ANLY 530 — Principles & Applications of Machine Learning

Ziyuan Huang, PhD

Late Spring 2026

Instructor	Ziyuan Huang, PhD
Email	zhuang7@harrisburgu.edu
Office Hours	By appointment — email to schedule at least 24 hours in advance
Class Meeting	Thursdays, 8:30 PM – 10:00 PM (Eastern)
Format	Synchronous Online (MS-Teams / Virtual Classroom)
Dates	March 14, 2026 – June 18, 2026 (14 weeks)
Credits	3.00 ∣ Reference: 30501 ∣ Dept: ANLY

Welcome

Welcome to **ANLY 530 — Principles & Applications of Machine Learning**! This course is your gateway to one of the most transformative fields in modern technology. Over the next 14 weeks, you will move from understanding *what* machine learning is to actually *building* ML systems that solve real problems.

You do not need to be a mathematics expert to succeed here. What you need is curiosity, persistence, and a willingness to write code and make mistakes. We will work through concepts together in class, and assignments are designed to give you hands-on practice at each step of the journey.

By the end of the course, you will have a portfolio of ML projects to show employers, a working knowledge of the most important algorithms in the field, and the confidence to pick up new ML tools on your own.

Course Description

This course introduces students to machine learning. It provides students with the cognitive, mathematical, and analytical foundation required for machine learning, including topics from data mining, pattern recognition, and supervised and unsupervised learning. Although it may be taken as a stand-alone course, it also prepares students for the complex, higher-level topics covered in Principles and Applications of Deep Learning.

This course aims to connect classroom learning to professional practice. Assignments designated with an asterisk (*) are linked to the student's current or future employment, enabling real-world application of the material covered in class.

Learning Objectives

At the end of this course, students will be able to:

1. Develop models that integrate state-of-the-art machine learning techniques with fundamental princi-

ples of cognitive science and mathematics.
2. Build sophisticated analytics solutions that apply machine learning to real-life problems.
3. **Communicate findings about machine learning clearly — within teams, across organizations, and to external stakeholders.**
4. **Evaluate** the strengths and limitations of different ML algorithms and select the appropriate method for a given problem.
5. **Build reproducible, well-documented machine learning pipelines** using R and Python.

Texts and Programs

Required Textbook

Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd Ed.) Aurelien Geron — O'Reilly Media, 2019

- **Free online access** (Harrisburg University credentials required): https://learning.oreilly.com/library/view/hands-on-machine-learning/9798341607972/ Log in with your @harrisburgu.edu account at learning.oreilly.com.

Required Software (free)

1. **R** and **RStudio Desktop**: https://www.r-project.org/ | https://posit.co/downloads/
2. **Python 3** via **Anaconda**: https://www.anaconda.com/ — includes Jupyter Notebook, Spyder, and all scientific libraries
3. Use the R `reticulate` package to knit RMarkdown reports that include Python code chunks

Supplementary References (optional, all free online)

- *An Introduction to Statistical Learning with Applications in R* (2nd Ed.) — James et al.: https://www.statlearning.com/
- *Pattern Recognition and Machine Learning* — Bishop: https://www.microsoft.com/en-us/research/uploads/prod/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf
- *Linear Algebra* — Hefferon: http://joshua.smcvt.edu/linearalgebra/

Course Policies

1. **Canvas:** All course materials, assignments, grades, and announcements are posted on Canvas. Check Canvas at least twice per week. Technical difficulties with Canvas are not an excuse for late submissions — plan ahead.
2. **Attendance & Participation:** Live Thursday sessions are required. Active participation (asking questions, contributing to discussion) forms part of your grade. If you must miss a class, notify the instructor *in advance*. More than two unexcused absences will result in a deduction from your participation grade.
3. **Assignments:** There are 8 assignments, each covering topics from recent lectures. Key policies:
   - Submit as a **compiled report** (HTML or PDF rendered from RMarkdown or Jupyter Notebook). Raw, uncompiled code files (`.R`, `.py`, `.ipynb` without output) will not be graded and will receive a zero.
   - Each assignment is graded once. Resubmissions are not accepted.
   - **Late penalty:** 10% of your earned score is deducted for each calendar day late. A score of 85 submitted 2 days late becomes 85 – 2(8.5) = **68**. Submissions more than 10 calendar days late receive a **zero**.

- All work must be your own. Identical or near-identical answers from multiple students, or answers copied verbatim from online sources, will receive a zero and may be referred for academic integrity review.
- Cite all sources in APA format (https://www.apa.org/).
- If you fall behind or face personal difficulties, reach out *early* — the late policy cannot be waived retroactively, but early communication may open other options.

4. **Assigned Readings:** Complete readings *before* class. Many ML concepts require multiple encounters to fully understand. Re-reading after class and working through code examples from the book is strongly encouraged.
5. **Email:** Use your official `@harrisburgu.edu` email for all course communications. Emails from personal addresses will not receive a response. Expect a reply within 48 hours on business days.
6. **Use of Artificial Intelligence (AI) Tools:** You may use AI assistants (e.g., ChatGPT, GitHub Copilot, Claude) to **help you understand** concepts, debug code, or improve writing clarity. However:
   - **You are responsible for every line of code and every claim in your submission.** If an AI generates incorrect code or a wrong conclusion and you submit it, the error is yours.
   - **Do not submit AI-generated text as your own written analysis.** Explain results in your own words.
   - **Disclose AI use:** Include a brief note (one sentence is fine) at the end of any submission where you used an AI tool, describing how you used it.
   - Submitting AI output verbatim without attribution is treated as plagiarism under the academic integrity policy.

HU Core Competencies

At the conclusion of this course, students will have demonstrated:

1. **Critical Thinking & Problem Solving —** Identifying problems, gathering and evaluating evidence, considering alternatives, and implementing solutions.
2. **Communication —** Expressing technical ideas clearly in written, oral, and visual formats using appropriate tools and resources.
3. **Teamwork & Collaboration —** Engaging with peers to deepen understanding through discussion and shared learning.
4. **Information Technology —** Making effective use of computational tools, data science platforms, and information resources.

Statement of Academic Integrity

Academic integrity is the pursuit of scholarly activity free from fraud and deception. Academic dishonesty includes, but is not limited to: cheating, plagiarism, fabrication of information or citations, facilitating dishonesty by others, submitting another person's work, submitting previously used work without disclosure, or tampering with other students' academic work. Violations will be thoroughly investigated, and punitive action taken where warranted.

Honor Code: We as members of the Harrisburg University community pledge not to cheat, plagiarize, steal, or lie in matters related to academic work. We honor and uphold the *HU Honor Code*.

Final Project

The final project is your opportunity to apply everything covered in the course to a real dataset of your choosing. You will go through the full machine learning workflow: problem framing, data acquisition and cleaning, exploratory analysis, model building, evaluation, and communication of results.

Deliverables

Deliverable	Format	Due
Project Proposal	1-page Word or PDF document submitted to Canvas	Week 3 (Mar 28)
Final Project Report	MS Word document using the **IEEE Access template** (provided on Canvas); 8 pages minimum, excluding references	Week 14 (Jun 18)
Final Project Presentation	15-minute recorded video with slides (uploaded to Canvas)	Week 14 (Jun 18)

Project Proposal

By **Week 3 (March 28, 2026)**, each student or team must submit a **one-page project proposal** to Canvas. The proposal is your opportunity to think through the problem before diving into data and code. It should address all five of the following points:

1. **Dataset and Source** — Identify the dataset you plan to use and provide its source (URL, repository name, or institution). Explain the size and structure of the data (number of rows, columns, variable types).
2. **Related Work** — Briefly review what others have done with this dataset or on this type of problem. Cite at least two references. Explain how your approach is different or what new angle you are exploring.
3. **Proposed Methods** — Describe the machine learning methods you plan to apply. You do not need to commit to a final list, but you should name at least two algorithms and explain why they are appropriate for your problem.
4. **Why Machine Learning?** — Explain why machine learning is a suitable approach for this problem. What makes it hard to solve with simple rules or statistics alone?
5. **Team Responsibilities** — If working in a team, list each member and their planned responsibilities (e.g., data cleaning, modeling, report writing, presentation). If working individually, state that this is a solo project.

   The proposal is graded on completeness and clarity, not on whether your final project matches it exactly. You may change your dataset or methods after submission — just notify the instructor.

Requirements

- **Citations:** The report must include **at least 12 references** cited in IEEE format. References should include primary sources such as peer-reviewed journal articles, conference papers, and authoritative textbooks. Blog posts and Wikipedia do not count toward the minimum.
- **Dataset:** Use a publicly available, real-world dataset (e.g., from Kaggle, UCI ML Repository, data.gov, or your workplace). Synthetic or toy datasets (like `iris`) are not acceptable for the final project.
- **Scope:** Apply at least **two different ML algorithms** and compare their performance rigorously.
- **Evaluation:** Report appropriate metrics (accuracy, RMSE, AUC-ROC, etc.) and explain what they mean in plain language.
- **Report format:** Use the **IEEE Access MS Word template** (file: `Access-Template-2024.docx`, available on Canvas). Your report must be **at least 8 pages** of content, not counting the references section. Structure your report following the IEEE Access conventions: Abstract, Introduction, Related Work, Methodology, Results, Discussion, and Conclusion.

- **Data Availability Statement:** Your report must include a *Data Availability Statement* (a standard IEEE Access requirement) declaring where the dataset can be accessed. If the data is publicly available, provide the URL or DOI. If the data is proprietary or cannot be shared, state that clearly and explain why.
- **Code Availability Statement:** Your report must include a *Code Availability Statement* stating that all code is publicly available on **GitHub**. You are required to create a GitHub repository for your project, push all code to it, and include the repository URL in this statement. The repository must be public and accessible by the submission deadline. Example statement: *"All code used in this study is publicly available at https://github.com//<username>/<repo-name>."*
- **Reproducibility:** The code in your GitHub repository must run cleanly from top to bottom with no errors. Include a `README.md` describing the environment setup and how to reproduce the results.
- **Presentation:** Record a **15-minute video** (screen capture with voiceover is fine — tools like Zoom, OBS, or Loom all work). Assume your audience is a non-technical manager. Explain your problem, your approach, and your findings clearly — minimize jargon. Upload the video or a shareable link to Canvas by the deadline.

Suggested Project Timeline

Milestone	Recommended Completion
Dataset selected & problem framed	Week 5 (Apr 11)
EDA and data cleaning complete	Week 8 (May 2)
First model trained and evaluated	Week 10 (May 16)
Model comparison and tuning complete	Week 12 (May 30)
Report draft complete	Week 13 (Jun 6)
Final report & presentation	Week 14 (Jun 18)

Grading

Component	Weight	Description
Assignments (8 total)	60%	7.5% each; best indicators of weekly mastery
Final Project	30%	Report 20%, Presentation 10%
Attendance & Participation	10%	Presence, engagement, and discussion contributions

Grade	Range	Meaning
A	90.00% and above	Excellent — exceeds expectations
B	80.00% – 89.99%	Good — meets all expectations
C	70.00% – 79.99%	Satisfactory — meets minimum expectations
F	Below 70.00%	Failing — does not meet minimum expectations

Submission Schedule

All deadlines are at **11:59 PM Eastern on the date listed**. The 10% per day late penalty applies immediately after this deadline. Submissions more than 10 days late receive a zero.

Table 1: Complete Submission Schedule — ANLY 530 Late Spring 2026

Deliverable	Due Date (11:59 PM ET)	Grade Weight	Zero After
Assignment 1 — Decision Trees	Saturday, April 04, 2026	60% combined (8 assignments)	April 14, 2026
Assignment 2 — Random Forests	Saturday, April 11, 2026	60% combined (8 assignments)	April 21, 2026
Assignment 3 — Regression	Saturday, April 18, 2026	60% combined (8 assignments)	April 28, 2026
Assignment 4 — Support Vector Machines	Saturday, April 25, 2026	60% combined (8 assignments)	May 05, 2026
Assignment 5 — Naive Bayes	Saturday, May 02, 2026	60% combined (8 assignments)	May 12, 2026
Assignment 6 — Unsupervised Learning	Saturday, May 09, 2026	60% combined (8 assignments)	May 19, 2026
Assignment 7 — Feature Engineering & PCA	Saturday, May 23, 2026	60% combined (8 assignments)	June 02, 2026
Assignment 8 — Ensemble Modeling	Saturday, June 06, 2026	60% combined (8 assignments)	June 16, 2026
Final Project Proposal	**Saturday, March 28, 2026**	**30% combined (final project)**	**April 07, 2026**
Final Project Report	**Saturday, June 13, 2026**	**30% combined (final project)**	**June 23, 2026**
Final Project Presentation	**Saturday, June 13, 2026**	**30% combined (final project)**	**June 23, 2026**

Class Schedule

The schedule below reflects the **actual Thursday class dates** (March 14 – June 18, 2026). Lecture topics may shift slightly based on course pace; the final project deadline will not change. All materials and assignments will be posted on Canvas.

Week 01 – March 14, 2026

Topic: Introduction to the Course & Machine Learning Landscape

- Syllabus, Canvas, project overview, and assignment expectations
- What is Machine Learning? Supervised, unsupervised, and reinforcement learning
- The ML workflow: data to model to evaluation to deployment
- *Reading:* Geron Ch. 1 — The Machine Learning Landscape

Week 02 – March 21, 2026

Topic: Python & R for Machine Learning

- Installation: R, RStudio, Anaconda, Jupyter
- Basics of Python and R: data types, control flow, functions
- RMarkdown + reticulate for reproducible reports
- Key libraries: NumPy, Pandas, Matplotlib, Scikit-Learn, tidyverse, ggplot2
- *Reading:* Geron Ch. 2 — End-to-End ML Project (setup sections)
- **Start Project Proposal** (due Week 3, March 28)

Week 03 – March 28, 2026

Topic: Decision Trees

- How decision trees split data (CART algorithm, Gini impurity, entropy)
- Overfitting, pruning, and tree depth
- Visualising and interpreting decision boundaries
- *Reading:* Geron Ch. 6 — Decision Trees
- **Start Assignment 1:** Decision Trees
- **Project Proposal Due** (11:59 PM, March 28, 2026)

Week 04 – April 04, 2026

Topic: Random Forests & Ensemble Methods

- From decision trees to random forests: bagging and feature randomness
- Feature importance
- Brief introduction to boosting (AdaBoost, Gradient Boosting)
- *Reading:* Geron Ch. 7 — Ensemble Learning and Random Forests
- **Assignment 1 Due** | **Start Assignment 2:** Random Forests

Week 05 – April 11, 2026

Topic: Regression

- Linear regression: OLS, gradient descent, cost functions
- Polynomial regression and bias-variance tradeoff
- Regularisation: Ridge, Lasso, ElasticNet
- Logistic regression for classification
- *Reading:* Geron Ch. 4 — Training Models
- **Assignment 2 Due** | **Start Assignment 3:** Regression

Week 06 – April 18, 2026

Topic: Support Vector Machines (SVM)

- Hard and soft margin classification
- The kernel trick: polynomial, RBF
- SVM for regression (SVR)
- *Reading:* Geron Ch. 5 — Support Vector Machines
- **Assignment 3 Due** | **Start Assignment 4:** Support Vector Machines

Week 07 – April 25, 2026

Topic: Probabilistic Learning — Naive Bayes & Classification Metrics

- Bayes theorem and conditional probability
- Naive Bayes classifier
- Evaluation: confusion matrix, precision, recall, F1-score, ROC-AUC
- *Reading:* Geron Ch. 3 — Classification
- **Assignment 4 Due** | **Start Assignment 5:** Naive Bayes

Week 08 – May 02, 2026

Topic: Unsupervised Learning

- Clustering: K-Means, DBSCAN, hierarchical clustering
- Anomaly detection and density estimation
- Gaussian Mixture Models
- *Reading:* Geron Ch. 9 — Unsupervised Learning Techniques
- **Assignment 5 Due** | **Start Assignment 6:** Unsupervised Learning

Week 09 – May 09, 2026

Topic: Preprocessing & Feature Selection

- Handling missing data, outliers, and class imbalance
- Scaling and encoding (StandardScaler, OneHotEncoder)
- Feature selection: filter, wrapper, and embedded methods
- Scikit-Learn Pipelines
- *Reading:* Geron Ch. 2 (pipeline sections) | Ch. 13 (preprocessing)
- **Assignment 6 Due**

Week 10 – May 16, 2026

Topic: Feature Engineering & Dimensionality Reduction

- Feature creation, interaction terms, and binning
- PCA: principal components and explained variance
- t-SNE for visualisation
- *Reading:* Geron Ch. 8 — Dimensionality Reduction
- **Start Assignment 7:** Feature Engineering & PCA

Week 11 – May 23, 2026

Topic: Model Evaluation & Selection

- Cross-validation (k-fold, stratified)
- Hyperparameter tuning: Grid Search, Randomised Search
- Learning curves and validation curves
- *Reading:* Geron Ch. 2 (evaluation sections)
- **Assignment 7 Due**

Week 12 – May 30, 2026

Topic: Advanced Ensemble Methods

- Gradient Boosting in depth (XGBoost, LightGBM)
- Stacking and blending
- *Reading:* Geron Ch. 7 (boosting sections)
- **Start Assignment 8:** Ensemble Modeling

Week 13 – June 06, 2026

Topic: Introduction to Neural Networks

- Perceptron and multi-layer perceptron (MLP)
- Backpropagation and activation functions
- Building and training a neural network with Keras
- *Reading:* Geron Ch. 10 — Introduction to Artificial Neural Networks with Keras
- **Assignment 8 Due**

Week 14 – June 13, 2026

Topic: Final Project Wrap-Up & Review

- Course review and key takeaways
- Q&A session; common pitfalls and lessons learned
- **Final Project Report & Recorded Presentation Due** (submit video link and report to Canvas by 11:59 PM)