Signal Classification Prediction Challenge ML Project Report

Team Members

1. [Kothamasu Naga Venkata Aditya] – [IMT2023033]
2. [Abhijit Dibbidi] – [IMT2023054]
3. [Adwait Priyadarshan] – [IMT2023502]

GitHub Link

`[https://github.com/erakin027/Signal-Cluster-Classification]`

Contents

1 Task 4

2 Dataset Description 4

2.1 Overview . . . 4

2.2 Variables . . . 4

3 Pre-processing & EDA 5

3.1 Initial Data Exploration . . . 5

3.2 Handling Missing Values and Duplicates . . . 5

3.3 Outlier Detection . . . 5

4 Data Preprocessing 5

4.1 Encoding Categorical Variables . . . 6

4.2 Dropping Useless Features . . . 6

4.3 Feature Correlation Analysis . . . 6

4.4 Standardization . . . 6

4.5 Dataset Splitting . . . 7

5 Exploratory Data Analysis 7

5.1 Feature Space Visualization . . . 7

5.2 Feature Relationships . . . 7

6 Models Tested 8

6.1 Clustering Models . . . 8

6.1.1 K-Means Clustering . . . 8

6.1.2 Gaussian Mixture Model (GMM) . . . 8

6.2 Classification Models . . . 8

6.2.1 Logistic Regression . . . 8

6.2.2 Support Vector Machine (SVM) . . . 8

6.2.3 Neural Network - Multilayer Perceptron (MLP) . . . 10

6.2.4 PyTorch Neural Network . . . 11

7 Model Comparison and Results 11

7.1 Validation Set Performance Rankings . . . 11

7.2 Final Model Selection . . . 11

7.3 Test Set Predictions . . . 12

8 Key Observations 12

9 Conclusion 13

1 Task

The primary objective of this project is to develop and compare multiple machine learning models to accurately predict signal categories (Group A, Group B, Group C) based on signal strength and response level measurements. This represents a classic supervised multiclass classification problem where the goal is to build predictive models using a training dataset containing signal metrics with their corresponding category labels. The trained models are then applied to a testing dataset containing input features without labels. The ultimate aim is to identify, tune, and deliver the best-performing classification model for predicting signal categories with high accuracy.

2 Dataset Description

2.1 Overview

The Signal Classification Dataset is designed to examine the factors influencing signal category determination. It contains 1,800 signal samples, with each record including signal strength and response level measurements, along with a ground-truth category classification. The objective of this challenge is to build predictive models that can estimate a signal's category based on its physical characteristics.

The training dataset `train.csv` consists of 1,438 rows and 4 columns, with the testing dataset `test.csv` on which categories need to be predicted consisting of 362 rows and 3 columns.

2.2 Variables

Table 1: Dataset Features and Description

Feature	**Description**
Sample ID	A unique identification number for each signal sample
Signal Strength	A numerical measure of the signal's intensity or power level (float)
Response Level	A numerical measure of the signal's response characteristics (float)
Target Variable	
Category	Classification of the signal into one of three groups: Group A, Group B, or Group C

3 Pre-processing & EDA

3.1 Initial Data Exploration

The dataset is analyzed by examining the first few rows, checking the data types of each column, and identifying any missing values. A statistical summary of the numerical columns is generated to assess the central tendency, variability, and potential outliers.

Data Types Detected:

- `sample_id`: int64
- `signal_strength`: float64
- `response_level`: float64
- `category`: int64 (after label encoding)

3.2 Handling Missing Values and Duplicates

A thorough examination of the dataset was conducted to identify missing values and duplicate records. The analysis revealed:

- **Missing Values**: 0 NaN values detected across all columns
- **Duplicates**: 0 duplicate records found
- **Data Quality**: The dataset is complete and requires no data imputation or removal

3.3 Outlier Detection

Outlier detection was performed using Z-score analysis on numerical features (`signal_strength` and `response_level`). Using a threshold of $|z| > 3$ to identify extreme values:

Outliers Found: 0

The absence of outliers indicates high data quality and clean signal measurements without extreme anomalous values that could skew model results.

4 Data Preprocessing

The dataset is prepared for modeling by completing the following tasks:

4.1 Encoding Categorical Variables

Categorical variables are encoded using Label Encoding to convert them into numerical format suitable for machine learning algorithms. The `category` feature mapping is as follows:

Table 2: Category Label Encoding

Category	**Encoded Value**
Group_A	0
Group_B	1
Group_C	2

4.2 Dropping Useless Features

The `sample_id` feature was dropped as it serves only as an identifier and has no correlation with signal category classification. This decision is supported by domain knowledge.

4.3 Feature Correlation Analysis

Correlation analysis was performed to understand relationships between features and the target variable:

Table 3: Correlation with Target Category

Feature	**Correlation with Category**
signal_strength	0.5752
response_level	-0.4130
sample_id	-0.0288

4.4 Standardization

Numerical features (`signal_strength` and `response_level`) are standardized using StandardScaler after the train-test split to avoid data leakage. Standardization ensures all features have similar scales with mean 0 and standard deviation 1, which is crucial for:

- Distance-based algorithms (SVM, KMeans, GMM)
- Neural networks and gradient-based optimization
- Ensuring fair feature importance across different scales

Important Note: Standardization is performed after train-test split to prevent data leakage. The scaler is fit on the training set and then applied to both validation and test sets.

4.5 Dataset Splitting

The dataset is divided into training and validation sets using stratified train-test split with an 80/20 ratio:

- Training set: 1,150 samples (80%)
- Validation set: 288 samples (20%)

Stratification ensures that each split maintains the same class distribution as the original dataset, enabling proper assessment of generalization ability.

5 Exploratory Data Analysis

5.1 Feature Space Visualization

Scatter plots were generated to visualize the 2D feature space and understand how the three signal categories separate in the signal strength vs. response level plane. The visualizations reveal:

- Clear separation between signal categories
- Linear decision boundaries are not sufficient
- Non-linear classifiers may perform better

5.2 Feature Relationships

Correlation heatmap analysis reveals:

- Signal Strength shows positive correlation (0.5752) with category
- Response Level shows negative correlation (-0.4130) with category
- No multicollinearity issues detected between numerical features
- Features demonstrate sufficient variance to support meaningful classification

6 Models Tested

Multiple machine learning models were trained and evaluated to identify the optimal approach for predicting signal categories.

6.1 Clustering Models

6.1.1 K-Means Clustering

K-Means clustering was applied with 3 clusters to match the number of signal categories. The cluster assignments were mapped to true labels using majority voting:

Validation F1 Score (Macro): 0.8326

6.1.2 Gaussian Mixture Model (GMM)

Gaussian Mixture Model with 3 components was employed to probabilistically assign samples to categories:

Validation F1 Score (Macro): 0.6942

GMM showed lower performance compared to K-Means, suggesting that the clusters do not follow Gaussian distributions.

6.2 Classification Models

6.2.1 Logistic Regression

Initial logistic regression with default parameters achieved F1 score of 0.8678. Further tuning was performed with hyperparameters:

- `max_iter`: 300
- `C`: 0.1 (inverse regularization strength)
- `solver`: lbfgs
- `class_weight`: balanced

Validation F1 Score (Macro): 0.8814

6.2.2 Support Vector Machine (SVM)

Multiple SVM configurations were tested:

SVM (Default RBF): Initial SVM with RBF kernel achieved:

Validation F1 Score (Macro): 0.9900

SVM (Tuned RBF): Tuned SVM with optimized hyperparameters:

- `kernel`: rbf
- `C`: 15
- `gamma`: scale
- `class_weight`: balanced

Validation F1 Score (Macro): 1.0000

Perfect F1 score indicates excellent separation and classification of all three signal categories on the validation set.

SVM (GridSearchCV Optimized): Systematic hyperparameter search was performed using GridSearchCV with stratified 5-fold cross-validation over:

- kernel: [linear, rbf]
- C: np.linspace$(1, 10, 100)$
- gamma: [scale, auto]

Optimal parameters identified:

- `kernel`: rbf
- `C`: 9.909
- `gamma`: scale

Validation F1 Score (Macro): 0.9950

6.2.3 Neural Network - Multilayer Perceptron (MLP)

MLP Configuration 1: Single hidden layer configuration:

- `hidden_layer_sizes`: (64,)
- `activation`: relu
- `solver`: adam
- `alpha`: 0.0001
- `learning_rate_init`: 0.001
- `max_iter`: 500

Validation F1 Score (Macro): 0.9855

MLP Configuration 2: Deep network with three hidden layers:

- `hidden_layer_sizes`: (128, 64, 32)
- `activation`: relu
- `solver`: adam
- `alpha`: 0.001
- `learning_rate_init`: 0.001
- `max_iter`: 1000
- `early_stopping`: True
- `validation_fraction`: 0.1
- `n_iter_no_change`: 20

Validation F1 Score (Macro): 0.9746

6.2.4 PyTorch Neural Network

A custom PyTorch-based MLP classifier was implemented with scikit-learn compatibility. GridSearchCV optimization was performed over:

- `hidden_dim`: [32, 64, 128]
- `hidden_layers`: [1, 2, 3]
- `dropout`: [0.1, 0.2, 0.3]
- `epochs`: [20, 30, 40]
- `lr`: [0.0001, 0.0003, 0.001]

Optimal parameters identified:

- `hidden_dim`: 64
- `hidden_layers`: 2
- `dropout`: 0.1
- `epochs`: 40
- `lr`: 0.0003

Validation F1 Score (Macro): 0.9664

7 Model Comparison and Results

7.1 Validation Set Performance Rankings

7.2 Final Model Selection

The tuned SVM with RBF kernel achieved perfect F1 score on the validation set, demonstrating exceptional performance. However, for conservative generalization, the GridSearchCV-optimized SVM was selected for test set predictions, as it represents the result of systematic hyperparameter optimization.

Table 4: Model Comparison by Validation F1 Score (Macro)

Rank	**Model**	**F1 Score**
1	SVM (Tuned RBF)	1.0000
2	SVM (GridSearchCV)	0.9950
3	SVM (Default RBF)	0.9900
4	Neural Network (64,)	0.9855
5	Neural Network (128,64,32)	0.9746
6	PyTorch Neural Network	0.9664
7	Logistic Regression	0.8814
8	K-Means	0.8326
9	GMM	0.6942

7.3 Test Set Predictions

The GridSearchCV-optimized SVM model was applied to the test dataset (362 samples) to generate category predictions. The output submission file consists of:

- Sample ID from test set
- Predicted Category (Group A, Group B, or Group C)

Total predictions generated: 362 samples

The predicted categories were decoded from their numerical representations back to original category labels using the inverse transformation of the LabelEncoder.

8 Key Observations

1. **Data Quality**: The absence of missing values, duplicates, and outliers indicates a well-curated dataset with excellent data integrity.

2. **Feature Sufficiency**: Two numerical features (signal strength and response level) provide sufficient information for effective multiclass classification, with clear discriminative power as evidenced by correlation analysis.

3. **Model Performance Hierarchy**: SVM significantly outperforms other algorithms, suggesting non-linear decision boundaries are essential for optimal classification. Neural networks also perform well, confirming the presence of non-linear patterns.

4. **SVM Superiority**: The RBF kernel's perfect validation F1 score indicates that the signal categories occupy well-separated regions in the 2D feature space with curved decision boundaries.

5. **Clustering vs Classification**: Classification models substantially outperform clustering approaches (GMM and K-Means), as supervised learning allows models to directly optimize for category prediction rather than discovering underlying cluster structure.
6. **Neural Network Insights**: While neural networks achieved high performance (F1 > 0.96), their results were slightly inferior to SVM, suggesting that for this dataset, simpler non-linear models are more effective than deep architectures. This indicates that the complexity of multi-layer networks is not necessary to capture the patterns in the data.
7. **Stratified Splitting**: The use of stratified train-test split ensured balanced class representation across training and validation sets, preventing biased model evaluation.
8. **Data Leakage Prevention**: Standardization was deliberately performed after train-test splitting to prevent information leakage from the training distribution into scaling parameters.

9 Conclusion

The Support Vector Machine with Radial Basis Function (RBF) kernel emerges as the optimal model for signal category classification. The systematic hyperparameter optimization via GridSearchCV identified the configuration:

- kernel: rbf
- C: 9.909
- gamma: scale

This model achieved a validation F1 score of 0.9950 and demonstrated robust performance characteristics. The SVM's superior performance compared to all other tested algorithms (neural networks, logistic regression, GMM, and K-Means) confirms that:

1. The signal categories possess non-linear decision boundaries
2. The feature space is sufficiently rich for accurate classification
3. RBF kernels are well-suited to the geometric structure of the problem

The model successfully generates predictions for all 362 test samples and has been exported for submission. The rigorous preprocessing pipeline, careful hyperparameter tuning, and comprehensive model comparison demonstrate best practices in machine learning project execution, resulting in a robust, well-validated classification system capable of accurately predicting signal categories.