Title: “Machine Learning in the Geosciences” 469/569

Course Description

The course is intended to introduce Machine Learning in Geosciences, the basics of computing, and methodologies in applied machine learning. The course focuses on canonical and topical data sets in seismology, oceanography, cryosphere, planetary sciences, geology, and geodesy. The methods taught include unsupervised clustering, logistic regression, random forest, support vector machine, and deep learning.

Course Overview

Quantitative data analysis is becoming a necessary skill for most geoscientists. The course is intended to provide an introduction to Machine Learning in Geosciences, the basics of computing, and methodologies in applied machine learning. The course is centered around experiential learning by applying machine learning to canonical and topical data sets in the geosciences in seismology, oceanography, cryosphere, planetary sciences, geology, and geodesy. The machine-learning methods taught include unsupervised clustering, logistic regression, random forest, support vector machine, linear-convolutional-recurrent neural networks, and deep learning. The students will learn and implement best practices in open and reproducible science.

Learning objectives

By the end of the quarter, the students should be able to:

- Demonstrate computing skills in python, jupyter notebooks, Git version control, and deploy scripts on local computers, cloud-hosted hubs, or cloud instances.
- Apply standard data manipulation strategies in Geosciences: data modalities (tabular, time series, and geospatial), data formats, data visualization, dimensionality reduction, and feature engineering.
- Describe and demonstrate the adoption of open science principles, science reproducibility, and digital scholarship.
- Describe the canonical use cases of AI/ML in geosciences (discovery, automation, signal processing, generative AI).
- Understand at least qualitatively how some of the advanced techniques (Fourier and wavelet transform, principal component analysis, …) manipulate and transform the data to interpret the output.

- Develop and apply standard machine-learning workflows: 1) Data preparation, 2) Model design, 3) Model training, validation, and evaluation.
- Describe and demonstrate the concepts of classic machine learning, deep learning, and construct ML workflows for feature extraction, classification, and regression.
- Describe the difference between various modes of supervision for ML (self, fully supervised, unsupervised).
- Demonstrate the ability to analyze and write a structured scientific paper.

Prerequisites

MATH 207 and MATH 208, or MATH 307 or 308, or AMATH 351 or 352, CS160 or CS163, or permission from the instructor.

Recommended: Knowledge in Matlab or python, AMATH301, 100- or 200-level courses in the Earth Sciences. Refreshers in computing skills will be provided.

Textbooks

Open-access main course resource:

- GeoSMART curriculum book.
- Deep Dive into Deep Learning [pytorch+tensorflow]
- Earth Data Science at CU Boulder

Recommended (but not required) depending on the student's own interest

- Practical books to apply ML tools to data:
    - Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow, Aurelien Geron (buy it here)
    - Deep Learning in Python, Francois Chollet (buy it here, github page here)

- Textbook on machine learning and statistical learning fundamentals:
    - Introduction to Machine Learning, 4th edition, Ethem Alpaydin (buy it here)
    - An Introduction to Statistical Learning with applications in R, James, Witten, Hastie, Tibshirani (buy it here)

- Introduction to basic data science in the geoscience (entry-level):
    - Intro to python programming for scientists: https://pythonnumericalmethods.berkeley.edu/notebooks/Index.html
- JupyterBook on open science:
    - The Turing Way: https://the-turing-way.netlify.app/welcome

Course Logistics

Instructors: Marine Denolle (mdenolle@uw.edu)

Teaching assistant: Akash Kharita (ak287@uw.edu)

Class meetings: MWF 9:30-10:50 (room JHN175)

Office hours: TuFr 10 or 11 (room JHN375)

Communication

- UW Registrar: https://www.washington.edu/students/timeschd/AUT2024/ess.html
- Canvas website:https://canvas.uw.edu/courses/1580283
- Github website: geo-smart.github.io/curriculumbook/
- Materials will be posted in the form of notebooks.
- ESS slack channel #datageoscience
- Emailing policy: Marine will prioritize responding to emails the hour after class and during office hours.
- Announcements will be posted on Canvas and Slack

Technology

The course revolves around computing for hands-on exercise. It is recommended that you bring a personal laptop. The use of a tablet is still possible but less practical. If you do not have your own personal laptop, consider borrowing one from the UW (Science and Technology Loan Program). All the software we will be using is free and platform-independent, meaning students may use macOS, Linux, or Windows operating systems.

We set up a UW-maintained JupyterHub that will provide browser access to computing. The UW JupyterHub will have the class python environment. Other options will be available such as the Google Colab notebook servers, which provide CPU, GPU, and TPU-free instances, and the Planetary Computer Hub. You may install the class python packages (X) using "!pip install X".

Running larger codes on your own laptop may be more efficient and flexible. We will code in Python and recommend installing Anaconda3 (individual edition) to manage python packages. We will provide a (YML) file to install the class python environment. We will learn basic cloud computing and teach tutorials on how to run large-scale jobs on AWS and the Planetary Computer Hub using CPU and GPU instances, accessible during and beyond the course offering. The UW Research Computing Club offers students access to the UW Hyak cluster.

Students will be required to have a Github account and submit their assignments on Github. If you do not have an account yet, sign up here. Instructors will provide training on how to use Github.

Evaluation

Grading policy 569:

- Reading and webinar assignments: 20%
- Homeworks: 40%
- Final project: 30%
- Quizzes: 10%
- All assignments must be turned in for a passing grade.

Grading policy 469:

- Reading and webinar assignments: 15%
- Homeworks: 45%
- Final project: 30%
- Quizzes: 10%
- All assignments must be turned in for a passing grade.

Engagement

Participation in the class is central to student learning. Engagement in the class will prepare students to enter the workforce. In-class engagement will improve the communication skills necessary in professional group settings. Engagement will be assessed with 5-min weekly quizzes shared in class only.

Late work policy:

Time-limited assignments or tasks are standard in most professional environments. This policy is designed so that every student is treated fairly when providing the materials at the same time. It is also designed to respect the teaching staff's own time that is limited to the instruction of this course. Returning all assignments at the same improves the quality and depth of evaluation and feedback.

You are allowed **once** up to 2 late days for homework. Use it wisely and in case of emergency! Email the teaching staff if you anticipate needing an extra day at least 48 hours before the deadline. Otherwise, you will receive a grade of zero.

Reading and webinar assignments

Each week, students will write a short report about either a paper or a webinar. Use the template on canvas and answer the questions when appropriate. Submissions of the report PDF are due Wednesdays at 11:59 pm PDT on canvas. The instructor will spend 15 minutes Monday morning summarizing the reading and webinar reports. Papers can be found and/or uploaded on a shared private course Google Drive here.

469 students can choose 4 out of the 8 reading assignments among the *recommended research overview* papers (*). 569 students are expected to complete the 8 reading assignments.

Week (due date)	**Theme**	**Recommended readings**
Week 1 (/)	ML in the geosciences	Bergen et al, 2019(*) ; Karpatne et al, 2018 ; White paper for Planetary Sciences 2020 (*); Li et al, 2020
Week 2 (10/12)	Open Science and Reproducibility	Donoho 2010; Gil et al, 2016;
Week 3 (10/22)	Data Practices, Cloud Computing	Wilkinson et al, 2016 (*); Tenopir et al, 2018 ; Stall et al, 2019; Amani et al, 2020; Oionomou et al, 2021; MacCarthy et al, 2020;
Week 4 (10/26)	Resampling methods and data bases	Lyons et al, 2018; Goetz et al, 2015 Puetz et al, 2018; Welty et al, 2020l; Weinert et al, 2021
Week 5 (11/2)	Final project proposal report	
Week 6 (11/9)	Dimensionality reduction and clustering	Yin et al, 2021; Unglert et al, 2016; Eymold and Jordan 2019; Sick et al, 2015; Roden et al, 2015; Santos et al, 2020; Michel et al, 2019, Holtzman et al, 2018
Week 7 (11/16)	ML project design, Classification, Logistic Regression, Random Forests, Support Vector	Kirkwood et al., 2016, Kortström et al, 2016;

	Machines	Rouet-Leduc et al, 2017; Rodrigyez-Galiano et al, 2012; Provost et al, 2017; Tesoriero et al, 2017; Cracknell et al, 2014; Petrelli & Perugini 2016; Crackwell & Reading 2014: Sun et al., 2022(*)
Week 8 (11/23)	Neural Networks	LeCun et al, 2015(*); DeVries et al, 2017; Roth and Tarantola 1994;
Week 9 (11/30)	Convolutional and Recurrent Neural Networks	Shoji et al, 2018; Perol et al. 2018; Zhang et al, 2017; Jum et al, 2020; Marchant et al. 2020
Week 10 (12/7)	GANs, Data Assimilation	Dupont al 2018; Mosser et al, 2020

(*) Review, accessible papers that 469 students can focus on.

Homework

Homework assignments will be posted on canvas after Wednesday's course at 11am and due the following Wednesday at 11:59 pm PDT on canvas. Computing exercises to perform on a notebook (hub provided by UW JupyterHub, Google Colab, or your own laptop).

Homework #1 (due /): Workbench setup (Git, Python, open science)
Homework #2 (due /):
Homework #3 (due /): Classification using classic ML:
of celestial objects using various classic ML classifiers
Unsupervised with K-means clustering
Homework #4 (due /): Deep Learning Event Classification

Final project

569: lead a project. 469: assist a 569 project

The final project will be a research-style project that will leverage the materials covered and apply them to new data in geosciences. The students will be evaluated on the following items:

1. Formulation of an Outstanding Research Question: an argument for the scientific inquiry based on Literature Review
2. Design and deploy a scientific workflow using appropriate computing resources.
3. Gather or curate a data set to be AI-ready (e.g., well-formatted, clean labels)
4. Present first-order analysis of their data with basic statistical distributions and correlations.
5. Develop and deploy a Classic Machine Learning algorithm
6. Develop and deploy a Deep Learning algorithm.
7. Assess the performance of the ML models and assess generalizability.
8. Ensure the reproducibility of the results
9. Demonstrate the scientific advances of the analysis.

The deliverables of the final projects are:

- A 5-page report (35% grade) with the 7 items described above. The report must have a section that describes each author's contribution statement following CRediT: https://casrai.org/credit/. A template will be provided in the form of a Word doc and of a Latex file. Place the name of the students responsible for each figure. 569 students are expected to oversee the report in its entirety. 469 students are expected to contribute to at least 1 figure and roughly 2 pages of text in the first iteration of the report.
- A GitHub repository with documentation on the data and codes (30% grade). 569 and 469 students will be expected to commit to the GitHub repository, with the frequency of the commits being counted as assessments.
- A 15-min presentation (35% grade). 569 students are expected to present ~70% of the presentation, 469 students are expected to present ~30% of the presentation. Every member is expected

A midterm project report is expected at the end of the first part of the class. Please see here the description of the report. It will count as a homework.

Rubric for Presentation (100 points, 35% of the final project grade)

Category	**Criteria**	**Points**
Content	*Quality and depth of research*	20
	Present a clear statement of an outstanding research question	
	Provides evidence and reference to the scientific literature to support the key points	

	Present an AI-ready data set and preliminary analysis.	
	Demonstrate a classic machine learning example.	
	Demonstrates a deep learning example	
	Discuss the computational time for training and deploying	
	Discuss the appropriateness of computational resources needed	
Structure and Organization	*Coherence and flow of the presentation*	15
	Logical structure (report and presentation: intro, body, conclusion; software: readme, env file, src/, data/, plots/,...)	
	Key points are clearly distinguished and emphasized	
Clarity and Delivery	*Effectiveness in communicating ideas*	20
	Clear articulation and pronunciation	
	Adequate volume and pace	
	Minimal reliance on notes, maintaining eye contact with the audience	
	Confident and professional demeanor	
Visual Aids	*Effectiveness of any supporting visual materials (slides, charts, plots)*	10
	Visual aids enhance, not distract, from the presentation	
	Information is presented clearly, is easy to follow, and uses appropriate design principles	
	Slides do not overwhelm with text or complex visuals	
Engagement and Interaction	*Ability to engage and interact with the audience*	10
	Encourages audience interaction through questions or active participation	
	Responds effectively to audience questions and comments	
Critical Thinking & Analysis	*Depth of analysis and reflection on the research topic*	15
	Demonstrates original thinking and critical engagement	

	with the research	
	Identifies limitations or future directions for research	
Professionalism	*Overall, the professional quality of the presentation*	10
	Respect for time limits, prepared with material	
	Appropriate attire and a respectful manner	

Rubric for Report (100 points, 35% of the final project grade)

Category	**Criteria**	**Points**
Content & Research Quality	*Quality, depth, and relevance of the research*	25
	Present a clear statement of an outstanding research question and place it in the context of an up-to-date literature review	
	Demonstrates the originality of the research	
	Present an AI-ready data set, preliminary analysis with correlation or description of basic data feature, and discuss potential data imbalance within the context of the stated problem.	
	Demonstrate strong understanding of classic machine learning and/or deep learning with an example	
	Discuss the performance evaluation in the context of training the model and for generalization beyond the data and domain presented in the report.	
	Discuss the computational time for training and deploying	
	Discuss the appropriateness of computational resources needed	
Structure and Organization	*Coherence, clarity, and flow of the report*	15
	Logical structure (introduction, body, conclusions, references)	
	Well-organized paragraphs with transitions between ideas	
	Key points and arguments are clearly presented and easy to follow	

	Figures presented have captions and clear labels and are references in the text.	
Clarity and Writing Style	*Quality & Effectiveness in Writing*	20
	Clear concise language	
	Minimal grammar or spelling errors	
	Professional and academic tone appropriate for the field	
Critical Thinking & Analysis	*Depth of analysis and reflection on the research topic*	20
	The analysis goes beyond simple description and shows depths of thought	
	Acknowledges alternative perspectives or potential limitations in the research	
	Demonstrates original thinking and critical engagement with the research	
Formatting & Citations	*Adherence to format guidelines and proper citation of sources*	10
	Follows formatting requirements (e.g., margins, font, length)	
	Correct use of citation style (e.g., APA, MLA, Chicago)	
	Correctly state the author's CredIT (every student enrolled needs to be associated with a contribution and the CreDIT statement will assert that)	

Rubric for GitHub (100 points, 30% of the final project grade)

Category	**Criteria**	**Points**
Code Quality	*Quality and functionality of the scripts and code*	25
	The code is clean, well-documented, and follows good programming practices	
	All scripts run without errors (when the environment is properly set up)	
	Code is modular, with reusable functions where appropriate	

	Scripts achieve the intended outcomes (e.g., generating plots, performing analysis)	
Reproducibility	*Ease of reproducing the analysis and results*	25
	The repository includes clear instructions (e.g., in a `README.md`) for setting up the environment and running the code	
	Jupyter notebooks, scripts, and any other files necessary to recreate the analysis are provided	
	Data (or instructions to access data) are included or referenced appropriately	
	Output (plots, tables) are reproducible using the code	
Organization & Structure	*Organization and clarity of the repository structure*	20
	Repository is well-organized with clear folder structure (e.g., separate folders for code, data, results, etc.)	
	File and folder names are descriptive and intuitive	
	README file provides a clear overview of the repository and how to navigate it	
Documentation	*Clarity and completeness of documentation*	15
	README file clearly explains the project, dependencies, and setup instructions	
	Code and notebooks are well-documented, including comments explaining key sections	
	Scripts include docstrings for functions and appropriate inline comments	
Environment Setup	*Provision of environment setup and dependency management*	10
	Includes a complete and working `conda` environment file (`environment.yml`) or `requirements.txt` for virtual environments	
	Environment file lists all necessary dependencies with correct versions	
	Instructions for setting up the environment are clear and easy to follow	

Version Control Practices	*Effective use of Git and GitHub features*	5
	Commits are frequent, descriptive, and reflect the progression of the project.	
	Clear use of branches, if applicable (e.g., for different features or phases of the project)	
	Issues, pull requests, or other GitHub collaboration tools are used effectively.	

How to use ChatGPT to improve your work

Please use the rubric above, attach a PDF, and ask chatGPT to grade it and provide constructive feedback. Save the report as a PDF to be uploaded to Canvas. Improve your work, and demonstrate it with a new evaluation using chatGPT.

569 project leaders perform all listed tasks. 469 project assistants help project leaders, in particular in tasks 4, 5, 6, 7. The collaboration between project leaders and assistants is similar to what can be expected during an undergraduate research experience. Every group member is expected to speak at the presentation.

The team will provide a brief progress report (½ page PDF to submit on canvas) at the dates listed above, which will help make progress during the quarter.

Project presentations will be hosted on Wednesday, December 4 and 6, 2024, at 9:30 a.m.

Collaboration Policy

The reading, webinar, and homework assignments are expected to be completed individually. The Students can do the final projects in groups.

Getting help with programming resources:

- Git http://swcarpentry.github.io/git-novice/
- Unix http://swcarpentry.github.io/shell-novice/
- Python https://swcarpentry.github.io/python-novice-inflammation/

Religious accommodation

Washington state law requires that UW develop a policy for the accommodation of student absences or significant hardship due to reasons of faith or conscience, or for organized religious activities. The UW's policy, including more information about how to request an accommodation, is available at Religious Accommodations Policy

(https://registrar.washington.edu/staffandfaculty/religious-accommodations-policy/). Accommodations must be requested within the first two weeks of this course using the Religious Accommodations Request Form (https://registrar.washington.edu/students/religious-accommodations-request/).

Summary Schedule

Module 1 (week 1): Intro to Machine-Learning in Geo and basic tool building
Module 2: (weeks 2 through 5) AI-ready Data in Geo
Module 3: (weeks 6 and 7) Classic Machine Learning in Geo
Module 4: (weeks 8 and 9) Deep Learning in Geo

Detailed Schedule

The course schedule is organized by methods but discussed and applied in the context of geoscientific applications.
Check this spreadsheet for more details.

Date	Lecture topic	Slide and resources	Assignments due
Wed 09/25	**Introduction** to ML in the geosciences and course logistics. Recent scientific discoveries. Open Geosciences	Slides, Slides_2023 Github account - Vscode installed - access to the Jhub. Clearly go to software carpentries. Group activities - get to know each other. Disciplinary affinities Set up office hours time.	
Fr 09/27	**ToolKit:** software, computing, and literature	Notebooks, Bash resources, Slides, MLGeoBook, chatGPT, CoPilot	Watch Open Science lecture
Mo	**Project**	Git, MLgeoBook, students	Readings #1 (review in

09/30	**Management**:	present a 1-slide summary lightning talk about their favorite review. **Show how to get data with notebooks.**	ML geoscience)
Wed 10/2	Data Wrangling #1: Data definition and formats, manipulating pandas	Slides (30 min) MLGeoBook format (30 min) Practice on pandas MLGeoBook	
Fr 10/4	Data Wrangling #3: loading and manipulating ND arrays	MLGeoBook	HWK1: Toolkit primer
Mo 10/7	Data Wrangling #4: visualization	Slides	Reading #2 (open science, Data Best Practices and Cloud computing)
Wed 10/9	Working with large data #1: resampling	MLGeoBook (will be upgraded)	
Fr 10/11	Data Sources in the geosciences. 30 min to formulate project team and	GROUP PROJECT. brainstorming	
Mo 10/14	Working with large data #2: transforms	MLGeoBook (to be upgraded)	Readings #3: examples of resampling and geoscience databases
Wed 10/16	Working with large data #3: synthetic data		
Fr 10/18	Working with large data #4: feature engineering		HWK2
Mo 10/21	Data characterization: correlation, distribution, overlap of	30 min on AI-ready data sets. 60min on data characterization.	Readings #4: AI-ready data

	features among classes, outliers		
Wed 10/23	Lecture 12: Working with large data #4: Dimensionality reduction	PCA	
Fr 10/25	ML: classification and regression problems, performance		**AI-ready dataset due**
Mo 10/28	Clustering #1: Kmeans& tsne	Book link here	
Wed 10/30	Clustering #2: Hierarchical		
Fr 11/1	Classic ML: methods and hyperparameter tuning		
Mo 11/4	Ensemble learning (xgboost+RF)	ML project design	Readings #5: clustering + CML
Wed 11/6	All CMLs compared - AutoML	• Add computational performance test	
Fr 11/8	Neural Networks and Gradient Descent	Single NN,	**HWK CML**
Wed 11/13	Multi Layers Perceptrons	Multi-layer NN	**CML applications to their final project**
Fr 11/15	Training Neural Networks	Training neural networks Cryo tutorial here.	
Mo 11/18	Convolutional Neural Networks	Convolutional Neural Networks,	Readings #6: DNNs
Wed 11/20	Encoder-Decoder (denoising, forecasting)		

Fr 11/22	Seq2Seq learning & forecasting	Recurrent Neural Networks, LSTMs , Book link	
Mo 11/25	Knowledge Guided ML - PINNs - Interpretable AI	https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2019MS002002	Readings #7. Knowledge Guided MLGEO, HWK4
Mo 12/2			
Wed 12/4	Foundational Model - generative AI - multi-modal prediction		
Mo 12/6	Project Presentations		
Wed 12/8	Project Presentations		