COMP9414 25T2
Artificial Intelligence

Assignment 1 – Artificial neural networks

Due: Week 5, Friday, 4 July 2025, 5 pm.

1 Problem overview

The Amazon rainforest is the largest tropical rainforest on Earth, renowned for its unparalleled biodiversity and vital role in regulating global climate patterns [1, 2]. It spans over 5.5 million km^2 and acts as a critical carbon sink by absorbing substantial amounts of atmospheric CO_2, thus mitigating global climate change [2]. However, recent decades have witnessed a concerning rise in temperatures across the Amazon basin, attributed primarily to climate change [3].

Within this vast rainforest (contoured in green in Figure 1), specific regions have experienced notably rapid increases in temperature, leading to more frequent and intense "hot events" [4]. These hot events threaten local ecosystems by increasing the risk of forest fires and have broader implications for global climate stability.

The occurrence and intensity of these hot events in the Amazon are influenced by large-scale climate drivers that modulate weather patterns worldwide [5]. In particular, four significant oceanic climate modes, El Niño Southern Oscillation (ENSO), Tropical South Atlantic (TSA), Tropical North Atlantic (TNA), and North Atlantic Oscillation (NAO), play crucial roles in determining temperature variations within the Amazon region. Each of these climate modes occurs in different oceanic regions surrounding the Amazon, as highlighted in Figure 1.

In this assignment, your task is to build a neural model using TensorFlow/Keras to predict monthly temperature and the occurrence of hot events in a specific Amazonian region, indicated in red in Figure 1. You will use monthly time-series data from 1982 to 2022, comprising temperature observations from the Amazon and indices representing the ENSO, TSA, TNA, and NAO climate modes.

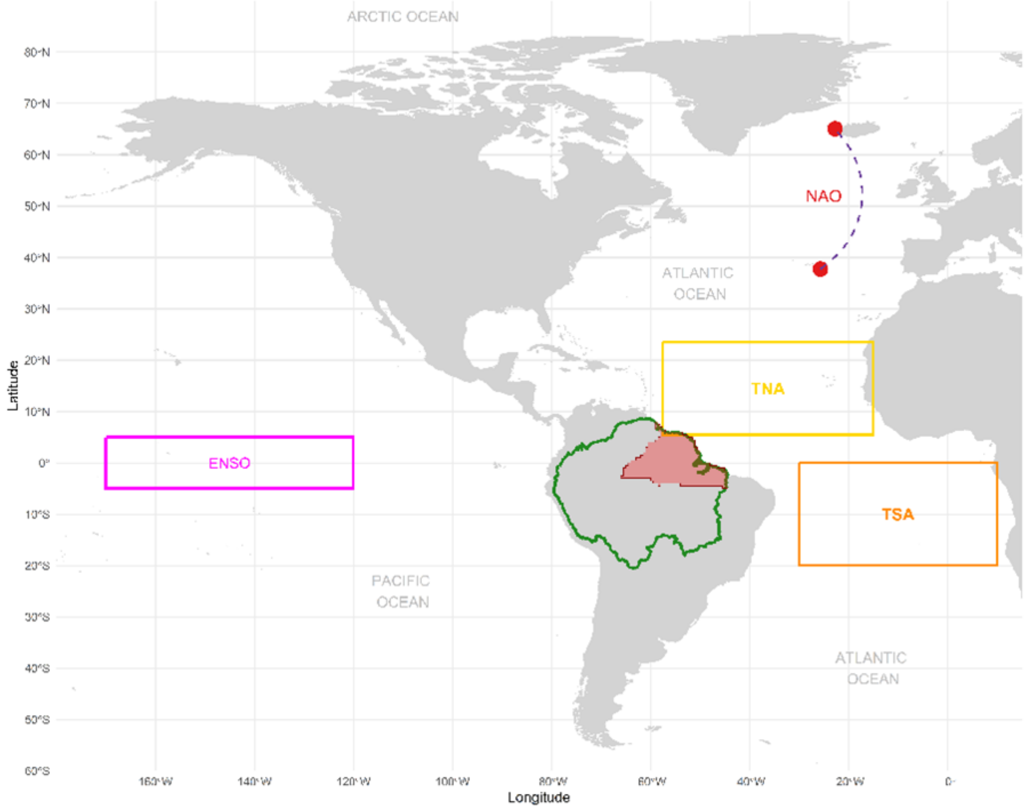

Figure 1: Map of the Amazon basin showing the study area (red) and the four surrounding climate-driver domains.

2 Provided Data

- Temperature time-series data and the values of the indices of climate modes ENSO, NAO, TSA and TNA as shown in Table 1 (`Amazon_temperature_student.csv`).
- Temperature thresholds provided for each month of the year (`threshold.csv`).

Table 1: Climate-mode indices: measurement, range and interpretation

Index	Measurement	Range	Interpretation
ENSO	Sea-surface temperature anomaly in Niño 3.4 region	−3 to 3 °C	+→ El Niño; −→ La Niña
NAO	Normalised sea-level pressure difference (Azores – Iceland)	−4 to 4	+→ stronger westerlies, milder winters; −→ reverse
TSA	SST anomaly in Tropical South Atlantic	−1 to 1 °C	+→ warmer South Atlantic waters
TNA	SST anomaly in Tropical North Atlantic	−1 to 1 °C	+→ warmer North Atlantic waters

3 Objective

Your goal is to build neural network models for two distinct tasks:

- **Task A (Classification):** Predict whether a hot event occurs.
- **Task B (Regression):** Predict the temperature.

---

4 Task A: Classification (Hot event detection)

Data preparation

(a) Define a binary variable named `Hot`: A month is classified as Hot if the monthly temperature exceeds the provided threshold for that specific month.

(i) Set `Hot` = 1 if the monthly temperature exceeds the monthly threshold.

(ii) Otherwise, set `Hot` = 0.

(b) Create a bar plot that summarises, for each year, the number of hot months.

Model development

(c) Randomly partition the dataset into training, validation, and test sets.

(d) *Pre-processing:* Apply any necessary transformations to the training set, then apply the same transformations to the validation set. Keep a record of all applied transformations.

(e) Build a neural network classifier to predict the occurrence of hot events: define the architecture and hyper-parameters (loss function, optimiser, batch size, learning rate, number of epochs). It is recommended that the total number of trainable parameters obey

$$N_{\text{params}} < \frac{N_{\text{samples}}}{10},$$

i.e. keep the parameter count below one-tenth of the sample size.

(f) Create a plot showing the accuracy (y-axis) versus the number of epochs (x-axis) for both the training and validation sets.

Model evaluation

(g) Apply the same transformations to the test set as you did to the training and validation sets.

(h) Use your model to predict the class `Hot` on the test set.

(i) Evaluate performance by plotting a confusion matrix. Note that your positive class is 1.

(j) Calculate: Balanced Accuracy, True Negative Rate (Specificity), and True Positive Rate (Sensitivity).

---

5 Task B: Regression (Temperature prediction)

For this task, directly predict temperature values rather than using the binary `Hot` variable.

Model development

(k) Randomly split the dataset into training, validation, and test sets.

(l) *Pre-processing:* Apply any necessary transformations to the *input features* of the training set, and replicate exactly the same transformations on the validation set. **Do *not* scale, normalise or otherwise transform the ground-truth (target) values.** Keep an explicit record of every transformation applied.

(m) Build your model by defining its architecture and training hyper-parameters: loss function, optimiser, batch size, learning rate and number of epochs. It is recommended that the total number of trainable parameters be less than one-tenth of the number of samples.

(n) Create a plot showing the values of the loss function (y-axis) versus the number of epochs (x-axis) for both the training and validation sets.

Model evaluation

(o) Evaluate the regression model by comparing the true and predicted temperature values on the test set. Use the Pearson Correlation Coefficient (r) and the Mean Absolute Error (MAE) as evaluation metrics. The metrics are defined as

$$r = \frac{\sum_{i=1}^{n}(y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}\sqrt{\sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})^2}}, \qquad \text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|,$$

where y_i is the true temperature, $\hat{y}_i$ is the predicted temperature, and n is the number of test samples.

Model development (year-wise split & target normalisation) [6]

(p) Split the data by whole calendar years: each year must appear in one subset only. Use the same proportions as in part (k) for train/validation/test. Fit a *separate* scaler to the training-set temperature targets and apply it to the validation and test targets; do *not* reuse the feature scaler. Retain the feature transformations specified in part (l) unchanged.

(q) Work with the train/validation/test partitions—and the target–scaler—already established in part (p). Now apply every remaining feature-level pre-processing step specified in part (l) to those same splits.

(r) Re-train the *same regression network and hyper-parameters defined in (m)* on the year-wise split (no new architecture).

(s) Create a plot showing the values of the loss function (y-axis) versus the number of epochs (x-axis) for both the training and validation sets.

Model evaluation (Year-wise Split)

(t) Evaluate the regression model by comparing the true and predicted temperature values on the test set. Use the Pearson Correlation Coefficient (r) and the Mean Absolute Error (MAE) as evaluation metrics.

---

6 Additional notes (apply to both tasks)

- You need to set a random seed to ensure that your results are reproducible.
- You must serialise both versions of every trained model *and* the corresponding feature-scaler objects feature scalers (and the separate target scalers from part (p) where applicable) for the random split **and** for the year-wise split.
- You cannot use `year` as a predictor. However, you may choose whether or not to include `month` as a predictor. If you include `month`, use cyclic encoding to represent it properly, ensuring your neural network correctly understands the continuity between December and January. To achieve cyclic encoding: Normalise the month to the range $[0, 2\pi]$ using: month normalised $= 2\pi \times \frac{\text{(month-1)}}{12}$.
- In the year-wise split, the years selected for training, validation, or testing do not need to be consecutive. For example, data from 1982 may belong to the training set, 1983 to the validation set, and 1984 to the test set.

7 Testing and discussing your code

Your notebook will be exercised live during the tutorial-based discussion (25 marks in total); Attendance is compulsory; submissions not discussed will receive zero.

You must build and save neural models for *both* tasks (classification and regression) under the random split and the year-wise split. For full marks you must include, **within the same Jupyter notebook**, a single code cell that

(i) loads the hold-out *hidden test dataset* dataset provided by the course tutors;

(ii) restores every serialised artefact:

- classifier model & its feature scaler,
- random-split regressor & feature scaler,
- year-wise regressor & *both* its feature scaler and its separate target-scaler;

(iii) evaluates the models, printing all required metrics *and*

- the confusion matrix for the classifier;
- a "true vs predicted" scatter plot for the random-split regressor;
- the same scatter plot for the year-wise regressor (after inverse-transforming the targets with its own scaler).

Code readability is worth 1 mark. During the discussion your tutor will award up to 8 marks per task based on your understanding: **8** Outstanding, **6** Great, **4** Fair, **2** Low, **0** Deficient/No answer.

8 Submitting your assignment

The assignment must be done individually. You need to submit your solution on Moodle. Your submission must include:

- A single Jupyter notebook (`.ipynb`).
- the serialised trained models for the classification and regression tasks;
- all associated scaler objects (feature scalers for every model and the separate target–scaler used in the year-wise regression).

The first line of your Jupyter notebook should display your full name and your zID as a comment. The notebook should contain all the necessary code for reading files, data preprocessing, network architecture, and result evaluation. Additionally, your file should include short text descriptions to help markers better understand your code. Please be mindful that providing clean and easy-to-read code is a part of your assignment.

You can submit as many times as you like before the deadline—later submissions overwrite earlier ones. After submitting your file, a good practice is to take a screenshot of it for future reference.

Late submission penalty: UNSW has a standard late submission penalty of 5% per day from your mark, capped at five days from the assessment deadline; after that students cannot submit the assignment.

9 Deadline and questions

Deadline: Week 5, Friday 4 July 2025, 5pm. Please use the forum on Moodle to ask questions related to the assignment. We will prioritise questions asked in the forum. However, you should not share your code to avoid making it public and possible plagiarism. If that's the case, use the course email `cs9414@cse.unsw.edu.au` as alternative.

Although we try to answer questions as quickly as possible, we might take up to 1 or 2 business days to reply, therefore, last-moment questions might not be answered timely.

For any questions regarding the discussion sessions, please contact directly your tutor. You can have access to your tutor email address through Table 3.

Table 2: Marking scheme for the assignment.

Criteria	**Marks**
Classification Task	
Accuracy–vs–epoch plot (training and validation)	1
Balanced Accuracy and Precision on the **test** set	1
Confusion matrix on the **hidden test dataset** dataset	1
Balanced Accuracy and Precision on the hidden test dataset dataset	1
Demonstrate complete understanding of the classification code and analysis during discussion	8
Regression Task (random-split and year-wise models)	
Loss–vs–epoch plot (training and validation) for *both* regressors	1
True–vs–predicted scatter plot on the **test** set for *both* regressors	1
MAE and Pearson correlation on the test set for *both* regressors	1
True–vs–predicted scatter plot on the **hidden test dataset** dataset for *both* regressors	1
Demonstrate complete understanding of the **random-split** regressor during discussion	6
Demonstrate complete understanding of the **year-wise** regressor (including target scaler) during discussion	2
Overall code readability, tidy structure, well-commented script	1
Total marks	**25**

Table 3: COMP9414 25T2 Tutorials

No.	**Class ID(s)**	**Tutor**	**Email**
1	4374, 4383	Dr Jingying Gao	`jingying.gao@unsw.edu.au`
2	4375, 4376	Kiran Jeet Kaur	`kiran_jeet.kaur@unsw.edu.au`
3	4377, 4381	Leman Kirme	`l.kirme@unsw.edu.au`
4	4378, 4389	Xinyi Li	`xinyi.li17@student.unsw.edu.au`
5	4379, 4385	John Chen	`xin.chen9@student.unsw.edu.au`
6	4380, 4399	Abhishek Pradeep	`abhishek.pradeep@student.unsw.edu.au`
7	4382, 4394	Janhavi Jain	`j.jain@unsw.edu.au`
8	4384, 4393	Maher Mesto	`m.mesto@unsw.edu.au`
9	4386, 4391	Peter Ho	`peter.ho2@student.unsw.edu.au`
10	4387, 4390	Yixin Kang	`yixin.kang@student.unsw.edu.au`
11	4388, 4397	Jonas Macken	`j.macken@student.unsw.edu.au`
12	4392, 4405	Malher Patel	`malhar.patel@unsw.edu.au`
13	4395, 4401	Ramya Kumar	`ramya.kumar1@unsw.edu.au`
14	4398, 4402	Zahra Donyavi	`z.donyavi@unsw.edu.au`
15	4396, 4403	Hadha Afrisal	`hadha.afrisal@unsw.edu.au`
16	4400, 4404	Joffrey Ji	`joffrey.ji@student.unsw.edu.au`

Plagiarism policy

All submitted work must be your own. Collaboration is limited to high-level discussion of concepts; code, text and plots must be authored individually. Suspected plagiarism will be reported in accordance with UNSW Academic Integrity guidelines.

References

[1] W. Milliken, D. C. Zappi, D. Sasaki, M. J. G. Hopkins, and R. T. Pennington. Amazon vegetation: how much don't we know and how much does it matter? *Kew Bulletin*, 65: 691–709, 2010. doi: 10.1007/s12225-010-9236-x.

[2] T. A. West, J. Börner, and P. M. Fearnside. Climatic benefits from the 2006–2017 avoided deforestation in amazonian brazil. *Frontiers in Forests and Global Change*, 2, 2019. doi: 10.3389/ffgc.2019.00052.

[3] D. C. Zemp, C. Schleussner, H. M. J. Barbosa, M. Hirota, V. Montade, G. Sampaio, A. Staal, L. Wang-Erlandsson, and A. Rammig. Self-amplified amazon forest loss due to vegetation-atmosphere feedbacks. *Nature Communications*, 8, 2017. doi: 10.1038/ncomms14681.

[4] F. L. V. Ferreira, A. A. Silva, V. J. d. Santos, and M. L. Calijuri. Deforestation in the legal amazon: the impacts of human action on the regional climate scenario. *Research Square*, 2021. doi: 10.21203/rs.3.rs-669277/v1.

[5] Y. Liu, Z. Li, X. Lin, and J. Yang. Enhanced eastern pacific enso-tropical north atlantic connection under greenhouse warming. *Geophysical Research Letters*, 48, 2021. doi: 10.1029/2021gl095332.

[6] Ó. Reyes and S. Ventura. Performing multi-target regression via a parameter sharing-based deep network. *International Journal of Neural Systems*, 29:1950014, 2019. doi: 10.1142/s012906571950014x.