

Introduction

02807 Computational Tools for Data Science

Today

- Introduction to the course
- Command line tools for working with data
- Version control (Git)

Teachers

- **Patrick Cording**
- Philip Bille
- Inge Li Gørtz
- Carsten Witt

Teaching assistants:

- Bastian Ellegård Grønager
- Sarah Alexandra Maria Van Dam

Me



PhD and postdoc

- Research in compressed data structures



Data engineer

- Data analysis and data pipelines



Data engineer

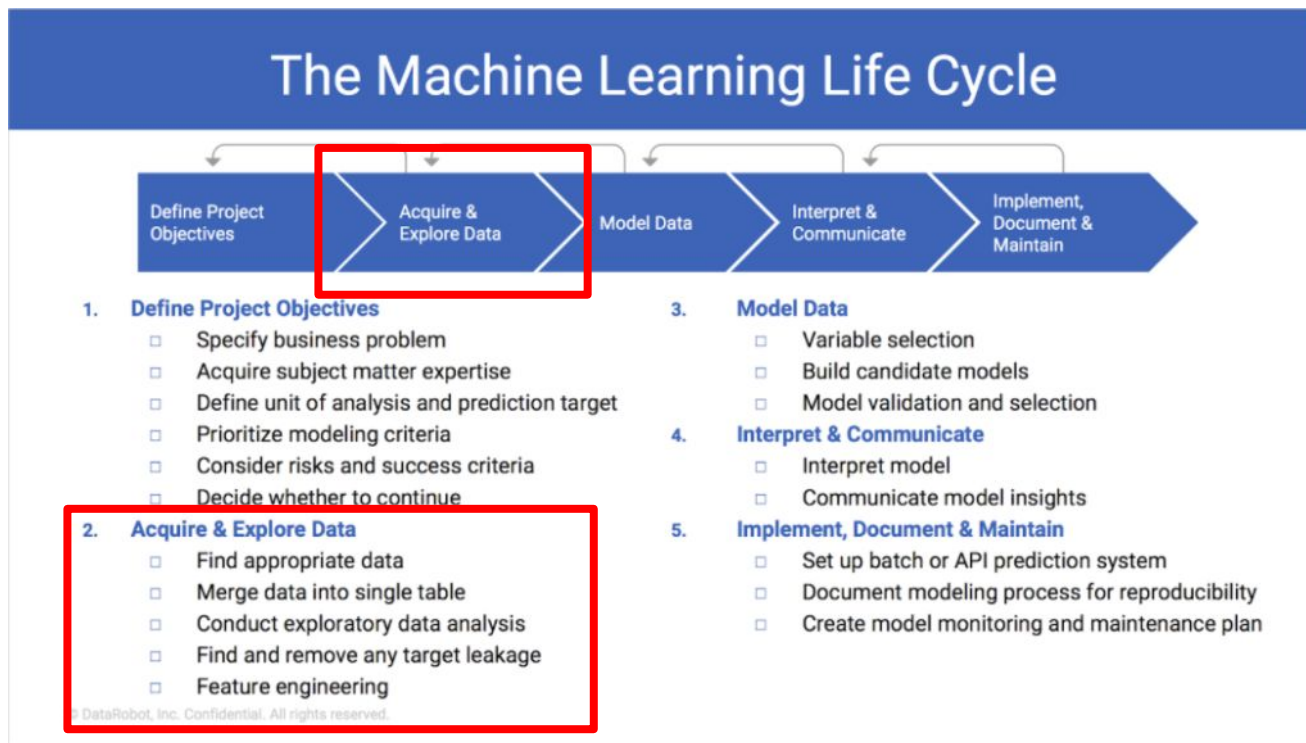
- Data pipelines for machine learning

You!

Survey: <http://tiny.cc/v594bz>

(link can be found on weekplan)

About this course



About this course

In this course you will learn about:

- **Tools and methods for working with data at scale.**

Course topics:

- Jupyter notebooks, Python for working with data, databases and SQL
- Streaming algorithms
- Apache Spark

Evaluation

- 3 projects
 - To be done in groups of at most 3 people
- 3 individual assignments
 - One or more exercises
 - Subject to the collaboration policy stated on the course website

Contacting me

- patrick.cording+**02807**@gmail.com
- No office hours

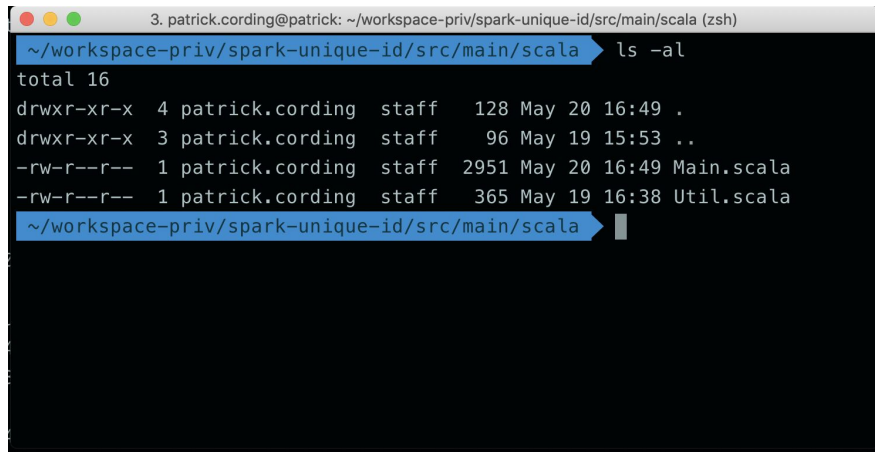
More practical info

- Not every week has a lecture. There is time allocated for project work.
- Last two lectures are guest lectures!

Command line tools

What is command line tools?

- A set of programs that you invoke from a terminal
- Each program does one thing, and it does it well



```
3. patrick.cording@patrick: ~/workspace-priv/spark-unique-id/src/main/scala (zsh)
~/workspace-priv/spark-unique-id/src/main/scala ➤ ls -al
total 16
drwxr-xr-x  4 patrick.cording  staff   128 May 20 16:49 .
drwxr-xr-x  3 patrick.cording  staff    96 May 19 15:53 ..
-rw-r--r--  1 patrick.cording  staff  2951 May 20 16:49 Main.scala
-rw-r--r--  1 patrick.cording  staff   365 May 19 16:38 Util.scala
~/workspace-priv/spark-unique-id/src/main/scala ➤
```

Why command line tools?

- A set of tools that have been developed over more than 40 years
- Always at hand
- REPL style workflow
- Fast
- Programmable
- Same across operating systems
- Great for working with data!

Version control

Why version control?

- Track changes of documents (source code)
- Work on several version of the same document
- Collaborate

Terminology

Repository: A collection of files under version control

Commit: To write your changes to a repository, generates a version

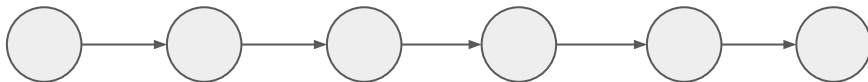
Branch: A sequence of commits

Checkout (a branch): To switch to a branch

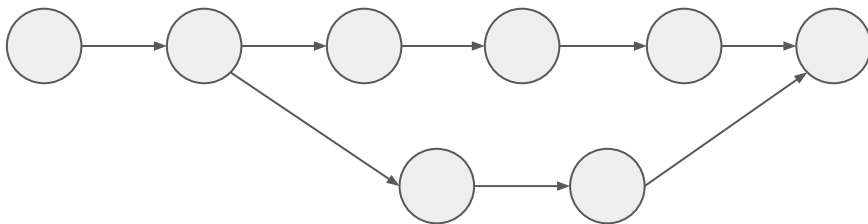
Merge: To write the changes on a branch to another branch

Version control basics

History



Collaboration



Version control with Git

- Git is by far the most widely used tool for version control
- Free remote repositories available on Github and DTU (GitLab)

Git on the command line

DEMO

Exercises

- Getting started with command line tools and Git
- Setting up Docker