what to do!

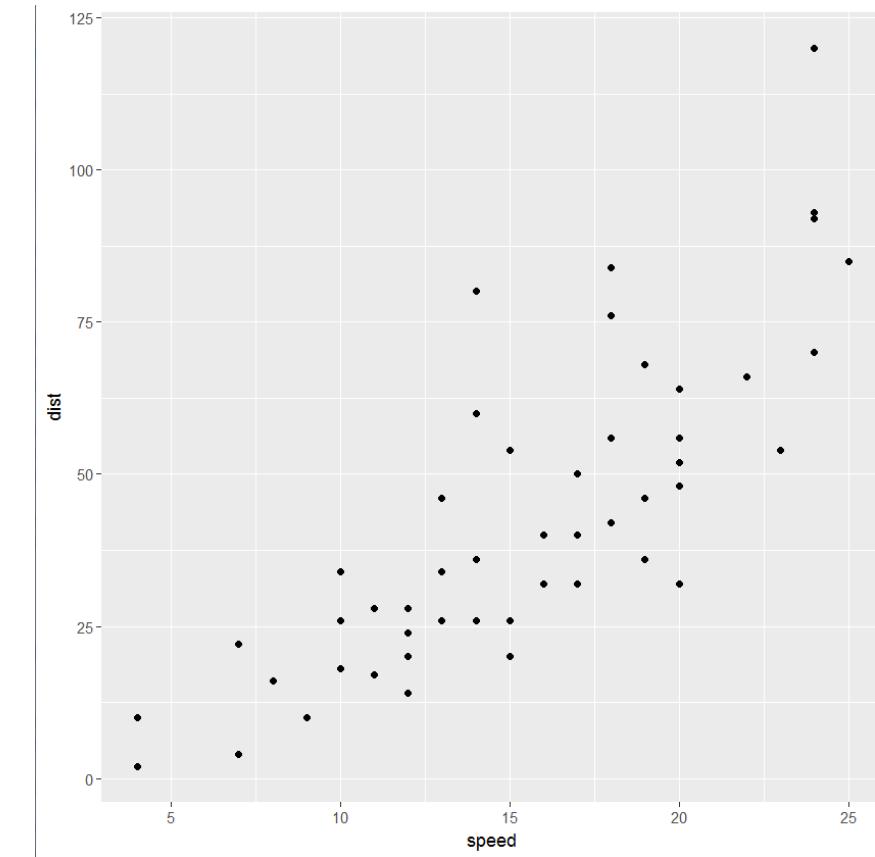
data – **WHAT DATA** to use

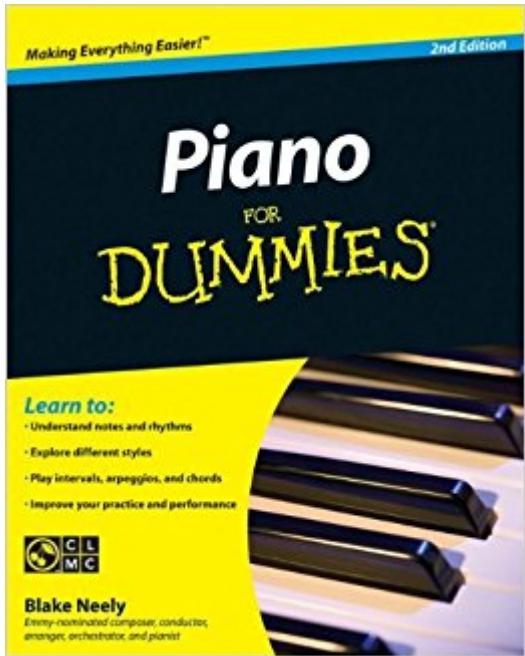
aes – **WHAT RELATIONSHIPS** to see ( $y \sim x$ )

geoms – **HOW WE WANT TO SEE** it

```
ggplot(data = df, aes(x, y)) +  
  geom_point()
```

*we will test this soon!*





Get your hands dirty!

# Session 1 - Practical part

## Producing figures in R