[COMP0214] Introduction to Machine Learning - Coursework 2025

1 Analyzing Retail Fuel Price Dynamics in New South Wales (2016–2025)

Retail fuel prices in New South Wales often exhibit pronounced **price cycles**: sharp increases followed by gradual declines. In this coursework you will clean a retail fuel dataset and build a simple forecasting model that predicts next-day prices for each fuel type sold at that different fuel stations.

1.1 Data Cleaning and Visualization

prepare the data and create visualization:

(i) Cleaning & summary [1 mark]

Report a short summary (rows before/after; date range; list of distinct `FuelCode`s; missing values handled) after:

- Parsing `PriceUpdatedDate` into *date*/*time*; ensuring `Price` is numeric (cents per litre).
- Removing exact duplicates; dropping implausible prices (state a simple valid range and justify).
- Aggregating to a **daily table per fuel**: for each (`FuelCode`, *date*) compute the **daily minimum** price.

(ii) Visualize [2 marks]

Produce **two** clear plots (titles/labels/legends) with a 1–2 sentence observation for one fuel station:

- *Time series:* daily price for **two** chosen `FuelCode`s on the same plot.
- *Distribution:* box (or violin) plot comparing daily prices across all available `FuelCode`s.

1.2 Forecasting Next-Day Prices

Build a **one-step-ahead** forecaster that, for each `FuelCode` and `Fuel Station`, predicts the **next day's** price using only information available up to the current day. Formally, learn a function

$$\hat{y}_{d+1}^{(c,s)} = f_{(c,s)}\big(\mathcal{H}_d^{(c,s)}\big),$$

where c indexes fuel codes, s represents fuel staion and $\mathcal{H}_d^{(c)}$ is the historical data for fuel c up to day d.

(i) Problem setup & data split [2 marks]
Define your forecasting setup clearly (inputs, target, and unit of time). Choose and *justify* a train/validation/test partition suitable for time-ordered data. Describe how you prevent any information leakage from the future into training.

(ii) Model design & training [2 marks]
Design *one* forecasting pipeline and apply it to the data. You may use any models covered in or outside the course. Document all choices (features you construct, model family, key hyperparameters) and your training procedure.

(iii) Evaluation & visualization [2 marks]
Evaluate on the held-out test set and report it. Include one **predicted vs. actual** line plot for any fuel and provide a brief ($\leq$100 words) interpretation of typical errors and any noticeable patterns.

1.3 Application & Limitations

(i) Practical use [1 mark]
Hypothetically, explain how one could use your model to get the cheapest fuel (could be any fuel code) if they must buy fuel every day.

(ii) Limitation & remedy [1 mark]
Identify **one key limitation** of your current forecasting system (e.g., in data, features, model, or evaluation). In $\leq$150 words, explain *how* you would address it, what change you would make, and *how* you would measure whether it improved performance.

2 WiFi Signal for Indoor Localization

We use a WiFi signal dataset for indoor localization. The dataset contains RSSI (Received Signal Strength Indicator) values from multiple access points collected in an indoor environment. The **training set** is provided and you may split for validation purposes. Your goal is to develop models to predict the robot's X, Y coordinates based on the RSSI data. The goal of this coursework is to develop models for indoor localization using the provided data and analyze their performance. The following provided dataset files will be used in this question:

- train_data_WIFI.csv

Please note that you will not have access to the data in the test set, as it is held out and your developed method will be eventually evaluated on the test set by us.

Table 1: Columns in `train_data_WIFI.csv`.

Column	Type / Unit	Example
`x`	float (m)	12.43
`y`	float (m)	-3.87
`rssi`	list of int (dB)	`[-73, -67, -81]`
`mac_addrs_idx`	list of int (ID)	`[102, 17, 45]`

`rssi` and `mac_addrs_idx` are **one-to-one aligned** in each row and have the **same length**, which can vary across samples depending on whether the robot is close enough to the routers.

2.1 Multi-Linear Regression Model for Indoor Localization

We start from the very basic model and your tasks include:

i. Develop a multi-linear regression model to predict the X and Y coordinates based on the RSSI values (unit: dB) transmitted by the access points (i.e., beacons). [1 mark]

ii. Report your model's performance on the validation set using the Mean Squared Error. [1 mark]

Note that there are *NaN* values in the training set. You are expected to address it by yourself.

2.2 Neural Network Model for Indoor Localization

We then move to using more complicated models to process the data. Your tasks include:

i. Develop a neural network (NN) model to predict the X and Y coordinates from the RSSI data. You are free to decide the architecture of the NN based on your own exploration and analysis. [2 marks]

ii. Report the validation results of your NN model and compare it with the linear regression model. [2 mark]

2.3 Dimensionality Reduction for Feature Visualization

Let's dive deep into what the Neural Network learns. Your tasks include:

i. Extract the feature representations from a layer in the NN that you think can be mostly useful. Without using the labelled coordinates, apply a clustering method on the extracted features and provide your findings. [2 marks]

ii. Apply a dimensionality reduction method (e.g., t-SNE, PCA) to visualize the feature space in 2D. Provide insights into the learned feature representations and their relation to the localization task.[2 marks]

2.4 Prediction and Analysis on Test Data

Finally, let's evaluate your models with our held-out test set. Your tasks include using the test data provided separately, predict the X and Y coordinates with the model you decide to use. You can only choose one model and report one test prediction file. [3 marks]

Deliverables

You are required to provide the following deliverables:

- Code: Submit a fully-commented code file in the format of **Jupyter Notebook.** Feel free to submit TWO Jupyter Notebooks for Q1 and Q2 separately.
- Report: Provide ONE report (maximum 8 pages, font size 10pt) detailing the answers to all the questions asked in this coursework.
- Predicted Coordinates: Submit a **.txt** file containing the predicted X, Y coordinates for the test data (as required in Task 1). The file should:
  - Contain only two columns, X and Y.
  - The first row should include the headers: X, Y.
  - Each subsequent row should contain the predicted coordinates, with X and Y values separated by a **comma**.
  - Name the file using your student number (e.g., zcxxxx.txt).

  Deadline: 12/12/2025 16:00.

Marking Scheme of Q2.4

After submitting your predictions (in the .txt file), we will provide you with the results for both the Mean Squared Error (MSE) and the Mean Percentage Error (MPE). The MPE between predicted $(\hat{x}, \hat{y})$ and true (x, y) values for M test sample is calculated in this way:

$$\text{MPE} = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{2}\left(\frac{|\hat{x}_i - x_i|}{|x_i|} + \frac{|\hat{y}_i - y_i|}{|y_i|}\right)$$

Marks will then be given by mapping your MPE to a 0–3 scale.

MPE Range	**Marks**
0%–5%	4.0
5%–15%	3.0
15%–25%	2.0
25%–35%	1.5
35%–45%	1
$> 45\%$	0.5 (for ack. submission)