

Introduction to Machine Learning — Final Project Guidelines

Project Objectives

The goal of the final project is for students to apply the **fundamental machine learning methods** learned in class to real-world or open datasets, and to experience the complete ML workflow (data preprocessing → model building → evaluation → analysis). The emphasis is on applying basic methods, not reproducing the latest deep learning architectures or chasing SOTA performance.

Project Format

- **Group work:** 4 students per team (same as class discussion groups).
 - **Project types** (choose one of the following):
 1. **Assigned datasets** (beginner-friendly):
 - [Titanic: Machine Learning from Disaster \(binary classification\)](#)
 - [House Prices: Advanced Regression Techniques \(regression\)](#)
 - [Digit Recognizer \(MNIST\) \(image classification\)](#)
 2. **Past Kaggle competitions** (choose simple datasets such as classification/regression tasks; no need to replicate top solutions, just implement baselines + method comparisons).
 3. **Self-chosen dataset** (from UCI Repository, open government data, Kaggle Datasets, etc.), subject to instructor approval.
-

Project Requirements

1. **Dataset Description**
 - Source and overview
 - Number of features, size, type (tabular/image/text, etc.)
 - Learning task (classification, regression, clustering, etc.)
2. **Preprocessing & Feature Engineering**
 - Handle missing values, normalize the data, and perform feature selection/transformation, providing a thorough explanation.
3. **Model Implementation & Comparison**
 - At least three machine learning methods (recommended: Logistic Regression, Decision Tree/Random Forest, SVM, KNN, simple NN, etc.)
 - Compare performance using appropriate metrics (Accuracy, F1-score, MSE, etc.)
 - Optional: try advanced methods (CNN, XGBoost, etc.), but not required
4. **Results & Discussion**

- Which method works best and why?
- Possible improvements?
- How results relate to dataset characteristics / feature engineering choices
- Ability to explain the choice of certain methods (e.g., XGBoost vs. AdaBoost vs. Logistic Regression).
- Provide a simple introduction to the strengths and weaknesses of the selected models.
- Explain the meaning of key hyperparameters (e.g., learning rate, epoch, batch size, loss function, regularization strength).
- It is not necessary to test all parameter combinations, but you should be able to explain what would be affected if those parameters were adjusted.

5. Deliverables

- Written report (5–8 pages PDF, including figures and references)
- Final presentation (10 minutes + 5 minutes Q&A)

Grading

- Preprocessing & Feature Engineering: 20%
- Model Design & Comparison: 30%
- Results & Discussion: 20%
- Report & Presentation: 20%
- **Bonus:** Kaggle submission or creative extensions may earn up to +10%

Timeline

- **Week 9 (before Oct 27 23:59):** Submit project proposal (1-page dataset and task description)
- **Week 15–16 (Dec 9 and 16):** Final project presentations & report submission

Notes

- The project emphasizes **application of class knowledge and teamwork**, not heavy hyperparameter tuning or GPU-intensive training.
- Interdisciplinary topics (medical, biotech, business, electrical engineering, etc.) are encouraged, but dataset size should remain manageable.
- Students are expected to find their own GPU resources (e.g., Google Colab) if needed.