What is Data Science?

Data Science Immersive

FLATIRON SCHOOL

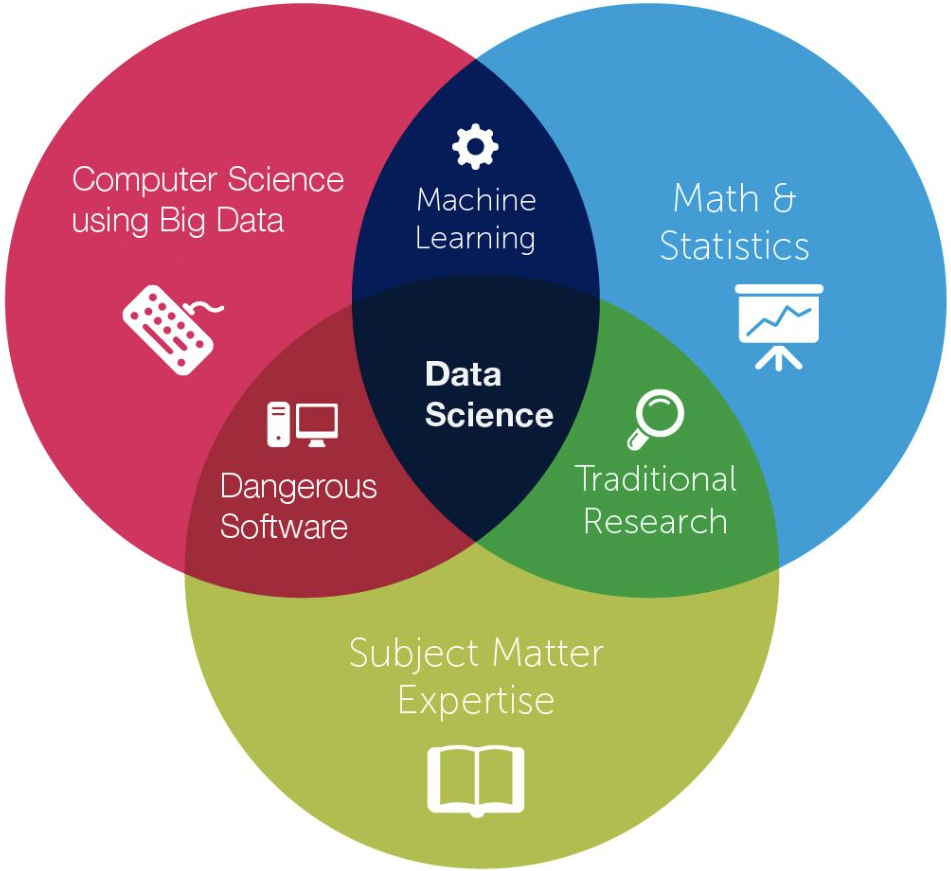
Overview of Day 1

- Introduction of Data Science
- Overview of Curriculum
- Setting up a data science environment

Introduction

- According to wikipedia, the definition of data science is “an **interdisciplinary** field that uses **scientific** methods, processes, algorithms and systems to extract **knowledge and insights from various forms**, both structured and unstructured”

What is data science?

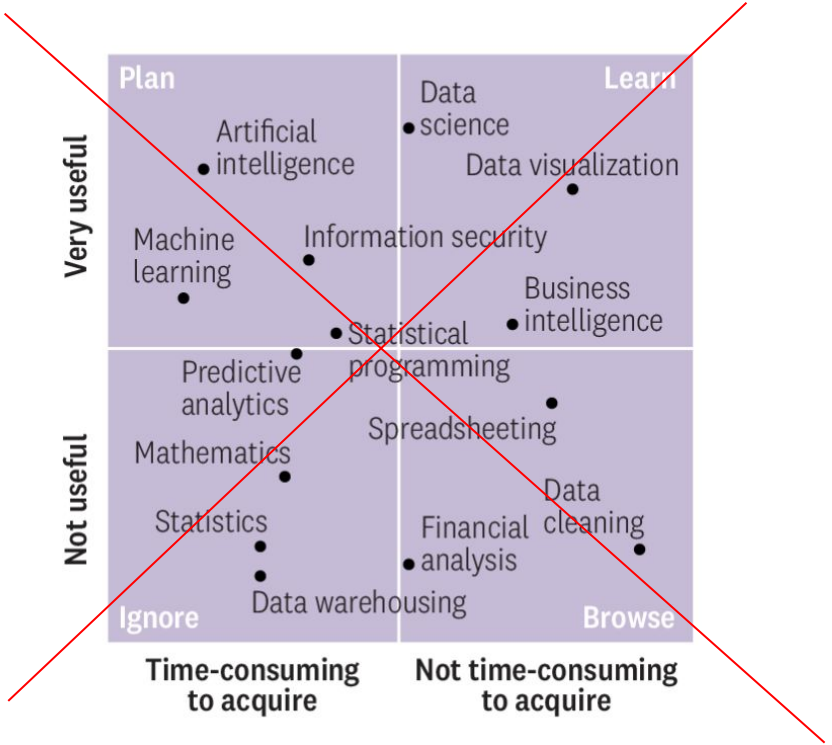


Introduction

“Data science is a "concept to unify **statistics**, **data analysis**, **machine learning** and their related methods" in order to "understand and analyze actual phenomena" with data. It employs techniques and theories drawn from many fields within the context of **mathematics**, **statistics**, **information science**, and **computer science**.”

What is Data Science?

- What should you learn in data science?



Ok, but what kind of questions can DS answer?

- How many minutes, on average, is an American person exposed to advertisement?
- Given the items you have purchased in the past, what items are you more likely to purchase in the future?
- Does the $5 coupon or 25% off result in more returning customers for your favorite store?
- Is a given image mountain or glacier?
- How should the government allocate educational resources?

Activity

- Talk to your neighbor, and either pick one of the questions above, or come up with your own question that can be solved with data science tools, and lay out your plans or thought processes on solving that

Data Science Process

- What is the actual process of doing data science?
- How much time do data scientists spend on each of these components in the process?

Activity

- Break into small groups, talk about the problem or question that you discussed earlier, and tell us the steps it would take to execute your solutions

Data Science Process

- Crisp-DM

Overview of Curriculum

- Learn.co as textbook
    - Readme's
    - Labs
- Lectures & presentations from instructor and TCF's
- Code challenge
- Blogging
- Projects & Science Fair
- Flatiron Students Present
- Feelings Fridays
- Other community events

Module Overview

- Module 1 - Fundamentals of Data Science

In this module, we will review and practice programming fundamentals, learn about probabilities and statistical distribution, as well as data acquisition. Topics such as probability density distribution, object oriented programming, and the concept of class variables will be introduced. We will also learn basic data visualization and manipulating tables using the Python numpy and pandas modules.

Module Overview

- Module 2 - Statistical testing and Data Analysis

In mod 2, we will learn everything you need to know about cleaning, and structuring your data, as well as statistical procedures for testing data. We will also learn statistical concepts such as hypothesis testing, z-test, and t-test which will allow you to draw meaning conclusions to your data. You will also learn all about linear regression and gradient descent, as well as all the assumptions underlying these algorithms.

Module Overview

- Module 3 - Machine Learning

In this unit we will learn all about the more complex machine learning algorithms such as decision tree, random forest, logistic regression. You will have a firm grasp on the distinction between supervised vs. unsupervised machine learning, as well as the ability to implement and interpret these algorithms on your own datasets.

Module Overview

- Module 4 - Big Data & Deep Learning

You will learn advanced topics in the data science world such as recommendation engines, neural networks, Markov Chain Monte Carlo simulation. You will be able to build a big data project using the techniques you learned from this chapter.

Setting Up a Data Science Environment

- Download Python 3.7 & Anaconda
- Slack
- Terminal 101
- Navigate Github & Git

Download Python 3.7

Anaconda 2018.12 For macOS Installer

Python 3.7 version *

Download

64-Bit Graphical Installer (652.7 MB)
64-Bit Command-Line Installer (557 MB)

Python 2.7 version *

Download

64-Bit Graphical Installer (640.7 MB)
64-Bit Command-Line Installer (547 MB)

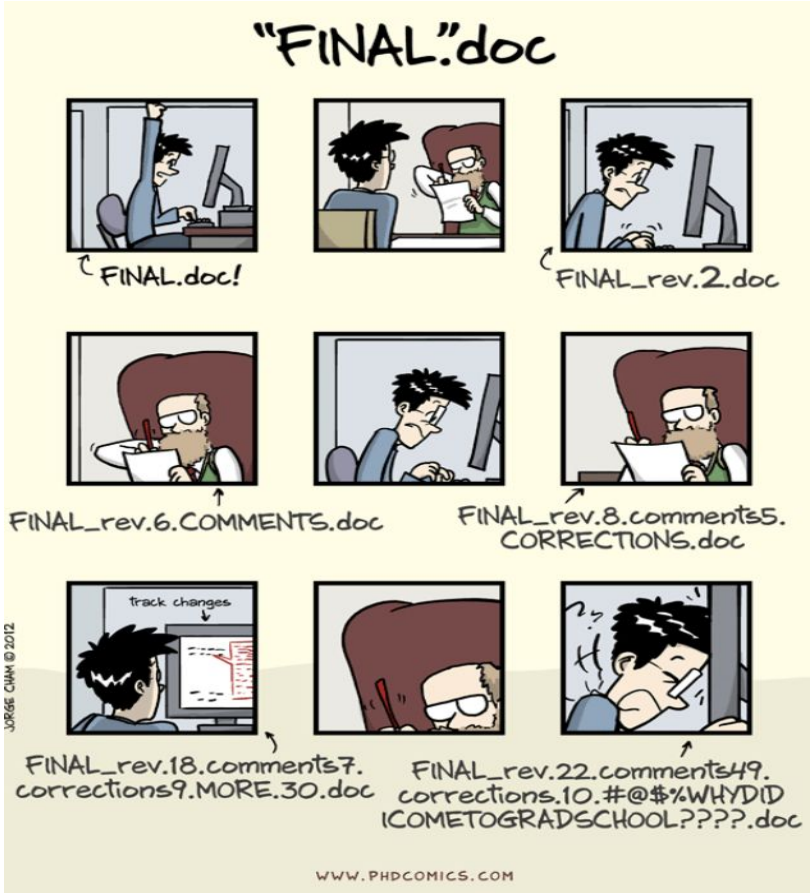
Navigate Anaconda

- Opening Jupyter Notebook from Anaconda

Terminal 101

- Basic terminal command:
  - pwd → print working directory
  - mkdir → make a directory
  - ls → list of documents in a directory
  - cd → change directory
  - cd .. → goes back to previous directory
  - cd ~ → goes back to root directory
  - Mv → move stuff around; change name
  - rm → remove stuff
    - rm -r directory
  - Opening jupyter notebook from terminal by typing command “Jupyter Notebook”
- Python commands from terminal
  - Pip install packages
  - Ipython vs python

Navigating Git & Github



Navigating Git & Github

- What is Git?
  - Git is an open source distributed version control tool for tracking changes in code.
- What is Github?
  - Github is a hosting services for git repositories (“repos”)

Navigating Git & Github

- Navigating Github

  Example - https://github.com/datasciencemasters/go

Fork

- “A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.”

Navigating Git & Github

- Setting up Github & Git
  - Github: to set up github, simply register on the website with your personal email
  - Git: to set up git, go to https://help.github.com/articles/set-up-git/
  - To configure Git
    - git config --global user.name “firstname lastname”
    - git config --global user.email “your email here”

Navigating Git & Github

- Create a new repo
- Cloning the repo to your local machine
- Making changes to the repo & push the changes
- Push the changes from

Git Commands

- create a file: **touch** [filename]
- Cloning a repo from github: **git clone [url]**
- check the status: **git status**
- track and stage a single file: **git add [filename]**
- track and stage all files: **git add .**
- commit with a message: **git commit -m "description of commit”**
- Go back to a previous commit git revert
- view the log: **git log --oneline**
- push changes: **git push** [remotename] [branchname]
- adding an upstream **git remote add** upstream [URL]
- Pull changes from remote: **git pull** [remotename] [branchname]

Git Commands