Comparative Computing

—

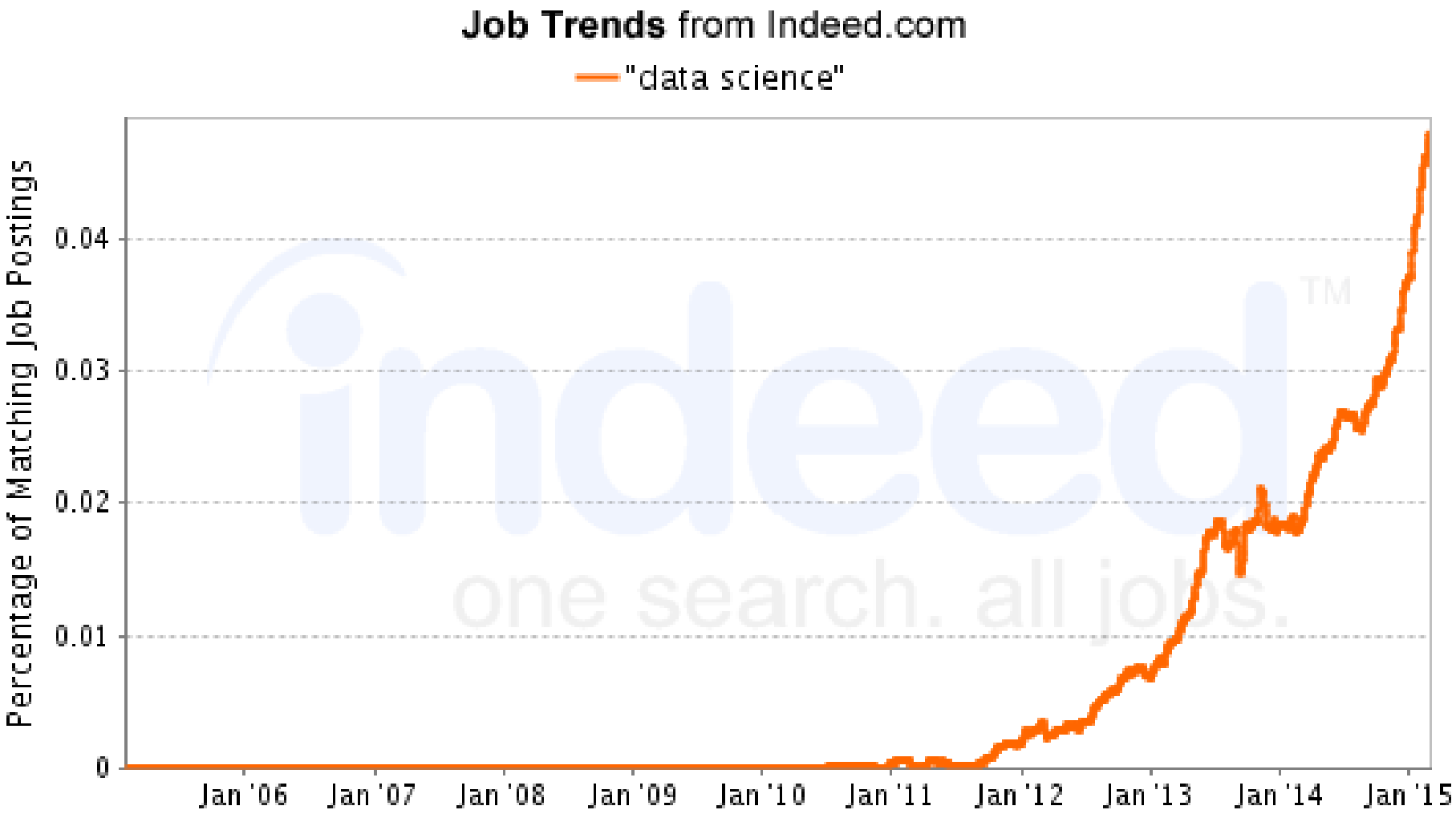
R, Python, Stata, and the shell

Winter Institute in Data Science and Big Data

Simon Heuberger

6 January 2021

"Data scientist: The sexiest job of the 21st century"



Source: https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

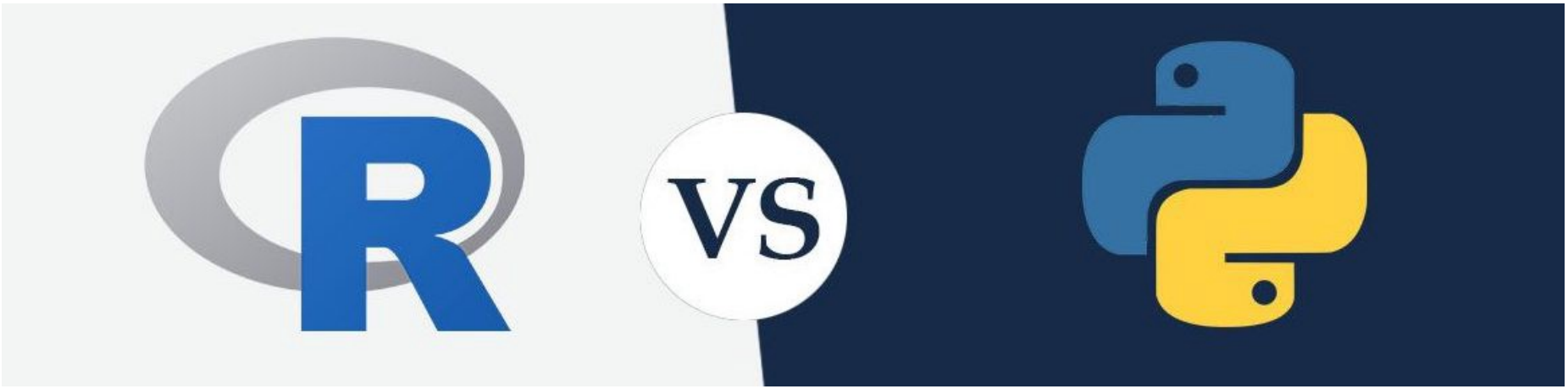
But what programming language to use?

- Widely spread in academia: `Stata`
- `R` and `Python`
- Often neglected add-on: The `shell`

Stata

- Proprietary (and expensive – $179/year for single student license)
- No choice of IDE
- Not really used in data science
- Nonetheless: What does it look like?
- Why no love for `Stata`?
  - Hard to analyse multiple datasets
  - Only most expensive versions work with large datasets
  - Limited resources/functionalities (e.g. predictive modelling, web scraping)
  - Not consistent with computer science programming (point-and-click)
  - Not open source
- "Data scientists rely on `Stata` because of its strong programming capabilities, reproducibility, extensibility, and interoperability"

Which leaves us with . . .



R: Lingua franca of statistics

Open-source
For statistical analyses
Academics, researchers, data scientists
Huge support community
1000s of packages (CRAN, GitHub)
Outstanding visualization
Advanced reporting (Markdown)

Can be slow
Steep learning curve
Finding right packages can take time
One IDE: RStudio

Source: https://www.datacamp.com/community/tutorials/r-or-python-for-data-analysis

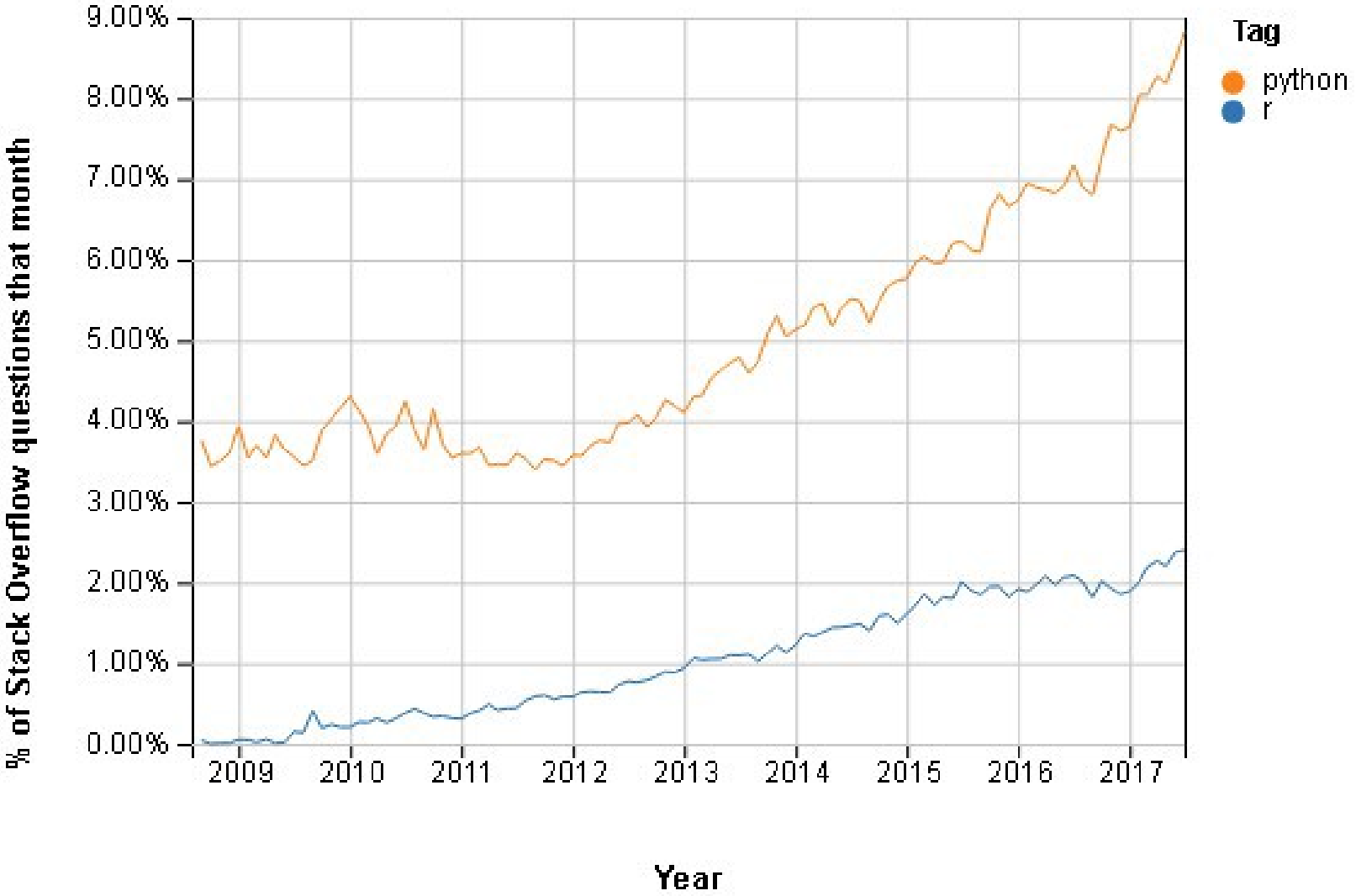
Python: Multi-purpose language

Open-source
For productivity and code readability
Programmers, developers, engineers
Huge support community
100,000s of packages (PyPI)
Moderate learning curve
Fast
Advanced deep/machine learning
Several IDEs: Spyder, Jupyter, Rodeo

Convoluted static visualization
Few(er) data science packages
Finding right packages can take time
Syntax changes between Python 2 and 3

Source: https://www.datacamp.com/community/tutorials/r-or-python-for-data-analysis

Popularity



Source: https://dzone.com/articles/r-or-python-data-scientists-delight

Jobs

Source: https://www.guru99.com/r-vs-python.html

Application

- Fit logistic regression, predict flower species based on measured features

Application

- Bootstrapping: Randomly resample 100,000 times from a population

Application

- File loading (4.8 GB)

```r
library(tidyverse)
start_time <- Sys.time()
df <- read_csv("library-collection-inventory.csv")
end_time <- Sys.time()
end_time - start_time
# Time difference of 6.814214 mins
```

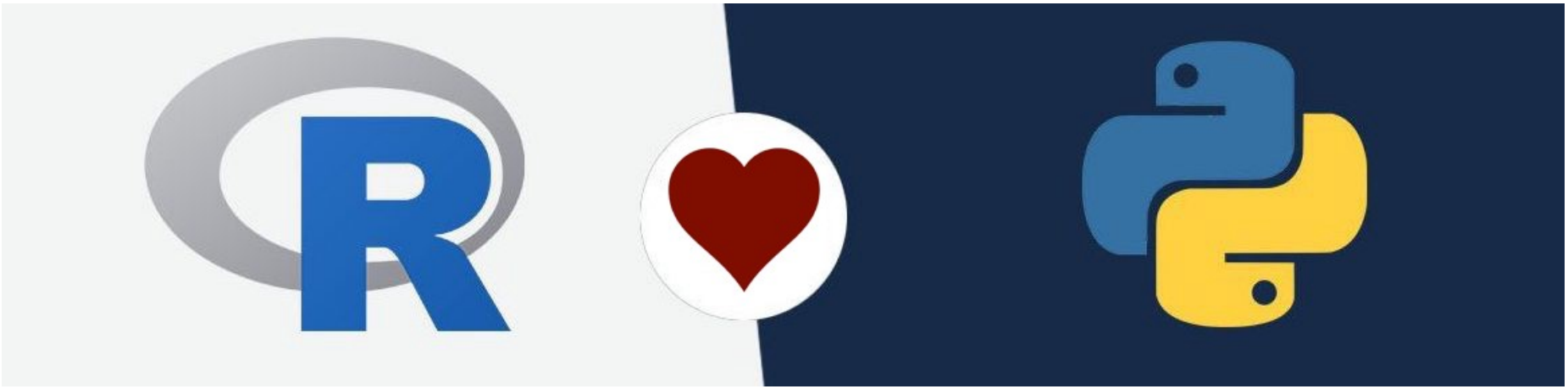
```python
import time
import pandas as pd
start = time.time()
y1 = pd.read_csv('library-collection-inventory.csv')
end = time.time()
print("Time difference of " + str(end - start) + " seconds")
# Time difference of 130.32760381698608 seconds
```

And the winner is . . .

. . . there isn't one

- One language isn't better than the other
- Both have pros and cons
- It all depends on what you need it for
  - What problems do you want to solve?
  - Which language do you have support for?
  - What are the net costs of learning the language?
- Researcher/Data Scientist? $\rightarrow$ `R`
- Developer/Programmer? $\rightarrow$ `Python`
- Best solution?

Use both!



Interwoven code: Machine learning

- Application: Build a random forest model that predicts wine quality
  - `R` for exploration because of `tidyverse` efficiency
  - `Python` for machine learning because of `sklearn` pipeline capability
  - `R` for visualization because of `Markdown` and `ggplot2`

Adapted from: https://www.business-science.io/business/2018/10/08/python-and-r.html

Add-on: the `shell`

- Open-source
- Around since the dawn of computers (more or less)
- Hugely beneficial
  - Remote machines
  - Cloud computing
  - Scripts that run for a long time