Interpretable Meta-Active Learning for Regression Ensemble Learning

Ons Saadallah[1], Zied Rouissi[1]

[1]AMZI SMART SOLUTIONS, Tunis, Tunisia

Abstract

Active learning has proven to be an effective approach for reducing the amount of labeled data required in supervised learning tasks, thereby reducing data annotation costs. While ensemble-based active learning schemes have been extensively studied for classification problems, there has been limited research on their applicability to regression tasks. In this paper, a novel active learning method for regression ensemble learning is proposed, which utilizes meta-learning. The meta-learning component is employed to predict continuous utility values for candidate unlabeled data points. The sample selection process is designed to consider both ensemble accuracy and diversity simultaneously. Furthermore, the ensemble model and the meta-learner share the same features, enabling the provision of suitable explanations for selecting specific samples during the active learning procedure, thus enhancing the ensemble performance. Empirical testing of the proposed method is conducted on various real-world regression datasets, evaluating its performance and scalability. The results demonstrate its competitiveness when compared to state-of-the-art approaches in active learning and ensemble learning for regression.

Keywords

Ensemble learning, interpretability, active learning, meta-learning

1. Introduction

In *supervised learning*, data collection and annotation are essential stages. In *passive learning*, training examples are chosen randomly from a distribution and labeled by an oracle. Usually, a significant amount of data points is needed to train a reliable machine learning model. However, data annotation can sometimes be associated with high costs. Henceforth, reducing the amount of labeled data points is necessary. This is broached in the machine learning literature with *Active Learning* (AL) [1]. The primary motivation for active learning (AL) stems from the idea that a model trained with a carefully selected small number of training data points can achieve comparable performance to a model trained on a larger randomly chosen dataset, all while being computationally more efficient and cost-effective [1]. Following this idea, starting from a small and non-optimal training set, AL aims at iteratively selecting unlabelled data points whose inclusion in the training set improves the performance of the machine learning model. The unlabelled data points are evaluated and sorted according to a utility measure that serves as a data selection criterion. The selected data point is labeled by an oracle and then added to

IAL@ECML-PKDD'23: 7th Intl. Worksh. & Tutorial on Interactive Adaptive Learning, Sep. 22nd, 2023, Torino, Italy

*Corresponding author.

saadallahons@gmail.com (O. Saadallah); dr.rouissi@amzi-ss.com (Z. Rouissi)

the training set. The entire procedure is iterated until a stopping criterion is met, e.g., a budget on the total number of points to be added or on annotation costs is consumed.

Compared to classification, active learning (AL) is less commonly used for regression tasks [2, 3], and even less so for regression ensembles [4]. Although several methods designed for classification have been adapted and applied to regression tasks for both ensemble learning [5] and active learning [6, 3], regression presents unique challenges that can result in poorly performing models [7]. One significant distinction between regression and classification is that the range of a model's output in regression is undefined and potentially infinite. This poses limitations on ensemble construction, such as the selection of base models, as many commonly used supervised learning models cannot predict beyond the range of observed labels in the training set, for example, Generalized Additive Models (GAM) [7]. For active learning, this also makes it non-trivial to apply a significant branch of AL approaches based on density estimation, like margin sampling-based strategies [8]. Moreover, in regression tasks, there is no concept of distance, making distance-based sampling approaches not applicable [2]. However, some methods have successfully transferred certain AL techniques originally developed for classification to regression, such as Query-By-Committee (QBC) [9] and Expected Model Change (EMC) [3, 2, 10].

Ensemble learning is widely known as an effective technique in machine learning for both classification [11] and regression [7], as it leverages the strengths of each base model and reduces the effects of overfitting and bias. Works on ensemble learning in the machine learning literature are focused on managing the base models in the different ensemble construction stages, namely base models generation [12, 11], selection or pruning [13], and aggregation or combination [14]. This model management involves, in some cases, training data sampling implicitly [15]. For example, in bagging [12], random sampling is applied to create bootstrap samples are then used for training the base models independently. Hence, bagging involves a blind sampling procedure, i.e., without taking into account data points properties. In boosting [16], a sequential data sampling process is involved by identifying data points with the highest prediction error and adjusting their weights to minimize the training error. Even though boosting performs informed sampling, i.e., by taking into account the prediction *hardness* of training samples, they rely on labeled data to evaluate the training prediction error and are prone to overfitting [17, 18]. Opposingly in this paper, we use AL as an informed sampling to improve the ensemble generalization performance of the ensemble and reduce data annotation costs.

To do so, we introduce METAL a novel, practically useful METa-Active Learning framework for learning regression ensembles. METAL is based on a meta-learning algorithm that learns the utility of a set of candidate unlabelled data points to be added to the training set for learning the ensemble model. Since *diversity* is a fundamental component in ensemble learning [19, 20], we devise the utility measure in such a way it takes into account both ensemble accuracy and diversity. In addition, both ensemble and meta-learner share the same set of features. Therefore, by evaluating the features' importance, we can provide a suitable interpretation for the reason behind selecting a sample by the active learning procedure to improve the ensemble model accuracy iteratively. The main contributions of this paper can be summarized as follows:
Reducing data annotation costs: A novel meta-Active Learning method is proposed for actively selecting unlabelled instances to be used for training an ensemble model for regression;
Informed sampling for regression ensemble learning: An *Informed Sampling* technique is

developed by training a meta-model for learning a utility function to be used for estimating the utility (i.e., informativeness) of an unlabelled data point using a set of carefully-crafted meta-features. The utility is devised to take into account both ensemble *accuracy* and *diversity*; **Interpretability:** We demonstrate that it is possible to provide a suitable interpretation for the reasons behind selecting a sample in the AL iterative process even with a heterogeneous ensemble model (i.e., an ensemble of regression models belonging to distinct families of machine learning models); **Empirical validation :** A comparative empirical study of METAL with state-of-the-art methods for active and ensemble learning for regression and a discussion of its implications in terms of predictive performance and scalability is provided.

We note that all the experiments are fully reproducible, and the code is available under this link[1]. The datasets are publicly available.

2. Literature Review

Opposingly to classification, AL studies gave less attention to regression [1]. However, some of the methods that were originally developed for classification are successfully transferred or adapted to regression, such as Expected Model Change (EMC) [2, 3] and Querry-By-Committee (QBC) paradigm [9]. For instance, Cai et al. [3, 2] showed that EMC outperforms Querry-By-Committee (QBC) on several benchmark regression data sets. However, one major limitation of the proposed EMC is that a large change in the model does not necessarily imply a better performance, as it may only be the result of selecting an outlier. Therefore, Authors in [10] have proposed an improved version of EMC that avoids the selection of outliers by embedding a local outlier probability for both linear and non-linear regression problems. In the same context, Seo et al. [21] relied on Gaussian Processes Regression properties to provide target distributions by estimating posterior mean and variance. These estimates are used for AL by querying data points with high estimated posterior variance. In the QBC paradigm [9], many models are trained to form a committee and predict labels of unlabelled data. Instances with the highest prediction disagreement between the committee members are selected. Many works focused on reformulating the committee disagreement measures to fit regression output by using variance-based measures [6]. Other works have focused on applying active sampling-based techniques to regression. For instance, authors in [22] propose two AL approaches based on greedy sampling. While the first approach is designed to select new samples that increase the diversity in the output space, the second one performs the selection by taking into account diversity in both input and output spaces.

Meta-learning was also applied for AL by learning the utility/informativeness of an unlabelled instance based on a set of characterizing meta-features, generally devised to take into consideration the main learner's performance and the characteristics of the problem. Even though the application of meta-learning for estimating the utility of unlabelled data appears to be very intuitive, only a few works applied it to learn AL sampling strategies and were mainly restricted to classification [23, 24]. Since very few works dealt with AL for regression ensembles, this section is dedicated to both classification and regression problems. When dealing with ensemble learning, QBC is one of the most suited AL methods since it is theoretically well-motivated by

[1] https://www.dropbox.com/sh/9g54gm4xksciaps/AACe8F9zF5id5ysBZYundQWGa?dl=0

the ensemble's error decomposition [9]. The expected ensemble error at a given data point can be decomposed into two main terms, namely, an averaged error term, measuring the average error of the ensemble base models, and an ambiguity term, which is simply the variance of the ensemble around the weighted mean and it measures the disagreement between the base models. The ambiguity term can be entirely estimated from unlabelled data, making thus the application of QBC straightforward. In addition, the decomposition states that if the ensemble is strongly biased, the ambiguity will be small because the base models encompass very similar functions and thus agree on data points outside the training set. Therefore, the ensemble error will be equal to the base models' average error. If, on the other hand, there is a large variance, the ambiguity, in this case, is high, and the ensemble error will be smaller than the average error. As a result, AL can be coupled with ensemble learning where selection is made in favor of unlabelled instances, maximizing the ambiguity term and contributing thus to minimizing the overall ensemble error [9]. Even though the definition of disagreement (i.e., ambiguity) is not restricted to discrete labels, QBC is widely applied to classification. Different adaptions of the disagreement measure are suggested and applied to learning classification ensembles [25, 26, 27, 28].

Active sampling is also applied to learn an ensemble of classifiers. Shan et al. [29] propose an ensemble framework composed of one static and one dynamic classifier built to react to different types of concept drifts in streaming data. The ensemble is combined with uncertainty estimation and random sampling strategies to decide whether to label the upcoming streaming instances for updating both classifiers or not. Some recent works applied meta-learning to learn AL procedures for ensemble methods. In [30], the authors proposed a deep ensemble learning model composed of a selector and a predictor. The selector is designed to actively select key load segments with the most similar patterns to the current training patterns. Taguchi et al. [4] introduced a meta-learning approach that predicts the expected error reduction for a candidate sample. The selection of new instances to be annotated is based on the prediction of the meta-learner, which plays the role of a selector. The predictor is an ensemble model. Feeding the same original features to both predictor and selector, the method is shown to be highly competitive to approaches relying on hand-crafted meta-features for the selector [31]. Most of the aforementioned works on AL are either focused on enhancing the ensemble diversity or accuracy exclusively and are mainly restricted to classification problems. In this work, we adapt meta-learning for AL on regression ensemble by taking into account both ensemble accuracy and diversity. We carefully devise the meta-features by taking into account the properties of the data and the regression task. In addition, we provide interpretations of active instance selection that are not restricted to a specific regression family of models.

3. Methodology

METAL combines both meta-learning and AL to reduce annotation costs and optimize the construction of an ensemble model for regression tasks. To do so, a meta-learner is trained to learn a utility measure that is devised to take into account both ensemble *accuracy* and *diversity*. The selection of unlabelled instances during the AL process is based on the expected values of the utility on the unlabelled set.

3.1. Notations and Problem Formulation

Let the dataset $\mathbb{D}$ be defined as $(x_i, y_i) \in \mathbb{D} \subset \mathbb{R}^n \times \mathbb{R} \, : \, i = \{1, \cdots, N\}$ and generated by an unknown function $f(x) = y$, where n is the number of features of a data point x, and y denotes a numerical response variable. We formulate a regression problem as the task of learning a function $\hat{f_\theta} : x_i \to \mathbb{R}$ such that

$$\hat{f_\theta}(x_i, \theta) \approx f(x_i) = y_i, \forall x_i \in \mathbf{X}, y_i \in \mathbf{Y} \tag{1}$$

where $\theta \in \mathbb{R}^n$ is an unknown (hyper)parameters vector.

Denote with F an ensemble of M of regression models $\hat{f_j}, j = \{1, \cdots, M\}$ Formally, F can be written as the convex weighted combination of the M base models. F on an input data point x is given by:

$$F(x) = \sum_{j=1}^{M} w_j f_j(x) \tag{2}$$

where $w_j, j \in [1, M]$ are the ensemble weights. The weights are constrained to be positive and sum to one. This constraint is necessary for some of the following results. For simplicity, we set the weights to be equal, i.e., $w_j = \frac{1}{M} \forall j \in [1, M]$. Denote with $\mathbb{D}_L$ the labeled data set, i.e., containing annotated data generated by the unknown function f. We split $\mathbb{D}_L$ into $\mathbb{D}_{train}$ that is used to train the models composing the ensemble F and $\mathbb{D}_{meta}$ that will be used to learn the meta-model. The meta-model is denoted by the *selector* in the following. Let $\mathbb{D}_U$ be the unlabelled data set. Our goal is to actively sample data points from $\mathbb{D}_U$ to learn the ensemble model F that best approximates f.

3.2. Combining AL with Ensemble Learning

Ensemble learning is inspired by the principle of committees. In fact, it is assumed that there is no single expert that outperforms all the others on every query. Instead, better overall performance may be obtained by combining the outputs of many experts, i.e., models. In this work, we use AL for annotation costs optimization and as an informed sampling strategy for the ensemble construction process. To do so, M different hypotheses are drawn from the data by means of an active sampling procedure. These hypotheses are used to generate the ensemble M base models. Initially, these models are created by sampling randomly with replacement M subsets $\mathbb{D}_{train,j}, j \in [1, M]$ from $\mathbb{D}_{train}$ with the same size. The M models can create either a homogeneous ensemble if they belong to the same family of regression models or a heterogeneous ensemble if they are selected from different families of models. Afterward, the selector is built and trained using $\mathbb{D}_{meta}$ and used afterward to predict the utility of the unlabelled instances in $\mathbb{D}_U$. The training The instance with the highest predicted utility x^* is selected to be added subsequently to the M subsets created to build the ensemble base models.

$$x^* = argmax_{x \in \mathbb{D}_U} = U(x) \tag{3}$$

where U denotes the utility measure of an unlabeled data point x. However, adding the same instance to these M subsets will increase their similarity after several iterations. The base models are expected to become similar in this case, e.g., as trained on the same data, especially in the case of homogeneous ensembles. This alters the diversity aspect of the ensemble. One solution to mitigate this issue is to consider the top M unlabelled instances with the highest predicted utility values from $\mathbb{D}_U$ to be added to each of the M subsets. This solution is inadequate when a maximum budget of annotations B per iteration is set to be less than the ensemble size (i.e., $B < M$) or in the case of the small dataset and big ensemble size. In the following subsection, we explain how the utility measure and the stopping criterion are devised to account for both accuracy and diversity of the ensemble.

3.3. Utility Criterion

Since the main learner is an ensemble model, the utility measure, which evaluates the utility of unlabelled instances, has to be defined with respect to ensemble properties. Ensemble diversity is considered to be one of the most important aspects of ensemble learning [32]. Even though the enforcement and the evaluation of diversity in regression ensembles are still quite unexplored topics [32, 33], the ensemble error decomposition schema presented by [9] can give some insights about the importance of diversity and individual ensemble base models' performance.

$$E_F = \mathbb{E}\Big[\big(f - F\big)^2\Big] = \sum_{j=1}^{M} w_j E_j - \sum_{j=1}^{M} w_j A_j = \overline{E} - \overline{A} \tag{4}$$

where $\overline{E} = \sum_{j=1}^{M} w_j E_j = \mathbb{E}\Big[w_j\big(f_j - f\big)^2\Big]$, $\overline{A} = \sum_{j=1}^{M} w_j A_j = \mathbb{E}\Big[w_j\big(f_j - F\big)^2\Big]$ and $w_j, j = \{1, \cdots, M\}$ are the ensemble weights. Equation 4 separates the generalization error into two terms. The first one, $\overline{E}$, is an aggregation of the base models' errors. The second term $\overline{A}$, called ambiguity, measures the variability/ disagreement among the base models' outputs and reflects diversity between them. It is straightforward to see that increasing the ambiguity yields a reduction in the overall ensemble error. However, since the overall error is always positive, $\overline{A}$ can be seen as a lower bound of $\overline{E}$. That is why a trade-off between decreasing $\overline{E}$ and increasing $\overline{A}$ should be established. Denote with x_i a given unlabelled data point and with $F_{\cup x_i}$ the ensemble composed of models trained on M subsets $\mathbb{D}_{train,j}, j = \{1, \cdots, M\}$ including the addition of a given instance x_i. The gain of x_i evaluated on $\mathbb{D}$ is giving by:

$$gain(x_i, \mathbb{D}) = E_F^{\mathbb{D}} - E_{F_{\cup x_i}}^{\mathbb{D}} \tag{5}$$

where $\mathbb{D}$ states for the dataset on which the ensemble F error is evaluated. x^* that maximizes the gain is simply selected.

$$x^* = \operatorname*{argmax}_{x_i \in \mathbb{D}_U} gain(x_i, \mathbb{D}) \tag{6}$$

However, to increase the ambiguity, instead of adding x^* to all models, i.e., to the training sets of all the models, we add it to the model f^* yielding the highest deviation from F on x^*.

$$f^* = \operatorname*{argmax}_{f_j, j \in \{1,\cdots,M\}} A_j(x^*) = \operatorname*{argmax}_{f_j, j \in \{1,\cdots,M\}} \big(f_j(x^*) - F(x^*)\big)^2 \tag{7}$$

```
Require: Training dataset $\mathbb{D}_{train}$
Split $\mathbb{D}_{train}$ into $\mathbb{D}_{F,train}$, $\mathbb{D}_{meta,train}$ and $\mathbb{D}_{meta,eval}$
Use $\mathbb{D}_{F,train}$ to generate $\mathbb{D}_{F,train,j}, j = \{1, \cdots, M\}$ to train the base models
Build the ensemble $F$
for Each $x_j \in \mathbb{D}_{meta,train}$ do
    Calculate $gain(x_j, \mathbb{D}_{meta,eval})$
end for
$G = \{gain(x_j, \mathbb{D}_{meta,eval}), j = 1, \cdots, |\mathbb{D}_{meta,train}|\}$
Train $MetaM$ on $\{X_{\mathbb{D}_{meta,train}}, G\}$
Return $MetaM$
```

Algorithm 1: Learning *MetaM*

3.4. METAL Framework

METAL is composed of three main stages. In the first stage, a meta-model *MetaM* is trained after preparing a meta-dataset to learn the gain induced by the addition of a given data instance to the ensemble training set and predict the gain of unlabelled data instances in $\mathbb{D}_U$. The second stage consists of selecting the data instance from $\mathbb{D}_U$ with maximal predicted gain and determining to which ensemble member should be added (See Eq.7). In the third stage, both training and unlabelled datasets are updated by adding the selected instance to the training subsets of the selected ensemble member and removing it from $\mathbb{D}_U$. The three stages are iterated until a stopping criterion is met (i.e., a maximum number of iterations in this work).

MetaM is used to learn the gain measure defined in Eq.5. Therefore, we split the original training set into $\mathbb{D}_{train}$ into three disjoint subsets $\mathbb{D}_{F,train}$, $\mathbb{D}_{meta,train}$ and $\mathbb{D}_{meta,eval}$. $\mathbb{D}_{F,train}$ is used to create the subsets $\mathbb{D}_{F,train,j}, j = \{1, \cdots, M\}$ to train the M ensemble members as explained in subsection 3.2. $\mathbb{D}_{meta,train}$ is the training dataset for the meta model *MetaM*. For each data instance, $x_j \in \mathbb{D}_{meta,train}$, the gain achieved by adding x_j to the training subsets $\mathbb{D}_{F,train,j}, j = \{1, \cdots, M\}$ is evaluated using evaluation dataset $\mathbb{D}_{meta,eval}$. The set of gains, $G = \{gain(x_j, \mathbb{D}_{meta,eval}), j = 1, \cdots, |\mathbb{D}_{meta,train}|\}$ calculated for the data in $\mathbb{D}_{meta,train}$ is regarded as the response variables corresponding to the explanatory variables $X_{\mathbb{D}_{meta,train}}$ in $\mathbb{D}_{meta,train}$. The pairs $\{(x_j, gain(x_j, \mathbb{D}_{meta,eval})), j = 1, \cdots, |\mathbb{D}_{meta,train}|\}$ are used for training *MetaM*. This meta-learning stage consists of learning the active learning process and is summarized in Algorithm 1.

MetaM is used afterward to predict the gain of data instances in $\mathbb{D}_U$. The sample x^* maximizing the gain is selected and attributed to the training set of the ensemble member verifying the condition in Eq.7.

3.5. Interpretability of sample selection

Our framework is composed of two models, a meta-model for predicting the gain of unlabelled samples and a main model for predicting the response variable. Both models share the same set of features. The gain predicted by the meta-model is related to the loss of the main model (i.e., ensemble). Therefore, the selected sample by the meta-model is also associated with the importance of the features of the sample to the main model. In this context, we use machine learning algorithms equipped with a measure of feature importance for the meta-model *MetaM* and the ensemble base models. The overall importance of each feature for the ensemble is assessed by computing the average of the corresponding normalized importance measures over all the base models. We monitor the change in the features' importance and in their empirical distributions over the active learning iterations in order to explain the rationale behind sample selection in the active learning process. Concrete examples are provided in Section 4.2.

4. Experiments

In this section, we present the experiments carried out to validate METAL and to answer the following research questions: **Q1:** How is the performance of METAL compared to state-of-the-art (SoA) methods for active learning for regression problems? **Q 2:** What is the impact of relying only on the averaged error or the ambiguity in evaluating the gain of an unlabelled data instance? **Q3:** Since active learning can be viewed as an *informed sampling* strategy for ensemble construction, how is the performance of METAL compared to the SoA ensemble methods for regression? **Q4:** How is the scalability of METAL in terms of computational resources compared to SoA methods for active learning? **Q5:** How the meta-model and the ensemble can be used together to provide a suitable interpretation for selecting a sample in the process of active learning to improve the ensemble performance?

4.1. Experimental Setup

The methods are evaluated using the root mean squared error (RMSE). In each experiment, the data is split into 30% for training, 20% for testing, and 50% is assumed to be an unlabelled dataset. For each dataset, a cross-validation (CV) with 10 folds is performed for the evaluation with 100 repetitions for the active learning methods (i.e., different initialization of the labeled set). We used 15 benchmarking datasets for our experiments. They are briefly summarized in Table 1. We note that all the experiments are fully reproducible, and the code is available under this link[2]. The datasets are publicly available.

ID	Dataset	Data Source		Data	characteristics
			Att.	Ins.	RT
1	2Dplanes	dcc.fc.up.pt	10	40768	[-999.709,999.961]
2	Friedman Domain	dcc.fc.up.pt	10	40768	[1.50,27.975]
3	Abalone	UCI	8	4177	[1.00,29.00]
4	Auto-Price	UCI	14	159	[9,46.60]
5	Bank8FM	DELVE	8	8192	[0,0.70]
6	House (8H)	DELVE	8	22784	[0,427300]
7	Bos.Housing	UCI	13	506	[5,50]
8	Cal.Housing	StatLib	8	20460	[14999,500001]
9	Kinematics	DELVE	8	8192	[0.04017,1.45852]
10	Pole Telecom	dcc.fc.up.pt	48	15000	[0.00,100.00]
11	puma8NH	DELVE	8	8192	[-0.085173,0.088266]
12	Puma32H	DELVE	32	8192	[-0.085173,0.088266]
13	Stock Prices	StatLib	10	950	[34,60.5]
14	Triazines	UCI	60	186	[0.10,0.90]
15	Red wine quality	UCI	12	1599	[3,8]

Table 1
List of Datasets used for the experiments.

[2] https://www.dropbox.com/sh/9g54gm4xksciaps/AACe8F9zF5id5ysBZYundQWGa?dl=0

4.1.1. METAL Set-up

We build a homogeneous ensemble of 10 Decision Trees (DTs) [34] generated with a bootstrap sampling process over the train labelled data.

As 30% of the total data size is kept for training data (i.e., $\mathbb{D}_{train}$), this proportion is split into 50% for $\mathbb{D}_{F,train}$, 30% for $\mathbb{D}_{meta,train}$ and 20% $\mathbb{D}_{meta,eval}$. The 10 DTs are initially trained using different random bootstraps of $\mathbb{D}_{train}$, (i.e., $\mathbb{D}_{F,train,j}, j = \{1, \cdots, 9\}$). The maximum number of iterations in the active learning process is set up to half of the unlabelled set size. The meta-learner is chosen to be a Random Forest (RF) [12].

4.1.2. S.o.A active and ensemble learning methods Set-up

We compare the performance of METAL against the following approaches for active and ensemble learning for regression.

Active Learning Methods

QBC [6]: Query-By-Committee: We adopt the variance reduction approach for QBC. A committee composed of the same base models as our ensemble in METAL , and the unlabelled sample with the maximum disagreement (i.e., variance) between the committee members is selected for annotation. The detailed procedure is explained in [6]. The prediction of the response variable is also generated by an ensemble of committee members.

EMC [3]: Expect model change: METAL is compared to EMC for the Gradient Boosting DT (GBDT) model as introduced in [2]. We use 10 decision trees for calculating the model change as a bigger number of trees is required to improve the accuracy of the method.

Ran: One sample is randomly selected for annotation from the unlabelled dataset.

GPF [4]: Gain Prediction Function is a meta-learning approach for active learning for regression that relies on using a meta-learner for predicting the gain formulated as only the amount of decrease in the prediction error ($gain(x_i, \mathbb{D}) = E_{\mathbb{D}}(F) - E_{\mathbb{D}}(F_{\cup x_i})$). The original work uses a single model as a main learner. To ensure a fair comparison with METAL . The main learner is an ensemble of the same base models as METAL ; the meta-learner is an RF.

M-Am: is a variant of METAL where only the ambiguity is kept in the definition of the gain: $gain(x_I, \mathbb{D}) = \overline{A}_{\mathbb{D}}(F_{\cup x_I}) - \overline{A}_{\mathbb{D}}(F)$. This variant is different from QBC since it relies on the meta-learner's prediction of ambiguity (i.e., variance) reduction instead of a direct computation.

M-Er: is a variant of METAL where only the error term is kept in the gain: $gain(x_I, \mathbb{D}) = \overline{E}_{\mathbb{D}}(F) - \overline{E}_{\mathbb{D}}(F_{\cup x_I})$.

Ensemble Methods

Active learning can also be viewed as an*informed sampling* strategy for optimizing ensemble construction by selecting samples that lead to enhanced **accuracy** and **diversity**. Therefore, we suggest comparing the ensemble built by METAL with ensemble methods using the whole available training set (i.e., initial labeled set and the assumed unlabelled set) (i.e., *passive/ blind sampling*). The hyperparameters values of the involved models are tuned with a random search over a 3-fold CV. We compare our method against the following:

RF [12]: Random Forest uses bagging to create an ensemble of regression trees.

GBM [35]: Gradient boosting machine that uses boosting to create an ensemble of regression trees.

ENS [34]: An ensemble that averages the same base models as the ensemble of METAL.

Stacking [36]: An ensemble that uses linear stacking to combine METAL base models instead of using simple average.

MetaBags [7]: A recent approach for learning heterogeneous ensemble for regression by using bagging on the meta-level to select and aggregate ensemble base models.

4.2. Results and Discussion

Table 2 shows the performance results of METAL against SoA methods for active and ensemble learning. Results are reported in terms of RMSE. The statistical significance of the results is assessed using the Bayesian correlated *t*-test with the significance level $\alpha = 0.05$, with the null hypothesis that a given learner wins against METAL after observing the results of all repetitions.

The results presented in Table 2 show that METAL outperforms existing SoA methods for both active and ensemble learning. In addition,METAL is almost never statistically significantly worse than any active learning method. It is also highly competitive with ensemble methods trained on a larger amount of data which shows the efficiency of active learning as an *informed sampling* strategy for building an ensemble, supporting thus the main active learning assumption stating that a better-performing machine learning model can be built using less amount of training data points carefully selected. This can be explained by orienting the selection towards establishing a trade-off between **accuracy** and **diversity**. METAL has the lowest average rank. These results illustrate the generalization power of METAL in both cases. Comparing METAL to its different variants (i.e., M-Er and M-Am), we see the clear advantage of integrating both error and ambiguity components in the definition of the gain. The results are worse with M-Er, meaning that ambiguity is needed to enforce a certain degree of diversity in the ensemble. However, it is not also sufficient on its own. This answers the research question **Q1-Q3**.

We present in Figure 1 a comparison of the averaged runtime of METAL compared to the remaining active learning methods. In METAL and GPF, the size of sub-dataset $\mathbb{D}_{train}$ for calculating the gain is fixed, and the cost for model construction is constant (i.e., bootstrapping only in the initialization). Let m be the cost for model construction in the gain calculation, c be the cost for predicting one sample in the unlabelled dataset, and n its size. The total cost is of order $O(nc + m)$. For QBC, we construct the committee trained with resampled datasets and predict the output for each sample in the unlabelled dataset. Then, the sample with the highest variance is selected for annotation. The computational cost for selecting a new sample is of order $O(b(nc + m))$, where b is the number of bootstrap sampling. In EMC, in addition to the

ID	QBC	EMC	Ran	GPF	M-Am	M-Er	METAL
1	1.95(0.15)	2.65(0.12)	2.25(0.23)	2.75(0.85)	2.15(0.45)	1.85(0.12)	**1.75(0.10)**
2	3.42(0.19)	3.43(0.19)	3.55(0.15)	3.65(0.30)	3.35(0.18)	3.54(0.15)	**2.65(0.13)**
3	3.05(0.28)	3.35(0.28)	3.15(0.25)	3.15(0.35)	2.98(0.23)	3.00(0.23)	**2.89(0.25)**
4	3.85(0.55)	4.74(0.44)	4.35(0.51)	3.96(0.46)	3.68(0.52)	4.05(0.75)	**3.39(0.61)**
5	60e-2(6e-3)	71e-2(3e-3)	71e-2(3e-3)	64e-2(7e-3)	68e-2(8e-3)	66e-2(8e-3)	**57e-2(6e-3)**
6	45e3(39e2)	43e3(40e2)	43e3(32e2)	46e3(29e2)	43e3(28e2)	44e3(28e2)	**42e3(12e2)**
7	6.75(1.85)	6.75(3.20)	7.34(2.16)	7.35(2.65)	6.05(1.10)	5.63(1.55)	**5.45(1.38)**
8	86e3(75e2)	88e3(34e2)	87e3(51e2)	94e3(60e2)	85e3(57e2)	86e3(58e2)	**76e3(25e2)**
9	0.23(7e-3)	0.23(7e-3)	0.23(8e-3)	0.23(9e-3)	0.23(2e-3)	0.23(6e-3)	**0.21(8e-3)**
10	**19.3(2.96)**	26.4(3.01)	30.8(6.00)	28.7(6.95)	25.4(4.86)	25.33(5.10)	23.94(6.43)
11	3.92(0.05)	4.66(0.07)	4.45(0.15)	4.05(0.34)	3.95(0.29)	3.98(0.23)	**3.65(0.18)**
12	22e-2(2e-3)	24e-2(2e-3)	26e-2(2e-3)	25e-2(3e-3)	24e-2(1.5e-3)	24e-2(2.2e-3)	24e-2(2e-3)
13	2.12(0.22)	3.14(0.29)	2.97(1.18)	3.01(1.40)	2.76(0.72)	2.33(0.16)	**1.85(0.17)**
14	0.14(1e-2)	0.14(1e-2)	0.14(1.4e-2)	0.15(1.4e-2)	0.14(2e-2)	0.14(1.8e-2)	**0.12(1.5e-2)**
15	0.75(1e-2)	0.71(4e-2)	0.73(3e-2)	0.76(4e-2)	0.74(3e-2)	0.74(3e-2)	**0.52(3e-2)**
∅ Rank	2.68(0.59)	5.37(1.49)	6.00(0.72)	4.31 (1.15)	3.28(1.30)	4.00(1.29)	**1.21(1.65)**
Loss/Win	7/1	10/0	10/0	9/0	8/2	9/1	*N/A*

ID	MetaBags	RF	ENS	GBM	Stack.	METAL
1	1.94(0.15)	3.35(0.16)	1.78(0.13)	2.07(0.15)	1.95(0.18)	**1.75(0.10)**
2	3.02(0.22)	4.14(0.21)	3.27(0.24)	2.96(0.15)	2.85(0.11)	**2.65(0.13)**
3	3.01(0.23)	3.55(0.46)	3.17(0.38)	3.06(0.46)	3.02(0.25)	**2.89(0.25)**
4	3.42(0.60)	5.65(1.11)	3.65(0.61)	3.55(0.55)	3.79(1.19)	**3.39(0.61)**
5	68e-3(1e-3)	99e-2(1e-3)	67e-2(1e-3)	48e-2(1e-3)	**40e-2(1e-3)**	57e-2(6e-3)
6	**38e3(49e2)**	45e3(37e2)	42e3(37e2)	41e3(43e2)	42e3(26e2)	42e3(12e2)
7	6.34(1.71)	7.55(3.24)	5.87(1.43)	8.00(2.20)	5.98(0.92)	**5.45(1.38)**
8	84e3(30e2)	96e3(46e2)	83e3(20e2)	88e3(36e2)	78e3(39e2)	**76e3(25e2)**
9	0.23(1.4e-2)	0.25(1.3e-2)	0.23(8e-3)	**0.21(1.4e-2)**	**0.21(1e-2)**	0.**21(8e-3)**
10	**21.7(1.85)**	39.1(2.02)	23.66(4.18)	35.90(1.6)	27.54(4.5)	23.94(6.43)
11	4.35(0.16)	4.97(0.15)	3.97(0.24)	4.23(0.15)	4.00(0.20)	**3.65(0.18)**
12	25e-2(5e-4)	26e-2(9e-4)	**24e-2(1e-2)**	28e-2(8e-4)	35e-2(2e-3)	**24e-2(2e-2)**
13	1.9(0.15)	4.55(0.33)	2.37(0.30)	2.44(0.24)	1.98(0.15)	**1.85(0.17)**
14	0.13(2e-2)	0.14(9.4e-3)	0.16(0.30)	0.15(0.24)	0.23(4.1e-2)	**0.12(1.5e-2)**
15	0.69(4e-2)	0.73(5e-2)	0.69(3e-2)	0.72(4e-2)	0.74(3e-2)	**0.52(3e-2)**
∅ Rank	3.70(1.64)	3.11(1.25)	4.00(1.62)	4.29(1.68)	4.05(1.88)	**1.65(1.41)**
Loss/Win	8/2	8/0	9/0	9/3	9/2	*N/A*

Table 2

Detailed predictive performance results, comparing SoA active learning methods vs. METAL -including variations of METAL (top) and SoA ensemble methods vs. METAL (bottom). The results report on the average and (std. dev) of RMSE. The last rows depict the wins and losses based on the Bayesian correlated t−test with the significance level $\alpha = 0.05$ and the null hypothesis that a given method wins against METAL.

prediction model, b bootstrap sample is created for training b GBDT models, and therefore the computational cost for EMC is of order $O(b(nc + m))$.

METAL and GPF, which perform model construction only a fixed number of times in each iteration, provide a clear computational advantage compared to QBC and EMC. We notice that despite their same computational order, EMC is slower than QBC. This can be explained by the

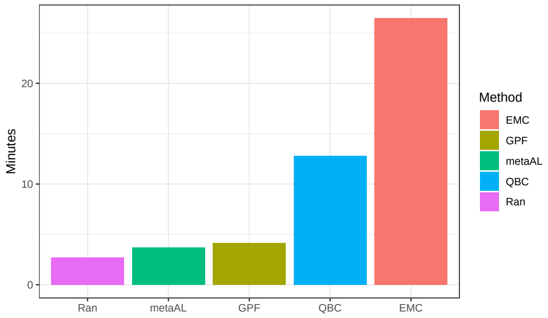


Figure 1: Averaged computational costs in minutes of active learning methods over the 15 data sets

AGE	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Value	15.80	82.10	91.90	85.84	96.85	98.90

LSTAT	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Value	1.98	12.15	14.59	14.25	17.02	27.26

DIS	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Value	1.69	2.047	2.327	2.471	2.696	6.062

Table 3
Summary of AGE, LSTAT, and DIS before selecting the 5th sample.

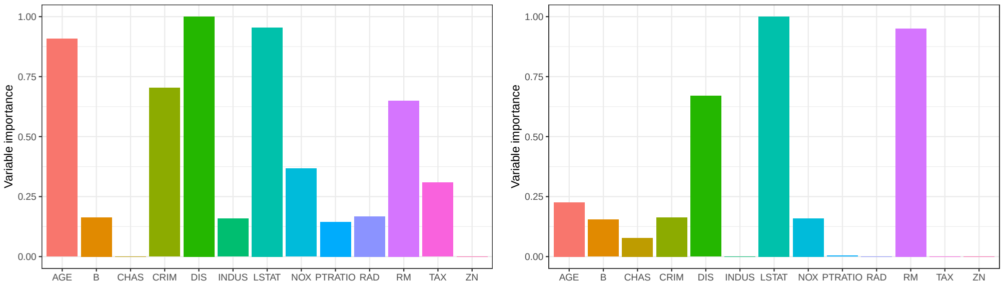
manipulation of DTs in GBDT models that are required to evaluate the model change. This answers the question **Q4**.

Regarding question **Q5**, we show an example of an interpretation of sample selection for the *Bos.Housing* dataset at a given iteration. In this experiment, the variable MEDV (i.e., the median value of owner-occupied homes from land information) is the response variable and is predicted by our ensemble using 13 other explanatory variables, as shown in Table 4. Since **MetaM** is an RF, it is possible to compute variables' importance. The main model is an ensemble of regression models, and variable importance can be obtained following the procedure detailed in Section 3.5. Figure 2a shows the importance of the normalized variables by the ensemble before the 5-*th* sample selection. It can be seen that LSTAT, DIS, and AGE have the highest importance at this stage.

Feature	CRIM	ZN	INDUS	CHAS	NOX	RM	**AGE**
Value	0.013	90	1.22	0.0	0.403	7.249	**21.9**
Feature	**DIS**	RAD	TAX	PTRATIO	B	**LSTAT**	
Value	**8.696**	5	226	17.9	395.93	**4.81**	

Table 4
Selected data sample at the 5-th iteration.

LSTAT shows the percentage of low-income inhabitants, and it is natural to assume that income is strongly related to the prices of houses. DIS presents the distances to the city center, and it naturally affects house prices. The AGE of the building also is highly correlated with its price. *MetaM* predicts the reduction in the ensemble error, which is reflected by the reduction of the averaged error and the increase in ambiguity once a data sample is added to the training set. We can see that LSTAT and DIS have higher importance than the other variables in learning the gain and lower importance for the AGE. Since the gain reflects a decrease in the ensemble



(a)

(b)

Figure 2: Variable importance for the ensemble (a) and *MetaM* (RF) (b) before selecting the 5-th sample.

loss after a data sample is added to the training set, the variable with high importance in the ensemble also has high importance in *MetaM* as naturally expected. In addition, the actual selected sample is shown in Table 4. It can be seen that the values of LSTAT are 4.81, DIS 8.696, and AGE 21.9. The summary of these variables in the training data (i.e., before the data sample selection) is shown in Table 3. From this table, it can be seen that the sample is selected from low-density regions, especially for AGE and DIS. Based on this result, we can conjecture that at this stage of the learning, the information on the AGE and distance DIS of buildings together with the percentage of the population engaged in low-salary occupations LSTAT is important for the ensemble accuracy, but a sufficient amount of information is not yet collected. That is why active sampling is made in favor of low-density regions.

5. Concluding Remarks

This paper introduces METAL a novel, practically useful meta-active learning method for learning regression ensembles. This work illustrates the combination of meta and active learning for optimizing ensemble building and enhancing its performance by contributing to the accuracy-diversity trade-off. The proposed method shows that it is possible to interpret the reason for sample selection without forcing restrictions on the ensemble construction. In future work, we aim to explore heterogeneous ensembles and how different families of machine learning models can be combined to improve prediction accuracy and support explainability.

Acknowledgments

This work is funded by AMZI SMART SOLUTIONS.

References

[1] B. Settles, Active learning literature survey, Technical Report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

[2] W. Cai, M. Zhang, Y. Zhang, Batch mode active learning for regression with expected model change, IEEE Transactions on Neural Networks and Learning Systems 28 (2017) 1668–1681. doi:10.1109/TNNLS.2016.2542184.

[3] W. Cai, Y. Zhang, J. Zhou, Maximizing expected model change for active learning in regression, in: 2013 IEEE 13th International Conference on Data Mining, IEEE, 2013, pp. 51–60.

[4] Y. Taguchi, K. Kameyama, H. Hino, Active learning with interpretable predictor, in: 2019 International Joint Conference on Neural Networks (IJCNN), 2019, pp. 1–8. doi:10.1109/IJCNN.2019.8852041.

[5] J. Mendes-Moreira, C. Soares, A. M. Jorge, J. F. D. Sousa, Ensemble approaches for regression: A survey, Acm computing surveys (csur) 45 (2012) 10.

[6] R. Burbidge, J. J. Rowland, R. D. King, Active learning for regression based on query by committee, in: International Conference on Intelligent Data Engineering and Automated Learning, Springer, 2007, pp. 209–218.

[7] J. Khiari, L. Moreira-Matias, A. Shaker, B. Ženko, S. Džeroski, Metabags: Bagged meta-decision trees for regression, in: Joint european conference on machine learning and knowledge discovery in databases, Springer, 2018, pp. 637–652.

[8] B. Settles, M. Craven, S. Ray, Multiple-instance active learning, in: Advances in neural information processing systems, 2008, pp. 1289–1296.

[9] A. Krogh, J. Vedelsby, Neural network ensembles, cross validation, and active learning, in: Advances in neural information processing systems, 1995, pp. 231–238.

[10] S. H. Park, S. B. Kim, Robust expected model change for active learning in regression, Applied Intelligence (2019) 1–18.

[11] L. Breiman, Random forests, Machine learning 45 (2001) 5–32.

[12] L. Breiman, Bagging predictors, Machine learning 24 (1996) 123–140.

[13] G. Tsoumakas, I. Partalas, I. Vlahavas, An ensemble pruning primer, Applications of supervised and unsupervised ensemble methods (2009) 1–13.

[14] A. Saadallah, K. Morik, Online ensemble aggregation using deep reinforcement learning for time series forecasting, in: 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2021, pp. 1–8.

[15] Z.-H. Zhou, Z.-H. Zhou, Ensemble learning, Springer, 2021.

[16] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, Journal of computer and system sciences 55 (1997) 119–139.

[17] R. Maclin, D. Opitz, An empirical evaluation of bagging and boosting, AAAI/IAAI 1997 (1997) 546–551.

[18] J. R. Quinlan, et al., Bagging, boosting, and c4. 5, in: Aaai/Iaai, vol. 1, 1996, pp. 725–730.

[19] G. Brown, J. L. Wyatt, P. Tiňo, Managing diversity in regression ensembles, Journal of machine learning research 6 (2005) 1621–1650.

[20] D. Wood, T. Mu, A. Webb, H. Reeve, M. Lujan, G. Brown, A unified theory of diversity in ensemble learning, arXiv preprint arXiv:2301.03962 (2023).

[21] S. Seo, M. Wallat, T. Graepel, K. Obermayer, Gaussian process regression: Active data selection and test point rejection, in: Mustererkennung 2000, Springer, 2000, pp. 27–34.

[22] D. Wu, C.-T. Lin, J. Huang, Active learning for regression using greedy sampling, Information Sciences 474 (2019) 90 – 105. URL: http://www.sciencedirect.com/science/article/pii/

S0020025518307680. doi:https://doi.org/10.1016/j.ins.2018.09.060.

[23] K. Konyushkova, R. Sznitman, P. Fua, Learning active learning from data, in: Advances in Neural Information Processing Systems, 2017, pp. 4225–4235.

[24] W.-N. Hsu, H.-T. Lin, Active learning by learning, in: Twenty-Ninth AAAI conference on artificial intelligence, 2015.

[25] X. Zhu, P. Zhang, X. Lin, Y. Shi, Active learning from stream data using optimal weight classifier ensemble, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 40 (2010) 1607–1621.

[26] C. Körner, S. Wrobel, Multi-class ensemble-based active learning, in: European conference on machine learning, Springer, 2006, pp. 687–694.

[27] P. Melville, R. J. Mooney, Diverse ensembles for active learning, in: Proceedings of the twenty-first international conference on Machine learning, ACM, 2004, p. 74.

[28] B. Krawczyk, Active and adaptive ensemble learning for online activity recognition from data streams, Knowledge-Based Systems 138 (2017) 69–78.

[29] J.-C. Shan, W.-K. Liu, C.-X. Chu, C.-F. Dai, Q.-B. Liu, Online active learning with drifted data streams using paired ensemble framework, in: ITM Web of Conferences, volume 12, EDP Sciences, 2017, p. 05016.

[30] Z. Wang, B. Zhao, H. Guo, L. Tang, Y. Peng, Deep ensemble learning model for short-term load forecasting within active learning framework, Energies 12 (2019) 3809.

[31] K. Konyushkova, R. Sznitman, P. Fua, Learning active learning from data, arXiv preprint arXiv:1703.03365 (2017).

[32] A. Saadallah, F. Priebe, K. Morik, A drift-based dynamic ensemble members selection using clustering for time series forecasting (2019).

[33] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, M. Woźniak, Ensemble learning for data stream analysis: A survey, Information Fusion 37 (2017) 132–156.

[34] R. T. Clemen, R. L. Winkler, Combining economic forecasts, Journal of Business & Economic Statistics 4 (1986) 39–46.

[35] J. H. Friedman, Greedy function approximation: a gradient boosting machine, Annals of statistics (2001) 1189–1232.

[36] D. H. Wolpert, Stacked generalization, Neural networks 5 (1992) 241–259.