Problem 1: Fraud Prediction

Problem Statement: The rise of digital payments has made transactions faster and more convenient, but it has also introduced significant risks of fraudulent activity. Credit card fraud not only results in financial losses for businesses and consumers but also undermines trust in payment systems. To combat this challenge, robust fraud detection systems are essential. As a Data Scientist at Sertis, you are tasked with developing a machine learning model to identify potentially fraudulent credit card transactions using a dataset of anonymized transactions made by European cardholders in 2023. Additionally, the client has requested for interpretability especially in cases where a transaction has been declared as fraud since one must know why a transaction has been declared as fraud. The dataset (here) contains over **550,000** records with features representing transaction characteristics while ensuring the privacy of cardholders. The dataset can be downloaded from here. Please refer to the **README** file, attached within, for detailed information on the dataset. For evaluating your model, consider the approach of splitting the given dataset into training and a test sets. You can use the test set to evaluate the performance of your model.

Evaluation Criteria

For the assignment that you choose, you will be tested on:

*1. **EDA and Conclusions**: Conduct an in-depth EDA on the chosen dataset and draw actionable insights. Reflect these insights in your feature engineering and modeling choices.*

*2. **Development**: Develop an appropriate algorithm/model using the chosen dataset. Justify your choices for your model/algorithm of choice, data splitting, and hyperparameter tuning. **Discuss the trade-offs involved**.*

*3. **Evaluation**: Clearly **explain** your evaluation criteria and the metrics of choice to gauge your algorithm/model's performance using these metrics. Discuss any issues encountered, such as overfitting or underfitting, lack of data, and how you mitigated them.*

*4. **Documentation**: Maintain thorough code documentation, including docstrings and inline comments. Good documentation includes not just what the code does, but **why** it does it.*

*5. **Reasoning**: As you work through your chosen problem, articulate your thought process. Explain the rationale behind your actions, your interpretations of the results, and your future steps.*

Problem 2: Chatbot with Retrieval Augmented Generation (RAG)

Problem Statement: Retrieval Augmented Generation (RAG) is an important module allowing a Large-Language Model (LLM) to process and extract knowledge from large amounts of external documents without finetuning it. You are asked to develop an LLM-based chatbot with your own implemented RAG module that creates a knowledge base from documents in a given dataset. The solution should demonstrate your understanding of document chunking, chunk embedding, building a vector database, and vector search. The given dataset also contains sets of questions used to test the RAG module.

The dataset consists of 4 CSV files:

1. *documents.csv* contains 20 documents used to build the RAG.
2. *single_passage_answer_questions.csv* contains a set of question-answer pairs in which a single document is required to generate the answer
3. *multi_passage_answer_questions.csv* contains a set of question-answer pairs in which several documents are required to generate the answer
4. *no_answer_questions.csv* contains a set of questions in which the answer is not included in the documents.

A bonus question: To ensure the safe, ethical, and reliable use of the chatbot, you are asked to add guardrails to reject queries that are misused or politically sensitive and to prevent unintended outputs.

Evaluation Criteria

- Practicality and clarity of the proposed solution
- Depth of understanding in document chunking, chunk embedding, and vector database
- Effectiveness of the RAG module and the guardrails (for the bonus question)

Problem 3: Computer vision for farming

You're a Computer Vision engineer and a client reaches out to you with a unique problem. The client is the owner of a strawberry farming company and they are planning to speed up their production estimation process by using computer vision. They are wondering if it would be possible to use CV in order to detect and count strawberries on trees.

For this, as a first proof of concept, they sent you a video they took of their farm and would like you to demonstrate that you can train an algorithm in order to detect, track and count the strawberries on their video.
After researching online, you found an open source dataset containing labeled images of strawberries that you are planning to use for your PoC development. You need to create a CV method that will ultimately take the test.mp4 video as input and return how many strawberries appear on the video.

This PoC will consist of 2 main components/steps:

1. Demonstrate the ability of computer vision to detect and locate strawberries in a static image. You can use the labeled dataset to train and evaluate such models and report the performance on this dataset.
2. Show the customer that we can apply some video processing techniques in order to extract the number of strawberries present in a video in the case of translating camera movement.

Some things to consider during development :

- Make sure to evaluate the strawberry detection or segmentation model for 1. using the open source data set eval/test set.
- In order to count the number of strawberries in the video, you can use any approach you want. However, a good option could be to use some tracking algorithm.
- You can apply tracking to individual strawberries or also potentially make use of their dense separated group setup.
- Every experiment and ideas are valued. You don't need to have a perfect strawberry detection model to start working on the second question.
- Outputting a video to showcase the results of the strawberry counting PoC is a big plus for the client to understand how well the method is working.

Evaluation Criteria

For the assignment that you choose, you will be tested on:
*1. **EDA and Conclusions**: Conduct an in-depth EDA on the chosen dataset and draw actionable insights. Reflect these insights in your feature engineering and modeling choices.*

*2. **Development**: Develop an appropriate algorithm/model using the chosen dataset. Justify your choices for your model/algorithm of choice, data splitting, and hyperparameter tuning. **Discuss the trade-offs involved**.*

*3. **Evaluation**: Clearly **explain** your evaluation criteria and the metrics of choice to gauge your algorithm/model's performance using these metrics. Discuss any issues encountered, such as overfitting or underfitting, lack of data, and how you mitigated them.*

*4. **Documentation**: Maintain thorough code documentation, including docstrings and inline comments. Good documentation includes not just what the code does, but **why** it does it.*

*5. **Reasoning**: As you work through your chosen problem, articulate your thought process. Explain the rationale behind your actions, your interpretations of the results, and your future steps.*

Submission Format

After completing your assignment, please submit your work using the following format:

- In case of codebase spanning multiple files (.py and/or jupyter notebooks), compress all your work into a .zip or .tar file
- For jupyter notebooks please ensure that your notebooks are clean, well-structured, well-documented-and are able to run without errors from start to finish taking the raw data as input
- You are required to submit the code ONLY.
- In case you clean the data or process it to become a different version from the original, only then you can submit the processed data files