

# Introduction to Python

Arieda Muço

Central European University

- My research focuses on two areas: Political and Development Economics. In my research, I deal with tons of data and (lots of) text data -> programming with Python. That's why this course.
- Introduce yourself. What are your expectations? Why are you here? What kind of data you are currently using or plan to use?

# Plan for this course

- Introduction to Python foundations
- Introduction to foundations of Natural Language Processing – concepts tightly linked to Machine Learning and Artificial Intelligence

# Academic Papers in Economics

TABLE 4  
PERCENT DISTRIBUTIONS OF METHODOLOGY OF PUBLISHED ARTICLES, 1963–2011\*

Year	Type of study				
	Theory	Theory with simulation	Empirical: borrowed data	Empirical: own data	Experiment
1963	50.7	1.5	39.1	8.7	0
1973	54.6	4.2	37.0	4.2	0
1983	57.6	4.0	35.2	2.4	0.8
1993	32.4	7.3	47.8	8.8	3.7
2003	28.9	11.1	38.5	17.8	3.7
2011	19.1	8.8	29.9	34.0	8.2

\* A type could not be assigned to seventeen of the articles published in 1963.

Hammermesh (2013)

# Background

- Old data, structured and small: (gdp, population, investment)
- New data, less structure and larger (scraped data, consumer search patterns, social networks, texts, images, audio, video...)
- New methods needed: data collection/management, workflow/collaboration, description/analysis

# Causal Inference and Machine Learning

- Causal Inference
  - ▶ Focus on one/few coefficients of interest (causal effect)
  - ▶ Use one main specification, show robustness to alternative specification and placebo tests
  - ▶ Model rarely evaluated (when pure inference we focus on in-sample-properties, mostly  $R^2$ )
- Machine Learning (ML)
  - ▶ Focus on prediction (and description)
  - ▶ Use data-driven model selection to have best prediction (treated as a black box)
  - ▶ Model is evaluated out-of-sample (e.g. cross-validation)

Use ML to identify the most meaningful predictive variables (i.e Lasso and Ridge), dimensionality reduction, generate outcome of interest  $Y$ , or/and main variable of interest  $X$

# Linguistic differences

	Econometrics	Machine Learning
$Y$	Outcome	Target
$X$	Independent Variables	Features

Note that Scikit-learn and books from the crowd refer to observations as "Samples". Confusing!

# Supervised vs Unsupervised Learning

- Supervised Learning:  $Y$ , the target, is available. Labeled data
  - ▶ Regression:  $Y$  is continuous
  - ▶ Classification:  $Y$  is categorical (binary or multi-class – ordered or not ordered)
- Unsupervised Learning:  $Y$  is not available
  - ▶ Exploratory data analysis and can be useful as a pre-processing step for supervised learning



# Other types of learning

- Deep Learning
- Semi-Supervised
- Active Learning
- Forecasting

# Know Your Task

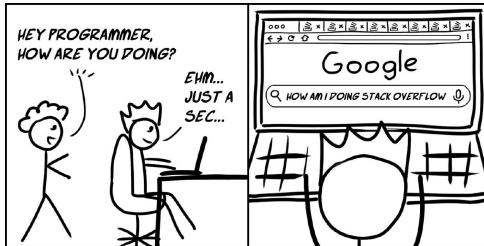
- Each algorithm is different in terms of what kind of data and what problem setting it works best for. When building an algorithm ask:
  - ▶ What question(s) am I trying to answer? Do I think the data collected can answer that question?
  - ▶ What is the best way to phrase my question(s) as a machine learning problem?
  - ▶ Have I collected enough data to represent the problem I want to solve?
  - ▶ What features of the data did I extract, and will these enable the right predictions?
  - ▶ How will I measure success in my application?
  - ▶ How will the machine learning solution will help my project?

# Know Your Data

- The most important task when working with data is knowing your data
  - ▶ All data related work
  - ▶ Extract features only if you know your data well enough

# Rules

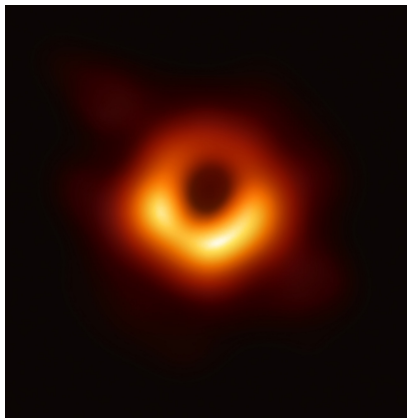
- Ask questions and feel free to Google or use Chat-GPT
  - ▶ Even software developers spend a lot of their coding time googling programming related questions. Don't feel bad about this
  - ▶ Important to know how to read error messages
    - ★ or google them
  - ▶ Stack Overflow was a programmer's best friend. Chat-GPT is now your programming buddy and personal assistant



# A bit about Python

- Programming language intended for general-purpose high-level language
- Web development, scientific and numeric education, desktop graphical user interface, software development
- Free and open source
- You can do everything that you can do in a programming language
- Big community (Google, Youtube, Nasa...)
- High readability (more than R or C)
- Python was first released in early 1980
  - ▶ Python 2 in 2000 and Python 3 in 2008

# Black Holes and Python



# Annoying things in Python

- Python 3 is not backward compatible with Python 2
  - ▶ In this course we will use Python 3. Python 2 is not supported anymore
  - ▶ If you are starting a new project, do so in Python 3
- Pandas Library (more on this next time)
  - ▶ But very useful
- + some minor things we'll cover throughout the course
  - ▶ example: `split()` vs `join()`
    - ★ `sentence = "We will rock you!"`
    - ★ `words = sentence.split(" ")` but `sentence = " ".join(words)`  
(?)



# Purpose of the workshop

- Programming in Python, Natural Language Processing and Machine Learning tools are (mildly put) very broad topics, and we will not be able to cover many(!) things
- Foundations concepts such that in the future you get confidence in starting to dig deeper into these topics

# Recommended Material

- Python
  - ▶ [Codecademy](#) is the place to start
  - ▶ [Automate the Boring Stuff with Python](#) and [The Real Python](#) are great sources
- Machine Learning
  - ▶ An Introduction to Statistical Learning (ISL) by Gareth, Witten, Hastie and Tibshirani
  - ▶ The Elements of Statistical Learning (ESL) by Hastie, Tibshirani, Friedman
  - ▶ Statistical Learning with Sparsity (SLS) by Hastie, Tibshirani, Wainwright
  - ▶ Introduction to Machine Learning with Python: A Guide for Data Scientists (IMLP) by Sarah Guido, and Andreas Muller
- Natural Language Processing
  - ▶ [Introduction to Information Retrieval](#) by Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze
  - ▶ Speech and Language Processing by Dan Jurafsky and James H. Martin