

Data Science for Business Analytics

Introduction

Haute Ecole de Gestion

Aleksandr Shemendyuk [aleksandr.shemendyuk@unil.ch]

Agenda

1 Introduction

2 Organization

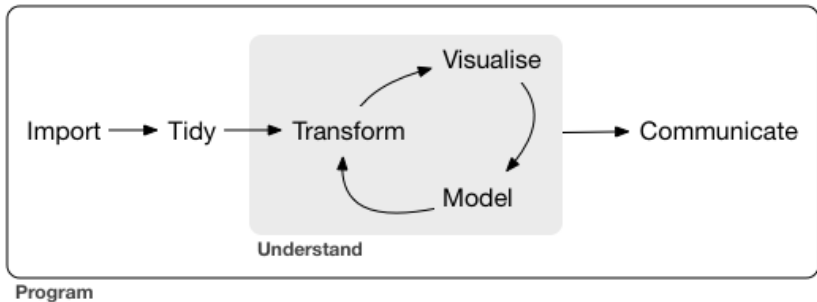
3 R

A little about me

- Born and raised in Geneva
- Education:
 - ▶ B.Sc. Physics (EPFL, '10)
 - ▶ M.Sc. Physics with minor in Financial Engineering (EPFL, '12)
 - ▶ Ph.D. Statistics (HEC Lausanne, '16)
- Then:
 - ▶ Worked a bit as a quant in finance in Zurich
 - ▶ Moved to New York
 - ▶ Became Professor in Statistics at Columbia University
 - ▶ Left academia to join Meta as a data scientist
 - ▶ Now in Geneva as a Professor of Business Analytics at HEG
- Hobbies:
 - ▶ Running
 - ▶ Spending time with my family
 - ▶ Watching bay area teams (go 49ers and Warriors!)

What you will learn

- Workflow of the modern Data Scientist



Statistical computing & data science

- What's the difference between data science and statistics?

“A data scientist is just a sexier word for statistician.”

— Nate Silver (outdated)

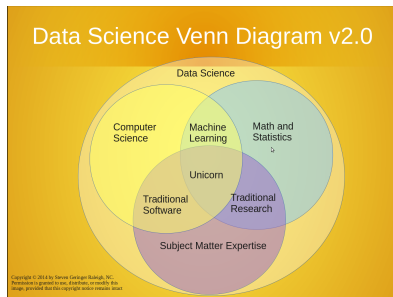
“A data scientist is a better computer scientist than a statistician and is a better statistician than a computer scientist.”

— Unknown (still accurate)

- What does a data scientist do?
 - ▶ There is not one correct answer.
 - ▶ Transform data into valuable information!
 - ▶ A data scientist spends a significant portion of time processing data and less time modeling data.

What is Data Science?

- **Wikipedia:** “the extraction of knowledge from data”
- Precise definition a bit unclear and controversial...
- Practitioners “agree” on the components of data science:
 - ▶ database management
 - ▶ gathering and cleaning
 - ▶ exploratory analysis
 - ▶ predictive modelling
 - ▶ data summary and visualization



Applications



Some of the hiring partners of *The Data Incubator*

- E-marketing
- Recommender systems
- Sport analytics
- Biotechnology
- Image or speech recognition
- Fraud and risk detection
- Social media
- redbfit scoring
- E-commerce
- Government analysis
- Gaming
- Price comparisons
- Airline routes planing
- Delivery logistics

Agenda

1 Introduction

2 Organization

3 R

Course Description

- Lectures (10:15-12:00):
 - ▶ Focus on introducing the concepts
 - ▶ Details sometimes left for you to read up on
- Exercise sessions (12:15-14:00):
 - ▶ Work on assignments and project
- Instructor: **Professor Alex** (Aleksandr Shemendyuk).
- Teaching assistants
 - ▶ Ilia Azizi
 - ▶ Léo Wenger
 - ▶ Elwin Freudiger

Grading

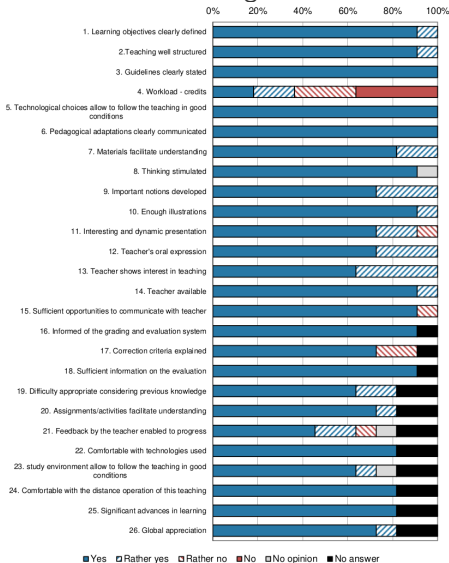
- Final grade based on:
 - ▶ Multiple choice questions (20%)
 - ▶ A project (80%)
- Multiple choice questions:
 - ▶ 15 minutes at the beginning of each of lectures 2, 3, 4, 5, 6
 - ▶ Worst grade **might** be dropped
- Project:
 - ▶ Project proposal (5%)
 - ▶ Project update (5%)
 - ▶ Video presentation (20%)
 - ▶ Final report (50%)
- For the project:
 - ▶ Groups of 2 or 3 members
 - ▶ More on that later
- Grades based on academic performance only!

Milestones

Date	Assignment
10/02	Quiz 1
10/09	Quiz 2
10/15	Project proposal
10/16	Quiz 3
10/23	Quiz 4
10/30	Quiz 5
11/12	Project update
12/15	Video presentation
12/22	Final Report

Notice of caution!!!

- Course evaluations last time I taught:



Tentative outline

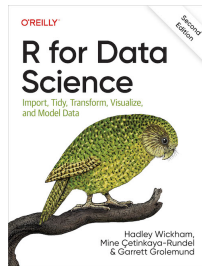
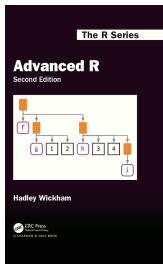
Date	Topic	Reading
09/25	Introduction	
10/02	Data Structures and Subsetting	ADVR 3+4
10/09	Control Flows and Functions	ADVR 5, 6, 9, 11
10/16	Data Wrangling	R4DS 4, 6, 8, 13-20
10/23	Visualization	R4DS 2, 10-12
10/30	R Projects and Quarto	R4DS 3, 5, 7, 9, 29, 30
11/06	Project Coaching	
11/13	Data Wrangling	R4DS 4, 6, 8, 13-20
11/20	Visualization	R4DS 2, 10-12
11/27	Project coaching	
12/04	Presentations/Dashboards/Interactivity	R4DS 29+30, htmlwidgets
12/11	Projects Coaching	
12/18	Projects Presentations	

(numbers in the third column are book chapters)

Course website

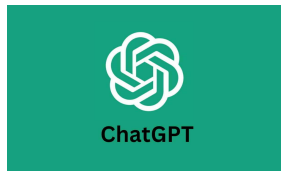
- Course website:
 - ▶ <https://moodle.unil.ch/course/view.php?id=29021>
 - ▶ Registration key: **dsfba_2023**
 - ▶ **A discussion forum**
 - ▶ PDFs of the lectures
 - ▶ Additional resources

Additional resources



- Books:
 - ▶ Advanced R
 - ▶ R for data science
- Additionally:
 - ▶ Rstudio cheat sheets
 - ▶ The CRAN website

Best place to look for answers?



... and moodle's forum!

Agenda

1 Introduction

2 Organization

3 R

S and R

- S
 - ▶ A statistical programming language
 - ▶ First appeared in 1976
 - ▶ Developed by John Chambers and (in earlier versions) Rick Becker and Allan Wilks of Bell Labs
 - ▶ John Chambers, *[the aim is] to turn ideas into software, quickly and faithfully*
- R
 - ▶ Modern implementation of S
 - ▶ First appeared in 1993
 - ▶ Created by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand
 - ▶ Currently developed by the *R Development Core Team*

Some “technical” details about R

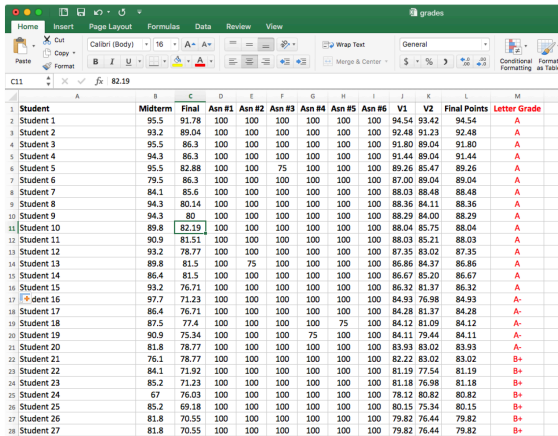
- Technical features:
 - ▶ **Available for Windows, macOS, and Linux**
 - ▶ Source code written primarily in C, Fortran, and R
 - ▶ Multi-paradigm: object-oriented, functional, procedural
 - ▶ Dynamically typed
 - ▶ Scripting language (interpreted)
 - ▶ **Wide variety of statistical and graphical techniques**
 - ▶ **Easily extensible through functions and packages**
 - ▶ **Read/write from/to various data sources**

What about Excel?



source: fantasyfootballanalytics.net

Excel is great for certain things...



Student	Midterm	Final	Asn #1	Asn #2	Asn #3	Asn #4	Asn #5	Asn #6	V1	V2	Final Points	Letter Grade
Student 1	95.5	91.78	100	100	100	100	100	100	94.54	93.42	94.54	A
Student 2	93.2	89.04	100	100	100	100	100	100	92.48	91.23	92.48	A
Student 3	95.5	86.3	100	100	100	100	100	100	91.80	89.04	91.80	A
Student 4	94.3	86.3	100	100	100	100	100	100	91.44	89.04	91.44	A
Student 5	95.5	82.88	100	100	75	100	100	100	89.26	85.47	89.26	A
Student 6	79.5	86.3	100	100	100	100	100	100	87.00	89.04	89.04	A
Student 7	84.1	85.6	100	100	100	100	100	100	88.03	88.48	88.48	A
Student 8	94.3	80.14	100	100	100	100	100	100	88.36	84.11	88.36	A
Student 9	94.3	80	100	100	100	100	100	100	88.29	84.00	88.29	A
Student 10	89.8	82.19	100	100	100	100	100	100	88.04	85.75	88.04	A
Student 11	90.9	81.51	100	100	100	100	100	100	88.03	85.21	88.03	A
Student 12	93.2	78.77	100	100	100	100	100	100	87.35	83.02	87.35	A
Student 13	89.8	81.5	100	75	100	100	100	100	86.86	84.37	86.86	A
Student 14	86.4	81.5	100	100	100	100	100	100	86.67	85.20	86.67	A
Student 15	93.2	76.71	100	100	100	100	100	100	86.32	81.37	86.32	A
Student 16	97.7	71.23	100	100	100	100	100	100	84.93	76.98	84.93	A-
Student 17	86.4	76.71	100	100	100	100	100	100	84.28	81.37	84.28	A-
Student 18	87.5	77.4	100	100	100	100	75	100	84.12	81.09	84.12	A-
Student 19	90.9	75.34	100	100	100	100	75	100	84.11	79.44	84.11	A-
Student 20	81.8	78.77	100	100	100	100	100	100	83.93	83.02	83.93	A-
Student 21	76.1	78.77	100	100	100	100	100	100	82.22	83.02	83.02	B+
Student 22	84.1	71.92	100	100	100	100	100	100	81.19	77.54	81.19	B+
Student 23	85.2	71.23	100	100	100	100	100	100	81.18	76.98	81.18	B+
Student 24	67	76.03	100	100	100	100	100	100	78.12	80.82	80.82	B+
Student 25	85.2	69.18	100	100	100	100	100	100	80.15	75.34	80.15	B+
Student 26	81.8	70.55	100	100	100	100	100	100	79.82	76.44	79.82	B+
Student 27	81.8	70.55	100	100	100	100	100	100	79.82	76.44	79.82	B+

source: github.com/jdwilson4

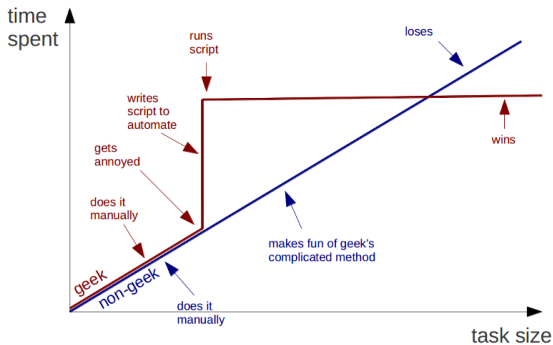
...but not everything

R's advantages:

- **Easier automation**
- **Better reproducibility**
- Faster computation
- Supports larger data sets
- Reads any type of data
- More powerful data manipulation capabilities
- Easier project organization
- Easier to find and fix errors
- Free & open source
- Advanced statistics capabilities
- State-of-the-art graphics
- Runs on many platforms
- Anyone can contribute packages to improve its functionality

Automation and reproducibility

Geeks and repetitive tasks



source: trendct.org

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2017-11-30, Kite-Eating Tree) [R-3.4.3.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

source: cran.r-project.org

RStudio

- An open-source integrated development environment (IDE)
- RStudio Desktop available for Windows, macOS, and Linux



RStudio

RStudio makes R easier to use. It includes a code editor, debugging & visualization tools.



Shiny

Shiny helps you make interactive web applications for visualizing data. Bring R data analysis to life.



R Packages

Our developers create popular packages to expand the features of R. Includes ggplot2, dplyr, R Markdown & more.

source: rstudio.com

Base R

- What is Base R?

“The package named base is in a way the core of R and contains the basic functions of the language, particularly, for reading and manipulating data.”

— *R for Beginners*, Emmanuel Paradis

- Base R includes all default code for performing common data manipulation and statistical tasks.
- You might recognize some Base R functions:
 - ▶ `mean()`, `median()`, `lm()`, `summary()`, `sort()`
 - ▶ `data.frame()`, `read.csv()`, `cbind()`, `grep()`, `regexpr()`
 - ▶ Many many more...
- If you don't recognize any Base R functions, don't worry!

The tidyverse

- Common criticisms of Base R:
 - ▶ The code doesn't flow as well as other languages.
 - ▶ Function names/arguments are often inconsistent/confusing.
 - ▶ Base R functions sometimes don't return type-stable objects.
 - ▶ Base R functions are not refined to run as fast as possible.
 - ▶ Other complaints exist...
- So what is the tidyverse? A collection of R packages
 - ▶ designed for data science,
 - ▶ sharing an underlying design philosophy, grammar, and data structures.
- Often perform the same tasks as Base R, but:
 - ▶ Relies on a **pipe** operator to help with the flow of the code.
 - ▶ More descriptive function names and consistent inputs.
 - ▶ Type-stable.
 - ▶ Often faster than common Base R functions.

Core tidyverse packages

- `ggplot2`: declarative graphics, based on The Grammar of Graphics.
- `dplyr`: grammar of data manipulation.
- `tidyr`: functions that help you get to tidy data.
- `readr`: reading in rectangular data.
- `purrr`: enhancing R's functional programming (FP).
- `tibble`: a modern `data.frame`.
- `stringr`: makes working with strings as easy as possible.
- `forcats`: useful tools for common problems with factors.

More on the tidyverse website!

Base R versus tidyverse

- Why ever use Base R?
 - ▶ Gets the job done!
 - ▶ To become an expert, you have to know Base R.
 - ▶ Some Base R functions are very common/useful, e.g., `mean()`.
- What should you learn first? Base R or tidyverse?
 - ▶ Some believe you should learn Base R first, others the tidyverse first.
 - ▶ Lately, more are shifting to tidyverse...

Install links

- R:
 - ▶ MacOS
 - ▶ Be mindful of the difference between Apple silicon (M1/M2) Macs and older Intel Macs
 - ▶ Windows
- RStudio Desktop:
 - ▶ MacOS 11+
 - ▶ Windows 10/11