Estimation of the Label-Noise Transition Matrix with Performance Guarantees via Selective Classification

Xabier de Juan[1] **Santiago Mazuelas**[1,2] **Yilun Zhu**[3] **Clayton Scott**[3]
[1]Basque Center of Applied Mathematics (BCAM)
[2]IKERBASQUE-Basque Foundation for Science
[3]Electrical and Computer Engineering, University of Michigan
{xdejuan, smazuelas}@bcamath.org {allanzhu, clayscot}@umich.edu

Abstract

Modern machine learning depends heavily on massive datasets, but obtaining high-quality annotations at scale is often expensive. As a result, learning from noisily-labeled data has become common, making accurate estimation of the label-noise transition matrix crucial. However, existing transition matrix estimators rely on the fragile estimation of class-posteriors and do not provide finite-sample performance guarantees. In this work, we propose a novel methodology to estimate the transition matrix based on one-sided selective classification. This approach bypasses class-posterior estimation, provides finite-sample performance guarantees, and leverages flexible learning methods for binary classification. Moreover, we introduce effective algorithms to implement the proposed methodology and provide their refined finite-sample performance bounds.

1 Introduction

The success of modern machine learning relies heavily on the availability of massive labeled datasets. However, obtaining high-quality annotations at scale is often cost-prohibitive, leading to the widespread adoption of cost-effective and less accurate labeling procedures [1–3]. Such compromise introduces label noise, where observed labels may differ from the underlying ground truth. The probabilities of these label flips are captured by the label-noise transition matrix.

An accurate estimate of the transition matrix can address many of the problems caused by label noise. For instance, loss correction techniques leverage the transition matrix to recover the Bayes-optimal classifier from noisy samples [4–9]. Beyond standard classification, a reliable estimate of the transition matrix can enable the construction of informative prediction sets for conformal prediction [10], the implementation of fair classification models under biased data [11], and the training of conditional diffusion models with noisy labels [12].

Existing transition matrix estimators heavily rely on the pointwise estimation of the noisy class-posterior [7, 8, 13–17]. This requirement is fragile since even a poor estimate at a single point can lead to a large subsequent estimation error for the transition matrix [18]. In addition, estimating the class-posterior is provably hard. For example, any estimator of the class-posterior suffers from the curse of dimensionality [19, 20], and common methods often result in inaccurate class-posterior estimations [21].

While recent works have established consistency for certain estimators of the label-noise transition matrix [15, 16], the literature lacks finite-sample performance guarantees. Approaches in the closely related problem of mixture proportion estimation (MPE) [5, 6, 22–26] offer finite-sample performance guarantees. However, most of these approaches are restricted to binary settings, while others rely on intractable algorithms.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

In this work, we introduce a methodology for estimating the label-noise transition matrix based on one-sided selective classification. This approach does not require pointwise class-posterior estimation and provides finite-sample performance guarantees. Moreover, our methodology allows us to estimate the transition matrix by adapting established techniques from selective classification. Specifically, the main contributions in the paper are as follows:

- We frame the estimation of each column of the transition matrix as a one-sided selective classification problem, where we minimize the false discovery rate subject to a minimum coverage constraint.
- We provide finite-sample error bounds for the proposed approach, and prove that our estimator achieves the parametric convergence rate up to a small bias. In particular, we show that our methodology avoids the curse of dimensionality, unlike existing estimators.
- We propose computationally tractable algorithms for our methodology that leverage general methods for binary classification.
- We provide a theoretical analysis of the proposed algorithms, showing that they admit refined finite-sample error bounds analogous to classical generalization bounds for classification.

2 Preliminaries

This section defines the label-noise transition matrix and describes existing methods for its estimation.

2.1 Label-noise transition matrix

Let $\mathcal{X} \subseteq \mathbb{R}^d$ be the feature space and let $\mathcal{Y} = \{1, 2, \ldots, |\mathcal{Y}|\}$ be the label set. Instance and clean-label pairs $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ follow an underlying distribution p, while their noisy-label counterparts $(X, \widetilde{Y}) \in \mathcal{X} \times \mathcal{Y}$ are drawn from the distribution $\widetilde{\mathrm{p}}$. We assume that the distribution of $\widetilde{Y}$ depends on Y, but not on X, i.e., X and $\widetilde{Y}$ are independent given Y. This assumption is referred to as class-dependent label noise and it is standard in the literature (see e.g., [7, 8, 13–18]). It has shown to be effective in practice [16, 18], even in scenarios where label noise may change across instances.

The label-noise transition matrix $\mathbf{T} \in [0,1]^{|\mathcal{Y}| \times |\mathcal{Y}|}$ is defined entry-wise as $T_{i,j} = \mathbb{P}(\widetilde{Y} = i \mid Y = j)$, for $i, j \in \mathcal{Y}$, so that $T_{i,j}$ is the probability that an instance with true label j is observed as noisy label i. We assume that $\mathbf{T}$ is a row-diagonally dominant matrix (i.e., $T_{i,i} > T_{i,j}$ for all $j \neq i$), paralleling common assumptions in the literature [8, 13, 15, 16, 27].

The goal is to estimate the matrix $\mathbf{T}$ using only noisy samples drawn from $\widetilde{\mathrm{p}}$. Specifically, we assume that we have access to n i.i.d. samples from $\widetilde{\mathrm{p}}$, $\{(x_k, \widetilde{y}_k)\}_{k=1}^{n} \subseteq \mathcal{X} \times \mathcal{Y}$.

2.2 Related work

The most common approach to estimate the transition matrix is based on identifying anchor points for each class [7, 8]. This method exploits the fact that clean and noisy class-posterior probabilities are related by the matrix $\mathbf{T}$, that is, $\boldsymbol{\eta}^{\mathrm{noisy}}(x) = \mathbf{T}\boldsymbol{\eta}(x)$ where $\eta_i^{\mathrm{noisy}}(x) = \mathbb{P}(\widetilde{Y} = i \mid X = x)$ and $\eta_i(x) = \mathbb{P}(Y = i \mid X = x)$.

The method relies on the *anchor-point assumption* [7, 8] which states that there are instances with clean class-posterior probability equal to one for each class. In particular, the anchor-based method exploits the fact that if $x^{(j)}$ is an anchor point for class j, the entries of the transition matrix are given by $T_{i,j} = \eta_i^{\mathrm{noisy}}(x^{(j)})$.

The anchor-based method [8] first obtains an estimate $\widehat{\boldsymbol{\eta}}^{\mathrm{noisy}}(x) \in [0,1]^{|\mathcal{Y}|}$ of the vector of noisy class-posteriors $\boldsymbol{\eta}^{\mathrm{noisy}}(x) \in [0,1]^{|\mathcal{Y}|}$. Then, for each class $j \in \mathcal{Y}$, the algorithm estimates the corresponding column of $\mathbf{T}$ through the following procedure.

1. Find a candidate for anchor point of the j-th class by obtaining the instance that maximizes the j-th component of $\widehat{\boldsymbol{\eta}}^{\mathrm{noisy}}$, $x^{(j)} \in \arg\max_{x \in \mathcal{X}} \widehat{\eta}_j^{\mathrm{noisy}}(x)$.
2. Estimate the (i, j)-th entry of $\mathbf{T}$ as $\widehat{T}_{i,j}^{\mathrm{anchor}}(x^{(j)}) = \widehat{\eta}_i^{\mathrm{noisy}}(x^{(j)})$.

Subsequent methods eliminated the explicit search for anchor points [13–17], yet they still critically rely on an accurate pointwise estimation of the class-posterior. This dependency creates a severe vulnerability since a poorly estimated class-posterior at a single point can produce a large estimation error for the transition matrix [18]. Furthermore, common methods (e.g., neural networks) are known to yield highly unreliable class-posterior estimates (see e.g., [21]). Specifically, if α is the smoothness parameter of the class-posterior and d is the dimension of the feature space, the minimax optimal pointwise estimation error of the class-posterior is $n^{-\alpha/(2\alpha+d)}$ [19, 20], yielding a prohibitively slow convergence rate, even in moderate dimensions. The recent work [18] bypasses the need for class-posterior estimation and instead relies on the minimization of noise-robust losses. However, this approach relies on multiple assumptions that are unlikely to hold in practice. For instance, Assumption 1 in [18] requires that a minimizer of cross-entropy is the only minimizer of the noise-robust loss, a condition that is often not satisfied (see e.g., [28, 29]).

Existing literature does not provide finite-sample performance guarantees and solely consistency has been established for some estimators [15, 16]. Estimators in the closely related field of MPE [5, 6, 22–26] instead estimate the inverse flip-rates $\mathbb{P}(Y = i \mid \widetilde{Y} = j)$, and offer attractive finite-sample performance guarantees that decay at the parametric rate of $O(1/\sqrt{n})$ up to a small bias term. However, the theoretical elegance of these MPE estimators comes at the cost of severe practical limitations. For instance, the estimator in [5, 6] is based on identifying subsets of the feature space using an exhaustive search that relies on VC dimension bounds, which are known to be loose [5]. Alternatively, the estimators in [23, 25] are limited to the binary case.

In this work, we propose to estimate the transition matrix using methods for one-sided selective classification. Standard selective classification (or classification with a reject option) [30–33] allows models to abstain when uncertain. This is a crucial capability when errors are significantly more expensive than abstaining (e.g., in medical applications [34, 35]). Formally, the goal is to learn a standard classifier f and a selection function $h\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$ [30] that together minimize the probability of misclassification on the accepted samples, $\mathbb{P}(Y \neq f(X) \mid h(X) = \mathtt{A})$, subject to a minimum coverage constraint for the selection function $\mathbb{P}(h(X) = \mathtt{A}) \geq \gamma$, where $\gamma \in (0, 1)$.

One-sided selective classification [36] simplifies the standard setting by fixing the prediction to a target class $j \in \mathcal{Y}$ (i.e., $f(x) = j, \forall x \in \mathcal{X}$). This problem arises in scenarios where it is necessary to isolate a high-purity subset of a single class, for instance to perform high-confidence auditing. The goal then reduces to learning a selection function h that minimizes the false discovery rate $\mathbb{P}(Y \neq j \mid h(X) = \mathtt{A})$ for the class $y = j$ under the coverage constraint $\mathbb{P}(h(X) = \mathtt{A}) \geq \gamma$.

3 Transition matrix estimation via one-sided selective classification

In this section, we present a novel methodology for estimating the transition matrix based on one-sided selective classification. For each class $j \in \mathcal{Y}$, we estimate the corresponding column of $\mathbf{T}$ through the following procedure.

1. Learn a selection function $h^{(j)}\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$ which minimizes the false discovery rate for the noisy label $\widetilde{y} = j$, subject to a minimum coverage constraint.

2. Estimate the j-th column of $\mathbf{T}$ by computing the empirical label probabilities of instances accepted by $h^{(j)}$.

Formally, we aim to solve for each class $j \in \mathcal{Y}$ the following problem

$$\begin{aligned} \widetilde{R}_\gamma^{(j)} = \min_h \quad & \mathbb{P}(\widetilde{Y} \neq j \mid h(X) = \mathtt{A}) \\ \text{s.t.} \quad & \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma \end{aligned} \tag{1}$$

where $\gamma \in (0, 1)$ is the minimum coverage constraint and the optimization is over all measurable selection functions $h\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$. The smallest possible false discovery rate subject to the minimum coverage constraint is denoted by $\widetilde{R}_\gamma^{(j)}$.

The estimate $\widehat{\mathbf{T}}$ is obtained using the selection functions and noisy samples as follows.

Definition 1. Let $h^{(j)}\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$ be a selection function for the j-th label, and let S be a subset of $\{1, 2, \ldots, n\}$ where $m = |S|$. For every $i \in \mathcal{Y}$ define

$$\widehat{T}_{i,j}(h^{(j)}; S) = \frac{\sum_{k \in S} \mathbb{I}\{h^{(j)}(x_k) = \mathtt{A}, \widetilde{y}_k = i\}}{\#_m(h^{(j)})} \tag{2}$$

where $\#_m(h^{(j)}) = \sum_{k \in S} \mathbb{I}\{h^{(j)}(x_k) = \mathtt{A}\}$. When the set of indices S is clear from the context we denote the estimator as $\widehat{T}_{i,j}(h^{(j)})$.

The estimator $\widehat{T}_{i,j}(h^{(j)}; S)$ is well suited to estimating the transition matrix. In particular, we have $\widehat{T}_{i,j}(h^{(j)}; S) \approx T_{i,j}$ for a near-optimal solution $h^{(j)}$ of (1). This holds because $\widehat{T}_{i,j}(h^{(j)}; S)$ is the empirical version of $\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}(X) = \mathtt{A})$, which is close to $\mathbb{P}(\widetilde{Y} = i \mid Y = j)$ for a selection function $h^{(j)}$ with small false discovery rate $\mathbb{P}(\widetilde{Y} \neq j \mid h^{(j)}(X) = \mathtt{A})$.

The next theorem shows that the error of the proposed estimator decreases as

$$|T_{i,j} - \widehat{T}_{i,j}(h^{(j)})| \lesssim \varepsilon_{\text{bias}} + \frac{1}{\sqrt{\gamma n}}$$

where γ is the minimum coverage constraint, and $\varepsilon_{\text{bias}}$ is the bias incurred due to the false discovery rate of the selection function.

Theorem 1. Let $h^{(j)} : \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$ be a selection function for the j-th label with $\mathbb{P}(h^{(j)}(X) = \mathtt{A}) \geq \gamma$, and $\varepsilon_{\text{opt}}(h^{(j)})$ be its excess false discovery rate, that is

$$\varepsilon_{\text{opt}}(h^{(j)}) = \mathbb{P}(\widetilde{Y} \neq j \mid h^{(j)}(X) = \mathtt{A}) - \widetilde{R}^{(j)}_{\gamma}.$$

If $\widehat{T}_{i,j}(h^{(j)}; S)$ for $|S| = m$ is the estimator in (2), then

$$\max_{i \in \mathcal{Y}} |T_{i,j} - \widehat{T}_{i,j}(h^{(j)}; S)| \leq C_T\big(R^{(j)}_{\gamma} + \varepsilon_{\text{opt}}(h^{(j)})\big) + \sqrt{\frac{2\log(|\mathcal{Y}|/\delta)}{\#_m(h^{(j)})}} \tag{3}$$

holds with probability at least $1 - \delta$ over the draw of the samples indexed by S, where $C_T = 2/(T_{j,j} - \max_{l \neq j} T_{j,l})$ and $R^{(j)}_{\gamma} = \min_h \{\mathbb{P}(Y \neq j \mid h(X) = \mathtt{A}) \,\text{s.t.}\, \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma\}$.

Proof. See Appendix A.1. □

The theorem above provides finite-sample performance guarantees for the proposed methodology that show a bias-variance decomposition for the error.

The bias term is given by $C_T\big(R^{(j)}_{\gamma} + \varepsilon_{\text{opt}}(h^{(j)})\big)$, where the constant C_T acts as a condition number for the transition matrix $\mathbf{T}$. This constant is inversely proportional to the margin $T_{j,j} - \max_{l \neq j} T_{j,l}$, meaning that the estimation problem is better conditioned when the diagonal dominance of $\mathbf{T}$ is more pronounced. The bound also incorporates a suboptimality gap, ε_{opt}, that accounts for the difficulty of finding the optimal selection function in (1) with finite data.

The term $R^{(j)}_{\gamma}$ is the smallest false discovery rate on clean data. As discussed at the end of this section, such smallest error is small as long as a relaxed anchor-point assumption is satisfied. In particular, the anchor-point assumption corresponds to the case where $R^{(j)}_{\gamma} \to 0$ as γ tends to 0. Therefore, Theorem 1 also provides performance guarantees for cases where the anchor-point assumption is not satisfied (i.e., $\lim_{\gamma \to 0} R^{(j)}_{\gamma} > 0$). In these cases, the limit value represents an unavoidable positive bias, while the proposed methodology remains effective as long as $R^{(j)}_{\gamma}$ takes small values.

The variance term $\sqrt{2\log(|\mathcal{Y}|/\delta)/\#_m(h^{(j)})}$ scales at the parametric rate of $O(1/\sqrt{n})$. Specifically, the minimum coverage constraint, $\mathbb{P}(h^{(j)}(X) = \mathtt{A}) \geq \gamma$, ensures that the number of accepted samples satisfies $\#_m(h^{(j)}) \gtrsim \gamma n$ with high probability.

The minimum coverage γ controls the trade-off between the bias and the variance terms described above. Decreasing γ reduces the value of $R_\gamma^{(j)}$, as a selection function that is required to accept fewer instances can provide a smaller false discovery rate. However, this reduction incurs a cost, as rejecting more samples shrinks $\#_m(h^{(j)})$, thereby increasing the variance. Choosing $\gamma = \omega(1/n)$ implies that both $R_\gamma^{(j)}$ and $1/\sqrt{\#_m(h^{(j)})}$ decrease with the number of samples n. In particular, if $R_\gamma^{(j)}$ decreases with γ as $O(\gamma^\rho)$, taking $\gamma = \Theta(n^{-1/(1+2\rho)})$ results in $R_\gamma^{(j)} + 1/\sqrt{\#_m(h^{(j)})} = O(n^{-1/2+1/(2+4\rho)})$, which is close to the parametric rate when ρ is large.

Existing theoretical results for MPE methods provide performance guarantees for estimates of the inverse flip-rates $\mathbb{P}(Y = i \mid \widetilde{Y} = j)$ that also reveal a bias-variance decomposition, with the variance term scaling as $O(1/\sqrt{n})$ [23, 25]. However, these results are limited to the binary setting and to specific learning methods. For instance, the estimator in [25] is restricted to the level sets of a fixed scoring function, while [23] is limited to kernel-based methods. In contrast, our approach addresses the estimation of transition matrices offering a more general methodology since it covers the multiclass setting and allows the usage of any method for one-sided selective classification.

To the best of our knowledge, the bounds in Theorem 1 represent the first finite-sample performance guarantees for methods that estimate the label-noise transition matrix. Using the same line of reasoning, the next result provides finite-sample guarantees for the anchor-based estimator.

Proposition 1. Let $\widehat{\boldsymbol{\eta}}^{\text{noisy}}$ be an estimate of the noisy class-posteriors $\boldsymbol{\eta}^{\text{noisy}}$. For every class $j \in \mathcal{Y}$, let $x^{(j)} \in \arg\max_{x \in \mathcal{X}} \widehat{\eta}_j^{\text{noisy}}(x)$, and define

$$\varepsilon_{\text{opt}}^{\text{anchor}}(x^{(j)}) = \mathbb{P}(\widetilde{Y} \neq j \mid X = x^{(j)}) - \widetilde{R}_0^{(j)}$$

where $\widetilde{R}_0^{(j)} = \min_{x \in \mathcal{X}} \mathbb{P}(\widetilde{Y} \neq j \mid X = x)$. Then, the anchor-based estimator $\widehat{T}_{i,j}^{\text{anchor}}(x^{(j)}) = \widehat{\eta}_i^{\text{noisy}}(x^{(j)})$ satisfies

$$\max_{i \in \mathcal{Y}} |T_{i,j} - \widehat{T}_{i,j}^{\text{anchor}}(x^{(j)})| \leq C_T(R_0^{(j)} + \varepsilon_{\text{opt}}^{\text{anchor}}(x^{(j)})) + \max_{i \in \mathcal{Y}} |\eta_i^{\text{noisy}}(x^{(j)}) - \widehat{\eta}_i^{\text{noisy}}(x^{(j)})| \quad (4)$$

where $R_0^{(j)} = \min_{x \in \mathcal{X}} \mathbb{P}(Y \neq j \mid X = x)$.

Proof. See Appendix A.2. □

A comparison between Theorem 1 and Proposition 1 indicates that the proposed methodology addresses the main limitation of the anchor-based method. More precisely, Proposition 1 shows that the anchor-based estimator suffers from the curse of dimensionality since it relies on an accurate class-posterior estimate. In particular, there are scenarios where the last term of the upper bound in (4) is exactly of the order $n^{-\alpha/(2\alpha+d)}$ [19, 20], which is a very slow rate for real-world high-dimensional datasets.

The value of $R_0^{(j)}$ quantifies deviations from the anchor-point assumption that corresponds to $R_0^{(j)}$ being equal to zero. The value $R_\gamma^{(j)}$ in Theorem 1 above plays a similar role to $R_0^{(j)}$ but providing more detailed information about the hardness of estimating the transition matrix, as described above. In particular, we have that $R_\gamma^{(j)}$ approaches $R_0^{(j)}$ when $\gamma \to 0$, and the speed with which $R_\gamma^{(j)}$ decreases with γ determines the best possible rate of decrease for the estimation error, as described above.

The suboptimality term for the anchor-based method $\varepsilon_{\text{opt}}^{\text{anchor}}$ also depends strongly on the pointwise estimation error of the class-posterior since $x^{(j)}$ is obtained as the instance with the highest class-posterior estimate. Therefore, there are scenarios where $\varepsilon_{\text{opt}}^{\text{anchor}} \asymp n^{-\alpha/(2\alpha+d)}$. In contrast, the suboptimality term for the presented methodology ε_{opt} is the excess false discovery rate in a problem of selective classification that is small under realistic assumptions. Specifically, the next section presents computationally tractable algorithms that provide selection functions with small suboptimality gap ε_{opt}.

4 Tractable algorithms with finite-sample performance guarantees

The learning methodology from Section 3 can be implemented utilizing general techniques for selective classification (also known as classification with rejection/abstention) [30, 31, 33]. In the following we present two effective approaches and provide their corresponding finite-sample performance guarantees.

4.1 Threshold selection approach

A prevalent approach within selective classification, classification with abstention, and classification with generalized metrics, is to rely on the confidence scores of a standard binary classifier [31, 37]. This approach can be adapted to our methodology for estimating the transition matrix as follows. First, split the noisy samples into two disjoint sets. Then, for each class $j \in \mathcal{Y}$, the algorithm estimates the corresponding column of $\mathbf{T}$ through the following procedure.

1. Learn a binary scoring function $s^{(j)} \colon \mathcal{X} \to \mathbb{R}$ to distinguish class $\widetilde{Y} = j$ from all others, using the samples in the first split. For every $\tau \in \mathbb{R}$, define a selection function as $h_\tau^{(j)}(x) = \mathtt{A}$ if $s^{(j)}(x) \geq \tau$, and $h_\tau^{(j)}(x) = \mathtt{R}$ otherwise.
2. Select the threshold $\widehat{\tau}$ that minimizes the empirical version of $\mathbb{P}(\widetilde{Y} \neq j \mid h_\tau^{(j)}(X) = \mathtt{A})$ (i.e., that maximizes $\widehat{T}_{j,j}(h_\tau^{(j)})$) while satisfying an empirical minimum coverage constraint, using the samples in the second split.
3. Estimate the (i,j)-th entry of $\mathbf{T}$ as $\widehat{T}_{i,j}(h_{\widehat{\tau}}^{(j)})$, using the samples in the second split.

This procedure is formalized in Algorithm 1 and theoretically analyzed in Theorem 2.

Algorithm 1 Transition matrix estimation via threshold selection

Input: $\{(x_k, \widetilde{y}_k)\}_{k=1}^n$ and lower bound for the number of accepted samples $N_+ > 0$.

Split $\{1, 2, \ldots, n\}$ into two disjoint sets, S_1 (size m_1) and S_2 (size $m_2 \geq N_+$)
for $j \in \mathcal{Y}$ **do**
  $T_{\text{temp}} \leftarrow 0$
  Learn a binary score function $s^{(j)} \colon \mathcal{X} \to \mathbb{R}$ to distinguish class $\widetilde{Y} = j$ from all others, using the samples indexed by S_1
  $(\tau_r)_{r=1}^{m_2} \leftarrow \mathtt{Sort}(\{s^{(j)}(x_k)\}_{k \in S_2})$ sort the scores in descending order
  for $r = N_+, \ldots, m_2$ **do**
    Define $h_{\tau_r}^{(j)}(x) = \mathtt{A}$ if $s^{(j)}(x) \geq \tau_r$, and $h_{\tau_r}^{(j)}(x) = \mathtt{R}$ otherwise, for every $x \in \mathcal{X}$
    if $\widehat{T}_{j,j}(h_{\tau_r}^{(j)}; S_2) \geq T_{\text{temp}}$ **then**
      $\widehat{\tau} \leftarrow \tau_r$
      $T_{\text{temp}} \leftarrow \widehat{T}_{j,j}(h_{\widehat{\tau}}^{(j)}; S_2)$
      $\widehat{T}_{i,j} \leftarrow \widehat{T}_{i,j}(h_{\widehat{\tau}}^{(j)}; S_2)$ for $i \in \mathcal{Y}$
return $\widehat{\mathbf{T}}$

In the next theorem, we show that the error of Algorithm 1 depends on the ranking excess risk of the learned score function in Step 4.

Theorem 2. Let $\widehat{\mathbf{T}}$ be the output of Algorithm 1 and fix $j \in \mathcal{Y}$. Let $\gamma = \mathbb{P}(h_{\widehat{\tau}}^{(j)}(X) = \mathtt{A})$ be the minimum coverage, that is $\gamma = \mathbb{P}(s^{(j)}(X) \geq \widehat{\tau})$.

If $\widehat{\tau}_\eta$ satisfies $\gamma = \mathbb{P}(\eta_j^{\text{noisy}}(X) \geq \widehat{\tau}_\eta) = \mathbb{P}(s^{(j)}(X) \geq \widehat{\tau})$, and we take

$$\mathcal{E}_{\text{rank}} = \mathbb{P}(\widetilde{Y} \neq j, s^{(j)}(X) \geq \widehat{\tau}, \eta_j^{\text{noisy}}(X) < \widehat{\tau}_\eta) - \mathbb{P}(\widetilde{Y} \neq j, s^{(j)}(X) < \widehat{\tau}, \eta_j^{\text{noisy}}(X) \geq \widehat{\tau}_\eta)$$

then

$$\max_{i \in \mathcal{Y}} |T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R_\gamma^{(j)} + \frac{\mathcal{E}_{\text{rank}}}{\gamma} \right) + \sqrt{\frac{1000(\log(8m_2) + \log(4|\mathcal{Y}|/\delta))}{\#_{m_2}(h_{\widehat{\tau}}^{(j)})}} \tag{5}$$

holds with probability at least $1 - \delta$ where $\#_{m_2}(h_{\widehat{\tau}}^{(j)})$ is the number of accepted samples in the second split.

Proof. See Appendix A.3 □

The theorem above shows that the error of Algorithm 1 is small when the learned score function is close to being order preserving with respect to the noisy class-posterior, i.e., $\mathcal{E}_{\text{rank}}$ is close to 0. In particular, if the score is order preserving (i.e., for every pair of instances $x, x' \in \mathcal{X}$, $s^{(j)}(x) < s^{(j)}(x') \implies \eta_j^{\text{noisy}}(x) \leq \eta_j^{\text{noisy}}(x')$) we have that $\mathcal{E}_{\text{rank}} = 0$. This follows since in that case $\{x \in \mathcal{X} : s^{(j)}(x) < \widehat{\tau}\} \subseteq \{x \in \mathcal{X} : \eta_j^{\text{noisy}}(x) < \widehat{\tau}_\eta\}$, and both sets have the same probability. Notice also that the existence of $\widehat{\tau}_\eta$ is guaranteed if $\eta_j^{\text{noisy}}(X)$ is continuously distributed.

The requirement of a low ranking excess risk $\mathcal{E}_{\text{rank}}$ is milder than assuming an accurate class-posterior estimation since commonly used classifiers (e.g., SVMs or boosted trees) are more effective at producing rankings than precise posterior probabilities [38].

The ranking excess risk $\mathcal{E}_{\text{rank}}$ decreases as m_1 grows when the score function is learned by empirical risk minimization over a hypothesis class $\mathcal{F}$. Specifically, results for the bipartite ranking problem in [39] can be used to show that the ranking excess risk satisfies

$$\mathcal{E}_{\text{rank}} \lesssim \mathcal{A}_{\text{rank}}(\mathcal{F}) + \sqrt{\frac{\mathsf{VC}(\mathcal{F}) \log(m_1)}{m_1}}$$

where $\mathcal{A}_{\text{rank}}(\mathcal{F})$ is the approximation error of the hypothesis class $\mathcal{F}$ for the ranking loss in [39] (difference between the best in class risk and the Bayes risk).

The hyperparameter N_+ serves to control the minimum coverage γ, which is of order N_+/m_2. Specifically, by Hoeffding's inequality we have that $\gamma \geq N_+/m_2 - O(1/\sqrt{m_2})$ with high probability. In practice, choosing N_+ involves a trade-off since it should be large enough to accept a sufficient number of instances (to reduce the variance), while remaining small enough to maintain a low false discovery rate (to minimize the bias). We defer a detailed discussion on the effect of N_+ to Appendix B.

In the following we present another algorithm for estimating $\mathbf{T}$ that satisfies further refined finite-sample guarantees.

4.2 Cost-sensitive risk minimization approach

A common approach in selective classification involves penalizing misclassifications and assigning a tunable cost to rejections [33, 40]. This idea can be adapted to our methodology for estimating the transition matrix as follows. First, split the noisy samples into two disjoint sets. Then, for each class $j \in \mathcal{Y}$, the algorithm estimates the corresponding column of $\mathbf{T}$ through the following procedure.

1. Learn a binary classifier $h_c^{(j)} \colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$ where $\mathtt{A}$ targets $\widetilde{Y} = j$ and $\mathtt{R}$ targets $\widetilde{Y} \neq j$, by penalizing the misclassification of the class $\widetilde{Y} = j$ and assigning a cost $c \in (0, 1)$ to all instances classified as $\mathtt{R}$. That is, we learn $h_c^{(j)}$ by minimizing the loss

$$\ell_c^{(j)}(h(x), \widetilde{y}) = \mathbb{I}\{\widetilde{y} \neq j, h(x) = \mathtt{A}\} + c\mathbb{I}\{h(x) = \mathtt{R}\}. \tag{6}$$

2. Select the cost $\widehat{c}$ that minimizes the empirical version of $\mathbb{P}(\widetilde{Y} \neq j \mid h_c^{(j)}(X) = \mathtt{A})$ (i.e., that maximizes $\widehat{T}_{j,j}(h_c^{(j)})$) while satisfying an empirical minimum coverage constraint, using the samples in the second split.

3. Estimate the (i, j)-th entry of $\mathbf{T}$ as $\widehat{T}_{i,j}(h_{\widehat{c}}^{(j)})$, using the samples in the second split.

This procedure is formalized in Algorithm 2 and theoretically analyzed in Theorem 3, where we prove that the error of Algorithm 2 depends on the (classification) excess risk of the solution to the cost-sensitive binary classification problem in Step 6 of Algorithm 2.

Algorithm 2 Transition matrix estimation via cost-sensitive risk minimization

Input: $\{(x_k, \widetilde{y}_k)\}_{k=1}^n$, grid spacing ε and lower bound for the number of accepted samples $N_+ > 0$.

1: Split $\{1, 2, \ldots, n\}$ into two disjoint sets, S_1 (size m_1) and S_2 (size $m_2 \geq N_+$)
2: **for** $j \in \mathcal{Y}$ **do**
3: $\quad T_{\text{temp}} \leftarrow 0$
4: $\quad$ **for** $q = 1, 2, \ldots, \lfloor 1/\varepsilon \rfloor$ **do**
5: $\quad\quad c \leftarrow \varepsilon q$
6: $\quad\quad$ Learn a cost-sensitive binary classifier $h_c^{(j)}$ with costs given by c as in (6), using the samples indexed by S_1
7: $\quad\quad$ **if** $\#_{m_2}(h_c^{(j)}) \geq N_+$ **and** $\widehat{T}_{j,j}(h_c^{(j)}; S_2) \geq T_{\text{temp}}$ **then**
8: $\quad\quad\quad \widehat{c} \leftarrow c$
9: $\quad\quad\quad T_{\text{temp}} \leftarrow \widehat{T}_{j,j}(h_{\widehat{c}}^{(j)}; S_2)$
10: $\quad\quad\quad \widehat{T}_{i,j} \leftarrow \widehat{T}_{i,j}(h_{\widehat{c}}^{(j)}; S_2)$ for $i \in \mathcal{Y}$
11: **return** $\widehat{\mathbf{T}}$

Theorem 3. Let $\widehat{\mathbf{T}}$ be the output of Algorithm 2 and fix $j \in \mathcal{Y}$. Let $\gamma = \mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{A})$ be the minimum coverage. If $h_{\widehat{c}}^{(j)}$ is the classifier corresponding with the output for column j and we take

$$\mathcal{E}_{\text{cost}} = \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h_{\widehat{c}}^{(j)}(X), \widetilde{Y})] - \inf_h \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h(X), \widetilde{Y})]$$

then,

$$\max_{i \in \mathcal{Y}} |T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R_\gamma^{(j)} + \frac{\mathcal{E}_{\text{cost}}}{\gamma} \right) + \sqrt{\frac{2 \log(|\mathcal{Y}|/(\varepsilon\delta))}{\#_{m_2}(h_{\widehat{c}}^{(j)})}} \tag{7}$$

holds with probability at least $1 - \delta$ where $\#_{m_2}(h_{\widehat{c}}^{(j)})$ is the number of accepted samples in the second split.

Proof. See Appendix A.4. □

Theorem 3 shows that the error of Algorithm 2 is small, provided that the binary cost-sensitive classification problem in Step 6 is solved with a small excess risk $\mathcal{E}_{\text{cost}}$. This requirement is significantly milder than that of a pointwise accurate class-posterior estimate, and the small ranking excess risk required by the thresholding approach in Theorem 2. In particular, achieving a small excess risk corresponds to an accurate classification of instances, whereas the thresholding approach additionally requires correct ranking of instances' probabilities.

In the next theorem, we provide performance guarantees for Algorithm 2 in terms of the excess risk of a convex surrogate loss and the ψ-transform of the surrogate from [41, Def. 2]. Specifically, let Φ be a classification-calibrated convex surrogate of the 0-1 loss $t \mapsto \mathbb{I}\{t < 0\}$ such as the hinge loss ($\Phi(t) = \max\{0, 1-t\}$) or the logistic loss ($\Phi(t) = \log(1+e^{-t})$) [41, Def. 1], we define the convex surrogate of $\ell_c^{(j)}$ as

$$L_c^{(j)}(f(x), \widetilde{y}) = \mathbb{I}\{\widetilde{y} \neq j\}\Phi(-f(x)) + c\Phi(f(x))$$

where f is a binary score function.

Theorem 4. Let $\widehat{\mathbf{T}}$ be the output of Algorithm 2 assuming that the classifiers learned in Step 6 take the form $h_c^{(j)}(x) = \operatorname{sign}(f_c^{(j)}(x))$ for $x \in \mathcal{X}$ with $f_c^{(j)}$ a binary score function. Fix $j \in \mathcal{Y}$, let $\gamma = \mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{A})$ be the minimum coverage and let

$$\mathcal{E}_\Phi = \mathbb{E}[L_{\widehat{c}}^{(j)}(f_{\widehat{c}}^{(j)}(X), \widetilde{Y})] - \inf_f \mathbb{E}[L_{\widehat{c}}^{(j)}(f(X), \widetilde{Y})]$$

be the excess risk of the convex surrogate loss. Then

$$\max_{i \in \mathcal{Y}} |T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R_\gamma^{(j)} + \frac{2\psi_\Phi^{-1}(\mathcal{E}_\Phi/2)}{\gamma} \right) + \sqrt{\frac{2 \log(|\mathcal{Y}|/(\varepsilon\delta))}{\#_{m_2}(h_{\widehat{c}}^{(j)})}} \tag{8}$$

holds with probability at least $1 - \delta$.

Proof. See Appendix A.5. □

Theorem 4 guarantees that the error of Algorithm 2 remains small as long as the learning technique used in Step 6 successfully minimizes a classification-calibrated convex surrogate loss. Moreover, reducing the excess risk of the convex surrogate loss to zero makes the suboptimality gap $2\psi_\Phi^{-1}(\mathcal{E}_\Phi/2)/\gamma$ converge to zero since the ψ_Φ-transform satisfies $\psi_\Phi^{-1}(t) \to 0$ when $t \to 0$ as shown in [41, Thm. 1].

In practice, the learning in Step 6 is performed by minimizing the empirical risk of the convex surrogate loss $L_c^{(j)}$ over a hypothesis class $\mathcal{F}$. By standard generalization bounds [42], we have $\mathcal{E}_\Phi \lesssim \mathcal{A}_{\text{surrogate}}(\mathcal{F}) + O(\sqrt{\mathsf{VC}(\mathcal{F})\log(m_1/\mathsf{VC}(\mathcal{F}))/m_1})$ where $\mathcal{A}_{\text{surrogate}}(\mathcal{F}) = \inf_{f\in\mathcal{F}} \mathbb{E}[L_{\widehat{c}}^{(j)}(f(X),\widetilde{Y})] - \inf_f \mathbb{E}[L_{\widehat{c}}^{(j)}(f(X),\widetilde{Y})]$ is the approximation error of the hypothesis class $\mathcal{F}$ for the convex surrogate loss $L_{\widehat{c}}^{(j)}$.

The choice of the convex surrogate Φ specifies the ψ_Φ^{-1}-transform, which determines the convergence rate of the term $\psi_\Phi^{-1}(\mathcal{E}_\Phi/2)$. For the hinge loss ($\Phi(t) = \max\{0, 1-t\}$), ψ_Φ is simply the identity function [41]. Therefore, the excess risk bound scales linearly, yielding

$$\max_{i\in\mathcal{Y}} |T_{i,j} - \widehat{T}_{i,j}| \lesssim R_\gamma^{(j)} + \mathcal{A}_{\text{surrogate}}(\mathcal{F}) + \sqrt{\frac{\mathsf{VC}(\mathcal{F})\log(m_1/\mathsf{VC}(\mathcal{F}))}{m_1}} + \sqrt{\frac{\log(|\mathcal{Y}|/\varepsilon)}{m_2}}.$$

For the logistic loss we have $\psi_\Phi^{-1}(t) \lesssim \sqrt{t}$ [41], which introduces a square-root dependence on the excess risk resulting in a slightly slower convergence rate.

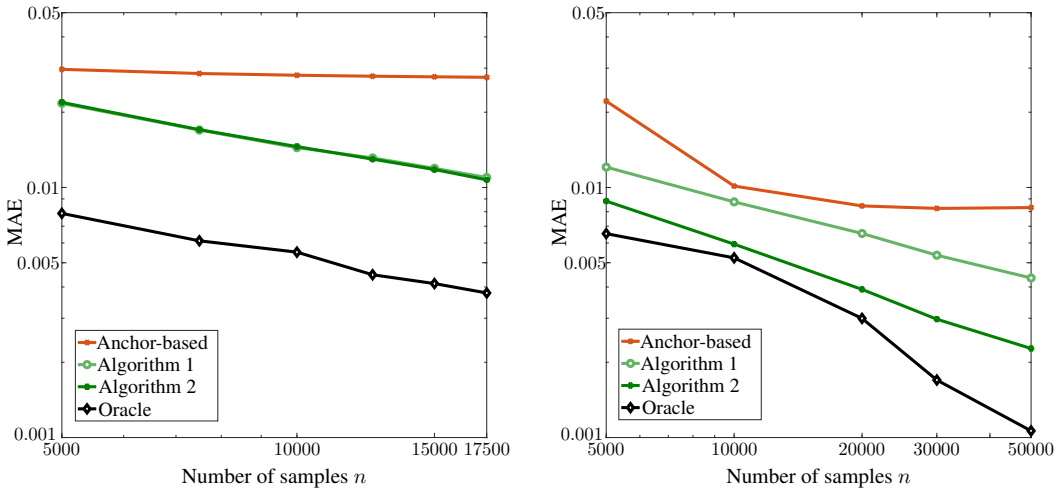
The results above show that the methodology presented can be implemented using flexible and effective techniques for selective classification. In particular, the theoretical results show that the proposed approach can result in significantly improved error rates that quickly decrease with the number of samples. In the following, we provide numerical results that illustrate such improved performance using real-world datasets.

4.3 Experimental results

In what follows, we provide experimental results that illustrate the theoretical results presented above. Specifically, we show that the proposed algorithms achieve significantly lower error rates than methods that rely on pointwise estimates of class-posteriors and that the error rates of the proposed algorithms converge at an order comparable to that of a strong oracle. Furthermore, in Appendix C.2 we provide additional experimental results on running times and dimensionality's impact. The code implementing the methods presented and reproducing the experiments can be found at `https://github.com/MachineLearningBCAM/label-noise-T-estimation-NeurIPS2026`.

Figure 1 shows how the error of different algorithms decreases with the number of samples n. We implement Algorithms 1 and 2, and the anchor-based method from [8] with random forests for the Letter dataset (Figure 1(a)) and neural networks for the MNIST dataset (Figure 1(b)). Both Algorithms 1 and 2 are executed with $N_+ = n/200$ and $m_1 = m_2 = n/2$. Moreover, we evaluate the algorithms against a strong oracle that learns $\mathbf{T}$ via maximum likelihood estimation (MLE). To act as a strong benchmark reference, the oracle has access to feature representations from a clean-data pre-trained network and initializes the MLE optimization directly at the ground-truth $\mathbf{T}$. Therefore, this oracle has a very small error corresponding to the finite-sample estimation error due to finite n. Further details regarding the implementation and additional experimental results are provided in Appendix C.

In Figure 1 we show that the proposed algorithms achieve significantly lower error rates than existing methods that rely on pointwise estimates of the class-posteriors, in accordance with our derived performance bounds. Moreover, the figure shows that the errors of both the proposed algorithms and the oracle exhibit the same decreasing trend as n increases, whereas the error of the anchor-based method plateaus.



(a) Letter dataset

(b) MNIST dataset

Figure 1: Mean Absolute Error (MAE) of different estimators as the number of training samples n increases. The errors of the presented algorithms decrease with the number of samples in a similar trend to the error of the oracle, whereas the error of the anchor-based method stagnates.

5 Conclusion

The paper presents a new methodology for the estimation of the label-noise transition matrix based on one-sided selective classification. In the proposed approach, the transition matrix is estimated by learning selection functions that minimize the false discovery rate subject to a minimum coverage constraint. We provide finite-sample error bounds for the proposed methodology and show that our approach avoids the curse of dimensionality, unlike existing estimators. Moreover, we propose effective algorithms that can be implemented with any method for binary classification, and provide their corresponding performance guarantees. The presented methodology can help to reduce the impact of label noise on supervised learning methods, making model deployment more cost-efficient by allowing the use of data labeled through inexpensive sources.

Limitations: The proposed methodology is broadly effective but does not scale well with the number of classes, as it requires solving different binary classification problems for each class to estimate the transition matrix. Additionally, like other estimators, the approach may struggle under class imbalance, which can affect the precision of the selection functions for minority classes. These drawbacks are of limited relevance in practice because the sub-problems are naturally parallelizable and can be solved using standard off-the-shelf methods. Furthermore, as noisy labels are inexpensive to acquire, the assumption of a sufficiently large training sample is typically satisfied in real-world applications, ensuring the approach remains both computationally and statistically viable.

Acknowledgments

Xabier de Juan and Santiago Mazuelas were supported by project PID2022-137063NB-I00 funded by MCIN/AEI/10.13039/501100011033 and the European Union "NextGenerationEU"/PRTR, BCAM Severo Ochoa accreditation CEX2021-001142-S/MICIN/AEI/10.13039/501100011033 funded by the Ministry of Science and Innovation (Spain), and programs BERC-2022–2025 and ELKARTEK funded by the Basque Government. Yilun Zhu and Clayton Scott were supported in part by the National Science Foundation under award 2551503, and by the Department of Defense, Defense Threat Reduction Agency under award HDTRA1-20-2-0002. Xabier de Juan acknowledges a predoctoral grant from the Basque Government.

References

[1] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8135–8153, 2023.

[2] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Conference on Computer Vision and Pattern Recognition*, 2015.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[4] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in Neural Information Processing Systems*, 2013.

[5] Clayton Scott. A rate of convergence for mixture proportion estimation, with application to learning from noisy labels. In *International Conference on Artificial Intelligence and Statistics*, 2015.

[6] Gilles Blanchard, Marek Flaska, Gregory Handy, Sara Pozzi, and Clayton Scott. Classification with asymmetric label noise: Consistency and maximal denoising. *Electronic Journal of Statistics*, 10(2):2780–2824, 2016.

[7] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 447–461, 2016.

[8] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Conference on Computer Vision and Pattern Recognition*, 2017.

[9] Lucia Filippozzi, Santiago Mazuelas, and Iñigo Urteaga. Minimax risk classifiers for mislabeled data: a study on patient outcome prediction tasks. In *Machine Learning for Healthcare Conference*, 2024.

[10] Matteo Sesia, Y. X. Rachel Wang, and Xin Tong. Adaptive conformal classification with noisy labels. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 87(3):796–815, 2024.

[11] Haoran Zhang, Olawale Elijah Salaudeen, and Marzyeh Ghassemi. On group sufficiency under label bias. In *Advances in Neural Information Processing Systems*, 2025.

[12] Byeonghu Na, Yeongmin Kim, HeeSun Bae, Jung Hyun Lee, Jung Se Kwon, Wanmo Kang, and Il-chul Moon. Label-noise robust diffusion models. In *International Conference on Learning Representations*, 2024.

[13] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? In *Advances in Neural Information Processing Systems*, 2019.

[14] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. In *Advances in Neural Information Processing Systems*, 2020.

[15] Yivan Zhang, Gang Niu, and Masashi Sugiyama. Learning noise transition matrix from only noisy labels via total variation regularization. In *International Conference on Machine Learning*, 2021.

[16] Xuefeng Li, Tongliang Liu, Bo Han, Gang Niu, and Masashi Sugiyama. Provably end-to-end label-noise learning without anchor points. In *International Conference on Machine Learning*, 2021.

[17] Mingyuan Zhang, Jane Lee, and Shivani Agarwal. Learning from noisy labels with no change to the training process. In *International Conference on Machine Learning*, 2021.

[18] Yong Lin, Renjie Pi, Weizhong Zhang, Xiaobo Xia, Jiahui Gao, Xiao Zhou, Tongliang Liu, and Bo Han. A holistic view of label noise transition matrix in deep learning and beyond. In *International Conference on Learning Representations*, 2023.

[19] László Györfi, Michael Kohler, Adam Krzyżak, and Harro Walk. *A Distribution-Free Theory of Nonparametric Regression*. Springer, New York, United States, 2002.

[20] Jing Lei. Classification with confidence. *Biometrika*, 101(4):755–769, 2014.

[21] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, 2017.

[22] Gilles Blanchard and Clayton Scott. Decontamination of mutually contaminated models. In *International Conference on Artificial Intelligence and Statistics*, 2014.

[23] Harish Ramaswamy, Clayton Scott, and Ambuj Tewari. Mixture proportion estimation via kernel embeddings of distributions. In *International Conference on Machine Learning*, 2016.

[24] Julian Katz-Samuels, Gilles Blanchard, and Clayton Scott. Decontamination of mutual contamination models. *Journal of Machine Learning Research*, 20(41):1–57, 2019.

[25] Saurabh Garg, Yifan Wu, Alexander J Smola, Sivaraman Balakrishnan, and Zachary Lipton. Mixture proportion estimation and PU learning: A modern approach. In *Advances in Neural Information Processing Systems*, 2021.

[26] Yilun Zhu, Aaron Fjeldsted, Darren Holland, George Landon, Azaree Lintereur, and Clayton Scott. Mixture proportion estimation beyond irreducibility. In *International Conference on Machine Learning*, 2023.

[27] Yilun Zhu, Jianxin Zhang, Aditya Gangrade, and Clayton Scott. Label noise: Ignorance is bliss. In *Advances in Neural Information Processing Systems*, 2024.

[28] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in Neural Information Processing Systems*, 2018.

[29] Philip M. Long and Rocco A. Servedio. The perils of being unhinged: On the accuracy of classifiers minimizing a noise-robust convex loss. *Neural Computation*, 34(6):1488–1499, 2022.

[30] Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(53):1605–1641, 2010.

[31] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, 2017.

[32] Peter L. Bartlett and Marten H. Wegkamp. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(59):1823–1840, 2008.

[33] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *International Conference on Algorithmic Learning Theory*, 2016.

[34] Blaise Hanczar and Edward R. Dougherty. Classification with reject option in gene expression data. *Bioinformatics*, 24(17):1889–1895, 2008.

[35] Akshay Swaminathan, Ivan Lopez, William Wang, Ujwal Srivastava, Edward Tran, Aarohi Bhargava-Shah, Janet Y Wu, Alexander L Ren, Kaitlin Caoili, Brandon Bui, Layth Alkhani, Susan Lee, Nathan Mohit, Noel Seo, Nicholas Macedo, Winson Cheng, Charles Liu, Reena Thomas, Jonathan H Chen, and Olivier Gevaert. Selective prediction for extracting unstructured clinical data. *Journal of the American Medical Informatics Association*, 31(1):188–197, 2024.

[36] Aditya Gangrade, Anil Kag, and Venkatesh Saligrama. Selective classification via one-sided prediction. In *International Conference on Artificial Intelligence and Statistics*, 2021.

[37] Oluwasanmi Koyejo, Nagarajan Natarajan, Pradeep Ravikumar, and Inderjit S. Dhillon. Consistent binary classification with generalized performance metrics. In *Advances in Neural Information Processing Systems*, 2014.

[38] Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *International Conference on Machine Learning*, 2006.

[39] Stéphan Clémençon, Gábor Lugosi, and Nicolas Vayatis. Ranking and empirical minimization of U-statistics. *Annals of Statistics*, 36(2):844–874, 2008.

[40] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Theory and algorithms for learning with rejection in binary classification. *Annals of Mathematics and Artificial Intelligence*, 92(2):277–315, 2024.

[41] Peter L. Bartlett, Michael I. Jordan, and Jon D. Mcauliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

[42] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, Cambridge, United States, 2018.

[43] Akshay Balsubramani, Sanjoy Dasgupta, Yoav Freund, and Shay Moran. An adaptive nearest neighbor rule for classification. In *Advances in Neural Information Processing Systems*, 2019.

[44] Dheeru Dua and Casey Graff. UCI machine learning repository. `http://archive.ics.uci.edu`, 2017.

[45] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[46] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Computer Science Department, University of Toronto, 2009.

[47] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

A Proofs

The proofs shown below utilize the following lemma that provides general upper bounds for the estimation error.

Lemma 1. Fix $i, j \in \mathcal{Y}$ and let $\mathcal{F}$ be a family of measurable subsets of $\mathcal{X}$. Then, for every $A \in \mathcal{F}$

$$|T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R_{\mathcal{F}}^{(j)} + \sup_{A' \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A') - \mathbb{P}(\widetilde{Y} = j \mid X \in A) \right) + |\mathbb{P}(\widetilde{Y} = i \mid X \in A) - \widehat{T}_{i,j}|$$

where $C_T = 2/(T_{j,j} - \max_{l \neq j} T_{j,l})$ and $R_{\mathcal{F}}^{(j)} = \inf_{A' \in \mathcal{F}} \mathbb{P}(Y \neq j \mid X \in A')$.

Proof. By the triangle inequality we get

$$|T_{i,j} - \widehat{T}_{i,j}| \leq |T_{i,j} - \mathbb{P}(\widetilde{Y} = i \mid X \in A)| + |\mathbb{P}(\widetilde{Y} = i \mid X \in A) - \widehat{T}_{i,j}|.$$

We apply Hölder's inequality in the first term so that

$$\begin{aligned} |T_{i,j} - \mathbb{P}(\widetilde{Y} = i \mid X \in A)| &= \left| T_{i,j} - \sum_{l=1}^{|\mathcal{Y}|} T_{i,l} \mathbb{P}(Y = l \mid X \in A) \right| \\ &= |(T_{i,1}, T_{i,2}, \ldots, T_{i,|\mathcal{Y}|})^{\mathrm{T}} (e_j - \mathbb{P}(\boldsymbol{Y} \mid X \in A))| \\ &\leq \|e_j - \mathbb{P}(\boldsymbol{Y} \mid X \in A)\|_1 = 2(1 - \mathbb{P}(Y = j \mid X \in A)) \end{aligned}$$

where e_j denotes the j-th vector from the canonical basis and $\mathbb{P}(\boldsymbol{Y} \mid X \in A)$ is the vector whose t-th component is $\mathbb{P}(Y = t \mid X \in A)$.

Using the definition of C_T, we have

$$1 - \mathbb{P}(Y = j \mid X \in A) \leq \frac{C_T}{2} (T_{j,j} - \mathbb{P}(\widetilde{Y} = j \mid X \in A)). \tag{9}$$

Such a bound can be obtained as follows.

$$\begin{aligned} \mathbb{P}(\widetilde{Y} = j \mid X \in A) &= \sum_{l=1}^{|\mathcal{Y}|} T_{j,l} \mathbb{P}(Y = l \mid X \in A) \\ &= T_{j,j} \mathbb{P}(Y = j \mid X \in A) + \sum_{l=1, l \neq j}^{|\mathcal{Y}|} T_{j,l} \mathbb{P}(Y = l \mid X \in A). \end{aligned} \tag{10}$$

Therefore,

$$\begin{aligned} \mathbb{P}(Y = j \mid X \in A) &= \frac{\mathbb{P}(\widetilde{Y} = j \mid X \in A) - \sum_{l=1, l \neq j}^{|\mathcal{Y}|} T_{j,l} \mathbb{P}(Y = l \mid X \in A)}{T_{j,j}} \\ &\geq \frac{\mathbb{P}(\widetilde{Y} = j \mid X \in A) - \max_{l \neq j} \{T_{j,l}\} (1 - \mathbb{P}(Y = j \mid X \in A))}{T_{j,j}}. \end{aligned}$$

Thus,

$$\left(1 - \frac{\max_{l \neq j} T_{j,l}}{T_{j,j}}\right) \mathbb{P}(Y = j \mid X \in A) \geq \frac{\mathbb{P}(\widetilde{Y} = j \mid X \in A) - \max_{l \neq j} \{T_{j,l}\}}{T_{j,j}}$$

whence (9) follows since $\max_{l \neq j} \{T_{j,l}\} / T_{j,j} < 1$.

The right hand side in (9) can be written as

$$\begin{aligned} T_{j,j} - \mathbb{P}(\widetilde{Y} = j \mid X \in A) = T_{j,j} &- \sup_{A' \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A') \\ &+ \sup_{A' \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A') - \mathbb{P}(\widetilde{Y} = j \mid X \in A). \end{aligned}$$

Then, the result is obtained because

$$T_{j,j} - \sup_{A' \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A') \leq 1 - \sup_{A' \in \mathcal{F}} \mathbb{P}(Y = j \mid X \in A') = R_{\mathcal{F}}^{(j)}$$

since

$$\sup_{A' \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A') \geq T_{j,j} \sup_{A' \in \mathcal{F}} \mathbb{P}(Y = j \mid X \in A')$$

as a consequence of (10), and we also have

$$T_{j,j} \sup_{A' \in \mathcal{F}} \mathbb{P}(Y = j \mid X \in A') \geq T_{j,j} + \sup_{A' \in \mathcal{F}} \mathbb{P}(Y = j \mid X \in A') - 1$$

since $(T_{j,j} - 1)\big(\sup_{A' \in \mathcal{F}} \mathbb{P}(Y = j \mid X \in A') - 1\big) \geq 0$. □

A.1 Proof of Theorem 1

The result is a particular case of Lemma 1 taking

$$\mathcal{F} = \{A_h \subseteq \mathcal{X} : h \colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\} \,\text{s.t.}\, \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma\}$$

where $A_h = \{x \in \mathcal{X} : h(x) = \mathtt{A}\}$ is the set of accepted instances of a selection function $h \colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$, and $A = \{x \in \mathcal{X} : h^{(j)}(x) = \mathtt{A}\}$. Therefore, Lemma 1 implies

$$|T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R_{\gamma}^{(j)} + \varepsilon_{\text{opt}}(h^{(j)})\right) + |\mathbb{P}(\widetilde{Y} = i \mid X \in A) - \widehat{T}_{i,j}|$$

since $R_{\gamma}^{(j)} = R_{\mathcal{F}}^{(j)}$ and $\varepsilon_{\text{opt}}(h^{(j)}) = \sup_{A_h \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A_h) - \mathbb{P}(\widetilde{Y} = j \mid X \in A)$.

Finally, the desired result follows since for each $i \in \mathcal{Y}$

$$|\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}(X) = \mathtt{A}) - \widehat{T}_{i,j}(h^{(j)})| \leq \sqrt{\frac{2\log(1/\delta)}{\#_m(h^{(j)})}} \tag{11}$$

with probability at least $1 - \delta$ (over the draw of the samples in S). To obtain (11), note that conditioned on the event that the number of accepted samples is $k = \#_m(h^{(j)})$, the random variable $k\widehat{T}_{i,j}(h^{(j)})$ follows a binomial distribution with parameters k and probability $\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}(X) = \mathtt{A})$. Then, the bound in (11) follows by the Chernoff bound conditionally on that event, and using the law of total probability varying the values of k.

Finally, taking a union bound over the $|\mathcal{Y}|$ classes we obtain

$$\max_{i \in \mathcal{Y}} |\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}(X) = \mathtt{A}) - \widehat{T}_{i,j}(h^{(j)})| \leq \sqrt{\frac{2\log(|\mathcal{Y}|/\delta)}{\#_m(h^{(j)})}}$$

that holds with probability at least $1 - \delta$. □

A.2 Proof of Proposition 1

The result is proven using Lemma 1 with $\mathcal{F} = \{\{x\} : x \in \mathcal{X}\}$ and $A = \{x^{(j)}\}$. Therefore,

$$|T_{i,j} - \widehat{T}_{i,j}^{\text{anchor}}(x^{(j)})| \leq C_T \left(R_0^{(j)} + \varepsilon_{\text{opt}}^{\text{anchor}}(x^{(j)})\right) + |\eta_i^{\text{noisy}}(x^{(j)}) - \widehat{\eta}_i^{\text{noisy}}(x^{(j)})|$$

since $R_0^{(j)} = R_{\mathcal{F}}^{(j)}$ and $\varepsilon_{\text{opt}}^{\text{anchor}}(x^{(j)}) = \sup_{A' \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A') - \mathbb{P}(\widetilde{Y} = j \mid X \in A)$. □

A.3 Proof of Theorem 2

The result is proven using Lemma 1 with

$$\mathcal{F} = \{A_h \subseteq \mathcal{X} : h \colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\} \,\text{s.t.}\, \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma\}$$

where $A_h = \{x \in \mathcal{X} : h(x) = \mathtt{A}\}$ is the set of accepted instances of a selection function $h\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$, and $A = \{x \in \mathcal{X} : h^{(j)}_{\widehat{\tau}}(x) = \mathtt{A}\}$. Moreover, we have $A \in \mathcal{F}$ since $\gamma = \mathbb{P}(h^{(j)}_{\widehat{\tau}}(X) = \mathtt{A})$ by hypothesis. Therefore, Lemma 1 implies

$$|T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R^{(j)}_{\gamma} + \varepsilon_{\text{opt}}(h^{(j)}_{\widehat{\tau}})\right) + |\mathbb{P}(\widetilde{Y} = i \mid X \in A) - \widehat{T}_{i,j}|$$

since $R^{(j)}_{\gamma} = R^{(j)}_{\mathcal{F}}$ and $\varepsilon_{\text{opt}}(h^{(j)}_{\widehat{\tau}}) = \sup_{A_h \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A_h) - \mathbb{P}(\widetilde{Y} = j \mid X \in A)$. Moreover, since $\widehat{T}_{i,j}$ is the empirical version of $\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}_{\widehat{\tau}}(X) = \mathtt{A})$, the uniform convergence of empirical conditional measures (Theorem 8 in [43]) implies

$$|\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}_{\widehat{\tau}}(X) = \mathtt{A}) - \widehat{T}_{i,j}| \leq \sqrt{\frac{1000(\log(8m_2) + \log(4/\delta))}{\#_{m_2}(h^{(j)}_{\widehat{\tau}})}}$$

with probability at least $1 - \delta$ (over the draw of the samples indexed by S_2). Taking a union bound over the $|\mathcal{Y}|$ classes we obtain

$$\max_{i \in \mathcal{Y}} |\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}(X) = \mathtt{A}) - \widehat{T}_{i,j}(h^{(j)})| \leq \sqrt{\frac{1000(\log(8m_2) + \log(4|\mathcal{Y}|/\delta))}{\#_{m_2}(h^{(j)}_{\widehat{\tau}})}}$$

with probability at least $1 - \delta$ (over the draw of the samples indexed by S_2).

Then, the result is obtained by bounding the suboptimality gap

$$\varepsilon_{\text{opt}}(h^{(j)}_{\widehat{\tau}}) = \mathbb{P}(\widetilde{Y} \neq j \mid h^{(j)}_{\widehat{\tau}}(X) = \mathtt{A}) - \min_{h} \{\mathbb{P}(\widetilde{Y} \neq j \mid h(X) = \mathtt{A}) \,\text{s.t.}\, \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma\}.$$

Note that $\widetilde{R}^{(j)}_{\gamma}$ is achieved by thresholding the noisy class-posterior at $\widehat{\tau}_{\eta}$. Therefore,

$$\begin{aligned}
\varepsilon_{\text{opt}}(h^{(j)}_{\widehat{\tau}}) &= \mathbb{P}(\widetilde{Y} \neq j \mid s^{(j)}(X) \geq \widehat{\tau}) - \mathbb{P}(\widetilde{Y} \neq j \mid \eta^{\text{noisy}}_j(X) \geq \widehat{\tau}_{\eta}) \\
&= \frac{1}{\gamma}(\mathbb{P}(\widetilde{Y} \neq j, s^{(j)}(X) \geq \widehat{\tau}) - \mathbb{P}(\widetilde{Y} \neq j, \eta^{\text{noisy}}_j(X) \geq \widehat{\tau}_{\eta})).
\end{aligned}$$

where the last equality holds since $\mathbb{P}(h^{(j)}_{\widehat{\tau}}(X) = \mathtt{A}) = \gamma = \mathbb{P}(\eta^{\text{noisy}}_j(X) \geq \widehat{\tau}_{\eta})$. Finally, it is easy to check that

$$\begin{aligned}
\mathbb{P}(\widetilde{Y} \neq j, s^{(j)}(X) \geq \widehat{\tau}) - \mathbb{P}(\widetilde{Y} \neq j, \eta^{\text{noisy}}_j(X) \geq \widehat{\tau}_{\eta}) &= \mathbb{P}(\widetilde{Y} \neq j, s^{(j)}(X) \geq \widehat{\tau}, \eta^{\text{noisy}}_j(X) < \widehat{\tau}_{\eta}) \\
&\quad - \mathbb{P}(\widetilde{Y} \neq j, s^{(j)}(X) < \widehat{\tau}, \eta^{\text{noisy}}_j(X) \geq \widehat{\tau}_{\eta}) \\
&= \mathcal{E}_{\text{rank}}.
\end{aligned}$$

□

A.4 Proof of Theorem 3

The result is proven using Lemma 1. Specifically, let

$$\mathcal{F} = \{A_h \subseteq \mathcal{X} : h\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\} \,\text{s.t.}\, \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma\}$$

where $A_h = \{x \in \mathcal{X} : h(x) = \mathtt{A}\}$ is the set of accepted instances of a selection function $h\colon \mathcal{X} \to \{\mathtt{A}, \mathtt{R}\}$, and let $A = \{x \in \mathcal{X} : h^{(j)}_{\widehat{c}}(x) = \mathtt{A}\}$. Moreover, we have $A \in \mathcal{F}$ since $\gamma = \mathbb{P}(h^{(j)}_{\widehat{c}}(X) = \mathtt{A})$ by hypothesis. Therefore, Lemma 1 implies

$$|T_{i,j} - \widehat{T}_{i,j}| \leq C_T \left(R^{(j)}_{\gamma} + \varepsilon_{\text{opt}}(h^{(j)}_{\widehat{c}})\right) + |\mathbb{P}(\widetilde{Y} = i \mid X \in A) - \widehat{T}_{i,j}|$$

since $R^{(j)}_{\gamma} = R^{(j)}_{\mathcal{F}}$ and $\varepsilon_{\text{opt}}(h^{(j)}_{\widehat{c}}) = \sup_{A_h \in \mathcal{F}} \mathbb{P}(\widetilde{Y} = j \mid X \in A_h) - \mathbb{P}(\widetilde{Y} = j \mid X \in A)$.

Moreover, by applying the Chernoff bound conditionally, for every c in the grid

$$\max_{i \in \mathcal{Y}} |\mathbb{P}(\widetilde{Y} = i \mid h^{(j)}_{c}(X) = \mathtt{A}) - \widehat{T}_{i,j}(h^{(j)}_{c}; S_2)| \leq \sqrt{\frac{2\log(|\mathcal{Y}|/\delta)}{\#_{m_2}(h^{(j)}_{c})}} \tag{12}$$

holds with probability at least $1-\delta$. To obtain (12), note that conditioned on the event that the number of accepted samples is $k = \#_{m_2}(h_c^{(j)})$, the random variable $k\widehat{T}_{i,j}(h_c^{(j)})$ follows a binomial distribution with parameters k and probability $\mathbb{P}(\widetilde{Y} = i \mid h_c^{(j)}(X) = \mathtt{A})$. Then, the bound in (12) follows by the Chernoff bound conditionally on that event, and using the law of total probability varying the values of k.

Therefore, taking a union bound over the grid of size $\lfloor 1/\varepsilon \rfloor$, we obtain for $c = \widehat{c}$

$$\max_{i \in \mathcal{Y}} |\mathbb{P}(\widetilde{Y} = i \mid h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \widehat{T}_{i,j}| \leq \sqrt{\frac{2\log(|\mathcal{Y}|/(\delta\varepsilon))}{\#_{m_2}(h_{\widehat{c}}^{(j)})}}$$

with probability at least $1-\delta$ (over the draw of the samples indexed by S_2).

Then, the result is obtained by bounding the suboptimality gap

$$\varepsilon_{\text{opt}}(h_{\widehat{c}}^{(j)}) = \mathbb{P}(\widetilde{Y} \neq j \mid h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \min_h \{\mathbb{P}(\widetilde{Y} \neq j \mid h(X) = \mathtt{A}) \,\text{s.t.}\, \mathbb{P}(h(X) = \mathtt{A}) \geq \gamma\}.$$

Let $h_*^{(j)}$ be a solution of the one-sided selective classification problem in (1) such that $\mathbb{P}(h_*^{(j)}(X) = \mathtt{A}) = \gamma$. Then,

$$\begin{aligned}
\varepsilon_{\text{opt}}(h_{\widehat{c}}^{(j)}) &= \mathbb{P}(\widetilde{Y} \neq j \mid h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \mathbb{P}(\widetilde{Y} \neq j \mid h_*^{(j)}(X) = \mathtt{A}) \\
&= \frac{1}{\gamma}(\mathbb{P}(\widetilde{Y} \neq j, h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \mathbb{P}(\widetilde{Y} \neq j, h_*^{(j)}(X) = \mathtt{A})).
\end{aligned}$$

where the last equality holds since $\mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) = \gamma = \mathbb{P}(h_*^{(j)}(X) = \mathtt{A})$. Finally, a simple calculation yields

$$\begin{aligned}
&\mathbb{P}(\widetilde{Y} \neq j, h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \mathbb{P}(\widetilde{Y} \neq j, h_*^{(j)}(X) = \mathtt{A}) \\
&\quad = \mathbb{P}(\widetilde{Y} \neq j, h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \mathbb{P}(\widetilde{Y} \neq j, h_*^{(j)}(X) = \mathtt{A}) \\
&\qquad + \widehat{c}(\mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{R}) - \mathbb{P}(h_*^{(j)}(X) = \mathtt{R})) \\
&\qquad - \widehat{c}(\mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{R}) - \mathbb{P}(h_*^{(j)}(X) = \mathtt{R})) \\
&\quad = \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h_{\widehat{c}}^{(j)}(X), \widetilde{Y})] - \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h_*^{(j)}(X), \widetilde{Y})] \\
&\qquad + \widehat{c}(\mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) - \mathbb{P}(h_*^{(j)}(X) = \mathtt{A})) \\
&\quad \leq \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h_{\widehat{c}}^{(j)}(X), \widetilde{Y})] - \inf_h \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h(X), \widetilde{Y})] \\
&\quad = \mathcal{E}_{\text{cost}}
\end{aligned}$$

where in the last inequality we have used that $\mathbb{P}(h_{\widehat{c}}^{(j)}(X) = \mathtt{A}) = \gamma = \mathbb{P}(h_*^{(j)}(X) = \mathtt{A})$ and $\inf_h \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h(X), \widetilde{Y})] \leq \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h_*^{(j)}(X), \widetilde{Y})]$. □

A.5 Proof of Theorem 4

The result is a consequence of Theorem 3. Then, the result is obtained by establishing the bound

$$\begin{aligned}
2\psi_\Phi\left(\frac{\mathbb{E}[\ell_{\widehat{c}}^{(j)}(h_{\widehat{c}}^{(j)}(X), \widetilde{Y})] - \inf_h \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h(X), \widetilde{Y})]}{2}\right) &\leq \mathbb{E}[L_{\widehat{c}}^{(j)}(f_{\widehat{c}}^{(j)}(X), \widetilde{Y})] \\
&\quad - \inf_f \mathbb{E}[L_{\widehat{c}}^{(j)}(f(X), \widetilde{Y})]
\end{aligned} \tag{13}$$

which we prove in what follows.

Let $x \in \mathcal{X}$ be arbitrary and for notational convenience let $\eta(x) = \mathbb{P}(\widetilde{Y} = j \mid X = x)$. We define the conditional $L_{\widehat{c}}^{(j)}$-risk as

$$C_\Phi(x; \alpha) \coloneqq \mathbb{E}[L_{\widehat{c}}^{(j)}(\alpha, \widetilde{Y}) \mid X = x], \quad \alpha \in \mathbb{R}.$$

A simple calculation produces

$$\begin{aligned} C_\Phi(x;\alpha) &= (1-\eta(x))\Phi(-\alpha) + \widehat{c}\Phi(\alpha) \\ &= S(x)\left((1-\eta'(x))\Phi(-\alpha) + \eta'(x)\Phi(\alpha)\right) \end{aligned} \tag{14}$$

where $S(x) = 1 - \eta(x) + \widehat{c}$ and $\eta'(x) = \widehat{c}/S(x) \in [0,1]$. Now define the conditional $\ell_{\widehat{c}}^{(j)}$-risk as

$$C(x;a) \coloneqq \mathbb{E}[\ell_{\widehat{c}}^{(j)}(a,\widetilde{Y}) \mid X = x], \quad a \in \{-1,+1\}$$

where for notational convenience $\ell_{\widehat{c}}^{(j)}(a,\widetilde{Y})$ is well defined for $a \in \{-1,1\}$ (we identify `A` with $+1$ and `R` with -1). By [41, Sec. 2.2] we have

$$\inf_f \mathbb{E}[L_{\widehat{c}}^{(j)}(f(X),\widetilde{Y})] = \mathbb{E}_X\left[\inf_{\alpha\in\mathbb{R}} C_\Phi(X;\alpha)\right], \tag{15}$$

$$\inf_h \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h(X),\widetilde{Y})] = \mathbb{E}_X\left[\inf_{a\in\{-1,+1\}} C(X;a)\right]. \tag{16}$$

Define the excess conditional risks

$$\begin{aligned} \Delta C_\Phi(x;\alpha) &\coloneqq C_\Phi(x;\alpha) - \inf_{\alpha\in\mathbb{R}} C_\Phi(x;\alpha), \\ \Delta C(x;a) &\coloneqq C(x;a) - \inf_{a\in\{-1,+1\}} C(x;a). \end{aligned}$$

A simple calculation yields

$$\Delta C(x;a) = |\eta(x) - (1-\widehat{c})|\mathbb{I}\{a \neq \operatorname{sign}(\eta(x) - (1-\widehat{c}))\}. \tag{17}$$

Now we proceed with the proof of (13) for an arbitrary score function $f^{(j)}$ and its corresponding binary classifier $h^{(j)}$, i.e., $h^{(j)}(x) = \operatorname{sign}(f^{(j)}(x))$.

Case 1: $\operatorname{sign}(2\eta'(x)-1) = \operatorname{sign}(f^{(j)}(x))$. We have $\operatorname{sign}(2\eta'(x)-1) = \operatorname{sign}(\eta(x) - (1-\widehat{c}))$, and thus by (17) we get

$$\Delta C(x; h^{(j)}(x)) = |\eta(x) - (1-\widehat{c})|\mathbb{I}\{\operatorname{sign}(f^{(j)}(x)) \neq \operatorname{sign}(\eta(x) - (1-\widehat{c}))\} = 0.$$

Since the ψ_Φ-transform satisfies $\psi_\Phi(0) = 0$ as shown in [41, Lemma 2], the bound

$$\Delta C_\Phi(x; f^{(j)}(x)) \geq 0 = 2\psi_\Phi\left(\frac{\Delta C(x; h^{(j)}(x))}{2}\right)$$

trivially holds.

Case 2: $\operatorname{sign}(2\eta'(x)-1) \neq \operatorname{sign}(f^{(j)}(x))$. For $t \in [0,1]$ we define

$$\begin{aligned} H(t) &= \inf_{\alpha\in\mathbb{R}} \left((1-t)\Phi(-\alpha) + t\Phi(\alpha)\right), \\ H^-(t) &= \inf_{\alpha \,\text{s.t.}\, \alpha(2t-1)\leq 0} \left((1-t)\Phi(-\alpha) + t\Phi(\alpha)\right) \end{aligned}$$

following [41, Def. 1]. Then, since $\operatorname{sign}(2\eta'(x)-1) \neq \operatorname{sign}(f^{(j)}(x))$, we have

$$\begin{aligned} \Delta C_\Phi(x; f^{(j)}(x)) &\geq S(x)(H^-(\eta'(x)) - H(\eta'(x))) && \text{(By (14))} \\ &= S(x)\psi_\Phi(2\eta'(x)-1) && \text{(definition of } \psi\text{, [41, Def. 2])} \\ &= S(x)\psi_\Phi(|2\eta'(x)-1|) && \text{(symmetry of } \psi\text{, [41, Lemma 2])} \\ &= S(x)\psi_\Phi\left(\frac{|\eta(x)-(1-\widehat{c})|}{S(x)}\right) && \left(|2\eta'(x)-1| = \frac{|\eta(x)-(1-\widehat{c})|}{S(x)}\right) \\ &= S(x)\psi_\Phi\left(\frac{\Delta C(x;h^{(j)}(x))}{S(x)}\right) && (|\eta(x)-(1-\widehat{c})| = \Delta C(x;h^{(j)}(x))) \\ &= 2\frac{S(x)}{2}\psi_\Phi\left(\frac{\Delta C(x;h^{(j)}(x))}{S(x)}\right) \\ &\geq 2\psi_\Phi\left(\frac{\Delta C(x;h^{(j)}(x))}{S(x)}\frac{S(x)}{2}\right) && (S(x)\leq 2 \text{ and } \psi_\Phi \text{ is convex}) \\ &= 2\psi_\Phi\left(\frac{\Delta C(x;h^{(j)}(x))}{2}\right). \end{aligned}$$

Therefore, we have shown that for every $x \in \mathcal{X}$,

$$\Delta C_\Phi(x; f^{(j)}(x)) \geq 2\psi_\Phi\left(\frac{\Delta C(x; h^{(j)}(x))}{2}\right).$$

Finally, since ψ_Φ is convex, by Jensen's inequality we obtain

$$\mathbb{E}_X[\Delta C_\Phi(X; f^{(j)}(X))] \geq \mathbb{E}_X\left[2\psi_\Phi\left(\frac{\Delta C(X; h^{(j)}(X))}{2}\right)\right] \geq 2\psi_\Phi\left(\frac{\mathbb{E}_X[\Delta C(X; h^{(j)}(X))]}{2}\right)$$

which proves (13) because

$$\mathbb{E}_X[\Delta C_\Phi(X; f^{(j)}(X))] = \mathbb{E}[L_{\widehat{c}}^{(j)}(f^{(j)}(X), \widetilde{Y})] - \inf_f \mathbb{E}[L_{\widehat{c}}^{(j)}(f(X), \widetilde{Y})]$$

and

$$\mathbb{E}_X[\Delta C(X; h^{(j)}(X))] = \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h^{(j)}(X), \widetilde{Y})] - \inf_h \mathbb{E}[\ell_{\widehat{c}}^{(j)}(h(X), \widetilde{Y})]$$

hold by (15) and (16), respectively. □

B Analysis of the effect of the number of accepted samples

In this appendix, we provide an analysis on the effect of the hyperparameter N_+ for Algorithms 1 and 2. In practice, the optimal choice of the lower bound for the number of accepted samples N_+ must balance the need for a sufficiently large number of accepted instances (to reduce the variance of the estimator) with the requirement of maintaining a low false discovery rate (to minimize the bias).

For Algorithm 1, we analyze the impact of N_+ by observing the empirical objective function $\widehat{T}_{j,j}(h_{\tau_k}^{(j)}; S_2)$ as a function of the number of accepted samples $k \in \{1, 2, \ldots, m_2\}$.

When plotting $(k, \widehat{T}_{j,j}(h_{\tau_k}^{(j)}; S_2))$, we typically observe three distinct regimes.

1. When the number of accepted samples is very small, the empirical probability $\widehat{T}_{j,j}(h_{\tau_k}^{(j)}; S_2)$ fluctuates due to high variance.

2. As the number of accepted samples increases, the variance decreases, and $\widehat{T}_{j,j}(h_{\tau_k}^{(j)}; S_2)$ stabilizes at a constant value which is close to the true transition probability $T_{j,j}$. Here, the corresponding selection functions accept enough instances to provide a low-variance estimate without accepting samples belonging to other true classes $y \neq j$, thereby keeping the false discovery rate small.

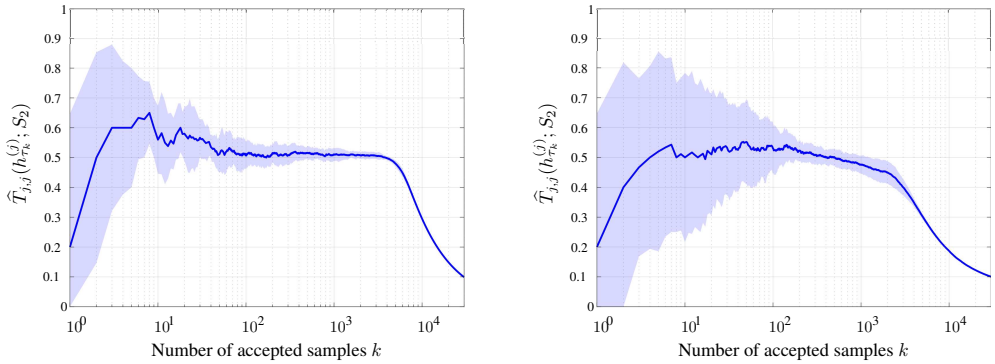
3. If the number of accepted samples increases beyond the plateau, the corresponding selection functions are forced to accept samples with labels $y \neq j$. As a consequence, the estimate $\widehat{T}_{j,j}(h_{\tau_k}^{(j)}; S_2)$ drops sharply from $T_{j,j}$.

Choosing the maximum number of accepted samples inside the plateau guarantees the lowest possible variance for the estimator while ensuring that the bias remains controlled.

We provide experimental results on the CIFAR-10 dataset under different noise scenarios that illustrate the three regimes (see Figure 2), and we provide further implementation details in Appendix C.

In Figure 3 we show that when the number of instances is small, the size of the plateau is smaller and there is more variability, leading to higher estimation errors.

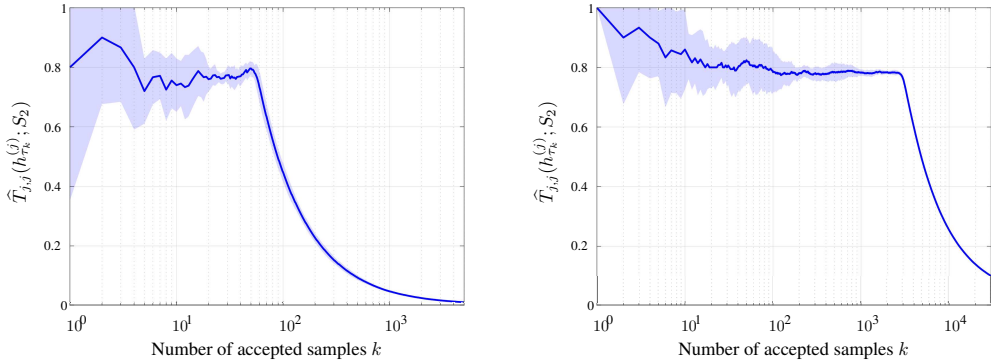
For Algorithm 2, we perform a similar analysis by plotting, for $c \in \{0.05, 0.1, \ldots, 0.95\}$, $(c, \widehat{T}_{j,j}(h_c^{(j)}; S_2))$ (blue line) and the number of accepted samples is plotted $(c, \#_{m_2}(h_c^{(j)}))$ (red line) since the number of accepted samples $\#_{m_2}(h_c^{(j)})$ typically increases monotonically with c. In Figure 4 we plot the corresponding graphs for the Satellite dataset.



(a) CIFAR-10 under uniform noise with $T_{j,j} = 0.5$.

(b) CIFAR-10 under flip noise with $T_{j,j} = 0.55$.

Figure 2: The three regimes described above appear on CIFAR-10. Moreover, the value of the plateau is close to $T_{j,j}$.



(a) CIFAR-10 using 10,000 samples.

(b) CIFAR-10 using 60,000 samples.

Figure 3: The value of the plateau is close to $T_{j,j} = 0.8$ and the noise is uniform. Moreover, with fewer samples the plateau is shorter and noisier.

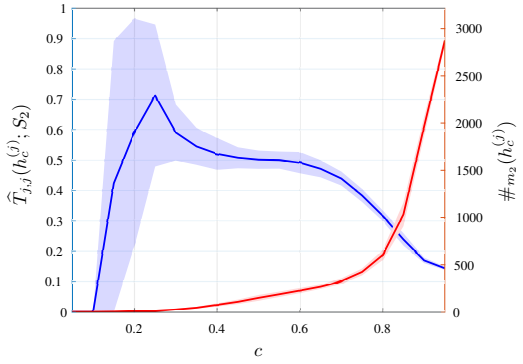


Figure 4: The value of the plateau is close to $T_{j,j} = 0.5$ and the noise is uniform.

Again, choosing the highest number of accepted samples inside the plateau ensures a low variance for the estimator while guaranteeing that the bias is small.

The existence and variability of the plateau is determined by the quality of the learned score functions or the binary classifiers in Algorithms 1 and 2, respectively.

C Implementation details and additional experimental results

In the following we provide further implementation details and describe the datasets used in the experimental results. Then, we complement the results in the main paper by including a table that compares the performance of the algorithms presented in the paper with the anchor-based method.

C.1 Implementation details

Datasets We evaluate our methodology on four publicly available benchmark datasets from the UCI repository [44] and Kaggle website `https://www.kaggle.com`. The main characteristics of the datasets used are summarized below.

- **Letter:** 20,000 samples, 26 classes, 16 features.
- **Satellite:** 6,435 samples, 6 classes, 36 features.
- **MNIST:** 70,000 samples, 10 classes, 784 features [45].
- **CIFAR-10:** 60,000 samples, 10 classes, 3,072 features [46].

Learned binary classifiers For the tabular datasets (Letter and Satellite), we employ Random Forests with 200 maximum splits, and a minimum leaf size of 5. For MNIST, we utilize a multilayer perceptron with two hidden layers of sizes 128 and 64, trained with a regularization parameter of 0.001. For CIFAR-10, we perform logistic regression using features extracted from the final layer of a pre-trained ConvNeXt based vision foundation model [47].

Label noise We generate the noise synthetically, following standard practices in the literature [16, 18]. Specifically, for a noise ratio $p \in (0, 1)$, we consider two deterministic noise regimes.

- p-uniform noise: The transition matrix $\mathbf{T}$ is defined such that for every column $j \in \mathcal{Y}$

$$T_{j,j} = 1 - p,\ T_{i,j} = \frac{p}{|\mathcal{Y}| - 1}, \quad \forall i \neq j.$$

- p-flip noise: $T_{1,1} = 1 - p$, $T_{2,1} = p$ and for every $j \in \{2, 3, \ldots, |\mathcal{Y}|\}$,

$$T_{j,j} = 1 - p,\ T_{j-1,j} = p.$$

For example, for Figure 1(a) we consider 0.5-uniform noise and for Figure 1(b) 0.2-label flip.

Implementation of the anchor-based method We implement the anchor-based method described in Section 2.2 by utilizing the 97th percentile of the noisy class-posterior rather than the maximum value, as suggested in the original work [8].

Implementation of the proposed algorithms For Algorithms 1 and 2 we split the samples into two sets of equal size ($m_1 = m_2 = n/2$).

Implementation of the oracle The oracle for the plots in Section 4.3 is implemented as follows. First, a feature representation $\varphi(x) \in \mathbb{R}^D$ is obtained from a network trained on clean labels, with a bias coordinate appended to the representation. Specifically, for the Letter dataset, we use the activations of the final hidden layer of the clean-data MLP (with two hidden layers of sizes 128 and 64), while for MNIST we use the pretrained ConvNeXt representations [47]. Given these features, we parametrize the clean class-posterior as $\widehat{\eta}(x; \mu) = \mathrm{softmax}(\varphi(x)^{\mathrm{T}}\mu)$ where $\mu \in \mathbb{R}^D$. Then, we jointly estimate the classifier parameters and transition matrix by minimizing the negative log-likelihood

$$\min_{\mu, \widehat{\mathbf{T}}} - \sum_{k=1}^{n} \log \left[\sum_{j \in \mathcal{Y}} \widehat{T}_{\widetilde{y}_k, j} \widehat{\eta}_j(x_k; \mu) \right]$$

subject to $\widehat{\mathbf{T}}$ being column-stochastic. The optimization is initialized with μ equal to the all-ones vector and $\widehat{\mathbf{T}}$ at the ground-truth transition matrix $\mathbf{T}$.

Number of repetitions The results presented at Section 4.3 (Figure 1) are averaged over 50 independent trials for each sample size n. In Appendix B, we report the mean over 100 repetitions, where the shaded regions denote the standard deviation. Finally, for the results in Appendix C.2, we report the mean and standard deviation using the full datasets over 5 repetitions for Tables 1 and 2, whereas for Table 3 we consider 20 repetitions.

C.2 Additional experimental results

Transition matrix estimation error Tables 1 and 2 report the Mean Absolute Error (MAE) of the estimated transition matrices under 0.2-uniform noise and 0.45-flip noise, respectively. In Table 3, we introduce noise in a more realistic and challenging way. Specifically, we consider asymmetric label-noise transition matrices with column-wise uniform noise where the noise parameter of each column follows a different value randomly sampled from an Unif$(0, 1/2)$ distribution.

We compare our proposed algorithms (Algorithm 1 and Algorithm 2) against the standard anchor-based method across multiple datasets. To evaluate how the choice of the underlying model impacts estimation, we employ both highly expressive classifiers (e.g., Neural Networks and Random Forests) and weaker, linear methods (Logistic Regression). For the high-dimensional CIFAR-10 dataset, we utilize a strong Neural Network architecture.

Table 1: MAE of different estimators under 0.2-uniform noise.

	CIFAR-10	MNIST		Letter		Satellite	
		NN	Logistic reg.	RF	Logistic reg.	RF	Logistic reg.
Anchor-based	$.009 \pm .001$	$.010 \pm .001$	$.016 \pm .001$	$.044 \pm .000$	$.047 \pm .000$	$.022 \pm .001$	$\mathbf{.067 \pm .023}$
Algorithm 1	$.008 \pm .001$	$.008 \pm .001$	$\mathbf{.007 \pm .000}$	$\mathbf{.005 \pm .000}$	$\mathbf{.021 \pm .001}$	$\mathbf{.019 \pm .002}$	$.068 \pm .010$
Algorithm 2	$\mathbf{.006 \pm .001}$	$\mathbf{.004 \pm .000}$	$.012 \pm .001$	$.006 \pm .000$	$.029 \pm .001$	$.021 \pm .004$	$.085 \pm .003$

Table 2: MAE of different estimators under 0.45-flip noise.

	CIFAR-10	MNIST		Letter		Satellite	
		NN	Logistic reg.	RF	Logistic reg.	RF	Logistic reg.
Anchor-based	$.029 \pm .011$	$.016 \pm .009$	$\mathbf{.023 \pm .003}$	$.046 \pm .001$	$.048 \pm .002$	$\mathbf{.039 \pm .011}$	$.085 \pm .056$
Algorithm 1	$.036 \pm .004$	$.033 \pm .003$	$.033 \pm .004$	$\mathbf{.014 \pm .001}$	$\mathbf{.031 \pm .000}$	$.044 \pm .005$	$\mathbf{.066 \pm .003}$
Algorithm 2	$\mathbf{.015 \pm .005}$	$\mathbf{.008 \pm .003}$	$.037 \pm .006$	$\mathbf{.014 \pm .001}$	$.039 \pm .001$	$.045 \pm .004$	$.083 \pm .003$

Table 3: MAE of different estimators under random asymmetric uniform noise.

	CIFAR-10	MNIST		Letter		Satellite	
		NN	Logistic reg.	RF	Logistic reg.	RF	Logistic reg.
Anchor-based	$.009 \pm .001$	$.011 \pm .001$	$.017 \pm .002$	$.041 \pm .001$	$.045 \pm .001$	$.021 \pm .003$	$.102 \pm .065$
Algorithm 1	$.009 \pm .003$	$.008 \pm .001$	$\mathbf{.008 \pm .001}$	$\mathbf{.006 \pm .001}$	$\mathbf{.021 \pm .001}$	$\mathbf{.020 \pm .002}$	$\mathbf{.068 \pm .010}$
Algorithm 2	$\mathbf{.006 \pm .001}$	$\mathbf{.005 \pm .001}$	$.011 \pm .003$	$.007 \pm .000$	$.029 \pm .001$	$.021 \pm .003$	$.089 \pm .011$

As observed across the three noise regimes, our proposed methods outperform the anchor-based baseline on almost all datasets. As suggested by theoretical results, the anchor-based estimator suffers fundamentally from the difficulty of accurately estimating the class-posterior pointwise. In contrast, our results empirically validate that accurate transition matrix estimation does not require full pointwise class-posterior recovery. Instead, by framing the problem through one-sided selective classification, our methodology only requires learning a sufficiently capable binary classifier.

Furthermore, comparing Table 1 and Table 2 illustrates the practical impact of the theoretical constant C_T derived in our finite-sample bounds. The 0.45-flip noise setting is significantly more challenging than the 0.2-uniform noise setting, which translates to a larger C_T. In accordance with our theory, this increased difficulty causes all evaluated methods to achieve a higher absolute error under flip noise. Nonetheless, our methods, particularly Algorithm 2 when implemented with strong classifiers like NNs on CIFAR-10 and MNIST, demonstrate good performance despite the harsh noise setting.

Table 3 shows that under a more challenging and realistic setting, the different methods present a similar behavior to the scenario from Table 1. Furthermore, in this more demanding setting, on every dataset, at least one of the proposed algorithms outperforms the anchor-based baseline.

Effect of the dimensionality In Figure 5, we show how the error of the different algorithms increases with the number of dimensions d. Specifically, we apply PCA to the ConvNeXt features of CIFAR-10, reducing to dimension $d \in \{100, 250, 500, 1000\}$, and train a logistic regression classifier on the reduced features under uniform 0.2-noise, with the number of training samples fixed at $n = 5000$.

The error of the anchor-based method grows with d, consistent with the theoretical discussion in the paper. In contrast, the error of Algorithms 1 and 2 remain essentially flat across the tested range of d, in line with Theorem 1's error bound, which has no explicit dependence on d.

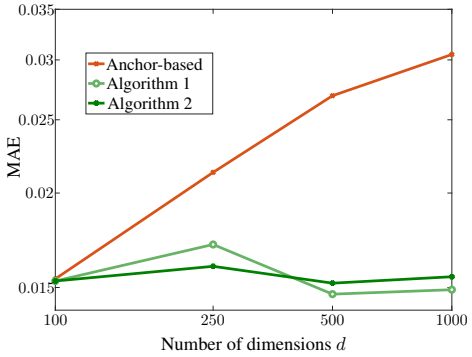


Figure 5: The presented methods remain insensitive to the dimension, whereas the error of the anchor-based method increases with the dimension.

Running times The computational cost of the presented algorithms increases with the number of classes because the number of learned binary classifiers grows linearly with $|\mathcal{Y}|$. Nevertheless, this increase is not a major concern because each classifier is trained independently for each class, allowing the learning tasks to be run in parallel.

In Figure 6, we report the median wall-clock training time (in seconds) over 10 executions of each algorithm on the Letter dataset (26 classes) using 26 CPUs.

Algorithm 1's runtime is comparable to the anchor-based method's, consistent with each class's binary problem being solved in parallel using one CPU per class. Algorithm 2 requires an additional grid search over $\lfloor 1/\varepsilon \rfloor = 20$ cost values per class ($\varepsilon = 0.05$), i.e., 520 binary problems in total. With class-level parallelization, Algorithm 2's measured runtime is at most 43 times that of Algorithm 1 across the different sample sizes, somewhat above the naive estimate of $20\times$ from grid size alone, reflecting implementation overheads. Warm-starting across adjacent grid values could further reduce this gap.

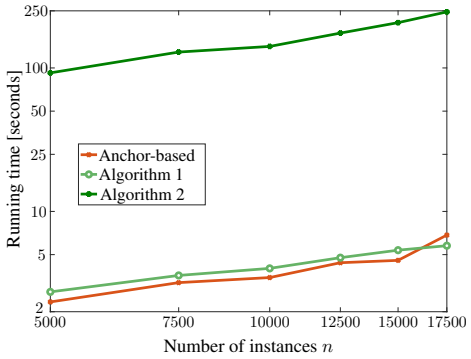


Figure 6: The proposed algorithms have low computational overhead in practice as an execution requires at most a few minutes to run.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction accurately describe the paper contributions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are discussed over the paper. In particular the theorems state the hypothesis required.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.

- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The detailed proofs appear in the appendices and the theorems state all the assumptions made.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.

   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.

   - All assumptions should be clearly stated or referenced in the statement of any theorems.

   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.

   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.

   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The experimentation carried out in the paper is described in Section 4 and detailed in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.

- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code and data is available in the Github repository `https://github.com/MachineLearningBCAM/label-noise-T-estimation-NeurIPS2026`

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.

   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.

   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental settings/details are described in detail in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The experimental results in Appendix B provide error bars in terms of standard deviations over different random partitions of the data. The table in Appendix C includes the standard deviations for all experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The experimental results in the paper can be carried out in a regular desktop machine in few hours,.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted conforms with such Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: The paper presents foundational research that is not tied to particular applications

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper presents foundational research that is not tied to particular applications/datasets.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper refers to all the relevant previous works.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The code provided is documented and easy to use.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: [N/A]

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.