Assessment Brief – 2024/2025

UNIVERSITY OF SURREY

AI AND SUSTAINABILITY – EEEM073

Coursework Deadline: 14th of May 2025 at 18:00 GMT

Module Aims

Your learning objectives for this module are based on the Module Descriptor, which are to:

- Develop an understanding of the core concepts, complementarities and differences between the sustainability of AI, and AI for sustainability.
- Apply advanced techniques that aim to develop more sustainable AI solutions employing time series processing, data-driven and physic-informed approaches.
- Develop state-of-the-art AI solutions that can support in achieving the United Nations' Sustainable Development Goals (UNSDGs).
- Apply advanced techniques that aim to develop sustainable development employing efficient and trustworthy AI approaches.

Assessment Overview

The coursework is 100% of the module assessment and is **individual** based. You will need to select an open-source dataset related to a sustainability application, and apply AI techniques to tackle a challenge related to that application, based on the topics covered in the lectures and labs. You must write code exclusively in Python using Jupyter Notebooks that read in, process, analyse and visualise the data chosen for this assessment. You must also prepare a report that introduces the problem and its relevance to any of the UNSDGs. This can be related to applications in industry, society, government etc., as long as the overall goal is to address a sustainability challenge (e.g. clean air, clean energy, climate change, food systems, smart and resilient industry, trustworthy AI etc.). The report must detail the data, explain the process and methods used in the code, present the results (from your code) and discussion (based on your objectives, application, data, methods, and results) in a coherent way. A full marking scheme is at the end of this brief. The final report and code must be an original submission.

Deliverables

You will create and deliver:

1- **Report (1 PDF file per student)**

You will produce a concise, non-academic report with a maximum of 25 pages, excluding references and appendices. The report should be succinctly written, making use of tables, figures and charts where appropriate and preferably to summarise ideas, methods and results, making it easier to read and to follow. For this, you must:

a) Provide a problem/application definition and objectives. It will contain an overview of your chosen application and dataset, including a data dictionary.

b) Explore your dataset and explain your choice of data preparation approaches as well as each of the steps. You need to make sure you justify your choices adequately.

c) Document your methodology to achieve your objectives, including how you chose appropriate AI algorithms, evaluation methodology including choice of metrics and other strategies for quantitative and qualitative analysis.

d) Justify your approach, along with your assumptions.

e) Provide the summarised results from your Python code in tables/figures/charts as for a non-academic audience. This must include (1) appropriate data exploration and pre-processing steps, (2) suitable modelling techniques, (3) an appropriate evaluation of your approach and modelling strategies, (4) adequate application of compression techniques to make models more lightweight and efficient while keeping similar performance as the uncompressed models, and (5) your results, discussion, limitations, conclusions and recommendations.

f) The report should be written in single column format using Times New Roman or Arial fonts size 11 or 12, using full justification. It must be concise, objective, coherent and cohesive. Avoid wasting of useful space, adding meaningless content, or adding many plots that do not serve to the purpose of presenting the work as per these guidelines. Every table, figure, plot, chart etc. must have a proper legend/caption that explains its contents – so do not use general text such as "Diagram of the Methodology" or "Training and validation loss". The report is expected to have around 20 pages, but no more than 25 pages, excluding references and appendices. You may choose to move some specific results to appendices as long as they present additional evidence and are not strictly necessary to understand the work or support the findings.

2- The dataset used in the assessment

3- Python code used in the assessment

You are required to write Python codes to meet the project objectives that are set in the report.

- The code must load and process the dataset, undertake your data pre-processing steps, and perform any data exploratory analysis.
- The code must be used to train the models using chosen AI algorithms, evaluate and output results, and provide visualisation as appropriate:
  - Your code must be properly commented to explain the steps and overall process.
  - Source code is provided on Surrey Learn for each individual lab that gives you example implementations in Python. You are welcome to use any/all/none of this Python codes, but must highlight in your code where you do so. You will not get a good mark if you only demonstrate a minimum level of skill by extensively use Python codes copied from other sources.
  - In general, when in doubt, provide a simple reference (at least as a courtesy) where you have taken and used any code from an external source – even if you have amended this. You will need to judge if you believe the code to be "sufficiently different" to the source so that you do not need to reference. If the code is generic or "obvious" then you may consider this unnecessary. Remember to include comments to document your code and choice of libraries. You should comment your own functions following the recommended approach in the labs.
  - Your Python code should output [text] tables and any plots/graphs/visualisations as necessary. These can be later exported and used in your report where appropriate.
  - It is advised to split the code in sequential notebooks that are required to run the whole pipeline, so that once each step/notebook is performed/excecuted, it saves its outputs to be used as input to the next notebook in the pipeline. For instance, you could have the following sequence of notebooks: "1. Data loading and preprocessing.ipynb", "2. Data exploratory analysis.ipynb", "3. AI modelling.ipynb", "4. Assessment and evaluation.ipynb", "5. Model compression.ipynb" etc.
  - You must prepare a README file explaining how your code was structured, what the required dependencies are, and any other information relevant to prepare the environment, run the code and reproduce the results reported. It is not advised to use very specific libraries or code that requires specialised hardware to run the co

Detailed Marking Scheme

Report 50%				
	Title Page (Project Title and Student Name)			
	Section	**Details**	**Weight**	**Marking Criteria**
1	**Introduction:** - Problem definition in AI and Sustainability - Basic data understanding - Establishing the application and setting the objectives	• A background to the general problem and why it is important in the context of AI and Sustainability. • You must give a clear problem statement with objectives, based on your problem, application and dataset, including any assumptions. • In other words, what problem you are trying to solve and why this is important for AI and Sustainability. • What are the major questions you wish to answer? • What techniques do you plan to use? • What data are you looking at and where is it from? • Overview of the data (size, number of features, features' data type, etc) • You must make sure that the dataset (possibly from combined sources) can be used to answer the questions/hypotheses. • Describe your approach in terms of the steps taken specific to your project. You must demonstrate your understanding of the chosen application and dataset, in context of your stated objectives, through proper data loading and initial inspection. These tasks are typically performed to ensure the validity of the data and to gain an initial understanding. Lastly, the student must mention and justify which UNSDGs their work contributes to.	10%	**Dataset/s Relevance, Complexity and Justification:** • The problem/research question, application, project objectives, and chosen techniques should be aligned to the chosen dataset/s and to an application in AI and Sustainability. • The chosen dataset must align with the task requirements and objectives. Assess how well the dataset is suited to answer the analytical questions and provide insights relevant to the task. • The complexity of the dataset in terms of the number of features, size of the dataset, and the range of values in each feature. A more complex dataset should be rewarded for the challenge it poses. • The rationale provided by the students for selecting the application and the dataset. Ensure it aligns with the goals and the potential insights that can be derived from the chosen data.

2	**Data Understanding, Pre-processing and Exploration** - **Data preparation and cleaning** - **Data exploration**	In general, this section focuses on preparing the data to be in a form suitable for subsequent modelling and evaluation. More particularly, you need to perform detailed data preparation and cleaning steps. This will require: • Dataset processing: The steps required to analyse each field/record (such as handling missing data and outliers) and how you might combine features or measure the quality of the dataset. • Perform necessary data transformation normalisation, scaling, encoding, feature selection, feature augmentation and constructions. • Prepare an adequate level of data exploration and visualisations that will provide you the means to better understand the data. This might include descriptive statistics, histogram plots, boxplots, and any other relevant graphical analysis.	10%	**<u>Data Cleaning:</u>** • How effectively the student handled missing data, outliers, and inconsistencies. Deduct points for inadequate or improper handling. **<u>Feature Engineering:</u>** • The creativity and effectiveness of feature engineering. Reward for creating new informative features or transforming existing ones. **<u>Scaling/Normalisation:</u>** • The student appropriately scaled or normalised the features, making sure they understand the importance of this step for various machine learning algorithms. **<u>Data Split:</u>** • The student correctly split the dataset into training, validation, and test sets, ensuring they understand the significance of proper data splitting for model evaluation. **<u>Data Exploration and Visualisation:</u>** • The student adequately analysed the data through a proper and comprehensive exploratory and visualisation analysis.
3	**Modelling** - **Machine learning algorithms implementation**	You are expected to implement and apply at least two supervised ML techniques to perform the modelling tasks according to your objectives, using ML algorithms chosen from those discussed in the labs for this module. In case the student decides to work with physics-informed neural networks (PINN), the two ML	15%	**<u>Model Selection:</u>** • The appropriateness of the chosen models for the given task, considering factors such as complexity, interpretability, and suitability for the problem.

		techniques can be the fully data-driven version, and the PINN version of the same technique. This stage typically requires repeated iterations of modelling that will involve: • The selection of ML modelling algorithms, e.g., choose appropriate algorithms from available Python libraries. • The choice of the ML algorithm parameters/hyper-parameters, e.g. to reduce overfitting and optimise performance you might try a range of values (tuning), adding dropout layers etc. • The training and validation of the ML model(s) using the pre-processed dataset. The design of the modelling tasks must be sound and justified, and each ML algorithm discussed along with its strengths and weaknesses in context of the real-world application in AI and sustainability. Appropriate parameter settings must be used, stated and justified. Inevitably, there will be elements of experimental trial and error.		**Hyperparameter Tuning:** • The effectiveness of hyperparameter tuning, considering how well the student improved model performance through this process. **Model Training and Validation:** • The student's ability to correctly use the training dataset and the correct set of ML models' parameters to train the models. • The ability to correctly use the validation dataset to assess whether the models are generalising well or not (i.e. presenting over/underfitting). • A reward will be given to those who successfully develop PINN to solve a sustainability challenge – when applicable.
4	**Performance evaluation** - **Comparison of models' performance**	The robust evaluation of the modelling results is critical to the success of a project. You must have a clear evaluation approach that is described and justified. The resulting models in your Python code must each be evaluated and compared for their ability to generalise on unseen data using your evaluation approach. The evaluation metrics should be adequate for the task at hand (i.e. classification and regression). Also, it is important to employ explainable AI techniques to	10%	**Model Evaluation:** • The right choice of the relevant evaluation metrics based on the established objectives to assess the performance and efficiency of the models, ensuring the understanding of the fundamentals of model training and evaluation and the concepts of sustainable AI and AI for sustainability.

		understand the reason why the models are taking their predictions. Additionally, the student must compare the sizes and the time required to train to perform inferences (predictions) for each model, discussing their differences/similarities in light of their characteristics and architectures, ensuring that at least one of the models has better computational efficiency than the other.		• The correct application of explainable AI techniques to understand how the models are making predictions.
5	**Model compression** - **Compress the models** - **Compare and evaluate with compressed and baseline models**	The student must compress the models to make them more lightweight and energy-efficient while ensuring similar performance compared to the uncompressed models. For this, to each model, the student should apply at least two compression techniques based on the labs, comparing them with each other and with the uncompressed models. The student must: • Present and discuss the results of compressing the ML models in terms of their performance and size. • Discuss the balance between decreasing model accuracy, improving computational performance when making predictions, and decreasing model size, considering aspects related to the expected accuracy for the selected problem and the sustainability of the AI models. To evaluate their performance and efficiency, the student must use the same evaluation metrics as used in Section 4.	5%	**Model Compression:** • The correct and successful selection and application of the relevant compression techniques to the uncompressed models, aiming to reduce their size and computational time for training/inference, assessing their performance and efficiency compared with each other and with the uncompressed models.

6	**Discussion, Conclusion and Recommendations** - **Discussion of the findings** - **Limitations** - **Conclusion** - **Future recommendations**	• Discuss the results and findings in light of the problem, objectives, data, and methodology. • Discuss the performance of the models, considering the chosen metrics. Important: although marks are not correlated with model performance, it is important that they are meaningful and a direct result of the student's efforts to create useful ML models for AI and Sustainability. Results must be compiled from the output of the Python code. These must be presented along with limitations, conclusions and recommendations in the written report suitable for a non-expert audience. The student must analyse and discuss them, confirming or rejecting any initial hypothesis, or indicating newly discovered findings, trends or patterns. This is an important part of your coursework – to be able to clearly communicate your results to a non-technical audience.	10%	**Interpretation:** • The depth and relevance of the insights drawn from the model's performance and how well the student connects these insights to sustainability implications. **Limitations, Conclusion and Recommendation:** • A reasonable presentation of current limitations of the work drawn from the results. • The clarity and completeness of the conclusion summarising the findings of the analysis. • The quality and feasibility of the recommendations provided on the analysis.

<table>
<tr><th colspan="2">Code 50%</th></tr>
<tr><th>Details</th><th>Marking Criteria</th></tr>
<tr><td>

Submitted as a single compressed (zip) file containing the Python code in Jupyter notebooks.
Each section must be run with associated output shown.

PLEASE NOTE:

If you choose to use and/or adapt code snippets from anywhere else (e.g., GitHub, Kaggle, Gen AI etc.), you must acknowledge this and provide the complete reference, to avoid any issues with plagiarism.
It is advised to split the code in sequential notebooks that are required to run the whole pipeline, so that once each step/notebook is performed/executed, it saves its outputs to be used as input to the next notebook in the pipeline. For instance, you could have the following sequence of notebooks: “1. Data loading and preprocessing.ipynb”, “2. Data exploratory analysis.ipynb”, “3. AI modelling.ipynb”, “4. Assessment and evaluation.ipynb”, “5. Model compression.ipynb” etc.

</td><td>

<u>Code Structure and Organisation:</u>
Modularity and Readability: how the code is well-organised into functions or modules, making it easy to read and understand.
Comments and Documentation: there are meaningful comments and documentation explaining the purpose and functionality of the code, including variable names and functions, to facilitate understanding and reading the code.
<u>Data Preparation:</u>
Data Cleaning: the data has been adequately cleaned, including handling missing values, outliers, and inconsistencies.
Data Transformation: necessary transformations have been applied to prepare the data for analysis (e.g., normalisation, encoding categorical variables).
<u>Exploratory Data Analysis (EDA):</u>
Insights and Visualisations: appropriate exploratory data analysis techniques was performed to gain insights into the data, and the results are effectively visualised.
Statistical Analysis: the code includes relevant statistical analysis to summarise and describe the data.
<u>Feature Engineering:</u>
Feature Augmentation: new features were derived from the original data to enhance predictive power or insights.
Feature Selection: if applicable, feature selection techniques were used to choose the most relevant features for the analysis.
<u>Modelling</u>:
Model Selection: appropriate models have been chosen based on the problem and data.
Model Training: models were trained using a suitable approach, and hyperparameters were tuned effectively.
Model Compression: models were compressed using suitable approaches, ensuring they got smaller, faster and yet performing similarly compared to their baseline, uncompressed models.
<u>Model Evaluation and Performance:</u>

</td></tr>
</table>

	Evaluation Metrics: appropriate evaluation metrics used to assess the models' performance and efficiency – including uncompressed and compressed models. Appropriate application of explainable AI techniques. **Comparisons:** multiple models were compared in terms of their performance, efficiency and sizes. • **<u>Data Visualisation and Interpretability:</u>** **Visualisations:** meaningful and informative visualisations created to present the analysis and results effectively and objectively. **Interpretability:** explanation of the models' results through an analysis interpretable to non-technical stakeholders.