National University of Sciences and Technology
School of Electrical Engineering and Computer Science
Department of Computing

CS 245: Machine Learning (2+1)

Fall 2025

Course Project

Brief Document

Announcement Date: 16th November 2025

Due Date: 14th December 2025 at 11:59 PM (on LMS)

Instructor: Mr. Usama Athar

Project Overview

This project is going to be the capstone assessment of our course “CS-245: Machine Learning”. You will work in groups of three to identify a pressing and unsolved problem within one of four thematic domains, apply machine learning models / techniques / algorithms / approaches to address it, and deliver a working Proof-of-Concept (PoC) alongside a rigorous IEEE-style technical report.

Unlike your lab sessions and assignments, your project is not going to be on a toy problem. In the course project, you will have the opportunity to demonstrate the following 5 aspects:

1. Critical thinking in problem identification
2. Data engineering skills across heterogeneous sources
3. ML systems thinking beyond model training
4. End-to-end product development from idea to working prototype
5. Technical communication at a professional standard

Learning Objectives

With successful completion of this project, you will:

- Identify meaningful, real-world ML problems within a structured domain
- Curate, clean, merge, and engineer features from multiple heterogeneous datasets
- Design and execute a complete ML pipeline with baseline comparisons
- Build a functional end-to-end system (not just a notebook)
- Communicate technical work using professional academic writing standards
- Work effectively in a team on a multi-week technical project

Project Themes

You must select **ONE** of the following four themes. Each theme provides domain structure while allowing creative problem formulation within that domain.

No.	Theme	Domains
1	*Urban Intelligence & City Futures*	*Cities, Environment, Mobility, Infrastructure, Public Services*
2	*Climate Resilience & Extreme Event Forecasting*	*Climate Anomalies, Extreme Weather, Environmental Risks, Disaster Preparedness*
3	*Health & Public Safety Intelligence*	*Public Health, Disease Surveillance, Emergency Systems, Urban Safety*
4	*Sustainable Agriculture & Food Security*	*Smart Agriculture, Crop Monitoring, Climate Stress on Food Systems, Resource Optimization*

T1: Urban Intelligence & City Futures

Some problem space examples in this theme include (i) predicting near-future urban flooding hotspots by combining rainfall forecasts, elevation maps, drainage infrastructure, historical flood events; (ii) forecasting air quality (PM2.5, PM10, NO_2) under varying traffic patterns, meteorological conditions, industrial activity; (iii) identifying high-risk road segments for traffic accidents under different temporal, weather, event conditions; (iv) predicting emergency service demand (ambulance, fire) based on city activity patterns and public events; (v) estimating neighborhood-level electricity demand anomalies for grid stress early warning.

Note: *The aforementioned examples are just to kickstart your brainstorming sessions within the group. Do not directly copy from these examples.*

Possible Data Sources:

- OpenAQ (air quality)
- City open data portals like NYC Open Data, London Datastore, Singapore Data.gov
- OpenWeatherMap or Meteostat
- CHIRPS (precipitation)
- OpenStreetMap (infrastructure, roads)
- Government traffic / accident datasets
- Local utility or energy datasets

T2: Climate Resilience & Extreme Event Forecasting

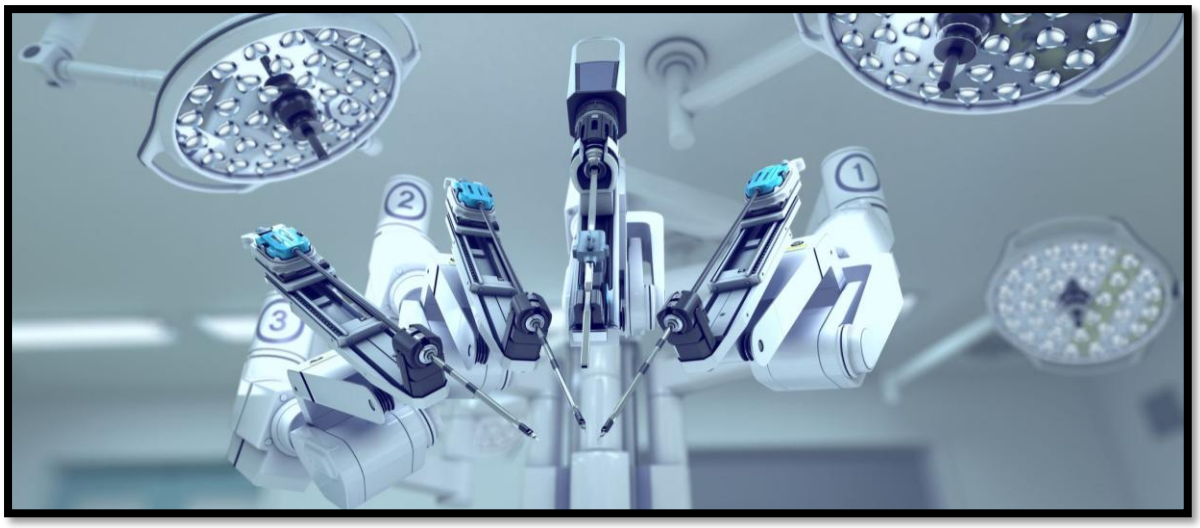
Some problem space examples in this theme include (i) short-term heatwave intensity prediction for vulnerable urban / rural populations; (ii) drought stress prediction for specific regions using multi-source climate indicators (rainfall, evapotranspiration, soil moisture, vegetation indices); (iii) river flow anomaly forecasting to support flood early warning systems; (iv) extreme rainfall event clustering and prediction for flood-prone regions; (v) wildfire risk zone identification using climate variables, vegetation dryness (NDVI), topography, and historical fire data.

Note: *The aforementioned examples are just to kickstart your brainstorming sessions within the group. Do not directly copy from these examples.*

Possible Data Sources:

- ERA5 Climate Reanalysis (temperature, humidity, wind, etc.)
- CHIRPS (precipitation)
- NASA POWER (solar, temperature, humidity)
- Pakistan Meteorological Department (if accessible)
- Global SPEI Database (drought index)
- MODIS NDVI or Google Earth Engine
- FAO Climate Data

T3: Health & Public Safety Intelligence



Some problem space examples in this theme include (i) predicting dengue/malaria outbreak severity across districts using weather patterns, mobility data, historical case counts; (ii) forecasting smog-related respiratory health risk by region using air quality and population density; (iii) predicting ambulance dispatch demand by time-of-day, day-of-week, weather, city events; (iv) identifying hospital overcrowding risk windows using historical admissions, seasonal patterns, current city activity; (v) classifying neighborhoods by public safety risk using crime data, lighting infrastructure, foot traffic proxies, and other possible/useful socioeconomic indicators.

Note: *The aforementioned examples are just to kickstart your brainstorming sessions within the group. Do not directly copy from these examples.*

Possible Data Sources:

- WHO Disease Outbreak News
- Health open data (Pakistan, India, other countries)
- Google Community Mobility Reports
- City-level crime datasets (NYC, Chicago, London, etc.)
- Weather APIs
- Pakistan Bureau of Statistics
- Traffic and road safety datasets

T4: Sustainable Agriculture & Food Security

Some problem space examples in this theme include (i) predicting crop heat stress risk across regions for the upcoming growing season; (ii) identifying crop disease outbreak risk using humidity, temperature, rainfall patterns, vegetation health trends; (iii) forecasting district-level crop production anomalies for food security planning; (iv) predicting irrigation demand anomalies to assist water resource managers; (v) clustering districts/regions by multi-dimensional food insecurity risk using climate, soil, production, and market data.

Note: *The aforementioned examples are just to kickstart your brainstorming sessions within the group. Do not directly copy from these examples.*

Possible Data Sources:

- NASA Earthdata (MODIS, NDVI)
- FAOSTAT
- Pakistan Bureau of Statistics (Agriculture)
- ERA5 Climate Data
- SoilGrids (soil properties)
- USGS Earth Explorer
- World Bank Climate Data

Project Requirements

(a) Group Formation

Groups of 3 or 2 students are allowed. If someone wants to take on the project individually, that is also possible. However, no group should exceed the limit of 3 members. All groups must submit their group details to the class representative by 21st November 2025. The class representative is to share the finalized list of groups with the instructor by 23rd November 2025. This list should include project title, chosen theme, names (of all group members), student IDs (of all group members), and abstract of the problem each group will be working on.

(b) Problem Identification & Dataset Curation

Your problem must be real and currently unsolved or under-addressed. It should be specific and well-scoped (not vague like "predicting weather"). Try to identify clear stakeholders who would benefit from the solution you will be working towards. Do not consider a trivial Kaggle competition problem already solved thousands of times. The problem itself should be feasible within the course timeline and your skillset.

Your datasets must come from at least 3 to 4 different sources. All data sources should be publicly available and properly cited. They should complement each other. For example (weather + traffic + accidents) or (climate + soil + crop yield). Data sources should ideally require meaningful integration / merging (temporal alignment, feature engineering). Keep in mind that the datasets need to be sufficiently large and complex (no toy datasets are to be considered). You must document the source, format, size, temporal/spatial coverage of each dataset, how you accessed/downloaded it, pre-processing steps required, feature engineering decisions, and justification for dimensionality reduction (if applied).

(c) Machine Learning Pipeline

Component	Requirement
Baseline Model	*Simple heuristic or statistical baseline (mean, median, last-value, moving average, etc.)*
Classical ML Models	*At least 2 to 3 models appropriate to your problem (Linear/Logistic Regression, Decision Trees, Random Forest, SVM, Naive Bayes, k-NN, clustering methods, etc.)*
Ensemble / Advanced Model	*At least 1 ensemble or hybrid method (Random Forest, Gradient Boosting, Stacking, Voting Classifier, etc.)*
Evaluation	*Use appropriate / relevant / useful evaluation metrics for your task.*
Comparison	*Clear comparison table + discussion of trade-offs (accuracy vs. interpretability vs. training time vs. deployment complexity)*

Train, validation, and test splits should be properly implemented. Apply cross-validation where appropriate. Document hyperparameter tuning accordingly. Make sure to apply feature importance analysis (at least for tree-based or linear models). Do not forget to perform error analysis. It should

address what does you model get wrong and why. Try to discuss the weaknesses of your models / approach as well.

(d) Proof of Concept (PoC) Implementation

A Jupyter notebook is NOT enough for this project. You must build a working, end-to-end system that demonstrates your solution / model / approach in action. Acceptable PoC formats can be found in the table below. You must choose at least one type or may combine multiple types.

PoC Type	Description	Tools
Web Dashboard	*Interactive visualization with predictions, maps, time-sliders, filters, etc.*	*Streamlit, Plotly Dash, Flask + HTML/CSS/JS*
Interactive Map Application	*Geospatial predictions displayed on a map*	*Folium + Flask, Leaflet.js, Streamlit + Pydeck*
CLI Tool with Visualizations	*Command-line interface that takes inputs and generates prediction reports/plots*	*Python Click/Argparse + Matplotlib/Seaborn*
Risk Monitoring App	*Tool that displays risk scores, alerts, trends over time*	*Streamlit, Dash, Gradio*
What-If Simulation Viewer	*User can adjust input parameters and see how predictions change*	*Streamlit sliders, Dash callbacks*
Data Explorer with Prediction Interface	*Users can explore datasets AND get predictions for selected conditions*	*Panel, Voila, Streamlit*

PoC must include user input mechanism, model prediction output, visualization component, some form of basic error handling, and a reproducible setup (README with installation/run instructions, requirements.txt, environment.yml). Record a 3- to 5-minute demo video showing your PoC in action. Walk through different use cases in that demo. Explain what the user sees and how the ML model supports the prediction. Upload to YouTube / Drive and include the link in your project report.

(e) Project Report

Your report must be written in **LaTeX** using **Overleaf**. It should include project title, details of group members, abstract, introduction, problem definition, data acquisition & preparation, feature engineering, detailed methodology, experimental setup, results, analysis, proof of concept, discussion, future work & limitations, conclusion, relevant references, and appendix (optional).

There is no strict limit on the length of the project report. All figures and tables must be properly captioned and referenced in text. Equations (if any) must be properly formatted using LaTeX math mode. Code snippets (if included) should use *lstlisting* or *minted* package. Properly cite all sources (datasets, papers, libraries, external code).

Note: *There is zero tolerance policy for plagiarism. All text must be original or properly quoted/cited. Use of AI writing assistants to generate report text is not allowed. You may use AI for brainstorming, debugging code, understanding concepts, but the report writing itself must be your own. Plagiarism will result in automatic failure of the project and potential disciplinary action.*

(f) Codebase Submission

Each group is to submit a GitHub repository (or ZIP file) containing a readable, well-commented code. The codebase should follow a modular structure (not one giant 2000-line notebook). It must be reproducible in nature. Anyone should be able to clone your repo and reproduce your results with minimal effort. README.md must be present with clear setup and run instructions. No hardcoded absolute paths are to be included. Always use relative paths.

Evaluation Rubric

Problem Identification & Novelty	----------	***15***
Dataset Curation & Integration	----------	***10***
ML Pipeline & Experimentation	----------	***20***
Proof of Concept (PoC) Implementation	----------	***15***
Project Report Quality	----------	***15***
Codebase Quality & Reproducibility	----------	***10***
Presentation (Final)	----------	***15***

Resources

LaTeX & Overleaf

- IEEE Conference Template on Overleaf
- Learn LaTeX in 30 Minutes
- How to cite in IEEE format

PoC Development

- Streamlit Documentation
- Plotly Dash
- Flask Quickstart
- Folium (Maps in Python)
- Seaborn
- Gradio

Additional Dataset Sources

- Google Dataset Search
- Kaggle Datasets
- UCI ML Repository
- Data.gov
- Our World in Data

Writing & Citation

- How to Write a Good Abstract
- Tips for Writing Technical Papers
- Zotero Reference Manager
- Mendeley Reference Manager

Note: *Full collaboration is expected and encouraged within your group. Across groups, you may discuss general approaches and share dataset sources, but you must NOT share code, models, or report text. You may use Stack Overflow, documentation, tutorials, AI tools, etc. BUT you must cite any significant code borrowed and understand what it does. You may use ChatGPT/GitHub Copilot/ or any other tool at your disposal for debugging, understanding error messages, brainstorming, writing modular code snippets, etc. You may NOT use them to write your report text or generate full solutions you do not understand.*

At the end of the project, each student will fill out a confidential peer evaluation form as well in which you will rate:

- *Each team member (including yourself) on a scale of 1 to 10 on different metrics*
- *Each of the other groups on a scale of 1 to 10 on their project*

---------- Best of Luck ----------