# Welcome to DS701

# Course overview

Welcome to Fall 2026 DS701 Tools for Data Science.

This course is a Master's level introduction to data science.

In this course you will:

- develop proficiency in working and analyzing data
- work with a wide range of data analysis techniques and tools

# Course Staff and Meeting Times

# Instructor

**Thomas Gardos**

- **Office Hours:** see the Piazza Staff Resource Page
  - and by appointment (email me to schedule)
- **Office Location:** CCDS 1623
- **Email:** tgardos <at> bu <dot> edu

# Course staff

**Teaching Fellow — Farid Karimli**

- Office: CCDS 1425
- Office Hours: see the Piazza Staff Resource Page
- Email: faridkar <at> bu <dot> edu

**Spark! program staff:** TBA

# Course Meeting Times

**Lecture (A1, also DS690 Q1)**

- **Mondays and Wednesdays, 12:20–2:05 PM** (105-minute sessions)
- 665 Commonwealth Avenue (CDS), **Room 263**

**Discussion Sections**

- Weekly 50-minute sections on **Fridays** — times and rooms are on MyBUStudent
- Your time for **project team meetings** and team work
- Midterm and final project **orals happen here** — teammates must share a section

Please attend the discussion section you signed up for.

> Registered in DS690? See me about a discussion-section assignment.

# Learning outcomes

The goal of the class is to:

- give you a hands-on understanding of classical (but still useful!) data analysis techniques (although we cover neural networks too!)
- help you gain proficiency in applying these techniques in a modern programming language (Python)

# We’ll Cover

# You'll be using

- Python
- Jupyter notebooks
- numpy
- pandas
- scikit-learn
- statsmodels
- matplotlib
- scipy
- networkx
- pytorch
- *and others…*

# Course Content and Sites

# Course webpage

Syllabus, lecture schedule, and course notes can be found on this webpage

https://ds701.cds.bu.edu/

This online text will evolve as the course progresses.

*Bookmark this page!*

# Course Notes Source

You have a few ways to follow along with the course lectures/notes:

1. Open the Colab version of the lecture and save to your own Google Drive when you see the see the badge 

.
2. Fork and clone the repository and run the Jupyter notebooks version of the lectures locally.

For lectures with python code, you can experiment with the code yourself.

Each lecture page also links the **slides**, as HTML and as a PDF, from badges at the top — the same deck used in class, for reviewing or annotating.

In general, you'll want to have your laptop computer with you at the lectures to follow along and participate in in-class activities.

# Tools and platforms

We will use:

1. Piazza for questions
2. Github for homeworks, midterm, lectures.
3. Gradescope for grading and grade management

You should already be signed up for Gradescope (if not, enroll using code sent via welcome email).

You can add yourself to Piazza if you are not already enrolled (again, use the code sent via welcome email).

You will need an account on Github. Please tell us your Github user name on this form.

If you don’t have an up to date Python installation, take care of that right away.

# Piazza

We will be using Piazza for class discussion.

You can use Piazza to get help fast and efficiently from classmates, the TAs, and the professors.

I encourage you to post your questions on Piazza.

Our class Piazza page is at

https://piazza.com/bu/fall2026/ds701

# Piazza Etiquette

Please be respectful on Piazza.

**Do**

- Ask questions about course material, logistics, etc.
- Answer posted questions when you know the answer (not homework)
- Tell people where to look for answers

**Don’t**

- Provide solutions to homework questions.

# Programming environment

We will use Python as the language for teaching and for assignments that require coding.

We have a chapter on installing python and a brief recap of some python fundamentals.

If you are don't feel proficient in python, you should review the chapter, or even better, complete a more comprehensive on-line course on python.

Check that you have **at least Python 3.12**, preferably 3.14 or later, and get comfortable with virtual environments and package managers — `venv`, `pip`, `miniconda`. We also use Jupyter notebooks and Google Colab.

# Grading

# Grading

# Quizzes and Optional Final

# Spark! Project Grading

The Spark! group project counts for **20%** of the course grade:

# Homeworks

Weekly **theory homework** prepares you for the quizzes — every problem is quiz-shaped, and each set should take 1–2 hours **without AI**.

Graded for **participation** (a good-faith attempt), not correctness — worked solutions are released after the due date.

**Handwritten on paper**, then scanned and uploaded to Gradescope. The quizzes are handwritten and closed-notes; practice in the same medium.

Ten sets, released Mondays and due the following Monday at 11:59 pm; solutions appear the day after each deadline, and no set is ever due on a quiz day.

> NOTE: Late assignments will be accepted up to 48 hours after the deadline with a 10% penalty.

You are expected to work individually on homeworks.

# Midterm and Final Team Projects

Two projects in **teams of three** (two by exception): an open-ended deeper dive on a dataset, applying techniques covered so far. Separate from Spark!.

# Spark! Group Project

# Spark! Project

Alongside the midterm and final team projects, you will work all semester with **BU Spark!** on a real data-science problem for an external client — a company, non-profit, or institution.

> Two different things, easy to confuse: **Spark! = one client project, all semester.** The **midterm and final team projects** are short, use curated datasets, and are examined orally in your discussion section.

# Spark!: weekly SCRUM

Every team uploads a SCRUM file to the project repository each week — a fast, concise status report answering:

# Project Expectations

# Spark! project teams

All Spark! project teams have the following structure:

# Spark! Collaboration

BU Spark! offers students an opportunity to work on technical projects provided by companies or organizations in the Greater Boston area through its experiential learning lab (X-Lab). Spark! has partnered with DS701 to offer a selection of external data science projects scoped to support the course's learning outcomes.

Learn more at bu.edu/spark.

# Participation and In-Class Activities

# Flipped Classroom Format

For each content session:

# Participation and Cold Calls

**Cold calls** keep you engaged and give you practice “thinking on your feet”: random selection, during both the recap and the activity.

**In-Class Activities** are small-group challenges reinforcing the day’s lecture. Autograded parts count for correctness; open-ended parts for participation. You may use AI — but be ready to justify your solution and demonstrate comprehension when staff check in.

# Thoughts on GenAI

GenAI is a powerful tool that can help you learn and be more productive and undoubtedly will be part of your workflow going forward. We will use it!

But, would you…

Figure 1: Via ChatGPT – *And yes, I'm aware of the irony.*

# Policies

# Generative AI Assistance (GAIA) Policy

We're in a unique period of time where AI tools are advancing rapidly and becoming indispensable to how we work. In fact proficiency with these tools is already expected of new-hires and you are one of the first to go through a degree program as "AI natives." We highly encourage you to make full use of AI assistive tools for:

- getting clarifications and explanations while watching pre-recorded lectures
- getting assistance in working on in-class activities and projects

But you will need to be able defend the output of AI tools and why you think it is valid, explain ways you can verify it, and how you might improve it.

On the other hand, we also want to make sure you are learning the foundational concepts and skills that will be expected of you in your future career. For foundational concepts, we want you to practice in the homeworks with minimal AI assistance and then demonstrate your understanding in the paper-and-pencil quizzes. Similarly, that is why we are cold calling during lecture recaps and in-class activities, and probing with oral examinations of the projects.

# Academic honesty

You may collaborate and discuss homework assignments with classmates, but you are solely responsible for what you turn in.

All forms of cheating (copying parts of a classmate's assignment, plagiarism from books or old posted solutions) are NOT allowed.

We – both teaching staff and students – are expected to abide by the guidelines and rules of the Academic Code of Conduct.

Graduate students must also be aware of and abide by the GRS Academic Conduct code.

# Before next class

Our next meeting is **Wednesday, September 9** — there is no class on Labor Day.

- We will have Spark! project pitches and open up the project preferences survey.
- Watch the first pre-recorded lecture and complete its knowledge check **before** class — this is the flipped format starting for real.
- Bring your laptop and have python installed as well as VS Code and/or Cursor.

And fill out the intro survey and indicate so in Gradescope.

# In-Class Activity

Make sure you are set up with the following:

- [ ] Python 3.12 or later installed
- [ ] Virtual Environment and package management such as
  - `venv` and `pip`
  - `miniconda`
  - `uv`
- [ ] VS Code with Copilot, Claude Code, Codex or similar
- [ ] Jupyter environment and VS Code Jupyter extension
- [ ] git and GitHub, preferable GitHub Desktop and authenticated

Ensure you have all these. Ask us for help or come to office hours.

When you are done, indicate so on Gradescope.