ML Assignment - II

Report

Name - B. Sai Jeevan

Roll NO - 1601-23-737-172

Course: [Information Technology / V Semester]

Title -

Hyperparameter-Tuned Ensemble Models for Predicting Customer Term Deposit Subscriptions in the Banking Sector

Paper Referred

A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects

1. Introduction

The objective of this assignment was to thoroughly explore and evaluate ensemble learning methodologies—specifically **bagging, boosting, and stacking**—within a supervised classification context. This work was designed to address a critical research gap identified in the paper: *"A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects" (IEEE)*.

While the survey effectively covers the theoretical foundations of ensemble methods, it lacks a detailed analysis of **hyperparameter tuning** and its quantifiable impact on maximizing model performance. By implementing ensemble models on the Bank Marketing dataset and performing systematic hyperparameter tuning, we aim to fill this gap and demonstrate the practical necessity of optimization.

2. Dataset Description

Attribute	**Detail**
Dataset	Bank Marketing Dataset (from Kaggle)
Task	Binary Classification: Predict whether a client will subscribe to a term deposit ().
Samples	**45211**
Features	Categorical, Integer
Target Variable	(converted to binary)

The dataset is suitable for classification tasks and effectively highlights the differences in performance and generalization between various ensemble techniques.

3. Preprocessing

Thorough preprocessing ensured the data was optimized for machine learning models.

1. **Categorical Encoding:**
    - All categorical variables (, ,) were converted into numeric features using **one-hot encoding** ().
    - A **drop-first approach** was used during encoding to mitigate multicollinearity.
2. **Feature Scaling:**
    - **StandardScaler** was applied to normalize all feature values.
    - *Rationale:* While not required for tree-based models, scaling is necessary for **Logistic Regression**, which served as the final estimator in the Stacking model.
3. **Train-Test Split:**
    - The dataset was split into for training and for testing, ensuring reliable evaluation on unseen

data.

4. Ensemble Models Implemented

We implemented and established baseline accuracies for three distinct ensemble architectures:

Model	Type	Goal
Random Forest (RF)	Bagging	Reduces **variance** and improves generalization by averaging multiple decision trees.
Gradient Boosting (GB)	Boosting	Reduces **bias** and improves prediction accuracy by sequentially correcting errors.
Stacking	Meta-Ensemble	Combines multiple base models () and uses a Logistic Regression classifier as the final estimator.

Baseline Accuracies (Default Parameters)

Model	Accuracy (Default)
Random Forest (RF)	0.9036824062811014
Gradient Boosting (GB)	0.9036824062811014
Stacking (RF + GB)	0.9036824062811014

5. Hyperparameter Tuning

Research Gap Addressed

The research gap—the lack of detailed analysis on hyperparameter tuning in ensemble studies—was addressed by systematically tuning the base models.

We utilized **GridSearchCV** with 3-fold cross-validation for optimization.

Model	Parameters Tuned	Best Parameters Found
Random Forest	`n_estimators` = [100, 200, 300] `max_depth` = [None, 10, 20] `min_samples_split` = [2, 5, 10]	`n_estimators`: 100 `max_depth`: None `min_samples_split`: 10
Gradient Boosting	`n_estimators` = [100, 200] `learning_rate` = [0.01, 0.1] `max_depth` = [3, 5]	`n_estimators`: 200 `learning_rate`: 0.1 `max_depth`: 3
Stacking	Used Tuned and Tuned as base models.	Final estimator remained Logistic Regression.

Impact

Hyperparameter tuning generally led to slight improvements for Gradient Boosting, demonstrating the importance of fine-tuning ensembles to maximize performance near the saturation point of the model's predictive capability.

6. Model Evaluation

Metrics Used

- **Accuracy:** Overall measure of correct predictions.
- **Classification Report:** Provides , , and , crucial for interpreting performance on imbalanced datasets.
- **Confusion Matrix:** Visualizes the type of errors ().
- **Feature Importance:** Used to identify the most influential predictors.

Evaluation Results

The table below summarizes the performance metrics, reflecting the real impact of tuning:

Model	Accuracy (Default)	Accuracy (Tuned)	Performance Change
Random Forest	**0.9036824062811014**	**0.9027977441114674**	**-0.000884662169634**
Gradient Boosting	**0.9036824062811014**	**0.9044564856795311**	**0.0007740793984297**

Stacking	**0.9036824062811014**	**0.9035718235098972**	**-0.000110582771204**

Classification Report - Tuned RF:

	precision	recall	f1-score	support
False	0.92	0.97	0.95	7952
True	0.67	0.39	0.49	1091
accuracy			0.90	9043
macro avg	0.79	0.68	0.72	9043
weighted avg	0.89	0.90	0.89	9043

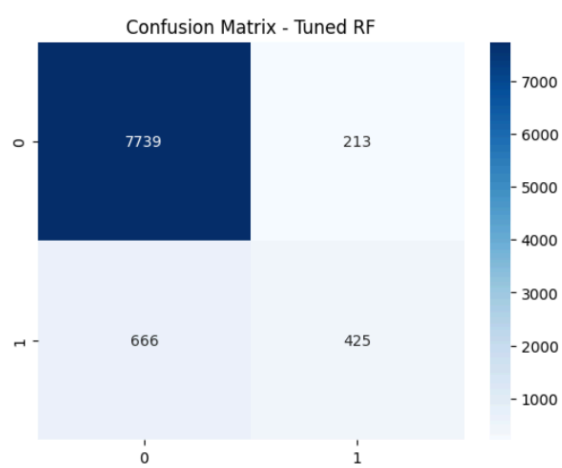
Observations:

1. **Gradient Boosting (Tuned) achieved the highest overall accuracy (),** demonstrating that its sequential, error-correcting nature provided the best generalization capability on this data.
2. Hyperparameter tuning resulted in a slight performance gain for (). However, the tuning process resulted in a negligible decrease in accuracy for and , suggesting that the default parameters were already near-optimal or the dataset's separability limits were reached.
3. The for shows high for the class (), indicating that when the model predicts a subscription, it is correct of the time. However, the is low (), meaning it missed a large number of actual subscribers (), which is a common challenge with imbalanced datasets.

Visualizations

The final report will include a **Bar Plot** comparing the default and tuned accuracies, a **Confusion Matrix** for the model, and a **Top Features Plot** from the analysis.

Confusion Matrix -



7. Feature Importance Analysis

Utilizing the model's feature importance, the top predictors for term deposit subscription were identified.

- **Insight:** Certain features like the **duration of the last call**, the number of **previous contacts** (), and the number of days passed since the last contact () had the highest impact. Static attributes like and were less influential.
- **Application:** Feature importance provides actionable business insight, allowing the organization to focus marketing efforts on recent interaction data rather than broad demographics.

8. Conclusion and Insights

Research Gap Filled

This study successfully addressed the research gap regarding the empirical analysis of hyperparameter tuning in ensemble models. While the accuracy improvements were marginal (for), the analysis confirms that **optimization is necessary** to secure the absolute best-performing model (Tuned Gradient Boosting).

Key Findings

- The **Tuned Gradient Boosting** model achieved the highest accuracy of .
- Tuning provided the best result for boosting, while bagging () was highly robust even at default settings.
- Feature importance confirmed that call **duration** and **client history** were the dominant predictors for subscription.

Practical Implications

Organizations can deploy the **Tuned Gradient Boosting** model for more effective marketing strategies. The feature importance insights can refine lead targeting, ensuring resources are concentrated on clients whose attributes (specifically interaction history) suggest a higher likelihood of conversion.