Rock vs Mine Prediction Using Machine Learning

by Mayank Bharti

Abstract

Sonar signal classification is a critical problem in underwater navigation and defense applications. This project focuses on the binary classification of sonar signals into Rock (R) or Mine (M) using machine learning techniques. Multiple supervised learning algorithms were implemented and evaluated under a unified preprocessing and validation framework. Proper feature scaling, pipeline-based modeling, and robust evaluation metrics were employed to ensure reliable performance. The results demonstrate that Support Vector Machines outperform other models on this dataset due to their effectiveness on small, high-dimensional data.

1 Introduction

Underwater object detection plays an essential role in marine exploration, navigation safety, and military defense systems. Sonar technology is commonly used to detect submerged objects by analyzing reflected acoustic signals. However, interpreting sonar signals manually is complex and error-prone due to noise and environmental variability.

Machine learning provides an automated and data-driven approach for identifying meaningful patterns in sonar data. This project applies various machine learning algorithms to classify sonar signals as either rock or mine. The primary goal is to evaluate and compare different models to identify the most suitable classifier for this task.

2 Problem Statement

The objective of this project is to build a machine learning model capable of accurately classifying sonar signals into two categories: rock and mine. The major challenges include handling high-dimensional feature space, limited sample size, and preventing model overfitting. A robust evaluation framework is required to ensure unbiased and reliable results.

3 Dataset Description

The dataset used in this project is the Sonar Dataset, a widely used benchmark dataset for binary classification. It consists of 208 observations, each having 60 numerical features. Each feature represents the energy of a sonar signal at a specific frequency band. The target variable has two classes:

- R – Rock
- M – Mine

The dataset is relatively small but high-dimensional, making model selection and regularization important considerations.

4 Exploratory Data Analysis

Initial exploratory analysis revealed that the dataset is moderately balanced between the two classes. Statistical summaries showed significant variation in feature scales, indicating the need for normalization. Correlation analysis suggested that many features are interdependent, which can impact probabilistic models such as Naive Bayes.

Exploratory analysis helped guide preprocessing decisions and model selection.

5 Data Preprocessing

Data preprocessing is a crucial step in machine learning pipelines. The dataset was divided into features and target variables. An 80:20 train-test split was applied with stratification to preserve class distribution.

Feature scaling was performed using StandardScaler. To prevent data leakage, scaling was applied only on the training data and then transferred to the test data using pipelines. This approach ensures that no information from the test set influences model training.

6 Machine Learning Algorithms

The following supervised learning algorithms were implemented and evaluated:

6.1 Logistic Regression

Logistic Regression was used as a baseline model due to its simplicity and interpretability. Class-weight balancing was applied to handle class distribution. Despite its linear nature, the model performed competitively on the dataset.

6.2 K-Nearest Neighbors

K-Nearest Neighbors is a distance-based classifier that assigns labels based on neighboring samples. The performance of KNN is highly sensitive to feature scaling and the choice of hyperparameters. Grid search was used to identify the optimal number of neighbors.

6.3 Support Vector Machine

Support Vector Machine constructs an optimal decision boundary by maximizing the margin between classes. The RBF kernel was used to capture non-linear relationships. SVM performed exceptionally well due to the high dimensionality and small size of the dataset.

6.4 Naive Bayes

Gaussian Naive Bayes is a probabilistic classifier based on the assumption of feature independence. Since many features in the dataset are correlated, this assumption was violated, leading to comparatively lower performance.

6.5 Decision Tree

Decision Trees provide interpretable decision rules but are prone to overfitting. To mitigate this issue, depth constraints were applied. However, performance remained unstable due to the limited dataset size.

6.6 Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees. It improved generalization compared to a single tree but was constrained by the dataset size and feature redundancy.

7 Model Evaluation

Model performance was evaluated using the following metrics:

- Accuracy
- Precision
- Recall
- F1-score

- Confusion Matrix

Cross-validation was employed to ensure stability and robustness of results. Accuracy alone was not relied upon, as it can be misleading in imbalanced datasets.

8 Results and Discussion

Among all models evaluated, Support Vector Machine achieved the highest and most consistent accuracy. Logistic Regression served as a strong baseline model. KNN showed high accuracy but was sensitive to parameter selection and data splits. Tree-based models exhibited overfitting tendencies, while Naive Bayes underperformed due to violated assumptions.

These results highlight the importance of aligning model choice with data characteristics.

9 Conclusion

This project successfully demonstrated the application of multiple machine learning algorithms for sonar signal classification. Proper preprocessing, pipeline-based modeling, and robust evaluation significantly improved model reliability. Support Vector Machine emerged as the most suitable classifier for the given dataset.

10 Future Scope

Future enhancements to this work may include:

- Dimensionality reduction using PCA
- Deep learning models such as neural networks
- ROC-AUC and Precision-Recall analysis
- Real-time deployment of the classification system

11 Tools and Technologies Used

- Python
- NumPy and Pandas
- Scikit-learn
- Jupyter Notebook