

Robust Regression with Ensembles Communicating over Noisy Channels

Yuval Ben-Hur, *Graduate Student Member, IEEE*, and Yuval Cassuto, *Senior Member, IEEE*

Abstract—As machine-learning models grow in size, their implementation requirements cannot be met by a single computer system. This observation motivates distributed settings, in which intermediate computations are performed across a network of processing units, while the central node only aggregates their outputs. However, distributing inference tasks across low-precision or faulty edge devices, operating over a network of noisy communication channels, gives rise to serious reliability challenges. We study the problem of an ensemble of devices, implementing regression algorithms, that communicate through additive noisy channels in order to collaboratively perform a joint regression task. We define the problem formally, and develop methods for optimizing the aggregation coefficients for the parameters of the noise in the channels, which can potentially be correlated. Our results apply to the leading state-of-the-art ensemble regression methods: bagging and gradient boosting. We demonstrate the effectiveness of our algorithms on both synthetic and real-world datasets.

Index Terms—ensemble learning, regression, channel noise, inference algorithms, distributed machine learning.

I. INTRODUCTION

MACHINE learning (ML) and artificial intelligence (AI) algorithms have been game changers across many technology fields over the past decade, achieving unprecedented performance in extremely complex tasks. ML models, such as deep neural networks (DNNs) that are a major thrust of this revolution, are driven mainly by their representation power. Complicated tasks addressed nowadays, such as image processing or natural language processing (NLP), require large models with billions of parameters. The exponential growth in model size has made training and inference incredibly challenging computational tasks.

The meteoric success of ML/AI has driven increasing demands for computing hardware, which is needed for improving performance and for extending into more challenging applications. However, it is becoming clear that the exponentially growing requirements of state-of-the-art models cannot be met by hardware scaling rates [1]. This motivates an increasing interest in *distributed ML/AI* that allow flexible and elastic

resource provisioning across multiple locations and entities. Distributed training and inference have two additional advantages stemming from the computations they perform on local data sets: reducing communication costs and improving privacy.

A natural framework for distributed ML/AI is *ensemble* learning algorithms. In this framework, the learning task is shared by multiple *base predictors*, whose outputs are *aggregated* at inference time to obtain a strong prediction. Ensemble methods have already been proven to achieve state-of-the-art performance [2] (and many more). In addition to their applicability to distributed inference over communication networks [3], [4], they are compatible with common AI chip architectures [5], so can potentially enrich the algorithmic stack in existing ML/AI chips.

This paper studies distributed *inference* over noisy channels, and in particular the task of *regression*. Regression seeks to fit a functional model to the relationship between a dependent variable and one or more independent variables called features. A regression model is usually trained from realizations of features and the corresponding, possibly inaccurate, values of the dependent variable. *Ensemble regression* is a powerful and elegant technique for constructing models that are aggregates of multiple base regressors. Such models play a key role in a wide range of real-world problems and applications [6]: from big-data analysis (e.g., time-series forecasting [7], outlier detection [8]) to estimation problems (e.g., source localization [3], image alignment [9]) and beyond. Further, regression may be regarded as a form of *data compression* from redundant features to task-relevant variables, thus ensemble regression may be used for distributed *task-oriented source coding*. This success and promise notwithstanding, when deploying ensemble regression models over noisy channels, the noise added in transit between the base regressors and the aggregating node may severely compromise the accuracy, and thus requires mitigation. The question, which we answer in this paper to the affirmative, is can we mitigate this degradation if we know the statistics of the noise added to the regressor outputs?

Recent attempts to implement learning ensembles on real hardware devices have encountered serious reliability issues due to noise [3], [4], [10]. In [10], 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 [3] and [4], neural network ensembles were deployed on wireless sensor networks for localization and navigation purposes. The network comprised low-precision edge devices that produce unreliable individual predictions, resulting in degraded overall prediction performance. Similarly, distributed deep neural networks (DDNN), in which a part of the neural network is duplicated across multiple edge devices that communicate with a server on the cloud, are also required to aggregate features transmitted over noisy channels [11]. A similar noisy setting also exists in neural-network hardware accelerators, wherein analog compute engines [12] aggregate neuron inputs contaminated by computation noise and non-linearities. Another emerging concept that benefits from the framework studied in this paper is over-the-air computation for distributed learning [13], which utilizes beamforming to implement physical summation (i.e., aggregation) of analog values sent over wireless channels by individual edge devices.

Most prior work dealing with noise in the ensemble machine-learning literature has focused on *noisy training data* [14] (e.g., model mixture contamination [15], or outlying samples [16]). Distributed inference ensembles plagued by channel noise have received attention recently, but mostly focused on the *classification* problem [17]–[21]. The problem of noisy ensemble regression has not received a prior systematic study, to the best of our knowledge. Our recent short paper [22] identifies the problem of using standard ensemble algorithms with communication noise, and suggests some basic optimization framework for the *mean squared error (MSE)* loss. The results of this present paper significantly extend that initial framework into general loss functions, such as *mean absolute error (MAE)* and general ℓ_p norms, apply to more ensemble algorithms, and include analytical results such as lower and upper error bounds. Earlier works addressed reliability challenges of ensemble regression using heuristic or ad-hoc solutions: from increasing the ensemble size [4], to heuristically re-weighting [3], and even completely ignoring [10] the outputs of individual base regressors.

Our study considers the two leading state-of-the-art ensemble regression methods: *bagging* [23], and

gradient boosting [24], [25]. The former trains each base regressor on a different sample from the data set, and the latter iteratively augments the ensemble with a base regressor that trains on the errors of the current ensemble members. A key feature of our proposed framework is that the noise-mitigating algorithms are developed around, and not as replacement, of the existing ensemble algorithms. This implies that any progress made in algorithms driving these methods can also benefit their deployment in noisy settings. Toward that, our optimization target for the noisy setting is the *aggregation coefficients* that weight the base regressors' contributions to the final regression. While bagging and gradient boosting have a structurally identical regression function, the difference in how they are trained dictates different noise mitigation: in bagging each regressor is trained independently, so the aggregation coefficients can be calculated “post-training” after all base regressors are known, while in gradient boosting each aggregation coefficient is set as part of a training iteration, so the noise-mitigating optimization needs to be done within the training sequence.

The results of the paper are organized as follows. In Section II, we formally define the noisy ensemble-regression problem and motivate it by illustrating the effect of channel noise on the final ensemble prediction. In Section III, we address robust regression for *bagging ensembles*. The sought measure of robustness is reduction in the excess regression loss due to noise, compared to unmitigated ensembles. We start with results for general ℓ_p -norm loss, and then propose optimization algorithms for the MSE and MAE loss functions. For MSE we give a closed-form expression for the aggregation coefficients in a generalized optimization problem that allows weighting the channel noise vs. the model error, which can be used to handle uncertainty in the noise model. For MAE we derive analytical results for normally distributed noise that enable a gradient-based optimization algorithm, as well as error upper and lower bounds. Section IV moves to address robust *gradient boosting*, where the main contribution is a training algorithm that sets the aggregation coefficients to optimize the *expected loss* in the noisy setting. For the MSE we derive a closed-form expression for the optimal coefficients. In section V, we present an empirical study of the proposed methods for synthetic and real-world datasets. The results show that the proposed algorithms significantly outperform baseline bagging aggregation algorithms for both MSE and MAE. For gradient boosting the results show that the noise-mitigated algorithm enables

a desired property of decreasing the error as the ensemble size grows, while the baseline algorithm exhibits increasing error once deployed with noise. The paper is finally concluded in Section VI.

II. DISTRIBUTED NOISY REGRESSION

A. Model formulation

Consider an unknown (deterministic) target function $f(\cdot) : \mathbb{R}^u \rightarrow \mathbb{R}$ that is required to be estimated given a set of data samples $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$, where the feature vector is $\mathbf{x}_i \in \mathbb{R}^u$ ($u \in \mathbb{N}$) and its corresponding value is $y_i \in \mathbb{R}$. We denote the sets of \mathbf{x}_i and y_i from which the dataset \mathcal{S} is comprised as \mathcal{X} and \mathcal{Y} , respectively. A base regressor (or sub-regressor) $\varphi(\cdot) : \mathbb{R}^u \rightarrow \mathbb{R}$ is a function aimed to estimate $f(\cdot)$, based on \mathcal{S} and a prescribed loss function $\delta(\cdot, \cdot)$ (measuring dissimilarity between y_i and $\varphi(\mathbf{x}_i)$) as a figure of merit. In *ensemble* regression, the regressor is implemented by aggregating predictions from multiple base regressors $\{\varphi_t(\cdot)\}_{t=1}^T$, as now defined. Throughout the paper, T denotes the *ensemble size*.

Definition 1 (regression ensemble). *Define a regression ensemble of size $T \in \mathbb{N}$ as a set of functions $\{\varphi_t(\cdot)\}_{t=1}^T$ where $\varphi_t : \mathbb{R}^u \rightarrow \mathbb{R}$ and an aggregation function $\hat{f} : \mathbb{R}^T \rightarrow \mathbb{R}$.*

The ensemble's prediction for a data sample \mathbf{x} is obtained by evaluating the ensemble aggregation function on the individual predictions of the base regressors, i.e., $\hat{f}(\varphi_1(\mathbf{x}), \dots, \varphi_T(\mathbf{x}))$. In this paper, we consider an ensemble regressor implemented in a distributed fashion over *noisy channels*, a setting that consists of the following components:

- 1) A set of individual computation nodes implementing the base regressors $\{\varphi_t(\cdot)\}_{t=1}^T$.
- 2) Noisy communication channels $C_t : \mathbb{R} \rightarrow \mathbb{R}$ connecting each computation node to a central processor.
- 3) A central processor that produces the final prediction by applying the ensemble aggregation function $\hat{f}(\tilde{\varphi}_1, \dots, \tilde{\varphi}_T)$ where $\tilde{\varphi}_t \triangleq C_t(\varphi_t(\cdot))$.

This setting, referred to as *noisy ensemble regression*, is depicted in Fig. 1 for additive noise channels $C_t : \tilde{\varphi}_t(\cdot) = \varphi_t(\cdot) + n_t$, where n_t is a random variable representing the additive noise. Correspondingly, the aggregated noisy prediction is denoted $\tilde{f}(\mathbf{x})$.

Due to their effectiveness and popularity, we specialize the discussion to *linearly* aggregated ensembles $\hat{f}(\mathbf{z}) = \boldsymbol{\alpha}^\top \mathbf{z}$, which in the noisy case give

$$\tilde{f}(\mathbf{x}) = \boldsymbol{\alpha}^\top \tilde{\boldsymbol{\varphi}}(\mathbf{x}), \quad (1)$$

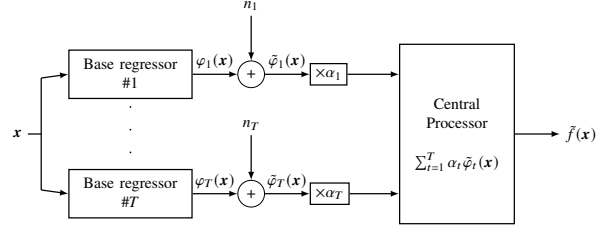


Fig. 1: Block diagram of a noisy ensemble regression system.

where $\tilde{\boldsymbol{\varphi}}(\mathbf{x}) = (\tilde{\varphi}_1(\mathbf{x}), \dots, \tilde{\varphi}_T(\mathbf{x}))^\top$ denotes a vector of channel outputs and $\boldsymbol{\alpha} \in \mathbb{R}^T$ is an aggregation coefficient vector. For every data sample $\mathbf{x} \in \mathbb{R}^u$, each prediction $\tilde{\varphi}_t(\mathbf{x})$ is weighted by an aggregation coefficient $\alpha_t \in \mathbb{R}$ to produce an optimized ensemble prediction. These coefficients are determined during training (or post-training) toward optimizing the final predictions, and they do not change at inference time.

For the remainder of this paper, we address this setting of a linearly-aggregated regression ensemble with additive noise. Our proposed methods aim to optimize the noisy ensemble prediction defined next.

Definition 2 (noisy ensemble prediction). *Let $\mathbf{n} = (n_1, \dots, n_T)^\top$ be a random vector. For a data sample \mathbf{x} , define the noisy ensemble prediction as*

$$\tilde{f}(\mathbf{x}) = \boldsymbol{\alpha}^\top \tilde{\boldsymbol{\varphi}}(\mathbf{x}), \text{ where } \tilde{\boldsymbol{\varphi}}(\mathbf{x}) = \boldsymbol{\varphi}(\mathbf{x}) + \mathbf{n}, \quad (2)$$

$\boldsymbol{\varphi}(\mathbf{x}) = (\varphi_1(\mathbf{x}), \dots, \varphi_T(\mathbf{x}))^\top$ is a vector of base-regressor outputs, and $\boldsymbol{\alpha} \in \mathbb{R}^T$ is the ensemble's coefficient vector.

While a “noiseless” regression ensemble can be trained to minimize the loss over model realizations (using a training dataset), the noisy setting requires to consider the expected loss for noise realizations as well. In other words, training should now seek not only low model error, but also robustness to aggregated prediction noise. It is therefore natural to consider the following “doubly” expected loss.

Definition 3. *Let $\tilde{f}(\cdot) : \mathbb{R}^u \rightarrow \mathbb{R}$ be a noisy ensemble prediction function with a coefficient vector $\boldsymbol{\alpha} \in \mathbb{R}^T$, and $f(\cdot)$ the target function to predict. For a loss function \mathcal{L} , define the expected loss as*

$$\tilde{J}_{\mathcal{L}} \triangleq J_{\mathcal{L}}(\tilde{f}) \triangleq \mathbb{E}_{\mathbf{n}} [\mathcal{L}(f, \tilde{f})], \quad (3)$$

where the loss is calculated over evaluations of the function $f(\mathbf{x})$ using a set of data samples $\mathbf{x} \in \mathcal{X}$ and the expectation is taken over realizations of the noise \mathbf{n} .

For example, the loss \mathcal{L} can be the *mean absolute error* (MAE) $\tilde{J}_1 \triangleq \mathbb{E}_{\mathbf{n}} \left[\frac{1}{|\mathcal{X}|} \sum_{\mathbf{x} \in \mathcal{X}} |f(\mathbf{x}) - \tilde{f}(\mathbf{x})| \right]$, the *mean squared error* (MSE) $\tilde{J}_2 \triangleq \mathbb{E}_{\mathbf{n}} \left[\frac{1}{|\mathcal{X}|} \sum_{\mathbf{x} \in \mathcal{X}} |f(\mathbf{x}) - \tilde{f}(\mathbf{x})|^2 \right]$, or the ℓ_p -norm based loss $\tilde{J}_{\ell_p} \triangleq \mathbb{E}_{\mathbf{n}} \left[\sqrt[p]{\frac{1}{|\mathcal{X}|} \sum_{\mathbf{x} \in \mathcal{X}} |f(\mathbf{x}) - \tilde{f}(\mathbf{x})|^p} \right]$.

The primary objective of this paper is to minimize this loss function. Practically, the model samples \mathbf{x} , $f(\mathbf{x})$ are realized through the training data samples in \mathcal{S} , and $\tilde{f}(\mathbf{x})$ is realized through noisy observations of the model $\hat{f}(\varphi(\mathbf{x}))$. Hence, in the forthcoming optimization results the loss is calculated using a sum over the training samples with y taking the role of $f(\mathbf{x})$, and then taking the expectation over the noise \mathbf{n} . For testing the generalization of the optimized models, a *test set* not used in training and optimization is taken as \mathcal{X} (with corresponding y values) in the loss expressions. Note that the noise $\mathbf{n} = \tilde{\varphi}(\mathbf{x}) - \varphi(\mathbf{x})$ is only present at inference: it is not visible during training, and in our post-training optimization only the *model* of the noise is assumed known (no noise samples are used in optimization). For comparing noisy to noiseless performance, we define the *noiseless loss* $J_{\mathcal{L}}$ in a similar way to $\tilde{J}_{\mathcal{L}}$, but with $\tilde{f}(\mathbf{x})$ replaced by $\hat{f}(\varphi(\mathbf{x}))$ and without the expectation over \mathbf{n} (clearly, because there is no noise).

To offer a degree of separation between the machine-learning layer and the communication layer, in the first part of the paper (Section III) we assume that the base regressors are trained without considering the noisy setting, thus mitigating the noise through the task of optimizing the aggregation coefficients post-training. Such a separation is motivated by settings in which the channels carrying predictions are not known at the time of training. We now state the primary problem addressed in the first part of this paper.

Problem 1. Given a dataset $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$ and a distribution of the noise \mathbf{n} , find the coefficient vector $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_T)^\top$ that minimizes $\tilde{J}_{\mathcal{L}}(\tilde{f})$ for an ensemble of base regressors $\{\varphi_t(\cdot)\}_{t=1}^T$.

Specifically, the majority of our results concentrates on the most widely accepted figures of merit in the context of regression: the MSE (\tilde{J}_2) and the MAE (\tilde{J}_1). Also, unless explicitly stated otherwise, the noise vector is assumed to be distributed with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}$. For some specific cases in the sequel, in which we assume the noise is normally distributed, we denote the probability density function (PDF) and the cumulative density function (CDF) of a normal random variable with mean 0 and variance 1 as $g(t) \triangleq \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}$ and $\phi(\psi) \triangleq \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\psi} e^{-\frac{t^2}{2}} dt$, respec-

tively. The error function, defined as $2\phi(\sqrt{2}x) - 1$, is denoted $\text{erf}(x)$ and the complementary error function $1 - \text{erf}(x)$ is denoted $\text{erfc}(x)$.

The approach we pursue in this paper to solve distributed noisy regression (Problem 1) is by “noise-robustifying” the two leading ensemble methods in the literature: bagging in Section III, and gradient boosting in Section IV.

B. Motivation for robust prediction

To motivate the problem of robust regression, we exemplify the adverse effect of noise on a standard regression bagging ensemble. Consider an ensemble of $T = 5$ decision-tree base regressors (with a maximal depth of 8), where each base regressor was trained on a different subset of the training set (known as “bagging”). The final prediction is obtained by averaging the individual predictions of the base regressors.

To conduct a consistent evaluation, we define the following signal-to-noise ratio (SNR) measures. For a single base regressor $\varphi_t(\cdot)$, we define SNR_t as $\frac{\varepsilon_y}{\sigma_t^2}$, where $\varepsilon_y \triangleq \frac{1}{N_s} \sum_{i=1}^{N_s} y_i^2$ is the normalized sum of squares of the training-dataset target values, and σ_t^2 is the t -th channel’s noise variance. For the entire ensemble, the SNR definition is extended to

$$\text{SNR} = \frac{T\varepsilon_y}{\text{Tr}(\boldsymbol{\Sigma})}, \quad (4)$$

where $\boldsymbol{\Sigma}$ is the noise covariance matrix and T is the ensemble size. The denominator of (4) measures the *total noise power* as the sum of noise variances over the T channels, while the numerator models the *total signal power* of T regressors, each estimating a function with average energy ε_y . When given in dB, the signal-to-noise ratio is calculated as $10 \log_{10}(\text{SNR})$.

In this sub-section, we exemplify a scenario in which a single base regressor communicates through a low-SNR channel, while the remainder of the ensemble is transmitted through channels with better SNRs. This setting is realized by a diagonal noise covariance matrix, in which the first diagonal element is set to factor $a \gg 1$ larger than the remaining diagonal elements. For a given ensemble SNR and a , the variance of the first channel $t = 1$ is set to

$$\sigma_1^2 = \frac{T\varepsilon_y}{\left(1 + \frac{T-1}{a}\right) \text{SNR}} \quad (5)$$

and the remaining variances $1 < t \leq T$ are $\sigma_t^2 = \frac{1}{a} \sigma_1^2$. We experiment with a synthetic dataset $y = f(x) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, 0.01)$ is a measurement noise and the target function is generated by sampling a sinusoidal sum: $f(x) = \sin(x) + \sin(6x)$, where $0 \leq x \leq 6$.

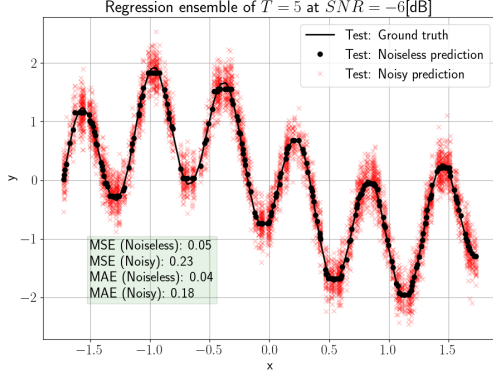


Fig. 2: Illustration of noisy and noiseless prediction for a Sine target function, with the corresponding values of its (root) MSE and MAE. Note that x was centralized to 0.

We evaluate in Fig. 2 the (root) MSE and the MAE with and without noise whose parameters are $\text{SNR} = -6\text{dB}$, $a = 20$. The prediction plots, as well as the resulting average MSE and MAE values (appearing on the lower left of the figure), illustrate the corrupting effects of channel noise on the quality of the ensemble’s prediction over the test set. In addition to the ground-truth values y (denoted: “Test: Ground truth”), Fig. 2 shows two additional curves: 1) predictions produced without noise (denoted “Test: Noiseless prediction”); and 2) predictions produced from the noisy base regressors (denoted “Test: Noisy prediction”).

The poor performance of the noisy predictor is easily observable. It can be seen that even a single base regressor with non-negligible noise can significantly degrade the overall ensemble performance for both the MSE and MAE measures. These results motivate the mitigation of base regressors’ channel noise by optimization of the aggregation coefficients.

III. ROBUST BAGGING

In the setting of bagging ensembles, the base regressors are individually trained on subsets of \mathcal{S} , and Problem 1 becomes post-training optimization of the aggregation coefficient vector α . The following result is obtained for a general ℓ_p -norm based loss.

Theorem 1. For any coefficient vector α and $p \in \mathbb{N}$,

$$\tilde{J}_{\ell_p}(\alpha) \leq J_{\ell_p}(\alpha) + \mathbb{E}_{\mathbf{n}} [|\alpha^\top \mathbf{n}|]. \quad (6)$$

where \mathbf{n} is the noise random vector and $J_{\ell_p}(\alpha)$ is the ℓ_p loss of the noiseless predictor $\alpha^\top \varphi(\cdot)$.

Proof. Using the triangle inequality to obtain an upper bound and then simplifying the resulting expressions, we get

$$\begin{aligned} \tilde{J}_{\ell_p}(\alpha) &= \mathbb{E}_{\mathbf{n}} \left[\sqrt[p]{\frac{1}{N_s} \sum_{i=1}^{N_s} |y_i - \alpha^\top \tilde{\varphi}(x_i)|^p} \right] \\ &= \mathbb{E}_{\mathbf{n}} \left[\sqrt[p]{\frac{1}{N_s} \sum_{i=1}^{N_s} |y_i - \alpha^\top \varphi(x_i) - \alpha^\top \mathbf{n}|^p} \right] \\ &\leq \sqrt[p]{\frac{1}{N_s} \sum_{i=1}^{N_s} |y_i - \alpha^\top \varphi(x_i)|^p} + \mathbb{E}_{\mathbf{n}} [|\alpha^\top \mathbf{n}|] \\ &= J_{\ell_p}(\alpha) + \mathbb{E}_{\mathbf{n}} [|\alpha^\top \mathbf{n}|]. \end{aligned} \quad (7)$$

□

Note that Theorem 1 applies to the MAE, because it coincides with the ℓ_1 loss for $p = 1$, but not to the MSE due to the square root in the ℓ_2 norm that is absent in the MSE. The upper bound of Theorem 1 provides insight into the structure of prediction errors in noisy ensemble regression: (6) facilitates a decomposition of the loss to two types. The first relates to model error, while the second is associated with the expected aggregated noise. In general, we seek to minimize both quantities for robust regression, but maintaining the distinction between them will be useful for specializing the optimization to particular system settings.

As the baseline approaches for ensemble aggregation, we consider the *basic ensemble method* (BEM) [26], according to which the coefficients are all equal

$$\alpha^{(BEM)} \triangleq \frac{1}{T} \mathbf{1}, \quad (8)$$

and the *generalized ensemble method* (GEM) [26], which sets α to minimize J_{ℓ_p} among *normalized* coefficients:

$$\alpha_{\ell_p}^{(GEM)} \triangleq \arg \min_{\alpha \in \mathbb{R}^T: \mathbf{1}^\top \alpha = 1} J_{\ell_p}(\alpha). \quad (9)$$

(recall that $J_{\ell_p}(\alpha)$ is the noiseless ℓ_p loss.) Note that GEM can be considered as a non-robust counterpart of our MSE-optimal approach presented now.

A. Robust bagging for MSE loss

We start by pursuing aggregation coefficients that optimize the mean squared error (MSE) loss. The MSE loss expression can be decomposed into two sub-terms similar to (6), but with equality instead of as an upper bound.

Proposition 2. Let Σ be the noise covariance matrix, and let $J_2(\alpha)$ be the noiseless MSE loss of a regression

ensemble aggregated with coefficients $\alpha \in \mathbb{R}^T$. Then, for any $\alpha \in \mathbb{R}^T$,

$$\tilde{J}_2(\alpha) = J_2(\alpha) + \alpha^\top \Sigma \alpha. \quad (10)$$

Proof. Recalling that $\mathbb{E}[\mathbf{n}] = \mathbf{0}$, the loss decomposes as follows

$$\begin{aligned} \tilde{J}_2(\alpha) &= \mathbb{E}_{\mathbf{n}} \left[\frac{1}{N_s} \sum_{i=1}^{N_s} |y_i - \tilde{f}(\mathbf{x}_i)|^2 \right] \\ &= \frac{1}{N_s} \sum_{i=1}^{N_s} (y_i - \alpha^\top \boldsymbol{\varphi}(\mathbf{x}_i))^2 + \mathbb{E}_{\mathbf{n}} \left[(\alpha^\top \mathbf{n})^2 \right]. \end{aligned} \quad (11)$$

The first term equals $J_2(\alpha)$ by definition, and the second term equals

$$\alpha^\top \mathbb{E}_{\mathbf{n}} [\mathbf{n} \mathbf{n}^\top] \alpha = \alpha^\top \Sigma \alpha.$$

□

Theorem 1 and Proposition 2 manifest an inherent trade-off between model error (the first terms in (6), (10)) and aggregated noise (the second terms). This motivates the refinement of Problem 1 to allow tuning of the contribution of each loss component. By introducing a coefficient $\lambda \geq 0$ that weights the aggregated-noise component, we get the following problem that optimizes a more general variant of the MSE loss.

Problem 1’. Given a dataset $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$, an ensemble of base regressors $\{\varphi_t(\cdot)\}_{t=1}^T$ and a zero-mean noise vector with covariance matrix Σ , find the coefficients $\alpha = (\alpha_1, \dots, \alpha_T)^\top$ that minimize

$$\tilde{J}_2^{(\lambda)}(\alpha) = J_2(\alpha) + \lambda \cdot \alpha^\top \Sigma \alpha. \quad (12)$$

We refer to solving Problem 1’ as the *trade-off ensemble method* (TEM). Note that (12) may be cast back to Problem 1 (for MSE loss) by setting $\lambda = 1$. In practice, *bounding* the aggregated-noise component may be more convenient than weighting it. We therefore re-formulate the TEM to solve the following equivalent constrained optimization problem

$$\min_{\alpha \in \mathbb{R}^T} J_2(\alpha) \text{ subject to } \alpha^\top \Sigma \alpha \leq C. \quad (13)$$

(13) considers the minimization of the “standard” (noiseless) model error while bounding the contribution of the aggregated noise to the overall loss term. This formulation gives precedence to minimizing the model-error term, while ensuring that the error from noise is kept sufficiently small (e.g., within the noise tolerance of the regression task). The advantage of this approach is that making the noise a secondary optimization criterion leads to better robustness to imperfect knowledge of the noise parameters. We now

proceed to solving this problem. The following characterizes the solution of (13) as a least-squares problem with a quadratic constraint (see [27]); afterwards we detail a method to solve the problem efficiently using an approximation.

Towards deriving the solution of (13), we define the prediction matrix of dimensions $N_s \times T$.

Definition 4. Given an ensemble of base regressors $\{\varphi_t(\cdot)\}_{t=1}^T$ and a set of inputs $\{\mathbf{x}_i\}_{i=1}^{N_s}$, define the prediction matrix as

$$\Phi \triangleq [\boldsymbol{\varphi}(\mathbf{x}_1), \dots, \boldsymbol{\varphi}(\mathbf{x}_{N_s})]^\top. \quad (14)$$

Theorem 3. Let Σ be the noise covariance matrix and let Φ be a rank- T prediction matrix of the regression ensemble. The optimal solution of the optimization problem in (13) is

$$\alpha^* = (\Phi^\top \Phi + \lambda N_s \Sigma)^{-1} \Phi^\top \mathbf{y}, \quad (15)$$

where $\mathbf{y} \triangleq (y_1, \dots, y_{N_s})^\top$ and λ is either 0 or given by equating C to

$$\mathbf{y}^\top \Phi (\Phi^\top \Phi + \lambda N_s \Sigma)^{-1} \Sigma (\Phi^\top \Phi + \lambda N_s \Sigma)^{-1} \Phi^\top \mathbf{y}. \quad (16)$$

Proof. This proof is similar to a proof given for a related optimization problem in [27]. Since the minimized function $J_2(\alpha) = \frac{1}{N_s} \|\Phi \alpha - \mathbf{y}\|_2^2$ and the inequality constraint function $\alpha^\top \Sigma \alpha - C$ are both convex (as quadratic forms), we can use the Karush-Kuhn-Tucker (KKT) conditions to find the optimal α . Since $\nabla J_2(\alpha) = \frac{2}{N_s} \Phi^\top (\Phi \alpha - \mathbf{y})$ and $\nabla (\alpha^\top \Sigma \alpha - C) = 2 \Sigma \alpha$, the KKT conditions are

$$\begin{aligned} (\alpha^*)^\top \Sigma \alpha^* - C &\leq 0 \\ \lambda &\geq 0 \\ \lambda ((\alpha^*)^\top \Sigma \alpha^* - C) &= 0 \end{aligned} \quad (17)$$

$$\frac{2}{N_s} \Phi^\top (\Phi \alpha^* - \mathbf{y}) + 2 \lambda \Sigma \alpha^* = \mathbf{0}$$

According to the last condition, $\alpha^* = (\Phi^\top \Phi + \lambda N_s \Sigma)^{-1} \Phi^\top \mathbf{y}$. Now, from the third condition either $\lambda = 0$ or $(\alpha^*)^\top \Sigma \alpha^* = C$. Hence, whenever the pseudo-inverse solution $\alpha^* = (\Phi^\top \Phi)^{-1} \Phi^\top \mathbf{y}$ satisfies the constraint $(\alpha^*)^\top \Sigma \alpha^* \leq C$, set $\lambda = 0$. When the constraint is not satisfied by the pseudo-inverse solution, it follows that $(\alpha^*)^\top \Sigma \alpha^* = C$, which gives (16) when α^* is substituted. □

Theorem 3 provides a closed-form expression for the optimal coefficients solving (13) when λ is known, in particular for $\lambda = 1$ corresponding to the standard MSE. However, given C the solution depends on λ , which is given only implicitly by (16). Although λ can be calculated by complex numerical methods, we

now suggest an alternative approach that allows to calculate λ in closed form after approximating the matrix inverse in (16) using the Neumann series.

Lemma 4. Let λ be a solution of the equation

$$\mathbf{y}^\top \Phi \mathbf{Z}^\top \Sigma \mathbf{Z} \Phi^\top \mathbf{y} = C, \quad (18)$$

where $\mathbf{Z} \triangleq (\mathbf{I} + \lambda (\Phi^\top \Phi)^{-1} \Sigma) (\Phi^\top \Phi)^{-1}$. Then λ can be found as a root of the second-order polynomial

$$a\lambda^2 + b\lambda + c, \quad (19)$$

where the coefficients are

$$\begin{aligned} a &= \mathbf{y}^\top \Phi \Sigma^\top (\Phi^\top \Phi)^{-1} \Sigma (\Phi^\top \Phi)^{-1} \Sigma \Phi^\top \mathbf{y} \\ b &= \mathbf{y}^\top \Phi \Sigma^\top \left((\Phi^\top \Phi)^{-1} + (\Phi^\top \Phi)^{-1} \right) \Sigma \Phi^\top \mathbf{y} \\ c &= \mathbf{y}^\top \Phi \Sigma \Phi^\top \mathbf{y} - C. \end{aligned} \quad (20)$$

We omit the proof of Lemma 4 since it can be verified by expanding (18) followed by straightforward rearrangements. Since \mathbf{Z} in Lemma 4 is a well-known approximation of $(\Phi^\top \Phi + \lambda \Sigma)^{-1}$ (see [28], Theorems 19.16(a), (b), (c)), finding λ by solving the quadratic polynomial (19) will give an approximation of the optimal value.

B. Robust bagging for MAE loss

In this subsection, we move to consider the mean absolute error (MAE) loss in the noisy regression problem. The expected MAE loss with respect to a certain dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$ is given by

$$\begin{aligned} \tilde{J}_1(\alpha) &= \mathbb{E}_{\mathbf{n}} \left[\frac{1}{N_s} \sum_{i=1}^{N_s} |\tilde{f}(\mathbf{x}_i) - y_i| \right] \\ &= \mathbb{E}_{\mathbf{n}} \left[\frac{1}{N_s} \sum_{i=1}^{N_s} |\alpha^\top \varphi(\mathbf{x}_i) - y_i + \alpha^\top \mathbf{n}| \right]. \end{aligned} \quad (21)$$

Note that specifically for $p = 1$ we have $\tilde{J}_1(\alpha) = \tilde{J}_{\ell_1}(\alpha)$ so the upper bound from Theorem 1 holds. For $p = 1$ the loss does not decompose with equality like the MSE in (10), but for normally distributed noise it can still be written in closed form.

Theorem 5. Let $\alpha \in \mathbb{R}^T$ be a coefficient vector and let the noise random vector be $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \Sigma)$. The expected MAE loss is

$$\begin{aligned} \tilde{J}_1(\alpha) &= \frac{1}{N_s} \sum_{i=1}^{N_s} \left[\sqrt{\frac{2}{\pi}} \sigma(\alpha) e^{-\frac{\mu_i^2(\alpha)}{2\sigma^2(\alpha)}} \right. \\ &\quad \left. + \mu_i(\alpha) \left(2\phi\left(\frac{\mu_i(\alpha)}{\sigma(\alpha)}\right) - 1 \right) \right], \end{aligned} \quad (22)$$

where $\mu_i(\alpha) \triangleq \alpha^\top \varphi(\mathbf{x}_i) - y_i$, $\sigma(\alpha) \triangleq \sqrt{\alpha^\top \Sigma \alpha}$ and $\phi(t)$ is the standard normal cumulative distribution function.

Proof. Recall that

$$\begin{aligned} \tilde{J}_1(\alpha) &= \mathbb{E}_{\mathbf{n}} \left[\frac{1}{N_s} \sum_{i=1}^{N_s} |\alpha^\top \mathbf{n} + \alpha^\top \varphi(\mathbf{x}_i) - y_i| \right] \\ &= \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} [|\alpha^\top \mathbf{n} + \alpha^\top \varphi(\mathbf{x}_i) - y_i|], \end{aligned} \quad (23)$$

where \mathbf{n} is $\mathcal{N}(\mathbf{0}, \Sigma)$. Hence, the expression inside the absolute value in (23) is a normal random vector with expectation $\mu_i(\alpha) = \alpha^\top \varphi(\mathbf{x}_i) - y_i$ and variance $\sigma^2(\alpha) = \alpha^\top \Sigma \alpha$. Therefore, the argument of the expectation in (23) is a *folded normal* random variable, whose expectation is known [29] to be the expression in the summation argument in (22). \square

Since $\tilde{J}_1(\alpha)$ does not facilitate a closed-form solution for the optimal α , we propose to optimize α iteratively via gradient-based methods. Toward that, we now calculate the gradient of $\tilde{J}_1(\alpha)$ with respect to α . For compactness of expressions, we denote the scalar $\rho_i(\alpha) \triangleq \frac{\mu_i(\alpha)}{\sigma(\alpha)}$, and its gradient with respect to α is denoted $\rho'_i(\alpha)$. The gradients of $\mu_i(\alpha)$ and $\sigma(\alpha)$ with respect to α are denoted $\mu'_i(\alpha)$ and $\sigma'(\alpha)$, respectively; the expressions for these gradients are derived as $\mu'_i(\alpha) = \varphi(\mathbf{x}_i)$, $\sigma'(\alpha) = \Sigma \alpha / \sqrt{\alpha^\top \Sigma \alpha}$ and

$$\rho'_i(\alpha) = \frac{\sigma(\alpha) \varphi(\mathbf{x}_i) - \mu_i(\alpha) \sigma'(\alpha)}{\sigma^2(\alpha)}. \quad (24)$$

Recall that the probability density function and the cumulative distribution function of the standard normal random variable are denoted $g(\cdot)$ and $\phi(\cdot)$, respectively. Then,

Proposition 6. The gradient of $\tilde{J}_1(\alpha)$ from Theorem 5 with respect to α is

$$\begin{aligned} \frac{1}{N_s} \sum_{i=1}^{N_s} \left[\sqrt{\frac{2}{\pi}} e^{-\frac{\rho_i^2(\alpha)}{2}} (\sigma'(\alpha) - \sigma(\alpha) \rho_i(\alpha) \rho'_i(\alpha)) \right. \\ \left. + \mu'_i(\alpha) [2\phi(\rho_i(\alpha)) - 1] + 2\mu_i(\alpha) \rho'_i(\alpha) g(\rho_i(\alpha)) \right]. \end{aligned} \quad (25)$$

Proof. The expression for $\tilde{J}_1(\alpha)$ in (22) comprises a sum over two terms. The gradient of the first term is (omitting the constant $\sqrt{2/\pi}$)

$$e^{-\frac{\rho_i^2(\alpha)}{2}} \sigma'(\alpha) - \sigma(\alpha) e^{-\frac{\rho_i^2(\alpha)}{2}} \rho_i(\alpha) \rho'_i(\alpha). \quad (26)$$

The gradient of the second term is

$$\mu'_i(\alpha) [1 - 2\phi(-\rho_i(\alpha))] + 2\mu_i(\alpha) g(-\rho_i(\alpha)) \rho'_i(\alpha). \quad (27)$$

Note that we used the fact that the derivative of $\phi(\cdot)$ is the normal density function. Finally, we use $g(t) = g(-t)$ and $\phi(-t) = 1 - \phi(t)$ to obtain (25). \square

The following Algorithm 1 gives a gradient-descent procedure to minimize $\tilde{J}_p(\alpha)$. Substituting (25) from Proposition 6 as $\nabla \tilde{J}_1(\alpha)$ in line 9 allows using the algorithm for the MAE ($p = 1$), which is done in the experimental study presented in Section V. To allow comparison with non-robust aggregation, we use as baseline Algorithm 1 with the noise-free loss $J_1(\alpha)$ replacing $\tilde{J}_1(\alpha)$, resulting in the noise-free gradient

$$\nabla J_1(\alpha) = \frac{1}{N_s} \sum_{i=1}^{N_s} \varphi(x_i) \text{sign}(\alpha^\top \varphi(x_i) - y_i). \quad (28)$$

Algorithm 1 Gradient-descent minimization of $\tilde{J}_p(\alpha)$

- 1: **Input:** $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$, $\{\varphi_t(\cdot)\}_{t=1}^T$, Σ - training dataset, trained base classifiers and noise covariance matrix
 - 2: **Output:** $\{\alpha_t\}_{t=1}^T$ - optimized aggregation coefficients.
 - 3: **Set:** i_{\min}, i_{\max} (# iter.), η (learn rate), γ (momentum), τ (tolerance) and ϵ (regularization)
 - 4: **Initialize:** $\alpha^{(1)} \leftarrow \frac{1}{T} \mathbf{1}$, $\delta_\alpha^{(-1)} \leftarrow \mathbf{0}$ and $i \leftarrow 0$
 - 5: **while** $i \leq i_{\max}$ **do**
 - 6: **if** $|\tilde{J}_p(\alpha^{(i)}) - \tilde{J}_p(\alpha^{(i-1)})| \leq \tau$ and $i \geq i_{\min}$ **then**
 - 7: **break**
 - 8: **else**
 - 9: $\delta_\alpha^{(i)} \leftarrow \gamma \cdot \delta_\alpha^{(i-1)} - \eta \cdot \frac{\nabla \tilde{J}_p(\alpha^{(i-1)})}{\sqrt{\sum_{j \leq i} (\nabla \tilde{J}_p(\alpha^{(j)}))^2 + \epsilon}}$
 - 10: $\alpha^{(i+1)} \leftarrow \alpha^{(i)} + \delta_\alpha^{(i)}$
 - 11: $i \leftarrow i + 1$
 - 12: **end if**
 - 13: **end while**
 - 14: $i^* \leftarrow \arg \min_{0 \leq j \leq i} \tilde{J}_p(\alpha^{(j)})$
 - 15: **return** $\alpha^{(i^*)}$
-

C. Performance bounds for robust MAE

Algorithm 1 is useful for optimizing the MAE in practice, but we now derive performance bounds on the MAE towards gaining insight on this optimization. These bounds are also a useful tool for evaluating the expected loss without actually optimizing it. We derive lower and upper bounds on the optimal MAE loss $\tilde{J}_1(\alpha^*)$, where α^* is a coefficient vector that minimizes the MAE.

Specifically, the upper bound depends on the performance of *noiseless* regression and the noise covariance matrix, and does not require to perform calculations

that involve the entire dataset. In the following Proposition 7, we bound the MAE loss of noisy regression by the sum of the noiseless regression's MAE and a noise-related term.

Proposition 7. *Let $\alpha^* \in \mathbb{R}^T$ be a coefficient vector that minimizes $\tilde{J}_1(\alpha)$ from Theorem 5 and let Σ be a positive-definite covariance matrix of the normally distributed noise vector. Then, for any $\alpha \in \mathbb{R}^T$,*

$$\tilde{J}_1(\alpha^*) \leq J_1(\alpha) + \sqrt{\frac{2}{\pi}} \alpha^\top \Sigma \alpha. \quad (29)$$

Proof. Since α^* minimizes $\tilde{J}_1(\cdot)$, we have the first inequality in

$$\tilde{J}_1(\alpha^*) \leq \tilde{J}_1(\alpha) \leq J_1(\alpha) + \mathbb{E}[\|\alpha^\top \mathbf{n}\|], \quad (30)$$

and the second inequality is due to the bound in Theorem 1. Since $\alpha^\top \mathbf{n}$ is a normal random variable with mean 0 and variance $\alpha^\top \Sigma \alpha$, then from properties of folded normal random variables, $\mathbb{E}[\|\alpha^\top \mathbf{n}\|] = \sqrt{\frac{2}{\pi}} \alpha^\top \Sigma \alpha$. \square

The implication of Proposition 7 is that any $\alpha \in \mathbb{R}^T$ that gives low loss in the noiseless setting can be used with noise, with its degradation bounded by the second term of (29). The bound of Proposition 7 is specialized below to two particular choices of α : 1) the BEM vector [26], and 2) the vector that minimizes the second term of (29).

For BEM we have $\alpha = \frac{1}{T} \mathbf{1}$, hence

$$\tilde{J}_1(\alpha^*) \leq J_1\left(\frac{1}{T} \mathbf{1}\right) + \frac{1}{T} \sqrt{\frac{2}{\pi}} S(\Sigma), \quad (31)$$

where $S(\Sigma)$ denotes the sum of elements in Σ . A possibly tighter upper bound is obtained by selecting α to be the normalized eigenvector of Σ that corresponds to the minimal eigenvalue. In this case,

$$\tilde{J}_1(\alpha^*) \leq J_1\left(\frac{\mathbf{v}^{(\Sigma)}}{\mathbf{1}^\top \mathbf{v}^{(\Sigma)}}\right) + \sqrt{\frac{2}{\pi}} \sigma_{\min}^2, \quad (32)$$

where σ_{\min}^2 is the minimal generalized eigenvalue of the pair Σ and $\mathbf{1}\mathbf{1}^\top$, and $\mathbf{v}^{(\Sigma)}$ is the corresponding eigenvector (i.e., σ_{\min}^2 is the minimal real number ω and $\mathbf{v}^{(\Sigma)}$ is a vector \mathbf{v} that together solve the equation $\Sigma \mathbf{v} = \omega \mathbf{1}\mathbf{1}^\top \mathbf{v}$). The motivation to consider normalized α vectors in (31), (32) is that normalization guarantees that if the individual base regressors are unbiased estimates of $f(\cdot)$, so is the aggregated ensemble regressor.

To complement the performance analysis, we also give a lower bound on the MAE. We use the normalization assumption, $\mathbf{1}^\top \alpha = 1$, to obtain a universal lower bound (independent of α) in Proposition 8.

Proposition 8. Let Σ be a positive-definite covariance matrix of the normally distributed noise vector, and let α^* be a normalized coefficient vector with minimal MAE loss $\tilde{J}_1(\cdot)$. Then,

$$\tilde{J}_1(\alpha^*) \geq J_1(\alpha^\dagger) + \frac{1}{N_s} \sum_{i=1}^{N_s} \bar{\Delta}_i \exp\left(-\text{sign}'(\bar{\Delta}_i) \frac{\bar{\mu}_i^2}{2\bar{\sigma}^2}\right) \quad (33)$$

where α^\dagger is the optimal noiseless coefficient vector, $\bar{\mu}_i \triangleq \max_{t=1,\dots,T} \{|\varphi_t(\mathbf{x}_i) - y_i|\}$, $\bar{\sigma}$ is the minimal generalized eigenvalue of the pair Σ and $\mathbf{1}\mathbf{1}^\top$, $\bar{\Delta}_i \triangleq \sqrt{2/\pi} \bar{\sigma} - \bar{\mu}_i$ and $\text{sign}'(z) = 1$ for $z \geq 0$ and 0 otherwise.

Proof. We start by modifying the MAE using some well-known identities related to the error function, $\text{erf}(z)$. We use compressed notation: μ_i for $\mu_i(\alpha^*)$ and σ for $\sigma(\alpha^*)$.

$$\begin{aligned} \tilde{J}_1(\alpha^*) &= \frac{1}{N_s} \sum_{i=1}^{N_s} \left[\sqrt{\frac{2}{\pi}} \sigma e^{-\frac{\mu_i^2}{2\sigma^2}} + \mu_i \left(1 - 2\phi\left(-\frac{\mu_i}{\sigma}\right)\right) \right] \\ &= \frac{\sigma}{N_s} \sum_{i=1}^{N_s} \left[\sqrt{\frac{2}{\pi}} e^{-\frac{\mu_i^2}{2\sigma^2}} + \frac{|\mu_i|}{\sigma} \left(1 - \text{erfc}\left(\frac{|\mu_i|}{\sqrt{2}\sigma}\right)\right) \right]. \end{aligned} \quad (34)$$

The first equality is from (22), and the second transition is due to the following identities: $1 - 2\phi(z) = \text{erf}(z/\sqrt{2})$, $\text{erf}(z) = -\text{erf}(-z)$, $z \cdot \text{erf}(z) = |z| \cdot \text{erf}(|z|)$ and finally $\text{erfc}(z) \triangleq 1 - \text{erf}(z)$.

The last expression in the previous equation can be lower bounded using $\text{erfc}(z) \leq \exp(-z^2)$ for $z > 0$, giving

$$\begin{aligned} \tilde{J}_1(\alpha^*) &\geq \frac{\sigma}{N_s} \sum_{i=1}^{N_s} \left[\sqrt{\frac{2}{\pi}} e^{-\frac{\mu_i^2}{2\sigma^2}} + \frac{|\mu_i|}{\sigma} \left(1 - e^{-\frac{|\mu_i|^2}{2\sigma^2}}\right) \right] \\ &= \frac{1}{N_s} \sum_{i=1}^{N_s} \Delta_i e^{-\frac{\mu_i^2}{2\sigma^2}} + \frac{1}{N_s} \sum_{i=1}^{N_s} |\mu_i|, \end{aligned} \quad (35)$$

where $\Delta_i \triangleq \sqrt{\frac{2}{\pi}} \sigma - |\mu_i|$. The last expression in (35) comprises two sums, the second of which is $J_1(\alpha^*)$, by definition. Clearly, the optimal noiseless coefficients α^\dagger satisfy $J_1(\alpha^*) \geq J_1(\alpha^\dagger)$, yielding

$$\tilde{J}_1(\alpha^*) \geq J_1(\alpha^\dagger) + \frac{1}{N_s} \sum_{i=1}^{N_s} \Delta_i \exp(-\mu_i^2/2\sigma^2). \quad (36)$$

The bound in (36) still depends on α^* through μ_i and σ . Towards avoiding this dependence, we define $\bar{\mu}_i \triangleq \max_{t=1,\dots,T} \{|\varphi_t(\mathbf{x}_i) - y_i|\}$ and

$$\bar{\sigma} \triangleq \min_{\alpha \in \mathbb{R}^T: \mathbf{1}^\top \alpha = 1} \sqrt{\alpha^\top \Sigma \alpha} = \sqrt{\min_{\alpha \in \mathbb{R}^T: \mathbf{1}^\top \alpha = 1} \alpha^\top \Sigma \alpha},$$

which is known to be equivalent to the square root of the minimal generalized eigenvalue of the pair Σ and

$\mathbf{1}\mathbf{1}^\top$. For normalized coefficients, i.e., $\mathbf{1}^\top \alpha = 1$, we can now further bound (36) as

$$\tilde{J}_1(\alpha^*) \geq J_1(\alpha^\dagger) + \frac{1}{N_s} \sum_{i=1}^{N_s} \bar{\Delta}_i \exp(-\mu_i^2/2\sigma^2), \quad (37)$$

where $\bar{\Delta}_i \triangleq \sqrt{\frac{2}{\pi}} \bar{\sigma} - \bar{\mu}_i$. The bound holds for normalized coefficients $\alpha^* = \alpha$ since

$$\begin{aligned} |\mu_i| &= |\alpha^\top \varphi(\mathbf{x}_i) - y_i| \leq \alpha^\top |\varphi(\mathbf{x}_i) - y_i \mathbf{1}| \\ &\leq \max_{t=1,\dots,T} |\varphi_t(\mathbf{x}_i) - y_i|. \end{aligned} \quad (38)$$

Since $\bar{\sigma} \leq \sigma$, note that the i^{th} summand in the sum on the right-hand side of (36) is lower bounded by $\bar{\Delta}_i \exp(-\frac{\mu_i^2}{2\bar{\sigma}^2})$ when $\bar{\Delta}_i \geq 0$; while for $\bar{\Delta}_i < 0$ it is bounded simply by $\bar{\Delta}_i$. We can therefore express the lower bound as

$$\tilde{J}_1(\alpha^*) \geq J_1(\alpha^\dagger) + \frac{1}{N_s} \sum_{i=1}^{N_s} \bar{\Delta}_i \exp\left(-\text{sign}'(\bar{\Delta}_i) \frac{\bar{\mu}_i^2}{2\bar{\sigma}^2}\right), \quad (39)$$

where the sign' function is defined as $\text{sign}'(z) = 1$ for $z \geq 0$ and $\text{sign}'(z) = 0$ otherwise. \square

The lower bound in Proposition 8 is useful when the values of $\bar{\Delta}_i$ are mostly positive, that is, when the loss is noise-dominated (observe that $\bar{\Delta}_i$ is a difference between a noise term and a model-error term). When the sum in (33) is dominated by negative $\bar{\Delta}_i$ s, the bound may become loose, and we can instead use the following simpler bound.

Proposition 9. Let α^* be a coefficient vector that minimizes the MAE loss $\tilde{J}_1(\cdot)$. Then,

$$\tilde{J}_1(\alpha^*) \geq J_1(\alpha^\dagger), \quad (40)$$

where α^\dagger is the optimal noiseless coefficient vector and $J_1(\cdot)$ is the noiseless MAE loss.

Proof. First, define $\chi_i(\alpha) \triangleq \alpha^\top \mathbf{n} + \alpha^\top \varphi(\mathbf{x}_i) - y_i$. Then, $\chi_i(\alpha)$ is a Gaussian random variable with mean $\mu_i(\alpha) = \alpha^\top \varphi(\mathbf{x}_i) - y_i$. Now, note that any random variable V satisfies $\mathbb{E}[|V|] \geq \mathbb{E}[V]$. Hence the first inequality in

$$\begin{aligned} \tilde{J}_1(\alpha^*) &= \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} [|\chi_i(\alpha^*)|] \geq \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} [\chi_i(\alpha^*)] \\ &= \frac{1}{N_s} \sum_{i=1}^{N_s} |\mu_i(\alpha^*)| = J_1(\alpha^*) \geq J_1(\alpha^\dagger), \end{aligned} \quad (41)$$

where α^\dagger is the optimal noiseless coefficient vector. The equalities are by definitions, and the last inequality holds since α^\dagger minimizes $J_1(\cdot)$. \square

IV. ROBUST GRADIENT BOOSTING

The ensemble bagging approach, studied in Section III, optimizes the aggregation coefficients of an already-trained ensemble of base regressors. A potentially stronger approach considers the existence of noise and its parameters during the training of the ensemble's base regressors themselves. Assuming knowledge of the noise statistics during training, we devise in this section a noise-mitigating training procedure based on the popular *gradient boosting* (GB) method [24], [25], [30].

In gradient boosting, base regressors are trained sequentially: every iteration calculates an ensemble regressor $\hat{f}_t(\cdot) = \sum_{\tau=1}^t \alpha_\tau \varphi_\tau(\cdot)$ by adding a base regressor $\varphi_t(\cdot)$ with its coefficient α_t . $\varphi_t(\cdot)$ is trained based on $\hat{f}_{t-1}(\cdot)$ from the previous iteration, where in GB the particular objective is that $\varphi_t(\cdot)$ will approximate the (negative) *gradient* of the loss with respect to $\hat{f}_{t-1}(\cdot)$, such that it can achieve a steep reduction in the loss. Once $\varphi_t(\cdot)$ is trained, its aggregation coefficient α_t is chosen such that the newly formed prediction $\hat{f}_t(\cdot)$ minimizes the loss calculated over the training dataset.

We devise a *robust* variant of gradient boosting, in which every base regressor is added into the ensemble with an aggregation coefficient that optimizes a *noise-informed* loss term. For training the base regressors, we use the standard-GB gradient estimation (without noise), such that the new variant can still benefit from the numerous algorithms available in the literature for training base-regressor models.

A general version of the proposed algorithm, presented specifically for the $J_p(\alpha)$ loss, is described in Algorithm 2. The procedure starts by initializing the first base regressor to a constant unity function $\varphi_1(\cdot) = 1$, and setting its corresponding coefficient α_1 such that the loss over the training dataset is minimized, *in expectation over the noise* (Line 3). The t^{th} base regressor ($t > 1$) is trained, as usual in the GB method, to approximate the negative gradient of the loss with respect to $\hat{f}_{t-1}(\cdot)$, which is denoted $\gamma_t^{(p)}$ for the loss function J_p (Line 5). For this gradient approximation, the objective is quadratic error, regardless of the specified loss J_p (Line 6). In case of the MSE loss (J_2), we get that $\gamma_t^{(2)}(\mathbf{x}_i) = 2(y_i - \hat{f}_{t-1}(\mathbf{x}_i))$; for the MAE loss (J_1) we get $\gamma_t^{(1)}(\mathbf{x}_i) = \text{sign}(y_i - \hat{f}_{t-1}(\mathbf{x}_i))$. Finally, the coefficient α_t is set to minimize the expected loss (over the noise) of this iteration's regressor $\tilde{f}_t(\cdot)$ (Line 7).

We now derive the optimal aggregation coefficients in (42) and (44) for MSE loss.

Algorithm 2 Robust Gradient Boosting for noisy regression with $\tilde{J}_p(\cdot)$ loss

- 1: **Input:** $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$ - training dataset.
- 2: **Output:** $\{\varphi_t(\cdot)\}_{t=1}^T, \{\alpha_t\}_{t=1}^T$ - trained base regressors and aggregation coefficients.
- 3: Initialize $\tilde{f}_1(\cdot) \triangleq \alpha_1 \varphi_1(\cdot)$ where $\varphi_1(\cdot) \equiv 1$ and

$$\alpha_1 = \arg \min_{\alpha \in \mathbb{R}} \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} [|y_i - \alpha \tilde{\varphi}_1(\mathbf{x}_i)|^p] \quad (42)$$

- 4: **for** $t = 2$ to T **do**
- 5: Calculate the loss negative gradient $\gamma_t^{(p)}(\cdot) = -\frac{\partial J_p(\hat{f}_{t-1}(\cdot))}{\partial \hat{f}_{t-1}(\cdot)}$
- 6: Train base regressor $\varphi_t(\cdot)$ to minimize the gradient approximation error

$$\frac{1}{N_s} \sum_{i=1}^{N_s} \left(\gamma_t^{(p)}(\mathbf{x}_i) - \varphi_t(\mathbf{x}_i) \right)^2 \quad (43)$$

- 7: Denote $\omega_i^{(t)} \triangleq y_i - \sum_{\tau=1}^{t-1} \alpha_\tau \tilde{\varphi}_\tau(\mathbf{x}_i)$ and set α_t as

$$\arg \min_{\alpha \in \mathbb{R}} \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} \left[\left| \omega_i^{(t)} - \alpha \tilde{\varphi}_t(\mathbf{x}_i) \right|^p \right] \quad (44)$$

- 8: Set $\tilde{f}_t(\cdot) = \sum_{\tau=1}^t \alpha_\tau \varphi_\tau(\cdot)$
 - 9: **end for**
 - 10: **return** $\{\varphi_t(\cdot)\}_{t=1}^T$ and $\{\alpha_t\}_{t=1}^T$
-

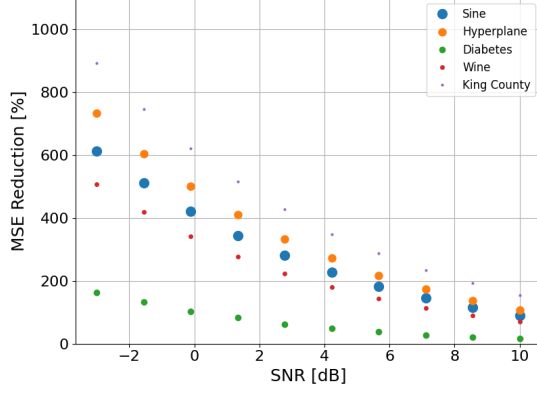
Proposition 10. Let $\alpha^* = (\alpha_1, \dots, \alpha_T) \in \mathbb{R}^T$ be the vector of aggregation coefficients obtained by Algorithm 2 for MSE loss ($p = 2$). Given that the noise random vector has mean $\mathbf{0}$ and covariance matrix Σ , the coefficients are given by $\alpha_1 = \frac{1}{(1+\sigma_1^2)N_s} \sum_{i=1}^{N_s} y_i$, and then for $t > 1$ recursively by

$$\alpha_t = \frac{\frac{1}{N_s} \sum_{i=1}^{N_s} \varphi_t(\mathbf{x}_i) \left(y_i - \hat{f}_{t-1}(\mathbf{x}_i) \right) + \sum_{\tau=1}^{t-1} \alpha_\tau \Sigma_{t,\tau}}{\sigma_t^2 + \frac{1}{N_s} \sum_{i=1}^{N_s} \varphi_t^2(\mathbf{x}_i)} \quad (45)$$

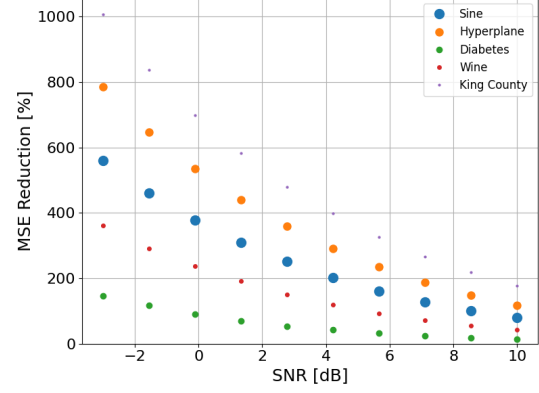
Proof. The first coefficient, α_1 , is set in (42) to minimize $\frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} [(y_i - \alpha(1 + n_1))^2]$. Since $\mathbb{E}_{\mathbf{n}} [n_1] = 0$ and $\mathbb{E}_{\mathbf{n}} [n_1^2] = \sigma_1^2$, the derivative with respect to α of the expectation becomes $\frac{2}{N_s} \sum_{i=1}^{N_s} (y_i - \alpha(1 + \sigma_1^2))$, and equating to zero gives the expression of α_1 .

At step $t > 1$, α_t minimizes in (44) the expectation

$$\frac{1}{N_s} \sum_{i=1}^{N_s} \mathbb{E}_{\mathbf{n}} \left[(y_i - \tilde{f}_{t-1}(\mathbf{x}_i) - \alpha (\varphi_t(\mathbf{x}_i) + n_t))^2 \right]. \quad (46)$$



(a) TEM, equi-variance noise.



(b) TEM, noisier-subset with $m = 2$, $a = 20$.

Fig. 3: MSE reduction of TEM bagging ensembles relative to the prior method GEM, as a function of SNR.

The expression above can be simplified using the facts $\mathbb{E}_{\mathbf{n}}[n_t] = 0$, $\mathbb{E}_{\mathbf{n}}[\tilde{f}_{t-1}(\mathbf{x}_i)] = \hat{f}_{t-1}(\mathbf{x}_i)$, and

$$\mathbb{E}_{\mathbf{n}}[\tilde{f}_{t-1}(\mathbf{x})n_t] = \mathbb{E}_{\mathbf{n}}\left[\sum_{\tau=1}^{t-1}\alpha_{\tau}n_{\tau}n_t\right] = \sum_{\tau=1}^{t-1}\alpha_{\tau}\Sigma_{t,\tau}. \quad (47)$$

Calculating the derivative of the simplified (46) with respect to α yields

$$-\frac{2}{N_s}\sum_{i=1}^{N_s}\varphi_t(\mathbf{x}_i)\left(y_i - \hat{f}_{t-1}(\mathbf{x}_i)\right) - 2\sum_{\tau=1}^{t-1}\alpha_{\tau}\Sigma_{t,\tau} + 2\alpha\left(\sigma_t^2 + \frac{1}{N_s}\sum_{i=1}^{N_s}\varphi_t^2(\mathbf{x}_i)\right). \quad (48)$$

Equating (48) to 0 finally gives (45). Since the MSE loss is a convex function, the obtained solution is the minimum. \square

Proposition 10 provides, for MSE loss ($p = 2$), closed-form solutions to the coefficient optimizations of Algorithm 2, and thus makes the algorithm simple to implement and fast to run. The same algorithm may be employed for MAE loss ($p = 1$) as well, while replacing in (44) the closed-form expressions for α_t with a search-based minimization algorithm. For example, the (scalar) derivative of the expected MAE can be derived in a similar way to Proposition 6, then employing it in a descent-based search of α_t .

V. EXPERIMENTAL RESULTS

In this section, we give an empirical evaluation of the advantages of the proposed robust methods in comparison to non-robust classical ensemble methods. We also show empirical results that improve our understanding of the proposed methods and the

considerations for their practical use. Toward these ends, we experimented with various datasets and several noise conditions¹. Besides the sinusoidal sum target function shown in Section II-B, we used another synthetic function and three real-world datasets:

- Hyperplane: a linear combination $f(\mathbf{x}) = \mathbf{c}^T \mathbf{x}$ where $\mathbf{x} \in \mathbb{R}^d$, $d = 3$ and \mathbf{c} is a vector of random real numbers.
- UCI diabetes patient records [32]: the dataset consists of $d = 10$ variables and $N_s = 442$ samples.
- White-wine quality [33]: the dataset consists of $d = 11$ variables and $N_s = 4898$ samples.
- King County house prices: the dataset consists of $d = 20$ variables and $N_s = 21613$ samples.

In the following experiments, each synthetic dataset comprised $N_s = 1000$ samples and was generated by summing the target function with independent and identically distributed random Gaussian measurement error $\epsilon \sim \mathcal{N}(0, 0.01)$ (not to be confused with the channel noise, which is the main object of interest). In addition, for convenience all datasets were standardized to have zero mean and unit standard deviation ($\epsilon_y = 1$).

During inference on each test subset, the individual predictions of the base regressors are added with noise before they are aggregated to produce the final prediction. For training and inference we followed the k -fold cross-validation procedure, according to which k equally-sized disjoint subsets of data samples were generated from the dataset. The following procedure is repeated for each such subset (fold): using the union of the complement $k - 1$ subsets, train an

¹Simulation code is available in [31].

ensemble and set the aggregation coefficients; then evaluate the prediction performance on the subset. The performance of each fold's ensemble is averaged over 100 noise realizations.

While this paper's results apply to general noise distributions, we focus in this section on two representative settings: a homogeneous setting in which all channels have equal noise variance; and a heterogeneous setting with two channel types, such that a part of the ensemble communicates through noisier channels compared to the remaining part. The corresponding noise covariance matrices are:

- *Equi-variance noise profile*: A diagonal matrix with equal non-zero elements, i.e., $\Sigma = \sigma^2 \mathbf{I}$ where \mathbf{I} is the $T \times T$ identity matrix and $\sigma^2 = \frac{\varepsilon_y}{\text{SNR}}$.
- *Noisier-subset noise profile*: Given a fraction $1/m$ of factor $a > 1$ noisier channels, a diagonal matrix whose non-zero elements are a permutation of the sequence $\underbrace{a\sigma^2, \dots, a\sigma^2}_{T/m \text{ times}}, \underbrace{\sigma^2, \dots, \sigma^2}_{T-T/m \text{ times}}$.

In this case, according to (4),

$$\sigma^2 = \frac{T \cdot \varepsilon_y}{\left(\frac{T}{m}a + \left(1 - \frac{1}{m}\right)T\right) \cdot \text{SNR}} = \frac{m \cdot \varepsilon_y}{(a + m - 1) \cdot \text{SNR}}. \quad (49)$$

Specifically, we used $m = 2$ in our simulations, hence the diagonal elements are $\sigma^2, a\sigma^2, \sigma^2, a\sigma^2, \dots$ and

$$\sigma^2 = \frac{\varepsilon_y}{\left(1 + \frac{a-1}{2}\right) \cdot \text{SNR}}. \quad (50)$$

A. Robust MSE-optimal bagging ensembles

We start with robust bagging ensembles optimized for MSE loss (Section III-A). Our focus in this part is the trade-off ensemble method (TEM), in particular, using Theorem 3 with $\lambda = 1$ to find the α that minimizes (12) in Problem 1'. We trained an ensemble of $T = 32$ base regressors, each of which is a decision tree (with varying depth, depending on the dataset). Per the bagging approach, each base regressor was trained on a random subset of the fold's training set. After obtaining the ensemble, we used (15) to calculate the MSE-optimal aggregation coefficients.

We measured the performance of the TEM coefficients by evaluating the robustness gain, i.e., the *MSE reduction* compared to the GEM coefficients in (9) (which do not take the noise into account). Formally, we measure

$$100 \cdot \frac{J_2(\alpha_{\text{GEM}}^\top \tilde{\varphi}(\cdot)) - J_2(\alpha_{\text{TEM}}^\top \tilde{\varphi}(\cdot))}{J_2(\alpha_{\text{GEM}}^\top \varphi(\cdot))} [\%], \quad (51)$$

where α_{GEM} and α_{TEM} are the GEM and TEM coefficients (respectively), and $\tilde{\varphi}$ denotes the noisy version of the bagging ensemble's base-regressor functions (note the noiseless φ in the denominator). The empirical expectations in (51) are calculated by averaging over both the multiple folds and the noise realizations. The results are plotted in Fig. 3 as a function of SNR, for all five datasets. It can be seen that for both noise profiles, TEM offers significant performance advantages: reaching MSE reductions of 200% to 1000% at lower SNRs, depending on the dataset. The advantage decreases as the SNR increases. The MSE achieved by TEM is only slightly degraded compared to a baseline noiseless ensemble (not shown in the figures). It may initially counter intuition that robust coefficients are needed even in the equi-variance case (where all channels are identical), but the empirical results show that this need in optimization is crucial. The intuitive explanation of this may be that differences in quality (in terms of model error) between regressors map to different coefficient assignments depending on the noise intensity.

Next in Fig. 4 and 5, we take a refined view into the TEM optimal solutions. In Fig. 4, we plot the distributions of the aggregation coefficients obtained for the two noise profiles tested in Fig. 3. Each plot in Fig. 4 is a histogram of the coefficients assigned to $T = 20$ regressors in the equi-variance (left) and noisier-subset (right) profiles for the sinusoidal sum dataset; each color/pattern represents a different fold. It can be observed that the equi-variance coefficients (left) are centered around the average of $0.05 (= 1/T)$, while in the noisier-subset case (right) the histogram is more spread out and bi-modal: the higher coefficients are for the regressors that have good channels, and those of the noisier channels are closer to zero.

Fig. 5 provides a deeper understanding of the relationship between the individual base-regressor's prediction quality (error and noise) and its coefficient assignment. The left bars (grey) describe an ensemble of $T = 20$ base regressors with a synthetic linearly-decreasing profile of model-error (top is best). The corresponding TEM aggregation coefficients are depicted by the right bars for three noise settings: noiseless (blue), weak noise (orange) and high noise (green). When there is no noise, the coefficients vary considerably depending (inversely proportional) on the regressor's model error. In contrast, when the noise level is highest, the coefficients are close to uniform, aiming to average out the noise. The weak noise exhibits coefficients that give a compromise between these two competing considerations.

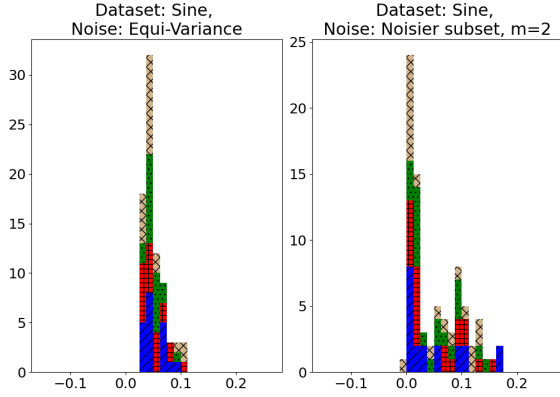


Fig. 4: Histogram of TEM (MSE-optimal) aggregation coefficients for the sinusoidal sum dataset with equi-variance noise (left) and $m = 2$ noisier-subset profile (right). The colors/patterns depict different folds of the training data.

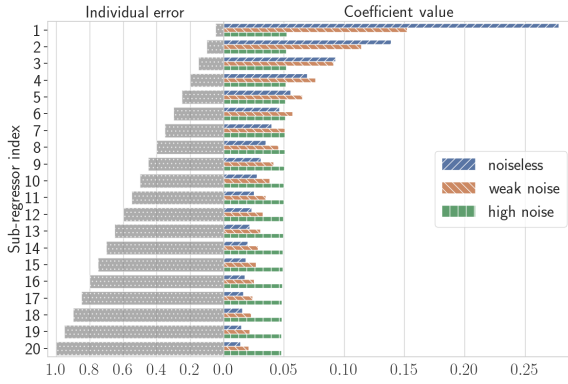


Fig. 5: TEM (MSE-optimal) aggregation coefficients (right) calculated with heterogeneous base-regressor model errors (left) for three equi-variance noise settings: noiseless (blue ‘/’), weak noise (orange ‘\’), and high noise (green ‘|’).

B. Performance under model uncertainty

To address uncertainties in the noise model, the TEM optimization in Section III-A includes the parameter λ that sets the relative importance of the channel noise in the total loss term $\tilde{J}_2^{(\lambda)}$. For example use of this parameter, consider the case where noise exists in the channels only intermittently at some fraction of the inference instances. Instead of attempting to solve a new optimization problem for each such case, we can invoke the optimization with a λ that best fits the actual noise behavior. In Fig. 6 we plot the MSE obtained by TEM when optimized for equi-variance noise with parameter σ , but tested with only *half* of

the inference instances having noise. The plot shows the average MSE (over both types of instances) as a function of λ , when $\text{SNR} = -6\text{dB}$ for the noisy instances. It can be seen that there is an intermediate value of $0 < \lambda < 1$ that minimizes the MSE, and that this value depends on the dataset. We can think of this optimal λ as measuring the sensitivity of the trained ensemble to the channel noise.

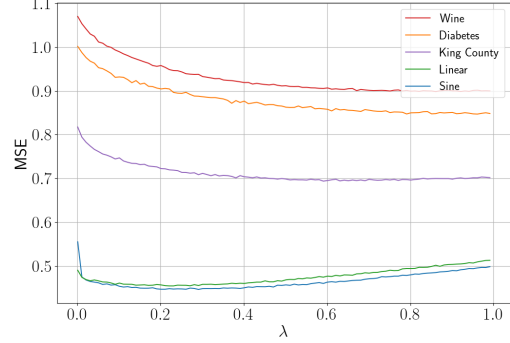


Fig. 6: MSE of TEM as a function of λ for partially noisy channels.

Additional variations in the model may arise from discrepancies between the actual noise distribution and the Gaussian noise assumed during optimization. To illustrate the robustness of our approach in the presence of such mismatches, Fig. 7 presents the MSE performance of coefficients optimized under a Gaussian-noise assumption but evaluated with Laplace-distributed noise with the same variance. As anticipated, a slight degradation in performance is observed. However, the coefficients continue to effectively mitigate the noise, even when the noise distribution deviates from the Gaussian assumption.

C. Robust MAE-optimized bagging ensembles

The focus in this sub-section moves to evaluating the performance of *MAE-optimized* robust ensembles (from Section III-B). To this end, we use Algorithm 1 with the gradient expression from Proposition 6 to MAE-optimize the coefficients of an ensemble of $T = 8$ decision-tree base regressors. In Fig. 8, we compare the MAE achieved by the MAE-robust coefficients relative to the non-robust coefficients calculated by Algorithm 1 modified to noise-free MAE with the gradient expression in (28). The results are shown for the five datasets and for both noise profiles. The best way to examine these results is starting from a point at the right end of the x-axis, and following the curves

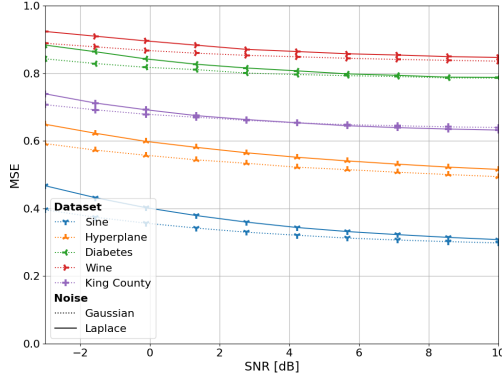


Fig. 7: MSE of TEM optimized for Gaussian noise and evaluated on Gaussian and Laplace noises in the equi-variance setting.

left-ward while the robust and non-robust diverge. It can be seen that significant gaps open between the curves, and that the robust MAE curves are much more flat in their increase as the SNR decreases. The plots reveal interesting dependencies on the dataset: for some (as Diabetes and Wine), the non-robust ensembles are more sensitive to noise than for others, for some (as Sine), the robust ensembles deal better with noisier-subset than with equi-variance.

In addition to seeing the actual MAE resulting from the proposed Algorithm 1, it is useful to examine the limits of the problem by plotting the MAE bounds derived in Section III-C. We present in Fig. 9 the lower and upper bounds on MAE for the five datasets, with the noisier-subset profile. The plotted lower bound is the maximum of the bounds in Propositions 8 and 9; while the plotted upper bound is the minimum of the two bounds induced by Proposition 7 in Eq. (31) and (32). Note that for the lower bounds we used a coefficient vector optimized for a noiseless setting ($\Sigma = \mathbf{0}$) as the optimal noiseless coefficient vector α^\dagger in (33) and (40). The bounds are tight for high SNR (essentially coinciding on the noiseless error); for a general SNR they together mark the range of MAE values that may be relevant for future optimization algorithms.

D. Robust training using gradient boosting

In this sub-section we evaluate the performance of robust gradient-boosting ensembles developed in Section IV. Without noise, gradient boosting has the advantage (in particular, over bagging) of regressors that adaptively refine the error of the prior iterations.

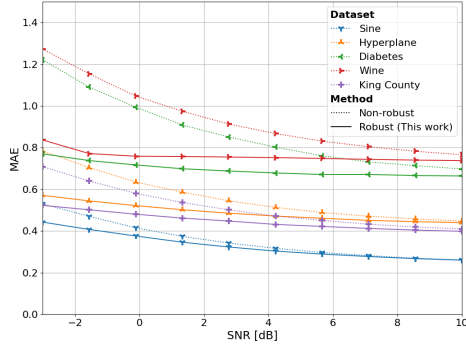
With noise, however, this advantage may turn into a disadvantage of increased sensitivity to noise in the more dominant regressors of the ensemble.

In the experiment, we pick two datasets, one natural and one synthetic, and train ensembles for them using Algorithm 2 for the MSE loss. Recall that part of the algorithm is setting the aggregation coefficients using Proposition 10. As a comparison baseline, we use the standard gradient-boosting (GB) algorithm that sets the aggregation coefficients to minimize the training error without considering the noise. We vary the ensemble size T , where each regressor in the ensemble is a depth-1 decision tree.

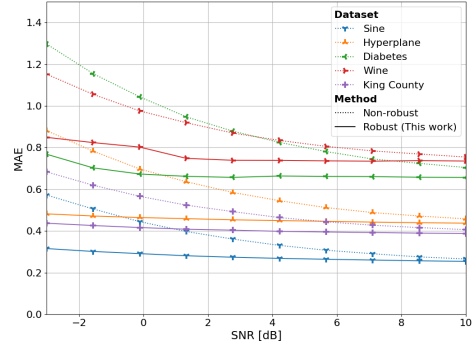
Fig. 10 depicts for SNR = 18dB the (root of the) MSE as a function of T for the Diabetes dataset (upper plots) and Sine dataset (lower plots). We show results for the two noise profiles: equi-variance (left) and noisier-subset (right). The first thing we notice in the plots is the behavior of the non-robust baseline algorithm: while without noise (Noiseless, black) the (root) MSE decreases as T grows, with noise (Noisy GB, red) the baseline exhibits the opposite trend of error *increasing* with T (!) In contrast, the Robust GB (blue) curve maintains the decreasing trend even with noise. A fourth curve (Noiseless Robust GB, purple) shows that the robust ensemble desirably performs better without noise than with noise. For the Diabetes dataset this performance is the same as the standard noiseless gradient boosting. These features underline the attractiveness of the robust ensembles: ensuring correct behavior in noisy settings without severely compromising the noiseless performance.

E. Complexity of coefficients optimization

We now examine the computational complexity associated with calculating the optimized aggregation coefficients for the methods proposed throughout this paper. It is important to note that using the optimized coefficients does *not* involve any increase in the complexity at inference time. So the following discussion only relates to the optimization stage, which needs to be performed only once per inference task, or whenever the noise statistics change. For optimization methods based on the MSE loss, closed-form algebraic expressions are available, allowing for precise complexity analysis. In Theorem 3, the computation of robust bagging coefficients requires $O(T^3 + T^2 N_s)$ operations. This includes the inversion of a $T \times T$ matrix and a subsequent matrix multiplication. In Proposition 10, the robust gradient-boosting coefficients require $O(T(T + N_s))$ operations.



(a) MAE-optimized, equi-variance noise.



(b) MAE-optimized, noisier subset with $m = 2$, $a = 20$.

Fig. 8: Comparison of MAE for robust (Algorithm 1) and non-robust bagging ensembles as a function of SNR.

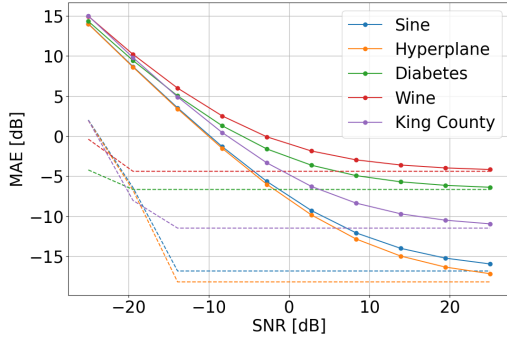


Fig. 9: Lower and upper bounds on the optimal MAE (in dB, as $10 \cdot \log_{10}(\cdot)$) as a function of SNR. Noise profile is “noisier subset” with $m = 2$, $a = 20$.

For the MAE loss, an iterative optimization algorithm (Algorithm 1) is employed. In the evaluations shown in this paper this algorithm runs up to 30,000 iterations, with each iteration involving a gradient computation (as detailed in Proposition 6), which incurs $O(T^2)$ operations per iteration.

VI. CONCLUSION

This work presents robust inference with regression ensembles operating over a noisy distributed setting. As a closed-form expression for MAE-optimal aggregation coefficients is unavailable, an immediate future-work direction is the derivation of simpler/tighter lower and upper MAE bounds toward easier optimization of the coefficients. In addition, we conjecture that tighter guarantees on the expected loss may also be obtained by assuming specific statistical properties of the noise.

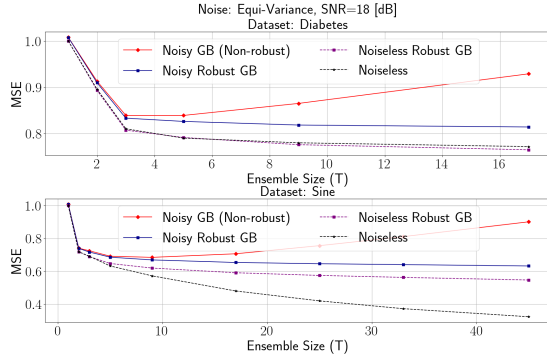
Another natural extension of this work is to address generalized aggregation schemes that consider non-scalar target functions $f(\mathbf{x})$ which appear in many real-world problems. In addition, this work can also be extended to additional performance criteria (e.g., $p = 0$, $p > 2$ or a linear combination of model error and aggregated noise in $p \neq 2$), additional noise regimes (e.g., quantization noise) and additional training, and aggregation methods (e.g., Stacking [34]).

ACKNOWLEDGMENT

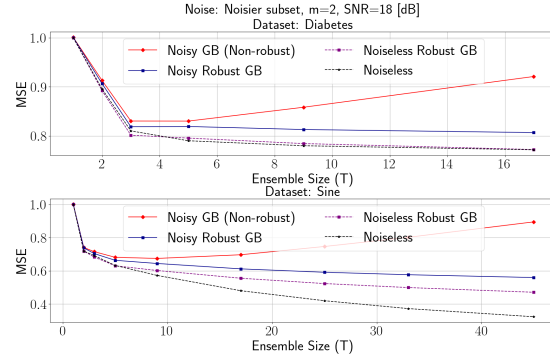
The authors wish to thank the JSAC reviewers and guest editor for valuable comments and suggestions that improved the paper. This work was supported in part by the US-Israel Binational Science Foundation under grant number 2023627.

REFERENCES

- [1] A. Gholami, “AI and memory wall,” medium.com/riselab, 2021, accessed: 23/12/2021. [Online]. Available: <https://medium.com/riselab/ai-and-memory-wall-2cb4265cb0b8>
- [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] 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.
- [4] 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, p. 3407, 2021.
- [5] Y. Chen, Y. Xie, L. Song, F. Chen, and T. Tang, “A survey of accelerator architectures for deep neural networks,” *Engineering*, vol. 6, no. 3, pp. 264–274, 2020.
- [6] O. Sagi and L. Rokach, “Ensemble learning: A survey,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.



(a) Equi-variance noise.



(b) Noisier subset with $m = 2$, $a = 20$.

Fig. 10: Comparison of (root) MSE obtained by robust and standard (“non-robust”) gradient boosting (GB) ensembles at SNR = 18dB as a function of the ensemble size T , for Diabetes (upper) and Sine (lower) datasets. The (root) MSEs of ensembles without noise (“Noiseless” and “Noiseless robust”) are shown for reference.

- [7] X. Qiu, L. Zhang, Y. Ren, P. Suganthan, and G. Amarutunga, “Ensemble deep learning for regression and time series forecasting,” in *Proc. 2014 IEEE Symp. on Computational Intelligence in Ensemble Learning*, 2014, pp. 1–6.
- [8] 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.
- [9] 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.
- [10] 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*, vol. 30, no. 3, pp. 291–302, 2022.
- [11] S. Teerapittayanon, B. McDanel, and H. Kung, “Distributed deep neural networks over the cloud, the edge and end devices,” *Proceedings of the 37th IEEE Intl. Conf. on Distributed Computing Systems*, 2017.
- [12] W. Haensch, T. Gokmen, and R. Puri, “The next generation of deep learning hardware: Analog computing,” *Proceedings of the IEEE*, vol. 107, no. 1, pp. 108–122, 2018.
- [13] W. Liu, X. Zang, Y. Li, and B. Vucetic, “Over-the-air computation systems: Optimization, analysis and scaling laws,” *IEEE Trans. on Wireless Communications*, vol. 19, no. 8, pp. 5488–5502, 2020.
- [14] P. Čížek and S. Sadıkoğlu, “Robust nonparametric regression: A review,” *WIREs Computational Statistics*, vol. 12, no. 3, p. e1492, 2020.
- [15] S. Du, Y. Wang, S. Balakrishnan, P. Ravikumar, and A. Singh, “Robust nonparametric regression under huber’s ϵ -contamination model,” *arXiv, math.ST*, 2018.
- [16] D. Blatná, “Outliers in regression,” *Trutnov*, vol. 30, pp. 1–6, 2006.
- [17] Y. Kim, Y. Cassuto, and L. R. Varshney, “Distributed boosting classifiers over noisy channels,” in *2020 IEEE 54th Asilomar Conf. on Signals, Systems, and Computers*, 2020, pp. 1491–1496.
- [18] Y. Cassuto and Y. Kim, “Boosting for straggling and flipping classifiers,” in *Proc. 2021 IEEE Intl. Symp. on Information Theory (ISIT)*, 2021, pp. 2441–2446.
- [19] 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 Information Theory (ISIT)*, 2022, pp. 2279–2284.
- [20] J. Shao, Y. Mao, and J. Zhang, “Learning task-oriented communication for edge inference: An information bottleneck approach,” *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 197–211, 2021.
- [21] Y. Ben-Hur, A. Goren, D. Klang, Y. Kim, and Y. Cassuto, “Ensemble classification with noisy real-valued base functions,” *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 4, pp. 1067–1080, 2023.
- [22] Y. Ben-Hur, Y. Cassuto, and I. Cohen, “Regression with an ensemble of noisy base functions,” in *2022 IEEE 32nd Intl. Workshop on Machine Learning for Signal Processing (MLSP)*, 2022, pp. 1–6.
- [23] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [24] J. H. Friedman, “Greedy function approximation: a gradient boosting machine,” *Annals of statistics*, pp. 1189–1232, 2001.
- [25] —, “Stochastic gradient boosting,” *Computational statistics & data analysis*, vol. 38, no. 4, pp. 367–378, 2002.
- [26] M. P. Perrone, *Improving regression estimation: Averaging methods for variance reduction with extensions to general convex measure optimization*. Brown University, 1993.
- [27] W. Gander, “Least squares with a quadratic constraint,” *Numerische Mathematik*, vol. 36, no. 3, pp. 291–307, 1980.
- [28] G. A. Seber, *A matrix handbook for statisticians*. John Wiley & Sons, 2008.
- [29] F. C. Leone, L. S. Nelson, and R. Nottingham, “The folded normal distribution,” *Technometrics*, vol. 3, no. 4, pp. 543–550, 1961.
- [30] L. Breiman, “Arcing the edge,” Citeseer, Tech. Rep., 1997.
- [31] Y. Ben-Hur, “Noisy-ensemble-regression,” 2024, available: <https://github.com/benhuryuval/Noisy-Ensemble-Regression>.
- [32] D. Dua and C. Graff, “UCI machine learning repository,” 2017, accessed: 10/01/2022. [Online]. Available: <http://archive.ics.uci.edu/ml>
- [33] 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. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/wine-quality>
- [34] L. Breiman, “Stacked regressions,” *Machine Learning*, vol. 24, no. 1, pp. 49–64, 1996.



Yuval Ben-Hur (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. (cum laude) degrees in Electrical Engineering from the Technion–Israel Institute of Technology, Haifa, Israel, in 2014 and 2018, respectively, where he is currently pursuing the Ph.D. degree in Electrical Engineering.

From 2013 to 2022, he was an Algorithm Engineer and a Team Leader with Israeli Defense Industry. Since 2022, he has been working on wireless communication systems as an Algorithm Engineer at Qualcomm. His research interests include the application of information and communication theory to emerging data storage and processing devices. He was a recipient of the Best Student Paper Award in data storage from the IEEE Communications Society in 2019 and the Best Student Paper Runner-Up Award from the 2022 IEEE International Workshop on Machine Learning for Signal Processing.



Yuval Cassuto (S'02–M'08–SM'14) is a faculty member at the Viterbi Department of Electrical and Computer Engineering, Technion – Israel Institute of Technology. His research interests lie at the intersection of the theoretical information sciences and the engineering of practical computing and storage systems. He has served on the technical program committees of leading conferences in both theory and systems.

During 2010–2011 he has been a Scientist at EPFL, the Swiss Federal Institute of Technology in Lausanne. From 2008 to 2010 he was a Research Staff Member at Hitachi Global Storage Technologies, San Jose Research Center. In 2018–2019 he held a Visiting Professor position at Western Digital Research, and a Visiting Scholar position at UC Berkeley. He received the B.Sc degree in Electrical Engineering, summa cum laude, from the Technion in 2001, and the M.S. and Ph.D. degrees in Electrical Engineering from the California Institute of Technology, in 2004 and 2008, respectively. From 2000 to 2002, he was with Qualcomm, Israel R&D Center, where he worked on modeling, design and analysis in wireless communications. Dr. Cassuto has won the Best Student Paper Award in data storage from the IEEE Communications Society in 2010 as a student, and in 2019 as an adviser. He also won faculty awards from Qualcomm, Intel, and Western Digital. Before that, he won the 2001 Texas Instruments DSP and Analog Challenge \$100,000 prize.