

COMP10020

Introduction to Programming II

Dr. Brian Mac Namee

brian.macnamee@ucd.ie

School of Computer Science
University College Dublin

TWO EXAMPLES

The Monty Hall Problem



The Big Deal



The Big Deal

1

2

3

- One door has a prize of a car behind it
- The other two doors have prizes of goats!

The Big Deal

1

2

3

- You have to pick a door to win the prize behind it

The Big Deal

1

2

3

- You have to pick a door to win the prize behind it
- But there is a twist!

The Big Deal

1

2

3

- After you pick a door, Monty reveals a goat behind one of the remaining doors

The Big Deal

1

2

3

- After you pick a door, Monty reveals a goat behind one of the remaining doors

The Big Deal

1

2

3

Let's have a go!

The Big Deal

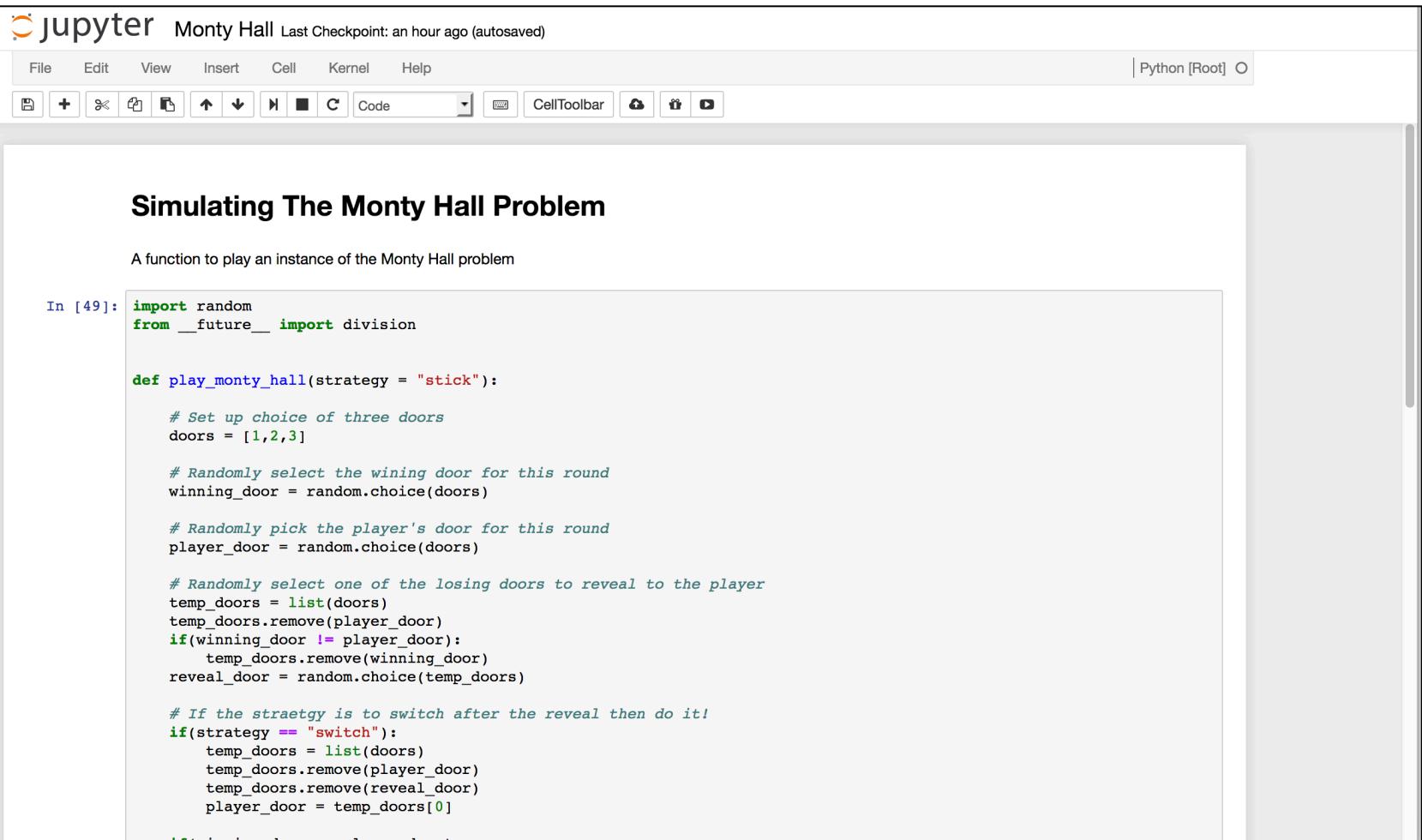
1

2

3

- The big question is should you stick or switch?
 - Stick?
 - Switch?
 - No difference?

Write Some Code!



The screenshot shows a Jupyter Notebook interface with the title "jupyter Monty Hall Last Checkpoint: an hour ago (autosaved)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, Help, and a toolbar with various icons. The code cell (In [49]) contains Python code for simulating the Monty Hall problem.

```
In [49]: import random
from __future__ import division

def play_monty_hall(strategy = "stick"):

    # Set up choice of three doors
    doors = [1,2,3]

    # Randomly select the winning door for this round
    winning_door = random.choice(doors)

    # Randomly pick the player's door for this round
    player_door = random.choice(doors)

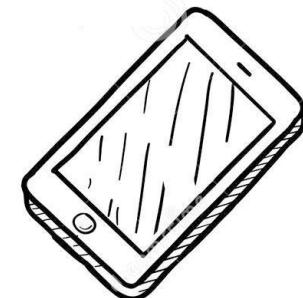
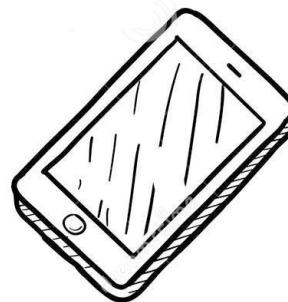
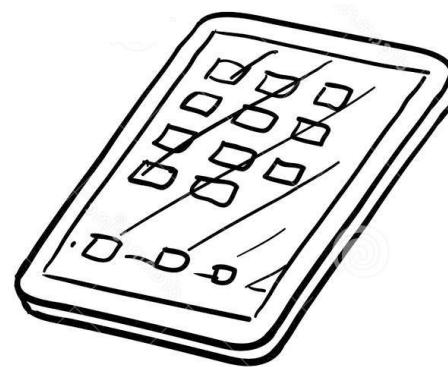
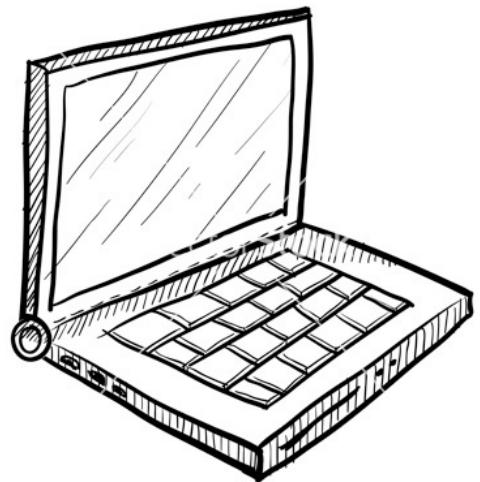
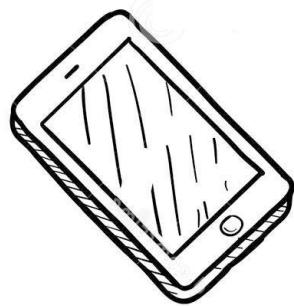
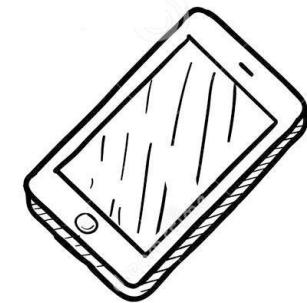
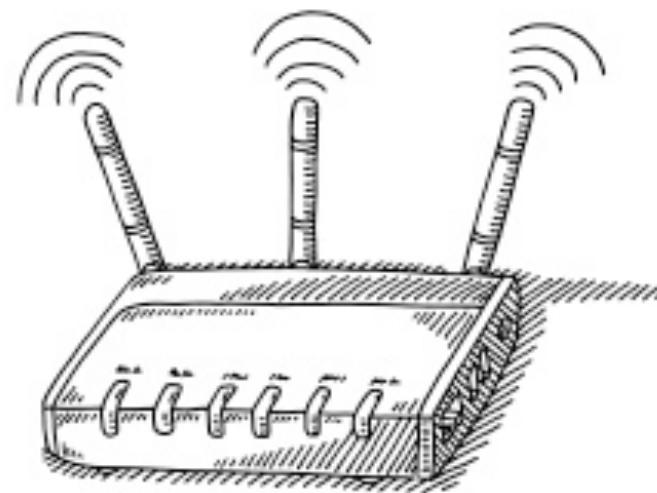
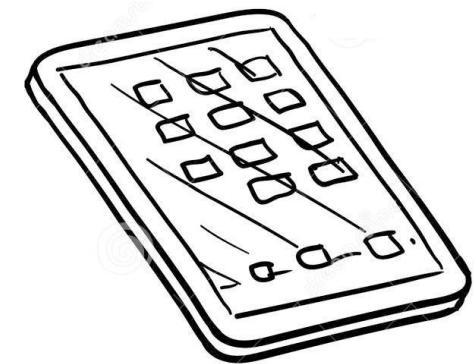
    # Randomly select one of the losing doors to reveal to the player
    temp_doors = list(doors)
    temp_doors.remove(player_door)
    if(winning_door != player_door):
        temp_doors.remove(winning_door)
    reveal_door = random.choice(temp_doors)

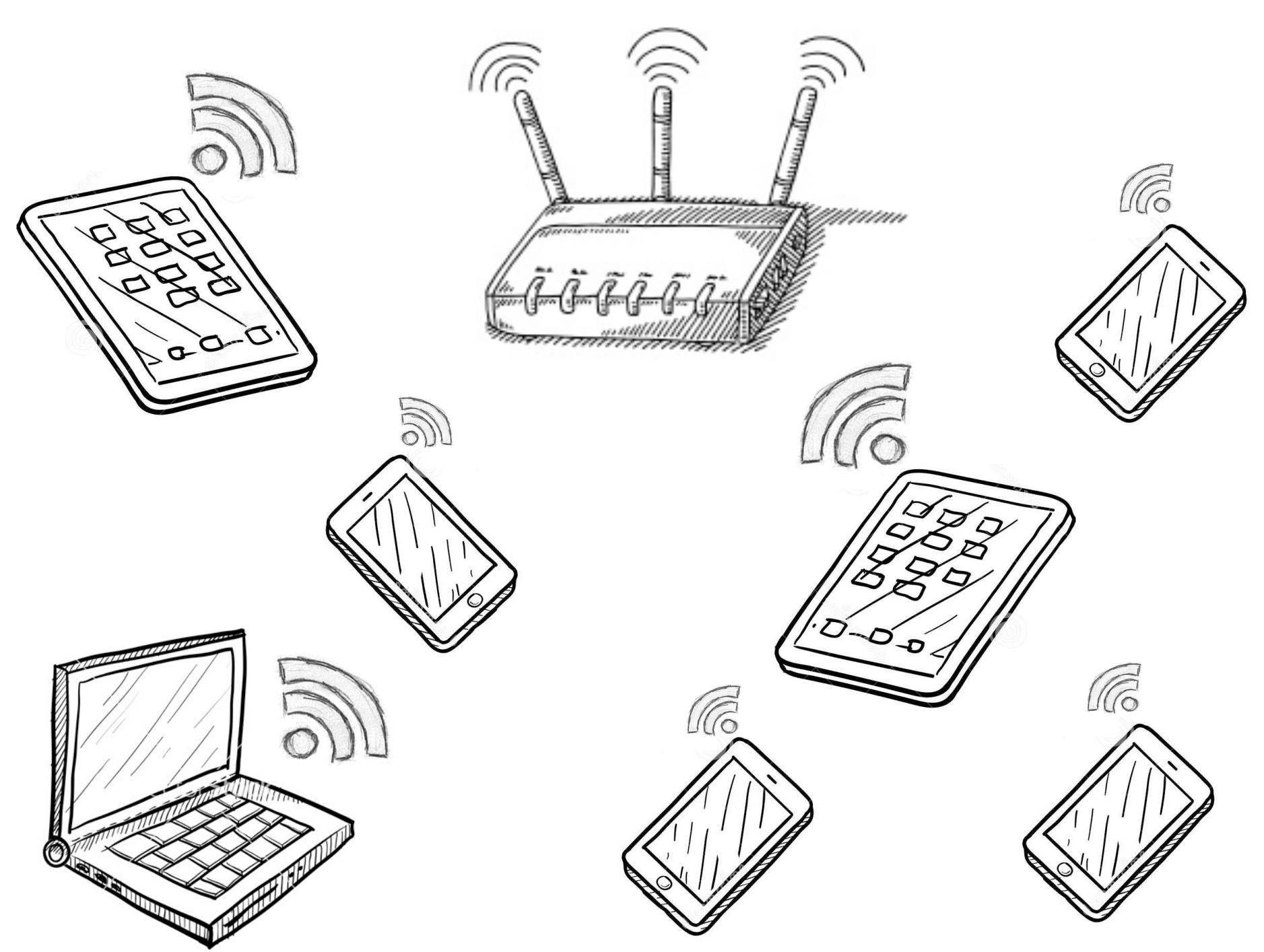
    # If the strategy is to switch after the reveal then do it!
    if(strategy == "switch"):
        temp_doors = list(doors)
        temp_doors.remove(player_door)
        temp_doors.remove(reveal_door)
        player_door = temp_doors[0]

    # If winning_door == player_door:

```







Write Some Code!

jupyter Probe Request Sniffer Runnable Instructions Last Checkpoint: 09/22/2016 (unsaved changes)

File Edit View Insert Cell Kernel Help Python [Root]

```
if hasattr(pkt, 'info'):
    ssid = pkt.info
    ssid = ssid.strip()
    ssid = ''.join(filter(lambda x:x in string.printable, ssid))
    # Not very good, but simple error handling - ssids of more than 32 characters probably indicate malform
    #if(len(ssid) <= 32 and ssid.find("\x00") == -1):
    if(all(c in string.printable for c in ssid) and (len(ssid) > 5)):
        devices[pkt.addr2]["ssids"].add(ssid)

print(str(len(devices)) + " found")
```

33 found

Write out the devices read in into a nice csv format

```
In [8]: with open('devices' + str(time.time()) + '.csv', 'wb') as csvfile:
    writer = csv.writer(csvfile, delimiter=',',
                        quotechar='|', quoting=csv.QUOTE_MINIMAL)
    writer.writerow(['ID', 'freq', 'manu', 'ssids', 'last_seen'])

    for d in devices:
        ssidString = " | ".join(str(e) for e in devices[d]['ssids'])
        writer.writerow([d, devices[d]['freq'], devices[d]['manu'], ssidString, devices[d]['last_seen']])
```

Generate a list of the ssids found

```
In [9]: ssidsFound = []

# Iterate through the devices list
for d in devices:

    # get the list of ssids saved from a device
    ssids = devices[d]["ssids"]
```

OVERVIEW

Module Outline

1. Python

- Revision
- Object oriented (OO) programming

2. Algorithms

- Problem solving
- Searching & ranking
- Data analysis

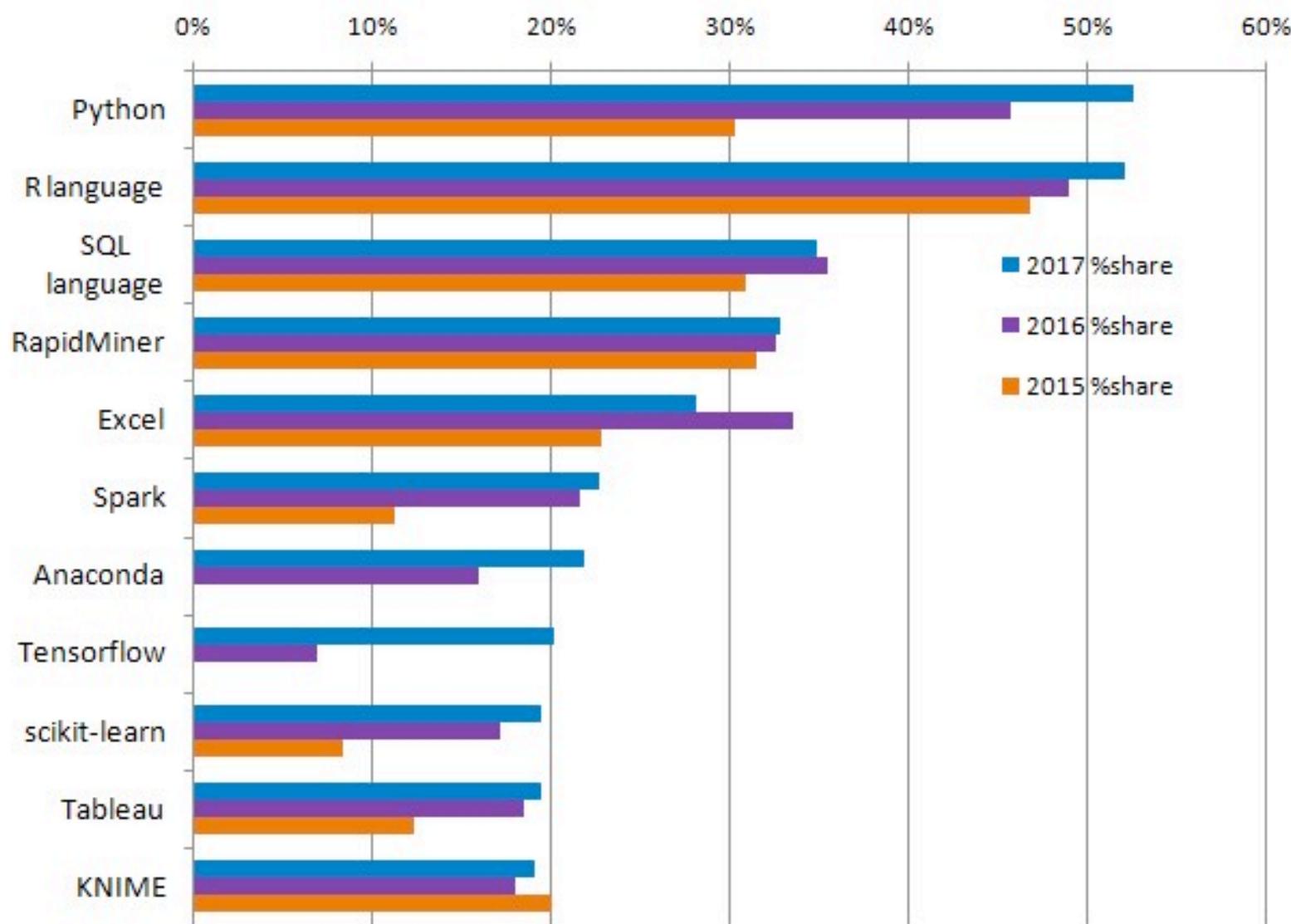
3. Data Science

- Data access
- Data manipulation
- Data analysis
- Data visualisation

WHY PYTHON?

Language Rank	Types	Spectrum Ranking
1. Python		100.0
2. C		99.7
3. Java		99.5
4. C++		97.1
5. C#		87.7
6. R		87.7
7. JavaScript		85.6
8. PHP		81.2
9. Go		75.1
10. Swift		73.7

KDnuggets Analytics, Data Science, Machine Learning Software Poll, top tools share, 2015-2017



Why Python?

Open source and well supported by freely available tools

Clean, concise, unambiguous syntax

Supports a variety of programming paradigms

- Simple scripts
- Object-oriented programming
- Interactive notebooks

Strong library support

Strong online community support

Why Python?

Open source and well supported by freely available tools

Clear

Supp

```
for line in open("file.txt"):  
    for word in line.split():  
        if word.endswith('ing'):  
            print(word)
```

Strong library support

Strong online community support

Why Python?

Open source and well supported by freely available tools

Clean, concise, unambiguous syntax

Supports a variety of programming paradigms

- Simple scripts
- Object-oriented programming
- Interactive notebooks

Strong library support

Strong online community support

Why Python?

Open source and well supported by freely available tools

Clean, concise

Supports a wide range of paradigms

- Simple syntax
- Object-oriented
- Interactive

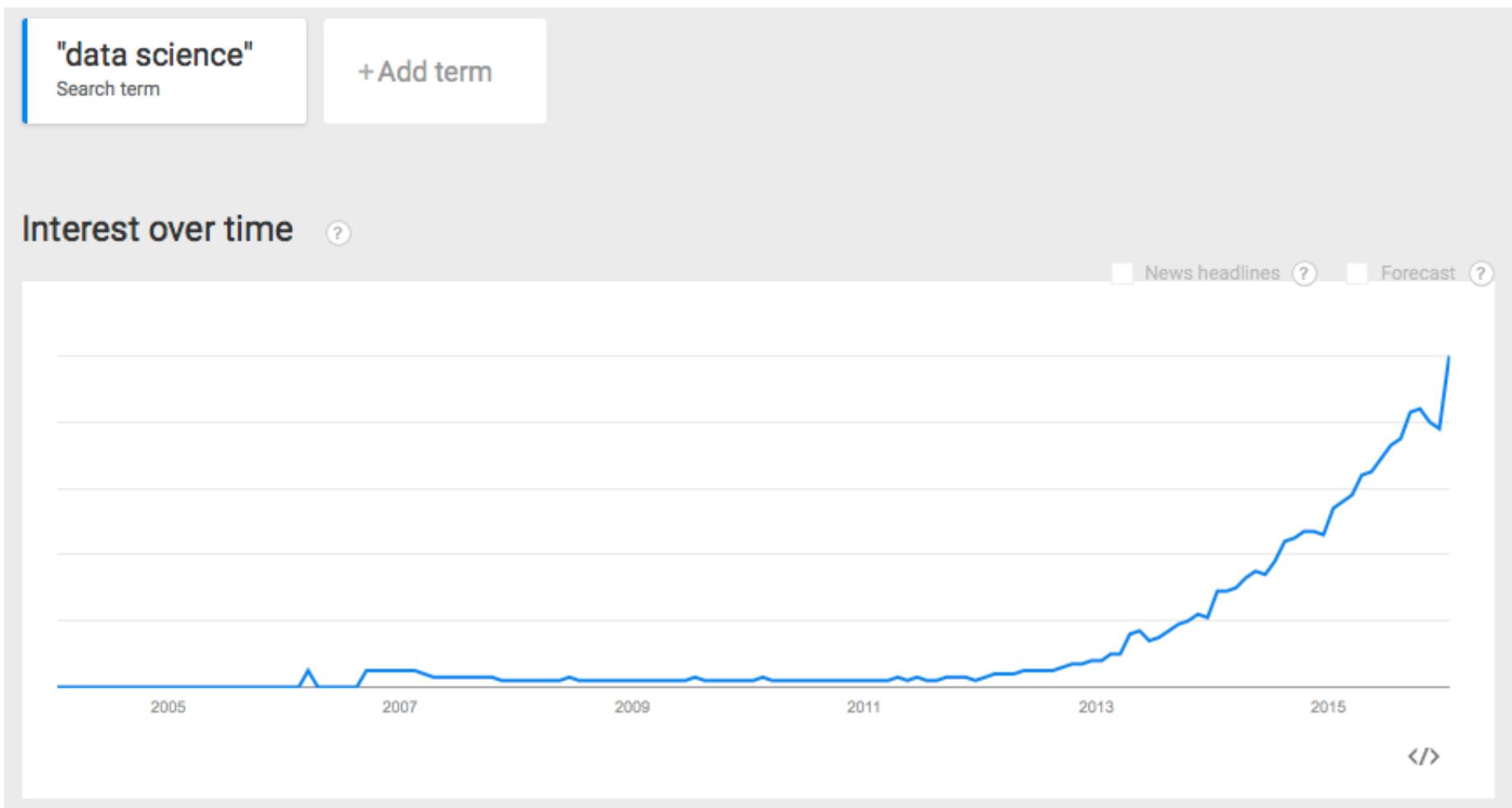
And you know it
already from
COMP10010!!

Strong library support

Strong online community support

WHAT IS DATA SCIENCE?

What Is Data Science?

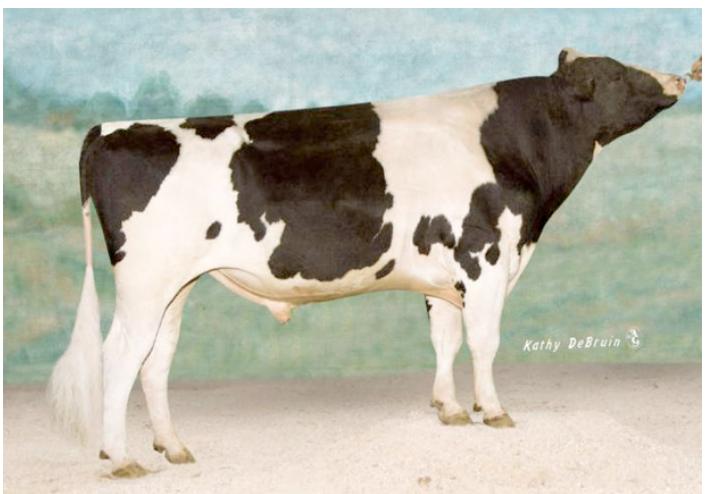
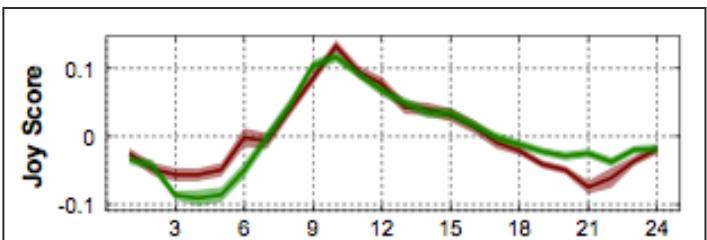
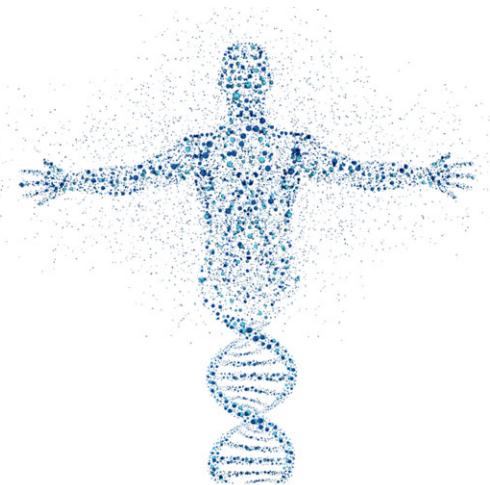


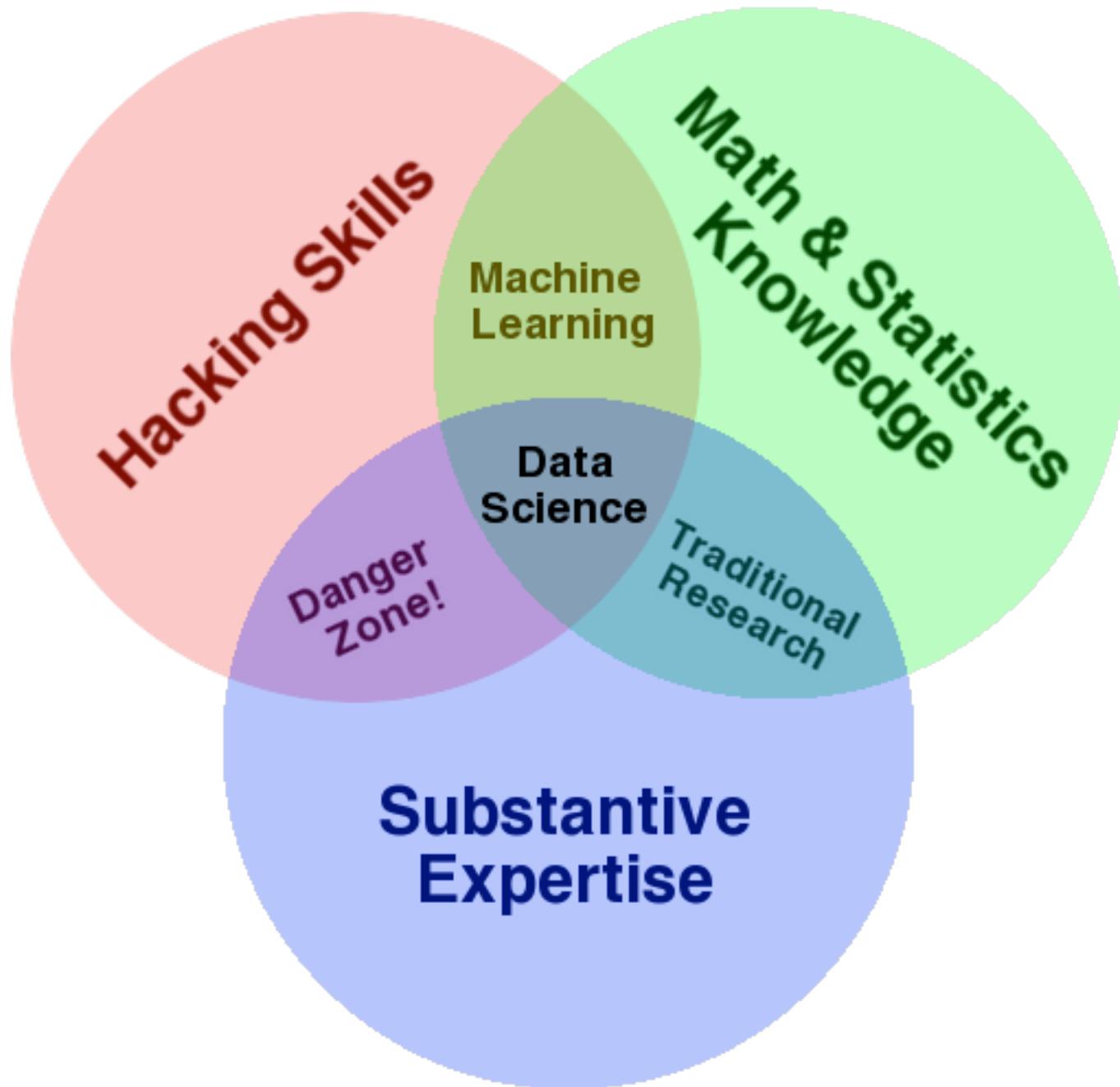
<https://www.google.com/trends/explore#q=%22data%20science%22&cmpt=q>

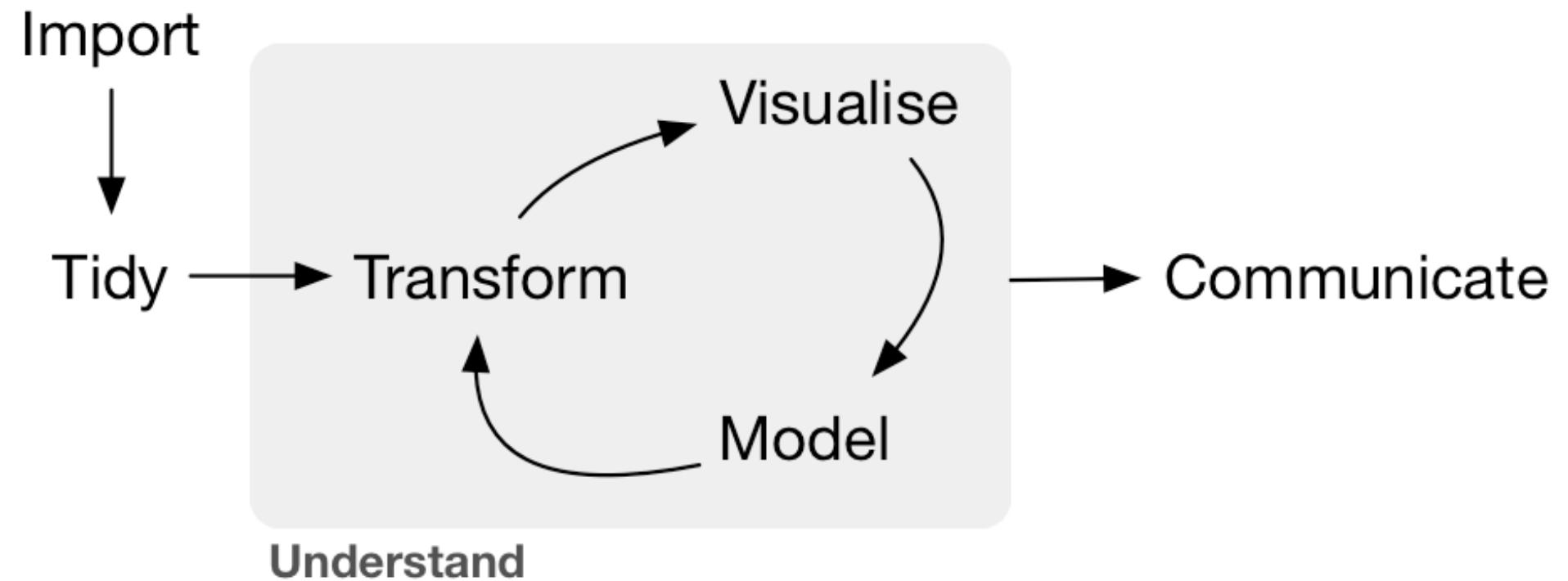
What Is Data Science?

At its core Data Science is about developing the infrastructure and processes for dealing with data at scale, recognising and understanding patterns within large, diverse datasets, generating predictions based on these patterns, and creating revealing visualizations and crafting compelling narratives with and about data

DATA-DRIVEN EVERYTHING







What Is Data Science?

Data Science brings together key ideas from multiple fields

- Computer science (algorithms, representation, visualization, application development)
- Statistics (modelling, analysis, prediction)
- Design (information design, interaction design)
- Psychology and cognitive science (language and perception),
- Humanities and social sciences (storytelling and narrative, social learning)

What Is Data Science?

Data Science brings together key ideas from multiple fields

- Computer science, visualization, statistics, design, psychology, perception
- Data science is also a nice application area off which we can hang a programming course
- Humanities and social sciences (storytelling and narrative, social learning)

ADMINISTRAVIA

Dr Brian Mac Namee

Lecturer in the School of Computer Science at University College Dublin

A principal investigator at the Centre for Applied Data Analytics Research (CeADAR)

A funded investigator at the Insight Centre for Data Analytics

Director of Training for The Analytics Store, a data analytics consultancy and training company

Practical Information

Lectures

- Tuesday 10:00 - 11:00 H1.26
- Thursday 10:00 - 11:00 H1.26

Labs

- Tuesday 16:00 - 18:00 B1.08 CSI
- Wednesday 16:00 - 18:00 B1.08 CSI
- Thursday 16:00 - 18:00 B1.08 CSI

Materials

- Moodle page currently open for registration via self-enrolment using password: **itp2access**
- <https://csmoodle.ucd.ie/moodle/course/view.php?id=662>

Practical Information

Lectures

- Tuesday 10:00 - 11:00 H1.26
- Thursday 10:00 - 11:00 H1.26

Labs

- Tuesday
- Wednesday
- Thursday

Labs do not start
until week 3

Materials

- Moodle page currently open for registration via self-enrolment using password: **itp2access**
- <https://csmoodle.ucd.ie/moodle/course/view.php?id=662>

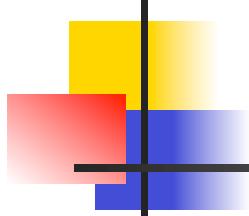
Assessment

Continuous assessment

- 60% of final mark
- A series of programming assignments in labs

End of semester exam

- 40% of final mark
- Exam covering practical and theoretical issues discussed in class



Plagiarism & UCD Computer Science

- **Plagiarism is a serious academic offence**
 - [Student Code, section 6.2 & 6.3] or [UCD Registry Plagiarism Policy] or [CS Plagiarism policy and procedures]
- Our staff and demonstrators are **proactive** in looking for possible plagiarism in all submitted work
- Suspected plagiarism is investigated by the CS Plagiarism subcommittee
 - Usually includes an interview with student(s) involved
 - 1st offence: **usually** 0 or NG in the affected components
 - 2nd offence: may be referred to the **University disciplinary committee**
- Student who enables plagiarism is equally responsible

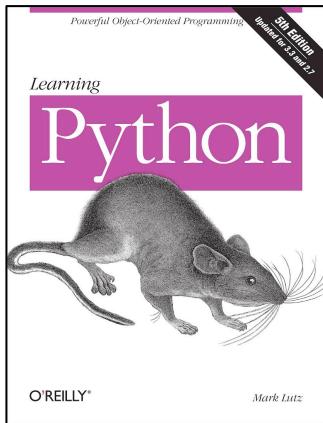
<http://www.ucd.ie/students/guide/academicregs.html>

<http://libguides.ucd.ie/academicintegrity>

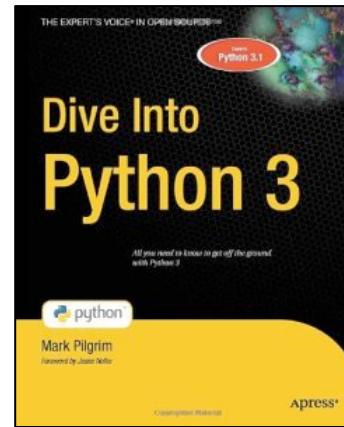
BOOKS & OTHER RESOURCES

Books & Other Resources

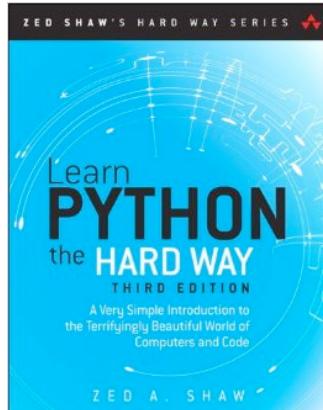
No specific textbook for this module



Learning Python
Mark Lutz
O'Reilly Media
www.learning-python.com/books/



Dive Into
Python 3
Mark Pilgrim
Apress
www.diveintopython3.net



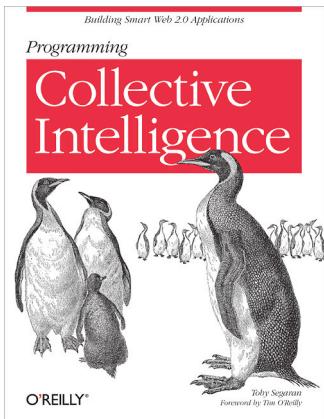
Learn Python the
Hard Way
Zed A. Shaw
www.learnpythonthehardway.org

Books & Other Resources

No specific textbook for this module



Fundamentals of Machine Learning for
Predictive Data Analytics
John D. Kelleher, Brian Mac Namee,
Aoife D'Arcy
MIT Press
www.machinelearningbook.com



Programming Collective
Intelligence
Toby Segaran
O'Reilly Media
www.kiwitobes.com