IMPERIAL
BUSINESS SCHOOL

Data Structures and Algorithms

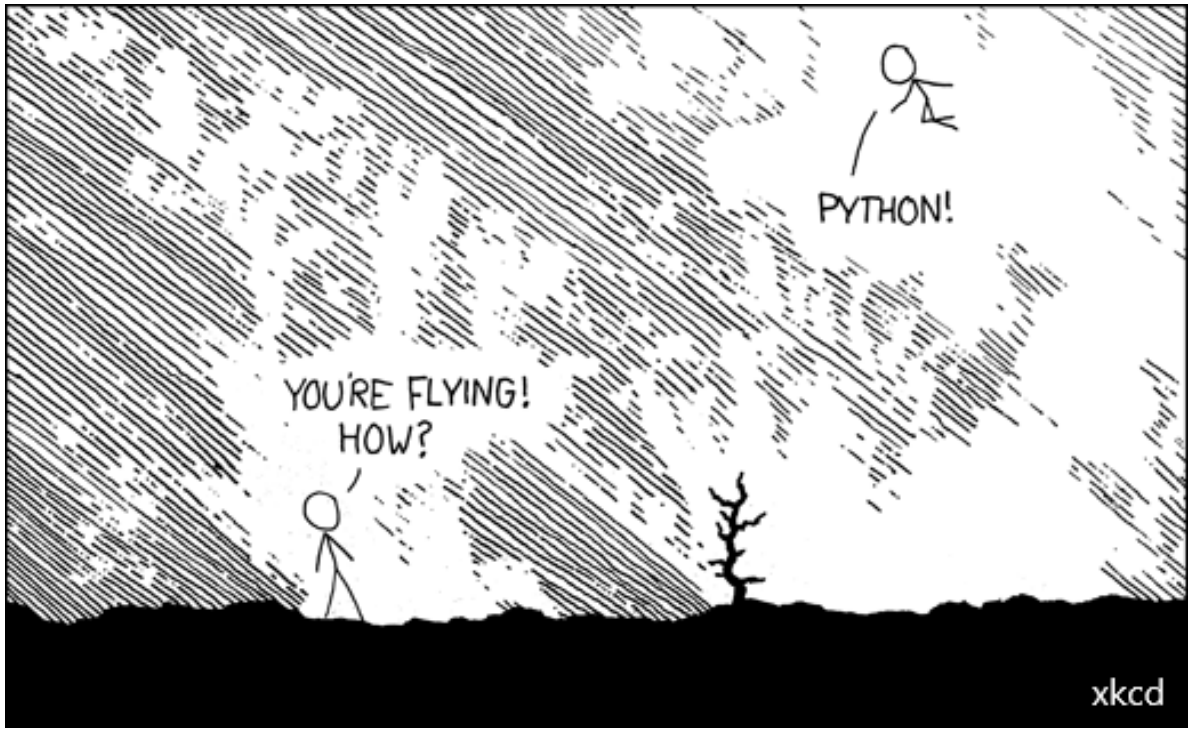
Lecture 1

Ryan Cory-Wright
r.cory-wright@imperial.ac.uk

Today

1. **What this module is about**
2. Introductions
3. What we will do
4. A discussion of Chat-GPT and other LLMs
5. Illustrating the language: Python + exercises

What this module is about



- Computational problem-solving
- Understanding what computers can and cannot do efficiently
- Making a computer do these things

US & WORLD

Prisons turn to computer algorithms for deciding who to parole

18 COMMENTS

By Jacob Kastrenakes on October 14, 2013 10:06 am Email @jake_k

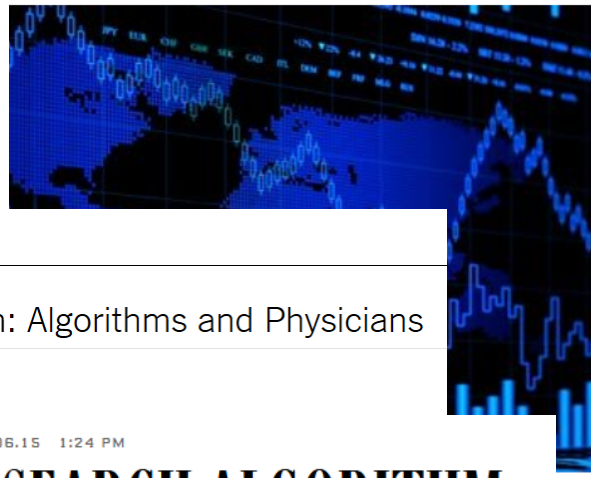
How algorithms rule the world

The NSA revelations highlight the role sophisticated algorithms play in sifting through masses of data. But more surprising is their widespread use in our everyday lives. So should we be more wary of their power?

TECH

At UPS, the Algorithm Is the Driver

Turn right, turn left, turn right: inside Orion, the 10-year effort to squeeze every penny from delivery routes



Music in the age of the algorithm

We now have instant access to almost any song. Could our tastes be narrowing as a result?

TheUpshot

Your New Medical Team: Algorithms and Physicians

Austin Frakt
THE NEW HEALTH CARE DEC. 7, 2015

ADAM ROGERS SCIENCE 08.06.15 1:24 PM

GOOGLE'S SEARCH ALGORITHM COULD STEAL THE PRESIDENCY

TECH • ARTIFICIAL INTELLIGENCE

The AI Arms Race Is Changing Everything

FELIX SALMON AND JON STOKES MAGAZINE 12.27.10 12:00 PM

ALGORITHMS TAKE CONTROL OF WALL STREET

A-HED

What's Hot in the Art World? Algorithms

Admirers hold on to computerized formulas; paying $2,500 for a 'qrpff' necktie

Hedge funds + Follow

Pedro Domingos
@pmddomingos

Starting May 25, the European Union will require algorithms to explain their output, making deep learning illegal.

3:59 AM · Jan 29, 2018

38 239 308 2

News Sport Weather iPlayer TV

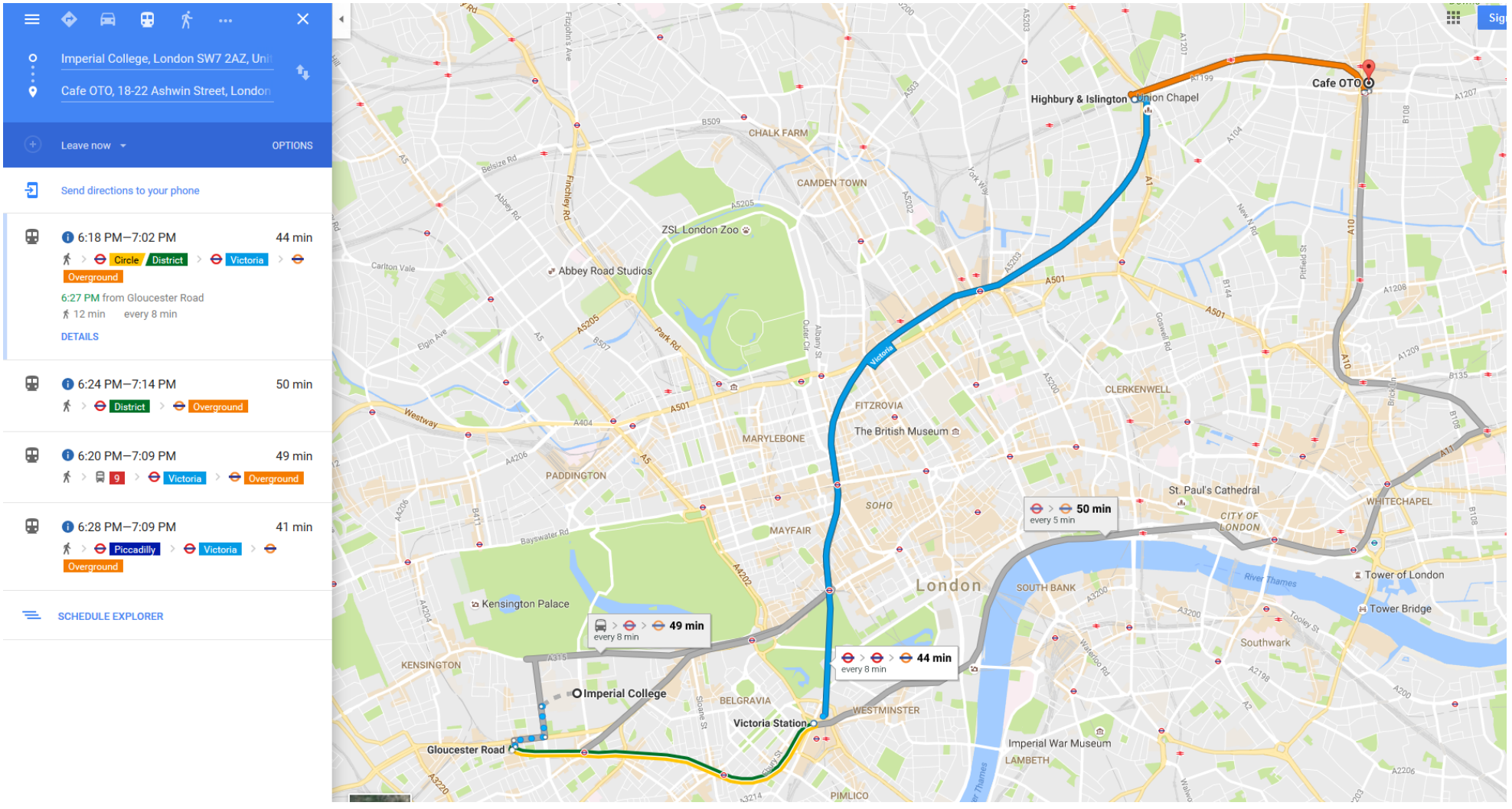
NEWS

Home UK World Business Politics Tech Science Health Education Ent

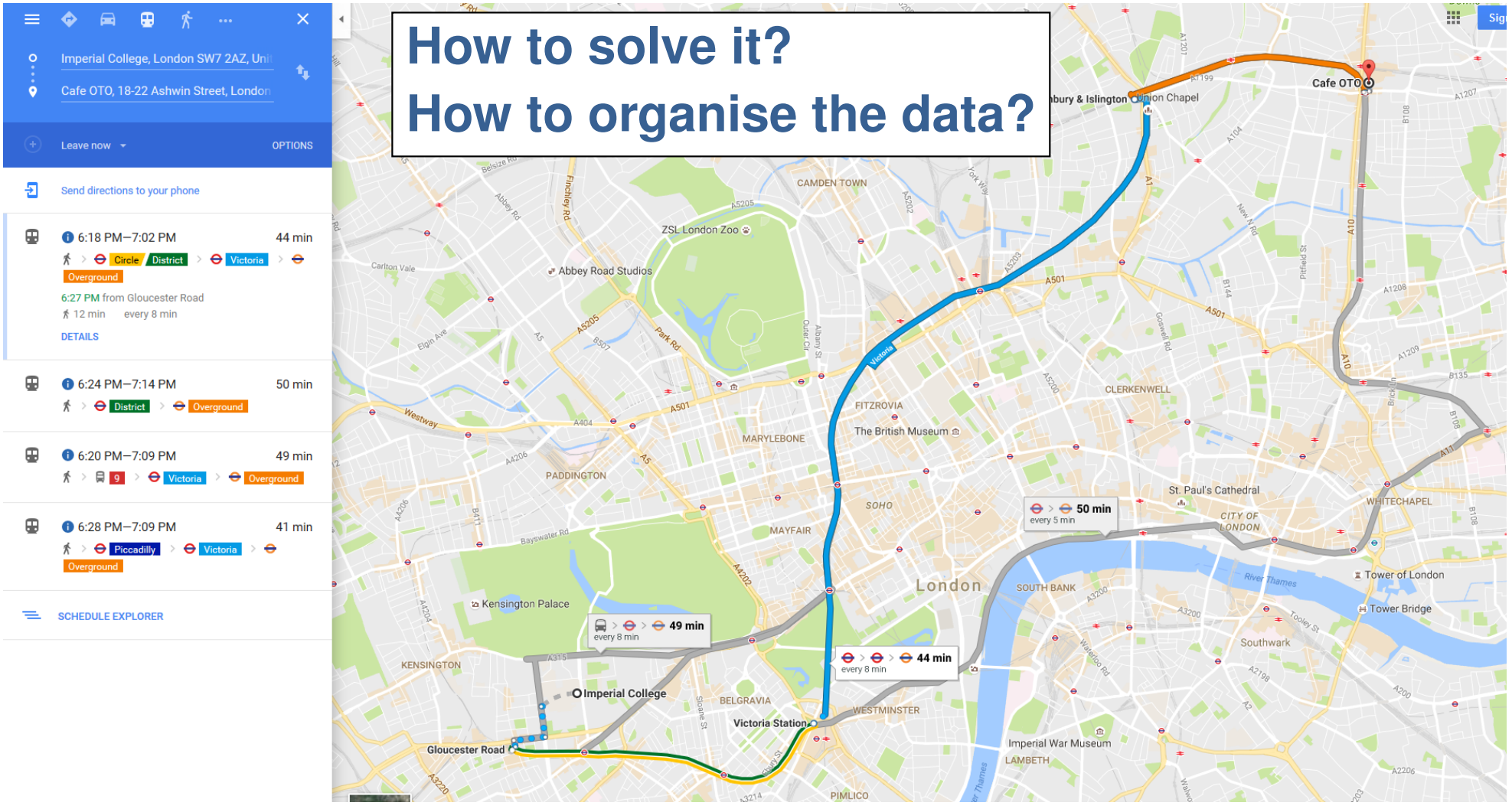
Technology

When algorithms control the world

By Jane Wakefield
Technology reporter



Getting from South Ken to a cafe in Dalston?



Getting from South Ken to a cafe in Dalston?

Solving computational problems

Data = digitised information

Data structures describe ways to organise data

Algorithms describe how we process data:

- Step-by-step instructions
- Take input data and produce output data

We write algorithms into **programs** (in Python)

Computers interpret and execute programs

What are computers good at?

- **Performing (well-defined) calculations** — built-in primitives
- **Remembering the results**

What are computers good at?

- **Performing (well-defined) calculations** — built-in primitives
- **Remembering the results**

But we also need **good algorithm design**

- How should we schedule flights at Heathrow airport?
- How should the NHS allocate patients to beds at a hospital?
- How should two-sided marketplaces (e.g., ride-sharing apps) match participants?

What are computers good at?

- **Performing (well-defined) calculations** — built-in primitives
- **Remembering the results**

But we also need **good algorithm design**

- How should we schedule flights at Heathrow airport?
- How should the NHS allocate patients to beds at a hospital?
- How should two-sided marketplaces (e.g., ride-sharing apps) match participants?

Computers **have a harder time**:

- Solving less well-defined problems: understanding humour? Deciding what art is good?
- Subject to ongoing research...

What are computers good at?

- **Performing (well-defined) calculations** — built-in primitives
- **Remembering the results**

But we also need **good algorithm design**

- How should we schedule flights at Heathrow airport?
- How should the NHS allocate patients to beds at a hospital?
- How should two-sided marketplaces (e.g., ride-sharing apps) match participants?

Computers **have a harder time**:

- Solving less well-defined problems: understanding humour? Deciding what art is good?
- Subject to ongoing research... Although LLMs are getting good!

What are computers good at?

- **Performing (well-defined) calculations** — built-in primitives
- **Remembering the results**

But we also need **good algorithm design**

- How should we schedule flights at Heathrow airport?
- How should the NHS allocate patients to beds at a hospital?
- How should two-sided marketplaces (e.g., ride-sharing apps) match participants?

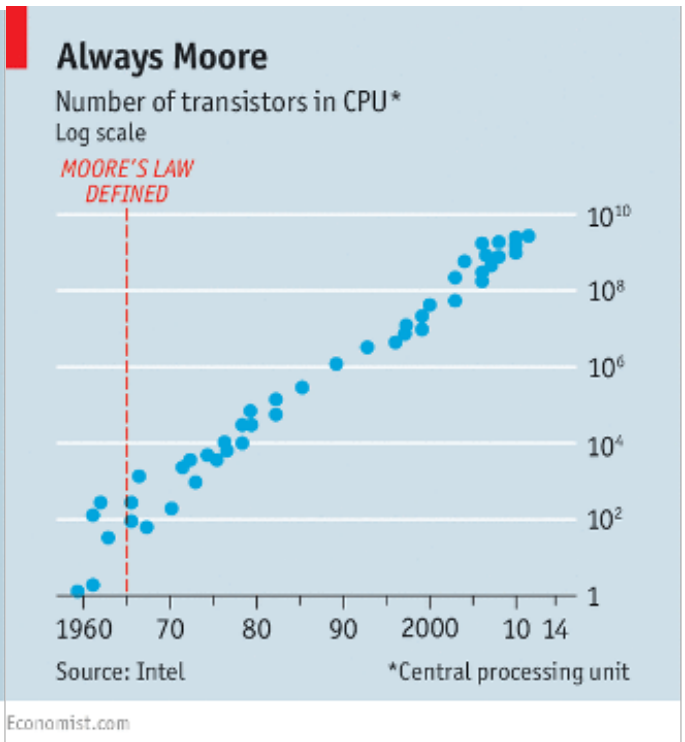
Computers **have a harder time**:

- Solving less well-defined problems: understanding humour? Deciding what art is good?
- Subject to ongoing research... Although LLMs are getting good!

This course focuses on understanding the first category

Why Now?

- Data and algorithms are not new—they have been around for decades.
- But today's environment creates new opportunities to transform business practices



Growth of Related Disciplines

Strong demand for professionals aware of algorithms and data

OCCUPATION	GROWTH RATE, 2022-32	2022 MEDIAN PAY
Wind turbine service technicians	45%	$57,320 per year
Nurse practitioners	45%	$121,610 per year
Data scientists	35%	$103,500 per year
Statisticians	32%	$98,920 per year
Information security analysts	32%	$112,000 per year
Medical and health services managers	28%	$104,830 per year
Epidemiologists	27%	$78,520 per year
Physician assistants	27%	$126,010 per year
Physical therapist assistants	26%	$62,770 per year
Software developers	26%	$127,260 per year

Figure: Fastest Growing US Occupations 2022-32 (projected). Source: US Bureau of Labor Statistics

Of the ten occupations on the list, four involve topics related to class!

A Lesson From Investing

Which line would you prefer to invest in?

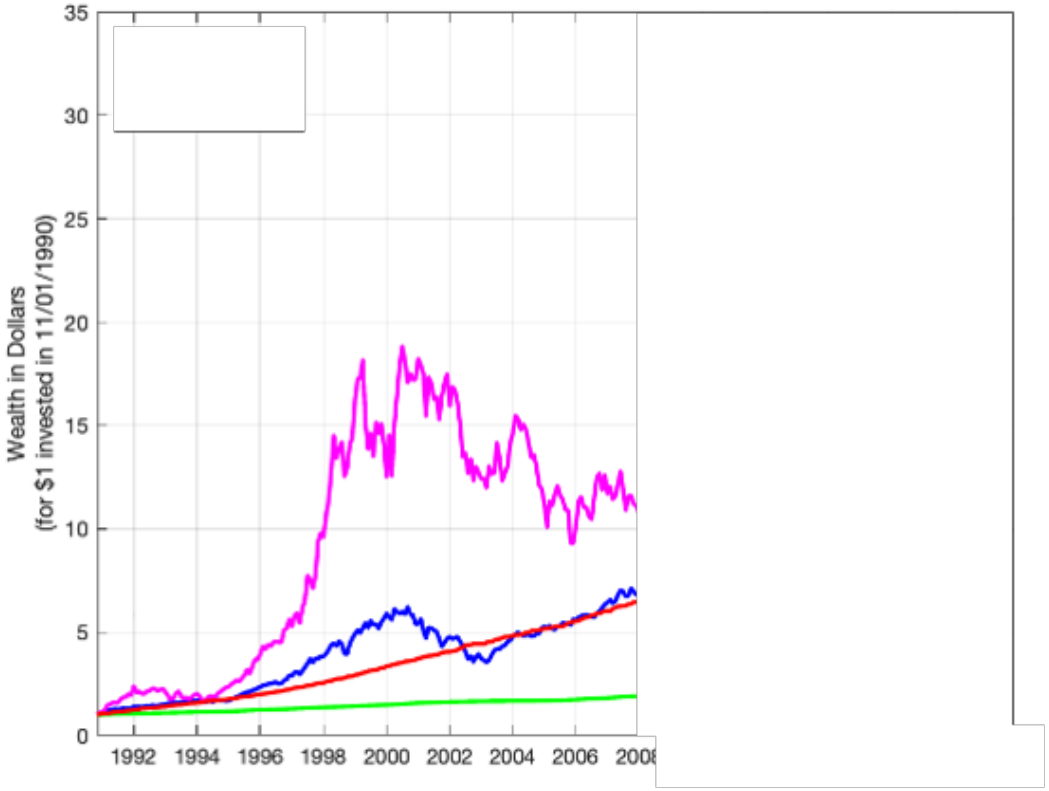


Figure: Different Options for Investing

A Lesson From Investing

Which line would you prefer to invest in?

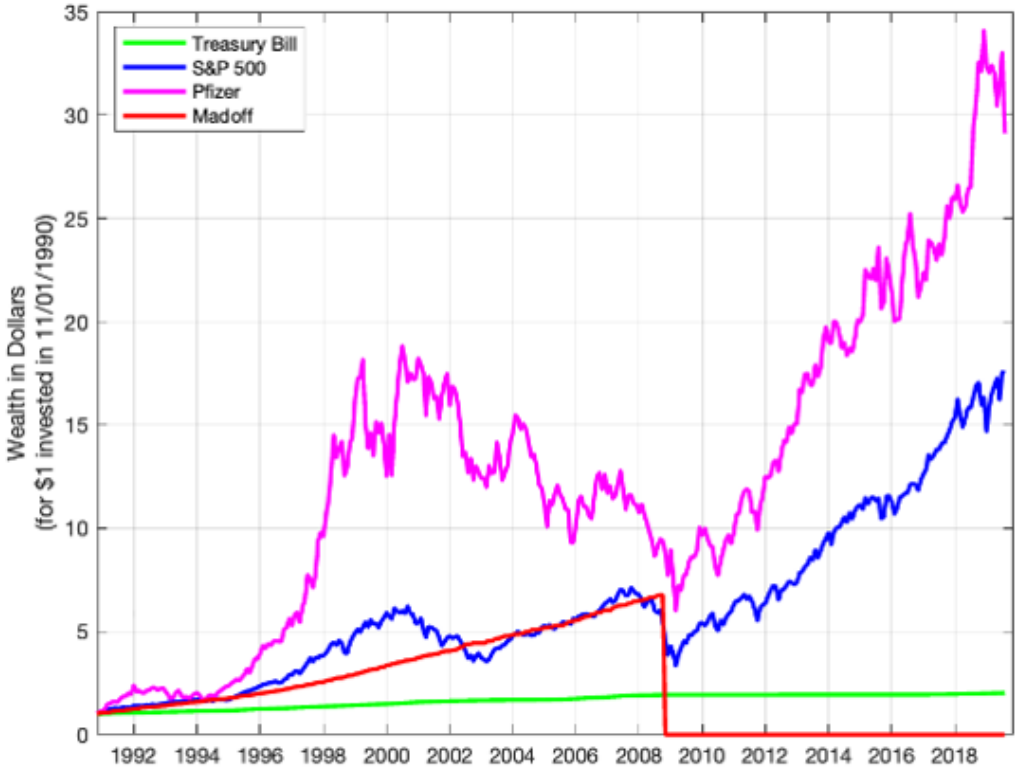


Figure: Different Options for Investing

A Lesson From Investing

Which line would you prefer to invest in?

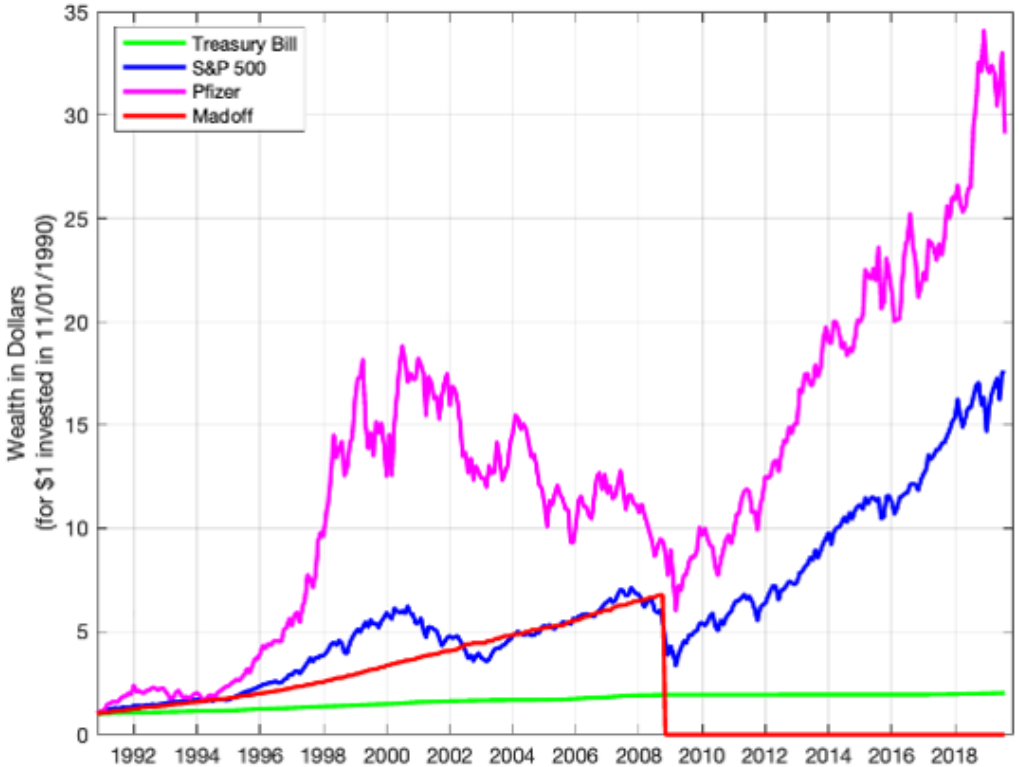


Figure: Different Options for Investing

Lesson: the future might not look like the past. Nonetheless, this class teaches transferable skills for reasoning about and solving

A Lesson from WWII

British RAF lost lots of planes to German anti-aircraft fire in WWII.
They mapped all bullet holes: where to put extra armour?

A Lesson from WWII

British RAF lost lots of planes to German anti-aircraft fire in WWII.
They mapped all bullet holes: where to put extra armour?

Lesson: the data may not tell the whole story; you also need **context**.

Why These Examples Are Relevant

Often need to use data and algorithms to make decisions about the real world that perform well in practice. Otherwise, you'll be guided by intuition alone, which is notoriously bad at making good decisions.

Why These Examples Are Relevant

Often need to use data and algorithms to make decisions about the real world that perform well in practice. Otherwise, you'll be guided by intuition alone, which is notoriously bad at making good decisions.

Exercise: write out a sequence of 20 coin tosses that look random. E.g. "HTH" for the first 3 tosses.

Why These Examples Are Relevant

Often need to use data and algorithms to make decisions about the real world that perform well in practice. Otherwise, you'll be guided by intuition alone, which is notoriously bad at making good decisions.

Exercise: write out a sequence of 20 coin tosses that look random. E.g. "HTH" for the first 3 tosses.

Count the longest consecutive sequence of "H". How many H's?

Why These Examples Are Relevant

Often need to use data and algorithms to make decisions about the real world that perform well in practice. Otherwise, you'll be guided by intuition alone, which is notoriously bad at making good decisions.

Exercise: write out a sequence of 20 coin tosses that look random. E.g. "HTH" for the first 3 tosses.

Count the longest consecutive sequence of "H". How many H's? Mathematically ("The Longest Run of Heads" by Mark F. Schilling, The College Mathematics Journal, 21(3):196–207, 1990.), the longest run should be of length around $\log_2(20) - 1 = 3.33$, i.e., 3.

Why These Examples Are Relevant

Often need to use data and algorithms to make decisions about the real world that perform well in practice. Otherwise, you'll be guided by intuition alone, which is notoriously bad at making good decisions.

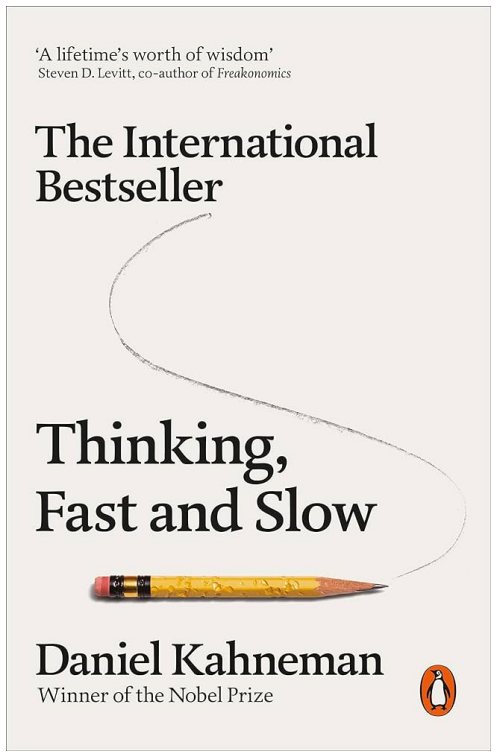
Exercise: write out a sequence of 20 coin tosses that look random. E.g. "HTH" for the first 3 tosses.

Count the longest consecutive sequence of "H". How many H's? Mathematically ("The Longest Run of Heads" by Mark F. Schilling, The College Mathematics Journal, 21(3):196–207, 1990.), the longest run should be of length around $\log_2(20) - 1 = 3.33$, i.e., 3.

Most of you will have a longest run of heads of length 2–3.

More on Predictable Irrationality

Over the next break, consider reading “Thinking, Fast and Slow” by Daniel Kahneman. Some of the experiments in the book have since been debunked (“More social science studies just failed to replicate. Here’s why this is good.” by Brian Resnick, Vox, 2018), but the overall story is still interesting.



Today

1. What this module is about
2. **Introductions**
3. What we will do
4. A discussion of Chat-GPT and other LLMs
5. Illustrating the language: Python + exercises

Team: Module Lead

- Ryan Cory-Wright, r.cory-wright@imperial.ac.uk
- Assistant Professor of Analytics and Operations at Imperial Business School since 2023
- Prev: Postdoc at IBM, PhD in Operations Research at MIT.
- Research: Optimization, Machine Learning, Statistics, Renewable Energy, AI
- Coding: Julia (native), Python, MATLAB, C++, Java, etc.

Team: Tutorial Leads

Tutorial Leader:

- Jay DesLauriers, j.deslauriers@imperial.ac.uk
- Senior Teaching Fellow in the Early Career Researcher Institute
- AI Futurist in Education for the Business School
- Research Fellow at the University of Westminster
- Cloud and Edge Orchestration, Swarm Computing

Tutorial Leader:

- Marius Mickunas, m.mickunas@imperial.ac.uk
- Tutor at Imperial Business School, Guest lecturer at the London Business School and ESMT Berlin
- PhD in Immunology from King's College London
- Data Analytics with Python/R, Statistics, Data Visualization, Optimization, Decision and Risk Analysis

Today

1. What this module is about
2. Introductions
3. **What we will do**
4. A discussion of Chat-GPT and other LLMs
5. Illustrating the language: Python + exercises

Philosophy

- Programmers are made, not born
- You will spend most of your time on this module working on exercises
- Mix readings/supplementary material and lectures: Lectures will focus on the most important ideas
- Ask for help
- We are here to help you learn by drinking from the firehose, not to drown you.

What happens in a session

Each session consists of two parts.

1. Lecture (synchronous, 2h)
Live class for discussing the main points of the session.
Second half is devoted to Python exercises.

2. (Important!) Review (asynchronous, 1-3h)
Practice the concepts from the session.

Module Roadmap

Week by week:

- Week one: Introductions
- Week two: Functions–let you make code modular
- Week three: Loops–let you make code iterative
- Week four: Computational complexity–let you understand “how fast is my code?”
- Week five: Sorting–how to analyze an algorithm

Module Roadmap

Week by week:

- Week six: Graphs and Search–incorporating networks
- Week seven: Shortest Path–Dijkstra's algorithm
- Week eight: Greedy Algorithms
- Week nine: Generative AI
- Week ten: A Mock Exam

Assessment

Session hand-ins (altogether 20%) from sessions 2-10. You will submit some of the exercises from each session, to ensure we are moving along at the same pace. You may work together on these and ask me or Marius/Jay for help during the tutorial sessions. We will make it clear well in advance which session hand-ins are graded.

Two individual homework assignments (20%) consisting of written and programming exercises.

Exam (60 %). I will post a mock exam.

Assessment

Session hand-ins (altogether 20%) from sessions 2-10. You will submit some of the exercises from each session, to ensure we are moving along at the same pace. You may work together on these and ask me or Marius/Jay for help during the tutorial sessions. We will make it clear well in advance which session hand-ins are graded.

Two individual homework assignments (20%) consisting of written and programming exercises.

Exam (60 %). I will post a mock exam.

Note! Grade expectations:

- 85% $\rightarrow$ A+, 70% $\rightarrow$ A, 60% $\rightarrow$ B, 50% $\rightarrow$ C (pass)
- Average exam grade is likely to be B
- **Anything above 70% in the exam is excellent**

Making the most of it

Preparation:

- Go through the preparation materials before the lecture.

Working on exercises:

- Code (usually) won't work perfectly on the first go. Don't give up! Try to find the error and fix it. (We will discuss **debugging** programs in Session 3.)
- You may work together on the session exercises (but not on the two individual homework assignments).
- Ask for help if you get stuck.
- You don't need to solve every exercise.

How to get help

1. **Ask during/after lectures**
   - We will work on exercises in the second half of each session
   - We are around to help
   - More details in a moment
2. **Ask during the tutorial**

Expectations

- Please be on time: classes will start at the latest 10 mins past the hour.
- Please don't copy each other; plagiarism gets reported directly to the program team, who have quite strict procedures they follow if plagiarism is detected. It is much better to ask the program team or us for help if you are struggling!

Today

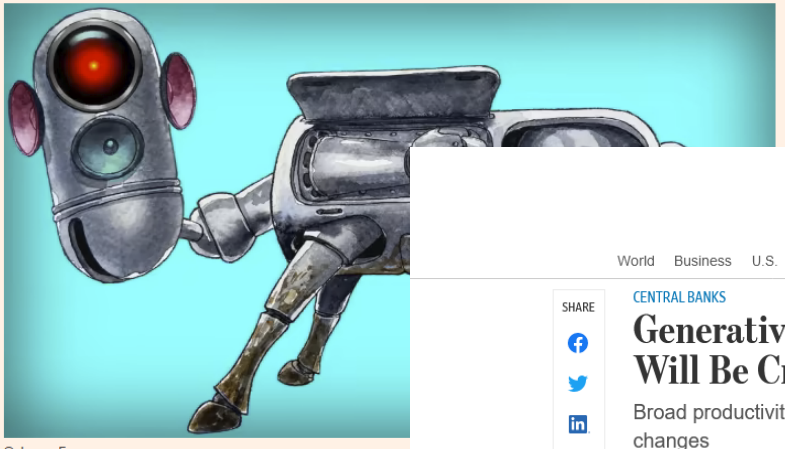
1. What this module is about
2. Introductions
3. What we will do
4. **A discussion of Chat-GPT and other LLMs**
5. Illustrating the language: Python + exercises

Opinion **Artificial intelligence**

The threat and promise of artificial intelligence

It might be the most transformative technology of all for human beings' sense of themselves

MARTIN WOLF + Add to myFT



© James Ferguson

The Economist

Menu | Weekly edition | The world in brief | Search

Podcasts | Babbage

Is GPT-4 the dawn of true artificial intelligence?

Our podcast on science and technology. This week, we delve under the hood of GPT-4 to figure out how revolutionary, and intelligent, this ground-breaking technology really is.

THE WALL STREET JOURNAL.

Subscribe | Sign In

English Edition | Print Edition | Video | Audio | Latest Headlines | More

World Business U.S. Politics Economy Tech Finance Opinion Arts & Culture Lifestyle Real Estate Personal Finance Health Science Style Sports

SHARE

CENTRAL BANKS

Generative AI Promises an Economic Revolution. Managing the Disruption Will Be Crucial.

Broad productivity and economic output gains may be coming, but knowledge workers will face a reckoning as the nature of work changes

IEEE Spectrum FOR THE TECHNOLOGY INSIDER

Type to search

NEWS | COMPUTING

How Coders Can Survive—and Thrive—in a ChatGPT World

›4 tips for programmers to stay ahead of generative AI

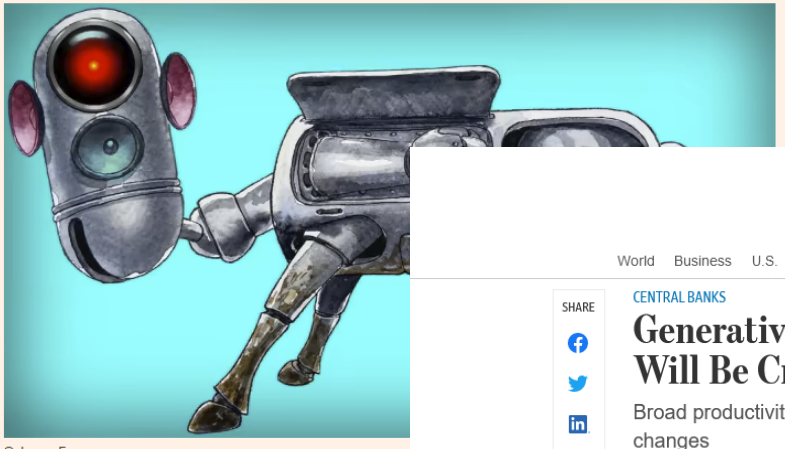
BY RINA DIANE CABALLAR | 03 JUL 2023 | 5 MIN READ

Opinion **Artificial intelligence**

The threat and promise of artificial intelligence

It might be the most transformative technology of all for human beings' sense of themselves

MARTIN WOLF + Add to myFT



© James Ferguson

The Economist

Menu | Weekly edition | The world in brief | Search

Podcasts | Babbage

Is GPT-4 the dawn of true artificial intelligence?

Our podcast on science and technology. This week, we delve under the hood of GPT-4 to figure out how revolutionary, and intelligent, this ground-breaking technology really is.

THE WALL STREET JOURNAL.

Subscribe | Sign In

English Edition | Print Edition | Video | Audio | Latest Headlines | More

World | Business | U.S. | Politics | Economy | Tech | Finance | Opinion | Arts & Culture | Lifestyle | Real Estate | Personal Finance | Health | Science | Style | Sports

SHARE

CENTRAL BANKS

Generative AI Promises an Economic Revolution. Managing the Disruption Will Be Crucial.

Broad productivity and economic output gains may be coming, but knowledge workers will face a reckoning as the nature of work changes

IEEE Spectrum FOR THE TECHNOLOGY INSIDER

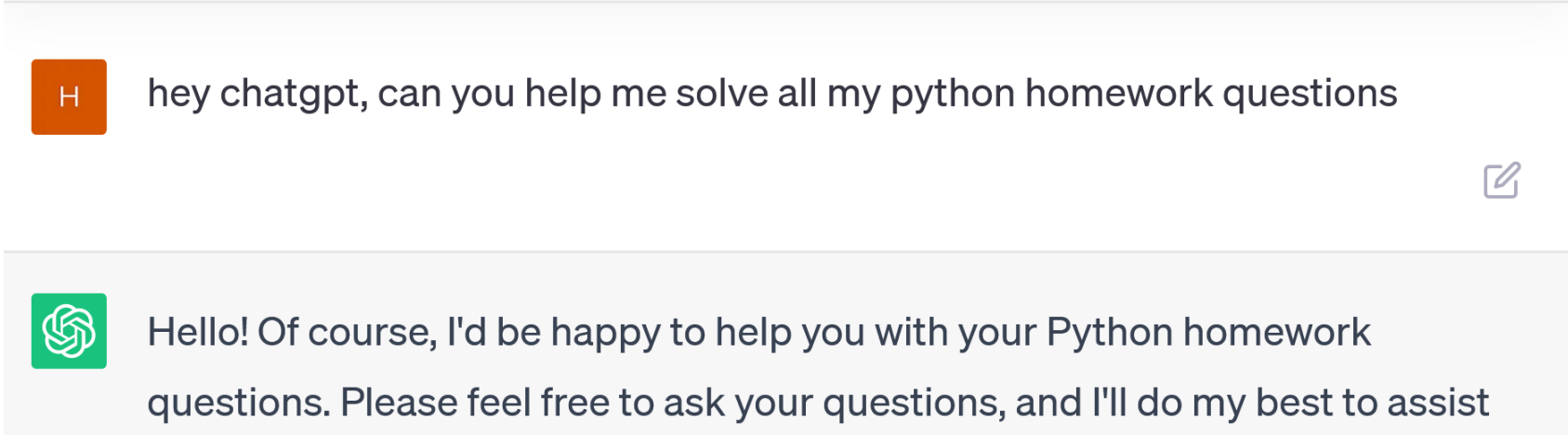
Type to search

NEWS | COMPUTING

How Coders Can Survive—and Thrive—in a ChatGPT World ›4 tips for programmers to stay ahead of generative AI

BY RINA DIANE CABALLAR | 03 JUL 2023 | 5 MIN READ

Productivity multiplier for programming?



We want to help you learn programming:

- Your homework submissions should be **your own unaided work**
- This means no copying code from any source
- Tools like GPT can handle many intro-level Python exercises
- If you ask it (or anyone) for a solution, you lose the learning opportunity
- Once you learn the basics, tools like GPT can have a **multiplier** effect on your productivity
- There is also research showing that using ChatGPT as part of coding is less satisfying (and more productive) than coding without AI. We aren't measuring productivity here–so enjoy it!

My Take on ChatGPT and Learning

- ChatGPT can be a useful resource for being more productive and quickly getting things done.
- But . . . you are here to learn, not get things done ASAP
- Ultimately, coding/algorithmic design is about training you to think clearly about a given situation and design a solution.
- Think about chess and computers: after Deep Blue beat Garry Kasparov, human+computer was better than computer alone for a while. But now, a computer alone (Stockfish) is better.
- If it can be repeated, it can be automated: you need to learn how to do new things. That means understanding the basics
- People sometimes say that writing is a way of forcing us to think, by analyzing, reflecting and reasoning. I think coding is the same thing. Do not let technology do the thinking for you.
- Incidentally: a good article on OpenAI
  `bloomberg.com/features/2025-sam-altman-interview`

This Take is Backed Up By Research!

Experiment with teaching 1000 high-school students in Turkey maths.

- Group 1: access to GPT-4 when studying
- Group 2: a custom version of GPT-4 with guardrails
- Group 3: control–only access to textbooks.

All three groups took an exam. Group 1 did 17% worse than control, and group 2 did about the same as control. Lesson: GPT-4 can harm your learning, as measured by an exam where you don't have access to GPT-4.

With that being said...

We know that you will use generative AI when coding in the workplace. Therefore, once we have taught you the basics, we will spend Lecture 10 of this module on using Generative AI for coding in Python (new this year).

We teach the basics first, because
AI is only useful if you can verify that the output is correct.

Weekly review tutorial with Marius or Jay

These are **compulsory**:

- We will begin by reviewing the previous lecture's topics
- Then exercises from the second half of the lecture + (sometimes) bonus exercises
- If you finish all exercises, you can leave the tutorial early

How it works (more details on the day):

1. Work on extra exercises OR
2. Get help on the regular lecture exercises
3. Marius/Jay will help with either

Today

1. What this module is about
2. Introductions
3. What we will do
4. A discussion of Chat-GPT and other LLMs
5. **Python exercises**

Python first steps

In this session, we will start working with Python:

- doing calculations
- storing the results in variables
- making decisions based on them

This requires learning the syntax and semantics of the Python language.

```
# "Will it rain today?"
rainfall = 2.3
if rainfall > 0.5:
    print('Bring an umbrella.')
else:
    print('No rain expected, enjoy your day!')
```

What is Python?

Python is a free object-oriented programming language that is widely used in industry:

- Beginner-friendly: Easier to debug than languages like C or C++
- More popular than similarly user-friendly alternatives like Julia or Java (I normally use Julia because it has nice mathematical optimization packages, but Python is better for ML)
- Many useful data science and machine learning libraries were developed natively in Python, e.g., Keras for deep learning.

Now for some instructions on GitHub Codespaces

What next

Short break, then:

1. Work through **Session 1**
   - Open it on Codespaces, as indicated, and work through HTML file `session_1.html` in the folder.
   - Don't worry if you don't finish everything now–it's not for credit (but the next one is).
   - Tutorials are there if you need more help.

Work on Session 1 exercises

Work through the HTML file `session_1.html`.
Remark: occasionally, the HTML file will refer to readings. These readings are *optional* and not examinable.

You can work together with your neighbors or on your own.

We are here to help. Raise your hand if you have a question.