

AI310 • Machine Learning Foundations

School of Computational Arts & Sciences • Undergraduate • 6 ECTS

Overview

Covers supervised learning end-to-end: baselines, feature engineering, train/validation discipline, and error analysis. Students practice building reproducible evaluation reports, spotting leakage/overfitting, and communicating results with clear metrics and caveats.

LOGISTICS

Credits: 6 ECTS

Level: Undergraduate

School: School of Computational Arts & Sciences

Prerequisites: None listed

Tags: ml, evaluation, fundamentals

Meeting time: 2x90min per week + optional practice clinic

Instruction mode: Lecture + guided exercises; weekly evaluation lab

LEARNING OUTCOMES

You will be able to:

- Build supervised learning baselines and compare them fairly
- Run error analysis to diagnose leakage and overfitting
- Communicate results with clear metrics and caveats
- Implement baseline models and compare them fairly against more complex approaches
- Explain overfitting, leakage, and common evaluation traps in applied ML
- Design feature pipelines with clear train/validation separation

ASSESSMENT

Components

- Assignments: 60%
- Evaluation report: 40%

You are assessed on clarity and correctness of the evaluation story: what you measured, why those metrics, and what the results mean. Submissions include a short reproducibility note describing environment, data splits, and how to rerun experiments.

WEEKLY PLAN

Schedule

Week 1: Setup + evaluation discipline

- Train/validation splits
- Baselines
- Reproducible experiments

Week 2: Features + models

- Feature engineering
- Model selection
- Regularization

Week 3: Error analysis

- Slicing
- Leakage checks
- Overfitting diagnosis

Week 4: Reporting

- Metrics
- Caveats
- Recommendations

Extended outline

- Baselines first: linear models, sanity checks, and error analysis
- Data splitting and leakage: where evaluation silently breaks
- Feature pipelines: categorical encoding, scaling, and ablations
- Model families overview: trees, margins, simple neural nets
- Metrics: classification vs regression; calibration and thresholds
- Mini-project: build and evaluate an end-to-end model with a report

POLICIES & RESOURCES

-
- Reproducibility: every report includes seeds, splits, and rerun steps.
 - Data ethics: document data provenance and any sensitive attributes.
 - Collaboration: discuss ideas freely; submit your own code and write-up.

Suggested resources

- A short evaluation checklist: splits, baselines, error slices
- Metric notes: accuracy vs F1, AUC, calibration and decision thresholds
- Template: experiment log with run IDs and parameter tracking