Python for Data Science 2: Guided Machine Learning Project

University: Tek-Up

Group: ING-4-SDIA

Tutor: Haythem Ghazouani

Academic Year 2025-2026

Contents

1 Introduction

This 13-week module is designed to immerse students in a professional Machine Learning workflow. Unlike introductory courses, this module focuses on the **end-to-end lifecycle**: from raw data acquisition to a production-ready containerized application.

Module Objective

To master advanced Python for Data Science by implementing a reproducible, scalable, and deployable Machine Learning pipeline integrating MLOps best practices.

2 Project Synopsis & Teams Choice

The **Bank Customer Churn** example used in the tutorials serves as a **Project Synopsis** (a comprehensive reference implementation).

Important: Dataset Validation

Students must work in teams. Each team is free to choose their own dataset and objective (e.g., Fraud Detection, Recommendation Systems, Energy Consumption Prediction).
The choice MUST be validated by the tutor before the end of Week 2.

3 Detailed 13-Week Roadmap

3.1 Phase 1: Foundations & Data (Weeks 1-3)

- ☐ **Week 1-2: Setup & Scraping:** Environment configuration (Conda/Virtalenv), Git workflow, and optional Web Scraping using BeautifulSoup/Selenium.
- ☐ **Week 3: Exploratory Data Analysis (EDA):** Deep dive into statistical analysis, correlation mapping, and identifying data quality issues.

3.2 Phase 2: The ML Pipeline (Weeks 4-8)

- ☐ **Week 4-5: Preprocessing & Feature Selection:** Building robust pipelines for encoding, scaling, and handling missing data. Applying Wrapper/Filter methods for feature selection.
- ☐ **Week 6-7: Advanced Modeling:** Implementing Ensembling (Random Forest, Voting) and Gradient Boosting (XGBoost, LightGBM, CatBoost).
- ☐ **Week 8: MLOps with MLflow:** Logging parameters, metrics, and artifacts. Transitioning from "scripts" to "experiment tracking".

3.3 Phase 3: Integration & Deployment (Weeks 9-13)

- ☐ **Week 9-10: Backend (FastAPI):** Serializing models and exposing them via REST API endpoints.

- ☐ **Week 11: Frontend (React):** Creating a user-friendly interface to interact with the model.
- ☐ **Week 12-13: Deployment (Docker):** Orchestrating the full stack (API + Front) using Docker and Docker Compose.

4 Evaluation Criteria

Projects will be assessed based on the following:

1. **Code Quality:** Modularization, PEP8 compliance, and documentation.
2. **Experiment Tracking:** Proper use of MLflow to compare at least 3 model architectures.
3. **Robustness:** Handling edge cases in the API and Frontend.
4. **Reproducibility:** Successful deployment via `docker-compose up`.
5. **Creativity:** Originality of the chosen dataset and feature engineering.

5 Workspace Structure

Students must strictly adhere to the following directory structure:

- `/`: Root containing `README.md`, `.gitignore`, and `docker-compose.yml`.
- `cours/`: Theory notes and project guide (PDF).
- `tutos/`: Lab instructions (PDF).
- `code/`: Modular Python scripts (`train.py`, `app.py`, etc.).
- `data/`: Dataset samples (strictly no large files in Git).