Big Data Analytics

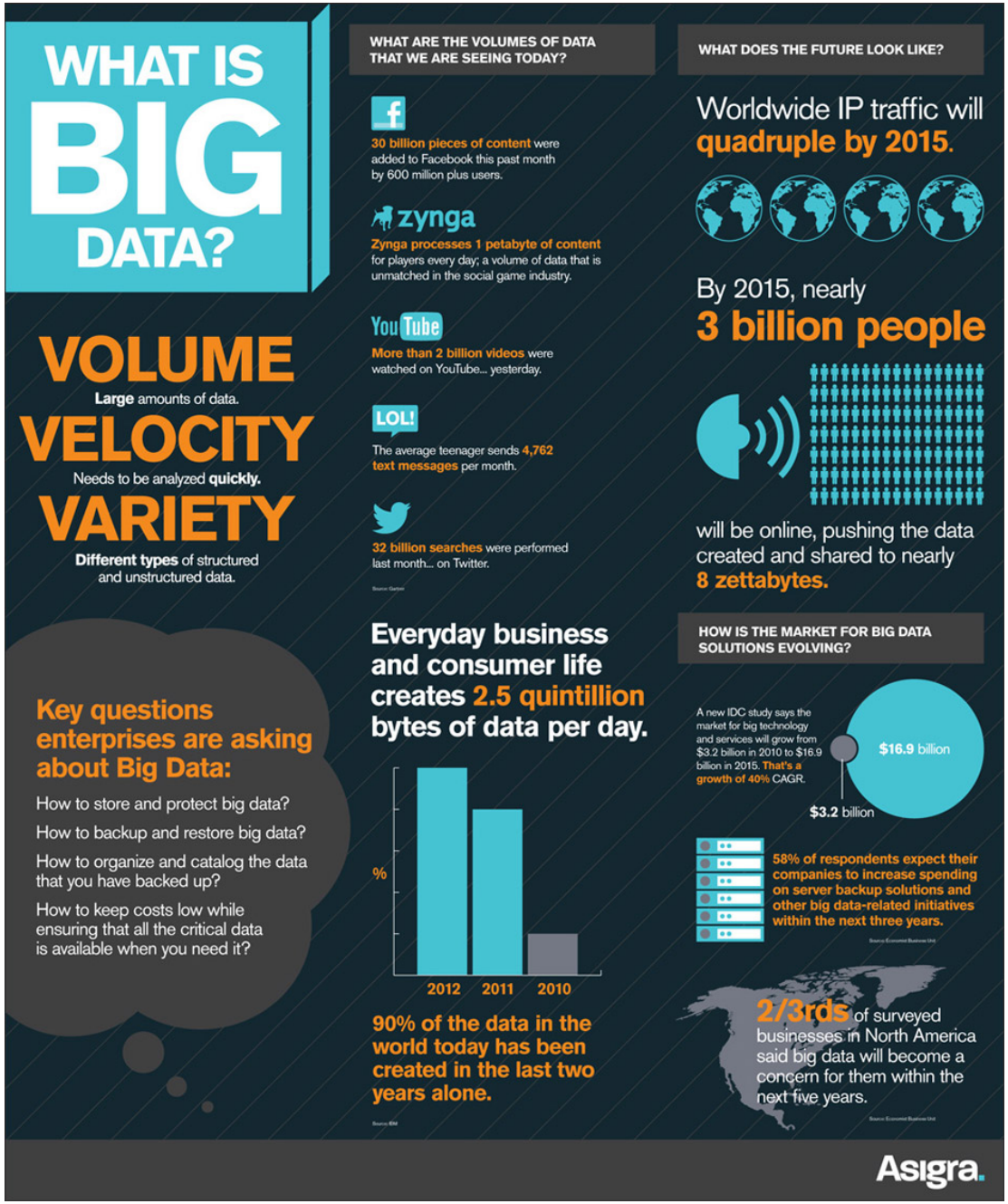
Lecture 1: Introduction

Prof. Dr. Ulrich Matter

25/02/2021

Big Data?


WHAT IS
BIG
DATA?



Posted on flickr by BBVAtech in 2012, by Asigra [CC BY 2.0]

Expert Survey (UC Berkeley, 2014)

- Ask 40 experts to define **“big data”**…

Expert Survey (UC Berkeley, 2014)

- Ask 40 experts to define **“big data”**…
- … get 40 different definitions :)

Expert Survey (UC Berkeley, 2014)

Image by Jennifer Dutcher, datascience@berkeley, source: https://datascience.berkeley.edu/what-is-big-data/

Expert Survey: Example 1

"Big Data is the result of **collecting information at its most granular level** — it's what you get when you instrument a system and keep all of the data that your instrumentation is able to gather."

Jon Bruner (Editor-at-Large, O'Reilly Media)

Expert Survey: Example 2

"Big data is data that contains enough observations to **demand unusual handling because of its sheer size**, though what is unusual changes over time and varies from one discipline to another."

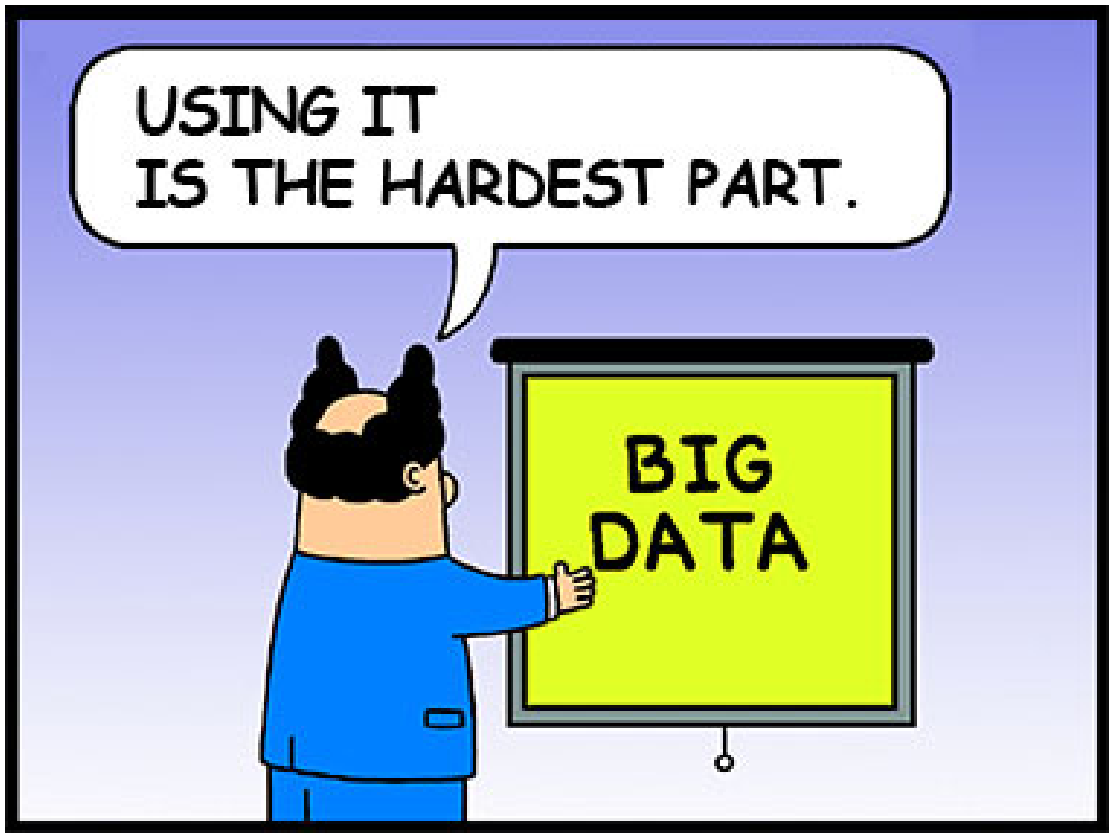
Annette Greiner
(Lecturer, UC Berkeley School of Information)

Expert Survey: Example 3

"[…] 'big data' will ultimately describe any dataset large enough to necessitate **high-level programming skill** and **statistically defensible methodologies** in order to transform the data asset into something of value."

Reid Bryant
(Data Scientist, Brooks Bell)

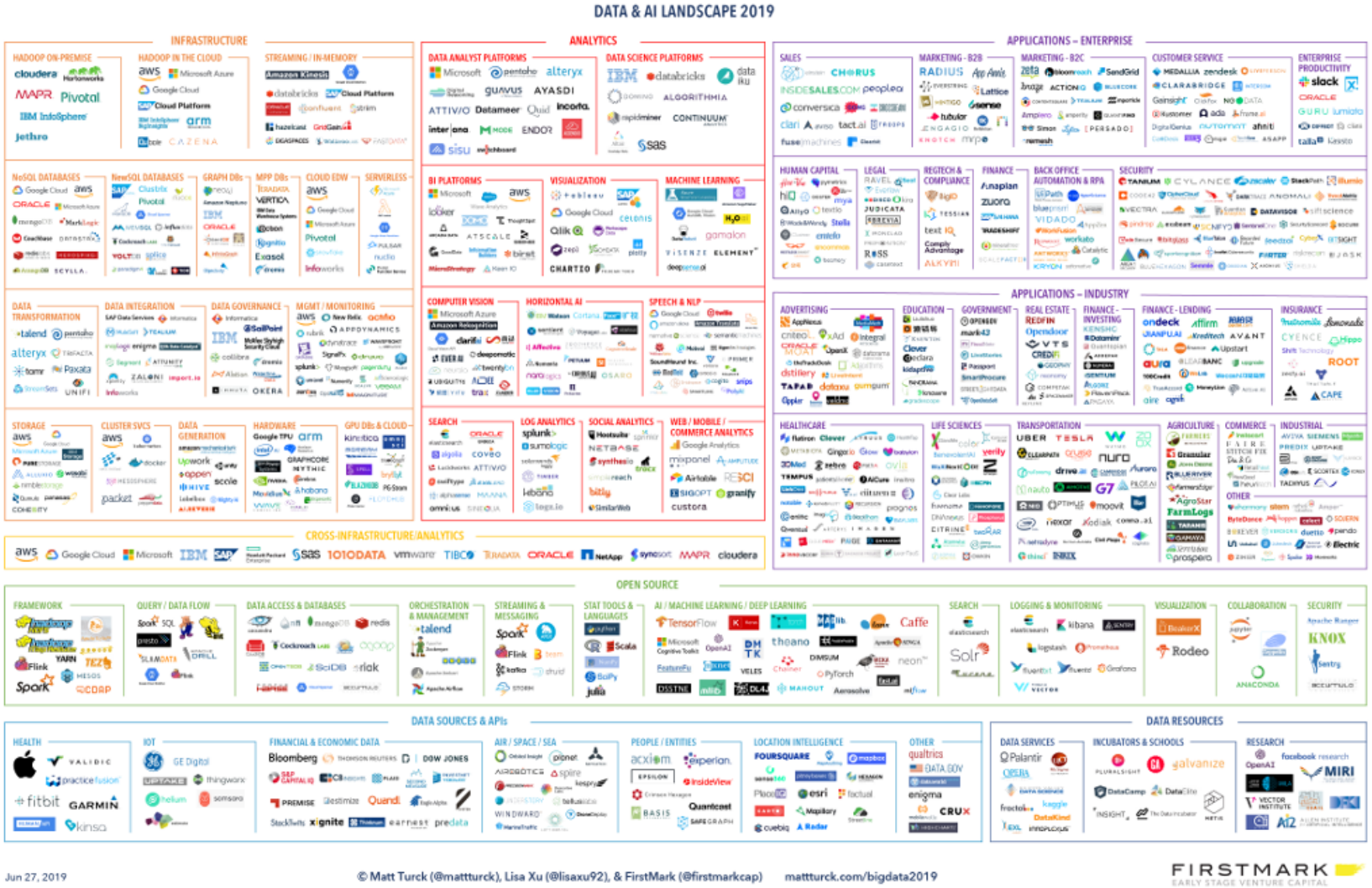
Conclusion



Conclusion

- Large amounts of data
- Various types/formats of data
- Unusual sources
- Speed of data flow/stream
- Use programming and statistics (in a broad sense) to extract value

‘Learn Big Data’?



‘Big Data Landscape (2019)’, source: http://mattturck.com

Domains Affected

- How to design/set-up the machinery to handle large amounts of data? (Hardware focuse, data engineering)
- How to use the existing machinery most efficiently for large amounts of data?
- How to approach the analysis of large amounts of data with econometrics?

Focus in This Course

- How to design/set-up the machinery to handle large amounts of data? (Hardware focuse, data engineering)
- **How to use the existing machinery most efficiently for large amounts of data?**
- **How to approach the analysis of large amounts of data with econometrics?**

Focus in This Course

- How to design/set-up the machinery to handle large amounts of data? (Hardware focuse, data engineering)
- **How to use the existing machinery most efficiently for large amounts of data?**
- **How to approach the analysis of large amounts of data with econometrics?**
  1. Compute ‘usual’ statistics based on large dataset (many observations).

Focus in This Course

- How to design/set-up the machinery to handle large amounts of data? (Hardware focuse, data engineering)
- **How to use the existing machinery most efficiently for large amounts of data?**
- **How to approach the analysis of large amounts of data with econometrics?**
  1. Compute ‘usual’ statistics based on large dataset (many observations).
  2. Practical handling of large data sets for applied econometrics (gathering, storage, preparation, etc.)

Big Data in Scientific Research

Big Data in the Sciences

- Mother nature always has provided the data, but…
  - … instruments have gotten more precise
  - … new measurement methods have been developed
- Prominent examples: Astronomy, Genomics/Bioinformatics

Photo by Joe Parks, (CC BY-NC 2.0) source: https://flic.kr/p/e2umhv

Big Data in the Sciences

Astronomy: SKA Radio Telescope

Image by SKA Organisation, source: https://www.skatelescope.org/multimedia/image

Big Data in the Sciences

Astronomy: SKA Radio Telescope

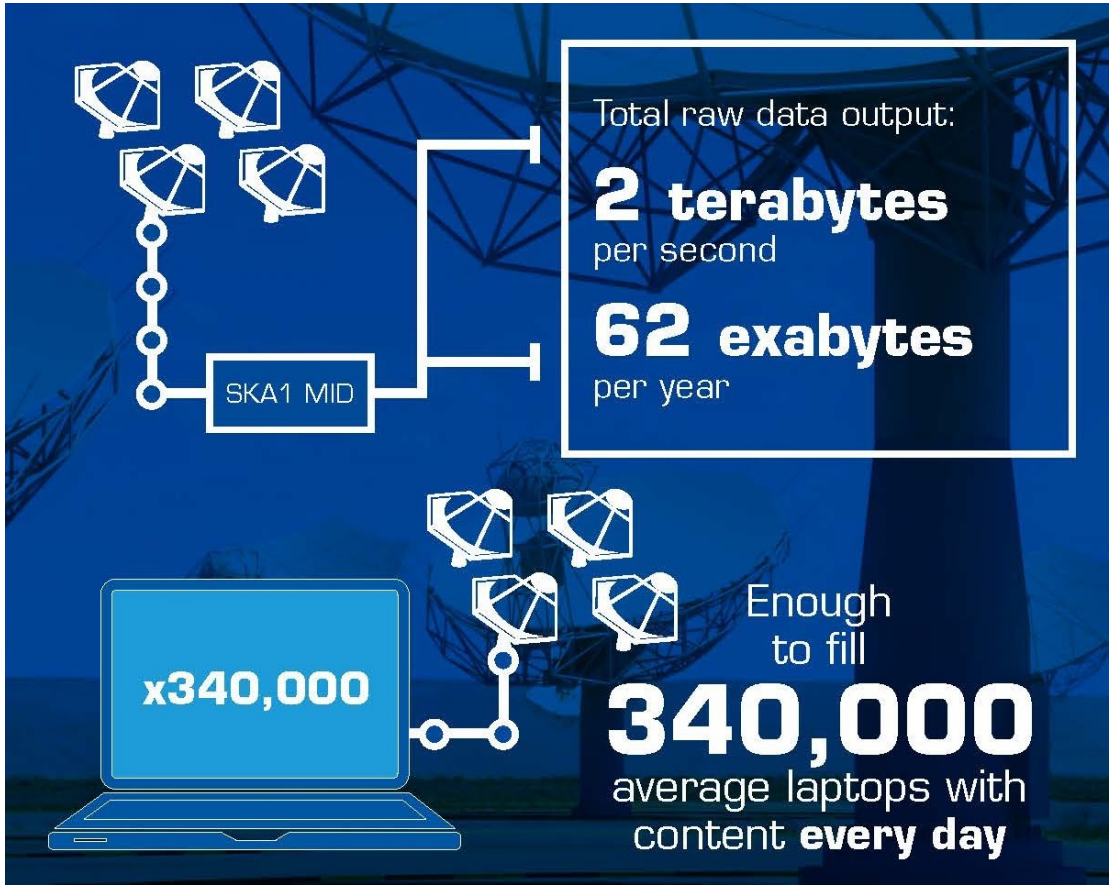
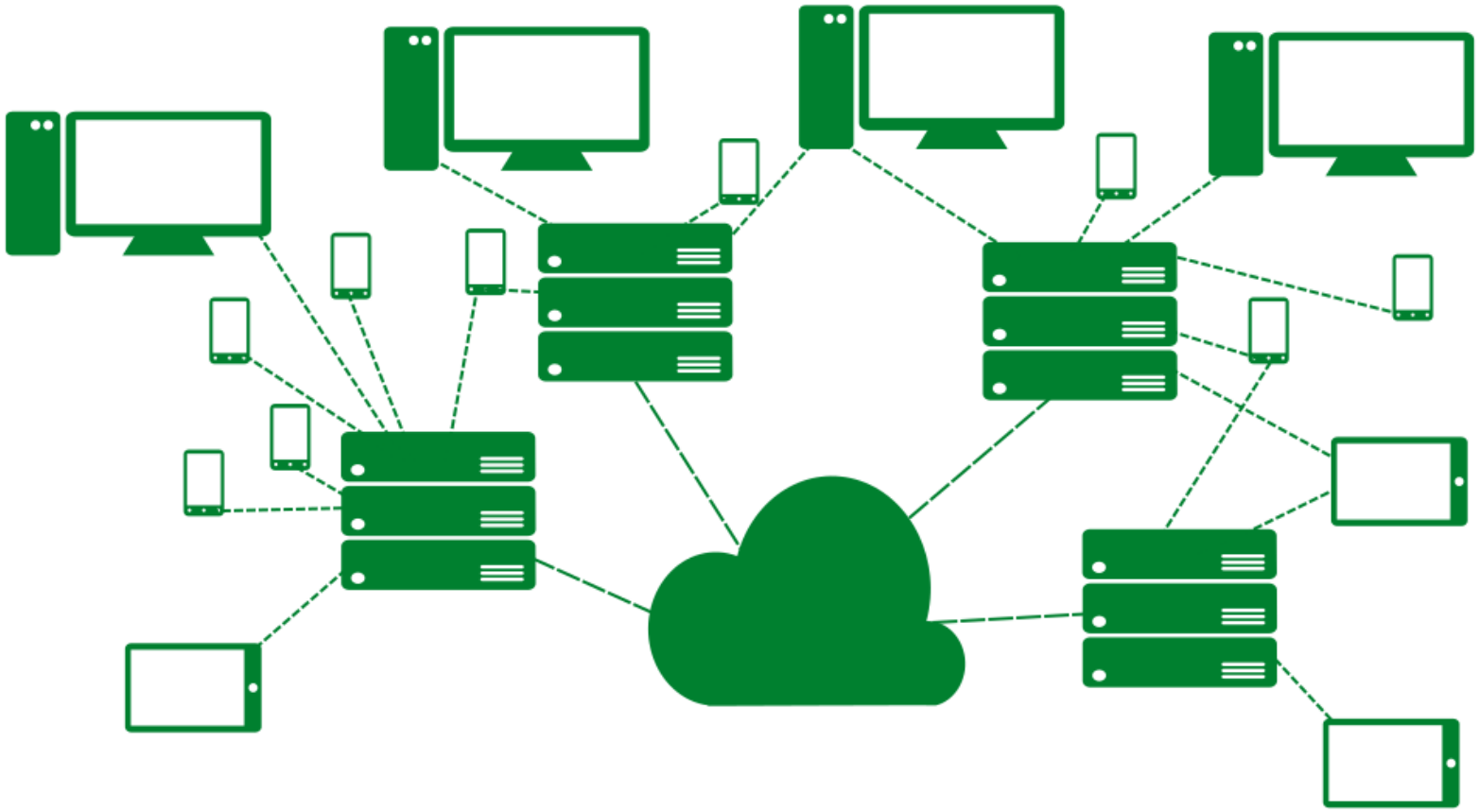


Image by SKY Organisation, source: https://www.skatelescope.org/multimedia/image

Big Data in the Social Sciences



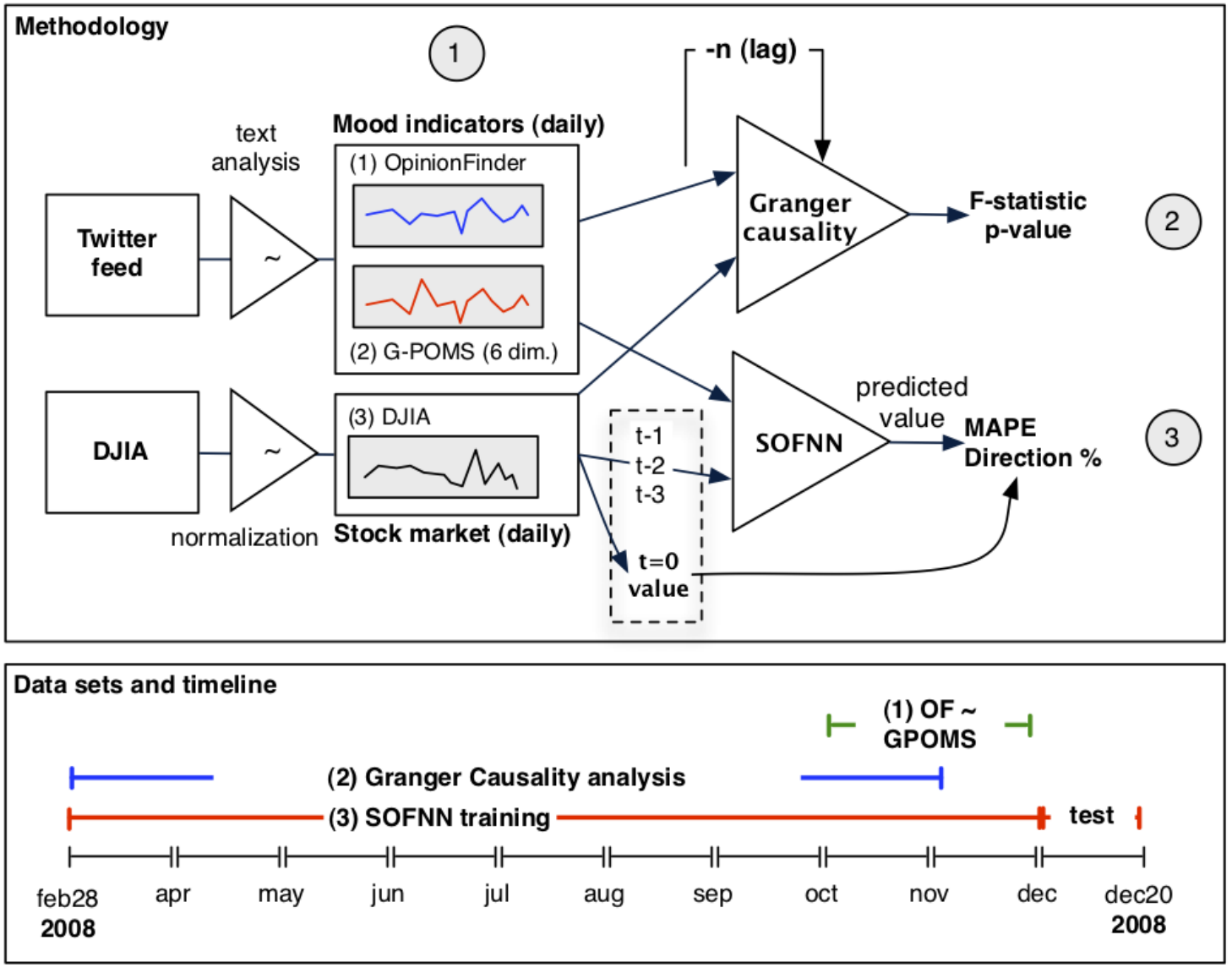
Big Data in the Social Sciences

- **Hardware**: Diffusion of the Internet and mobile-phone networks.
- **Software**: Web 2.0 Technologies (APIs, JSON, Programmable Web, etc.).

Big Data in the Social Sciences

- **Hardware**: Diffusion of the Internet and mobile-phone networks.
- **Software**: Web 2.0 Technologies (APIs, JSON, Programmable Web, etc.).
  - Backbone of social media and many prominent web services (e.g., Google Maps).
  - Data integration across platforms and services.
  - Exchange of data between/across applications.

Big Data in the Social Sciences/Economics



Source: Bollen, Mao, and Zeng (2011)

Big Data in the Social Sciences/Economics

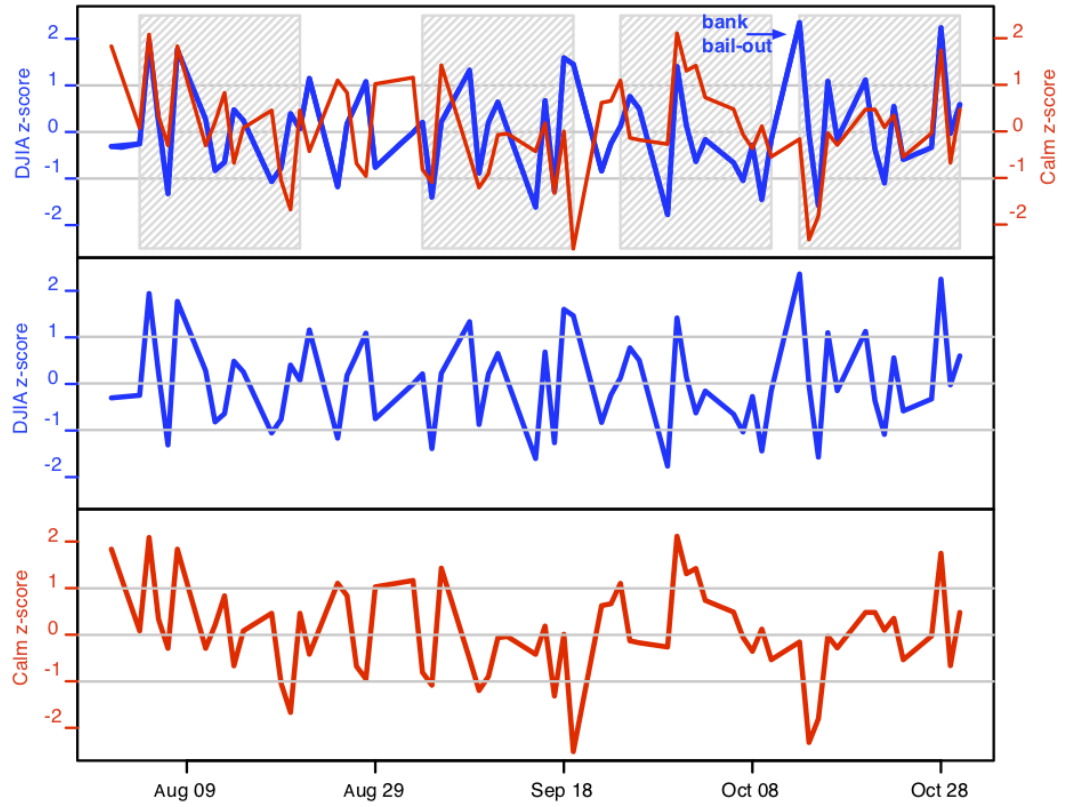
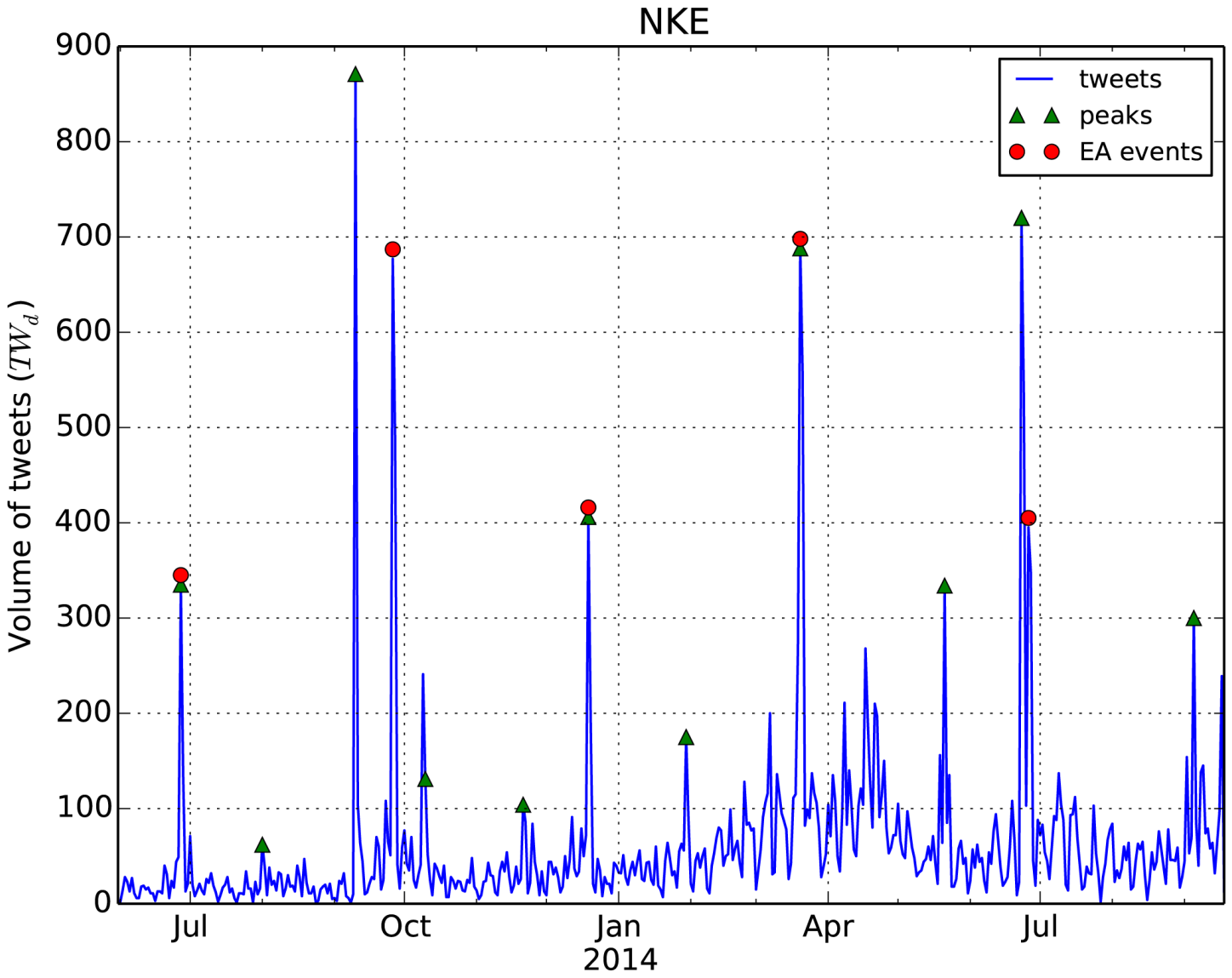


Fig. 3. A panel of three graphs. The top graph shows the overlap of the day-to-day difference of DJIA values (blue: $\mathbb{Z}_{D_t}$) with the GPOMS' Calm time series (red: $\mathbb{Z}_{X_t}$) that has been lagged by 3 days. Where the two graphs overlap the Calm time series predict changes in the DJIA closing values that occur 3 days later. Areas of significant congruence are marked by gray areas. The middle and bottom graphs show the separate DJIA and GPOMS' Calm time series.

Source: Bollen, Mao, and Zeng (2011)

Big Data in the Social Sciences/Economics



Source: Ranco (2015)

Big Data in the Social Sciences/Economics

- Often tied to web applications and digitization of economic and political processes.

Big Data in the Social Sciences/Economics

- Often tied to web applications and digitization of economic and political processes.
- **Volume** of data is substantial (but usually smaller than in the natural sciences).

Big Data in the Social Sciences/Economics

- Often tied to web applications and digitization of economic and political processes.
- **Volume** of data is substantial (but usually smaller than in natural sciences)
- **Variety** and **variability** often more challenging than in natural sciences.
  - Various sources
  - Data generation/sensors are independent from research endeavor.

Big Data in the Social Sciences/Economics

- Often tied to web applications and digitization of economic and political processes.
- **Volume** of data is substantial (but usually smaller than in natural sciences)
- **Variety** and **variability** often more challenging than in natural sciences.
  - Various sources
  - Data generation/sensors are independent from research endeavor.
- Questions/problems often similar to applied research in the industry.
  - Key difference: usually no streaming applications (**velocity** not that much of an an issue).

This Course

Three Parts

1. Big Data: Basic Concepts
2. Local Big Data Analytics
3. Advanced Topics

Objectives

- Understand the **concept of Big Data** in the context of economic research.
- Understand the **technical challenges** of Big Data Analytics and how to practically deal with them.
- Know how to **apply** the relevant R packages and programming practices to effectively and efficiently handle large data sets.

Schedule

Time frame	Room	Course number	Event
8:15 AM - 10:00 AM 25/02/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 04/03/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 11/03/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 18/03/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 25/03/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 01/04/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 22/04/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 29/04/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 06/05/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 20/05/2021	OVL	8,272,1.00	**Big Data Analytics**
8:15 AM - 10:00 AM 27/05/2021	OVL	8,272,1.00	**Big Data Analytics**

Schedule

1. **Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.**
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. **Computation and Memory in Applied Econometrics.**
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. **Computation and Memory in Applied Econometrics II.**
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. **Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.**
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. **Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.**
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. **Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al. (2015); Schwabish (2014).**
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. **Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.**
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. **Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.**
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. **Applied Econometrics with Spark; Machine Learning and GPUs.**
10. Project Presentations.
11. Project Presentations; Q&A.

Schedule

1. Introduction: Big Data, Data Economy. Walkowiak (2016): Chapter 1.
2. Computation and Memory in Applied Econometrics.
3. Computation and Memory in Applied Econometrics II.
4. Advanced R Programming. Wickham (2019): Chapters 2, 3, 17,23, 24.
5. Import, Cleaning and Transformation of Big Data. Walkowiak (2016): Chapter 3: p. 74-118.
6. Aggregation and Visualization. Walkowiak (2016): Chapter 3: p. 118-127; Wickham et al.(2015); Schwabish (2014).
7. Data Storage, Databases Interaction with R. Walkowiak (2016): Chapter 5.
8. Cloud Computing: Introduction/Overview, Distributed Systems, Walkowiak (2016): Chapter 4.
9. Applied Econometrics with Spark; Machine Learning and GPUs.
10. **Project Presentations.**
11. **Project Presentations; Q&A.**

Examination: Part I

- Decentral - Group examination ‘paper’ (all given the same grades) (60%).
- Group size: 3 (or 2) students.
- Take-home exercises: Application of basic concepts in R when working with big data. Conceptual questions related to the application.

Hand in on June 14 2021, 16:00
More details next week

Examination: Part II

- Decentral
- Group examination: presentation + code (all given the same grades) (40%)
- Big data analytics group projects: Own approach/strategy, implemented in R, presentation of results in class.

20 May 2021, 08:15-10:00
27 May 2021, 08:15-10:00
More details next week

Approach


R

Prerequisites?

- Basic R programming skills.
- Build on concepts taught in Data Analytics I (and more basic econometrics courses).
    - Brief review of concepts, but no additional introduction.

R used in two ways

- A tool to analize problems posed by large datasets.
  - For example, memory usage (in R).
  - (Idea behind 'advanced R programming part)
- A practical tool for Big Data Analytics.

Example

Preparations

```
# read dataset into R
economics <- read.csv("../data/economics.csv")
# have a look at the data
head(economics, 2)
```

```
##         date   pce    pop psavert uempmed unemploy
## 1 1967-07-01 507.4 198712    12.5     4.5     2944
## 2 1967-08-01 510.5 198911    12.5     4.7     2945
```

```
# create a 'large' dataset out of this
for (i in 1:3) {
     economics <- rbind(economics, economics)
}
dim(economics)
```

```
## [1] 4592    6
```

Example

Compute the real personal consumption expenditures (`pce`): Divide each value of `pce` by the deflator `1.05`.

```
# Naïve approach (ignorant of R)
deflator <- 1.05 # define deflator
# iterate through each observation
pce_real <- c()
n_obs <- length(economics$pce)
for (i in 1:n_obs) {
  pce_real <- c(pce_real, economics$pce[i]/deflator)
}

# look at the result
head(pce_real, 2)
```

```
## [1] 483.2381 486.1905
```

Example

How long does it take?

```
# Naïve approach (ignorant of R)
deflator <- 1.05 # define deflator
# iterate through each observation
pce_real <- list()
n_obs <- length(economics$pce)
time_elapsed <-
     system.time(
         for (i in 1:n_obs) {
              pce_real <- c(pce_real, economics$pce[i]/deflator)
})

time_elapsed
```

```
##    user  system elapsed
##   0.082   0.004   0.086
```

Example

Assuming a linear time algorithm ($O(n)$), we need that much time for one additional row of data:

```
time_per_row <- time_elapsed[3]/n_obs
time_per_row
```

```
##      elapsed
## 1.872822e-05
```

Example

If we deal with big data, say 100 million rows, that is

```
# in seconds
(time_per_row*100^4)
```

```
##  elapsed
## 1872.822
```

```
# in minutes
(time_per_row*100^4)/60
```

```
## elapsed
## 31.2137
```

```
# in hours
(time_per_row*100^4)/60^2
```

```
##   elapsed
## 0.5202284
```

Example

What happens in the background?

- Evaluation/computation
- Memory allocation/deallocation

Example

Can we improve this?

```
# Improve memory allocation (still somewhat ignorant of R)
deflator <- 1.05 # define deflator
n_obs <- length(economics$pce)
pce_real <- list()
# allocate memory beforehand
# tell R how long the list will be
length(pce_real) <- n_obs
```

Example

Can we improve this?

```
# Improve memory allocation (still somewhat ignorant of R)
deflator <- 1.05 # define deflator
n_obs <- length(economics$pce)
pce_real <- list()
# allocate memory beforehand
# tell R how long the list will be
length(pce_real) <- n_obs
# iterate through each observation
time_elapsed <-
     system.time(
         for (i in 1:n_obs) {
              pce_real[[i]] <- economics$pce[i]/deflator
})

time_elapsed
```

```
##    user  system elapsed
##   0.005   0.000   0.004
```

Example

Any improvements?

```
time_per_row <- time_elapsed[3]/n_obs
time_per_row
```

```
##      elapsed
## 8.710801e-07
```

Example

```
# in seconds
(time_per_row*100^4)
```

```
##  elapsed
## 87.10801
```

```
# in minutes
(time_per_row*100^4)/60
```

```
## elapsed
##  1.4518
```

```
# in hours
(time_per_row*100^4)/60^2
```

```
##    elapsed
## 0.02419667
```

This looks much better, but we can do even better…

Example

Can we further improve this?

```
# Do it 'the R way'
deflator <- 1.05 # define deflator
# Exploit R's vectorization!
time_elapsed <-
     system.time(
     pce_real <- economics$pce/deflator
          )
# same result
head(pce_real, 2)
```

```
## [1] 483.2381 486.1905
```

```
# but much faster!
time_elapsed
```

```
##    user  system elapsed
##       0       0       0
```

```
time_per_row <- time_elapsed[3]/n_obs
```

Example

In fact, `system.time()` is not precise enough to capture the time elapsed…

```
# in seconds
(time_per_row*100^4)
```

```
## elapsed
##       0
```

```
# in minutes
(time_per_row*100^4)/60
```

```
## elapsed
##       0
```

```
# in hours
(time_per_row*100^4)/60^2
```

```
## elapsed
##       0
```

Example

Use `microbenchmark::microbenchmark()` to measure the elapsed time in microseconds (millionth of a second)

```
library(microbenchmark)
# measure elapsed time in microseconds (avg.)
time_elapsed <-
  summary(microbenchmark(pce_real <- economics$pce/deflator))$mean

# per row (in sec)
time_per_row <- (time_elapsed/n_obs)/10^6
```

Example

Improvement with vectorization (again, assuming 100 million rows)

```
# in seconds
(time_per_row*100^4)
```

```
## [1] 0.1266548
```

```
# in minutes
(time_per_row*100^4)/60
```

```
## [1] 0.002110914
```

```
# in hours
(time_per_row*100^4)/60^2
```

```
## [1] 3.51819e-05
```

What do we learn from this?

1. How R allocates and deallocates memory can have a substantial effect on computation time.
   - (Particularly, if we deal with a large dataset!)
2. In what way the computation is implemented can matter a lot for the time elapsed.
   - (For example, loops vs. vectorization/apply)

Course Resources

Literature

Books

Walkowiak, Simon (2016): Big Data Analytics with R. Birmingham, UK: Packt Publishing.

- Available here

Wickham, Hadley (2019): Advanced R. Second Edition, Boca Raton, FL: CRC Press

Literature

Journal Articles

Wickham, Hadley and Dianne Cook and Heike Hofmann (2015): Visualizing statistical models: Removing the blindfold.Statistical Analysis and Data Mining: The ASA Data Science Journal. 8(4):203-225.

Schwabish, Jonathan A. (2014): An Economist's Guide to Visualizing Data.Journal of Economic Perspectives. 28(1):209-234.

Lecture notes and slides will point to further reading...

Notes, Slides, Code, et al.

- umatter.github.io/courses
- github.com/umatter/BigData

Suggested Learning Procedure

- Clone/fork the course's GitHub-repository
- During class, use the Rmd-file of the slide-set as basis for your notes
- After class, enrich/merge/extend your notes with the lecture notes.

TODO (for next week!)

- Install R, RStudio
- Set up your own GitHub-account
- Get familiar with Git/GitHub

Q&A

- General questions about the course?
- Exchange students: additional information regarding prerequisites

References

Bollen, Johan, Huina Mao, and Xiaojun Zeng. 2011. “Twitter Mood Predicts the Stock Market.” **Journal of Computational Science** 2 (1): 1–8. https://doi.org/https://doi.org/10.1016/j.jocs.2010.12.007.

Ranco, Darko AND Caldarelli, Gabriele AND Aleksovski. 2015. “The Effects of Twitter Sentiment on Stock Price Returns.” **PLOS ONE** 10 (9): 1–21. https://doi.org/10.1371/journal.pone.0138441.