Data analysis and machine learning with R: an overview

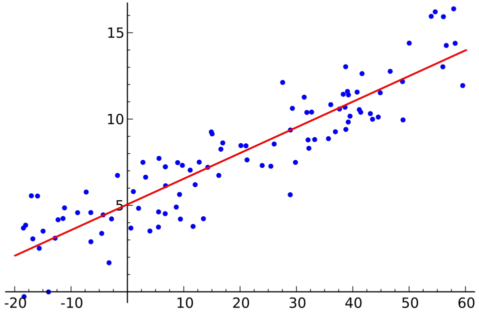
Enrique Garcia Ceja
May 18 2018

R is a programming language for statistical computing and graphics.

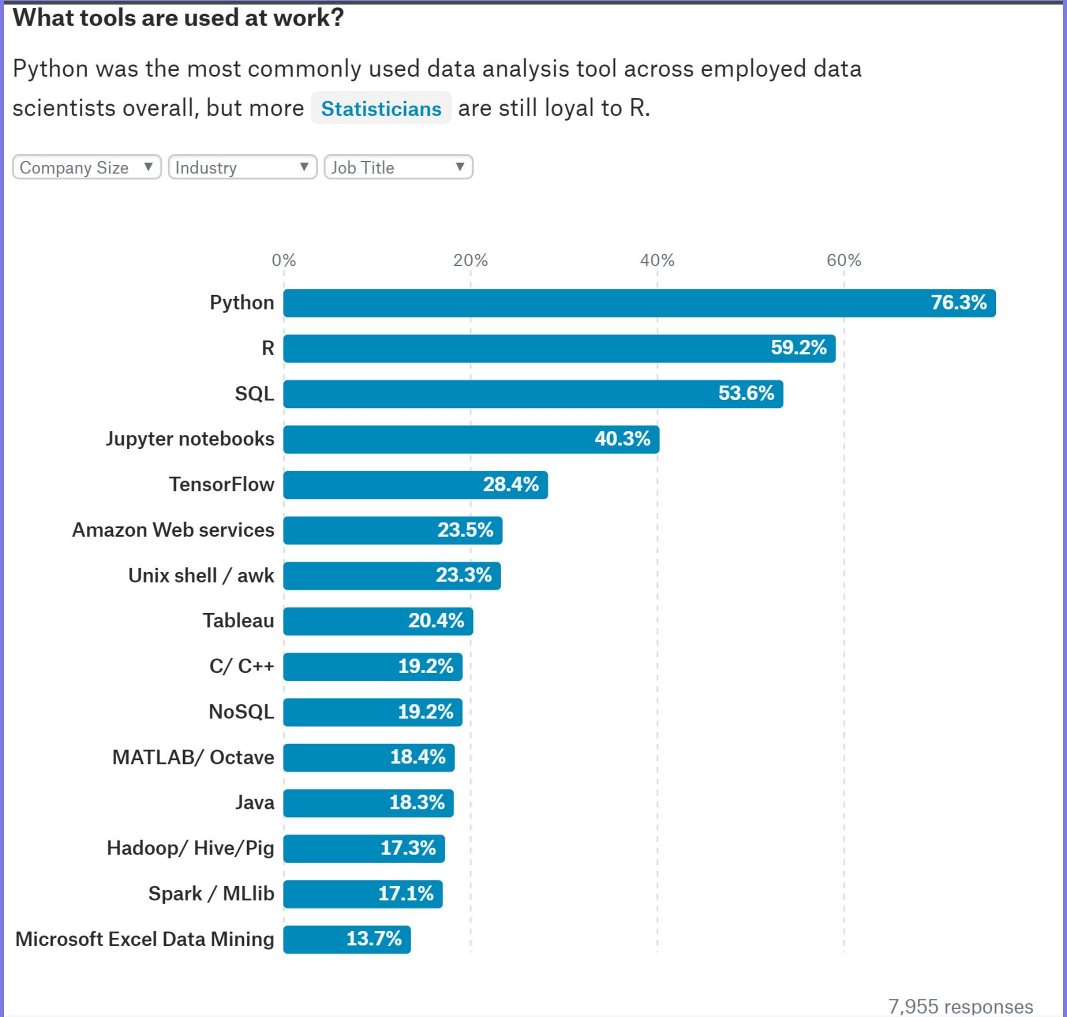
Is an implementation of the S programming language and was created by Ross Ihaka and Robert Gentleman.

Stable beta version released in 2000.

- Linear and nonlinear modelling
- Classical statistical tests
- Time-series analysis
- Classification
- Clustering
- Deep learning
- Visualization
- Spatial data



Some stats

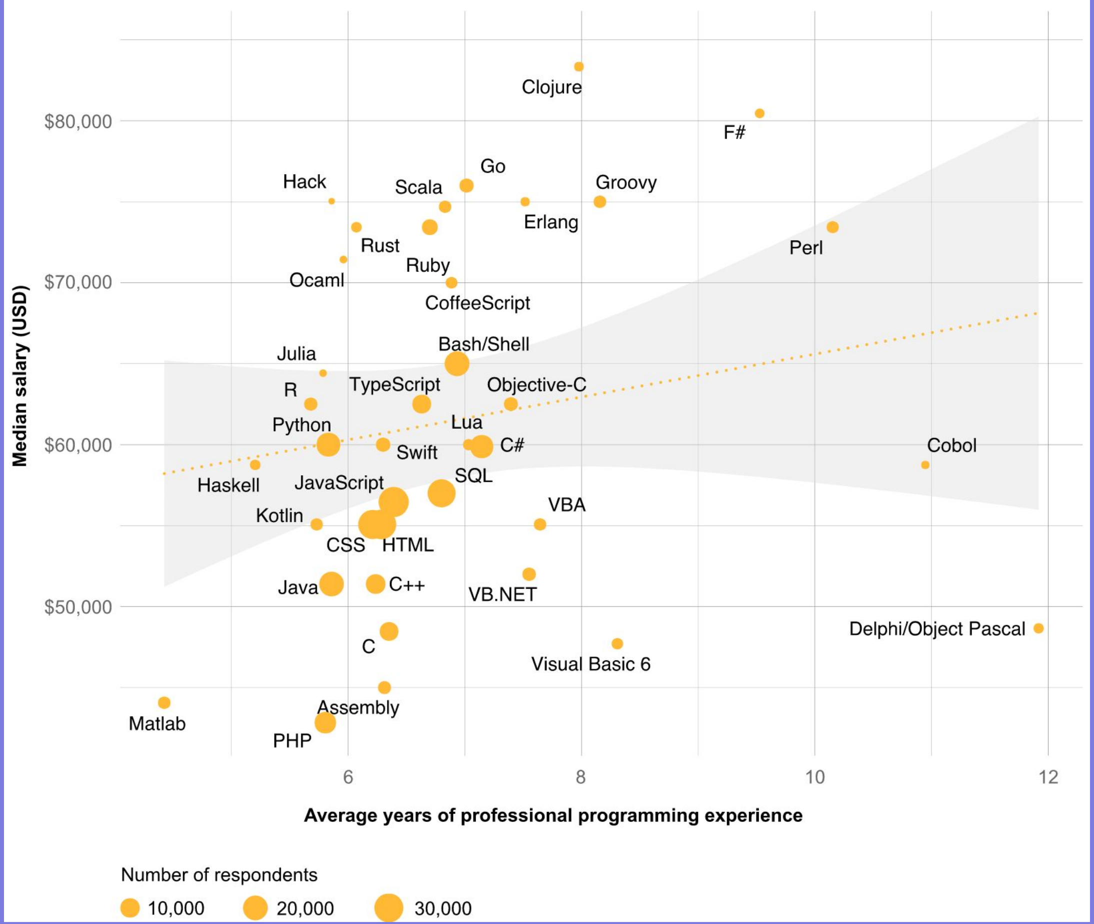


What tools are used at work?

Python was the most commonly used data analysis tool across employed data scientists overall, but more Statisticians are still loyal to R.

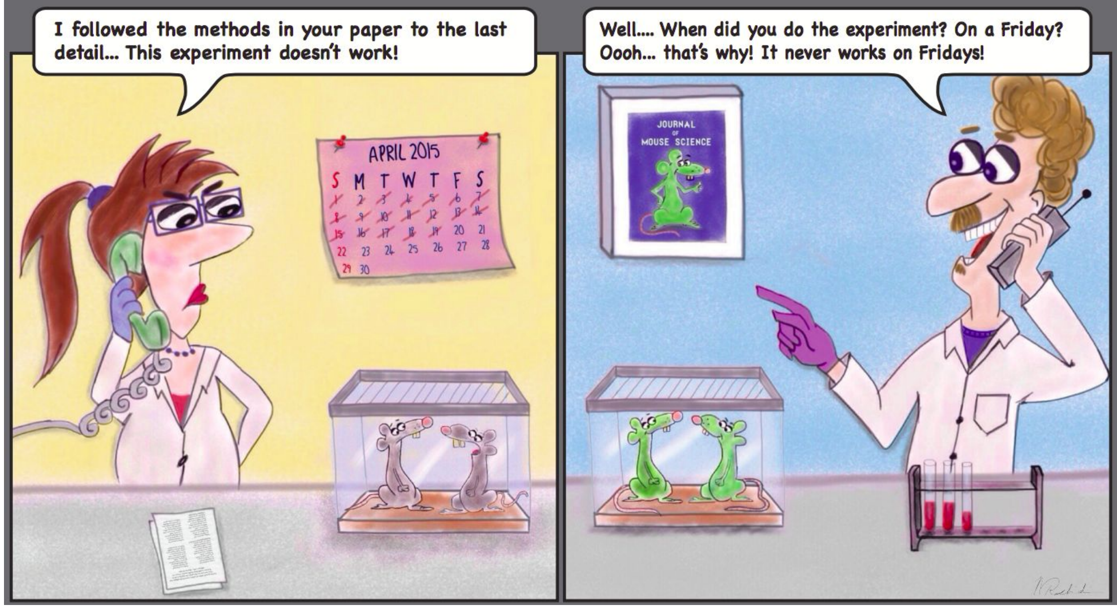
Kaggle survey 2017
https://www.kaggle.com/surveys/2017

Some stats



Stackoverflow developer survey 2018
https://insights.stackoverflow.com/survey/2018?utm_source=so-owned&utm_medium=meta&utm_campaign=dev-survey-2018-promotion#work-salary-and-experience-by-language

REPRODUCIBILITY

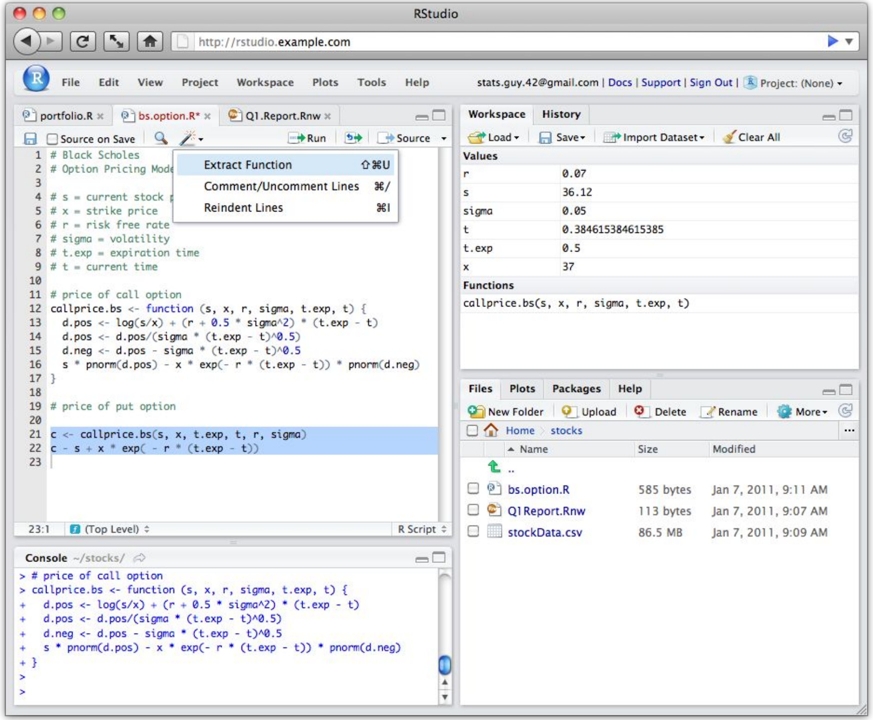

I followed the methods in your paper to the last detail... This experiment doesn't work!
APRIL 2015
S M T W T F S
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30
Well.... When did you do the experiment? On a Friday? Oooh... that's why! It never works on Fridays!
JOURNAL OF MOUSE SCIENCE

More than 70% of researchers have tried and failed to reproduce another scientist's experiments, and more than half have failed to reproduce their own experiments (Monya Baker, 2016).

The R ecosystem provides tools for reproducible research such as Markdown documents, interactive Notebooks, etc.

RStudio

One of the most popular IDE.



```
# Black Scholes
# Option Pricing Mode

# s = current stock p
# x = strike price
# r = risk free rate
# sigma = volatility
# t.exp = expiration time
# t = current time

# price of call option
callprice.bs <- function (s, x, r, sigma, t.exp, t) {
  d.pos <- log(s/x) + (r + 0.5 * sigma^2) * (t.exp - t)
  d.pos <- d.pos/(sigma * (t.exp - t)^0.5)
  d.neg <- d.pos - sigma * (t.exp - t)^0.5
  s * pnorm(d.pos) - x * exp(- r * (t.exp - t)) * pnorm(d.neg)
}

# price of put option

c <- callprice.bs(s, x, t.exp, t, r, sigma)
c - s + x * exp( - r * (t.exp - t))

```

```
> # price of call option
> callprice.bs <- function (s, x, r, sigma, t.exp, t) {
+   d.pos <- log(s/x) + (r + 0.5 * sigma^2) * (t.exp - t)
+   d.pos <- d.pos/(sigma * (t.exp - t)^0.5)
+   d.neg <- d.pos - sigma * (t.exp - t)^0.5
+   s * pnorm(d.pos) - x * exp(- r * (t.exp - t)) * pnorm(d.neg)
+ }
>
>
```

Values	
r	0.07
s	36.12
sigma	0.05
t	0.384615384615385
t.exp	0.5
x	37

Functions

callprice.bs(s, x, r, sigma, t.exp, t)

	Name	Size	Modified
	..		
□	bs.option.R	585 bytes	Jan 7, 2011, 9:11 AM
□	Q1Report.Rnw	113 bytes	Jan 7, 2011, 9:07 AM
□	stockData.csv	86.5 MB	Jan 7, 2011, 9:09 AM

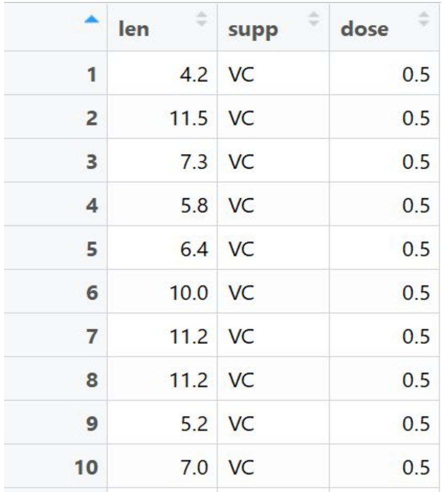
OUTLINE

- Basics
- R Notebooks
- Time series visualization
- Data frames
- Interactive presentations
- Machine learning
- Deep Learning with Keras

Data frames

A **data frame** is the most common data structure in R. It can be thought of as a table.

Columns can be of different types (numeric, string, date, boolean, etc).



	len	supp	dose
1	4.2	VC	0.5
2	11.5	VC	0.5
3	7.3	VC	0.5
4	5.8	VC	0.5
5	6.4	VC	0.5
6	10.0	VC	0.5
7	11.2	VC	0.5
8	11.2	VC	0.5
9	5.2	VC	0.5
10	7.0	VC	0.5

CODE

Code used in this presentation

https://github.com/enriquegit/data-analysis-r

References

- Monya Baker, "1,500 scientists lift the lid on reproducibility", Nature 533, 452-454 (26 May 2016) doi:10.1038/533452a