Lecture 1 - Introduction & Python Setup

Management Science

Dr. Tobias Vlćek

About this Course

About me

- Field: Optimizing and simulating complex systems
- Languages: of choice: Julia, Python and Rust
- Interest: Modelling, Simulations, Machine Learning
- Teaching: OR, Algorithms, and Programming
- Contact: vlcek@beyondsimulations.com

. . .

> Tip
>
> I really appreciate active participation and interaction!

What is

Management

Science?

Photo by Patrick Konior on Unsplash

Management Science

Management science is an interdisciplinary field that applies research-based methods, such as modeling, statistics, and algorithms, to solve complex organizational problems and guide strategic decisions, seeking optimal or near-optimal outcomes across institutions, corporations, and governments.[1]

. . .

It is not the science of management!

[1] An Introduction to Management Science: Quantitative Approaches to Decision Making (15 ed.). Boston: Cengage Learning, Inc. 2019. ISBN 978-1-337-40652-9.

Course Outline

- Part I: Python Foundation (Lectures 1-3)
  - Variables, loops, functions, data science tools
- Part II: Management Science Tools (Lectures 4-9)
  - Monte Carlo, Scheduling, Routing, Metaheuristics
- Part III: Consulting Competition (Lectures 10-12)
  - Client briefings, development, final presentations

. . .

> **!** Important
>
> You're now consultants learning to solve real business problems!

From Amazon to Hospitals

Where Algorithms Make Millions

- Amazon: Routing algorithms save millions in delivery costs
- Uber: Real-time matching algorithms connect drivers and riders
- Hospitals: Scheduling algorithms optimize shifts and OR usage
- Airlines: Revenue algorithms price tickets dynamically
- Manufacturing: Production scheduling reduces waste and delays

. . .

Management Science is everywhere!

This Course Approach

The Consultant Mindset

- You'll work on business problems
- Learn to communicate solutions to non-technical clients
- Build practical tools that solve actual challenges
- Compete in a consulting competition with real scenarios

. . .

> Tip
>
> Think like a consultant: What value does my solution bring to the business?

Course Structure

- 12 lectures of intensive learning
- 3 hours per lecture (with breaks!)
- Interactive format:
  - Concepts & Setup

    - Hands-on Python notebooks
    - Consulting Competition
- Final competition: Apply everything you've learned

. . .

Each lecture builds on the previous - attendance is important!

Grading

Grade Composition:

- Course has 100 points, 50 needed to pass
- 2 Assignments: 30 points each
- Final Consulting Project: 40 points

. . .

> ! Important
>
> We will have several competitions including our final competition, where you will have the chance to earn bonus points!

Assignments

Assignment Strategy:

- Start with easy warm-ups in tutorials
- Progress to real-world problems
- All assignments have clear rubrics
- Bonus points for creative solutions!

. . .

> ⚠ Warning
>
> Late submissions: –10% per day (max 3 days)

Learning Python

- In my experience, the best way to learn is by doing!
- Here, we will focus on decision algorithms
- You will start to learn Python by doing the tutorials

. . .

> Tip
>
> Don't worry, I will help you out if you have any questions!

What to Expect

- No prior programming required - we start from zero!
- But experience is helpful!
- Fast-paced but with lots of support
- Practical focus - less theory, more doing
- Mistakes are welcome - they're how we learn

. . .

> Tip
>
> The course gets easier after week 3 - the hardest part is getting started!

Course Goals

By the end, you will be able to:

1. Write Python code to solve business problems
2. Apply algorithms for scheduling, routing, and optimization
3. Simulate uncertainty using Monte Carlo methods
4. Present solutions like a management consultant
5. Build tools that create business value

. . .

You'll leave with a portfolio of ideas to work on real solutions!

Why Python?

- Origins: Conceived in late 1980s as a teaching and scripting language
- Simple Syntax: Python's syntax is mostly straightforward and very easy to learn
- Versatility: Used in web development, data analysis, artificial intelligence, and more
- Community Support: A large community of users worldwide and extensive documentation

Help from AI

- You are allowed to use AI in the course
- I use it as well (e.g., Claude, ChatGPT, Gemini, ...)
- These tools are great for learning Python!
- Can help you a lot to get started with programming
- I will also teach you how to use it effectively

. . .

> Warning
>
> But you should not simply use them to replace your learning.

How to learn programming

My Recommendation

1. Be present: Attend the lecture and solve the tutorials
2. Put in work: Repeat code and try to understand it
3. Do coding: Run code, play around, modify, and solve
4. Compete: Participate in the competitions to learn

. . .

> Tip
>
> Great resources to start are books and small challenges. You can find a list of recommendations in the literature recommendations.

Don't give up!

Programming is like learning to ride a bike

. . .

- You'll fall a few times
- It feels impossible at first
- Then suddenly... it clicks!

. . .

> ! Important
>
> Lectures 1-3 are the hardest. Push through and it gets much easier!

Setting up Python

What is an IDE?

- Integrated Development Environment = application
- It allows you to write, run and debug code scripts
- Other IDEs include for example:
  - PyCharm from JetBrains
  - Zed

Install VS Code

- Download and install from the website
- Built for Windows, Linux and Mac
- Install the Python and Jupyter extension
- Great! First steps are done

. . .

> Tip
>
> Unsure on how to work with VS Code and notebooks? Ask me! I'm happy to help you out!

Installation of Python with `uv`

- We will use `uv` to install and manage Python versions
- It works on Windows, Mac and Linux
- It helps us to manage packages and virtual environments
- Now, we all go here and install `uv` and Python

. . .

> Tip
>
> If the installation does not work, let me know!

Notebooks with uv

Quick Check

- Have you installed `uv` and initialized the project?
- Great! Before we continue, check the following:
  - ⍰ You have a folder for the course
  - ⍰ You have initialized `uv` with `uv init` inside the folder
  - ⍰ You can see a file called `pyproject.toml` in the folder

. . .

> Warning
>
> Something not working yet? Ask me!

Using Notebooks

- Now we need to add a kernel to our project
- Run `uv add --dev ipykernel` from your terminal
- Now run `uv add jupyter` in the terminal
- This allows us to use `uv` Python in notebooks
- Next, install the extension Jupytext for Notebooks
- Done? Perfect. Now we can start!

Working with Notebooks

- Now you can download the files from the website
- Just click the first session with code and open it
- Select `Python` on the right side
- Download and save the files to your course folder
- Open them and select “Open with Jupyter Notebook”

. . .

> **Tip**
>
> That was the hardest part today!

Any questions

so far?

What’s Next?

After the break, we’ll dive into:

- Tutorial 1: Variables and basic data types
- Tutorial 2: Lists and loops
- Tutorial 3: Conditionals and control flow

. . .

> **Note**
>
> Ready for the tutorials? Make sure your Jupyter notebook is working before we continue!

The End

That’s it for our introduction!

Make sure you have:

- ? VS Code installed and running
- ? Python environment set up with `uv`
- ? Jupyter notebooks working
- ? The tutorial files downloaded

. . .

> **Tip**
>
> Take a 10-minute break, then we'll start with the interactive notebooks.

Literature

Interesting Literature on Algorithms

- Christian, B., & Griffiths, T. (2016). Algorithms to live by: the computer science of human decisions. First international edition. New York, Henry Holt and Company.[2]
- Ferguson, T.S. (1989) 'Who solved the secretary problem?', Statistical Science, 4(3). doi:10.1214/ss/1177012493.

Books on Programming

- Downey, A. B. (2024). Think Python: How to think like a computer scientist (Third edition). O'Reilly. Here
- Elter, S. (2021). Schrödinger programmiert Python: Das etwas andere Fachbuch (1. Auflage). Rheinwerk Verlag.

...

> **Note**
>
> Think Python is a great book to start with. It's available online for free. Schrödinger Programmiert Python is a great alternative for German students, as it is a very playful introduction to programming with lots of examples.

More Literature

For more interesting literature, take a look at the literature list of this course.

Bibliography

[2] A great inspiration to learn more about Algorithms!