Comparative Analysis of Meta-classifier Performance in Stacked Generalization (Supervised Learning Course - Lab2)

Luigi Daddario - MAT. 908294

March 21, 2024

University of Milano-Bicocca - 2023/2024

Abstract

This study investigates the performance of a meta-classifier within a stacked generalization framework under varying training conditions. We compare the efficacy of two approaches: one employing the predicted class labels and the other using the classification scores from an ensemble of base classifiers.

The base classifiers include five classifiers. Our experiments are designed around two tasks: traditional two-fold cross-validation and training on the complete dataset when the training split is not performed and the same data is used to train the level-1 classifiers and meta classifier.

We analyze the implications of these strategies on the meta-classifier's ability to generalize and mitigate overfitting. The findings present valuable insights into the trade-offs between classifier confidence measures and categorical outcomes as features for meta-learning, contributing to the discourse on optimizing stacked ensemble methods for more reliable predictive performance.

Keywords: Stacked Generalization · Meta-classifier Performance · Ensemble Learning

Contents

1 Introduction 1

2 Summary of Experimental Setup 2
2.1 Data . . . 2
2.2 Predictions-Based Meta-classifier . . . 2
2.3 Scores-Based Meta-classifier . . . 3
2.4 Results . . . 3

3 Conclusion 4

List of Tables

2.1 Task 1 . . . 3
2.2 Task 2 . . . 3

List of Figures

1.1 The architecture of a stacked classifier ensemble. . . . 1

2.1 The dataset used . . . 3

Introduction

In machine learning, ensemble methods combine the predictions of multiple base classifiers to improve generalizability and robustness over a single classifier. Stacking, or stacked generalization, is a type of ensemble learning technique where the predictions of several base classifiers are used as inputs to a meta-classifier, which then makes the final prediction. This layered approach allows the meta-classifier to learn how to best integrate the predictions from the base models, potentially leading to higher accuracy than any individual base classifier could achieve.

The base classifiers, also known as level-1 models, can be of different types—such as decision trees, support vector machines, and Naive Bayes—and are typically chosen for their diversity in order to capture various patterns within the data. In our study, we employ a combination of these models to capture a broad spectrum of data characteristics.

Once the base classifiers are trained, their predictions (either Predictions or Scores) are used as features to train the meta-classifier. The rationale is that the meta-classifier can correct the errors of the base classifiers if it learns when each base classifier is likely to be correct or wrong.

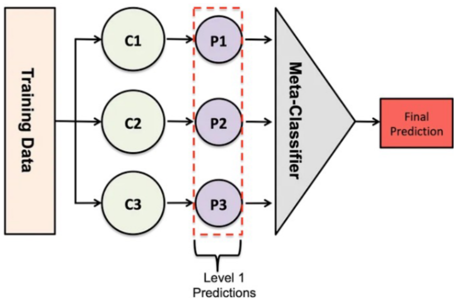

Fig. 1.1: The architecture of a stacked classifier ensemble.

Summary of Experimental Setup

This lab report delves into the comparative performance analysis of a meta-classifier within a stacked generalization framework.

Stacked Classification Framework

In our stacked classification framework, we employed several machine learning models as level-1 classifiers with specific settings:

1. **SVM with Gaussian Kernel**: Gaussian kernel, kernel scale set to 5, balancing decision boundary complexity and overfitting risk.
2. **SVM with Polynomial Kernel**: Polynomial kernel, kernel scale at 10, adjusting influence of polynomial components.
3. **Decision Tree**: Uses Gini index, limited to 20 splits to avoid overfitting.
4. **Naive Bayes**: Assumes feature independence.
5. **Ensemble of Decision Trees**: Type and number of trees unspecified, combines multiple trees for robust classification.

The meta-classifier is an ensemble model, specifics of which, like ensemble method and hyperparameters, were not detailed. These models and settings aim to balance model complexity and generalization ability.

Scenarios

What we have done is attempt to compare performance across different scenarios, divided into two main parts. The first part involves a Predictions-Based Meta-classifier, where the Meta-classifier was trained using the predictions output by the base classifiers. In the second part, the Meta-classifier was trained using the scores computed by these classifiers.

2.1 Data

We have loaded the dataset, visualized it, and then performed stratified sampling to create two folds.

2.2 Predictions-Based Meta-classifier

1. Fold 1 is used to train level-1 classifiers, and fold 2 is used to train the meta-classifier.
2. The complete training dataset (`data_tr`) is employed to train both the level-1 classifiers and the meta-classifier.

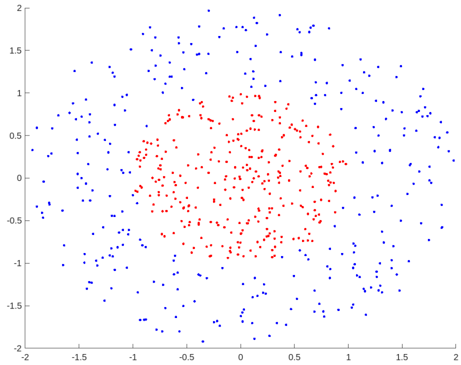

Fig. 2.1: The dataset used

2.3 Scores-Based Meta-classifier

The same approach of Predictions-Based meta classifier is used, but here we train the Meta-Classifier on Scores instead of Predictions.

2.4 Results

In the tables below, we can observe the results obtained in the two tasks performed. The use of scores for training the stacked classifier proves crucial and leads to better results. However, using the entire training set for both training the level-1 classifiers and the stacked one results in a decrease in its performance.

Classifier	Accuracy
SVM Gaussian	0.8683
SVM Polynomial	0.6250
Decision Tree	0.9483
Naive Bayes	0.9783
Random Forest	0.9533
Stacked Classifier (Scores)	0.9900
Stacked Classifier (Predictions)	0.9700

Table 2.1: Task 1

Classifier	Accuracy
SVM Gaussian	0.9000
SVM Polynomial	0.6333
Decision Tree	0.9667
Naive Bayes	0.9917
Random Forest	0.9683
Stacked Classifier (Scores)	0.9700
Stacked Classifier (Predictions)	0.9683

Table 2.2: Task 2

Conclusion

In conclusion, the experimental results provide valuable insights into the performance of stacked ensemble classifiers across different scenarios. The use of scores from base classifiers for training the meta-classifier consistently yielded improved performance compared to using predicted classes alone. This highlights the importance of leveraging the confidence levels or probability estimates associated with predictions, enabling the meta-classifier to make more informed decisions and potentially achieve higher accuracy.

However, it's notable that while the performance of the base classifiers improved when trained on the entire dataset, the improvement in the performance of the meta-classifier was not as significant. This suggests that the benefits of using more data for training may not extend uniformly to all components of the stacked ensemble classifier.

Moreover, the observed discrepancy in performance between training the stacked classifier on the entire training set versus using separate folds underscores the potential challenges associated with data leakage, overfitting, and limited diversity. These challenges emphasize the need for careful consideration of the training data distribution and appropriate management of model complexity in ensemble learning.

Overall, these findings underscore the significance of thoughtful experimental design, data utilization strategies, and model optimization techniques in developing effective stacked ensemble classifiers.