

Data Analysis and Machine Learning Education in the Physics Department at the University of Illinois

Mark Neubauer

University of Illinois at Urbana-Champaign





Origins of my DS/ML course in Physics

The idea for developing a course in data analysis and ML applications for our undergraduates grew out of townhall discussions ~6 years ago in our department

- At the time, there were no such courses in our Physics department (now there are several)
- Data science was on the rise, especially driven by the revolution around machine learning
- Students recognize the high value of training at the intersection of data science, AI and physics

I pitched this new course in 2017 at our PAB meeting & taught in Fall 2018, Spring 2019, Fall 2019, Spring 2022





PHY 398MLA

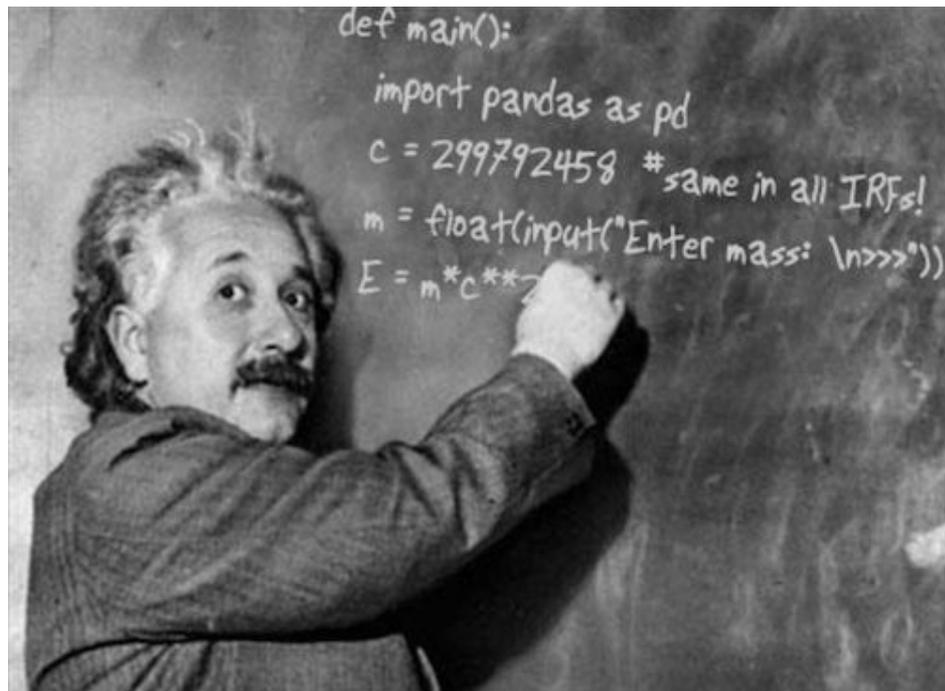
Data Analysis & ML Applications for Physicists

Topical focus:

- Techniques for analysis and interpretation of scientific data
- Machine learning principles and applications to physics

Technical approach:

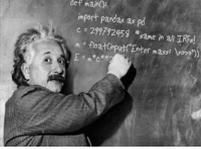
- Open everything
- Minimize coding focus: Data Science to advance Physics is the goal





PHY 398MLA

Course website: <http://illinois-mla.github.io/syllabus>



Data Analysis & ML Applications for Physicists

Motivation

- ❖ We live in a data-centric world, with people and machines learning from vast amounts of data.
- ❖ Early-career physicists need a solid understanding in the basics of data analysis, data-driven inference and machine learning, and a working knowledge of modern tools and techniques from data science

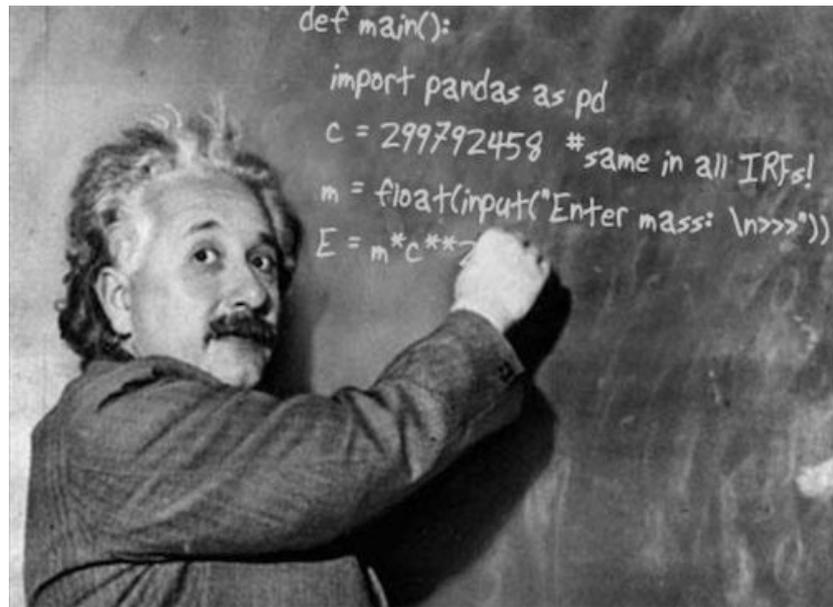


PHY 398MLA

Data Analysis & ML Applications for

Why do ~~Physicists~~ Physicists?

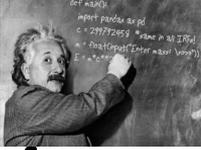
- Data science is an exploding area both academia and industry
 - E.g. we find that many problems in physics map to vision problems amenable to automation via ML methods
- Many of the future (and current) jobs will be in this area
- Physics students need a solid foundation in data analysis & interpretation to do research. They need this education and training to be productive in science and thrive as the next generation of leaders





PHY 398MLA

Course website: <http://illinois-mla.github.io/syllabus>



Data Analysis & ML Applications for Physicists

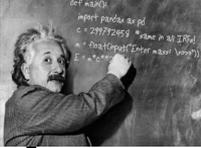
Structure

- ❖ Two credit-hour course
- ❖ One 2 hour lecture each week
 - Attendance and class participation are mandatory
- ❖ 8 Homework assignments
 - New problems posted ~weekly, due within 1 week
- ❖ A final project based on analysis of open science data



PHY 398MLA

Course website: <http://illinois-mla.github.io/syllabus>



Data Analysis & ML Applications for Physicists

Pedagogy in Motion

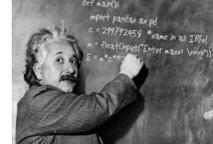
- ❖ At the center of the course is *scientific data*
 - Students learn how to analyze it, simulate it, gain insight from it, and get machines to learn from it
- ❖ Students acquire a working knowledge of **DS tools**
 - All lecture materials, homework and projects are **python-based** within in **Jupyter notebooks** (.ipynb)
 - Materials are managed within a **Github organization**
 - Students submit homework (.ipynb) to their private Github repo. Homework is managed by **nbgrader**
 - In-class time is a mixture of **collective listening** and **active learning** with *in-situ* coding exercises



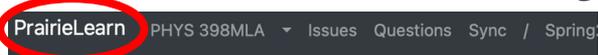


PHY 398MLA

Course website: <http://illinois-mla.github.io/syllabus>



Data Analysis & ML Applications for Physicists



PrairieLearn is an online problem-driven learning system for creating homeworks and tests

- Developed by the University of Illinois in 2014

Assessments		
		AID
Lecture Activities		
L1	Lecture 01	lectures/lecture01
L2	Lecture 02	lectures/lecture02
L3	Lecture 03	lectures/lecture03
L4	Lecture 04	lectures/lecture04
L5	Lecture 05	lectures/lecture05
L6	Lecture 06	lectures/lecture06
L7	Lecture 07	lectures/lecture07
L8	Lecture 08	lectures/lecture08
L9	Lecture 09	lectures/lecture09
L10	Lecture 10	lectures/lecture10
L11	Lecture 11	lectures/lecture11
L12	Lecture 12	lectures/lecture12
L13	Lecture 13	lectures/lecture13
L14	Lecture 14	lectures/lecture14

Course Projects

P1 Final Project projects/FinalProject

Homeworks

HW1	Homework 01	homework/homework01
HW2	Homework 02	homework/homework02
HW3	Homework 03	homework/homework03
HW4	Homework 04	homework/homework04
HW5	Homework 05	homework/homework05
HW6	Homework 06	homework/homework06
HW7	Homework 07	homework/homework07
HW8	Homework 08	homework/homework08





PHY 398MLA

Course website: <http://illinois-mla.github.io/syllabus>



Data Analysis & ML Applications for Physicists

PrairieLearn Workspaces

allow students to work in persistent remote containers via in-browser frontends such as VS Code and JupyterLab.

Workspace questions are integrated with the standard PrairieLearn autograding pipeline.

The remote containers are configured by instructors to provide [custom, uniform environments per question](#).

PL Workspace: homework/homework08 **RUNNING**

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
Homework...	4 minutes ago

Launcher Homework8.ipynb

Homework 8

```
[ ]: %matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns; sns.set()
import numpy as np
import pandas as pd

[ ]: from sklearn import model_selection, ensemble

[ ]: import autograd.numpy as anp
import autograd

[ ]: from mls import locate_data, cv_summary
```

Problem 1

The default score function used by sklearn to evaluate how well a regression model

```
[ ]: def calculate_R2(y_data, y_pred):
    """Calculate the coefficient of determination R2 for two arrays.

    Parameters
    -----
    y_data : array
        Array of data values, must have the same shape as y_pred.
    y_pred : array
        Array of predicted values, must have the same shape as y_data.

    Returns
    -----
    float
        Calculated coefficient of determination R2.
    """
    assert y_data.shape == y_pred.shape
    # YOUR CODE HERE
    raise NotImplementedError()
```





PHY 398MLA

Course website: <http://illinois-mla.github.io/syllabus>

Docker container:

<https://github.com/illinois-mla/phys-398-mla-image>



Data Analysis & ML Applications for Physicists

illinois-mla / phys-398-mla-image Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

matthewfeickert docs: Add developer instructions (#19) ✓ d4daf34 9 days ago 18 commits

.github/workflows	ci: Run test builds on all but push events (#6)	5 months ago
docker	feat: Add tensorflow probability and edward2 to the environment (#18)	2 months ago
LICENSE	feat: Add Dockerfile and CI (#1)	5 months ago
Makefile	fix: Add mIs library from GitHub (#8)	5 months ago
README.md	docs: Add example use instructions (#5)	5 months ago
development.md	docs: Add developer instructions (#19)	9 days ago
run_jupyter_standalone.sh	docs: Add example use instructions (#5)	5 months ago

README.md

PHYS 398 MLA Docker Image

Docker image for the PHYS 398 MLA used to power the PrairieLearn workspaces.

Docker Images passing docker pulls 436 image size 2.3 GB

Installation

- Check the [list of available tags on Docker Hub](#) to find the tag you want.
- Use `docker pull` to pull down the image corresponding to the tag. For example:

```
docker pull physicsillinois/phys-398-mla:latest
```

About

PHYS 398 MLS Docker Image

Readme MIT license 0 stars 2 watching 1 fork

Releases

No releases published [Create a new release](#)

Packages 1

phys-398-mla-image

Contributors 2

matthewfeickert Matthew Feickert
msneubauer Mark Neubauer

Languages

Dockerfile 61.9% Shell 19.9%
Python 11.6% Makefile 6.6%





PHY 398MLA

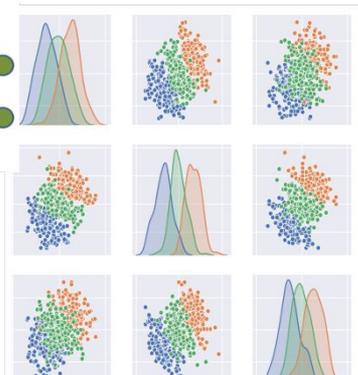
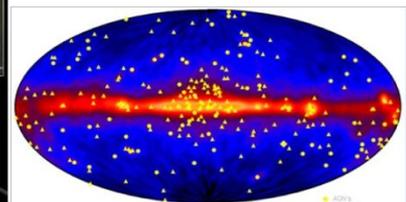
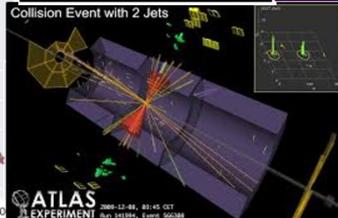
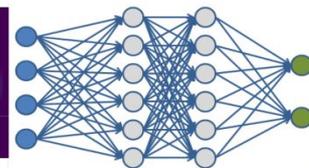
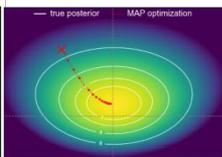
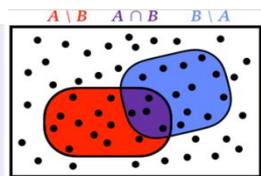
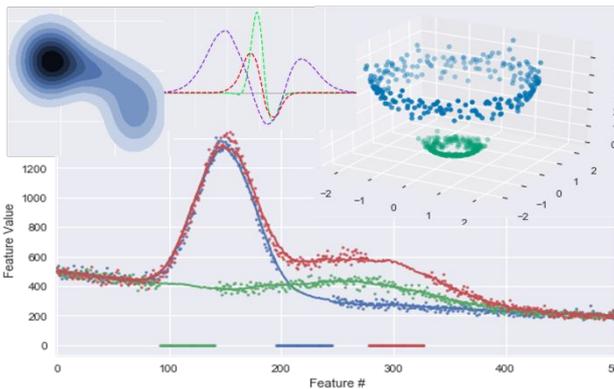
Course website: <http://illinois-mla.github.io/syllabus>



Data Analysis & ML Applications for Physicists

Topics

- 1) Handling and Visualizing Data
- 2) Finding structure in data
- 3) Measuring and reducing dimensionality
- 4) Adapting linear methods to nonlinear problems
- 5) Estimating probability density
- 6) Probability theory
- 7) Statistical methods
- 8) Bayesian statistics
- 9) Markov-chain Monte Carlo in practice
- 10) Stochastic processes and Markov-chain theory
- 11) Variational inference
- 12) Optimization
- 13) Computational graphs
- 14) Probabilistic programming
- 15) Bayesian model selection
- 16) Learning in a probabilistic context
- 17) Supervised learning in Scikit-Learn
- 18) Cross validation
- 19) Neural networks
- 20) Deep learning





PHY 398MLA

Data Analysis & ML Applications for Physicists

Some challenges:

- Coding is new for many students. Many are unfamiliar with Python, Git, Jupyter notebooks, etc ... → provide ample resources/examples
- Challenge to keep notebooks working over years (e.g. TF API changes, ...) and the software packages/tools current
- Modules for physics applications (thanks to DSECOP Fellows!)
- Access to GPU resources for training deep neural network models

Work in progress:

- Developing an advanced version for MEng in Instrumental Physics @ Illinois



<https://a3d3.ai>

