Computational Modeling, Statistical Analysis and Machine Learning in Science

38-615 & 09-615, Fall 2022

Olexandr (or Oles for short) Isayev
Department of Chemistry, CMU
olexandr@cmu.edu

Outline

- Course logistics
- Course objectives
- Course overview
- Anaconda Python& Jupyter notebook ecosystem

Intro!

- Introduce yourself in a few sentences.
- What do you expect from this course?
- Tell one funny fact about yourself!

Class schedule

Tue & Thu 10:10- 11:30 AM

Canvas: https://canvas.cmu.edu/courses/31028

Slack or Discord channel ?

Piazza ?

Course requirements

Designed for STEM master students & MCS recent graduates

Open to senior–year undergraduate students (with permission)

This is **practical**, application-oriented course, requiring skill in algorithmic problem solving.

We will use **Python** based tools and libraries. Prior programming experience with Python is needed.

Prerequisites: probability, linear algebra, statistical thermodynamics and quantum mechanics and related quantitative courses

Programming experience

ML experience

Jupyter Notebook / Lab

M.S. in Data Analytics for Science (MS-DAS) program

- New program, second year
- Please gave your feedback. olexandr@cmu.edu
- Take advantage of small class size!
- Job market is tough, but we are here to help!

Learning objectives

- Know how to explore and visualize scientific data
- Compare and contrast different types of data and representations.
- Understand core components of data analytics pipeline: visualization, exploratory data analysis, classification, regressions, prediction etc.
- Be able to analyze scientific data using a variety of machine learning approaches.
- Implement and analyze well-known existing ML algorithms.
- Integrate multiple components of practical machine learning in a single system: data preprocessing, learning, regularization, model selection and be familiar with programming tools to accomplish it.
- Hands on experience with real-world cases on how ML could address challenges in STEM sciences.

Course Outline

- Exploratory data analysis and visualization
- Unsupervised learning, clustering, dimensionality reduction
- Supervised learning, model training and evaluation
- Linear and nonlinear models
- Classification, SVM, kernel methods
- Decision trees and RF
- Probabilistic methods

Course Structure

Tuesday

- Lecture materials

Thursday

- Recital/practice
- Tutorials
- Lab discussions

Reading

No textbook

Readings will be provided on Canvas portal and lectures.

The readings for this course are required.

We recommend you read them **before** the lecture.

- Optional topics
- Very useful in practice
- Extend skills

Course Grades

5% for attendance

5% for class participation

50% for Lab assignments

40% for final open-ended class project

Bonus points for Top Kaggle leaderboard score

Final Project

- Work in teams 2-3 people.
- Open-ended project!
- Solve a science related problem with machine learning! Use your domain expertise
- Encourage to use your data
- Jupyter notebook, which mixes together written markdown and code portions or python script and report.
  - ~2000 words (2-3 pages of text)
  - ~500 lines of code
- All text and code must be your own work

Final Project

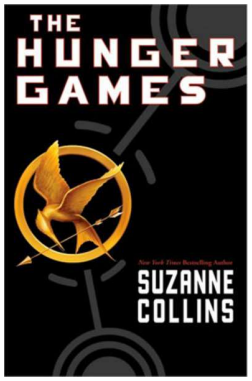
- Submit one paragraph project proposal (September)
- Short project talk (Pitch! 1-2 slides) (~Early Oct)
- Presentation during last week & also project final report
- You will be graded by the course instructor and other students taking the course (peers)

Lab Exercises / Home Assignments

- Lab1: EDA
- Lab2: Clustering
- Lab3: Linear Methods
- Lab4: Classification
- Lab 5: Regression

Lab auto-grading

Brought to you by



- Five Labs per semester
- Short solution discussions after each Lab
- Assignments will include 2 parts: *programming* component and *kaggle* component (autoscoring)
- Sometimes you will compete with each other…

Dashboard | Public Leaderboard - Heritage Health Prize

This leaderboard is calculated on approximately 30% of the test data.
The final results will be based on the other 70%, so the final standings may be different.

#	Δ1w	Team Name * in the money	Score	Entries
1	–	EXL Analytics *	0.443793	555
2	–	POWERDOT	0.447651	671
3	–	Dolphin	0.450403	555
4	↑1	jack3	0.451425	455
5	↓1	Hopkins Biostat	0.451569	444
6	–	Xing Zhao	0.453081	161
7	–	Old Dogs With New Tricks	0.454096	370
8	–	Areté Associates	0.454424	112
9	–	Alice Sasandr	0.454670	376
10	↑9	J.A. Guerrero	0.454728	173

Rough plan for next few weeks

Aug 30: Lecture 0 Intro & Class logistics
Sep 01: Lecture 1 - Data, data types, formats, basic analysis

Sep 06: Lecture 2 - Data Visualization
Sep 06: Lab 1 released
Sep 08: Recital time - time for setup, refresher/sci tutorial?

Sep 13: Lecture 3 - Dimensionality reduction
Sep 15: Recital time

~Sep 19: Lab 1 Due

Sep 20: Lecture 4 – Unsupervised Learning
Sep 22: Lab1 discussions; Lab 2 released

Questions?

Anaconda Python

- For the class, we strongly recommend you use Anaconda Python
- This distribution of Python, includes most libraries and tools

https://www.anaconda.com/download/

Installing additional packages

There are two general ways to install additional packages

```
conda install <package name>

     conda search <package name>
     conda list


pip install <package name>
```

Jupyter notebook

- Notebook documents (are documents produced by the Jupyter Notebook App, which contain both computer code (python) and rich text elements (paragraph, equations, figures, links, etc…).

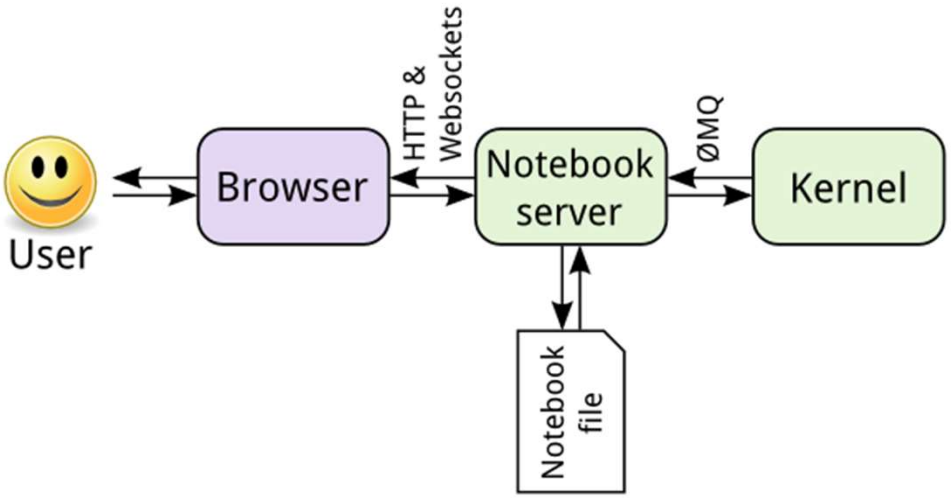
- Notebook documents are both human-readable documents containing the analysis description and the results (figures, tables, etc..) as well as executable documents which can be run to perform data analysis.

- More info about Jupyter here: http://www.jupyter.org

Kernels

Behind every notebook runs a kernel. When you run a code cell, that code is executed within the kernel and any output is returned back to the cell to be displayed.

The kernel's state persists over time and between cells — it pertains to the document as a whole and not individual cells.



Jupyter notebook

Launch jupyter via the command:

```
jupyter notebook
```

Open in browser: http://localhost:8888

New (alternative) environment: `jupyter lab`