IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Digital Object Identifier N/A

ICS-5110 Generative AI Journal

JEROME AGIUS[1], ISAAC MUSCAT[2], and KYLE DEMICOLI[3]
[1]jerome.agius.21@um.edu.mt
[2]isaac.muscat.21@um.edu.mt
[3]kyle.demicoli.21@um.edu.mt

I. INTRODUCTION

The integration of Generative AI tools has become increasingly prevalent in today's academic world thus, it is crucial that these AI tools are used to enhance one's work as opposed to serving as a substitute to the human element. Furthermore, it is crucial that generative AI usage is well documented within academic works for the benefit of future readers, as this can aid in exposing AI tool misuse and over reliance whilst shining a light on prominent tools and pipelines. This section aims to outline the AI tools used throughout the ICS5110 Applied Machine Learning project and the reasoning for their use.

Throughout the project, four AI tools were predominantly used consisting of ChatGPT [1], Perplexity [2], ChatPDF [3] and Jenni.AI [4]. The former two are widely known LLMs whereas the latter two are more niche, providing functionality not available within the free versions of ChatGPT and Perplexity.

A. CHATGPT

ChatGPT served as the primary tool throughout the entire project, prominently due to its large knowledge base, dependable responses, and shareable chat functionality. It was mostly used for four main tasks consisting of idea generation, paraphrasing, general guidance, and explanations. However, this tool proved quite limited in its applicability save for the aforementioned use cases as it lacked the ability to provide links to relevant academic works, opting to invent links and papers to provide a response; furthermore, it had no ability to read and comment on PDFs. These limitations were present due to using the free version of ChatGPT.

B. PERPLEXITY

Perplexity served as a secondary tool to ChatGPT due to its ability to reference and link to academic works. This proved crucial to circumvent the limitations of ChatGPT's free tier. Although Perplexity provided the same overall functionality as ChatGPT, we found that its responses were often repetitive, opting to present the same reworded information when prompted to elaborate or expand upon a certain topic. This was most likely a limitation present within the free tier.

C. CHATPDF

ChatPDF provides the capability to feed and query specific PDF documents. This was particularly useful for deducing the relevance of academic papers and pinpointing any relevant sections within. By enabling direct interaction with uploaded papers, it eliminated the need to read entire documents, allowing us to quickly extract relevant information and assess the pertinence of a source. This tool served to substitute ChatGPTs free tier limitations providing us with a manner to streamline the process of engaging with academic literature.

D. JENNI.AI

Jenni.AI is an AI tool designed to assist with writing tasks centred around user-defined titles. While its primary function is text generation, we utilised it for its secondary feature: generating citations relevant to specific written content. As a relatively new tool in the market, we employed Jenni.AI selectively to identify references for information that, while familiar to the team, required citations to enhance clarity and credibility for readers less familiar with the subject matter.

These AI tools were used in conjunction with one another to build upon the provided responses whilst mitigating the limitations present in each one. The various use cases, applications and prompts used are outlined in Sections III and IV below.

II. ETHICAL CONSIDERATIONS

The widespread use of generative AI tools within the academic field introduces several ethical issues which must be carefully addressed. These issue include but are not limited to data bias, data privacy, data integrity, misinformation and unintentional plagiarism.

A. DATA BIAS

Generative AI models are trained on extensive datasets sourced from the internet, which inherently carry a variety of biases. These biases can range from gender, racial, cultural, selection and more. However, in the case of academia these tend to have minimal impact as the models tend to suffer mostly from temporal and exposure bias within this context. Temporal bias refers to the fact that the majority of AI tools are trained on past data or have a hard limit such as ChatGPT being limited to data up to December 2023 [5]. As such, over-reliance on these models is prone to cause referencing to out of data research and may cause current research to be built upon out of date or disproved ideas harming the progress being made by the academic community. Furthermore, exposure bias presents a similar problem in that the AI tools promote the same research papers when prompted on a particular topic, this in turn can result in the stagnation of ideas as newer papers are based on similar research whilst overshadowing newer research on which the models have yet to be trained on. These biases are being phased out to a certain degree as models such as Perplexity are capable of searching the internet to enhance response and recommendations. However, this functionality is sure to have its own limitation resulting in similar biases and effects later down the line.

B. PRIVACY CONCERNS

The use of generative AI often involves processing user input, which may include sensitive or personal data. Despite knowing that these AI tools may retain your data for further training, they are so ingrained into many people's day-to-day life due to their ease of use that users often ignore this aspect of these models. Furthermore, once data has been trained upon by these models its impossible to remove this content from these models due to how training alters the models. This makes it quite difficult to comply with privacy regulations such as the GDPR [6] and the "right to be forgotten" due to these models being unable to unlearn the data they were exposed to. To ensure a degree of data privacy several actions can be taken [7]:

- **Data Privacy Vaults** - Data Privacy Vaults store sensitive data within a particular geographic location with tightly controlled access. An anonymised version of this data is also created to be used with third-parties such as LLMs without exposing any Personal Identifiable Information (PII) or sensitive information.
- **Zero Trust Model** - These encapsulate structures wherein only select individuals have access to sensitive data. These are commonly in place alongside Data Privacy Vaults.
- **Data Anonymisation and Tokenisation** - Sensitive data can be anonymised or tokenised before being used in LLMs. Tokenisation replaces sensitive information with non-sensitive equivalents, which helps to keep PII out of the model and reduces the risk of exposure during training and inference
- **Controlled Access to Private LLMs** - This involves running LLMs locally thereby data is kept locally and secure.

C. ACADEMIC INTEGRITY

Generative AI tools provide the capacity to enhance research efficiency and productivity through the generation of coherent and well-structured text, summarisation of large volumes of information, and the refinement of ideas. However, their use raises several concerns, relating to plagiarism and academic integrity. One major issue is the presentation of AI work as ones own which not only compromises academic integrity but also diminishes the individuals capacity for research and engagement with the source material, undermining the development of essential research skills such as critical analysis, synthesis, and the ability to evaluate sources effectively.

To address these concerns, academic institutions and research organisations are emphasising transparency with the usage of AI and encouraging its use as a tool aimed at minimising tediousness and streamlining the research process as opposed to serving as a full substitute to the research process. Emphasis is made on the ability for these models to make mistakes thereby encouraging research to thoroughly review and critically assess the generated content to ensure its accuracy and relevance. Most importantly, AI generated work ought to be paraphrased thus, making it ones own work as opposed to being copied verbatim to ensure that the material is understood and appropriately integrated into research [8, 9].

D. ADDRESSING MISINFORMATION

Generative AI models may sometimes produce "hallucinated" outputs, statements that seem factual but lack a basis in reliable data. This issue arises from the probabilistic nature of these models, which predict text based on patterns in training data. Such misinformation is especially problematic in academia, where accuracy and source reliability are essential.

Errors in AI-generated content can lead to flawed conclusions or the spread of false knowledge, particularly when users fail to critically assess its validity. To mitigate this, researchers should cross-check AI outputs against credible references. Developers can also improve AI systems by integrating fact-checking algorithms and transparency features, ensuring AI is used responsibly in academic contexts.

III. METHODOLOGY

This section will outline the various ways in which the AI tools mentioned in Section I consisting of ChatGPT, Perplexity, ChatPDF and Jenni.AI were used throughout the project to enhance development.

These tools were used both independently and together forming a pipeline. Independent use consisted mostly of prompting ChatGPT with regards to resolving coding errors, model implementation best practices, ideas on how to structure the project document and more. Perplexity was often used as a backup when ChatGPT responses were unsatisfactory or when responses required supporting citations. Furthermore, perplexity often served as the primary LLM when we were initially looking for relevant research due to its ability to provide links and academic related responses. Along the same vein Jenni.AI was prominently used when we wrote paragraphs that contained information which required citations due to it not necessarily being common knowledge for the general reader. This was also used to expand upon text by finding citations related to said text which in turn would lead us to papers which we could use to expand our project further. Finally, ChatPDF was used throughout to minimise the time spent reading papers as it provided the ability to query the papers directly whilst also highlighting paper sections from which the responses were derived facilitating easy navigation of the paper for further clarification if the models responses were not satisfactory.

In addition to their independent usage, these tools were integrated together to form iterative pipelines wherein one tools output would be fed to another for refinement prior to being re-fed to the original tool in hope of a more refined response, this was mostly done between ChatGPT and Perplexity wherein ChatGPT would often be used for its ability to better articulate and simplify complex topics whilst Perplexity for its ability to ensure consistency with academic works. Furthermore the papers referenced in these responses would be downloaded and skimmed through for relevance however when this was not possible to do on a short time frame these papers would be fed to ChatPDF to derive a quick summary and determine the papers relevance to the project. The general pipeline steps are outlined in Figure 1. Furthermore, Section IV denotes a variety of prompts used throughout the development and documentation process.

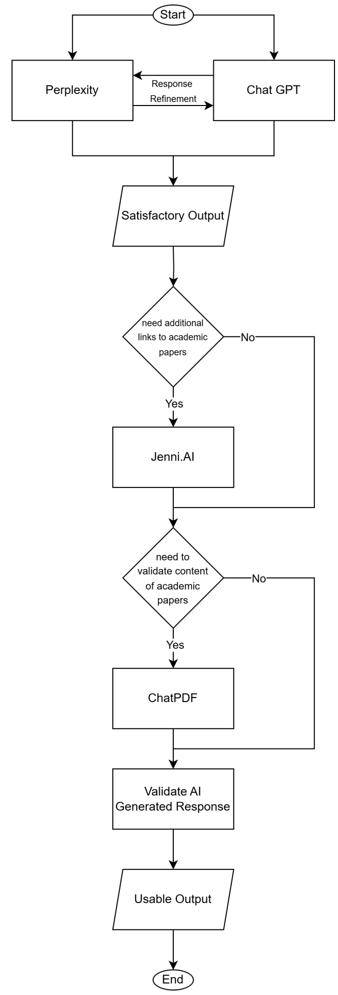


FIGURE 1. AI Tool Pipeline

IV. PROMPTS AND RESPONSES

Throughout the development of this project, ChatGPT was utilised for many tasks, including idea generation, decision-making support as well as the clarification of information. Whilst all three members of this group utilised these tools separately, some prompts and outputs retrieved overlapped and were similar. Some examples of prompts made as a group can be seen in the following figures. In Figure 2, ChatGPT was prompted to start generating ideas for which Machine Learning techniques to implement. Given that it was already beforehand decided as a group that a classification task would be implemented, this was specified to the tool. As can be seen from the results in Figures 3, 4, 5 and 6, the SVM, Logistic Regression and Random Trees techniques were all mentioned in the output.

Give me a list of machine learning techniques, specifically for classification tasks.

2/2

FIGURE 2. ChatGPT Machine Learning Techniques Prompt

Here's a comprehensive list of machine learning techniques commonly used for classification tasks:

Traditional Algorithms

1. **Logistic Regression**
   - Binary and multiclass classification with linear decision boundaries.
2. **Linear Discriminant Analysis (LDA)**
   - Assumes normally distributed features, finds linear combinations for class separation.
3. **k-Nearest Neighbors (k-NN)**
   - Instance-based algorithm that classifies based on the majority vote of neighbors.
4. **Decision Trees**
   - Rule-based method that splits data into classes via hierarchical decisions.
5. **Naive Bayes**
   - Probabilistic classifier assuming feature independence (e.g., Gaussian, Multinomial).
6. **Support Vector Machines (SVM)**
   - Maximizes margin between classes, works with linear and non-linear kernels.

FIGURE 3. ChatGPT Machine Learning Techniques Output (1 / 4)

Ensemble Methods

1. **Random Forests**
   - Collection of decision trees with randomized features to reduce overfitting.
2. **Gradient Boosting Machines (GBMs)**
   - Builds models sequentially to correct errors of prior models (e.g., XGBoost, LightGBM, CatBoost).
3. **Bagging (Bootstrap Aggregating)**
   - Aggregates predictions from multiple models trained on bootstrapped datasets (e.g., Bagged Trees).
4. **AdaBoost (Adaptive Boosting)**
   - Focuses on misclassified instances, adjusting weights iteratively.
5. **Stacking (Stacked Generalization)**
   - Combines predictions of multiple models using a meta-model.

FIGURE 4. ChatGPT Machine Learning Techniques Output (2 / 4)

Neural Networks

1. **Multilayer Perceptron (MLP)**
   - Fully connected feedforward neural network for general classification.
2. **Convolutional Neural Networks (CNNs)**
   - Specialized in image and spatial data classification.
3. **Recurrent Neural Networks (RNNs)**
   - Handles sequential data like text and time series (e.g., LSTM, GRU).
4. **Transformers**
   - Powerful for text classification tasks (e.g., BERT, GPT-based models).

Probabilistic Models

1. **Bayesian Networks**
   - Represents probabilistic relationships between features and classes.
2. **Hidden Markov Models (HMMs)**
   - Commonly used for sequence classification tasks.

FIGURE 5. ChatGPT Machine Learning Techniques Output (3 / 4)

Unsupervised or Semi-supervised Techniques Adapted for Classification

1. **Clustering-Based Methods**
   - K-means or hierarchical clustering for indirect classification.
2. **Self-Training**
   - Uses labeled data to label the unlabeled dataset iteratively.
3. **Generative Adversarial Networks (GANs)**
   - Fine-tuned for semi-supervised classification.

Hybrid and Specialized Techniques

1. **Hyperparameter Optimization Techniques**
   - Grid Search, Random Search, Bayesian Optimization for classifier tuning.
2. **Evolutionary Algorithms**
   - Genetic programming for evolving classification models.
3. **Rule-Based Classification**
   - Creates interpretable rules for decision-making (e.g., RIPPER, PART).
4. **Metric Learning**
   - Learns optimal distance metrics for class separation (e.g., Siamese Networks).
5. **Fuzzy Logic-Based Classifiers**
   - Uses fuzzy sets for dealing with uncertainty in classification.

FIGURE 6. ChatGPT Machine Learning Techniques Output (4 / 4)

Furthermore, to search for information backed by research papers, Perplexity was employed. Once the three machine learning techniques were selected thanks to ChatGPT's idea generation, Perplexity was used to better understand how each of these techniques works. This can be seen in Figures 7, 8, 9, 10.

Explain to me what Logistic Regression is.

FIGURE 7. Perplexity Logistic Regression Prompt

Perplexity

Logistic regression is a widely used statistical method for binary classification problems, where the goal is to predict the probability of a binary outcome based on one or more independent variables. This technique is particularly valuable in fields such as marketing, finance, and healthcare, where decisions often hinge on yes/no or true/false outcomes.

Definition and Purpose

Logistic regression is a **supervised learning algorithm** that estimates the relationship between a dependent binary variable (the outcome) and one or more independent variables (predictors). The dependent variable can take on two possible values, commonly represented as 0 and 1, where 1 indicates the occurrence of an event (e.g., success) and 0 indicates non-occurrence (e.g., failure) [1] [2] [4].

The primary purpose of logistic regression is to model the probability that a given input point belongs to a particular category. For instance, it can be used to predict whether a customer will purchase a product based on their demographic data or whether a patient will respond to a specific treatment based on their medical history.

FIGURE 8. Perplexity Logistic Regression Output (1/3)

Mathematical Foundation

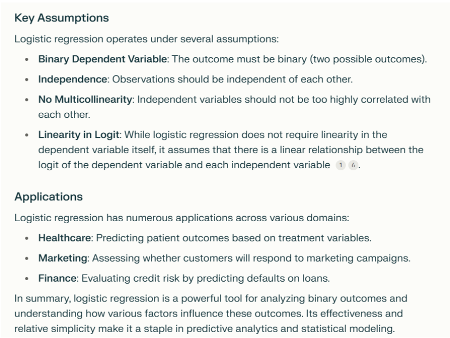
At the heart of logistic regression lies the **logistic function**, also known as the **sigmoid function**, which maps any real-valued number into the range between 0 and 1. This function is defined mathematically as:

$$P(Y=1|X) = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\ldots+\beta_k X_k)}}$$

Here, $P(Y=1|X)$ represents the probability that the dependent variable Y equals 1 given the independent variables X, while β_0 is the intercept and $\beta_1, \ldots, \beta_k$ are coefficients corresponding to each independent variable [3] [4].

The model uses **maximum likelihood estimation** to find the best-fitting parameters (β) that maximize the likelihood of observing the given data [2] [5]. Unlike linear regression, logistic regression does not assume that there is a linear relationship between the independent and dependent variables; instead, it models the log-odds of the probability of success.

FIGURE 9. Perplexity Logistic Regression Output (2/3)


Key Assumptions

Logistic regression operates under several assumptions:

- **Binary Dependent Variable**: The outcome must be binary (two possible outcomes).
- **Independence**: Observations should be independent of each other.
- **No Multicollinearity**: Independent variables should not be too highly correlated with each other.
- **Linearity in Logit**: While logistic regression does not require linearity in the dependent variable itself, it assumes that there is a linear relationship between the logit of the dependent variable and each independent variable 1 6.

Applications

Logistic regression has numerous applications across various domains:

- **Healthcare**: Predicting patient outcomes based on treatment variables.
- **Marketing**: Assessing whether customers will respond to marketing campaigns.
- **Finance**: Evaluating credit risk by predicting defaults on loans.

In summary, logistic regression is a powerful tool for analyzing binary outcomes and understanding how various factors influence these outcomes. Its effectiveness and relative simplicity make it a staple in predictive analytics and statistical modeling.

FIGURE 10. Perplexity Logistic Regression Output (3/3)

Furthermore, to save time reading through research papers which may be irrelevant, ChatPDF was utilised to upload the research paper and query the LLM to deduce the relevance of the paper and extract the important parts of the text. An exemplary prompt and output can be seen in Figures 11 and 12.

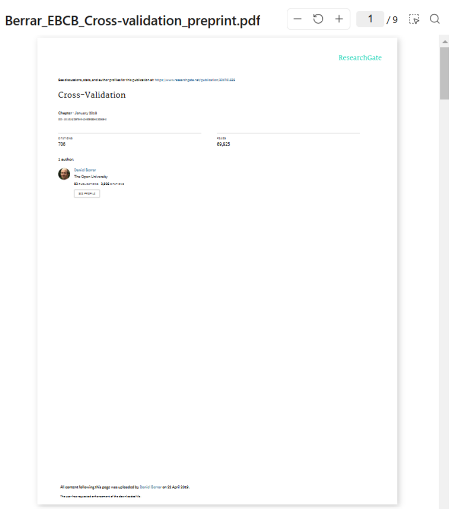
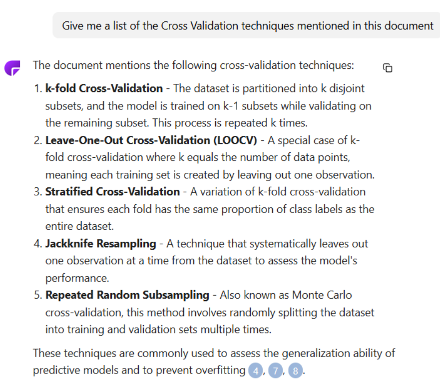


FIGURE 11. ChatPDF File Inputted


Give me a list of the Cross Validation techniques mentioned in this document

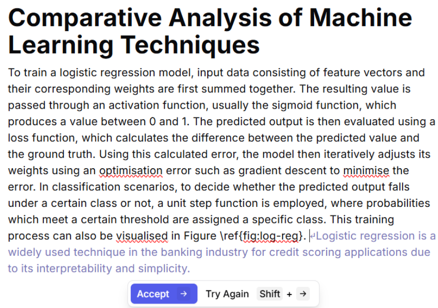
The document mentions the following cross-validation techniques:

1. **k-fold Cross-Validation** - The dataset is partitioned into k disjoint subsets, and the model is trained on k-1 subsets while validating on the remaining subset. This process is repeated k times.
2. **Leave-One-Out Cross-Validation (LOOCV)** - A special case of k-fold cross-validation where k equals the number of data points, meaning each training set is created by leaving out one observation.
3. **Stratified Cross-Validation** - A variation of k-fold cross-validation that ensures each fold has the same proportion of class labels as the entire dataset.
4. **Jackknife Resampling** - A technique that systematically leaves out one observation at a time from the dataset to assess the model's performance.
5. **Repeated Random Subsampling** - Also known as Monte Carlo cross-validation, this method involves randomly splitting the dataset into training and validation sets multiple times.

These techniques are commonly used to assess the generalization ability of predictive models and to prevent overfitting 4, 7, 8.

FIGURE 12. ChatPDF Prompt / Output

Finally, Jenni.ai was used to assist with retrieving citations to back claims made within the paper. These statements, whilst common knowledge for people with experience within this field, require research papers to support them. An exemplary paragraph and outputted citation can be seen in Figures 13 and 14.


Comparative Analysis of Machine Learning Techniques

To train a logistic regression model, input data consisting of feature vectors and their corresponding weights are first summed together. The resulting value is passed through an activation function, usually the sigmoid function, which produces a value between 0 and 1. The predicted output is then evaluated using a loss function, which calculates the difference between the predicted value and the ground truth. Using this calculated error, the model then iteratively adjusts its weights using an optimisation error such as gradient descent to minimise the error. In classification scenarios, to decide whether the predicted output falls under a certain class or not, a unit step function is employed, where probabilities which meet a certain threshold are assigned a specific class. This training process can also be visualised in Figure \ref{fig:log-reg}. Logistic regression is a widely used technique in the banking industry for credit scoring applications due to its interpretability and simplicity.

Accept Try Again Shift +

FIGURE 13. Jenni.ai Input

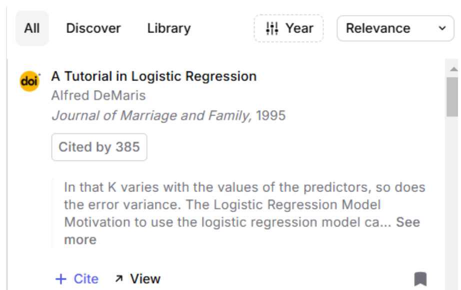


FIGURE 14. Jenni.ai Citation Output

V. IMPROVEMENTS, ERRORS AND CONTRIBUTIONS

Throughout the development of this project, Generative AI was used for multiple different tasks, be it research, code development and even documentation writing.

During research, Generative AI proved invaluable in explaining information, particularly when understanding machine learning techniques, their mechanisms, and applications. This facilitated comprehension of the basic concepts of the techniques selected, highlighted their strengths, limitations as well their most common applications. Apart from the techniques, GenAI was also used to better understand how to explore the dataset selected. Despite most outputs being valid and helping us better understand the content we were working on, there were cases where the model seemed to misinterpret our prompts. For instance, in some cases, despite specifying the type of analysis or outcome we were aiming for, the responses generated by the model were overly generic. This highlighted the importance of creating precise and unambiguous prompts when interacting with Generative AI tools. Misinterpretations often led to confusion, requiring additional research or rephrasing of the queries to obtain relevant insights.

Following this, Generative AI was used in code development, particularly for two tasks, code generation and bug fixing. For the former, GenAI was prompted asking what packages to use for the machine learning techniques, in which sklearn was recommended. Further to this, the hyper-parameter grid used during hyper-parameter optimisation was suggested by GenAI, in which we filled in with a variety of possible values. Apart from code generation, GenAI was also prompted to help with bug fixing. In a few instances, ChatGPT was prompted to solve a syntax issue, which normally is due to a variable name being spelt incorrectly. This helps locate the error quicker than manually going through every line to search for the mistake. Unfortunately, not ever bug was solved through AI, particularly ones with package issues. When setting up a virtual environment for the project, some code chunks could not be run due to mismatching package versions. Despite all the information being provided to the model, it failed to provide the correct way to go about solving this issue, instead we had to search through the web ourselves to find a solution.

During documentation writing, GenAI was prompted particularly to review the writing's clarity and conciseness. The changes suggested would improve the general structure and flow of the content to be more understandable and straight to the point. Furthermore, as was mentioned in Section I, Perplexity and Jenni.ai were utilised to retrieve citations for claims made throughout the documentation. This made the tedious task of sifting through research papers to identify relevant information to mention in the documentation much easier and less time-consuming. However, extra care had to be made when using information generated by some models, particularly ChatGPT, which was quite prone to hallucinating and coming up with statements not backed by any literature.

VI. INDIVIDUAL REFLECTION

A. JEROME AGIUS

Using generative AI throughout the project proved beneficial, significantly streamlining my workflow and deepening my understanding of technical concepts. I utilised four generative AI tools, each serving distinct roles as discussed throughout this paper. While my perspective on AI in academic work has remained unchanged, as I have always advocated for its role in enhancing education, my experience reinforced the importance of its incorporation within education to reduce the tedious aspects of research, implementation and education in favour of focusing on the more complex or engaging aspects of a topic. However, prior to this project I had never had the freedom to fully utilise AI tools for academic purposes and it was surprising how efficient the entire process is and how easily previously time-consuming tasks can be carried out. However, this increased efficiency hinged on the proper application of the AI tools as using ChatGPT for paper summarisation often resulted in more time being wasted trying to get a usable output as opposed to when ChatPDF was used for the same purpose. Throughout the project I found these generative tools to be most useful throughout the coding process, offering constructive feedback on how code can be refined and optimised. Additionally, it played a crucial role in the writing process, providing actionable suggestions for extending content and identifying areas that required additional detail or clarity.

Excluding generic conversations regarding idea generation or code / text improvements some noteworthy use cases of these tools include the use of ChatGPT to aid in the setup for GitHub pages as denoted in Figures 15 - 17. Additionally, ChatPDF was also used to carry out one final check after careful review from my end to ensure that all aspects of the documentation were in line with the brief as denoted in Figure 18 which depicts a sample of the conversation due to its length.

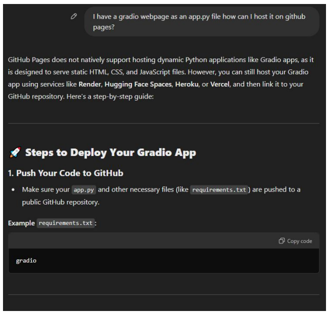


FIGURE 15. GitHub Pages Setup Conversation (1/3)

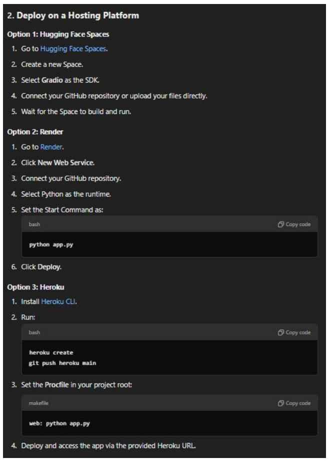


FIGURE 16. GitHub Pages Setup Conversation (2/3)

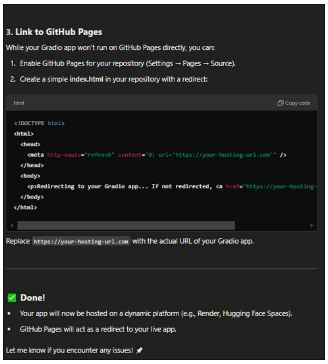


FIGURE 17. GitHub Pages Setup Conversation (3/3)

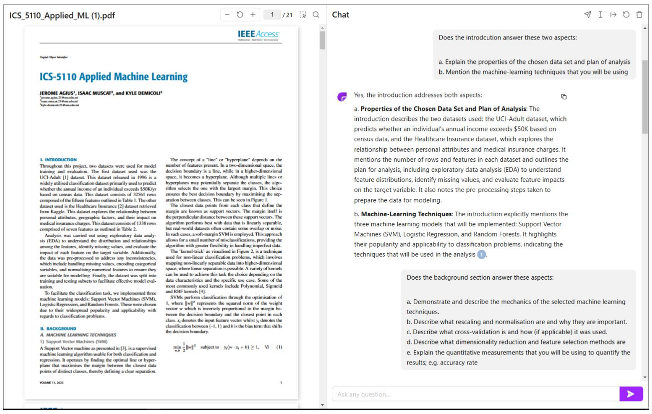


FIGURE 18. ChatPDF Document Proof Read Conversation (Sample)

B. ISAAC MUSCAT

Throughout this research, I experimented with different types of Generative AI tools. Prior to this project, the only model I was familiar with was ChatGPT, however through research and discussions with my colleagues, I discovered new tools to utilise, these being Perplexity, ChatPDF and Jenni.ai. Apart from these different models, I also started experimenting more with prompt styles, by specifying the length of prompts to avoid having to read through a lengthy answer, specifying the style of language to use (normally to utilise simple words for better clarity) and also specifying the format of the output, whether it is a paragraph, a list or even a table. Such experimentation often leads to a more accurate and tailored response from the model.

In the case of data exploration, ChatGPT was prompted to better understand what types of exploration techniques could be used given the format of the dataset and its features. Furthermore, when developing Logistic Regression, the model was prompted to better understand what is the difference between base Logistic Regression and Multinomial Regression. Finally, when implementing cross validation, ChatGPT was prompted to decide which technique to implement. The decision to utilise SKFCV was then supported by papers found which are mentioned in the Background section. The other LLMs were then utilised for research purposes when writing the documentation for the project, particularly for text conciseness and clarity. Examples of prompts and outputs retrieved from ChatGPT can be seen in Figures 19 - 23.

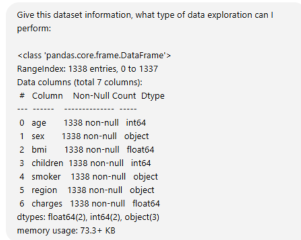


FIGURE 19. Dataset Exploration Prompt

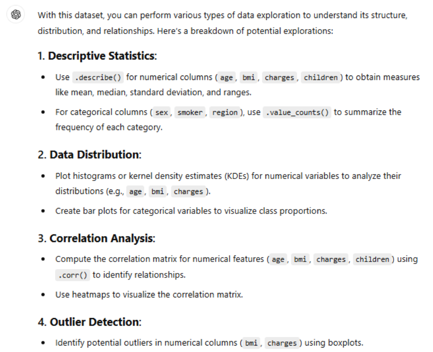


FIGURE 20. Dataset Exploration Output (1 / 2)

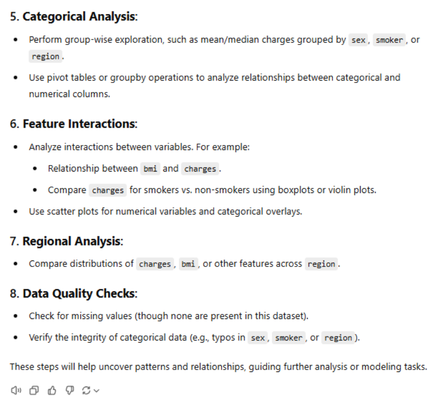
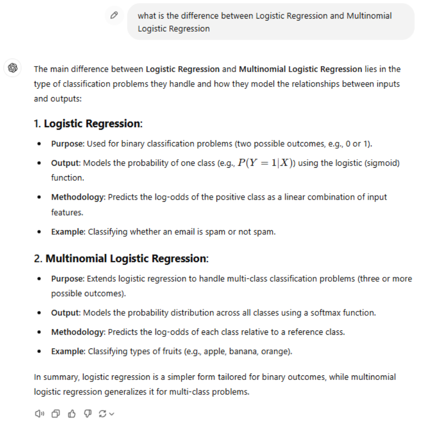


FIGURE 21. Dataset Exploration Output (2 / 2)



what is the difference between Logistic Regression and Multinomial Logistic Regression

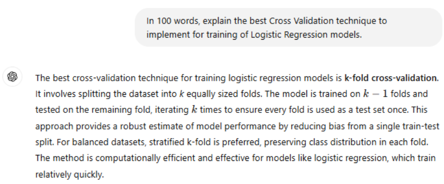
The main difference between **Logistic Regression** and **Multinomial Logistic Regression** lies in the type of classification problems they handle and how they model the relationships between inputs and outputs:

1. **Logistic Regression**:
   - **Purpose**: Used for binary classification problems (two possible outcomes, e.g., 0 or 1).
   - **Output**: Models the probability of one class (e.g., $P(Y = 1|X)$) using the logistic (sigmoid) function.
   - **Methodology**: Predicts the log-odds of the positive class as a linear combination of input features.
   - **Example**: Classifying whether an email is spam or not spam.

2. **Multinomial Logistic Regression**:
   - **Purpose**: Extends logistic regression to handle multi-class classification problems (three or more possible outcomes).
   - **Output**: Models the probability distribution across all classes using a softmax function.
   - **Methodology**: Predicts the log-odds of each class relative to a reference class.
   - **Example**: Classifying types of fruits (e.g., apple, banana, orange).

In summary, logistic regression is a simpler form tailored for binary outcomes, while multinomial logistic regression generalizes it for multi-class problems.

FIGURE 22. Logistic Regression Prompt + Output



In 100 words, explain the best Cross Validation technique to implement for training of Logistic Regression models.

The best cross-validation technique for training logistic regression models is **k-fold cross-validation**. It involves splitting the dataset into k equally sized folds. The model is trained on $k-1$ folds and tested on the remaining fold, iterating k times to ensure every fold is used as a test set once. This approach provides a robust estimate of model performance by reducing bias from a single train-test split. For balanced datasets, stratified k-fold is preferred, preserving class distribution in each fold. The method is computationally efficient and effective for models like logistic regression, which train relatively quickly.

FIGURE 23. Cross Validation Prompt + Output

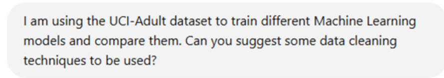
C. KYLE DEMICOLI

Our project's use of generative AI tools improved our workflow and enhanced my understanding of challenging concepts. I worked with several generative AI systems during this project, with ChatGPT serving as the primary tool. Many elements of our research and execution phases were greatly aided by this technology. I have always had a positive opinion of AI's place in academics and firmly believe that it has the ability to improve educational procedures.

During this project, I had the opportunity to thoroughly examine the usefulness of AI tools in real-world scenarios. I was very pleased with how these technologies made labour-intensive jobs more efficient. Using AI-driven data analysis tools, for instance, sped up our data processing activities and gave us more time for strategic model development planning.

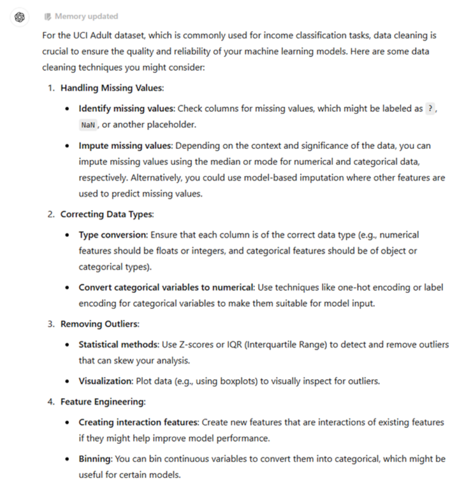
However, there was a learning curve associated with the use of AI tools. Obviously, certain responses needed modifications and specific inputs to provide beneficial outcomes, which highlighted how crucial it is to understand the tools' strengths and limitations. This was made clear when we employed AI platforms for research, as some of them offered more useful feedback than others.

During this process, the incorporation of AI, particularly ChatGPT, was also crucial for our project's documentation and reporting stages. It provided customised recommendations that enhanced our material and made difficult subjects easier to understand, guaranteeing thorough and clear deliverables. Figures 24 - 32 below shows examples of prompts and outputs generated through ChatGPT.



I am using the UCI-Adult dataset to train different Machine Learning models and compare them. Can you suggest some data cleaning techniques to be used?

FIGURE 24. Data Cleaning Techniques Prompt

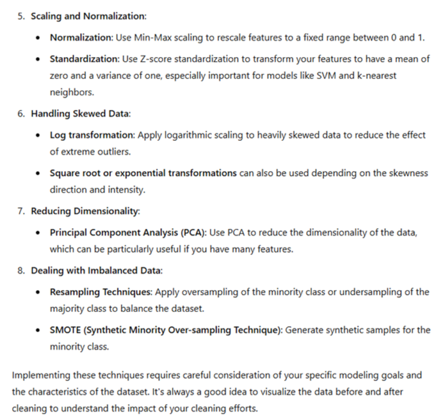


Memory updated

For the UCI Adult dataset, which is commonly used for income classification tasks, data cleaning is crucial to ensure the quality and reliability of your machine learning models. Here are some data cleaning techniques you might consider:

1. **Handling Missing Values**:
   - **Identify missing values**: Check columns for missing values, which might be labeled as `?`, `NaN`, or another placeholder.
   - **Impute missing values**: Depending on the context and significance of the data, you can impute missing values using the median or mode for numerical and categorical data, respectively. Alternatively, you could use model-based imputation where other features are used to predict missing values.
2. **Correcting Data Types**:
   - **Type conversion**: Ensure that each column is of the correct data type (e.g., numerical features should be floats or integers, and categorical features should be of object or categorical types).
   - **Convert categorical variables to numerical**: Use techniques like one-hot encoding or label encoding for categorical variables to make them suitable for model input.
3. **Removing Outliers**:
   - **Statistical methods**: Use Z-scores or IQR (Interquartile Range) to detect and remove outliers that can skew your analysis.
   - **Visualization**: Plot data (e.g., using boxplots) to visually inspect for outliers.
4. **Feature Engineering**:
   - **Creating interaction features**: Create new features that are interactions of existing features if they might help improve model performance.
   - **Binning**: You can bin continuous variables to convert them into categorical, which might be useful for certain models.

FIGURE 25. Data Cleaning Techniques Output (1/2)



5. **Scaling and Normalization**:
   - **Normalization**: Use Min-Max scaling to rescale features to a fixed range between 0 and 1.
   - **Standardization**: Use Z-score standardization to transform your features to have a mean of zero and a variance of one, especially important for models like SVM and k-nearest neighbors.
6. **Handling Skewed Data**:
   - **Log transformation**: Apply logarithmic scaling to heavily skewed data to reduce the effect of extreme outliers.
   - **Square root or exponential transformations** can also be used depending on the skewness direction and intensity.
7. **Reducing Dimensionality**:
   - **Principal Component Analysis (PCA)**: Use PCA to reduce the dimensionality of the data, which can be particularly useful if you have many features.
8. **Dealing with Imbalanced Data**:
   - **Resampling Techniques**: Apply oversampling of the minority class or undersampling of the majority class to balance the dataset.
   - **SMOTE (Synthetic Minority Over-sampling Technique)**: Generate synthetic samples for the minority class.

Implementing these techniques requires careful consideration of your specific modeling goals and the characteristics of the dataset. It's always a good idea to visualize the data before and after cleaning to understand the impact of your cleaning efforts.

FIGURE 26. Data Cleaning Techniques Output (2/2)

Certainly! Random Forest is a versatile and powerful machine learning algorithm that belongs to the ensemble learning family. It builds upon the concept of decision trees, combining multiple trees to improve accuracy and control over-fitting, which is a common problem in single decision trees.

Key Concepts:

1. **Ensemble Method:** Random Forest uses the ensemble method, which involves combining multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. In the case of Random Forest, it aggregates the results of multiple decision trees.
2. **Decision Trees:** At its core, Random Forest uses multiple decision trees to make predictions. Each tree in the forest is built from a random sample of the data, and at each node, a subset of features is randomly chosen to decide the best split.
3. **Bagging (Bootstrap Aggregating):** Random Forest applies the general technique of bagging, where multiple samples from the dataset are taken with replacement (bootstrap samples), and a model (in this case, a decision tree) is built on each sample. The final output decision is made based on the majority voting (for classification) or average (for regression) of all trees.
4. **Random Feature Selection:** When building each tree, Random Forest introduces additional randomness. Instead of searching for the most significant feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity among the trees, which generally produces a stronger overall model.

Advantages:

- **Robustness:** Random Forest is less prone to overfitting than a single decision tree because it averages multiple trees to curb excessive variance.
- **Handling Large Data Sets:** It can handle large datasets with higher dimensionality and can evaluate thousands of input variables without variable deletion.
- **Versatility:** It is effective for both regressions and classification tasks and does well on both linear and non-linear problems.
- **Feature Importance:** It provides a good indicator of the importance of features, which can be useful in feature selection.

FIGURE 27. Random Forest Research Prompt

Certainly! Random Forest is a versatile and powerful machine learning algorithm that belongs to the ensemble learning family. It builds upon the concept of decision trees, combining multiple trees to improve accuracy and control over-fitting, which is a common problem in single decision trees.

Key Concepts:

1. **Ensemble Method:** Random Forest uses the ensemble method, which involves combining multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. In the case of Random Forest, it aggregates the results of multiple decision trees.
2. **Decision Trees:** At its core, Random Forest uses multiple decision trees to make predictions. Each tree in the forest is built from a random sample of the data, and at each node, a subset of features is randomly chosen to decide the best split.
3. **Bagging (Bootstrap Aggregating):** Random Forest applies the general technique of bagging, where multiple samples from the dataset are taken with replacement (bootstrap samples), and a model (in this case, a decision tree) is built on each sample. The final output decision is made based on the majority voting (for classification) or average (for regression) of all trees.
4. **Random Feature Selection:** When building each tree, Random Forest introduces additional randomness. Instead of searching for the most significant feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity among the trees, which generally produces a stronger overall model.

Advantages:

- **Robustness:** Random Forest is less prone to overfitting than a single decision tree because it averages multiple trees to curb excessive variance.
- **Handling Large Data Sets:** It can handle large datasets with higher dimensionality and can evaluate thousands of input variables without variable deletion.
- **Versatility:** It is effective for both regressions and classification tasks and does well on both linear and non-linear problems.
- **Feature Importance:** It provides a good indicator of the importance of features, which can be useful in feature selection.

FIGURE 28. Random Forest Research Output (1/2)

Disadvantages:

- **Complexity:** More complex than a single decision tree, it requires more computational resources and is slower to train.
- **Interpretability:** Unlike a single decision tree, which can be easily visualized and interpreted, the ensemble nature of Random Forest makes it more like a "black box".

Applications:

Random Forest has been successfully applied in various fields including but not limited to:

- **Banking** (for credit scoring and fraud detection),
- **Medicine** (for predicting diseases and conditions),
- **Biology** (for classifying species),
- **E-commerce** (for recommendation systems), and
- **Stock Market Analysis.**

Due to its accuracy and ease of use, Random Forest is a favorite tool among data scientists and is widely used in both academic and industrial settings.

FIGURE 29. Random Forest Research Output (2/2)

What are some evaluation metrics or visualization tools that can be used in order to observe how a machine learning model is performing for a classification task.

FIGURE 30. Evaluation Metrics Research Prompt

Evaluating the performance of a machine learning model for a classification task is crucial to understand its effectiveness and areas for improvement. Here are several commonly used evaluation metrics and visualization tools that can help you assess your model's performance:

Evaluation Metrics

1. **Accuracy:**
   - Measures the ratio of correctly predicted observations to the total observations. It's simple and intuitive but can be misleading if the data is imbalanced.
2. **Precision and Recall:**
   - **Precision** (Positive Predictive Value): The ratio of correctly predicted positive observations to the total predicted positives. It's crucial when the cost of a false positive is high.
   - **Recall** (Sensitivity or True Positive Rate): The ratio of correctly predicted positive observations to all actual positives. It's essential when the cost of a false negative is high.
3. **F1 Score:**
   - The weighted average of Precision and Recall. This score takes both false positives and false negatives into account. It is particularly useful if you seek a balance between Precision and Recall and there is an uneven class distribution.
4. **Confusion Matrix:**
   - A table used to describe the performance of a classification model on a set of test data for which the true values are known. It provides insights into the types of errors made by the classifier.
5. **ROC Curve (Receiver Operating Characteristic curve):**
   - A graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. It is plotted with True Positive Rate (Recall) against the False Positive Rate.
6. **AUC (Area Under the Curve):**
   - AUC - ROC curve is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The higher the AUC, the better the model is at predicting 0s as 0s and 1s as 1s.

FIGURE 31. Evaluation Metrics Research Output (1/2)

Visualization Tools

1. **Matplotlib and Seaborn:**
   - Python libraries that offer a wide range of capabilities for creating static, animated, and interactive visualizations in Python. Useful for plotting ROC curves, precision-recall curves, and confusion matrices.
2. **Plotly and Dash:**
   - These libraries provide more interactive plots and can be particularly useful for creating dynamic visualizations that allow users to explore different thresholds and effects on model metrics.
3. **Confusion Matrix Visualization:**
   - Tools like Scikit-learn offer utilities to easily plot confusion matrices that can be further enhanced using Matplotlib or Seaborn for a more detailed visual representation.
4. **Yellowbrick:**
   - A machine learning visualization library based on Matplotlib that provides methods for visually representing different model performance metrics such as ROC curves, confusion matrices, and classification reports.
5. **TensorBoard:**
   - Originally designed for TensorFlow, it helps in tracking metrics like loss and accuracy during the training process, visualizing the model graph, viewing histograms of weights, biases, or other tensors as they change over time.

Incorporating these metrics and tools into your machine learning workflow will provide you with a comprehensive view of your model's performance, allowing you to make informed decisions about how to improve and adjust the model.

FIGURE 32. Evaluation Metrics Research Output (2/2)

REFERENCES

[1] OpenAI, *Chatgpt: Optimizing language models for dialogue*, Accessed: 2024-12-10, 2023. [Online]. Available: https://openai.com/chatgpt.

[2] P. AI, *Perplexity ai: Search engine*, Accessed: 2024-12-10, 2024. [Online]. Available: https://www.perplexity.ai/.

[3] ChatPDF, *Chatpdf: Ai for pdf querying*, Accessed: 2024-12-10, 2024. [Online]. Available: https://www.chatpdf.com/.

[4] Jenni.AI, *Jenni.ai: Ai writing assistant*, Accessed: 2024-12-10, 2024. [Online]. Available: https://app.jenni.ai/.

[5] OpenAI, *Models overview*, Accessed: 2024-12-10, 2024. [Online]. Available: https://platform.openai.com/docs/models.

[6] GDPR-info.eu, *General data protection regulation (gdpr) information*, Accessed: 2024-12-10, 2024. [Online]. Available: https://gdpr-info.eu/.

[7] S. Overflow, *Privacy in the age of generative ai*, Accessed: 2024-12-10, 2023. [Online]. Available: https://stackoverflow.blog/2023/10/23/privacy-in-the-age-of-generative-ai/.

[8] K. Nat, *Ethical considerations in artificial intelligence*, Accessed: 2024-12-10, 2024. [Online]. Available: https://blog.mdpi.com/2024/02/01/ethical-considerations-artificial-intelligence/.

[9] E. Partnerships, *Responsible and ethical use of ai in higher education*, Accessed: 2024-12-10, 2024. [Online]. Available: https://eimpartnerships.com/articles/responsible-and-ethical-use-of-ai-in-higher-education.