Mitigating Noise in Ensemble Classification with Real-Valued Base Functions

Yuval Ben-Hur*, Asaf Goren*, Da-El Klang*, Yongjune Kim† and Yuval Cassuto‡
*‡The Viterbi Faculty of Electrical and Computer Engineering, Technion–Israel Institute of Technology, Haifa, Israel
†DGIST, Daegu, South Korea
Emails: *{yuvalbh, asaf.goren, daelklang}@campus.technion.ac.il, ,†yjk@dgist.ac.kr, ,‡ycassuto@ee.technion.ac.il

***Abstract*—In data-intensive applications, it is advantageous to perform some partial processing close to the data, and communicate to a central processor the partial results instead of the data itself. When the communication medium is noisy, one must mitigate the resulting degradation in computation quality. We study this problem for the setup of binary classification performed by an ensemble of functions communicating real-valued confidence levels. We propose a noise-mitigation solution that works by optimizing the aggregation coefficients at the central processor. Toward that, we formulate a post-training gradient algorithm that minimizes the error probability given the dataset and the noise parameters. We further derive lower and upper bounds on the optimized error probability, and show empirical results that demonstrate the enhanced performance achieved by our scheme on real data.**

I. Introduction

Consider the classical supervised binary-classification problem, in which a classifier function has to be estimated given a training set of labeled data points $\mathcal{S} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$, where $\boldsymbol{x}_i \in \mathcal{X}^d$ ($\mathcal{X}^d$ is the d-dimensional data alphabet), and $y_i \in \mathcal{Y} = \{-1, +1\}$ is the binary label. The objective is to find a function $f(\cdot) : \mathcal{X}^d \to \mathcal{Y}$ that generalizes the relation between the input space $\mathcal{X}^d$ and outputs $\mathcal{Y}$, based on the training set $\mathcal{S} \subseteq \mathcal{X}^d \times \mathcal{Y}$. In this paper we are interested in *distributed* $f(\cdot)$ functions, which obtain their output by aggregating partial inference values communicated from multiple nodes/units over noisy channels. This setup is motivated by emerging compute architectures for artificial intelligence (AI), which perform complex inference tasks by circuits of simple compute nodes connected by non-ideal wires.

In our studied setup, there are T base nodes and each node $t \in \{1, \ldots, T\}$ implements an inference function $h_t(\cdot) : \mathcal{X}^d \to \mathbb{R}$ ($\mathbb{R}$ denotes the set of real numbers). At classification time, each node sends its output $h_t(\boldsymbol{x})$ to a central processor over a noisy channel. The central processor performs the final classification by taking the *sign of a (weighted) sum* of the values received from the base nodes. Our objective is to study the design of various components of the underlying system, toward maximizing its classification accuracy.

The aforementioned setup is motivated by *ensemble methods in machine learning* [1] that obtain powerful classification functions by aggregating an ensemble of weak base functions. The output of each base function can be thought of as its "soft vote" toward the final classification. In the most general case, as we assume in this work and as employed by the powerful *Real AdaBoost* method [2], these "soft votes" are real numbers. Ensemble methods, however, assume that the base-function outputs are delivered to the aggregating central processor *noise free*. Thus our objective in this paper is to introduce measures for mitigating the degradation from noise, while assuming that the base inference functions $\{h_t(\cdot)\}_{t=1}^{T}$ are given to us by some state-of-the-art ensemble method (which we do not control, and may not even know).

The main contribution of this work (Section IV) is the formulation of an optimization approach to improve ensemble classification's resilience to noise. For any ensemble functions $\{h_t(\cdot)\}_{t=1}^{T}$ and parameters of the corresponding additive Gaussian noise channels, our optimization algorithm finds aggregation coefficients at the central processor that minimize the probability of mismatch between the noisy and noiseless classifications. This is done by deriving the mismatch probability as a function of the ensemble functions, noise parameters, and aggregation coefficients, and then performing empirical risk minimization on the training dataset by unconstrained gradient descent. We also derive lower and upper bounds on the mismatch probability that add insight and help in predicting the classification performance. In Section V, we demonstrate the improvement offered by our approach on several real datasets. Earlier in the paper in Section III, we examine simpler complementary approaches to mitigate noise by allocating noise variances and transmit powers among the T channels. The different models suggested for the problem are reminiscent of classical problems in communications, only that the objective here is to protect from noise the ensemble's final classification value, and not the individual base-function outputs as in classical communication.

This work extends recent work addressing ensembles with noisy *binary* base functions [3], [4], which is a weaker model than the real-valued functions addressed here. The approach of this work also differs from the prior work in optimizing the mismatch probability in a *post-training procedure*, while [3] used data-oblivious resource allocation and [4] modified the ensemble training algorithms. Earlier work addresses noise in the training procedure [5], [6], [7] (label noise), and [8], [9] (training on noisy hardware), which is an important but

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

complementary problem.

II. Ensemble inference with additive noise

Consider a binary ensemble classifier $f(\cdot) : \mathcal{X}^d \to \mathcal{Y}$ implemented on a system comprising three types of elements: trained base functions $\{h_t(\cdot)\}_{t=1}^T$, communication channels and a central processor. The central processor aggregates $\{\tilde{h}_t(\cdot)\}_{t=1}^T$, which denote noisy versions of the values generated by the base functions. In the sequel we refer to the outputs of the base functions as *confidence levels*.

Definition 1. *Let $\{n_t\}_{t=1}^T$ be a set of random variables. Define the noisy confidence-levels $\{\tilde{h}_t(\boldsymbol{x})\}_{t=1}^T$ as*

$$\tilde{h}_t(\boldsymbol{x}) = h_t(\boldsymbol{x}) + n_t. \tag{1}$$

Let $f(\cdot)$ be an aggregation function and $\boldsymbol{x} \in \mathcal{X}^d$ a data sample. The following notation specifies the application of $f(\cdot)$ on the noisy confidence levels

$$\tilde{f}(\boldsymbol{x}) = f\left(\tilde{h}_1(\boldsymbol{x}), \ldots, \tilde{h}_T(\boldsymbol{x})\right). \tag{2}$$

We now characterize the classification-error probability of a noisy ensemble classifier $\tilde{f}(\cdot)$ over the training dataset $\mathcal{S}$. When inferring with noisy confidence levels, errors occur either due to limited generalization capability of the trained model, or due to the noise (1) added at inference time. The average classification-error probability for the dataset $\mathcal{S}$ is defined as

$$\tilde{P}_e\left(\mathcal{S}\right) \triangleq \frac{1}{N}\sum_{i=1}^{N} \Pr\{\tilde{f}(\boldsymbol{x}_i) \neq y_i\}, \tag{3}$$

where the probability space corresponds to the noise distribution.

Let us now define the *mismatch probability*, which measures the contribution of the noise to classification errors.

Definition 2. *Let $f(\cdot)$ be an aggregation function and let $\boldsymbol{x} \in \mathcal{X}^d$ be a data sample. The mismatch probability of $\tilde{f}(\boldsymbol{x})$ is defined as*

$$\tilde{P}(\boldsymbol{x}) \triangleq \Pr\left\{\tilde{f}(\boldsymbol{x}) \neq f(\boldsymbol{x})\right\}. \tag{4}$$

The average mismatch probability for a dataset $\mathcal{S}$ is

$$\tilde{P}\left(\mathcal{S}\right) = \frac{1}{N}\sum_{i=1}^{N} \tilde{P}(\boldsymbol{x}_i). \tag{5}$$

Let the classification error rate of the trained model for dataset $\mathcal{S}$ *without noise* be $P_e(\mathcal{S}) = \frac{1}{N}\sum_{i=1}^{N}\left[f(\boldsymbol{x}_i) \neq y_i\right]$ where $[\pi]$ denotes the indicator function of the predicate π. Using Definition 2, we can upper bound the classification-error probability of the noisy classifier for dataset $\mathcal{S}$ as follows [3]:

$$\tilde{P}_e(\mathcal{S}) \leq P_e(\mathcal{S}) + \tilde{P}(\mathcal{S}). \tag{6}$$

This inequality is true for each individual data sample $\boldsymbol{x}$, and therefore also holds for the average error probability as well. In the sequel, we omit the dataset argument $\mathcal{S}$ from the error and/or mismatch probabilities, when it is clear from the context. Note that P_e depends on the learned model, its training process (e.g., Real AdaBoost [2]) and the data set $\mathcal{S}$, but is independent of the noise. Hence, in order to reduce the deterioration in classification performance introduced by the noise, we can minimize $\tilde{P}$ rather than $\tilde{P}_e$. This observation is useful since minimizing $\tilde{P}_e$ directly is difficult due to its much more complex dependencies.

While the proposed framework is developed for general channels, we focus on the class of additive Gaussian channels. Hence, for the remainder of this paper the vector of additive random noise variables $\boldsymbol{n}$ is defined as follows.

Definition 3. *Let $\boldsymbol{\Sigma}$ be a positive semi-definite diagonal matrix. Define the noise vector $\boldsymbol{n} = (n_1, \ldots, n_T)$, where $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$.*

The random vector $\boldsymbol{n}$ is re-drawn from the distribution for each classification instance $\boldsymbol{x}$. Although this paper's results apply to general ensembles obtained by arbitrary training procedures, we give Real AdaBoost as a concrete method for obtaining such ensembles. Real AdaBoost [2] is an algorithm for obtaining base classifiers that output real values as their confidence levels. This allows to use a simple unweighted aggregation decision rule,

$$f(\boldsymbol{x}) = \operatorname{sign}\Big(\sum_{t=1}^{T} h_t(\boldsymbol{x})\Big). \tag{7}$$

Note that this is not case for the discrete version of AdaBoost [10], in which every $h_t(\cdot)$ has outputs in $\{-1, +1\}$, and the final decision rule is a weighted sum $f(\boldsymbol{x}) = \operatorname{sign}(\sum_{t=1}^{T} a_t h_t(\boldsymbol{x}))$ where a_t is a real coefficient optimized during training.

III. Simple setups for noisy classification

In this section we consider two natural models for noisy classification, both motivated directly by communication-theoretic combining schemes. We first examine the simple unweighted aggregation (used, e.g., in Real AdaBoost), and then study a gain-controlled decision rule.

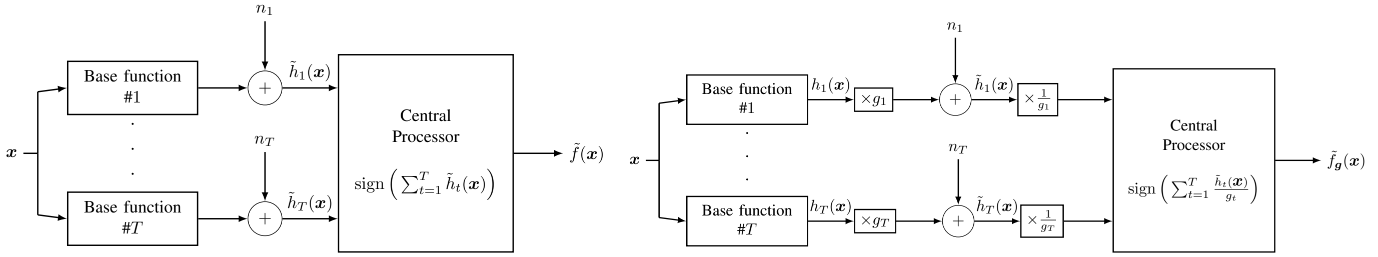
A. Noisy unweighted aggregation

The setup of noisy real-valued confidence levels with unweighted aggregation is depicted in Fig. 1a. Since the decision rule is given by $\tilde{f}(\boldsymbol{x}) = \operatorname{sign}\left(\sum_{t=1}^{T} \tilde{h}_t(\boldsymbol{x})\right)$, this setup naturally leads to the following noise allocation problem.

Problem 1. *Let ζ be a non-negative real number and let the noise random vector $\boldsymbol{n}$ be an independent Gaussian vector whose standard deviations are denoted $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_T)$. Find*

$$\min_{\boldsymbol{\sigma}} \tilde{P}(\mathcal{S}) \;\; s.t. \;\; \sum_{t=1}^{T} \sigma_t^2 = \zeta. \tag{8}$$

Toward optimizing Problem 1, we obtain a closed-form expression for the mismatch probability of $\tilde{f}(\cdot)$.


n_1
Base function #1
$\tilde{h}_1(\boldsymbol{x})$
$\boldsymbol{x}$
n_T
Base function #T
$\tilde{h}_T(\boldsymbol{x})$
Central Processor
$\text{sign}\left(\sum_{t=1}^T \tilde{h}_t(\boldsymbol{x})\right)$
$\tilde{f}(\boldsymbol{x})$
$h_1(\boldsymbol{x})$
$\times g_1$
$\times \frac{1}{g_1}$
$h_T(\boldsymbol{x})$
$\times g_T$
$\times \frac{1}{g_T}$
$\text{sign}\left(\sum_{t=1}^T \frac{\tilde{h}_t(\boldsymbol{x})}{g_t}\right)$
$\tilde{f}_{\boldsymbol{g}}(\boldsymbol{x})$

(a) Unweighted aggregation

(b) Equalized-gain aggregation

Fig. 1: Simple aggregation setups for noisy base-classifiers

Proposition 1. *Let $\boldsymbol{x} \in \mathcal{X}^d$ and let $\tilde{f}(\boldsymbol{x}) = \text{sign}\left(\sum_{t=1}^T \tilde{h}_t(\boldsymbol{x})\right)$. The mismatch probability of $\tilde{f}(\boldsymbol{x})$ is*

$$\tilde{P}(\boldsymbol{x}) = Q\left(\frac{\left|\sum_{t=1}^T h_t(\boldsymbol{x})\right|}{\sqrt{\sum_{t=1}^T \sigma_t^2}}\right), \tag{9}$$

where $Q(\cdot)$ is the tail distribution function of the standard normal (Gaussian) distribution.

Proof. Assume, from symmetry, that $f(\boldsymbol{x}) > 0$. Then, by definition

$$\tilde{P}(\boldsymbol{x}) = \Pr\left\{\sum_{t=1}^T (h_t(\boldsymbol{x}) + n_t) < 0\right\} = Q\left(\frac{\left|\sum_{t=1}^T h_t(\boldsymbol{x})\right|}{\sqrt{\sum_{t=1}^T \sigma_t^2}}\right), \tag{10}$$

where the last transition holds since $\boldsymbol{n}$ is an independent Gaussian vector. □

Considering Problem 1 in view of the mismatch probability in (9), we get that any noise allocation leads to the same objective value of $\tilde{P} = Q\left(\frac{|\sum_{t=1}^T h_t(\boldsymbol{x})|}{\sqrt{\zeta}}\right)$. Hence, alternative approaches – beyond unweighted aggregation – have to be considered for controlling the mismatch probability.

B. Noisy gain-constrained equalized aggregation

Inspired by classical multi-channel equalization problems [11], we proceed beyond unweighted aggregation to consider an equalized power-allocation problem. As depicted in Fig. 1b, each channel is allocated a gain factor g_t that multiplies the classifier output before transmission. The inverse of this factor is applied upon aggregation at the central processor. This equalization with the inverse is motivated toward maintaining the same end-to-end average confidences as in the noiseless setup. Similarly to Proposition 1, the mismatch probability is now given by

$$\tilde{P}(\boldsymbol{x}) = Q\left(\frac{\left|\sum_{t=1}^T h_t(\boldsymbol{x})\right|}{\sqrt{\sum_{t=1}^T \sigma_t^2/g_t^2}}\right). \tag{11}$$

Assuming constrained overall gain, we define the gain-allocation mismatch-minimization problem.

Problem 2. *Given a non-negative real number G and $\boldsymbol{n}$'s noise standard deviations $(\sigma_1, \ldots, \sigma_T)$, find*

$$\min_{(g_1,\ldots,g_T)\in\mathbb{R}^T} \sum_{t=1}^T \sigma_t^2/g_t^2 \;\; s.t. \;\; \sum_{t=1}^T g_t^2 \le G, \;\; g_t \ge 0. \tag{12}$$

Problem 2 allows to control the mismatch probability through gain allocation. For example, in the special case of equal noise variances (i.e., $\sigma_t = \sigma$ for all $t = 1, \ldots, T$), the uniform allocation $g_t = \sqrt{G/T}$ is optimal according to the Karush–Kuhn–Tucker (KKT) conditions. For general standard-deviation values, we have the following theorem.

Theorem 1. *The optimal gains are given by $g_t^* = \sqrt{\frac{G}{\sum_{\tau=1}^T \sigma_\tau}} \cdot \sigma_t$; the optimal mismatch probability is*

$$\tilde{P}^*(\boldsymbol{x}) = Q\left(\sqrt{G} \cdot \frac{\left|\sum_{t=1}^T h_t(\boldsymbol{x})\right|}{\sum_{t=1}^T \sigma_t}\right). \tag{13}$$

We omit the proof of Theorem 1 due to lack of space.

IV. Noisy inference through re-weighted aggregation

Following the simple aggregation setups of Section III, in the remainder of the paper we pursue an aggregation setup that gives rise to a more interesting optimization framework. In this setup, we assume control of neither the transmission gains nor the noise allocation, thus having the original (trained) confidence levels sent through the specified noisy channels. At aggregation time, each noisy classifier output is multiplied by a coefficient α_t (without enforcing end-to-end gain constraints as in Problem 2). These coefficients are optimized to minimize the mismatch probability through a post-training procedure, which re-weights the confidence levels before aggregation, according to their importance *in the noisy inference problem*. Since re-weighting is done on the received values at the central processor, these coefficients can be set freely without worrying about power constraints (needed in Problems 1 and 2).

A. Optimized re-weighted aggregation

As depicted in Fig. 2, the final decision obtained for a data sample $\boldsymbol{x}$ by re-weighted aggregation is given by

$$\tilde{f}_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \text{sign}\left(\sum_{t=1}^T \alpha_t \tilde{h}_t(\boldsymbol{x})\right). \tag{14}$$

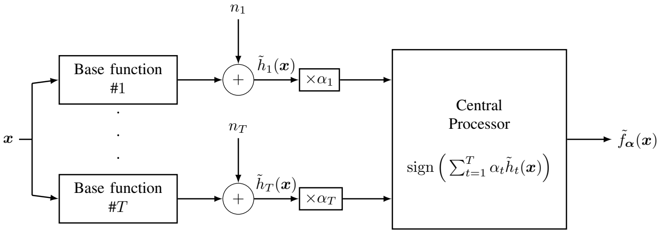

n_1
Base function #1
$\tilde{h}_1(\boldsymbol{x})$
$\times\alpha_1$
$\boldsymbol{x}$
n_T
Base function #T
$\tilde{h}_T(\boldsymbol{x})$
$\times\alpha_T$
Central Processor
$\text{sign}\left(\sum_{t=1}^{T}\alpha_t\tilde{h}_t(\boldsymbol{x})\right)$
$\tilde{f}_{\boldsymbol{\alpha}}(\boldsymbol{x})$

Fig. 2: Re-weighted aggregation with noisy base functions.

For clarity and brevity of expressions, we formulate, analyze and solve the coefficient-optimization problem using vector forms. We denote $\boldsymbol{h}(\boldsymbol{x}) = (h_1(\boldsymbol{x}),\dots,h_T(\boldsymbol{x}))$, $\tilde{\boldsymbol{h}}(\boldsymbol{x}) = (\tilde{h}_1(\boldsymbol{x}),\dots,\tilde{h}_T(\boldsymbol{x}))$ and $\boldsymbol{\alpha} = (\alpha_1,\dots,\alpha_T)$. Also, let $\boldsymbol{H}(\boldsymbol{x}) \triangleq \boldsymbol{h}(\boldsymbol{x})^\top\boldsymbol{h}(\boldsymbol{x})$ and let $R_{\boldsymbol{\alpha}}(\boldsymbol{x})$ denote the *generalized Rayleigh quotient* $\frac{\boldsymbol{\alpha}\boldsymbol{H}(\boldsymbol{x})\boldsymbol{\alpha}^\top}{\boldsymbol{\alpha}\boldsymbol{\Sigma}\boldsymbol{\alpha}^\top}$. We omit the $\boldsymbol{x}$ argument from $\boldsymbol{h}(\boldsymbol{x})$, $\tilde{\boldsymbol{h}}(\boldsymbol{x})$ and $\boldsymbol{H}(\boldsymbol{x})$ when it is clear from the context.

Theorem 2. *Let $\boldsymbol{x}\in\mathcal{X}^d$ and let $\boldsymbol{\alpha}\in\mathbb{R}^T$. The mismatch probability of $\tilde{f}_{\boldsymbol{\alpha}}(\boldsymbol{x})$ is*

$$\tilde{P}_{\boldsymbol{\alpha}}(\boldsymbol{x}) = Q\left(\text{sign}\left(\mathbf{1}\boldsymbol{H}(\boldsymbol{x})\boldsymbol{\alpha}^\top\right)\sqrt{R_{\boldsymbol{\alpha}}(\boldsymbol{x})}\right). \tag{15}$$

Proof. The mismatch probability is given by

$$\Pr\{\tilde{f}_{\boldsymbol{\alpha}}(\boldsymbol{x}) \neq f(\boldsymbol{x})\} = \Pr\left\{\boldsymbol{\alpha}\left(\boldsymbol{h}^\top + \boldsymbol{n}^\top\right)\boldsymbol{h}\mathbf{1}^\top < 0\right\}. \tag{16}$$

Denote $A = \boldsymbol{\alpha}\boldsymbol{h}^\top\boldsymbol{h}\mathbf{1}^\top$, $B = \boldsymbol{\alpha}\boldsymbol{n}^\top\boldsymbol{h}\mathbf{1}^\top$ and note that $B \sim \mathcal{N}(0,\ \mathbf{1}\boldsymbol{h}^\top\boldsymbol{h}\mathbf{1}^\top\boldsymbol{\alpha}\boldsymbol{\Sigma}\boldsymbol{\alpha}^\top)$. Therefore,

$$\tilde{P}_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \Pr\{B < -A\} = Q\left(\frac{\boldsymbol{\alpha}\boldsymbol{h}^\top\boldsymbol{h}\mathbf{1}^\top}{\sqrt{\mathbf{1}\boldsymbol{h}^\top\boldsymbol{h}\mathbf{1}^\top\boldsymbol{\alpha}\boldsymbol{\Sigma}\boldsymbol{\alpha}^\top}}\right), \tag{17}$$

which can be manipulated to give (15). □

Optimized re-weighting is now defined as the coefficient assignment that minimizes $\tilde{P}_{\boldsymbol{\alpha}}(\mathcal{S})$ provided by Theorem 2.

Problem 3. *Given a dataset $\mathcal{S} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$, find the vector $\boldsymbol{\alpha}^*$ that minimizes the average mismatch probability,*

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^T}\ \left\{\frac{1}{N}\sum_{i=1}^{N}\tilde{P}_{\boldsymbol{\alpha}}(\boldsymbol{x}_i)\right\}. \tag{18}$$

Toward solving Problem 3 via unconstrained minimization, we derived the gradient of the objective with respect to the coefficients vector $\boldsymbol{\alpha}$. The gradient is denoted $\nabla\tilde{P}_{\boldsymbol{\alpha}}$, but its derivation is omitted due to lack of space. Alg. 1 is a momentum gradient-descent algorithm for minimizing the mismatch probability over a dataset $\mathcal{S}$. Note that the dataset labels y_i are *n*ot used for optimizing the coefficients; nor do any noise samples.

B. Bounds on the average mismatch probability

To understand the limits to reliable noisy classification, we derive lower and upper bounds on the optimized average mismatch probability $\tilde{P}_{\boldsymbol{\alpha}^*}(\mathcal{S})$ for a given dataset $\mathcal{S}$. The bounds provide fundamental limits on the optimal performance for a given data set, trained ensemble, and noise covariance.

Algorithm 1 Gradient-descent minimization of $\tilde{P}_{\boldsymbol{\alpha}}$

```
Input: $\{h_t(\cdot)\}_{t=1}^{T}$: trained base functions, $\{\boldsymbol{x}_i\}_{i=1}^{N}$: training data samples, $\boldsymbol{\Sigma}$: noise covariance matrix
Output: $\{\alpha_t\}_{t=1}^{T}$: aggregation coefficients
Set: $i_{\max}$ (# iter.), $\eta$ (learn rate), $\gamma$ (momentum), $\tau$ and $\epsilon$
Initialize: $\boldsymbol{\alpha}^{(0)} \leftarrow \mathbf{1}$ , $\delta\boldsymbol{\alpha}^{(-1)} \leftarrow \mathbf{0}$ and $i \leftarrow 0$
while $i \le i_{\max}$ do
    $\delta\boldsymbol{\alpha}^{(i)} \leftarrow \gamma\cdot\delta\boldsymbol{\alpha}^{(i-1)} - \eta\cdot\frac{\nabla\tilde{P}_{\boldsymbol{\alpha}}}{|\nabla\tilde{P}_{\boldsymbol{\alpha}}|_{\ell_2}+\epsilon}$
    $\boldsymbol{\alpha}^{(i+1)} \leftarrow \boldsymbol{\alpha}^{(i)} + \delta\boldsymbol{\alpha}^{(i)}$
    if $|\tilde{P}_{\boldsymbol{\alpha}^{(i)}} - \tilde{P}_{\boldsymbol{\alpha}^{(i-1)}}| \le \tau$: break; else: $i \leftarrow i+1$
end while
$i^* \leftarrow \arg\min_{0\le j\le i}\tilde{P}_{\boldsymbol{\alpha}^{(j)}}$
return $\boldsymbol{\alpha}^{(i^*)}$
```

Theorem 3. *Let $\boldsymbol{\alpha}^*\in\mathbb{R}^T$ be a solution to Problem 3. Then,*

$$\tilde{P}_{\boldsymbol{\alpha}^*} \le \frac{1}{2N}\sum_{i=1}^{N}\exp\left(-\frac{1}{2}R_{\mathbf{1}}(\boldsymbol{x}_i)\right). \tag{19}$$

Proof. The optimal coefficient vector $\boldsymbol{\alpha}^*$ minimizes $\tilde{P}_{\boldsymbol{\alpha}}$. Therefore, according to (5) and (15), we have

$$\tilde{P}_{\boldsymbol{\alpha}^*} \le \tilde{P}_{\mathbf{1}} = \frac{1}{N}\sum_{i=1}^{N}Q\left(\sqrt{\frac{\mathbf{1}\boldsymbol{H}(\boldsymbol{x}_i)\mathbf{1}^\top}{\mathbf{1}\boldsymbol{\Sigma}\mathbf{1}^\top}}\,\text{sign}\left(\mathbf{1}\boldsymbol{H}(\boldsymbol{x}_i)\mathbf{1}^\top\right)\right). \tag{20}$$

Since $\boldsymbol{H}(\boldsymbol{x}_i)$ is positive semi-definite for every $i = 1,\dots,N$, we get that $\mathbf{1}\boldsymbol{H}(\boldsymbol{x}_i)\mathbf{1}^\top \ge 0$. The proof is concluded by applying the well-known [12] upper-bound $Q(z) \le \frac{1}{2}\exp\left(-\frac{z^2}{2}\right)$ for $z \ge 0$ on each of the summands. □

Based on properties of $R_{\boldsymbol{\alpha}}(\boldsymbol{x})$, we obtain a lower-bound on $\tilde{P}_{\boldsymbol{\alpha}^*}$ as well.

Theorem 4. *Let $\boldsymbol{\alpha}\in\mathbb{R}^T$ be an arbitrary coefficients vector. If $\boldsymbol{\Sigma}$ is a positive-definite matrix,*

$$\tilde{P}_{\boldsymbol{\alpha}} \ge \frac{1}{N}\sum_{i=1}^{N}Q\left(\sqrt{\lambda_i^{\max}}\right), \tag{21}$$

where $\lambda_i^{\max}$ is the largest eigenvalue of the matrix $\boldsymbol{\Sigma}^{-1}\boldsymbol{H}(\boldsymbol{x}_i)$. In the special case when $\boldsymbol{\Sigma}$ is a diagonal matrix, we get

$$\tilde{P}_{\boldsymbol{\alpha}} \ge \frac{1}{N}\sum_{i=1}^{N}Q\left(\sqrt{\boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\Sigma}^{-1}\boldsymbol{h}(\boldsymbol{x}_i)^\top}\right). \tag{22}$$

Proof. From Theorem 2, we have

$$\tilde{P}_{\boldsymbol{\alpha}} = \frac{1}{N}\sum_{i=1}^{N}Q\left(\text{sign}\left(\mathbf{1}\boldsymbol{H}(\boldsymbol{x}_i)\boldsymbol{\alpha}^\top\right)\sqrt{R_{\boldsymbol{\alpha}}(\boldsymbol{x}_i)}\right). \tag{23}$$

Since $x \le |x|$ and $Q(\cdot)$ is monotonically decreasing, then $Q(x) \ge Q(|x|)$. Therefore, $N \cdot \tilde{P}_{\boldsymbol{\alpha}}$ is lower-bounded by

$$\begin{aligned}\sum_{i=1}^{N} Q\Big(\sqrt{R_{\boldsymbol{\alpha}}(\boldsymbol{x}_i)}\Big) &\ge \min_{\boldsymbol{\alpha}\in\mathbb{R}^T} \sum_{i=1}^{N} Q\Big(\sqrt{R_{\boldsymbol{\alpha}}(\boldsymbol{x}_i)}\Big) \\ \ge \sum_{i=1}^{N} \min_{\boldsymbol{\alpha}_i\in\mathbb{R}^T} Q\Big(\sqrt{R_{\boldsymbol{\alpha}_i}(\boldsymbol{x}_i)}\Big) &= \sum_{i=1}^{N} Q\Big(\sqrt{\max_{\boldsymbol{\alpha}_i\in\mathbb{R}^T} R_{\boldsymbol{\alpha}_i}(\boldsymbol{x}_i)}\Big).\end{aligned} \tag{24}$$

It is well known [13] that for symmetric $\boldsymbol{H}(\boldsymbol{x}_i)$ and positive-definite $\boldsymbol{\Sigma}$ the generalized Rayleigh quotient is maximized by the largest eigenvalue of $\boldsymbol{\Sigma}^{-1}\boldsymbol{H}(\boldsymbol{x}_i)$, which gives (21). When $\boldsymbol{\Sigma}$ is diagonal, it can be verified that $\boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\Sigma}^{-1}\boldsymbol{h}(\boldsymbol{x}_i)^\top$ is the largest eigenvalue, yielding (22). □

V. Performance Evaluation

We now evaluate Alg. 1 and the bounds in Section IV using real-world data. In the following experiments, we trained an ensemble of decision stumps using Real AdaBoost, and then applied Alg. 1 to obtain the optimized aggregation coefficients. For consistent performance evaluation in diverse scenarios, we define a *signal-to-noise ratio (SNR)* measure that quantifies the severity of the noise relative to the classifier confidence-levels.

Definition 4. *The average SNR is defined as*

$$SNR \triangleq \frac{1}{N}\sum_{i=1}^{N} SNR_i, \;\; \text{where } SNR_i = \frac{||\boldsymbol{h}(\boldsymbol{x}_i)||_2^2}{||\boldsymbol{\sigma}||_2^2}. \tag{25}$$

We experiment with 3 well-known and widely used datasets:

1) Parkinson's disease [14] - A set consisting of $N = 195$ patient tests with $d = 22$ features labeled by a positive/negative diagnosis of Parkinson's disease.
2) Heart disease [15] - A set consisting of $N = 297$ patient tests with $d = 13$ features labeled as healthy/sick.
3) Breast cancer [16] - A set consisting of $N = 569$ tissue tests with $d = 30$ features labeled as benign/malignant.

For each dataset $\mathcal{S}'$, we use a random training set $\mathcal{S} \subseteq \mathcal{S}'$ comprising 80% of the data samples in $\mathcal{S}'$. The remaining 20% are used as a test set for evaluating classification performance over the noisy channels. To obtain a reliable estimate of the classification-error probability, we draw 500 realizations of the training set $\mathcal{S}$. For each realization, the ensemble is trained using $\mathcal{S}$ and the coefficients are optimized using $\mathcal{S}$ and $\boldsymbol{\Sigma}$. The classification-error probability is then evaluated *on the test set* $\mathcal{S}' \setminus \mathcal{S}$, where noise is redrawn for each data sample.

Fig. 3 shows the classification-error probabilities as a function of the average SNR for equal-variance independent Gaussian channels. The plots show the performance of the optimized $\tilde{f}_{\boldsymbol{\alpha}}(\cdot)$ (with markers) and the unweighted $\tilde{f}(\cdot)$, amounting to $\boldsymbol{\alpha} = \mathbf{1}$, (without markers), for all three datasets and with ensemble sizes $T = 30, 45, 60$ (respectively). Clearly, optimized re-weighting outperforms unweighted aggregation over the entire SNR range, with a typical gap of around 5dB. The error curve behavior for extreme SNRs is also as expected: approaching random-guessing for low SNRs and coinciding with unweighted aggregation for high SNRs.

Fig. 4 plots lower-bounds and upper-bounds on the mismatch probability for all three data sets. We chose the ensemble size for each data set so that it corresponds to Fig. 3. Interestingly, the bounds predict the relations between the error probabilities of the different datasets. Furthermore, the bounds seem to be valid for the test set, although calculated using the training set, indicating successful generalization.

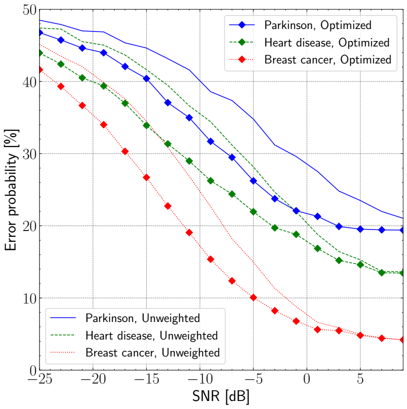


Fig. 3: Classification performance comparison between unweighted and re-weighted aggregation.

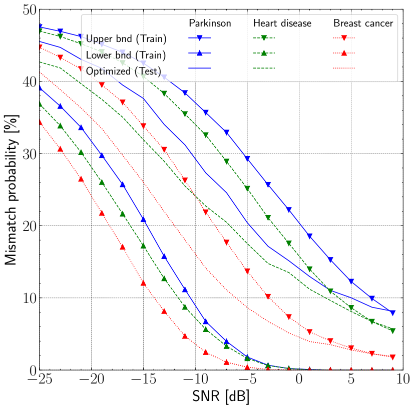


Fig. 4: The optimized (test set) mismatch probability compared to the (train set) upper and lower bounds in Eq. (20) and (22)

VI. Conclusion

This paper addresses the fundamental problem of binary ensemble classification with noisy real-valued base functions. Focusing on additive Gaussian noise, we optimize classification performance and provide rigorous performance guarantees. Experiments conducted with our proposed approach provide empirical evidence for improved classification accuracy.

Interesting future work includes extending this approach to non-Gaussian noise models, e.g. quantization noise due to limited precision. Moreover, similar techniques can be applied to neural networks, in which every neuron performs noisy aggregation when operated on noisy hardware.

References

[1] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*. Springer, 2000, pp. 1–15.

[2] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine learning*, vol. 37, no. 3, pp. 297–336, 1999.

[3] Y. Kim, Y. Cassuto, and L. R. Varshney, "Distributed boosting classifiers over noisy channels," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, 2020, pp. 1491–1496.

[4] Y. Cassuto and Y. Kim, "Boosting for straggling and flipping classifiers," in *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 2441–2446.

[5] D. Angluin and P. Laird, "Learning from noisy examples," *Machine Learning*, vol. 2, no. 4, pp. 343–370, 1988.

[6] B. Frénay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 5, pp. 845–869, 2013.

[7] Z. Xiao, Z. Luo, B. Zhong, and X. Dang, "Robust and efficient boosting method using the conditional risk," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 7, pp. 3069–3083, 2017.

[8] Z. Wang, R. E. Schapire, and N. Verma, "Error adaptive classifier boosting (EACB): Leveraging data-driven training towards hardware resilience for signal inference," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 4, pp. 1136–1145, 2015.

[9] Z. Wang, K. H. Lee, and N. Verma, "Overcoming computational errors in sensing platforms through embedded machine-learning kernels," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 8, pp. 1459–1470, 2014.

[10] 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.

[11] J. G. Proakis and M. Salehi, *Digital communications*. McGraw-Hill Communications Inc, 2001.

[12] I. M. Jacobs and J. Wozencraft, *Principles of communication engineering*. Wiley, 1965.

[13] B. Ghojogh, F. Karray, and M. Crowley, "Eigenvalue and generalized eigenvalue problems: Tutorial," *arXiv preprint arXiv:1903.11240*, 2019.

[14] M. Little, "UCI machine learning repository: Parkinsons data set," 2008, available: https://archive.ics.uci.edu/ml/datasets/parkinsons.

[15] A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano, "UCI machine learning repository: Heart disease data set," 1988, available: https://archive.ics.uci.edu/ml/datasets/heart+disease.

[16] W. H. Wolberg, W. N. Street, and O. L. Mangasarian, "UCI breast cancer wisconsin (diagnostic) data set," 1995, available: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic).

[17] J. D. Garrett, "garrettj403/SciencePlots," Sep. 2021. [Online]. Available: http://doi.org/10.5281/zenodo.4106649