CSCE 50603-001: Group Project Description

Due 11:59pm Thursday, December 4, 2025

The purpose of the final project is to deepen our exploration of machine learning with real-world data. To do this you will need to write code, run it on the data, make some figures, and write a few pages describing your task, the algorithm(s) you used and the results you obtained. You are free to use any online code or third-party sources as long as it is publicly available. However, please make sure that your own entries are developed by your team members - the purpose of this project is to give you practical experience, so it will not be helpful if you just follow instructions from others.

1 Form teams

Please form teams of 1-3 students who share your interests and with whom you will directly collaborate, and send me the list of team members by October 26.

2 Pick a dataset and tasks

For the dataset and tasks of the project, here are three basic options:

1. **Hotel Booking Demand (H1/H2).** *Kaggle Repository:* `https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand`
   Scope & size: Bookings from a city hotel (H2) and a resort hotel (H1) in Portugal; ∼119,000 rows and ∼32 features.

   Suggested targets & tasks:

   - **Classification:** `is_canceled` (cancellation prediction); `is_repeated_guest`; other classification via `assigned_room_type` vs. `reserved_room_type` (binary "upgraded?" or multiclass).
   - **Regression:** `adr` (average daily rate) and `lead_time`. Optional derived target: total nights = `stays_in_week_nights + stays_in_weekend_nights`.

Design notes/pitfalls: Exclude `reservation_status` and `reservation_status_date` when predicting cancellation to prevent leakage (these features may contain direct implication of the target). Handling high-cardinality fields (`country/agent/company`) needs careful encoding (e.g., use target/frequency encoding other than one-hot encoding). Class imbalance is common for `is_canceled`. Report AUPRC (Area Under the Precision-Recall Curve) and AUROC in addition to accuracy/F1.

2. **US Accidents (since 2016).** *Kaggle Repository:* `https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents`
   About: Large, countrywide traffic accident records from 49 U.S. states, enriched with weather/points-of-interest and temporal context.
   Size/features: Millions of rows (versioned) with ∼47 attributes (timestamps, geolocation, distance, weather, road features, day/night, etc.).

   Suggested targets & tasks:

   - **Classification:** `Severity` (ordinal 1–4). You may treat it as multiclass classification.
   - **Regression:** incident duration (derive: `End_Time - Start_Time`) or `Distance(mi)` at onset.

   Design notes/pitfalls: For predicting severity-at-onset, exclude post-outcome fields (`End_Time`, post-dated `Weather_Timestamp`) to prevent leakage. The dataset is large, and you may use stratified sampling for downsampling. Handling high-cardinality `City/Street/Zipcode` needs careful encoding. Class imbalance is common. Report AUPRC (Area Under the Precision-Recall Curve) and AUROC in addition to accuracy/F1.

3. **Taiwan Credit Card Default.** *Kaggle Repository:* `https://www.kaggle.com/datasets/uciml/default-of-credit-card-clients-dataset`
   Scope & size: Monthly billing and payment records from ∼30,000 Taiwanese credit card clients (2005), each with 23 predictive attributes covering demographics, credit limits, and six months of repayment history.

   Suggested targets & tasks:

   - **Classification:** `default.payment.next.month` (binary next-month default).
   - **Auxiliary tasks:** Categorize clients by credit limit (`LIMIT_BAL`).

   Design notes/pitfalls: Dataset is moderately imbalanced (∼22% defaults). Use stratified sampling and evaluate with AUROC/AUPRC. Normalize highly skewed monetary fields and encode categorical variables (`EDUCATION`, `MARRIAGE`, `SEX`). Consecutive month features (`BILL_AMT1{6`, `PAY_AMT1{6`) are highly correlated, and regularization or feature selection is helpful.

4. **Fashion-MNIST Image Classification.** *Kaggle Repository:* `https://www.kaggle.com/datasets/zalando-research/fashionmnist`
   Scope & size: 70,000 grayscale images (28 × 28) of clothing items from 10 balanced

categories (T-shirt/top, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, ankle boot). Each image provides 784 pixel features and a single label.

Suggested targets & tasks:

- **Classification:** `label` (10-class clothing type).
- **Auxiliary tasks:** Binary grouping (e.g., footwear vs. non-footwear) or dimensionality reduction (PCA, t-SNE) for visualization.

Design notes/pitfalls: Normalize pixel values to [0,1] or standardize by mean/std. Start with MLP baselines before CNNs. Report confusion matrix and per-class F1 since visually similar categories (shirt vs. T-shirt) are often confused. The dataset is well-balanced; accuracy is reliable but still use a validation split to monitor overfitting.

5. **Custom.** Find the dataset and tasks on your own but make sure you select a one that can be executed reasonably before the deadline.

Each team please discuss about what topic you want to work on and send me the topic by October 26.

3 Choose a technique (or more) to explore

Your project should implement one or more machine learning algorithms and apply them to the data. A key point is that you must explore your approach(es); so you must do more than simply download a publicly available package and run it with default settings, or with a few values for regularization. You must at least explore the method fully enough to understand how changes might affect its performance (e.g., tuning hyperparameters), verify that your findings make sense (e.g., using cross-validation), and then use your findings to optimize your performance.

4 Write it up

Your team will produce a single write-up document, at least 3 pages long, describing the problem you chose to tackle and the methods you used to address it, including which model(s) you tried, how you trained them, how you selected any parameters they might require, and how they performed on the test data. Consider including tables of performance of different approaches, or plots of performance used to perform model selection (i.e., parameters that control complexity).

Within your document, please try to describe to the best of your ability who was responsible for which aspects (which models, etc.), and how the team as a whole put the ideas together.

Please write the document in the NeurIPS format. The template can be found here: `https://neurips.cc/Conferences/2023/PaperInformation/StyleFiles`.

5 Submission

You final submission must include your write-up and code. The submission deadline is 11:59pm on December 4, 2025.

6 Presentation

Each team should give a presentation about the project in class before Thanksgiving. It is OK if you have not finish the whole project by the time of presentation. In that case, you need to talk about what you have done and what you plan to do in the remaining time.

7 Grading

I am looking for several elements to be present in any good project. These are:

1. Exploration of at least two techniques in machine learning. For example, using neural networks, support vector machines, or random forests are great ideas; if you do this, explore in some depth the various options available to you for parameterizing the model, controlling complexity, etc.
2. Performance validation. You should practice good form of validation to assess your models' performance, do model selection, combine models, etc. For example, you could use accuracy, precision, recall, and F1 score to assess the performance of classification models. ROC behavior is also a good option. You could plot how the performance changes with values of hyper-parameters. When there are multiple hyper-parameters, you can also use grid search. Your write-up should describe how you assess your models' performance using tables or figures.
3. Adaptation to under- and over-fitting. Machine learning is not very"one size fits all" - it is impossible to know for sure what model to choose, what features to give it, or how to set the parameters until you see how it does on the data. Therefore, much of machine learning revolves around assessing performance (e.g., is my poor performance due to under-fitting, or over-fitting?) and deciding how to modify your techniques in response. Your write-up should describe how, during your process, you decided how to adapt your models and why.

8 Use of Generative AI

You can use Generative AI in the project. However, when incorporating it, please explicitly state in your report the specific purposes for which it was utilized, such as data pre-processing, model selection, performance evaluation, etc. Be aware that using AI may lead to different grading criteria. This might involve potentially exploring a broader range of machine learning algorithms or working on relatively more challenging problems. Specifically, Prompt Engineering is an eligible topic, which is a process of carefully crafting and refining prompts. You can do prompt engineering in your project and demonstrate your prompt design process to get the desired output.