Unit 1 - Cost of Teacher and Test - Evaluation, Deployment And Monitorisation Of Models

Ole Magnus Laerum[1] and Piotr Franc[2]

Polytechnic University of Valencia (UPV), Spain

Abstract. This is a report of Task of Unit 1 of the class "Evaluation and Development of Models" at Polytechnic University of Valencia. This report concerns two tasks of evaluation of ML models, the first regarding "Cost of Teacher", the second of "Cost of Test".

1 Introduction

1.1 Cost of Teacher

Machine learning models often require a lot of labeled data for training, and it is often both time-consuming and costly to acquire the labels for the data. This task and part of Unit 1 addresses this problem by reducing the amount of labeled data needed for a good machine learning model.

To achieve this, this unit divides the data from the BostonHousing dataset with the target value *cmedv* (median value of owner-occupied homes) into three parts, **train1, train2** and **test**. Firstly, different techniques are tested on the labeled and "small" dataset of **train1**. After finding a model, **train2** is treated as a non-labeled data set, of which instances are selected incrementally to further train the model. The model is also incrementally tested using the **test** data set by RMSE (Root Mean Squared Error) [1].

This method simulates a realistic semi-supervised machine learning scenario [1], where the cost of teaching is reduced while still achieving high accuracy and low RMSE.

1.2 Cost of Test

In machine learning applications, both the cost of acquiring attribute values and the cost of incorrect predictions impact the overall model effectiveness. The Cost of Test problem examines how to balance these competing factors to minimize total costs.

Using the Breast Cancer dataset, this task evaluates attribute combinations with varying test costs and misclassification penalties to identify the best predictive power relative to acquisition cost. This cost-sensitive approach is crucial in fields like medical diagnostics, where balancing accuracy and cost is essential for viable models.

2 Methodology - Cost of Teacher

The approach to solve this task is split into two parts. First, different models are explored on the "smaller" and labeled dataset **train1**. Secondly, data are incrementally and "blindly" added from **train2** and tested on **test**.

The models tested and explored in this unit are **Linear Regression, Decision Tree, Random Forest, Extra Trees, Gradient Boosting, XGBoost, AdaBoost Regressor, Bagging Regressor** and **Stacking**.

Linear regression is perhaps the simplest method here, only fitting a linear equation to the data by minimizing the squared error. It is not very robust, as it is sensitive to outliers, but it is very interpretable [2]. Not a good model if the relationships in the data are not linear.

A **decision tree** method recursively divides the data into subsets based on characteristics. It is prone to overfitting, but can be tuned into better robustness by changing the maximum depth of the trees. It also is very interpretable, as the decisions can be visualized as a tree, thus the name.

Random Forest is a widely used machine learning practice that sets up a ensemble of decision trees on bootstrapped data, and then averages the predictions of the decision trees. Due to the averaging of the decision trees, it is very robust in regards to overfitting and noisy data.

Extra Trees is a different type of ensemble than random forest, that has a higher degree of randomness by choosing the splits at random rather than optimal. This often leads to lower variance but could increase bias. It differs from random forest also in that it uses the whole dataset to build each tree, instead of bootstrapping as in random forest.

Gradient boosting is an ensemble technique that builds models sequentially by training each new model to correct the errors of the previous trained models by minimizing a loss function. It is sensitive to noise, and is not very interpretable. **XGBoost** on the other hand [3], is a highly optimized, scalable and regularized version of gradient boosting.

AdaBoost Regressor is a popular version of the boosting technique [3], that iteratively emphasizes the instances of previous learners that were predicted poorly. Can be sensitive to noise, since outliers can be interpreted as poor predictions.

Bagging regressor is a machine learning method that trains multiple bootstrapped subsets of the data, and then combines the outputs of the multiple base models [3]. It is quite robust to noise as it reduces variance.

Stacking regressor is the last model used in this unit, and it simply combines multiple models by training a meta-model based on the predictions of the many models [3]. Can be very hard to interpret, but can nevertheless be excellent at prediction and robustness, as it combines the strengths of several methods.

Of these nine models and methods, the hypothesis for this task is that the stacking regressor will perform best, as it combines the benefits of several models.

2.1 Model exploration on labeled data

To explore these different models, the Python code runs trains and evaluates each model in the function *train_ and_ evaluate()* with the labeled data from **train1**. The RMSE and R^2 scores are saved, sorted by RMSE and printed and plotted in an increasing manner, best to worst.

,

2.2 Incremental instance selection from unlabeled data

For the second part, the top five models are listed and incrementally trained with **train2**. The **train2** dataset is randomly shuffled before this to simulate selection of instances from unlabeled data. The models are then incrementally trained with increasing amount of instances, and the RMSE scores are saved for each iteration. Finally, all results are plotted in a graph.

3 Methodology - Cost of Test

3.1 Model Training Methodology

Our primary goal is to minimize the total cost, which includes both misclassification costs and the expense of acquiring diagnostic attributes. We implement a Random Forest classifier with a deliberately lowered classification threshold of 0.1 to address the asymmetric cost structure where false negatives (20 units) are penalized five times more heavily than false positives (4 units).

3.2 Choosing the parameters

The methodology for the Cost of Test task employs a two-phase approach using the Breast Cancer dataset. First, we evaluate each attribute individually by training Random Forest models using a single attribute at a time. For each model, we calculate three costs: the error cost, the test cost, and the total cost (sum of error and test costs). Based on these individual evaluations, we identify Marg.adhesion as the most promising attribute with the lowest total cost.

Depending on the results, we can apply different strategies to select the optimal set of attributes to create our model:

If test costs are a minor factor, we can prioritize relevant metrics that assess attribute quality, allowing for more complex models with a larger number of attributes.

If test costs dominate the overall cost, minimizing the number of attributes becomes a priority.

If the data is balanced, we'll explore attributes which contain the most information.

4 Results

4.1 Cost of Teacher

For the first part of the Cost of Teacher task, the results are plotted in figure 3. As expected, the stacking method results in the lowest RMSE score, combined with the highest R^2 score. Notably, the Extra Trees method is very close to achieving the same results.

For the second part of the Cost of Teacher task, the results are plotted in figure 2. The simplest method would be to simply use the best model from exercise 1, and incrementally learn from dataset **train2**. However, as can be seen from figure 2, even though Stacking achieved the best RMSE score from training **train1**, for the dataset **train2**, Extra Trees actually performs best when adding incrementally more data from **train2**. All five models have relatively high RMSE scores for low amounts of data, but quickly reduces to a score between 6 and 4 after 80 instances. After 80 instances, the RMSE change flattens out, indicating that more data might actually make the model slightly worse, at least not improving it.

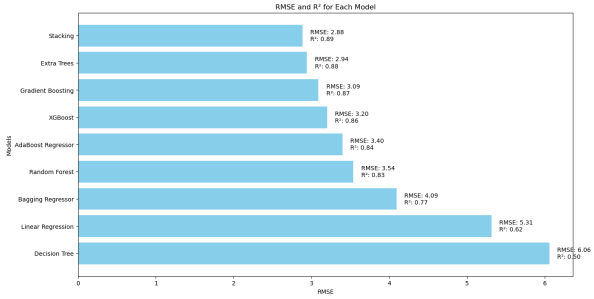

Fig. 1: Unit 1 Exercise 1 Results – RMSE and R^2 scores from model exploration

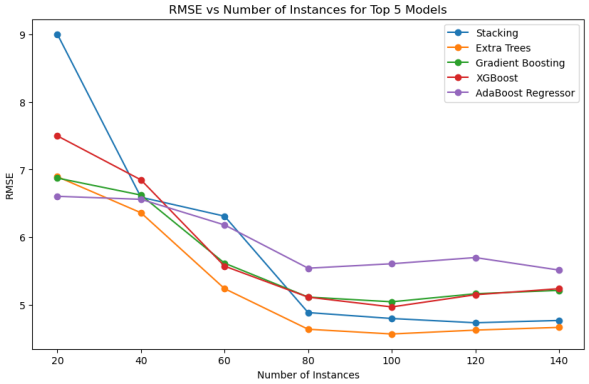

Fig. 2: Unit 1 Exercise 2 Results – RMSE scores from incremental learning of the top five models

4.2 Cost of Test

Our analysis of the Breast Cancer dataset revealed that using the Marg.adhesion attribute alone achieved the global minimum total cost of 422, with test costs accounting for 170 of this total. This finding is significant because adding any additional attributes would increase testing costs beyond this optimal value.

Our incremental model evaluation, which systematically combined attributes sorted by individual performance, demonstrated a clear trade-off between test costs and misclassification costs. The visualization of this relationship shows misclassification costs generally decreasing as more attributes (and consequently higher test costs) are incorporated (though around 5th attribute we see the signs of overfitting).

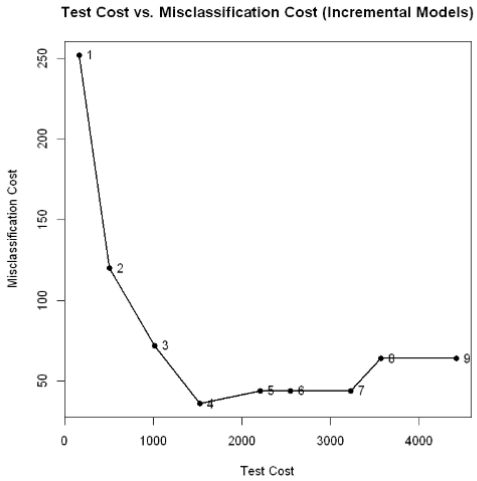

Fig. 3: Test Cost vs Misclassification Costs

5 Conclusions

5.1 Cost of Teacher

Even though the stacking model performed best for the dataset **train1**, it seems that extra trees performs best for the incremental unlabeled adding of instances while training **train2**. However, all models seems to be achieving decent results already after 80 instances added, after which no further significant improvement are made. This would indicate that in the real scenario, if **train2** actually would be totally unlabeled, it would not be necessary to add more than 80 labels of instances. This knowledge of the data and models can potentially save a lot of the cost of teaching, of which is the main concept of this task.

5.2 Cost of Test

In the Cost of Test task, using only the *Marg.adhesion* attribute minimized the total cost, highlighting the importance of attribute selection. Adding more attributes reduced misclassification costs but increased testing costs, showing diminishing returns. In applications like medical diagnostics, focusing on the most informative attributes, such as *Marg.adhesion*, can create more cost-effective models.

6 References

References

1. C. Ferri and C. Monserrat, “T1: Model evaluation,” PowerPoint slides, 2025, lecture presented in "Evaluation, Deployment and Monitoring of Models", UPV - Universitat Politècnica de València.
2. ——, “T3: Reliable models,” PowerPoint slides, 2025, lecture presented in "Evaluation, Deployment and Monitoring of Models", UPV - Universitat Politècnica de València.
3. ——, “T2: Hybrid models,” PowerPoint slides, 2025, lecture presented in "Evaluation, Deployment and Monitoring of Models", UPV - Universitat Politècnica de València.