REGRESSION WITH AN ENSEMBLE OF NOISY BASE FUNCTIONS

Yuval Ben-Hur, Yuval Cassuto, and Israel Cohen

Viterbi Department of Electrical and Computer Engineering,
Technion – Israel Institute of Technology, Haifa 3200003, Israel

ABSTRACT

Ensemble methods achieve state-of-the-art performance in many real-world regression problems while enjoying structural compatibility for modern decentralized computing architectures. However, the implementation of ensemble regression on distributed systems may compromise its cutting-edge performance due to computing and communication reliability issues. This paper introduces robust ensemble combining techniques designed to integrate multiple noisy predictions into a single reliable prediction. Experiments conducted with synthetic and real-world datasets in various noise regimes illustrate our robust methods' superiority over non-robust counterparts.

Index Terms— Ensemble learning, distributed regression, inference noise.

1. INTRODUCTION

Learning algorithms consume unprecedented amounts of computing and storage resources [1]. The pursuit for better model performance, realized by ever-growing hardware demands, fuels the race toward denser and more efficient hardware devices. However, hardware scaling rates cannot meet the exponentially growing requirements of state-of-the-art learning algorithms [1]. This situation has ignited a renewed interest in *ensemble* learning algorithms. Such algorithms allow distributing computation and storage loads between multiple sub-systems while maintaining high-end performance [2]. Moreover, ensemble learning algorithms exhibit natural error tolerance [3], which is a desired property for reliable operation over distributed architectures that are typically faulty.

Recent attempts to implement learning ensembles on efficient hardware devices have encountered serious reliability issues [4, 5, 6]. In [4], for example, an ensemble of convolutional neural networks (CNN), used for image classification and implemented on FPGA accelerators, suffered from performance degradation due to errors induced by radiated noise. In [5] and [6], neural network ensembles were deployed on wireless sensor networks for localization and navigation purposes. The network comprised low-precision edge devices that produced unreliable individual predictions, resulting in degraded overall prediction performance. In all those cases, it was shown that the performance of ensemble learners is severely compromised by various fault mechanisms. Yet, only ad-hoc solutions were proposed: from increasing the ensemble size [6], to heuristically re-weighting [5], and even completely ignoring [4], specific ensemble predictions. Therefore, it is clear that the robustness of the ensemble to certain implementation faults is a key issue that needs addressing toward implementation on next-generation devices.

Extending a recent work on robust ensemble *classification* [7], we focus in this work on ensemble *regression*. Regression is the task of fitting a functional model to the relationship between a dependent variable and one or more independent variables. A regression model is usually inferred from realizations of independent variables and the corresponding, possibly inaccurate, values of a dependent variable. Ensemble regression is a powerful and elegant technique for constructing regression models that play a key role in a wide range of real-world problems and applications [8]: from big-data analysis (e.g., time-series forecasting [9], outlier detection [10]) to estimation problems (e.g., source localization [5], image alignment [11]) and beyond.

The robustness of regression models (and specifically ensemble regressors) was previously addressed mainly for adverse *training* data [12] (e.g., model mixture contamination [13], or outlying samples [14]). Over the years, extensive research efforts were invested toward devising robust ensemble training procedures, such as (variants of) Bagging [15], Stacking [16] and Boosting [17]. Additionally, various methods for *combining* the individual predictions of ensemble members were proposed and analyzed in terms of generalization performance and robustness. The most notable and prevalent techniques are the basic/generalized ensemble methods (BEM/GEM) [18] and the linear regressor (LR) [19, 20]. Yet, robust *training* of the model does not necessarily guarantee good performance when the *prediction* process is unreliable.

This paper introduces novel ensemble combining techniques for robust regression, which are optimal in terms of

This work was supported in part by the Israel Science Foundation.

978-1-6654-8547-0/22//$31.00 ©2022 IEEE

their mean-squared error (MSE). First, we define an MSE criterion that incorporates both generalization error and prediction noise, thereby capturing the *expected* performance of noisy ensemble regression. Then, robust versions of the basic ensemble method (BEM) and the generalized ensemble method (GEM) are developed. Within the class of normalized linear combining methods, we prove that robust GEM is the optimal combining approach and that robust BEM is the optimal *data-independent* combining approach in the sense of *expected MSE*. We also develop a robust version of the linear regressor (LR), which optimizes the *empricial MSE*, thereby discarding the need in generalization error statistics (as required, e.g., by GEM). We compare our suggested robust methods (BEM, GEM, and LR) with their non-robust counterparts using three synthetic and three real-world datasets. Our robust methods achieve significantly lower MSE over a wide range of signal-to-noise ratios (SNRs) for all examined datasets. For low SNR, the test error achieved by robust methods is up to 12 dB lower than that obtained by non-robust methods, depending on the specific method and dataset.

The rest of the paper is organized as follows. In Section 2, we define the noisy ensemble regression problem and review prior work on ensemble combining techniques. Based on this framework, we design in Section 3 the robust regression ensemble combining methods. Section 4 includes simulation results of synthetic and real-world datasets. Finally, the paper is concluded in Section 5.

2. MODEL FORMULATION

Consider an unknown (deterministic) target function $f(\cdot) : \mathcal{X}^d \to \mathbb{R}$ that is to be estimated given a set of data samples $\mathcal{S} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_s}$, where $\boldsymbol{x}_i \in \mathcal{X}^d, d \in \mathbb{N}$ and $y_i \in \mathbb{R}$. The values y_i are assumed to obey a probability rule, which represents additive measurement error or training noise (i.e., $y_i = f(\boldsymbol{x}) + \epsilon$ where usually $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$). A regressor (or a predictor) $\hat{f}(\cdot) : \mathcal{X}^d \to \mathbb{R}$ is an estimator of $f(\cdot)$ that is generated based on $\mathcal{S}$. Focusing on ensemble regression, the predictor $\hat{f}(\cdot)$ is formulated by fusing the predictions of multiple base-functions $\{\hat{f}_t(\cdot)\}_{t=1}^T$.

Definition 1 (regression ensemble)**.** *Define a regression ensemble as the set of functions $\{\hat{f}_t(\cdot)\}_{t=1}^T$ where $\hat{f}_t : \mathcal{X}^d \to \mathbb{R}$ and $T \in \mathbb{N}$.*

Due to their effectiveness and prevalence, we limit the discussion to linear ensemble integration methods, in which

$$\hat{f}(\boldsymbol{x}) = \boldsymbol{\alpha}^\top \boldsymbol{\varphi}(\boldsymbol{x}), \tag{1}$$

where $\boldsymbol{\varphi}(\boldsymbol{x}) = \left(\hat{f}_1(\boldsymbol{x}), \ldots, \hat{f}_T(\boldsymbol{x})\right)^\top$ and $\boldsymbol{\alpha} \in \mathbb{R}^T$. The simplest normalized linear combining technique is the basic ensemble method (BEM) [18]. BEM calculates the arithmetic mean of individual ensemble predictions by setting $\boldsymbol{\alpha}$ in (1) to

$$\boldsymbol{\alpha}_{BEM} = \frac{1}{T}\mathbf{1}, \tag{2}$$

where $\mathbf{1} = (1, \ldots, 1)^\top$. In other words, BEM weights all base functions uniformly, with no consideration regarding the data or individual generalization errors of ensemble members. The main advantages of BEM are its universality and numerical stability.

The generalized ensemble method (GEM) is a related yet more complex technique [18]. In GEM, the covariance matrix of the individual base-predictor errors, denoted here as $\boldsymbol{\mathcal{E}}$, is used to determine the weights

$$\boldsymbol{\alpha}_{GEM} = \frac{\boldsymbol{\mathcal{E}}^{-1}\mathbf{1}}{\mathbf{1}^\top \boldsymbol{\mathcal{E}}^{-1}\mathbf{1}}. \tag{3}$$

Formally, $\boldsymbol{\mathcal{E}}$ is defined as the covariance of

$$\boldsymbol{\varepsilon}(\boldsymbol{x}) \triangleq \left(\hat{f}_1(\boldsymbol{x}) - f(\boldsymbol{x}), \ldots, \hat{f}_T(\boldsymbol{x}) - f(\boldsymbol{x})\right)^\top. \tag{4}$$

In practice, inverting $\boldsymbol{\mathcal{E}}$ may be computationally unstable, hence $\boldsymbol{\alpha}_{GEM}$ is set to the eigenvector of $\boldsymbol{\mathcal{E}}$ whose corresponding eigenvalue is the minimal eigenvalue of $\boldsymbol{\mathcal{E}}$, normalized such that the sum of its elements equals 1. To simplify notations in the sequel, for a (non-zero) positive semi-definite matrix $\boldsymbol{A} \in \mathbb{R}^{T \times T}$, we define $\boldsymbol{v}_i^{\boldsymbol{A}}$ to be the i-th eigenvector of $\boldsymbol{A}$, where the eigenvectors are ordered such that their corresponding eigenvalues are non-decreasing. For example, $\boldsymbol{v}_1^{\boldsymbol{A}}$ is the eigenvector that corresponds to the minimal eigenvalue of $\boldsymbol{A}$.

Although GEM is theoretically superior over BEM in terms of MSE, it is much less stable since it commonly requires estimating and/or inverting covariance matrices. These disadvantages can be circumvented by minimizing the *empirical* MSE over the training (and/or validation) set(s). The most prominent methods of this kind are the interpolating predictor [16], which constrains $\boldsymbol{\alpha}$ to be component-wise non-negative, and the linear regressor (LR) [19], which does not impose any constraints on $\boldsymbol{\alpha}$. The LR regressor is defined as

$$\boldsymbol{\alpha}_{LR} = (\boldsymbol{F}^\top \boldsymbol{F})^{-1} \boldsymbol{F}^\top \boldsymbol{y}, \tag{5}$$

where $\boldsymbol{y} = (y_1, \ldots, y_{N_s})^\top$ and

$$\boldsymbol{F} = (\boldsymbol{\varphi}(x_1), \ldots, \boldsymbol{\varphi}(x_{N_s}))^\top$$

is the the empirical prediction matrix. It was proved [19] that $\boldsymbol{\alpha}_{RL}$ minimizes the empirical MSE $||\boldsymbol{F}\boldsymbol{\alpha} - \boldsymbol{y}||_2^2$.

Our model considers the aggregation of ensemble predictions corrupted by additive noise.

Definition 2 (noisy prediction)**.** *Denote $\boldsymbol{n} = (n_1, \ldots, n_T)^\top$ where $\mathbb{E}[\boldsymbol{n}] = \mathbf{0}$ and its covariance matrix is $\boldsymbol{\Sigma}$. For a data sample $\boldsymbol{x} \in \mathcal{X}^d$, define the noisy prediction as*

$$\tilde{f}(\boldsymbol{x}) = \boldsymbol{\alpha}^\top \tilde{\boldsymbol{\varphi}}(\boldsymbol{x}), \text{ where } \tilde{\boldsymbol{\varphi}}(\boldsymbol{x}) = \boldsymbol{\varphi}(\boldsymbol{x}) + \boldsymbol{n}. \tag{6}$$

An important and widely-used performance evaluation criterion for regression is the mean squared error (MSE).

Definition 3 (mean squared error, MSE)**.** *Given a target function $f(\cdot)$ and a data sample $\boldsymbol{x} \in \mathcal{X}^d$, define the mean squared error (MSE) of the predictor $\hat{f}(\boldsymbol{x})$ as*

$$J(\hat{f}(\boldsymbol{x})) = \mathbb{E}_f \left[\left(f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x}) \right)^2 \right], \tag{7}$$

where the expectation is over realizations of the model f.

Usually, the model f is realized through the data samples. Hence the expectation is performed over $\boldsymbol{x}$, assuming that model realizations and different data samples are interchangeable. Therefore, the MSE of the predictor $\hat{f}$ is defined as $J(\hat{f}) = \int_{\boldsymbol{x} \in \mathcal{X}^d} J(\hat{f}(\boldsymbol{x})) p(\boldsymbol{x}) d\boldsymbol{x}$. Standard ensemble training and integration techniques usually seek minimal MSE.

For *noisy* prediction, however, ensemble integration should be optimized such that the expected MSE over both model realizations *and* noise realizations is minimized. In other words, the noisy setup requires not only low generalization error, but also robustness to prediction noise. The objective of this paper is, therefore, to minimize the "doubly" expected MSE

$$\mathbb{E}_{f,\tilde{f}} \left[\left(f(\boldsymbol{x}) - \tilde{f}(\boldsymbol{x}) \right)^2 \right], \tag{8}$$

under different assumptions on the coefficient vector $\boldsymbol{\alpha}$.

3. ROBUST ENSEMBLE INTEGRATION

In this section, we propose new and robust versions of the aforementioned linear ensemble integration methods: BEM, GEM, and LR. Toward this goal, we start with an analytic derivation of the expected MSE as a figure of merit for robust regression.

3.1. Expected MSE for robust prediction

The expected MSE, described in (8), can be derived analytically for additive noise independent of the base functions $\{\hat{f}_t(\cdot)\}_{t=1}^T$

Theorem 1. *Let $\{\hat{f}_t(\cdot)\}_{t=1}^T$ be a regression ensemble that produces a noisy prediction $\tilde{f} = \boldsymbol{\alpha}^\top (\boldsymbol{\varphi} + \boldsymbol{n})$, where $\mathbb{E}[\boldsymbol{n}] = \boldsymbol{0}$ and $\mathbf{cov}[\boldsymbol{n}] = \boldsymbol{\Sigma}$. Then, the expected MSE of $\tilde{f}$ is*

$$J(\tilde{f}) = J(\hat{f}) + \boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}. \tag{9}$$

Proof. The expected MSE of $\tilde{f}$ is

$$J(\tilde{f}) = \int_{\boldsymbol{x} \in \mathbb{R}^d} \mathbb{E}_{\boldsymbol{n}} \left[\left(\tilde{f}(\boldsymbol{x}) - f(\boldsymbol{x}) \right)^2 \right] p(\boldsymbol{x}) d\boldsymbol{x}, \tag{10}$$

where $p(\boldsymbol{x})$ is the probability of $\boldsymbol{x}$. For every $\boldsymbol{x} \in \mathcal{X}^d$, the expected MSE of $\tilde{f}(\boldsymbol{x})$ is

$$\begin{aligned} \mathbb{E}_{\boldsymbol{n}} \left[\left(f(\boldsymbol{x}) - \tilde{f}(\boldsymbol{x}) \right)^2 \right] &= \mathbb{E}_{\boldsymbol{n}} \left[\left(f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x}) - \boldsymbol{\alpha}^\top \boldsymbol{n} \right)^2 \right] \\ &= \left(f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x}) \right)^2 + \boldsymbol{\alpha}^\top \mathbb{E}_{\boldsymbol{n}} \left[\boldsymbol{n}\boldsymbol{n}^\top \right] \boldsymbol{\alpha} \\ &= J\left(\hat{f}(\boldsymbol{x}) \right) + \boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}. \end{aligned} \tag{11}$$

The proof is completed by taking the expectation over $\boldsymbol{x}$ in (11). □

The resulting expression of the expected MSE provides insight into the structure of prediction errors. The total error comprises two components: the first relates to model generalization (training-oriented), while the second is associated with the aggregated noise (prediction-oriented). In the sequel, we refer to the former as the *generalization error* and the latter as *prediction error*. Robust ensemble integration methods seek to minimize both quantities.

3.2. Robust normalized linear combining

We now harness the expected MSE from Theorem 1 toward deriving robust versions of BEM and GEM.

3.2.1. Robust BEM

Following the design principles of BEM, we derive *robust BEM* as a coefficient assignment that is universal over the model and/or generalization error, thus providing numerical stability and computational simplicity. Since the generalization error in (9) is model dependent, the robust BEM coefficients are designed to minimize the remaining prediction error, i.e., $\boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}$. In the following theorem we derive the robust BEM and prove its optimality.

Theorem 2. *Let $\boldsymbol{\Sigma}$ be the noise covariance (positive semi-definite) matrix. Then, the robust BEM coefficients, given by*

$$\boldsymbol{\alpha}_{rBEM} = \frac{\boldsymbol{v}_1^{\boldsymbol{\Sigma}}}{\boldsymbol{1}^\top \boldsymbol{v}_1^{\boldsymbol{\Sigma}}}, \tag{12}$$

minimize the prediction error $\boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}$ among the set of normalized coefficient vectors $\{\boldsymbol{v} \in \mathbb{R}^T : \boldsymbol{1}^\top \boldsymbol{v} = 1\}$.

Proof. The form of $\boldsymbol{\alpha}_{rBEM}$ being a normalized minimizer of $\boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}$ is a known property, which can be proved by substituting $\boldsymbol{\alpha} = \frac{\boldsymbol{a}}{\boldsymbol{1}^\top \boldsymbol{a}}$ into the objective function, and minimizing the resulting generalized Rayleigh quotient $\frac{\boldsymbol{a}^\top \boldsymbol{\Sigma} \boldsymbol{a}}{\boldsymbol{a}^\top \boldsymbol{1}\boldsymbol{1}^\top \boldsymbol{a}}$. □

However, to balance the two terms of (9), it may be beneficial to incorporate into $\boldsymbol{\alpha}$ subsequent eigenvectors as well, thus reducing the generalization variance at the cost of potentially increasing the prediction error. We, therefore, define a hyper-parameter $M \in \mathbb{N}$, which describes the number of eigenvectors included in the calculation of the robust BEM coefficients. Selecting $M = 1$ naturally guarantees minimal prediction error, but choosing $M > 1$ is also justified by virtue of ensemble diversity toward producing low generalization error [21]. For any $1 \leq M \leq T$, we set the robust BEM coefficients to be the arithmetic mean of the selected eigenvectors, as follows,

$$\boldsymbol{\alpha}_{rBEM} = \frac{1}{M} \sum_{m=1}^{M} \frac{\boldsymbol{v}_m^{\boldsymbol{\Sigma}}}{\mathbf{1}^\top \boldsymbol{v}_m^{\boldsymbol{\Sigma}}}. \tag{13}$$

3.2.2. *Robust GEM*

Moving to GEM, we design *robust GEM* as an optimal linear normalized ensemble combining method in terms of the expected MSE. Given Theorem 1 and [18], and assuming the MSE has zero mean, the expected MSE can be formulated as

$$J(\tilde{f}) = \boldsymbol{\alpha}^\top \boldsymbol{\mathcal{E}} \boldsymbol{\alpha} + \boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}, \tag{14}$$

where the generalization error is expressed through $\boldsymbol{\mathcal{E}}$, the covariance matrix of individual base-function errors (as in (3)). Similarly to (12), we set the robust GEM coefficients to be

$$\boldsymbol{\alpha}_{rGEM} = \boldsymbol{v}_1^{\boldsymbol{\mathcal{E}}+\boldsymbol{\Sigma}}, \tag{15}$$

thus guaranteeing minimal expected MSE (as we prove in the following theorem). Unlike robust BEM, incorporating additional eigenvectors is unnecessary in this case, since the overall expected MSE is minimized by $\boldsymbol{v}_1^{\boldsymbol{\mathcal{E}}+\boldsymbol{\Sigma}}$.

Theorem 3. *Let $\boldsymbol{\Sigma}$ and $\boldsymbol{\mathcal{E}}$ be the covariance matrices of $\boldsymbol{n}$ and ε, respectively. Then, the robust GEM coefficients $\boldsymbol{\alpha}_{rGEM}$ minimize the expected MSE $J(\tilde{f})$.*

Proof. Both $\boldsymbol{\mathcal{E}}$ and $\boldsymbol{\Sigma}$ are positive semi-definite, since they are covariance matrices. Hence, their sum is also positive semi-definite (with rank greater or equal than the individual ranks). From this point, the arguments in the proof of Theorem 2 can be adapted to $\boldsymbol{\mathcal{E}} + \boldsymbol{\Sigma}$. □

Since $\boldsymbol{\mathcal{E}}$ is typically unavailable in practice, it is common to use an estimate of the covariance matrix, given by $\frac{1}{N_s-1} \sum_{i=1}^{N_s} \boldsymbol{\varepsilon}(\boldsymbol{x}_i)\boldsymbol{\varepsilon}(\boldsymbol{x}_i)^\top$, where

$$\boldsymbol{\varepsilon}(\boldsymbol{x}) = \left(\hat{f}_1(\boldsymbol{x}) - y(\boldsymbol{x}), \ldots, \hat{f}_T(\boldsymbol{x}) - y(\boldsymbol{x})\right)^\top.$$

We used this estimate for the simulations presented in the sequel.

3.3. Robust unconstrained linear combining

We proceed with a robust version of the LR. Following its formulation as the minimizer of the *empirical* MSE of a dataset $\mathcal{S}$, for *robust* LR we consider the *empirical expected* MSE

$$J_{\text{emp}}(\mathcal{S}) \triangleq \mathbb{E}_{\boldsymbol{n}}\left[||\tilde{\boldsymbol{F}}\boldsymbol{\alpha} - \boldsymbol{y}||_2^2\right], \tag{16}$$

where $\tilde{\boldsymbol{F}}_{i,t} \triangleq f_t(\boldsymbol{x}_i) + n_{i,t}$ and $\boldsymbol{n}_i = (n_{i,1}, \ldots, n_{i,T})$ are mutually independent random vectors (i.e., $\mathbb{E}\left[n_{i,t}n_{j,t'}\right] = 0$ for every $1 \leq i \neq j \leq N_s$ and $t \neq t'$) with covariance matrix $\mathbf{cov}\left[\boldsymbol{n}_i\right] = \boldsymbol{\Sigma}$. The robust LR coefficients are derived in the following theorem.

Theorem 4. *Denote $\boldsymbol{F} = (\boldsymbol{\varphi}(x_1), \ldots, \boldsymbol{\varphi}(x_{N_s}))^\top$ and $\boldsymbol{N} = (\boldsymbol{n}_1, \ldots, \boldsymbol{n}_{N_s})^\top$ where $\mathbf{cov}\left[\boldsymbol{n}_i\right] = \boldsymbol{\Sigma}$ for $1 \leq i \leq N_s$. Then, the robust LR coefficients*

$$\boldsymbol{\alpha}_{rLR} = (\boldsymbol{F}^\top \boldsymbol{F} + N_s \boldsymbol{\Sigma})^{-1} \boldsymbol{F} \boldsymbol{y}. \tag{17}$$

minimize the empirical expected MSE from (16).

Proof. We first decompose the empirical MSE to

$$\mathbb{E}\left[||\tilde{\boldsymbol{F}}\boldsymbol{\alpha} - \boldsymbol{y}||_2^2\right] = \mathbb{E}\left[\boldsymbol{\alpha}^\top \tilde{\boldsymbol{F}}^\top \tilde{\boldsymbol{F}} \boldsymbol{\alpha} - 2\boldsymbol{y}^\top \tilde{\boldsymbol{F}} \boldsymbol{\alpha} + \boldsymbol{y}^\top \boldsymbol{y}\right]. \tag{18}$$

Obviously, $\boldsymbol{y}^\top \boldsymbol{y}$ does not depend on $\boldsymbol{\alpha}$ and $\mathbb{E}\left[\boldsymbol{y}^\top \tilde{\boldsymbol{F}}^\top \boldsymbol{\alpha}\right] = \boldsymbol{y}^\top \boldsymbol{F}^\top \boldsymbol{\alpha}$. Since $\boldsymbol{\varphi}(\cdot)$ is independent of $\boldsymbol{n}$ for every $\boldsymbol{x}$, the remaining term in (18) can be manipulated to

$$\mathbb{E}\left[\boldsymbol{\alpha}^\top \tilde{\boldsymbol{F}}^\top \tilde{\boldsymbol{F}} \boldsymbol{\alpha}\right] = \boldsymbol{\alpha}^\top (\boldsymbol{F}^\top \boldsymbol{F} + N_s \boldsymbol{\Sigma}) \boldsymbol{\alpha}. \tag{19}$$

Using these identities and equating the derivative of the right-hand-side of (18) to zero yields $\boldsymbol{\alpha}_{rLR}$ from (17), which therefore minimizes $J_{\text{emp}}(\mathcal{S})$. □

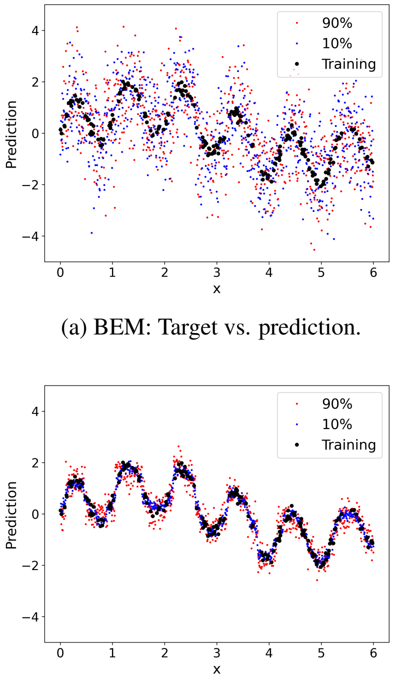
4. EXPERIMENTAL RESULTS

In this section, the theoretical framework of noisy prediction (provided in Section 2) is employed for various datasets and several noise conditions. We experiment with our proposed robust integration methods from Section 3 using an ensemble of $T = 10$ base functions, each of which is a depth 6 decision tree trained individually. Based on 20% of the available examples, the ensemble is trained to predict the target function using the Bagging [15] procedure, i.e., each decision tree is trained individually over a randomly-selected subset of training data samples. The remaining data samples are used as a test set for performance evaluation. During inference on the test set, the individual predictions of the ensemble members are added with noise before they are combined to produce the final prediction.

We aim to exemplify the benefits of our robust methods, compared to non-robust classical methods, by providing an empirical comparative evaluation of the two approaches. To conduct a consistent evaluation, we define the following SNR measure

$$\text{SNR} = \frac{\mathbf{var}(\boldsymbol{y})}{\text{Tr}(\boldsymbol{\Sigma})}, \tag{20}$$

where $\boldsymbol{y}$ is a vector consisting of the training dataset target values and $\boldsymbol{\Sigma}$ is the noise covariance.



(a) BEM: Target vs. prediction.

(b) Robust BEM: Target vs. prediction.

Fig. 1: Predictions of BEM and robust BEM for a sinusoidal sum target function at SNR=-30 dB with two noise regimes: 10% and 90% noisy base functions.

To motivate our model and the effectiveness of robust methods, we start our experiments with a synthetic dataset generated by a sinusoidal sum target function

$$f(x) = \sin(x) + \sin(6x) \text{ where } 0 \leq x \leq 6. \tag{21}$$

In this specific scenario, we illustrate the performance of BEM as the integration method. We tested the final ensemble prediction for a fixed SNR=-30 dB with independent and identically distributed Gaussian noise. We selected the fraction of noisy predictors to be either 10% or 90%, and set the remaining predictors with noise variances equal to the measurement noise ϵ ($\sigma_\epsilon^2 = 0.1$).

The adverse effect of corrupt predictors on the final prediction, when standard ensemble integration techniques are employed, is illustrated in Fig. 1a. It can be seen that even a small fraction of noisy predictors (e.g., 10%) can significantly degrade the overall ensemble performance, at a given SNR. The performance of robust integration methods, on the other hand, is exemplified in Fig. 1b. Comparing the two figures demonstrates the enhanced capability of robust methods to generate an accurate final prediction from noisy individual predictions. Furthermore, robust BEM enjoys improved performance due to its noise-informed design as the fraction of noisy predictors is reduced.

We experimented with several additional datasets to further validate the added value of robust ensemble integration. To the sinusoidal sum target function, we added two other synthetic functions and three real-world datasets:

- Synthetic linear combination: $f(\boldsymbol{x}) = \boldsymbol{c}^\top \boldsymbol{x}$ where $\boldsymbol{x} \in \mathbb{R}^d$, $d = 15$ and $-4 \leq \|\boldsymbol{x}\|_{\ell_1} \leq 4$ and $\boldsymbol{c}$ is a vector of random real numbers.
- Synthetic exponential sum: $f(x) = \exp(-x^2) + 1.5\exp(-(x-2)^2)$ where $0 \leq x \leq 6$.
- King County house prices: The dataset consists of $d = 20$ variables and $N_s = 21613$ samples.
- UCI diabetes patient records [22]: The dataset consists of $d = 10$ variables and $N_s = 442$ samples.
- White wine quality [23]: The dataset consists of $d = 11$ variables and $N_s = 4898$ samples.

Each of the synthetic datasets comprised of $N_s = 1000$ training samples, each of which generated by summing the target function with independent and identically distributed random Gaussian measurement noise $\epsilon \sim \mathcal{N}(0, 0.1)$.

We evaluate the performance of each of the suggested robust integration methods by calculating the gain in MSE compared to its non-robust counterpart, i,e.,

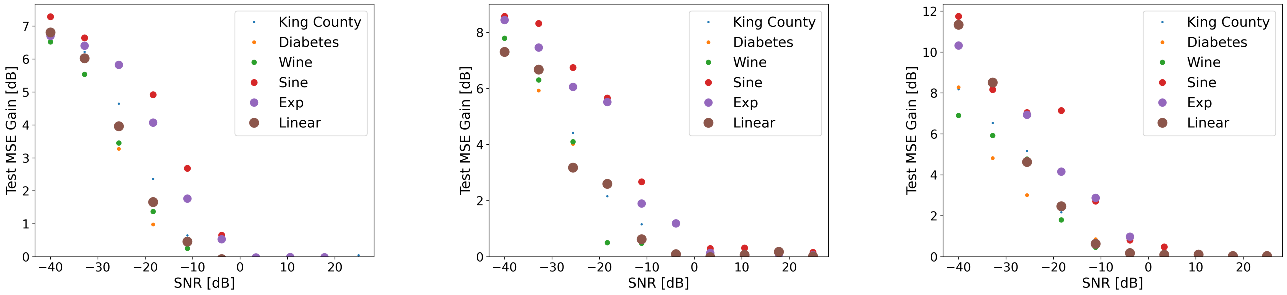
$$\mathcal{G}_\Theta = 10\log_{10}\left(\frac{\mathbb{E}[(f(\boldsymbol{x}) - \boldsymbol{\alpha}_\Theta^\top \tilde{\boldsymbol{\varphi}}(\boldsymbol{x}))^2]}{\mathbb{E}[(f(\boldsymbol{x}) - \boldsymbol{\alpha}_{r\Theta}^\top \tilde{\boldsymbol{\varphi}}(\boldsymbol{x}))^2]}\right), \tag{22}$$

where $\Theta \in \{\text{BEM, GEM, LR}\}$.

Figure 2 presents the MSE gains $\mathcal{G}_{BEM}$, $\mathcal{G}_{GEM}$ and $\mathcal{G}_{LR}$. The gains were calculated for all six datasets over a wide range of SNRs. While the MSE achieved by robust and non-robust methods is similar for high SNRs, in low SNRs (less than 0 dB) robust methods significantly outperform non-robust ones. For some datasets and SNRs, the MSE gain can reach up to 12 dB, i.e., an order of magnitude reduction obtained by the robust methods. We generally find that robust integration is highly effective for noisy prediction. Due to their noise-informed design, robust integration methods were able to produce accurate predictions from very noisy individual predictions, especially compared to non-robust methods. We deduce that the poor performance of the classical techniques stems from the noise-oblivious design.

5. CONCLUSIONS

We have developed a new theoretical framework for ensemble regression with noisy base functions. Inspired by classical linear ensemble integration methods, we suggested several noise-robust generalized methods that achieve better performance when tested with synthetic datasets and real-world regression problems. This work can be extended by deriving robust versions for additional ensemble integration methods.



(a) BEM

(b) GEM

(c) LR

Fig. 2: Gain in prediction MSE obtained by robust methods compared to their non-robust counterparts.

Also, we concentrated on Bagging as a training procedure, but the proposed framework can be employed for other techniques (e.g., Stacking and Boosting). Finally, we conjecture that analytical properties and guarantees on the expected MSE may be obtained under natural assumptions regarding the prediction noise's statistical properties.

References

[1] A. Gholami, "AI and memory wall," medium.com/riselab, March 2021, Accessed: 23/12/2021.

[2] N. García-Pedrajas, C. Hervás-Martínez, and D. Ortiz-Boyer, "Cooperative coevolution of artificial neural network ensembles for pattern classification," *IEEE Trans. on Evolutionary Computation*, vol. 9, no. 3, pp. 271–302, 2005.

[3] J. Mendes-Moreira, C. Soares, A. M. Jorge, and J. F. De Sousa, "Ensemble approaches for regression: A survey," *ACM Computing Surveys (CSUR)*, vol. 45, no. 1, pp. 1–40, 2012.

[4] Z. Gao, H. Zhang, Y. Yao, J. Xiao, S. Zeng, G. Ge, Y. Wang, A. Ullah, and P. Reviriego, "Soft error tolerant convolutional neural networks on FPGAs with ensemble learning," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 2022.

[5] W. Kim, J. Park, J. Yoo, H. J. Kim, and C. G. Park, "Target localization using ensemble support vector regression in wireless sensor networks," *IEEE Trans. on Cybernetics*, vol. 43, no. 4, pp. 1189–1198, 2012.

[6] W. He, D. Yang, H. Peng, S. Liang, and Y. Lin, "An efficient ensemble binarized deep neural network on chip with perception-control integrated," *Sensors*, vol. 21, no. 10, 2021.

[7] Y. Ben-Hur, A. Goren, D. Klang, Y. Kim, and Y. Cassuto, "Mitigating noise in ensemble classification with real-valued base functions," in *2022 IEEE Intl. Symp. on Inf. Theory (ISIT)*.

[8] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, pp. e1249, 2018.

[9] X. Qiu, L. Zhang, Y. Ren, P. Suganthan, and G. Amaratunga, "Ensemble deep learning for regression and time series forecasting," in *Proc. 2014 IEEE Symp. on Computational Intelligence in Ensemble Learning*, 2014, pp. 1–6.

[10] H. Kaneko, "Automatic outlier sample detection based on regression analysis and repeated ensemble learning," *Chemometrics and Intelligent Lab. Systems*, vol. 177, pp. 74–82, 2018.

[11] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2014, pp. 1867–1874.

[12] P. Čížek and S. Sadıkoğlu, "Robust nonparametric regression: A review," *WIREs Computational Statistics*, vol. 12, no. 3, pp. e1492, 2020.

[13] S. Du, Y. Wang, S. Balakrishnan, P. Ravikumar, and A. Singh, "Robust nonparametric regression under huber's ϵ-contamination model," *arXiv, math.ST*, 2018.

[14] D. Blatná, "Outliers in regression," *Trutnov*, vol. 30, pp. 1–6, 2006.

[15] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[16] L. Breiman, "Stacked regressions," *Machine Learning*, vol. 24, no. 1, pp. 49–64, 1996.

[17] Y. Freund and R. E Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[18] M. P. Perrone, *Improving Regression Estimation: Averaging Methods for Variance Reduction with Extensions to General Convex Measure Optimization*, Ph.D. thesis, Citeseer, 1993.

[19] M. LeBlanc and R. Tibshirani, "Combining estimates in regression and classification," *Journal of the American Statistical Association*, vol. 91, no. 436, pp. 1641–1650, 1996.

[20] C. J. Merz, *Classification and Regression by Combining Models*, Ph.D. thesis, University of California, Irvine, 1998.

[21] G. Zenobi and P. Cunningham, "Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error," in *Proc. European Conf. on Machine Learning*. Springer, 2001, pp. 576–587.

[22] D. Dua and C. Graff, "UCI machine learning repository," 2017, Accessed: 10/01/2022.

[23] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis, "Modeling wine preferences by data mining from physicochemical properties," *Decision Support Systems*, vol. 47, no. 4, pp. 547–553, 2009.