**Supervised Learning on European Topology 6 Paths Dataset**

**Task Overview**

We have a unique dataset derived from a European topology comprising 6 paths. This dataset includes information on 76 channels with specific features, and the goal is to predict the GSNR (Generalized Signal-to-Noise Ratio), making it a regression task. Your task as an intern will be to apply supervised learning methods to build a predictive model for GSNR using an Artificial Neural Network (ANN), perform hyperparameter tuning through grid search to find the best parameters, analyze the effects of learning rate, batch size, and other hyperparameters, and document your findings and methodologies in a detailed report.

**Dataset Description**

* **Channels**: 76
* **Features**:
  + Power: Signal power for each channel
  + NLI (Non-Linear Interference): Measure of interference on each channel
  + ASE (Amplified Spontaneous Emission): Noise figure for each channel
  + Total Distance: Total distance the signal travels
  + Span: Number of spans in the signal path
  + Frequency: Frequency of the channel (not significant for GSNR prediction)
* **Label**:
  + GSNR (Generalized Signal-to-Noise Ratio): Target variable to predict. Please select one GSNR column as the label and discard the rest from the dataset.

**Tasks and Expectations**

1. **Data Exploration and Preprocessing**:
   * Analyze the dataset to understand the distribution of features and the target variable.
   * Handle any missing values or anomalies in the data.
   * Perform feature scaling or normalization if necessary.
   * Visualize the relationships between features and the target variable.
2. **Feature Engineering**:
   * Consider creating new features or modifying existing ones to improve model performance.
   * Analyze feature importance to identify which features are most predictive of GSNR.
3. **Feature Selection**:
   * Apply feature selection methods to determine the significance of each feature.
   * Identify and discard the least important features, including the frequency feature, to simplify the model and potentially improve performance.
4. **Model Selection and Training**:

* Implement an Artificial Neural Network (ANN) for the regression task.
* Split the data into training, validation, and test sets to evaluate model performance.
* Use cross-validation techniques to ensure robust evaluation.
* Perform hyperparameter tuning using grid search to find the best parameters for the ANN, including learning rate, batch size, number of layers, and number of neurons per layer.
* Experiment with different network architectures, including varying the number of hidden layers and the number of neurons per layer.
* Test different activation functions such as ReLU, sigmoid, and tanh to determine which performs best.
* Explore different optimization algorithms like Adam, SGD, and RMSprop.
* Implement regularization techniques, including L1 and L2 regularization, to prevent overfitting.
* Apply dropout layers to reduce the risk of overfitting.
* Implement early stopping by monitoring the validation loss during training and stopping.

1. **Model Evaluation**:
   * Evaluate the ANN model using appropriate metrics for regression tasks (e.g., Mean Squared Error, R-squared, Mean Absolute Error).
   * Fine-tune the model parameters to optimize performance.
   * Analyze the effects of learning rate, batch size, and other hyperparameters on the model's performance.
   * Assess overfitting by comparing training and validation performance, and apply techniques like early stopping, dropout, and regularization.
2. **Reporting**:
   * Provide a detailed analysis of the results, including visualizations of model performance and feature importance (jupyter notebook).
   * Document the hyperparameter tuning process, showing the impact of different hyperparameters on model performance.
   * Explain the methods used to prevent overfitting and their effectiveness.
3. **Submission**:
   * Please submit the code and report via email by [28th June]. Ensure your code is well-organized and your report is clear and concise.