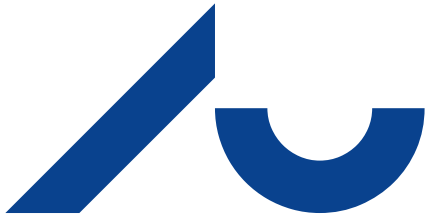

AARHUS
UNIVERSITY

Class 1: Introduction + Python

Computational Analysis of Text, Audio, and Images, Fall 2023

Aarhus University

Mathias Rask (mathiasrask@ps.au.dk)

Aarhus University

Today's Menu

Lecture

- "Hello World!"
- Course Overview
- Computational Methods and Political Science
- Python

Coding Tutorial

Lab

Lecture

Table of Contents

Me:

- PhD student
- trained political scientists with a special focus on machine learning and 'data science'
- Research focuses on the behavior of political elites ($\approx$ politicians) using unstructured data sources, inparticular text and audio
- Lives with my wife and daughter in Copenhagen, a sports enthusiast

You:

- name
- favorite course so far
- level of programming
- expectation(s)

Table of Contents

Organization

- Five ‘main topics’
- Within each ‘main topic’, two or three ‘topics’
- One ‘topic’ ≈ One ‘class’
- Within each ‘topic’, one or two ‘subtopics’

Class	Date	Main Topic		Topic
1	September 6	Python	I	Introduction to course + Python
2	September 7		II	Python lab session
3	September 13	ML Basics	I	Learning from data
4	September 20		II	ML lab session
5	September 27		III	Neural nets
6	October 4	Text	I	Text basics
7	October 11		II	Topic models and dictionaries
8	October 25		III	Embeddings
9	November 1	Audio	I	Audio basics
10	November 1		II	Audio measurement
11	November 8		III	Speech and speaker recognition
12	November 15	Images	I	Image basics
13	November 22		II	Object detection and face recognition
14	November 29		III	Image classification

* Wednesday, October 18 does not feature any class due to the Autumn break in week 42.

Classes

Time: Wednesdays 09 – 12

Location: 1330-038

Structure:

- 09.15-10.00: Lecture
- 10.00-10.15: Break
- 10.15-10.45: Coding Tutorial
- 10.45-11.00: Break
- 11.00-12.00: Lab

Material: `https://github.com/mraskj/css_fall2023`

Classes

09.15-10.00: Lecture

- Cover the most relevant parts from the readings – a mixture of methods and applications – sometimes theory-heavy, sometimes not.
- *Try* to balance theory, measurement validation, and implementation
- Usually 45 minutes, sometimes an hour.
- Slides are available as:
  - **`class01-slides.pdf`**

10.15-10.45: Coding Tutorial

- Class demonstration of how to implement code snippets and methods
- Tutorials are available as:
  - **`class01-tutorial.ipynb`**

11.00-12.00: Lab

- Hands-on exercises *in* class
- I'll be available throughout the lab session to help you if needed –

Exam

- 7-day take-home exam
- December 12 at 09.00 to December 19 at ??
- Cover topics from all five ‘main topics’ with various weights.
- Likely to feature one or two papers, which you should engage with

Table of Contents

Today's Readings

* Y. Theocharis and A. Jungherr, "Computational social science and the study of political communication," *Political Communication*, vol. 38, no. 1-2, pp. 1–22, 2021
* J. Grimmer and B. M. Stewart, "Text as data: The promise and pitfalls of automatic content analysis methods for political texts," *Political analysis*, vol. 21, no. 3, pp. 267–297, 2013
* J. Y. Kim and Y. M. M. Ng, "Teaching computational social science for all," *PS: Political Science & Politics*, vol. 55, no. 3, pp. 605–609, 2022

Main takeaway(s):

1. Interpret measures with care (Grimmer and Stewart, 2013)
2. Integrate measures with theory (Theocharis and Jungherr, 2021)

Defining the Scope of Computational Methods

Question: What is "computational methods and analysis"?

- how does it relate to the *techniques* and *tools* you learned in Methods 1 and 2? (e.g. regression analysis)
- how does data such as *text*, *audio*, and *images* differ from the data used in Methods 1 and 2?

→ Talk to your neighbor for 4-5 minutes

Take-aways: What is "computational methods and analysis"?

- very loosely defined...
  - → a broad definition also contains standard regression analysis
- "the degree to which research projects demand the inclusion and development of computational methods over the course of a project" (Theocharis and Jungherr, 2021, p. 5)
  - → implicitly includes the amount of data into definition
  - → question to you: should we also include the type of data?

Principles of Computational Analysis

Grimmer and Stewart (2013) put forward four principles:

1. All models are wrong
2. Computational analysis amplifies human capabilities
3. Different tasks require different tools
4. Validation!

I'll add a few to the list:

5. Computational analysis is dimensionality reduction
6. Garbage in, garbage out

The “Big Picture’

Kim and Ng (2022): The ”big picture” is crucial to learn computational methods and analysis effectively.

What’s the ”big picture” in this course?

- *Not* learning how neural networks update their weights using backpropagation
- *Not* learning how to convert (audio) signals from the time to frequency domain
- *Instead*: learning how to derive and measure **political** relevant behavior and concepts
- *Instead*: learning how to process **unstructured** data into a format that we can analyze using simple tools (e.g. regression models)

Table of Contents

What is Python?

Python is an open-source, general-purpose scripting language.

- *Open-source:* Free to use – maintained by a community
- *General-purpose:* Very flexible! Used by a variety of industries to automate procedures – very popular for data science
- *Scripting:* A series of commands that produce something – like a recipe.
- *Language:* Python is a language, not an application!
    - No drop-down menu to choose from
    - Requires to write commands within the language's accepted rules

How does Python work?

Three keywords: *Libraries*, *Versions*, and *Environments*

- *Libraries:* Base Python + thousands of wonderful libraries (e.g. Pandas)
- *Versions:* Multiple Python versions – some libraries require specific versions
- *Environments:* A way to keep dependencies required by different projects separate by creating isolated Python environments for them

Writing Python Code

Python has multiple code editors:

- PyCharm (popular to write packages)
- VS Code
- Text editors (Atom, VIM)
- Jupyter notebooks (interactive code editor)
- Google Colab
- ...

→ We will work with notebooks either locally or using Colab

Python vs. R

Table 1: Comparison of Python and R

Task	Python	R
General Purpose Programming	Great	OK
Causal Inference	Poor	Great
Web Scraping	Great	OK
Data Visualisation	OK	Great
Machine Learning	Great	Great
Natural Language Processing	Great	Great
Computer Vision	Great	Poor
Audio	Great	Poor
Industry Demand	Great	OK

Coding Tutorial

Virtual Environments

Package dependencies will cause you trouble at some point. To avoid it, we create virtual environments, which keep dependencies for certain packages isolated to that specific environment.

Open the terminal and create an environment:

```
conda create --name NAME-OF-ENVIRONMENT python=PYTHON-VERSION

conda create --name css-fall2023-topic1 python=3.10
```

Commands I

See a list of conda environments you have:

```
conda env list
```

To activate an environment:

```
conda activate css-fall2023-topic1
```

Verify version of Python is correct:

```
python --version
```

See installed packages:

```
pip list
```

Commands II

Deactivate an environment:

```
conda deactivate
```

Remove an environment

```
conda env remove --name css-fall2023-topic1
```

Commands III

Install the latest version of the package

Pip: `pip install PACKAGE`

Conda: `conda install -c conda-forge PACKAGE`

Install a specific version of a package

Pip: `pip install PACKAGE==1.4.3`

Conda: `conda install PACKAGE=1.4.3`

Lab

Jupyter Notebook

Google Colab

References

[1] Y. Theocharis and A. Jungherr, "Computational social science and the study of political communication," *Political Communication*, vol. 38, no. 1-2, pp. 1–22, 2021.

[2] J. Grimmer and B. M. Stewart, "Text as data: The promise and pitfalls of automatic content analysis methods for political texts," *Political analysis*, vol. 21, no. 3, pp. 267–297, 2013.

[3] J. Y. Kim and Y. M. M. Ng, "Teaching computational social science for all," *PS: Political Science & Politics*, vol. 55, no. 3, pp. 605–609, 2022.