Welcome to Your Second Step of the Challenge!

Total Points: 100

In this challenge, we will generate the proper pipeline for processing, creating a model, training a model, and evaluating a model for protein classification. The dataset provided is `metadata_org_w_features.csv`.

This challenge is divided into two parts that should be addressed in the Notebook:

1. Coding: Focused on addressing the challenge of Machin Lering. (60 points)
2. Questions: Designed to explore and address the subcategories within the challenge. (40 points)

Data Overview:

The dataset contains the following columns:

1. Entry and Protein Class
2. Sequence and Sequence Length
3. Selected PDB
4. Amino Acid Frequencies
5. All possible dipeptide frequencies
6. Reduced Amino Acid Alphabet Frequencies
7. N-Gram Profiles of Reduced Amino Acid Alphabet
8. Protein Properties

Questions to Explore

1. Features

- What additional feature could significantly enhance the model's ability to differentiate between these protein classes? Consider biochemical, structural, or functional attributes. (2 points)
- How would you preprocess the data for this new feature to ensure it's in a format suitable for machine learning algorithms? For instance, does it require normalization, one-hot encoding, or other transformations? (2 points)
- How can you evaluate the impact of the newly added feature on the model's performance? Would methods like feature importance analysis or ablation studies help determine its contribution to the classification task? (2 points)
- Code (10 point)

2. Constructing the Dataset

- What criteria will you use to select and extract the relevant fields for constructing a high-quality dataset suitable for protein classification? (2 points)
- How will you handle missing or incomplete data within the extracted fields to ensure the dataset remains robust and reliable for training the machine learning model? (2 points)
- Coding section (20 points)

3. Processing the Dataset

- How does applying min-max normalization to the features (X_train and X_test) ensure consistent scaling, and why is this important for machine learning algorithms? (2 points)
- How does fitting the one-hot encoder on all possible labels improve the generalization of the model when transforming training and testing labels? (2 points)
- How can you verify that the transformations (min-max normalization and one-hot encoding) have been correctly applied and are functioning as intended across both the training and testing datasets? (2 points)
- Coding section (10 points)

4. Creating a Model

- Which factors should you consider when selecting a machine learning model (e.g., Random Forest, Linear Model, etc.) for this classification task, and how do these choices impact the model's performance? (2 points)
- How can you ensure that the chosen model's initialization parameters (e.g., number of estimators, learning rate) are appropriate for the dataset and the specific problem being solved? (2 points)
- Why is it important to document the library or framework (e.g., scikit-learn, PyTorch) used for model implementation, and how does the choice of library affect the model's flexibility and scalability? (2 points)
- Coding section (10 points)

5. Training a Model

- How does the choice of data format (raw, scaled, or other forms) impact the model's training process and its ability to generalize to unseen data? (2 points)
- What metrics or methods will you use to evaluate whether the model has been effectively fitted to the training data? (2 points)
- How can you ensure that the model fitting process does not lead to overfitting or underfitting, and what strategies could be employed to address these issues if they arise? (2 points)
- Coding section (10 points)

6. Evaluate your model

- Why is it important to evaluate and compare your model's performance across different evaluation metrics, such as accuracy, precision, recall, and F1 score, and what do these differences reveal about its strengths and weaknesses? (3 points)
- If you were to visualize your model's evaluation results (e.g., with confusion matrices, ROC curves, or precision-recall plots), what insights could these visualizations provide about its classification capabilities? (3 points)
- Suppose your model performs exceptionally well on the training set but poorly on the test set—how might you diagnose and address this discrepancy during the evaluation phase? (3 points)
- If you needed to present the evaluation results of your model to a non-technical audience, how would you simplify the findings while ensuring they still convey the model's effectiveness? (3 points)
- Coding section (10 points)