DA5401 A4: GMM-Based Synthetic Sampling for Imbalanced Data

Objective: This assignment will challenge you to apply a sophisticated, model-based approach to tackle the class imbalance problem. You will use a Gaussian Mixture Model (GMM) to generate synthetic samples for the minority class, and then evaluate its effectiveness compared to a baseline model. This assignment focuses on the **theoretical and practical aspects of using probabilistic models for data augmentation**.

---

1. Problem Statement

You are a data scientist tasked with building a fraud detection model for a financial institution. You have been given a highly imbalanced dataset where a tiny fraction of transactions are fraudulent. Your main challenge is to create a training set that allows a classifier to learn the nuances of the minority (fraudulent) class without overfitting or misclassifying. You will implement a GMM-based synthetic data generation pipeline and analyze its impact on model performance.

You will submit a Jupyter Notebook with your complete code, visualizations, and a plausible story that explains your findings. The notebook should be well-commented, reproducible, and easy to follow.

Dataset: The dataset is available on Kaggle: Credit Card Fraud Detection.

---

2. Tasks

Part A: Baseline Model and Data Analysis [To Borrow from A3]

1. **Data Loading and Analysis:**
   - Load the `creditcard.csv` dataset.
   - Print the class distribution and discuss the degree of imbalance.
2. **Model Training:**
   - Split the dataset into training and testing sets. **Crucially, the test set should be an accurate reflection of the original class imbalance.**
   - Train a **Logistic Regression** classifier on the imbalanced training data to establish a performance baseline.
3. **Baseline Evaluation:**
   - Evaluate the model's performance on the test set. Explain why metrics such as **Precision, Recall, and F1-score for the minority class** are more informative than accuracy for this problem.

Part B: Gaussian Mixture Model (GMM) for Synthetic Sampling [35 points]

1. **Theoretical Foundation [5]:**

    - In a markdown cell, explain the fundamental difference between **GMM-based synthetic sampling** and simpler methods like SMOTE.
    - Discuss why GMM is theoretically better at capturing the underlying data distribution, especially when the minority class has multiple sub-groups or complex shapes in the feature space.
2. **GMM Implementation [10]:**
    - Fit a **Gaussian Mixture Model** to the **training data of the minority class only**.
    - Explain how you determined the optimal number of components (k) for the GMM. You can use a metric like the **Akaike Information Criterion (AIC)** or **Bayesian Information Criterion (BIC)** to justify your choice.
3. **Synthetic Data Generation [10]:**
    - Use the fitted GMM to **generate a sufficient number of new synthetic samples** to balance the dataset. Explain the process of sampling from a GMM.
    - Combine these newly generated samples with the original training data.
4. **Rebalancing with CBU [10]**:
    - Use clustering-based Undersampling on the majority dataset to bring it down to a suitable population.
    - Use GMM-based synthetic sampling on the minority dataset to match the majority population and hence create a balanced dataset.

Part C: Performance Evaluation and Conclusion [15 points]

1. **Model Training and Evaluation [5]:**
    - Train a new **Logistic Regression** classifier on the GMM-balanced training data (both versions).
    - Evaluate the model's performance on the same, original, imbalanced test set from Part A.
2. **Comparative Analysis [5]:**
    - Create a summary table or a bar chart comparing the Precision, Recall, and F1-score of the GMM-based model against the baseline model.
    - Discuss the impact of GMM-based oversampling on the classifier's performance. Did it improve the model's ability to detect the minority class?
3. **Final Recommendation [5]:**
    - Based on your analysis, provide a clear recommendation on the effectiveness of using GMM for synthetic data generation in this context. Justify your answer using both your results and your theoretical understanding of the method.

---

3. Submission Guidelines

- The assignment is due on **15th September 2025, 4.30 pm**. ***[2 pm if done remotely]***

- Submit a single Jupyter Notebook with all your code, visualizations, and answers to the conceptual questions.
- Ensure your code is clean, readable, and well-commented.

Evaluation Criteria:

- Correct implementation of the GMM-based oversampling pipeline.
- Correct use of evaluation metrics and a valid comparison with the baseline.
- Clear and insightful explanation of the theoretical concepts.
- Well-reasoned conclusions based on the empirical results.

PS: You may reuse the results from A3 in A4.

Good luck!