School of Computing, Engineering and Built Environment

Department of Computing

Artificial Intelligence and Machine Learning

Module Code: MMI226824

Coursework 1-1st diet

Issued: 31st October 2023

This coursework comprises 50% of the overall mark for the module.

This coursework is to be submitted electronically via GCULearn, no later than:

Hand-in date: Sunday 10th December 2023 (end of day)

An average student should be able to complete this assignment in about 20 hours of work.

Attention is drawn to the university regulations on plagiarism. Whilst discussion of the coursework between individual students is encouraged, the actual work has to be undertaken individually. Collusion may result in a zero mark being recorded for the coursework for all concerned and may result in further action being taken.

COURSEWORK 1

This module has two coursework components (CW1 and CW2), this is CW1 accounting for 50% of the module's mark. The pass mark for each coursework element is 50% (although a mark over the 45% threshold in one coursework can be compensated by a mark over 50% in the other, if the average results at least 50%).

In this coursework (CW) you will implement the first six steps of the Machine Learning Pipeline from as shown in the book *"AI with Python"* by Artasanchez and Prateek [1], as shown in Figure 1.

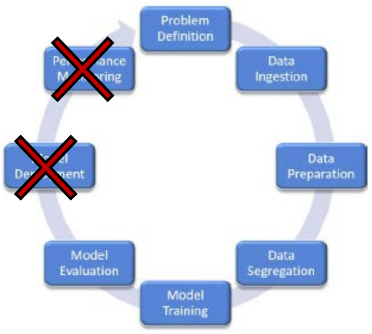


Figure 1: First six steps in a Machine Learning Pipeline [1]

Data

You will use a **modified** version of an air quality dataset [2] which is posted on **GCU Learn** (***AirQuality.zip***). Please ensure to **use the modified version available on GCULearn**.

The dataset contains 9358 instances of hourly-averaged responses from an array of 5 metal-oxide chemical sensors embedded in an air quality chemical multi-sensor device. The device was located on the field in a significantly polluted area, at road level, within an Italian city. Data were recorded from March 2004 to February 2005 (one year).

- Hourly-averaged **raw sensor data** (as integer values). Those are the variables of interest.
- **Ground-truth** (GT): hourly-averaged concentrations for Carbon Monoxide (CO, in mg/m^3), Non-Methanic Hydrocarbons (NMHC, in μg/m^3), Benzene (in μg/m^3), Total Nitrogen Oxides (NOx, in part per billion -ppb-), and Nitrogen Dioxide (NO2, in μg/m^3). These are provided by a co-located *reference certified analyser*.
- Evidences of *cross-sensitivities* as well as both concept and sensor *drifts* are present as described in De Vito et al. [3], eventually affecting sensors concentration estimation capabilities.

Target tasks

This CW will require you to outline a detailed problem definition, ingest, prepare, and segregate the data, for subsequent training and evaluation of two **machine learning models** to:

1. **Predict the CO concentration** (in mg/m^3) based on, at least, the PT08.S1(CO) raw sensor readings, day of the week and time. Maybe temperature and humidity can play a role as well? Use CO(GT) as the ground truth.
2. **Define your own Air Quality Index** by combining the ground-truth readings of several gases. Then, use ML to predict your defined Air Quality Index from several **raw sensor readings** and other columns of interest (obviously without using the ground truth columns).

You are required to submit a Jupyter Notebook based on the template available on GCU Learn (***CW1_template.ipynb***).

Jupyter Notebook Contents

The various sections in the notebook should include code, in-code comments and **appropriate Markdown cells** describing your approach chosen and discussing the results.

In detail, sections should include the following:

1. Introduction and Problem Definition

- Textual description providing an overview over the data
- A discussion on why the problems described in the tasks below are regression or classification ones
- A detailed problem statement question as discussed in Lecture 3.

2. Data Ingestion

- Code to load the data into a suitable format to be used in the notebook
- A description of the statistical data types for each field in the file "*AirQuality.csv*"

3. Data Preparation

You should assume that exploratory data analysis has taken place and the following was concluded:

- Last 2 columns are empty
- Missing values (due to sensors or comms malfunctions / out of battery) are indicated with a value of **-200** (not as NaN). There are missing values in all columns; the number of missing values is relatively small in some columns, but massive in the NMHC(GT) column
- The peak road usage hours are 8-12AM and 6-10PM on working days and, to a lesser degree, 9am-12am on non-working days.
- Valley road usage is during the central hours of the night (2-6am)

Your data preparation steps should therefore include the following:

- Drop the empty columns, and any other column that you won't need for your analysis (reasoning why)
- Replace the "-200" values by `np.nan` for the correct operations of the usual functions
- Create a new attribute (column) indicating the day of the week, for instance using:
  `df["DayOfWeek"] = pd.to_datetime(df["Date"], dayfirst=True).dt.day_name()`
- Create a new field that indicates whether it is a peak time or not
- Create a new field that indicates whether it is valley time or not

The following steps will need to be done independently for **Task 1 (CO prediction)** and for **Task 2 (your Air Quality Index prediction)**.

4. Further data preparation, and Data Segregation

- Further dropping of the columns that won't be used for this task 1
- Automatically fill in the missing values in the columns that you are going to use, with values that would most closely mirror the actual missing values (explain why)
- A justification (and potential application) of whether you should use data binning or not
- Suitable encoding of the data
- Code and justification for the selection and application of a suitable data split

- For Task 2, create a new column for your personal Air Quality Index (that would be the target column to predict)
- The above steps, and your segregation, should be different for Task 1 and Task 2.

5. Model Training

- For each Task, selection of one Regression model and justification why it is suitable for the task. You must use a different Regression model for each Task.
- Only one of those models should be Tree-Based (e.g. Random Forest or Decision Tree)
- Application of those models with default parameters as a baseline on the data
- Utilisation of manual or automatic hyperparameter optimization and justification of your choices to create "optimized" versions of each regression model

6. Model Evaluation

- For each Task, selection of appropriate regression metrics and a written outline why they are suitable for this data
- A comparison of the baseline models to the "optimized" versions and an evaluation of the results

7. Conclusion

- For each Task, a conclusion and interpretation of the results and suggestion of potential improvements

You should provide **sufficient written documentation** in the notebook **Markdown cells** to show that you understood and have justified the steps that have been implemented. Marks may not be awarded if the code has insufficient explanations and the information provided is just contained within a few code cells.

The maximum word limit for all Markdown cells (excluding inline code comments or the "Sources" Markdown cell) is **1500 Words**.

You should use machine learning libraries such as ***scikit-learn*** to assist your implementation and training of algorithms. However, **copying code steps** from external sources which have used the same, or similar dataset is not permitted and may lead to zero marks being awarded for this particular section. You should write the code yourself and demonstrate your understanding using the textual explanations you provide.

Plagiarism

You should also pay attention to the university's codes and practices[1] as well as their plagiarism regulations[2]. Any kind of content (images, text, code) that was copied from any source and used in your coursework **without acknowledging of the source** is bad academic practice and could fall under plagiarism.

The discussion of coursework between students is encouraged, but **the work must be undertaken individually**. Collusion (copying work between students) may result in a zero mark being recorded for everyone involved and further action being taken.

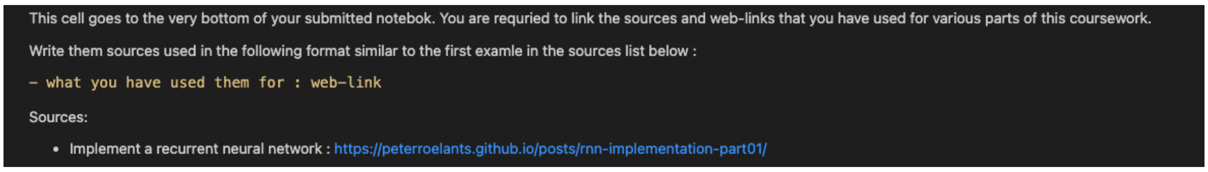
[1] https://www.gcu.ac.uk/academicquality/regulationsandpolicies/universityassessmentregulationsandpolicies/
[2] https://www.gcu.ac.uk/library/smile/plagiarismandreferencing/

Sources

To avoid plagiarism or poor academic practice, you need to ensure that you specify where you obtained any material you use and how you have modified it.

This MUST include **specific web addresses** (not just *google.co.uk* or similar).

A template for this is included on at the very bottom of the supplied template notebook and shown in Figure 2.



This cell goes to the very bottom of your submitted notebok. You are requried to link the sources and web-links that you have used for various parts of this coursework.

Write them sources used in the following format similar to the first examle in the sources list below :

```
- what you have used them for : web-link
```

Sources:

- Implement a recurrent neural network : https://peterroelants.github.io/posts/rnn-implementation-part01/

Figure 2: "Sources" markdown cell to be included at the very end of the submission

Coursework Submission

You are expected to submit your Jupyter Notebook with your code and markdown cells, using the following **naming** scheme:

CW1_LastName_FirstName_StudentID.ipynb

For instance, if your name is Nicola Sturgeon with the StudentID S1234567, name the file:

CW1_sturgeon_nicola_S1234567.ipynb

Coursework files are submitted using GCU Learn.

Marking Scheme

The marking scheme which will be used to assess the coursework is appended at the end of this document.

References

[1] Artasanchez, Alberto, and Prateek Joshi. Artificial Intelligence with Python: Your complete guide to building intelligent apps using Python 3. x. Packt, (2020)

[2] "Air quality dataset". https://archive.ics.uci.edu/dataset/360/air+quality

[3] S. De Vito, E. Massera, M. Piga, L. Martinotto, G. Di Francia, "On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario", Sensors and Actuators B: Chemical, Volume 129, Issue 2, 2008, Pages 750-757, ISSN 0925-4005, https://doi.org/10.1016/j.snb.2007.09.060.

Assessment Rubric

	80-100	70-79	65-69	60-64	50-59	40-49	1-39	0	Mark
	Exceptional	Excellent	Very Good	Good	Satisfactory	Marginal Fail	Clear Fail	N/S	
Criteria	Demonstrates **exceptional or consistently excellent** ability, skills and behaviours across specified items	Demonstrates **mostly excellent** ability, skills and behaviours across specified items.	Demonstrates **overall very good** ability, skills and behaviours across specified items.	Demonstrates **overall good** ability, skills and behaviours across specified items	Demonstrates **overall satisfactory** ability, skills and behaviours across specified items	Demonstrates **overall poor** ability, skills and behaviours across specified items with **some satisfactory** elements	Demonstrates **overall poor** ability, skills and behaviours across specified items with **no satisfactory** elements		
Introduction and model definition (10%) *Clear intro and problem definition.* *Clear explanation of why each task is a regression or a classification problem.*					Basic description given in the booklet plus some additional contribution.				
Data ingestion and preparation (15%) *Sensible data ingestion and preparation steps have been included and clearly explained.*					Data is correctly loaded. *Most* requirements in booklet are met, using the most basic approach.				
Task 1 (30%) • *Further task-specific data preparation* • *Data segregation* • *Model training* • *Model evaluation*					CO concentration prediction (mg/m^3) from raw CO sensor data only				
Task 2 (30%) • *Further task-specific data preparation* • *Data segregation* • *Model training* • *Model evaluation*					Air quality index defined and predicted using only 2 variables. Defined air quality index might be not relevant health-wise.				
Conclusions (15%) *Detailed and meaningful concluding statements*					Summary of results for each task with little critical discussion. No possible proposed improvements.				
Total									