Rejection via Learning Density Ratios

Alexander Soen[†]
Amazon
The Australian National University
alexander.soen@anu.edu.au

Hisham Husain[⋆]
hisham.husain@protonmail.com

Philip Schulz
Amazon
phschulz@amazon.com

Vu Nguyen
Amazon
vutngn@amazon.com

Abstract

Classification with rejection emerges as a learning paradigm which allows models to abstain from making predictions. The predominant approach is to alter the supervised learning pipeline by augmenting typical loss functions, letting model rejection incur a lower loss than an incorrect prediction. Instead, we propose a different distributional perspective, where we seek to find an idealized data distribution which maximizes a pretrained model's performance. This can be formalized via the optimization of a loss's risk with a φ-divergence regularization term. Through this idealized distribution, a rejection decision can be made by utilizing the density ratio between this distribution and the data distribution. We focus on the setting where our φ-divergences are specified by the family of α-divergence. Our framework is tested empirically over clean and noisy datasets.

1 Introduction

Forcing Machine Learning (ML) models to always make a prediction can lead to costly consequences. Indeed, in real-world domains such as automated driving, product inspection, and medical diagnosis, inaccurate prediction can cause significant real-world harm [16, 29, 53, 50]. To deal with such a dilemma, selective prediction and classification with rejection were proposed to modify the standard supervised learning setting [15, 17, 72]. The idea is to allow for a model to explicitly reject making a prediction whenever the underlying prediction would be either inaccurate and / or uncertain.

In classification, a *confidence-based* approach can be utilized, where a classifier is trained to output a "margin" which is used as a confidence score for rejection [7, 31, 71, 59, 53]. The model rejects whenever this confidence score is lower than an assigned threshold value. A key aspect of these approaches is that they rely on good probability estimates $\Pr(\mathsf{Y} \mid \mathsf{X} = x)$ [60], *i.e.*, being calibrated [71, 53]. While some approaches avoid explicit probability estimation, these are typically restricted to binary classification [7, 31, 47]. Empirically, approaches utilizing confidence scores have shown to outperform other methods, even with simple plugin estimates for probabilities [26, 39, 58].

Another *classifier-rejection* approach aims to simultaneously train a prediction and rejection model in tandem [16, 17, 53]. These approaches are theoretically driven by the construction of surrogate loss functions, but in the multiclass classification case many of these loss functions have been shown to not be suitable [53]. For the multiclass classification setting, one approach connects classification with rejection to cost-sensitive classification [14]. In practice, these classification-rejection approaches require models to be trained from scratch using their specific loss function and architecture — if there

(†) Work done while interning at Amazon.
(*) Work done while employed at Amazon.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

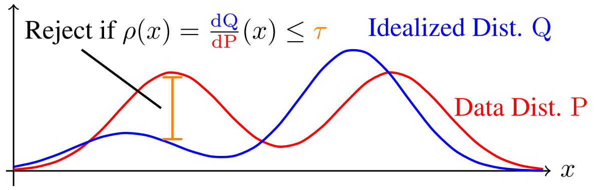
Reject if $\rho(x) = \frac{\mathrm{dQ}}{\mathrm{dP}}(x) \leq \tau$
Idealized Dist. Q
Data Dist. P
x

Figure 1: An *idealized distribution* Q is learned to *minimizes* the loss of a model. We then compare Q with the original data distribution P via a density ratio $\rho = \mathrm{dQ}/\mathrm{dP}$. A rejection criteria is defined via threshold value τ.

Algorithm 1 Density-Ratio Rejection

```
Require: Model $h$; Divergence $\alpha \geq 1$;
    Reg. $\lambda > 0$; Threshold $\tau \in [0, 1]$.
Calibrate $h$ if needed.
Calculate density ratio $\rho^{\alpha}_{\lambda}$
    (either Corollary 4.1 or 4.6).
Normalize $\rho^{\alpha}_{\lambda}$ with Monte-Carlo or
    Eq. (17).
Calculate rejector $r_\tau$ via Def. 3.2.
output $r_\tau$
```

is an existing classifier for the dataset, it must be discarded. A recent approach proposes to learn a post-hoc rejector on top of a pretrained classifier via surrogate loss functions [48].

In this work, to learn rejectors we shift from a loss function perspective to a *distributional* perspective. Given a model and a corresponding loss function, we find a distribution where the model and loss *performs "best"* and compare it against the data input distribution to make rejection decisions (see Fig. 1 and Algorithm 1 with equations boxed). As such, the set of rejectors that we propose creates a rejection decision by considering the *density ratio* [67] between a "best" case (idealized) distribution and the data distribution, which can be thresholded by different values τ to provide different accuracy vs rejection percentage trade-offs. To learn these density ratios for rejection, we consider a risk minimization problem which is regularized by φ-divergences [1, 19]. We study a particular type of φ-divergences as our regularizer: the family of α-divergences which generalizes the KL-divergence. To this end, one of our core contributions in this work is providing various methods for constructing and approximating idealized distributions, in particular those constructed by α-divergences.

The idealized distributions that we consider are connected to adversarial distributions examined in *Distributionally Robust Optimization* (DRO) [63] and the distributions learned in *Generalized Variational Inference* (GVI) [42]; and, as such, the closed formed solutions found for our idealized distribution have utility outside of rejection. Furthermore, when utilizing the KL-divergence and Bayes optimal models, we recover the well known optimal rejection policies, *i.e.*, Chow's rule [15, 72]. Our rejectors are then examined empirically on 6 different datasets and under label noise corruption.

In summary, our contributions are the following:

- We present a new framework for learning with rejection involving the density ratios of idealized distributions which mirrors the distributions learned in DRO and GVI;
- We show that rejection policies learned in our framework can theoretically recover optimal rejection policies, *i.e.*, Chow's rule;
- We derive optimal idealized distributions generated from α-divergences;
- We present a set of simplifying assumptions such that our framework can be utilized in practice for post-hoc rejection and verify this empirically.

2 Preliminaries

Notation Let $\mathcal{X}$ be a domain of inputs and $\mathcal{Y}$ be output targets. We primarily consider the case where $\mathcal{Y}$ is a finite set of N labels $[N]$, where $[N] \doteq \{0, \ldots, N-1\}$. We also consider output domains $\mathcal{Y}'$ which is not necessarily the same as the output data $\mathcal{Y}$, *e.g.*, class probabilities estimates in classification. Denote the underlying (usually unknown) input distribution as $\mathrm{P}_{\mathrm{x,y}} \in \mathcal{D}(\mathcal{X} \times \mathcal{Y})$. The marginals of a joint distribution are denoted by subscript, *e.g.*, $\mathrm{P}_{\mathrm{y}} \in \mathcal{D}(\mathcal{Y})$ for the marginal on the label space. We denote conditional distributions, *e.g.*, $\mathrm{P}_{\mathrm{x|y}}$. For marginals on $\mathcal{D}(\mathcal{X})$, the subscripting of x is implicit with $\mathrm{P} = \mathrm{P}_{\mathrm{x}}$. The empirical distributions of P are denoted by $\hat{\mathrm{P}}_N$, with N denoting the number of samples used to generate it. Denote the Iverson bracket $[\![p]\!] = 1$ if the predicate p is true and $[\![p]\!] = 0$ otherwise [43]. The maximum $[z]_+ \doteq \max\{0, z\}$ is shorthanded.

Learning with Rejection We first recall the standard *risk minimization* setting for learning. Suppose that we are given a (point-wise) loss function $\ell \colon \mathcal{Y} \times \mathcal{Y}' \to \mathbb{R}_{\geq 0}$ which measures the level of disagreement between predictions and data. Given a data distribution $\mathrm{P}_{\mathrm{x,y}}$ and a hypothesis set of models $h \in \mathcal{H}$, we aim to minimize the expected risk w.r.t. $h \colon \mathcal{X} \to \mathcal{Y}'$,

$$\operatorname*{argmin}_{h \in \mathcal{H}} \; \mathbb{E}_{\mathrm{P}_{\mathrm{x,y}}}[\ell(\mathsf{Y}, h(\mathsf{X}))]. \tag{1}$$

One way of viewing learning with rejection is to augment the risk minimization framework by learning an additional rejection function $r \colon \mathcal{X} \to \{0,1\}$. This can be formally defined using a regularization term $c \in \mathbb{R}_{\geq 0}$ which controls the rate of rejection:

$$\operatorname*{argmin}_{(h,r) \in \mathcal{H} \times \mathcal{R}} \; \mathbb{E}_{\mathrm{P}_{\mathrm{x,y}}}[(1 - r(\mathsf{X}))\ell(\mathsf{Y}, h(\mathsf{X}))] + c \cdot \mathrm{P}[r(\mathsf{X}) = 1], \tag{2}$$

where $\mathcal{R}$ denotes a hypothesis set of rejection functions. Once minimized, a combined model f which can return a rejection token ® to abstain from predictions can be defined,

$$f(x) = \begin{cases} \text{®} & \text{if } r(x) = 1 \\ h(x) & \text{if } r(x) = 0. \end{cases} \tag{3}$$

Suppose that $\mathcal{R}$ is complete (contains all possible rejectors). In such a case, one can see that setting $c = \infty$ reduces Eq. (2) to standard risk minimization setting given by Eq. (1). Furthermore, setting $c = 0$ will result in r *always* rejecting, *i.e.*, the case where there is no rejection cost. Some values of $c \in \mathbb{R}_{\geq 0}$ can be redundant. If the loss ℓ is bounded above by B, then any $c > B$ is equivalent to setting $c = B$ — the optimal $r^\star$ will be to *never* reject. In classification, where ℓ is taken to be the zero-one-loss, c is typically restricted to values in $(0, 0.5)$ as otherwise low confidence prediction can be accepted [53] — our work only considers this case. [59] explores the $c \in [0.5, 1]$ scenario.

So far, the learning task has been left general. By considering *Class Probability Estimation* (CPE) [61], we recover familiar optimal rejection and classifier pairs.

Theorem 2.1 (Optimal CPE Rejection / Chow's Rule). *Let us consider the binary CPE setting, where $\mathcal{Y} = \{0,1\}$, $\mathcal{Y}' = [0,1]$, and ℓ be any proper*[1] *loss function [60]* (e.g., *log loss). Then w.r.t. Eq.* (2)*, the optimal classifier is given by $h^\star(x) = \mathrm{P}[\mathsf{Y} = 1 \mid \mathsf{X} = x]$ and the optimal rejector is given by $r^\star(x) = \llbracket \mathbb{E}_{\mathsf{Y} \sim h^\star(\mathsf{X}=x)}[\ell(\mathsf{Y}, h^\star(x))] \geq c \rrbracket$.*

The theorem can easily be generalized to non-binary cases. We note that Theorem 2.1 is a generalization of the well known Chow's rule of classification with rejection [15, 14]. Indeed, taking ℓ to be the zero-one-loss function, we get $r^\star(x) = \llbracket |2 \cdot \mathrm{P}[\mathsf{Y} = 1 \mid \mathsf{X} = x] - 1| \geq 1 - c \rrbracket$. One can further clarify this by noticing that $\Pr[r(\mathsf{X}) = 1] = \mathbb{E}_{\mathrm{P}}[r(\mathsf{X})]$. In the general proper loss case, the optimal rejector is thresholding a generalized entropy function (known as the conditional Bayes risk [60]). This would correspond to thresholding the class probabilities $\mathrm{P}[\mathsf{Y} = y \mid \mathsf{X} = x]$ (via the *Bayes posterior* $h^\star$), but with different thresholding values per class $y \in \mathcal{Y}$ (unless ℓ is a symmetric loss function).

Although the objective of Eq. (2) follows the *cost-based model* of rejection [15], other models of rejection exist in the literature. Alternatively, the *bounded improvement model* of rejection [54, 29] maximizes coverage (non-rejection rate) whilst maintaining a constraint on the performance of the rejection measured by the selective risk (the first term of Eq. (2) inversely weighted by the coverage). Optimality conditions of the bounded improvement have been explored in [27]. In the *bounded abstention model*, the constraint and objective is switched – the selective risk is minimized with a constraint on minimum coverage [54]. Our objective function Eq. (2) (and the cost-based model) can be interpreted as a change in the type of risk considered and a change in the hard coverage constraint to a soft constraint w.r.t. bounded abstention. The optimal strategies of these approaches are explored in [28]. Alongside Section 1, further details about rejection can be found in [34].

Generalized Variational Inference Generalized Variational Inference (GVI) [42] provides a framework for a generalized set of entropy regularized risk minimization problems. In particular, GVI generalizes Bayes' rule by interpreting the Bayesian posterior as the solution to a minimization problem [73]. The Bayes' rule minimization is given by,

$$\operatorname*{argmin}_{\mathrm{Q} \in \mathcal{D}(\Theta)} \; \mathbb{E}_{\theta \sim \mathrm{Q}}\left[-\sum_{i=1}^{N} \log p(z_i \mid \theta) \right] + \mathrm{KL}(\mathrm{P} \parallel \mathrm{Q}), \tag{4}$$

where $\theta \sim \mathrm{Q}$ denotes a set of model parameters with likelihood function $p(x \mid \theta)$ and $\{z_i\}$ denotes data. Here, $\mathrm{P} \in \mathcal{D}(\Theta)$ denotes the prior and the optimal $\mathrm{Q}^\star$ denotes the posterior in Bayes' rule.

[1] Properness ensures that the true class probability is the minimizer of the loss in expectation.

For GVI, we generalize a number of quantities. For instance, the 'loss' considered can be altered from the log-likelihood $\log p(z \mid \theta)$ to an alternative loss function over samples $\{z_i\}$. The divergence function $\mathrm{KL}(\mathrm{P} \parallel \mathrm{Q})$ can also be altered to change the notion of distributional distance. Furthermore, the set of distributions being minimized $\mathcal{D}(\Theta)$ can also be altered to, *e.g.*, reduce the computational cost of the minimization problem. One can thus alter Eq. (4) to give the following:

$$\operatorname*{argmin}_{\mathrm{Q} \in \mathcal{Q}} \; L(\mathrm{Q}) + \lambda \cdot D(\mathrm{P} \parallel \mathrm{Q}), \tag{5}$$

where $\lambda > 0$ and L, D, and $\mathcal{Q}$ denote the generalized loss, divergence, and set of distribution, respectively. Here λ seeks to act as a regularization constant which can be tuned.

In our work, we will consider a GVI problem which changes the loss function and divergence to define an entropy regularized risk minimization problem. Our loss function corresponds to the learning setting. For the change in divergence, we consider φ-divergence [1, 19], which are otherwise referred to as f-divergences [62] or the Csiszár divergence [18].

Definition 2.2. Let $\varphi \colon \mathbb{R} \to (-\infty, \infty]$ be a convex lower semi-continuous function with $\varphi(1) = 0$ then the corresponding φ-divergence over non-negative measures for P, Q is defined as:

$$D_\varphi(\mathrm{P} \parallel \mathrm{Q}) \doteq \int_{\mathcal{X}} \varphi\left(\frac{\mathrm{dQ}}{\mathrm{dP}}\right) \mathrm{dP}, \quad \text{if } \mathrm{Q} \ll \mathrm{P}; \qquad \text{otherwise,} \quad D_\varphi(\mathrm{P} \parallel \mathrm{Q}) = +\infty. \tag{6}$$

We note that for φ-divergences, the regularization constant in Eq. (5) can be absorbed into the φ-divergence generator, *i.e.*, $\lambda \cdot D_\varphi(\mathrm{P} \parallel \mathrm{Q}) = D_{\lambda \cdot \varphi}(\mathrm{P} \parallel \mathrm{Q})$.

Distributionally Robust Optimization A related piece of literature is Distributionally Robust Optimization (DRO) [63], where the goal is to find a distribution that maximizes (or minimizes) the expectation of a function from a prescribed *uncertainty* set. Popular candidates for these uncertainty sets include all distributions that are a certain radius away from a fixed distribution by some divergence. φ-divergences have been used to define such uncertainty set [8, 22, 45]. Given radius $\varepsilon > 0$, define $B^\varphi_\varepsilon(\mathrm{P}) \doteq \{\mathrm{Q} \in \mathcal{D}(\mathcal{X} \times \mathcal{Y}) : D_\varphi(\mathrm{P} \parallel \mathrm{Q}) < \varepsilon\}$. Given a point-wise loss function $\ell \colon \mathcal{Y} \times \mathcal{Y}' \to \mathbb{R}$, DRO alters risk minimization, as per Eq. (1), to solve the following optimization problem:

$$\min_{h \in \mathcal{H}} \; \max_{\bar{\mathrm{Q}}_{\mathrm{x,y}} \in B_\varepsilon(\mathrm{P})} \; \mathbb{E}_{\bar{\mathrm{Q}}_{\mathrm{x,y}}} \left[\ell(\mathsf{Y}, h(\mathsf{X}))\right]. \tag{7}$$

The max over $\bar{\mathrm{Q}}$ is typically over the target space $\mathcal{Y}$. That is, $\bar{\mathrm{Q}}(x, y) = \mathrm{Q}(y) \cdot \mathrm{P}(x \mid y)$ and the max is adversarial over the marginal label distribution [74]. Note that converting the ε-ball constraint into a Lagrange multiplier, the inner optimization over Q in Eq. (7) mirrors Eq. (5). The connection between GVI and adversarial robustness has been previously noted [37].

Typically in DRO and related learning settings, the construction of the adversarial distribution defined by the inner maximization problem is implicitly solved. For example, when φ is twice differentiable, it has been shown that the inner maximization can be reduced to a variance regularization expression [22, 21]; whereas other choices of divergences such as kernel Maximum Mean Discrepancy (MMD) yields kernel regularization [66] and Integral Probability Metrics (IPM) correspond to general regularizers [36]. Another popular choice is the Wasserstein distance which has shown strong connections to point-wise adversarial robustness [10, 11, 64].

The aforementioned work, however, seek only to find the value of the inner maximization in Eq. (7) without considering the form the optimal adversarial distribution takes.

3 Rejection via Idealized Distributions

We propose learning rejection functions $r \colon \mathcal{X} \to \{0, 1\}$ by comparing data distributions P to a learned idealized distribution Q (as per Fig. 1). An idealized distribution (w.r.t. model h) is a distribution which when taken as data results in low risk (per Eq. (1)). Q are idealized rather than 'ideal' as they do not solely rely on a model's performance, but are also regularized by their distance from the data distribution P. Formally, we define our idealized distribution via a GVI minimization problem.

Definition 3.1. Given a data distribution $\mathrm{P}_{\mathrm{x,y}} \in \mathcal{D}(\mathcal{X} \times \mathcal{Y})$ and a φ-divergence, an *idealized distribution* $\mathrm{Q} \in \mathcal{D}(\mathcal{X})$ for a fixed model h and loss ℓ is a distribution given by

$$\operatorname*{arginf}_{\mathrm{Q} \in \mathcal{D}(\mathcal{X})} \; L(\mathrm{Q}) + \lambda \cdot D_\varphi(\mathrm{P} \parallel \mathrm{Q}), \tag{8}$$

where $L(\mathrm{Q}) \doteq \mathbb{E}_{\mathrm{Q}}[L'(\mathsf{X})]$ and $L'(x) \doteq \mathbb{E}_{\mathrm{P}_{\mathrm{y|x}}}\left[\ell(\mathsf{Y}, h(x))\right]$.

Given the objective of Eq. (8), an idealized distribution Q will have high mass when $L'(x)$ is small and low mass when $L'(x)$ is large. The φ-divergence regularization term prevents the idealized distributions from collapsing to a point mass. Indeed, without regularization the distance from P, idealized distributions would simply be Dirac deltas at values of $x \in \mathcal{X}$ which minimize $L'(x)$.

With an idealized distribution Q, a rejection can be made via the density ratio [67] w.r.t. P.

Definition 3.2. Given a data distribution P and an idealized distribution $\mathrm{Q} \ll \mathrm{P}$, the τ-ratio-rejector w.r.t. Q is given by $r_\tau(x) \doteq [\![\rho(x) \leq \tau]\!]$, where $\rho(x) \doteq \mathrm{dQ}/\mathrm{dP}(x)$.

Definition 3.2 aims to reject inputs where the idealized rejection distribution has lower mass than the original data distribution. Given Definition 3.1 for idealized distribution, small values of $\rho(x)$ corresponds to regions of the input space $\mathcal{X}$ where having lower data probability would decrease the expected risk of the model. Note that we do not reject on regions with high density ratio ρ as $L'(x)$ would be necessarily small or the likelihood of occurrence $\mathrm{P}(x)$ would be relatively small. We restrict the value of τ to $(0, 1]$ to ensure that rejection is focused regions where $L'(x)$ is high with high probability w.r.t. $\mathrm{P}(x)$ — further noting that $\tau = 0$ always rejects.

Although Definitions 3.1 and 3.2 suggests that we should learn distributions Q (and P) separately to make our rejection decision, in practice, we can learn the density ratio dQ/dP directly. Indeed, through the definition of Definition 2.2, we have that equivalent minimization problem:

$$\operatorname*{argmin}_{\rho\colon \mathcal{X}\to\mathbb{R}_+} \; \mathbb{E}_{\mathrm{P}_{\mathrm{x,y}}}\left[\rho(\mathsf{X}) \cdot \ell(\mathsf{Y}, h(\mathsf{X})) + \lambda \cdot \varphi(\rho(\mathsf{X}))\right]; \qquad \text{s.t.} \quad \mathbb{E}_{\mathrm{P}}\left[\rho(\mathsf{X})\right] = 1. \tag{9}$$

Such an equivalence has been utilized in DRO previously [23, Proof of Theorem 1]. Notice that the learning density ratio ρ in Eq. (9) is analogous to the acceptor $1 - r(x)$ in Eq. (2). Indeed, ignoring the normalization constraint $\mathbb{E}_{\mathrm{P}}\left[\rho(\mathsf{X})\right] = 1$, by restricting $\rho(x) \in \{0, 1\}$, letting $\lambda = c$, and letting $\varphi(z) = [\![z = 0]\!]$, the objective function of Eq. (9) can be reduced to:

$$\mathbb{E}_{\mathrm{P}_{\mathrm{x,y}}}\left[\rho(\mathsf{X}) \cdot \ell(\mathsf{Y}, h(\mathsf{X}))\right] + c \cdot \mathrm{P}\left(\rho(\mathsf{X}) = 0\right). \tag{10}$$

Given the restriction of ρ to binary outputs $\{0, 1\}$ (and Definition 3.2), we have that $r_\tau(x) = 1 - \rho(x)$. As such, Eq. (10) in this setting is equivalent to the minimization of r in Eq. (2) (with $\mathcal{R}$ as all possible functions $\mathcal{X} \to \{0, 1\}$). Through the specific selection of φ and restriction of r, we have shown that rejection via idealized distributions generalizes the typical learning with rejection objective Eq. (2).

By utilizing form of φ-divergences, we find the form of the idealized rejection distributions Q and their corresponding density ratio ρ used for rejection.

Theorem 3.3. *Given Definition 3.1, the optimal density ratio function ρ of Eq. (9) is of the form,*

$$\rho^\varphi_\lambda(x) = (\varphi')^{-1}\left(\frac{a(x) - L'(x) + b}{\lambda}\right), \tag{11}$$

where $a(x)$ are Lagrange multipliers to ensure non-negativity $\rho^\varphi_\lambda(\cdot) \geq 0$; and b is a Lagrange multiplier to ensure the constraint $\mathbb{E}_{\mathrm{P}}\left[\rho^\varphi_\lambda(\mathsf{X})\right] = 1$ is satisfied. Furthermore, the optimal idealized rejection distribution is given by: $\mathrm{Q}^\varphi(x) = \mathrm{P}(x) \cdot \rho^\varphi_\lambda(x)$.

Taking h as the Bayes posterior $h^\star$, L' becomes a function of the ground truth posterior $\Pr(\mathsf{Y} \mid \mathsf{X} = x)$. Hence taking the output h as a neural network plugin estimate of the underlying true posterior $\Pr(\mathsf{Y} \mid \mathsf{X} = x)$ (see Section 4.3) yields an approach similar to softmax response, *i.e.*, rejection based on the output of h when it outputs softmax probabilities [29]. Theorem 3.3 presents a general approach to generating rejectors from these plugin estimates, as a function of ℓ and φ.

Connections to GVI and DRO The formulation and solutions to the optimization of idealized distributions has several connections to GVI and DRO. In contrast to the setting of GVI, Eqs. (4) and (5), the support of the idealized distributions being learned in Definition 3.1 is w.r.t. inputs $\mathcal{X}$ instead of parameters. Furthermore, the inner maximization of the DRO optimization problem, Eq. (7), seeks to solve a similar form of optimization. For idealized distributions, the maximization is switched to minimization and we consider a distribution over inputs $\mathcal{X}$ instead of targets $\mathcal{Y}$. Indeed, notice that the explicit inner optimization of DRO (in Eq. (7)) over Q can be expressed as the following via the Fan's minimax Theorem [25, Theorem 2]:

$$\sup_{\lambda > 0} \inf_{\mathrm{Q}_\lambda \in \mathcal{D}(\mathcal{X})} -L(\mathrm{Q}_\lambda) + \lambda \cdot (D_\varphi(\mathrm{Q}_\lambda \,\|\, \mathrm{P}) - \varepsilon). \tag{12}$$

Notably, the loss $L(\mathrm{Q})$ in Eq. (7) can be simply negated to make the optimization over Q in Eq. (12) equivalent to DRO Eq. (8) (noting the only requirement for Eq. (7) to Eq. (12) is that $L(\mathrm{Q})$ is a linear functional of Q). This shows that switching the sign of the loss function changes idealized distributions of Definition 3.1 to DRO adversarial distributions. Indeed, through the connection between our idealized distributions and DRO adversarial distributions, the distributions $\mathrm{Q}^{\varphi}(x)$ will have the same form as the optimal rejection distributions implicitly learned in DRO.

Corollary 3.4. *Suppose $\mathrm{Q}^{\varphi}_{\lambda}$ denotes the optimal idealized distribution in Theorem 3.3 (switching $L(\mathrm{Q})$ to $-L(\mathrm{Q})$) for a fixed $\lambda > 0$. Further let $\lambda^{\star} \in \operatorname{arginf}_{\lambda>0}\{-L(\mathrm{Q}^{\varphi}_{\lambda}) + \lambda \cdot (D_{\varphi}(\mathrm{Q}^{\varphi}_{\lambda} \parallel \mathrm{P}) - \varepsilon)\}$. Then the optimal adversarial distribution in the inner minimization for DRO (Eq. (7)) is $\mathrm{Q}^{\varphi}_{\lambda^{\star}}$.*

As such, the various optimal idealized distributions (w.r.t. Definition 3.1) for rejection we will present in the sequel can be directly used to obtain the optimal adversarial distributions for DRO.

4 Learning Idealized Distributions

In the following section, we explore optimal closed-form density ratio rejectors. We first examine the easiest example — the KL-divergence — and then consider the more general α-divergences.

4.1 KL-Divergence

Let us first consider the KL-divergence [2] for constructing density ratio rejectors via Theorem 3.3.

Corollary 4.1. *Let $\varphi(z) = z \log z - z + 1$ and $\lambda > 0$. The optimal density ratio of Definition 3.1 is,*

$$\rho^{\mathrm{KL}}_{\lambda}(x) = \frac{1}{Z} \cdot \exp\left(\frac{-L'(x)}{\lambda}\right), \qquad \text{where } Z = \mathbb{E}_{\mathrm{P}}\left[\exp\left(\frac{-L'(x)}{\lambda}\right)\right]. \tag{13}$$

One will notice that $\rho^{\mathrm{KL}}_{\lambda}$ corresponds to an exponential tilt [24] of P to yield a *Gibbs distribution* Q^{KL}. The KL density ratio rejectors are significant in a few ways. First, we obtain a closed-form solution due to the properties of 'log' and complementary slackness, *i.e.*, $a(\cdot) = 0$. Secondly, due to the properties of exp, the normalization term b has a closed form solution given by the typical log-normalizer term of exponential families.

Another notable property of utilizing a KL idealized distribution is that it recovers the previously mentioned optimal rejection policies for classical modelling with rejection via cost penalty.

Theorem 4.2 (Informal)**.** *Given the CPE setting Theorem 2.1, if $h = h^{\star}$ is optimal, then there exists a r^{KL}_{τ} which is equivalent to the optimal rejectors in Theorem 2.1.*

Theorem 4.2 states that the KL density rejectors (with correctly specified λ and τ) with optimal predictors $h^{\star}$ recovers the optimal rejectors of the typical rejection setting, *i.e.*, Chow's rule.

Until now, we have implicitly assumed that the true data distribution P is accessible. In practice, we only have access empirical $\hat{\mathrm{P}}_N$, defining subsequent rejectors $\hat{\rho}_{\lambda,N}$. We show that for the KL rejector, $\hat{\mathrm{P}}_N$ is enough given the following generalization bound.

Theorem 4.3. *Assume we have bounded loss $|\ell(\cdot,\cdot)| \leq B$ for $B > 0$, $\hat{\mathrm{P}}_N \subset \mathrm{P}$ with h.p., and $\mathcal{T} \subset \mathrm{Supp}(\mathrm{P})$. Suppose $M = |\mathcal{T}| < +\infty$, then with probability $1 - \delta$, we have that*

$$\sup_{x \in \mathcal{T}} \left|\rho^{\mathrm{KL}}_{\lambda}(x) - \hat{\rho}^{\mathrm{KL}}_{\lambda,N}(x)\right| \leq C \cdot \sqrt{\frac{2}{N} \log\left(\frac{2M}{\delta}\right)},$$

where $C = \exp\left(B/\lambda\right)^3 \cdot \sinh\left(B/\lambda\right)$.

Looking at Theorem 4.3, essentially we pay a price in generalization M for each element $x \in \mathcal{T}$ we are testing for rejection. For generalization, it is useful to consider how N, M changes our rate in Theorem 4.3. If we assume that the test set $\mathcal{T}$ is small in comparison to the N samples used to generate empirical distribution $\hat{\mathrm{P}}$, then the $\mathcal{O}(1/\sqrt{N})$ rate will dominate. A concrete example of this case is when $|\mathcal{X}|$ is finite. A less advantaged scenario is when $M \approx N$, yielding $\mathcal{O}(\log(N)/\sqrt{N})$ — the scenario where we test approximately the same number of data points as that used to learn the rejector. This rate will still decrease with $N \to \infty$, although with a $\log N$ price.

4.2 Alpha-Divergences

Although the general case of finding idealized rejection distributions for φ-divergences is difficult, we examine a specific generalization of the KL case, the α-divergences [3].

Definition 4.4. For $\alpha \in \mathbb{R}$, the α-divergence D_α is defined as the φ_α-divergence, where

$$\varphi_\alpha(z) \doteq \begin{cases} \frac{4}{1-\alpha^2}\left(1 - z^{\frac{1+\alpha}{2}}\right) - \frac{2}{1-\alpha}(z-1) & \text{if } \alpha \neq \pm 1 \\ -\log z + (z-1) & \text{if } \alpha = -1 \\ z \log z - (z-1) & \text{if } \alpha = 1 \end{cases} .$$

We further define ψ_α, where $\psi_\alpha(z) = z^{\frac{1-\alpha}{2}}$ when $\alpha \neq 1$ and $\psi_\alpha(z) = \log z$ when $\alpha = 1$.

Note that taking $\alpha = 1$ recover the KL-divergence. The α-divergence covers a wide range of divergences including the Pearson χ^2 divergence ($\alpha = 3$). For the density ratio, α-divergences with $\alpha \neq 1$ (*i.e.* not KL) can be characterized as the following.

Theorem 4.5. *Let $\alpha \neq 1$ and $\lambda > 0$. For φ_α, the optimal density ratio rejector $\rho^\alpha_\lambda(x)$ is,*

$$\rho^\alpha_\lambda(x) = \psi_\alpha^{-1}\left(\frac{2}{\alpha - 1} \cdot \left(a(x) - \frac{L'(x)}{\lambda} + b\right)^{-1}\right) \tag{14}$$

$a(x)$ are Lagrange multipliers for positivity; and b is a Lagrange multiplier for normalization.

One major downside of using general φ-divergences is that solving the Lagrange multipliers for the idealized rejection distribution is often difficult. Indeed, the "log" and "exp" ensures non-negativity of the idealized distribution when the input data P is in the interior of the simplex; and also provides a convenient normalization calculation. For α-divergences, the non-negative Lagrange multipliers $a(\cdot)$ can be directly solved given certain conditions.

Corollary 4.6. *Suppose $\alpha > 1$ and $\lambda > 0$, then Eq. (14) simplifies to,*

$$\rho^\alpha_\lambda(x) = \left[\left(\frac{\alpha - 1}{2} \cdot \left(b - \frac{L'(x)}{\lambda}\right)\right)^{\frac{2}{\alpha-1}}\right]_+ , \tag{15}$$

where we take non-integer powers of negative values as 0.

On the other hand, for $\alpha \leq -1$, $D_\alpha(\mathrm{P} \parallel \mathrm{Q}) = \infty$ whenever Q is on the boundary whenever P is not on the boundary [3, Section 3.4.1]. As such, we can partially simplify Eq. (14) for $\alpha \leq -1$.

Corollary 4.7. *Suppose $\alpha \leq -1$, $\lambda > 0$, and P lies in the simplex interior, then $a(\cdot) = 0$ in Eq. (14).*

Both Corollaries 4.6 and 4.7 can provide a unique rejector policy than the KL-divergence variant. Corollary 4.7 can provide a similar effect when $a(x) \neq 0$ for all x. Nevertheless, having to determine which inputs $a(x) \neq 0$ and solving these values are difficult in practice. As such, we focus on $\alpha > 0$. If there are values of $L'(x)$ with high risk, the $\max$ will flatten these inputs to 0 in Corollary 4.6. However, if the original model h performs well and $L'(x)$ is relatively small for all x, then it is possible that the $\max$ is not utilized. In such a case, the α-divergence rejectors can end up being similar — this follows the fact that (as $\varphi''_\alpha(1) > 0$) locally all α-divergences will be similar to the χ^2 / ($\alpha = 3$)-divergence [56, Theorem 7.20]. This ultimately results in α-divergences being similar to, *e.g.*, Chow's rule when $h_y(x) \approx \Pr(\mathsf{Y} = y \mid \mathsf{X} = x)$ in classification via Theorem 2.1.

Among the $\alpha > 0$ cases, we examine the χ^2-divergence ($\alpha = 3$) which results in closed form for bounded loss functions and sufficient large regularizing parameter λ.

Corollary 4.8. *Suppose that $\ell(\cdot, \cdot) \leq B$ and $\lambda > \max_x L'(x) - \mathbb{E}_{\mathrm{P}}[L'(x)]$. Then,*

$$\rho^{\alpha=3}_\lambda(x) = 1 + \frac{\mathbb{E}_{\mathrm{P}}[L'(x)] - L'(x)}{\lambda}. \tag{16}$$

The condition on λ for Corollary 4.8 is equivalent to rescaling bounded loss functions ℓ. Indeed, by fixing $\lambda = 1$, we can achieve a similar Theorem with suitable rescaling of ℓ. Nevertheless, Eq. (16) provides a convenient form to allow for generalization bounds to be established.

Theorem 4.9. *Assume we have bounded loss $|\ell(\cdot, \cdot)| \leq B$ for $B > 0$, $\lambda > 2B$, $\hat{\mathrm{P}}_N \subset \mathrm{P}$ with h.p., and $\mathcal{T} \subset \mathrm{Supp}(\mathrm{P})$. Suppose $M = |\mathcal{T}| < +\infty$, then with probability $1 - \delta$, we have that*

$$\sup_{x \in \mathcal{T}} \left| \rho_{\lambda}^{\alpha=3}(x) - \hat{\rho}_{\lambda}^{\alpha=3}(x) \right| \leq \frac{B}{\lambda} \cdot \sqrt{\frac{2}{N} \log\left(\frac{2M}{\delta}\right)}.$$

Notice that Theorem 4.9's sample complexity is equivalent to Theorem 4.3 up to constant multiplication. Hence, the analysis of Theorem 4.3 regarding the scales of N, M hold for Theorem 4.9.

A question pertaining to DRO is what would be the generalization capabilities of the corresponding adversarial distributions $\hat{\mathrm{Q}}_{\lambda,N} \doteq \hat{\mathrm{P}}_N \cdot \hat{\rho}_{\lambda,N}$ (through Corollary 3.4). On a finite domain, via Theorems 4.3 and 4.9 and a simple triangle inequality, one can immediately bound the total variation $\mathrm{TV}(\hat{\mathrm{Q}}_{\lambda,N}, \mathrm{Q}_{\lambda}) \leq \mathcal{O}(1/\sqrt{N})$, see Appendix M for further details.

4.3 Practical Rejection

We consider practical concerns for utilizing the KL-or (α > 1)-divergence case, Eqs. (13) and (15), for post-hoc rejection. To do so, we need to estimate the loss $L'(x)$ and rejector normalizer Z or b.

Loss: The former is tricky, we require an estimate to evaluate $L'(x) = \mathbb{E}_{\mathrm{P}_{\mathrm{y}|\mathrm{x}}}[\ell(\mathsf{Y}, h(x))]$ over any possible $x \in \mathcal{X}$ to allow us to make a rejection decision. Implicitly, this requires us to have a high quality estimate of $\mathrm{P}_{\mathrm{y}|\mathrm{x}}$. In a general learning setting, this can be difficult to obtain — in fact it is just the overall objective that we are trying to learning, *i.e.*, predicting a target y given an input x. However, in the case of CPE (classification), it is not unreasonable to obtain an *calibrated estimate* of $\mathrm{P}_{\mathrm{y}|\mathrm{x}}$ via the classifier $h \colon \mathcal{X} \to \mathcal{D}(\mathcal{Y})$ [55]. In Section 5, we utilize temperature scaling to calibrate the neural networks we learn to provide such an estimate [32]. Hence, we set $L'(x) = \mathbb{E}_{\mathsf{Y} \sim h(x)}[\ell(\mathsf{Y}, h(x))]$. For proper CPE loss functions, $\mathbb{E}_{\mathsf{Y} \sim h(x)}[\ell(\mathsf{Y}, h(x))]$ acts as a generalized entropy function. As such, the rejectors Eqs. (13) and (15) act as functions over said generalized entropy functions. It should be noted that simply considering the softmax outputs of neural networks for probability estimation have seen prior success [29, 39]. This is equivalent to taking the 0-1-loss function [15, 35, 28] and taking a plugin estimate of probabilities via the neural network's output. The study of using plugin estimates have also been explored in the model cascade literature [40].

Normalization: For the latter, we utilize a sample based estimate (over the dataset used to train the rejector) of $\mathbb{E}_{\mathrm{P}}[\rho_{\lambda}^{\varphi}(\mathsf{X})]$ is utilized to solve the normalizers Z and b. In the case of the KL-divergence rejector, this is all that is required due to the convenient function form of the Gibbs distribution, *i.e.*, the normalizer Z can be simply estimated by a sample mean. However, for $\alpha > 1$-divergences b needs to be found to determine the normalization. Practically, we find b through bisection search of

$$\mathbb{E}_{\hat{\mathrm{P}}}\left[\left(\frac{\alpha - 1}{2} \cdot \left(b - \frac{L'(x)}{\lambda}\right)\right)^{\frac{2}{\alpha - 1}}\right]_{+} - 1 = 0. \tag{17}$$

This practically works as the optimization b is over a single dimension. Furthermore, we can have that $b > \min_x \{L'(x)/\lambda\}$. As an upper bound over the possible values of b, we utilize a heuristic where we multiple the corresponding maximum of $L'(x)/\lambda$ with a constant.

Threshold τ: In addition to learning the density ratio, an additional consideration is how to tune τ in the rejection decision, Definition 3.2. Given a fixed density estimator ρ, change τ amounts to changing the rate of rejection. We note that this problem is not limited to our density ratio rejectors, where approaches with rejectors $r \in \mathcal{R}$ via a (surrogate) minimization of Eq. (2) may be required multiple rounds of training with different rejection costs c to find an accept rate of rejection. In our case, we have a fixed ρ which allows easy tuning of τ given a validation dataset, similar to other confidence based rejection approaches, *e.g.*, tuning a threshold for the margin of a classifier [7].

5 Experiments

In this section, we evaluate our distributional approach to rejection across a number of datasets. In particular, we consider the standard classification setting with an addition setting where uniform label noise is introduced [4, 30]. For our density ratio rejectors, we evaluate the KL-divergence based

Table 1: Summary of rejection methods over all baselines and datasets targeting 80% coverage. Each cell reports the "accuracy [coverage]" values, **bold** for most accurate, and s.t.d. reported in Appendix.

		Base	KL-Rej	(α=3)-Rej	PredRej	CSS	DEFER	GCE
Clean	HAR	97.38 [100]	99.93 [81]	**99.93** [82]	98.86 [85]	99.58 [81]	99.44 [80]	99.31 [82]
	Gas Drift	94.10 [100]	**99.16** [80]	**99.16** [80]	98.12 [80]	98.68 [80]	98.06 [80]	97.62 [80]
	MNIST	98.55 [100]	99.93 [87]	99.93 [88]	99.18 [74]	**99.95** [83]	99.93 [80]	99.85 [80]
	CIFAR-10	90.20 [100]	**97.22** [80]	97.17 [81]	91.40 [74]	95.45 [81]	93.72 [81]	94.25 [81]
	OrganMNIST	89.10 [100]	**96.55** [80]	96.52 [80]	93.79 [82]	94.49 [80]	93.47 [80]	93.68 [80]
	OctMNIST	91.93 [100]	97.08 [81]	**97.18** [80]	93.43 [86]	95.40 [81]	94.66 [85]	94.91 [87]
Noisy (25%)	HAR	96.51 [100]	98.56 [81]	98.56 [81]	97.22 [80]	97.82 [81]	97.78 [69]	**98.85** [80]
	Gas Drift	93.84 [100]	97.30 [80]	97.28 [80]	95.87 [82]	98.71 [80]	**99.02** [77]	97.52 [75]
	MNIST	97.88 [100]	99.89 [80]	99.89 [81]	98.00 [93]	99.94 [80]	99.93 [81]	**99.95** [81]
	CIFAR-10	85.31 [100]	92.25 [82]	**92.50** [81]	85.84 [88]	89.58 [82]	90.93 [80]	92.22 [81]
	OrganMNIST	89.10 [100]	95.86 [82]	**96.29** [81]	93.40 [84]	96.74 [81]	94.67 [80]	94.48 [80]
	OctMNIST	91.89 [100]	**97.17** [80]	97.10 [81]	93.42 [83]	95.49 [81]	94.08 [89]	94.63 [78]

rejector (Corollary 4.1) and (α=3)-based rejectors (Corollary 4.6) with 50 equidistant $\tau \in (0, 1]$ values. For our tests, we fix $\lambda = 1$. Throughout our evaluation, we assume that a neural network (NN) model without rejection is accessible for all (applicable) approaches. For our density ratio rejectors, we utilize the log-loss, practical considerations in Section 4.3, and Algorithm 1.[2]

To evaluate our density ratio rejectors and baselines, we compare accuracy and acceptance coverage. Accuracy corresponds to the 0-1 loss in Eq. (1) and the acceptance coverage is the percentage of non-rejections in the test set. Ideally, a rejector smoothly trade-offs between accuracy and acceptance, *i.e.*, a higher accuracy can be achieved by decreasing the acceptance coverage by rejecting more data. We study this trade-off by examining multiple cut-off values τ and rejection costs c.

Dataset and Baselines We consider 6 multiclass classification datasets. For tabular datasets, we consider the gas drift dataset [68] and the human activity recognition (HAR) dataset [5]. Each of these datasets consists of 6 classes to predict. Furthermore, we utilize a two hidden layer MLP NN model for these datasets. We consider the MNIST image dataset [46] (10 classes), where we utilize a convolutional NN. Additionally, we consider 3 larger image datasets with ResNet-18 architecture [33]: CIFAR-10 [44] (10 classes); and OrgMNIST / OrganSMNIST (11 classes) and OctMNIST (4 classes) from the MedMNIST collection [69, 70]. These prediction models are trained utilizing the standard logistic / log-loss without rejection and then are calibrated via temperature scaling [32]. For each of these datasets, we utilize both clean and noisy variants. For the noisy variant, we flip the class labels of the train set with a rate of 25%. We note that the test set is clean in both cases. All evaluation uses 5-fold cross validation. All implementation use PyTorch and training was done on a `p3.2xlarge` AWS instance.

We consider 4 different baseline for comparison. Each is trained with 50 equidistant costs $c, \tau \in [0, 0.5)$, except on OctMNIST which uses 10 equidistant costs (selecting c, τ discussed in Appendix P). One baseline used corresponds to a modification of [49]'s cross-entropy surrogate approach (DEFER) originally used for the learning to defer literature (see [14, Appendix A.2]). This approach treats the rejection option as a separate class and solves a $|\mathcal{Y}| + 1$ classification problem. A generalization of DEFER is considered which utilizes generalized cross-entropy [75] as its surrogate loss (GCE) [13]. We also consider a cost-sensitive classification reduction (CSS) of the classification with rejection problem [14] utilizing the sigmoid loss function. The aforementioned 3 baselines all learn a model with rejection simultaneously, *i.e.*, a pretrained model cannot be utilized. We also consider a two stage predictor-rejector approach (PredRej) which learns a rejector from a pretrained classifier [48].

Results Table 1 presents a tabular summary of accuracy and coverage values when targeting 80% coverage values; and Fig. 2 presents a summary plot of the acceptance coverage versus model accuracy after rejection, focused around the 60% to 100% coverage region. This plot is limited to HAR, Gas Drift, and MNIST due to space limitations however the deferred datasets show curves where the density ratio rejector dominate with better accuracy and coverage in the plotted region, with corresponding extended plots for all datasets in Appendix Q . Over all folds for MNIST our density ratio rejectors take approximately $\approx 1/2$ hour to fit. A single baseline (fixed c) takes upwards of 2 hour for a single fold. Overall, given that the underlying model is calibrated, we find that our

[2] Our rejector's code public at: `https://github.com/alexandersoen/density-ratio-rejection`.

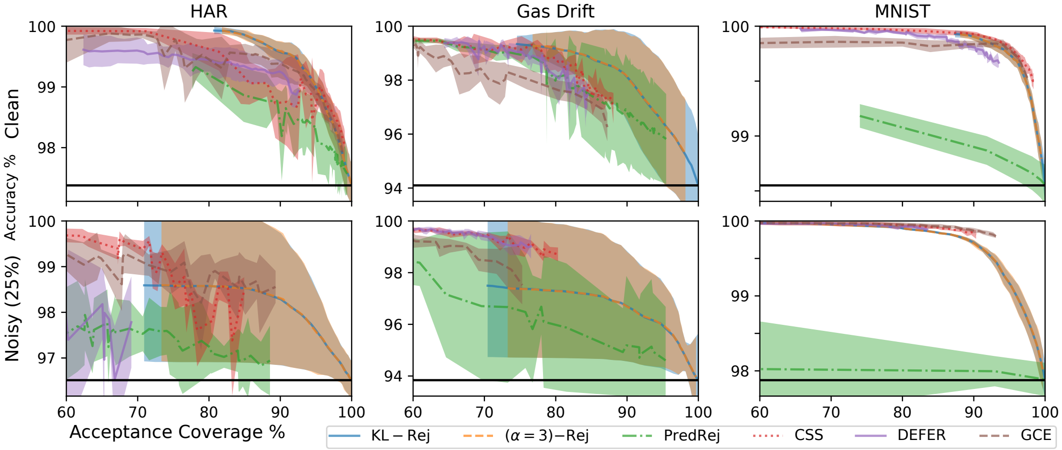

Figure 2: Accuracy vs coverage plots across select datasets and all approaches, with 50 equidistant $\tau \in (0, 1]$ and $c \in [0, 0.5)$ values (sorted by coverage). The black horizontal line depicts base models trained without rejection. Missing approaches in the plots indicates that the model rejects more than 60% of test points or has accuracy below the base model. Shaded region indicates ±1 s.t.d. region.

density ratio rejector are either competitive or superior to the baselines. One might notice that the aforementioned baselines do not or provide poor trade-offs for coverage values $> 95\%$ (as per Fig. 2). Indeed, to achieve rejection with high coverage (without architecture tuning), approaches which 'wrap' a base classifier seem preferable, *i.e.*, PredRej and our density ratios rejectors. Even at lower coverage targets (80%), Table 1 shows that density-ratio methods are comparable or superior in the more complex datasets of CIFAR-10 and the MedMNIST collection. If large models are allowed to be used for the rejector — as per the MNIST case — CSS, DEFER, and GCE can provide superior accuracy vs acceptance coverage trade-offs (noisy MNIST). However, this is not always true as per CIFAR-10 where all approaches are similarly effected by noise; or OctMNIST and OrgMNIST where approaches only slightly change with noise. The latter appears to be a consequence of label noise not effecting the Base classifier's accuracy (using the larger ResNet-18 architecture), as per Table 1.

Among the approaches which 'wrap' the base classifier h, we find that these approaches have higher variance ranges than the other approaches. In particular, the randomness of the base model potentially magnifies the randomness after rejection. The variance range of the base model tends to increase as the noise increases (additional ranges of noise for HAR and Gas Drift are presented in the Appendix). The influence on rejection is unsurprising as these 'wrapping' approaches predict via a composition of the original model (and hence inherits its randomness across folds). In general, our density ratio rejector outperforms PredRej. However, it should be noted that PredRej does not require a calibrated classifier. Among the density ratio rejectors, between KL and $\alpha = 3$, the only variation is in coverage region that the $\tau \in (0, 1]$ threshold covers. This follows from the fact that for similar distributions, φ-divergences act similarly [56, Theorem 7.20]. We find this pattern holds for other values of α.

6 Limitations and Conclusions

We propose a new framework for rejection by learning idealized density ratios. Our proposed rejection framework links typically explored classification with rejection to generalized variational inference and distributionally robust optimization. It should be noted that although we have focused on classification, $L'(\mathrm{Q})$ could in theory be replaced by any other loss functions. In this sense, one could adapt this approach to other learning tasks such as regression, discussed in Appendix N. Furthermore, although we have focused on φ-divergences, there are many alternative ways idealized distribution can be constructed, *e.g.*, integral probability metrics [51, 9]. One limitation of our distributional way of rejecting is the reliance on approximating $\mathrm{P}[\mathsf{Y} \mid \mathsf{X}]$ with model h. In future work, one may seek to approximate the density ratio ρ by explicitly learning densities Q and P or via gradient based methods (for the latter, see Appendix O).

Acknowledgments and Disclosure of Funding

We thank the reviewers and the area chair for the various suggestions which improved the paper. A majority of the work was done whilst AS was interning at Amazon.

References

[1] Syed Mumtaz Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1):131–142, 1966.

[2] S.-I. Amari and H. Nagaoka. *Methods of Information Geometry*. Oxford University Press, 2000.

[3] Shun-ichi Amari. *Information geometry and its applications*, volume 194. Springer, 2016.

[4] Dana Angluin and Philip Laird. Learning from noisy examples. *Machine learning*, 2:343–370, 1988.

[5] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, Jorge Luis Reyes-Ortiz, et al. A public domain dataset for human activity recognition using smartphones. In *Esann*, volume 3, page 3, 2013.

[6] Arthur Asuncion and David Newman. Uci machine learning repository, 2007.

[7] Peter L Bartlett and Marten H Wegkamp. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(8), 2008.

[8] Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

[9] Jeremiah Birrell, Paul Dupuis, Markos A Katsoulakis, Yannis Pantazis, and Luc Rey-Bellet. (f, Γ)-Divergences: interpolating between f-divergences and integral probability metrics. *The Journal of Machine Learning Research*, 23(1):1816–1885, 2022.

[10] Jose Blanchet and Karthyek Murthy. Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600, 2019.

[11] Jose Blanchet, Yang Kang, and Karthyek Murthy. Robust wasserstein profile inference and applications to machine learning. *Journal of Applied Probability*, 56(3):830–857, 2019.

[12] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.

[13] Yuzhou Cao, Tianchi Cai, Lei Feng, Lihong Gu, Jinjie Gu, Bo An, Gang Niu, and Masashi Sugiyama. Generalizing consistent multi-class classification with rejection to be compatible with arbitrary losses. *Advances in Neural Information Processing Systems*, 35:521–534, 2022.

[14] Nontawat Charoenphakdee, Zhenghang Cui, Yivan Zhang, and Masashi Sugiyama. Classification with rejection based on cost-sensitive classification. In *International Conference on Machine Learning*, pages 1507–1517, 2021.

[15] C Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, 16(1):41–46, 1970.

[16] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Boosting with abstention. *Advances in Neural Information Processing Systems*, 29, 2016.

[17] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *Algorithmic Learning Theory: 27th International Conference, ALT 2016, Bari, Italy, October 19-21, 2016, Proceedings 27*, pages 67–82. Springer, 2016.

[18] Imre Csiszár. Eine informationstheoretische ungleichung und ihre anwendung auf beweis der ergodizitaet von markoffschen ketten. *Magyer Tud. Akad. Mat. Kutato Int. Koezl.*, 8:85–108, 1964.

[19] Imre Csiszár. Information-type measures of difference of probability distributions and indirect observation. *studia scientiarum Mathematicarum Hungarica*, 2:229–318, 1967.

[20] Imre Csiszár, Paul C Shields, et al. Information theory and statistics: A tutorial. *Foundations and Trends® in Communications and Information Theory*, 1(4):417–528, 2004.

[21] John Duchi and Hongseok Namkoong. Variance-based regularization with convex objectives. *The Journal of Machine Learning Research*, 20(1):2450–2504, 2019.

[22] John C Duchi, Peter W Glynn, and Hongseok Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 46(3): 946–969, 2021.

[23] Krishnamurthy Dj Dvijotham, Jamie Hayes, Borja Balle, Zico Kolter, Chongli Qin, Andras Gyorgy, Kai Xiao, Sven Gowal, and Pushmeet Kohli. A framework for robustness certification of smoothed classifiers using f-divergences. In *International Conference on Learning Representations*, 2019.

[24] Bradley Efron. *Exponential families in theory and practice*. Cambridge University Press, 2022.

[25] Ky Fan. Minimax theorems. *Proceedings of the National Academy of Sciences of the United States of America*, 39(1):42, 1953.

[26] Leo Feng, Mohamed Osama Ahmed, Hossein Hajimirsadeghi, and Amir H. Abdi. Towards better selective classification. In *The Eleventh International Conference on Learning Representations*, 2023.

[27] Vojtech Franc and Daniel Prusa. On discriminative learning of prediction uncertainty. In *International Conference on Machine Learning*, pages 1963–1971, 2019.

[28] Vojtech Franc, Daniel Prusa, and Vaclav Voracek. Optimal strategies for reject option classifiers. *Journal of Machine Learning Research*, 24(11):1–49, 2023.

[29] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. *Advances in neural information processing systems*, 30, 2017.

[30] Aritra Ghosh, Naresh Manwani, and PS Sastry. Making risk minimization tolerant to label noise. *Neurocomputing*, 160:93–107, 2015.

[31] Yves Grandvalet, Alain Rakotomamonjy, Joseph Keshet, and Stéphane Canu. Support vector machines with a reject option. *Advances in neural information processing systems*, 21, 2008.

[32] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330, 2017.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[34] Kilian Hendrickx, Lorenzo Perini, Dries Van der Plas, Wannes Meert, and Jesse Davis. Machine learning with a reject option: A survey. *Machine Learning*, 113(5):3073–3110, 2024.

[35] Radu Herbei and Marten H Wegkamp. Classification with reject option. *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, pages 709–721, 2006.

[36] Hisham Husain. Distributional robustness with ipms and links to regularization and gans. *Advances in Neural Information Processing Systems*, 33:11816–11827, 2020.

[37] Hisham Husain and Jeremias Knoblauch. Adversarial interpretation of bayesian inference. In *International Conference on Algorithmic Learning Theory*, pages 553–572, 2022.

[38] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456, 2015.

[39] Paul F Jaeger, Carsten Tim Lüth, Lukas Klein, and Till J. Bungert. A call to reflect on evaluation practices for failure detection in image classification. In *The Eleventh International Conference on Learning Representations*, 2023.

[40] Wittawat Jitkrittum, Neha Gupta, Aditya K Menon, Harikrishna Narasimhan, Ankit Rawat, and Sanjiv Kumar. When does confidence-based cascade deferral suffice? *Advances in Neural Information Processing Systems*, 36, 2024.

[41] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*, 2014.

[42] Jeremias Knoblauch, Jack Jewson, and Theodoros Damoulas. An optimization-centric view on bayes' rule: Reviewing and generalizing variational inference. *The Journal of Machine Learning Research*, 23(1):5789–5897, 2022.

[43] Donald E Knuth. Two notes on notation. *The American Mathematical Monthly*, 99(5):403–422, 1992.

[44] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Toronto, ON, Canada, 2009.

[45] Henry Lam. Robust sensitivity analysis for stochastic systems. *Mathematics of Operations Research*, 41(4):1248–1275, 2016.

[46] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]*, 2, 2010.

[47] Naresh Manwani, Kalpit Desai, Sanand Sasidharan, and Ramasubramanian Sundararajan. Double ramp loss based reject option classifier. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 151–163. Springer, 2015.

[48] Anqi Mao, Mehryar Mohri, and Yutao Zhong. Predictor-rejector multi-class abstention: Theoretical analysis and algorithms. In *International Conference on Algorithmic Learning Theory*, pages 822–867, 2024.

[49] Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning*, pages 7076–7087, 2020.

[50] Hussein Mozannar, Hunter Lang, Dennis Wei, Prasanna Sattigeri, Subhro Das, and David Sontag. Who should predict? exact algorithms for learning to defer to humans. In *International Conference on Artificial Intelligence and Statistics*, pages 10520–10545, 2023.

[51] Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997.

[52] Harikrishna Narasimhan, Wittawat Jitkrittum, Aditya K Menon, Ankit Rawat, and Sanjiv Kumar. Post-hoc estimators for learning to defer to an expert. *Advances in Neural Information Processing Systems*, 35:29292–29304, 2022.

[53] Chenri Ni, Nontawat Charoenphakdee, Junya Honda, and Masashi Sugiyama. On the calibration of multiclass classification with rejection. *Advances in Neural Information Processing Systems*, 32, 2019.

[54] Tadeusz Pietraszek. Optimizing abstaining classifiers using roc analysis. In *Proceedings of the 22nd international conference on Machine learning*, pages 665–672, 2005.

[55] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

[56] Yury Polyanskiy and Yihong Wu. *Information theory: From coding to learning*. Cambridge university press, 2024.

[57] Andrea Pugnana and Salvatore Ruggieri. A model-agnostic heuristics for selective classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 9461–9469, 2023.

[58] Andrea Pugnana, Lorenzo Perini, Jesse Davis, and Salvatore Ruggieri. Deep neural network benchmarks for selective classification. *Journal of Data-centric Machine Learning Research*, 2024.

[59] H. G. Ramaswamy, Ambuj Tewari, and Shivani Agarwal. Consistent algorithms for multiclass classification with an abstain option. *Electronic Journal of Statistics*, 12:530–554, 2018.

[60] Mark D Reid and Robert C Williamson. Composite binary losses. *The Journal of Machine Learning Research*, 11:2387–2422, 2010.

[61] Mark D Reid and Robert C Williamson. Information, divergence and risk for binary experiments. *Journal of Machine Learning Research*, 12:731–817, 2011.

[62] Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016.

[63] Herbert E Scarf. A min-max solution of an inventory problem. Technical report, RAND CORP SANTA MONICA CALIF, 1957.

[64] Aman Sinha, Hongseok Namkoong, and John Duchi. Certifiable distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018.

[65] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[66] Matthew Staib and Stefanie Jegelka. Distributionally robust optimization and generalization in kernel methods. In *Advances in Neural Information Processing Systems*, pages 9131–9141, 2019.

[67] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

[68] Alexander Vergara, Shankar Vembu, Tuba Ayhan, Margaret A Ryan, Margie L Homer, and Ramón Huerta. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166:320–329, 2012.

[69] Jiancheng Yang, Rui Shi, and Bingbing Ni. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 191–195. IEEE, 2021.

[70] Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

[71] Ming Yuan and Marten Wegkamp. Classification methods with reject option based on convex risk minimization. *Journal of Machine Learning Research*, 11(1), 2010.

[72] Ahmed Zaoui, Christophe Denis, and Mohamed Hebiri. Regression with reject option and application to knn. *Advances in Neural Information Processing Systems*, 33:20073–20082, 2020.

[73] Arnold Zellner. Optimal information processing and bayes's theorem. *American Statistician*, pages 278–280, 1988.

[74] Jingzhao Zhang, Aditya Krishna Menon, Andreas Veit, Srinadh Bhojanapalli, Sanjiv Kumar, and Suvrit Sra. Coping with label shift via distributionally robust optimisation. In *International Conference on Learning Representations*, 2021.

[75] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31, 2018.

Supplementary Material

This is the Supplementary Material to Paper "Rejection via Learning Density Ratios". To differentiate with the numberings in the main file, the numbering of Theorems is letter-based (A, B, ...).

Table of contents

Proof

Deferred Content

A Proof of Theorem 2.1

Proof. We first rewrite the coverage probability as an expectation:

$$\mathbb{E}_{\mathrm{P}}[(1 - r(\mathsf{X})) \cdot \ell(\mathsf{Y}, h(\mathsf{X}))] + c \cdot \Pr[r(\mathsf{X}) = 1] = \mathbb{E}_{\mathrm{P}}[(1 - r(\mathsf{X})) \cdot \ell(\mathsf{Y}, h(\mathsf{X})) + c \cdot r(\mathsf{X})].$$

We note that point-wise, the *proper* loss ℓ is minimized by taking the Bayes optimal classifier $\eta^\star(x) = \Pr[\mathsf{Y} = +1 \mid \mathsf{X} = x]$. Thus taking the argmin over all possible CPE classifiers, $h^\star = \eta^\star$. We note that the point-wise risk taken by the Bayes optimal classifier is typical denoted as the Bayes point-wise risk $\underline{L}(x) = \mathbb{E}_{\mathrm{P}_{\mathrm{y}|\mathrm{x}=x}}[\ell(\mathsf{Y}, \eta^\star(x))]$ [60, 61].

As such, we are left to minimize r over,

$$\begin{aligned}
&\mathbb{E}_{\mathrm{P}}[(1 - r(\mathsf{X})) \cdot \ell(\mathsf{Y}, \eta^\star(\mathsf{X})) + c \cdot r(\mathsf{X})] \\
&= \mathbb{E}_{\mathrm{P}_{\mathrm{x}}} \mathbb{E}_{\mathrm{P}_{\mathrm{y}|\mathrm{x}=\mathsf{X}}}[(1 - r(\mathsf{X})) \cdot \ell(\mathsf{Y}, \eta^\star(\mathsf{X})) + c \cdot r(\mathsf{X})] \\
&= \mathbb{E}_{\mathrm{P}_{\mathrm{x}}}[(1 - r(\mathsf{X})) \cdot \underline{L}(\mathsf{X}) + c \cdot r(\mathsf{X})] \\
&= \mathbb{E}_{\mathrm{P}_{\mathrm{x}}}[\underline{L}(\mathsf{X})] + \mathbb{E}_{\mathrm{P}_{\mathrm{x}}}\left[r(\mathsf{X}) \cdot (c - \underline{L}(\mathsf{X}))\right].
\end{aligned}$$

Thus, we immediately get the optimal $r^\star(x) = [\![\underline{L}(x) \geq c]\!]$. □

B Proof of Theorem 3.3

Proof. We use the theory of Lagrange multipliers to make different constraints explicit optimization problems.

Let us first consider the reduction from learing explicit distributions Eq. (8) to density ratios Eq. (9). First note that the objective Eq. (8) can be written as follows:

$$\begin{aligned}
&L(\mathrm{Q}) + \lambda \cdot D_\varphi(\mathrm{P} \parallel \mathrm{Q}) \\
&= \int L'(x) \mathrm{d}\mathrm{Q}(x) + \lambda \cdot \int \varphi\left(\frac{\mathrm{dQ}}{\mathrm{dP}}\right) \mathrm{dP}(x) \\
&= \int L'(x) \frac{\mathrm{dQ}}{\mathrm{dP}}(x) \mathrm{dP}(x) + \lambda \cdot \int \varphi\left(\frac{\mathrm{dQ}}{\mathrm{dP}}(x)\right) \mathrm{dP}(x) \\
&= \mathbb{E}_{\mathrm{P}}\left[L'(x) \cdot \frac{\mathrm{dQ}}{\mathrm{dP}}(x) + \lambda \cdot \varphi\left(\frac{\mathrm{dQ}}{\mathrm{dP}}(x)\right)\right]
\end{aligned}$$

The reduction to Eq. (9) now follows from a reduction from minimizing over Q to dQ/dP, noting that Q is restricted to be on the simplex. As such, the simplex constraints are transfered to:

$$\frac{\mathrm{dQ}}{\mathrm{dP}} \geq 0 \quad \text{and} \quad \int \mathrm{dP}(x) \cdot \frac{\mathrm{dQ}}{\mathrm{dP}}(x) = \mathbb{E}_{\mathrm{P}}\left[\frac{\mathrm{dQ}}{\mathrm{dP}}(\mathsf{X})\right] = 1,$$

where the former is the non-negativity of simplex elements and the latter is the normalization requirement. Hence, taking $\rho = \rho_\lambda \doteq \frac{\mathrm{dQ}}{\mathrm{dP}}$ completes the reduction. (we remove the subscript "λ" for the rest of the proof)

As such we have the optimization problem in Eq. (9), where we will convert the constraints into Lagrange multipliers, defining,

$$\begin{aligned}
\mathcal{L}(\rho; a, b) &= \mathbb{E}_{\mathrm{P}}\left[\rho(\mathsf{X}) \cdot L'(\mathsf{X}) + \lambda \cdot \varphi(\rho(\mathsf{X}))\right] - \int a'(x)\rho(x)\mathrm{d}x + b \cdot (1 - \mathbb{E}_{\mathrm{P}}[\rho(\mathsf{X})]) \\
&= \mathbb{E}_{\mathrm{P}}\left[\rho(\mathsf{X}) \cdot L'(\mathsf{X}) + \lambda \cdot \varphi\left(\rho(\mathsf{X})\right) - a(\mathsf{X}) \cdot \rho(\mathsf{X}) - b \cdot \rho(\mathsf{X})\right] + b,
\end{aligned}$$

where $a(x) = a'(x) \cdot \mathrm{P}(x)$.

We can obtain the first order optimality conditions by taking the functional derivative. Suppose that $\delta > 0$ and $h \colon \mathcal{X} \to \mathbb{R}$ is any function. The functional derivative is given by,

$$\begin{aligned}
&\left.\frac{\mathrm{d}}{\mathrm{d}\delta} \mathcal{L}(\rho + \delta \cdot h; a, b)\right|_{\delta=0} \\
&= \left.\mathbb{E}_{\mathrm{P}}\left[h(\mathsf{X}) \cdot (L'(\mathsf{X}) + \lambda \cdot \varphi'\left(\rho(\mathsf{X}) + \delta \cdot h(\mathsf{X})\right) - a(\mathsf{X}) - b)\right]\right|_{\delta=0} \\
&= \mathbb{E}_{\mathrm{P}}\left[h(\mathsf{X}) \cdot (L'(\mathsf{X}) + \lambda \cdot \varphi'\left(\rho(\mathsf{X})\right) - a(\mathsf{X}) - b)\right].
\end{aligned}$$

Thus, the first order condition is,

$$0 = \frac{\mathrm{d}}{\mathrm{d}\delta} \mathcal{L}(\rho + \delta \cdot h; a, b) \bigg|_{\delta=0} = \mathbb{E}_{\mathrm{P}} \left[h(\mathsf{X}) \cdot (L'(\mathsf{X}) + \lambda \cdot \varphi' (\rho(\mathsf{X})) - a(\mathsf{X}) - b) \right],$$

for all $h \colon \mathcal{X} \to \mathbb{R}$. As the condition must hold for all h for an optimal $\rho^\star$, we have that for all $x \in \mathcal{X}$,

$$\begin{aligned} 0 &= L'(x) + \lambda \cdot \varphi' (\rho^\star(x)) - a(x) - b \\ \iff \qquad \varphi' (\rho^\star(x)) &= \frac{b + a(x) - L'(x)}{\lambda} \\ \iff \qquad \rho^\star(x) &= (\varphi')^{-1} \left(\frac{b - L'(x) + a(x)}{\lambda} \right). \end{aligned}$$

As required. □

C Proof of Corollary 3.4

Proof. First notice that the inner maximization in the DRO optimization Eq. (7) can be simplified as follows:

$$\begin{aligned} \sup_{\mathrm{Q} \in B_\varepsilon(\mathrm{P})} L(\mathrm{Q}) &= - \left(\inf_{\mathrm{Q} \in B_\varepsilon(\mathrm{P})} -L(\mathrm{Q}) \right) \\ &= - \left(\inf_{\mathrm{Q} \in \mathcal{D}(\mathcal{X})} \sup_{\lambda \geq 0} -L(\mathrm{Q}_\lambda) - \lambda \cdot (\varepsilon - D_\varphi(\mathrm{P} \parallel \mathrm{Q}_\lambda)) \right) \\ &= - \left(\sup_{\lambda \geq 0} \inf_{\mathrm{Q}_\lambda \in \mathcal{D}(\mathcal{X})} -L(\mathrm{Q}_\lambda) - \lambda \cdot (\varepsilon - D_\varphi(\mathrm{P} \parallel \mathrm{Q}_\lambda)) \right), \end{aligned}$$

where the last inequality follows from Fan's minimax Theorem [25, Theorem 2] noting that $\mathrm{Q} \mapsto L(\mathrm{Q}_\lambda)$ is linear and the selected φ per Definition 2.2 makes $\mathrm{Q} \mapsto D(\mathrm{P} \parallel \mathrm{Q}_\lambda)$ a convex lower semi-continuous function.

Now notice that the inner minimization of this simplification is exactly our idealized rejection distribution objective Eq. (8) when we negate the loss. As such, noticing that solutions to Eq. (8) are exactly given by $\mathrm{P} \cdot \rho$ yields the result after optimizing for the 'arguments' in the above DRO objective for both λ and Q_λ. □

D Proof of Corollary 4.1

We defer the proof of Corollary 4.1 to Appendix G, which covers any α including KL ($\alpha = 1$).

E Proof of Theorem 4.2

We breakdown Theorem 4.2 into two sub-theorems Theorem E.1 for each of the settings.

Theorem E.1. *Given the binary CPE setting presented in Theorem 2.1 and that we are given the optimal classifier $h^\star(x) = \mathrm{P}(\mathsf{Y} = 1 \mid \mathsf{X} = x)$, for any $\lambda > 0$ there exists a $\tau > 0$ such that the r_τ^{KL} rejector generated the optimal density ratio in Corollary 4.1 is equivalent to the optimal rejector in Theorem 2.1.*

The proofs of each are similar. We first make the observation that the rejector function r_τ^{KL} can be simplified as follows:

$$r_\tau^{\mathrm{KL}} = \llbracket L'(x) \leq -\lambda \left(\log Z + \log \tau \right) \rrbracket,$$

Now we note that given a fixed $\lambda > 0$ (which also fixes Z), the RHS term has a one-to-one mapping from $\mathbb{R}_+$ to $\mathbb{R}$. Thus all that is to verify is that thresholding $L'(x)$ is equivalent to the rejectors of Theorems 2.1 and N.1.

Proof. For the CPE case, from assumptions, we have that $L'(x) = \mathbb{E}_{\mathsf{Y} \sim \mathrm{P}(\mathsf{Y} \mid \mathsf{X} = x)} [\ell(\mathsf{Y}, h^\star(x))] = \mathbb{E}_{\mathsf{Y} \sim h^\star(\mathsf{X} = x)} [\ell(\mathsf{Y}, h^\star(x))$ as $h^\star$ is optimal. Thus thresholding used in r_τ^{KL} is equivalent to thresholding $L'(x)$ by keeping λ fixed and changine τ appropriately. □

F Proof of Theorem 4.3

To prove the theorem, we will be using the standard Hoeffding's inequality [12].

Theorem F.1 (Hoeffding's Inequality [12, Theorem 2.8])**.** *Let* $\mathsf{X}_1, \ldots, \mathsf{X}_n$ *be independent random variables such that* X_i *takes values in* $[a_i, b_i]$ *almost surely for all* $i \leq n$*. Defining* $\mathsf{X} = \sum_i \mathsf{X}_i - \mathbb{E}X_i$*, then for every* $t > 0$*,*

$$\Pr\left(\mathsf{X} \geq t\right) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$

Proof. Let us denote Z to be the normalizer with the true expectation $\mathbb{E}_{\mathrm{P}}$ and $\hat{Z}$ to be the normalizer with the empirical expectation $\mathbb{E}_{\hat{\mathrm{P}}}$.

As ℓ is bounded, we take note of the following bounds,

$$\exp(-B/\lambda) \leq \max\left\{\exp\left(\frac{-L'(x)}{\lambda}\right), Z, \hat{Z}\right\} \leq \exp(B/\lambda).$$

This can be simply verified by taking the smallest and largest values of the exp.

Now we simply have

$$\begin{aligned}
|\rho^{\mathrm{KL}}(x) - \hat{\rho}^{\mathrm{KL}}(x)| &= \exp\left(\frac{-L'(x)}{\lambda}\right) \cdot \left|\frac{1}{Z} - \frac{1}{\hat{Z}}\right| \\
&\leq \exp\left(\frac{B}{\lambda}\right) \cdot \left|\frac{1}{Z} - \frac{1}{\hat{Z}}\right| \\
&= \exp\left(\frac{B}{\lambda}\right) \cdot \frac{|Z - \hat{Z}|}{|Z \cdot \hat{Z}|} \\
&\leq \exp\left(\frac{B}{\lambda}\right)^3 \cdot |Z - \hat{Z}|.
\end{aligned}$$

Notice that $|Z - \hat{Z}|$ can be bounded via concentration inequality on (bounded) random variable $\exp\left(\frac{-L'(\mathsf{X})}{\lambda}\right)$.

Thus, we have that by Theorem F.1

$$\Pr\left(\left|\mathbb{E}_{\mathrm{P}}\left[\exp\left(\frac{-L'(x)}{\lambda}\right)\right] - \mathbb{E}_{\mathrm{P}}\left[\exp\left(\frac{-L'(x)}{\lambda}\right)\right]\right| > t\right) \leq 2\exp\left(\frac{-2 \cdot N \cdot t^2}{(b-a)^2}\right),$$

where $(b-a)^2 = (\exp(B/\lambda) - \exp(-B/\lambda))^2 = 4\sinh^2(B/\lambda)$.

Taking a union bound over $\mathcal{T}$, we have that

$$\Pr\left(\exists x \in \mathcal{T} : \left|\mathbb{E}_{\mathrm{P}}\left[\exp\left(\frac{-L'(x)}{\lambda}\right)\right] - \mathbb{E}_{\mathrm{P}}\left[\exp\left(\frac{-L'(x)}{\lambda}\right)\right]\right| > t\right) \leq 2M\exp\left(\frac{-N \cdot t^2}{2\sinh^2(B/\lambda)}\right).$$

Thus taking $t = \sinh(B/\lambda) \cdot \sqrt{\frac{2}{N} \cdot \log\left(\frac{2M}{\delta}\right)}$, we have that with probability $1 - \delta$ for $\delta > 0$, for any $x \in \mathcal{T}$,

$$\begin{aligned}
|\rho^{\mathrm{KL}}(x) - \hat{\rho}^{\mathrm{KL}}(x)| &\leq \exp\left(\frac{B}{\lambda}\right)^3 \cdot |Z - \hat{Z}| \\
&\leq \exp\left(\frac{B}{\lambda}\right)^3 \cdot \sinh(B/\lambda) \cdot \sqrt{\frac{2}{N} \cdot \log\left(\frac{2M}{\delta}\right)}.
\end{aligned}$$

As required.

□

G Proof of Theorem 4.5

Before proving the theorems, we first state some basic properties of ψ_α.

Lemma G.1. *For $\alpha \neq -1$, $\psi_\alpha(u \cdot v) = \psi_\alpha(u) \cdot \psi_\alpha(v)$ and $\psi_\alpha(1/v) = 1/\psi_\alpha(v)$. Furthermore, these statements hold when ψ_α is replaced with ψ_α^{-1}.*

Lemma G.2.

$$\psi_\alpha^{-1}(z) = \begin{cases} z^{\frac{2}{1-\alpha}} & \textit{if } \alpha \neq 1 \\ \exp(z) & \textit{otherwise} \end{cases}. \tag{18}$$

The above statements follows directly from definition and simple calculation. The next statement directly connects φ'_α to ψ_α.

Lemma G.3. *For $\alpha \neq 1$,*

$$(\varphi'_\alpha)(z) = \frac{2}{\alpha - 1} \cdot \psi_\alpha(1/z). \tag{19}$$

Proof. We prove this via cases.

- $\alpha = -1$:

$$\varphi'_{-1}(z) = \frac{\mathrm{d}}{\mathrm{d}z}(-\log z) = -z^{-1} = -1 \cdot z^{-\frac{1-\alpha}{2}} = \frac{2}{\alpha - 1} \cdot \psi_\alpha(1/z).$$

- $\alpha \neq \pm 1$:

$$\varphi'_\alpha(z) = \frac{\mathrm{d}}{\mathrm{d}z}\left(\frac{4}{1-\alpha^2} \cdot \left(1 - z^{\frac{1+\alpha}{2}}\right)\right) = -\frac{2}{1-\alpha} z^{\frac{\alpha-1}{2}} = -\frac{2}{1-\alpha} \cdot z^{-\frac{1-\alpha}{2}} = \frac{2}{\alpha - 1} \cdot \psi_\alpha(1/z)$$

□

We now define the following constants depending on α:

$$c_\alpha = \begin{cases} \psi_\alpha^{-1}\left(-\frac{2}{1-\alpha}\right) & \text{if } \alpha \neq 1 \\ \psi_\alpha^{-1}(-1) = \exp(-1) & \text{otherwise.} \end{cases} \tag{20}$$

Lemma G.4. *For $\alpha \neq 1$,*

$$(\varphi'_\alpha)^{-1}(z) = c_\alpha \cdot \frac{1}{\psi_\alpha^{-1}(z)}. \tag{21}$$

Proof. We prove this via cases.

- $\alpha = -1$:

$$\varphi'_{-1}(z) = \frac{\mathrm{d}}{\mathrm{d}z}(-\log z) = -z^{-1} = -1 \cdot z^{-\frac{1-\alpha}{2}} = -1 \cdot \psi_\alpha(1/z).$$

Then,

$$(\varphi'_{-1})^{-1} = \frac{1}{\psi_\alpha^{-1}(-1 \cdot z)} = \psi_\alpha^{-1}\left(\frac{1}{-1 \cdot z}\right) = \psi_\alpha^{-1}(-1) \cdot \frac{1}{\psi_\alpha^{-1}(z)}. = c_\alpha \cdot \frac{1}{\psi_\alpha^{-1}(z)}.$$

- $\alpha \neq \pm 1$:

$$\varphi'_\alpha(z) = \frac{\mathrm{d}}{\mathrm{d}z}\left(\frac{4}{1-\alpha^2} \cdot \left(1 - z^{\frac{1+\alpha}{2}}\right)\right) = -\frac{2}{1-\alpha} z^{\frac{\alpha-1}{2}} = -\frac{2}{1-\alpha} \cdot z^{-\frac{1-\alpha}{2}} = -\frac{2}{1-\alpha} \cdot \psi_\alpha(1/z)$$

Then,

$$(\varphi'_\alpha)^{-1} = \frac{1}{\psi_\alpha^{-1}\left(-\frac{1-\alpha}{2} \cdot z\right)} = \frac{1}{\psi_\alpha^{-1}\left(-\frac{1-\alpha}{2}\right) \cdot \psi_\alpha^{-1}(z)} = \psi_\alpha^{-1}\left(-\frac{2}{1-\alpha}\right) \cdot \frac{1}{\psi_\alpha^{-1}(z)} = c_\alpha \cdot \frac{1}{\psi_\alpha^{-1}(z)}$$

□

Lemma G.5. *For $\alpha = 1$,*

$$(\varphi'_\alpha)^{-1}(z) = c_\alpha \cdot \psi_\alpha^{-1}(z). \tag{22}$$

Proof. • $\underline{\alpha = 1}$:

$$\varphi'_1(z) = \frac{\mathrm{d}}{\mathrm{d}z}(z \log z) = \log z + 1$$

Then,

$$(\varphi'_{-1})^{-1} = \exp(z-1) = \exp(-1) \cdot \exp(z) = \psi_1^{-1}(-1) \cdot \psi_1^{-1}(z) = c_\alpha \cdot \psi_\alpha^{-1}(z)$$

□

Thus now via Lemmas G.4 and G.5, we can prove the Theorems.

Proof of Corollary 4.1 and Theorem 4.5. The proof follows from utilizing either Lemmas G.4 and G.5 in conjunction with Theorem 3.3.

• $\underline{\alpha = 1}$: We have that,

$$\begin{aligned}\rho_\lambda^\varphi(x) &= (\varphi')^{-1}\left(\frac{a(x) - L'(x) + b}{\lambda}\right) \\ &= c_\alpha \cdot \psi_\alpha^{-1}\left(\frac{a(x) - L'(x) + b}{\lambda}\right) \\ &= \exp\left(\frac{a(x) - L'(x) + b - 1}{\lambda}\right).\end{aligned}$$

As $\rho_\lambda^\varphi(x)$ for $\alpha = 1$ is already positive by 'exp', by complementary slackness, the Lagrange multipliers $a(\cdot) = 0$. Hence, we can further simplify the above,

$$\rho_\lambda^\varphi(x) = \exp\left(\frac{-L'(x) + b - \lambda}{\lambda}\right).$$

The normalizer then can be easily calculated, renaming $b' = b - \lambda$, we simplify have that

$$\begin{aligned}\rho_\lambda^\varphi(x) &= \exp\left(\frac{-L'(x) + b'}{\lambda}\right) \\ &= \frac{1}{\exp(-b'/\lambda)} \cdot \exp\left(\frac{-L'(x)}{\lambda}\right),\end{aligned}$$

which by normalization condition and setting $Z \doteq \exp(-b'/\lambda)$,

$$\begin{aligned}1 &= \mathbb{E}\left[\frac{1}{Z} \cdot \exp\left(\frac{-L'(x)}{\lambda}\right)\right] \\ \iff \quad Z &= \mathbb{E}\left[\exp\left(\frac{-L'(x)}{\lambda}\right)\right],\end{aligned}$$

which completes the case (Corollary 4.1).

• $\underline{\alpha \neq 1}$: We have that,

$$\begin{aligned}\rho_\lambda^\varphi(x) &= (\varphi')^{-1}\left(\frac{a(x) - L'(x) + b}{\lambda}\right) \\ &= c_\alpha \cdot \left(\psi_\alpha^{-1}\left(\frac{a(x) - L'(x) + b}{\lambda}\right)\right)^{-1} \\ &\overset{(a)}{=} \psi_\alpha^{-1}\left(\frac{2}{\alpha - 1}\right) \cdot \left(\psi_\alpha^{-1}\left(\frac{a(x) - L'(x) + b}{\lambda}\right)\right)^{-1} \\ &= \psi_\alpha^{-1}\left(\frac{2}{\alpha - 1}\right) \cdot \psi_\alpha^{-1}\left(\left(\frac{a(x) - L'(x) + b}{\lambda}\right)^{-1}\right) \\ &= \psi_\alpha^{-1}\left(\frac{2}{\alpha - 1} \cdot \left(\frac{a(x) - L'(x) + b}{\lambda}\right)^{-1}\right),\end{aligned}$$

where (a) we exploit the that $(\psi_\alpha^{-1}(z))^{-1} = \psi_\alpha^{-1}(1/z)$ via Lemma G.1. □

H Proof of Corollary 4.6

Proof. First we simplify the density ratio rejector.

$$\rho_\lambda^\alpha(x) = \psi_\alpha^{-1}\left(\frac{2}{\alpha-1}\cdot\left(a(x) - \frac{L'(x)}{\lambda} + b\right)^{-1}\right)$$
$$= \left(\frac{\alpha-1}{2}\cdot\left(a(x) - \frac{L'(x)}{\lambda} + b\right)\right)^{\frac{\alpha-1}{2}}.$$

We suppose that it is possible for

$$\left(\frac{\alpha-1}{2}\cdot\left(a(x) - \frac{L'(x)}{\lambda} + b\right)\right)^{\frac{\alpha-1}{2}} < 0$$

for values of $a(x)$, b, and λ. Otherwise, $\alpha(x) = 0$ and we are done due to the above equation's non-negativity.

Let $x \in \mathcal{X}$ be arbitrary. Suppose that $a(x) = 0$. Then by complementary slackness, we have that

$$\left(\frac{\alpha-1}{2}\cdot\left(-\frac{L'(x)}{\lambda} + b\right)\right)^{\frac{\alpha-1}{2}} > 0 \iff \qquad b - \frac{L'(x)}{\lambda} > 0.$$

By contra-positive, we have that $b - L'(x)/\lambda \leq 0$ implies that $a(x) \neq 0$. By prime feasibility, in this case we also have $\rho(x) = 0$. We can solve either case by using the maximum as stated. □

I Proof of Corollary 4.7

Proof of Corollary 4.6. The proof directly follows from a property of the α-divergence when one of the measure have disjoint support. From [3] we have.

Theorem I.1 ([3, Section 3.4.1 (4)]). *For α-divergences, we have that*

1. *For $\alpha \leq -1$, $D_\alpha(\mathrm{P} \parallel \mathrm{Q}) = \infty$ when $\mathrm{P}(x) \neq 0$ and $\mathrm{Q}(x) = 0$ for some $x \in \mathcal{X}$.*

This result immediately gives the result, as otherwise the objective function is ∞. □

J Proof of Corollary 4.8

Proof. We first note simplifying $\rho_\lambda^{\alpha=3}$ via Corollary 4.6 yields:

$$\rho_\lambda^{\alpha=3}(x) = \max\left\{0, b - \frac{L'(x)}{\lambda}\right\}.$$

Thus our goal is to solve b to give $\mathbb{E}_{\mathrm{P}}[\rho_\lambda^{\alpha=3}(\mathsf{X})] = 1$.

Now, consider the following,

$$1 + \frac{\mathbb{E}_{\mathrm{P}}\left[L'(\mathsf{X})\right]}{\lambda} - \frac{L'(x)}{\lambda} > 0 \iff \lambda > L'(x) - \mathbb{E}_{\mathrm{P}}\left[L'(\mathsf{X})\right],$$

where the latter holds uniformly for all x from assumptions on λ.

Thus setting $b = 1 + \frac{\mathbb{E}_{\mathrm{P}}[L'(\mathsf{X})]}{\lambda}$, we simplify

$$\int \max\left\{0, 1 + \frac{\mathbb{E}_{\mathrm{P}}\left[L'(\mathsf{X})\right]}{\lambda} - \frac{L'(\mathsf{X})}{\lambda}\right\} \mathrm{dP}(x)$$
$$= \int 1 + \frac{\mathbb{E}_{\mathrm{P}}\left[L'(\mathsf{X})\right]}{\lambda} - \frac{L'(\mathsf{X})}{\lambda} \mathrm{dP}(x)$$
$$= 1.$$

As such, $b = 1 + \frac{\mathbb{E}_{\mathrm{P}}[L'(\mathsf{X})]}{\lambda}$ solves the required normalization. Substituting b back into $\rho_\lambda^{\alpha=3}(x)$ yields the Theorem. □

K Proof of Theorem 4.9

Proof. First we note that for any meas R, $\lambda > 2B$ implies that $\lambda > L'(x) - \mathbb{E}_{\mathrm{R}}\left[L'(\mathsf{X})\right]$ (taking largest and smallest values of ℓ.

As such, for both $\mathrm{P}, \hat{\mathrm{P}}$ have closed forms Corollary 4.8

Thus, the bound can be simply shown to have,

$$|\rho_\lambda^{\alpha=3}(x) - \hat{\rho}_\lambda^{\alpha=3}(x)| = \frac{1}{\lambda} \cdot \left| \mathbb{E}_{\mathrm{P}}\left[L'(\mathsf{X})\right] - \mathbb{E}_{\hat{\mathrm{P}}}\left[L'(\mathsf{X})\right] \right|.$$

Thus by Hoeffding's inequality Theorem F.1 and union bound (see for instance the proof of Theorem 4.3), setting $t = B \cdot \sqrt{\frac{2}{N} \log\left(\frac{2M}{\delta}\right)}$, we have that with probability $1-\delta$ for all $x \in \mathcal{T}$,

$$|\rho_\lambda^{\alpha=3}(x) - \hat{\rho}_\lambda^{\alpha=3}(x)| \leq \frac{B}{\lambda} \cdot \sqrt{\frac{2}{N} \log\left(\frac{2M}{\delta}\right)}.$$

As required. □

L Broader Impact

The paper presents work which reinterprets the classification with rejection problem in terms of learning distributions and density ratios. Beyond advancing Machine Learning in general, potential societal consequences include enhancing the understanding of the rejection paradigm and, consequentially, the human-in-the-loop paradigm. The general rejection setting aims to prevent models from making prediction when they are not confident, which can have societal significance when deployed in high stakes real life scenarios — allowing for human intervention.

M Distribution Generalization Bounds

In the following, we seek to provide generalization bounds on $\hat{\mathrm{Q}}_{\lambda,N} \doteq \hat{\mathrm{P}}_N \cdot \hat{\rho}_{\lambda,N}$. That is, one seeks to know that as $N \to \infty$ how and if $\hat{\mathrm{Q}}_{\lambda,N} \to \mathrm{Q}$. A natural measure of distance for probability measures is *total variation* [20],

$$\mathrm{TV}(\mathrm{P},\mathrm{Q}) \doteq \frac{1}{2} \cdot \|\mathrm{P} - \mathrm{Q}\|_1 = \int |\mathrm{Q}(x) - \mathrm{P}(x)| \, \mathrm{d}x.$$

Let us also define

$$\|\mathrm{P} - \mathrm{Q}\|_\infty = \sup_{x \in \mathcal{X}} |\mathrm{Q}(x) - \mathrm{P}(x)|.$$

One immediately gets a rate if we assume that $|\mathcal{X}|$ is finite and we have a bound for $\hat{\rho}_{\lambda,N}$.

Theorem M.1. *Suppose that $|\mathcal{X}| = M < \infty$ is finite and bounded density ratio $|\hat{\rho}_{\lambda,N}| \leq B'$ for $B' > 0$. Then, if we have that with probability $1-\delta$ that ,*

$$\|\rho(x) - \hat{\rho}_{N,\lambda}(x)\|_\infty \leq C \cdot \left(\sqrt{\frac{1}{N} \cdot \log \frac{2M}{\delta}} + C' \right),$$

for some $C, C' > 0$, then we have that

$$\mathrm{TV}(\mathrm{Q}, \hat{\mathrm{Q}}_{\lambda,N}) = \mathcal{O}\left(\sqrt{\frac{M^2 \log M}{N}} \right). \tag{23}$$

Proof. Noting that the empirical distribution is a sum of Dirac deltas $\hat{\mathrm{P}}_N(x) = \sum_{i \in [N]} \delta(x - x_i)$, we can establish a simple bound via a consequence of Hoeffding's Theorem Theorem F.1 (also noting that $\mathrm{P} = \mathbb{E}\hat{\mathrm{P}}_N$). We have that,

$$\Pr(|\mathrm{P}(x) - \hat{\mathrm{P}}_N(x)| \geq t) \leq 2 \exp\left(-2Nt^2\right).$$

Thus,

$$\Pr(\forall x \in \mathcal{X} : |\mathrm{P}_N(x) - \hat{\mathrm{P}}_N(x)| \geq t) \leq 2M \exp\left(-2Nt^2\right).$$

Setting $t = \sqrt{\frac{1}{2N} \log \frac{2M}{\delta}}$, we have with probability $1 - \delta$

$$\|\mathrm{P} - \hat{\mathrm{P}}_N\|_\infty \leq \sqrt{\frac{1}{2N} \log \frac{2M}{\delta}}.$$

We now consider the following:

$$\begin{aligned}
\|\mathrm{Q} - \hat{\mathrm{Q}}_{\lambda,N}\|_\infty &= \|\mathrm{P} \cdot \rho - \hat{\mathrm{P}}_N \cdot \hat{\rho}_{\lambda,N}\|_\infty \\
&= \|\mathrm{P} \cdot (\rho - \rho_{\hat{\lambda,N}}) + (\mathrm{P} - \hat{\mathrm{P}}_N) \cdot \hat{\rho}_{\lambda,N}\|_\infty \\
&\leq \|\mathrm{P} \cdot (\rho - \hat{\rho}_{\lambda,N})\|_\infty + \|(\mathrm{P} - \hat{\mathrm{P}}_N) \cdot \hat{\rho}_{\lambda,N}\|_\infty \\
&\leq \|(\rho - \hat{\rho}_{\lambda,N})\|_\infty + B' \cdot \|(\mathrm{P} - \hat{\mathrm{P}}_N)\|_\infty.
\end{aligned}$$

Taking a union bound of the above inequality and our assumption, we have that, for some $C > 0$, with probability $1 - \delta$,

$$\begin{aligned}
\|\mathrm{Q} - \hat{\mathrm{Q}}_{\lambda,N}\|_\infty &\leq \|(\rho - \hat{\rho}_{\lambda,N})\|_\infty + B' \cdot \|(\mathrm{P} - \hat{\mathrm{P}}_N)\|_\infty \\
&\leq C \cdot \sqrt{\frac{1}{N} \log \frac{4M}{\delta} + C'} + B' \cdot \sqrt{\frac{1}{2N} \log \frac{4M}{\delta}} \\
&= \mathcal{O}\left(\sqrt{\frac{\log M}{N}}\right).
\end{aligned}$$

Converting the bound to TV amounts to simply summing over $\mathcal{X}$, which gives $\mathrm{TV}(\mathrm{Q}, \hat{\mathrm{Q}}_{\lambda,N}) = \mathcal{O}(\sqrt{(M^2 \log M)/N})$, as required. □

Notably, with appropriate assumptions, Theorems 4.3 and 4.9 can be used with Theorem M.1 to get bounds for $\hat{\mathrm{Q}}^{\mathrm{KL}}_{\lambda,N}$ and $\hat{\mathrm{Q}}^{\alpha=3}_{\lambda,N}$.

N Rejection for Regression

In the main-text of the paper we have focused on classification. However, many of the idea discussed can be extended for the regression setting. For instance, similar to Chow's Rule in Theorem 2.1 we can express the regression equivalent to the optimal solution of Eq. (2).

Theorem N.1 (Optimal Regression Rejection [72])**.** *Let us consider the regression setting, where $\mathcal{Y} = \mathcal{Y}' = \mathbb{R}$ and $\ell(y, y') = \frac{1}{2}(y - y')^2$. Then w.r.t. Eq. (2), the optimal model is given by $h^\star(x) = \mathbb{E}_{\mathrm{P}}[\mathsf{Y} \mid \mathsf{X} = x]$ and the optimal rejector is given by $r^\star(x) = [\![\sigma^2(x) \leq c]\!]$, where $\sigma^2(x)$ is the conditional variance of Y given $\mathsf{X} = x$.*

For regression, there is no clear analogous notion of output "confidence score" unless the model explicitly outputs probabilities. Indeed, rejection method for regression explicitly requires the estimation of the target variable's conditional variance [72].

Similar to the CPE case, our KL density ratio rejector can provide a rejection policy equivalent to the typical case.

Theorem N.2. *Given the regression setting presented in Theorem N.1 and that we are given optimal regressor $h^\star(x) = \mathbb{E}_{\mathrm{P}}[\mathsf{Y} \mid \mathsf{X} = x]$, for any $\lambda > 0$ there exists a $\tau > 0$ such that the r^{KL}_τ rejector generated the optimal density ratio in Corollary 4.1 is equivalent to the optimal rejector in Theorem N.1.*

Proof. The proof follows similarly to the proof of Theorem E.1. The regression case follows almost identically, by noticing that $L'(x) = \sigma^2(x)$ is the variance. This is similar to the proof of Theorem N.1 [72]. □

Despite the equivalence, there is a difficult in using the density ratio rejectors, as per the closed form equations of Section 4, for regression. Estimating $L'(x) = \sigma^2(x)$ is challenging. Unlike classification where learning calibrated classifiers has a variety of approaches, learning a regression model which explicitly outputs probabilities is quite difficult. As such, approximating $\mathrm{P}(\mathsf{Y} \mid \mathsf{X} = x)$ with the model $h(x)$ cannot be done.

O Gradient of Density Ratio Objective

As an alternative to the closed-form rejectors explored in the main-text, one may want to explore a method to learn ρ iteratively. We consider the gradients of the optimization problem in Eq. (9). In practice, we found that we were unable to learn such rejectors via taking gradient updates and thus leave a concrete implementation of the idea for future work.

The idea comes from utilizing the "variational" aspect of the GVI formulation (which was not explored in the main-text). We suppose that the rejectors we are interested in come from a parameterized family. In particular, we consider the self-normalizing family $\rho_\vartheta / Z_\vartheta$, where $Z_\vartheta = \mathbb{E}_{\mathrm{P}}[\pi_\vartheta(\mathsf{X})]$ normalizes the rejector such that $\mathbb{E}_{\mathrm{P}}[\rho_\vartheta(\mathsf{X})] = 1$. Having the Z_ϑ normalizing term means that the constraint in Eq. (9) is satisfied for any ϑ. The only constraint that we must have for π_ϑ is non-negativity, *i.e.*, π_ϑ is a neural network with exponential last activation functions from $\mathcal{X} \to \mathbb{R}_+$. By setting a parametric form of ρ_ϑ, we implicitly restrict the set of idealized distributions to $\mathrm{Q}_\vartheta = \mathrm{P} \cdot \rho_\vartheta$. The gradients of such a parametric form can be calculated as follows.

Corollary O.1. *Let $\rho_\vartheta = \pi_\vartheta(x)/Z_\vartheta$. Then the gradient of Eq. (9) w.r.t. ϑ is given by,*

$$\mathbb{E}_{\mathrm{P}_{\mathrm{x,y}}}\left[\nabla_\vartheta \left(\frac{\pi_\vartheta(\mathsf{X})}{Z_\vartheta}\right) \cdot \left(\varphi'\left(\frac{\pi_\vartheta(\mathsf{X})}{Z_\vartheta}\right) + \lambda \cdot \ell(\mathsf{Y}, h(\mathsf{X}))\right)\right]. \tag{24}$$

Proof. The proof is immediate from differentiation of Eq. (9). □

An alternative form of Eq. (24) can be found by noticing that $\nabla f = f \cdot \nabla \log f$. This provides an expression in terms of the log density ratio.

$$\mathbb{E}_{\mathrm{P}_{\mathrm{x,y}}}\left[\frac{\pi_\vartheta(\mathsf{X})}{Z_\vartheta} \cdot \nabla_\vartheta \log \left(\frac{\pi_\vartheta(\mathsf{X})}{Z_\vartheta}\right) \cdot \left(\varphi'\left(\frac{\pi_\vartheta(\mathsf{X})}{Z_\vartheta}\right) + \lambda \cdot \ell(\mathsf{Y}, h(\mathsf{X}))\right)\right].$$

One will notice that the gradient is in-fact the log-likelihood of the idealized distribution: noting P free of ϑ, we have $\nabla_\vartheta \log(\pi_\vartheta(\mathsf{X})/Z_\vartheta) = \nabla_\vartheta \log(\mathrm{P}(\mathsf{X}) \cdot \pi_\vartheta(\mathsf{X})/Z_\vartheta) = \nabla_\vartheta \log \mathrm{Q}_\vartheta$. As such, the gradient Eq. (24) is equivalent to the gradient of a weighted log-likelihood.

Despite the potential nice form of the gradient, we found that learning rejector through this was not possible. One limiting factor of computing such gradients is that we need to estimate Z_ϑ at each gradient calculation, *i.e.*, this must happen whenever ϑ changes. This can be particularly costly when $\mathcal{X}$ is high-dimensional. Secondly, we suspect that the model capacity of π_ϑ was not sufficient: we only tested on simple neural networks and convolutions neural networks, mirroring the architecture of the base classifiers used in our experimental setting.

P Finding the Best Rejection Cost

Both the baselines and density ratio approaches evaluated in Section 5 require a tuning of hyperparameters c, τ. Practically, we are more interested in the rejection rate $\mathrm{P}[r(\mathsf{X}) = 1]$ rather than the abstract choices of c, τ. Hence, one often wishes to select c, τ according to a coverage constraint.

One of the simplest choices of selecting c, τ is to utilize a calibration set and approximate the rejection rate / coverage on this data. Our experiments in Section 5 mirrors this in Tables 1 and I by treating the final test set as the calibration data.

Other works in the literature utilize more sophisticated methods for deriving coverage constraints. [29] picks threshold values which provide a guarantee on a coverage-modified risk requirement. Another approach [57] does not require an additional calibration set and instead fits threshold values τ by exploiting stacking confidences and (stratified) cross-fitting. Although one could consider such approaches for all baselines, including those which utilize the cost-model of rejection via c, one will have to pay a price in retraining the learned rejector r for each instance of the hyperparameter c that is searched. As a result, approaches which can trade-off accuracy and coverage via a threshold variable τ (rather than an expensive retraining) are computationally preferable when exhaustively searching for a good c, τ.

We leave more sophisticated methods for selecting τ for density ratio rejectors for future work.

Q Additional Experimental Details

Q.I Training Settings

The neural network architecture used to train the base classifiers and baselines are almost identical. For the baseline approaches which have output dimension which is different than the output of the original neural network, we modify the last linear layer of the base classifier's architecture to fit the baseline's requirements, *e.g.*, adding an additional output dimension for rejection in DEFER. Our architectures utilize batch normalization [38] and dropout [65] in a variety of places. Training settings are mostly identical, with some baselines requiring slight changes.

The base model's architecture is as follows.

- HAR (CC BY 4.0): We utilize a two hidden layer neural network with batch normalization. Both hidden layer is 64 neurons and the activation function is the sigmoid function. We take a 64 batch size, 40 training epochs and a 0.0001 learning rate.
- Gas Drift (CC BY 4.0): We utilize a two hidden layer neural network with batch normalization. Both hidden layers are 64 neurons and the activation function is the sigmoid function. We take a 64 batch size, 40 training epochs and a 0.0001 learning rate.
- MNIST (CC BY-SA 3.0): We utilize a convolutional neural network with two convolutional layers and two linear layers. The architecture follows directly from the MNIST example for PyTorch. We utilize the sigmoid function activation function. We take a 256 batch size, 40 training epochs and a 0.0001 learning rate.
- CIFAR-10 (CC BY 4.0): We utilize a ResNet-18 classifier. Random cropping and horizontal flipping data augmentation is utilized in training. We take a 256 batch size, 40 training epochs and a 0.0001 learning rate.
- OrgMNIST (CC BY 4.0): We utilize a ResNet-18 classifier. We take a 256 batch size, 40 training epochs and a 0.0001 learning rate.
- OctMNIST (CC BY 4.0): We utilize a ResNet-18 classifier. We take a 256 batch size, 40 training epochs and a 0.0001 learning rate.

For CSS, as noted in [53], batch normalization is needed at the final layer to stabilize training.

All training utilizes the Adam [41] optimizer.

All datasets we consider are in the public domain, *e.g.*, UCI [6].

Q.II Extended Table and Plots

Table I presents an extended version of Table 1 over coverage targets of 80% and 90%. The standard deviation is additionally reported. One can see the observation from the main text are consistent with this extended table.

Plots Figs. I to III show Fig. 2 over and extend region of acceptable coverage percentages. In addition, we include a larger range of noise rates for HAR and Gas Drift. For MNIST, we explore a larger range of noise rates for MNIST in Fig. VII for our density ratio rejectors. We find that the findings in the main text are extended to these additional noise rates. In particular, we find that the our density ratio rejectors can be competitive with the various baseline approaches. We find that our density ratio approaches can have more fine-grained trade-offs at higher ranges of acceptance coverage. This is an important region where the budge for rejection may be low (and only a few examples, *e.g.* $< 10\%$, can be rejected). Indeed, the baseline approaches which do not 'wrap' a base model require a lower *maximum acceptance coverage* as the noise rate increases (the approaches require a higher rejection $\%$ for any type of rejection). Nevertheless, we do see a downside of the density ratio approach: the

Table I: Extended Summary performance of rejection methods over all baselines and datasets targeting 80% and 90% coverage. Each cell reports the "accuracy (accuracy s.t.d.)[coverage (coverage s.t.d.)]" values.

			Base	KL-Rej	(α=3)-Rej	PredRej	CSS	DEFER	GCE
Target Coverage: 80%	Clean	HAR	97.38 (0.34) [100.00 (0.00)]	99.93 (0.06) [80.74 (0.84)]	99.93 (0.06) [81.82 (0.74)]	98.86 (0.44) [84.86 (8.45)]	99.58 (0.31) [81.39 (1.48)]	99.44 (0.19) [80.16 (1.54)]	99.31 (0.52) [81.78 (4.69)]
		Gas Drift	94.10 (1.87) [100.00 (0.00)]	99.16 (0.62) [80.04 (0.78)]	99.16 (0.62) [80.24 (0.75)]	98.12 (1.03) [80.09 (6.03)]	98.68 (0.39) [80.33 (1.18)]	98.06 (0.97) [80.49 (1.09)]	97.62 (0.36) [80.43 (3.92)]
		MNIST	98.55 (0.19) [100.00 (0.00)]	99.93 (0.03) [87.37 (0.66)]	99.93 (0.03) [88.08 (0.58)]	99.18 (0.11) [74.03 (4.05)]	99.95 (0.03) [83.21 (0.89)]	99.93 (0.01) [80.38 (1.60)]	99.85 (0.04) [80.02 (3.75)]
		CIFAR-10	90.20 (0.29) [100.00 (0.00)]	97.22 (0.18) [80.27 (0.23)]	97.17 (0.17) [80.53 (0.21)]	91.40 (0.89) [74.42 (16.73)]	95.45 (0.95) [80.56 (1.14)]	93.72 (0.94) [80.57 (4.78)]	94.25 (0.31) [80.91 (0.83)]
		OrganMNIST	89.10 (1.06) [100.00 (0.00)]	96.55 (0.78) [80.25 (1.42)]	96.52 (0.82) [80.33 (1.18)]	93.79 (0.95) [81.77 (6.86)]	94.49 (0.65) [80.17 (4.32)]	93.47 (0.55) [80.16 (4.59)]	93.68 (1.73) [80.04 (2.48)]
		OctMNIST	91.93 (0.22) [100.00 (0.00)]	97.08 (0.20) [80.94 (0.28)]	97.18 (0.19) [80.22 (0.20)]	93.43 (0.81) [85.57 (8.22)]	95.40 (1.06) [81.05 (2.58)]	94.66 (0.31) [85.32 (4.92)]	94.91 (0.79) [86.67 (3.26)]
	Noisy (25%)	HAR	96.51 (0.44) [100.00 (0.00)]	98.56 (1.66) [81.16 (15.40)]	98.56 (1.66) [81.13 (15.41)]	97.22 (0.55) [80.24 (7.73)]	97.82 (0.35) [80.50 (0.67)]	97.78 (1.00) [69.11 (6.36)]	98.85 (0.43) [80.18 (3.11)]
		Gas Drift	93.84 (1.81) [100.00 (0.00)]	97.30 (2.60) [80.02 (16.41)]	97.28 (2.59) [80.09 (16.38)]	95.87 (2.48) [81.51 (9.20)]	98.71 (0.27) [80.32 (1.91)]	99.02 (0.48) [76.54 (2.11)]	97.52 (0.75) [75.31 (2.76)]
		MNIST	97.88 (0.23) [100.00 (0.00)]	99.89 (0.01) [80.44 (1.54)]	99.89 (0.01) [80.64 (1.36)]	98.00 (0.20) [92.95 (10.17)]	99.94 (0.03) [80.38 (1.08)]	99.93 (0.03) [81.45 (0.58)]	99.95 (0.03) [81.20 (1.55)]
		CIFAR-10	85.31 (0.18) [100.00 (0.00)]	92.25 (0.33) [81.61 (0.98)]	92.50 (0.38) [80.71 (1.00)]	85.84 (1.17) [88.25 (23.07)]	89.58 (1.28) [81.83 (2.48)]	90.93 (0.45) [80.29 (1.59)]	92.22 (0.12) [80.55 (0.41)]
		OrganMNIST	89.10 (0.77) [100.00 (0.00)]	95.86 (1.36) [82.07 (1.95)]	96.29 (0.94) [80.93 (1.16)]	93.40 (0.74) [83.78 (5.13)]	96.74 (0.68) [81.19 (0.77)]	94.67 (1.17) [80.39 (5.45)]	94.48 (0.77) [80.39 (3.51)]
		OctMNIST	91.89 (0.20) [100.00 (0.00)]	97.17 (0.21) [80.29 (0.87)]	97.10 (0.21) [80.72 (0.78)]	93.42 (1.10) [83.14 (12.31)]	95.49 (1.35) [81.20 (4.24)]	94.08 (0.64) [89.10 (2.42)]	94.63 (1.14) [77.56 (13.79)]
Target Coverage: 90%	Clean	HAR	97.38 (0.34) [100.00 (0.00)]	99.56 (0.17) [90.34 (0.38)]	99.58 (0.17) [90.19 (0.42)]	98.19 (0.74) [90.20 (14.01)]	98.75 (0.73) [90.89 (1.26)]	99.23 (0.19) [90.34 (0.29)]	99.37 (0.24) [90.27 (0.75)]
		Gas Drift	94.10 (1.87) [100.00 (0.00)]	98.05 (1.71) [90.24 (1.58)]	98.12 (1.58) [90.06 (1.37)]	96.64 (2.06) [90.30 (6.65)]	96.96 (0.56) [88.09 (0.70)]	97.46 (0.44) [84.51 (2.08)]	96.26 (0.53) [87.24 (2.24)]
		MNIST	98.55 (0.19) [100.00 (0.00)]	99.89 (0.04) [90.57 (0.55)]	99.89 (0.04) [90.84 (0.53)]	98.87 (0.13) [91.82 (1.52)]	99.93 (0.02) [90.42 (0.36)]	99.80 (0.03) [90.22 (0.39)]	99.84 (0.03) [90.72 (3.41)]
		CIFAR-10	90.20 (0.29) [100.00 (0.00)]	94.41 (0.21) [90.19 (0.19)]	94.43 (0.21) [90.14 (0.20)]	90.32 (0.22) [93.66 (10.05)]	92.77 (0.68) [90.16 (0.72)]	92.98 (0.86) [91.01 (1.63)]	91.54 (0.44) [90.43 (0.47)]
		OrganMNIST	89.10 (1.06) [100.00 (0.00)]	92.60 (0.87) [90.15 (1.43)]	92.57 (0.94) [90.27 (1.20)]	91.44 (1.68) [90.49 (7.46)]	92.52 (0.66) [90.42 (1.78)]	92.35 (0.58) [90.16 (1.68)]	90.88 (0.75) [90.07 (2.00)]
		OctMNIST	91.93 (0.22) [100.00 (0.00)]	94.98 (0.26) [91.35 (0.43)]	95.17 (0.25) [90.69 (0.38)]	92.23 (0.60) [96.63 (4.21)]	91.96 (1.94) [91.50 (1.62)]	93.49 (0.46) [91.18 (2.08)]	94.20 (0.68) [90.38 (2.12)]
	Noisy (25%)	HAR	96.51 (0.44) [100.00 (0.00)]	98.28 (1.44) [90.11 (8.10)]	98.29 (1.45) [90.40 (7.86)]	96.94 (0.77) [88.55 (4.82)]	98.48 (0.29) [84.94 (1.07)]	97.78 (1.00) [69.11 (6.36)]	98.56 (0.35) [89.34 (1.23)]
		Gas Drift	93.84 (1.81) [100.00 (0.00)]	96.58 (2.05) [90.75 (7.96)]	96.68 (2.11) [90.04 (8.34)]	95.11 (1.85) [91.71 (6.72)]	98.71 (0.27) [80.32 (1.91)]	99.02 (0.48) [76.54 (2.11)]	97.52 (0.75) [75.31 (2.76)]
		MNIST	97.88 (0.23) [100.00 (0.00)]	99.71 (0.03) [90.01 (0.99)]	99.70 (0.04) [90.03 (0.89)]	98.00 (0.20) [92.95 (10.17)]	99.82 (0.06) [90.33 (0.61)]	99.89 (0.03) [83.47 (0.43)]	99.86 (0.02) [90.81 (0.85)]
		CIFAR-10	85.31 (0.18) [100.00 (0.00)]	89.43 (0.39) [90.44 (0.94)]	89.42 (0.39) [90.43 (0.92)]	85.66 (0.69) [91.06 (17.80)]	89.71 (1.01) [83.40 (1.44)]	89.94 (0.92) [83.03 (1.17)]	92.14 (0.24) [81.90 (1.23)]
		OrganMNIST	89.10 (0.77) [100.00 (0.00)]	92.56 (0.62) [90.22 (1.70)]	92.53 (0.64) [90.29 (1.60)]	92.24 (0.76) [90.61 (2.35)]	91.92 (1.05) [90.22 (1.17)]	91.77 (0.94) [90.11 (2.06)]	92.17 (0.24) [90.91 (0.45)]
		OctMNIST	91.89 (0.20) [100.00 (0.00)]	95.22 (0.21) [90.47 (0.52)]	95.23 (0.23) [90.45 (0.45)]	92.41 (0.36) [90.70 (8.09)]	93.28 (0.52) [91.49 (0.93)]	93.63 (0.40) [90.89 (3.05)]	93.39 (0.89) [90.22 (0.89)]

quality of the density ratio rejectors is dependent on the initial model. As such, at higher levels of noise there can be higher variation in the quality of rejection, see Fig. II. Interestingly, for MNIST and CIFAR-10, Figs. III and IV, the base model the density ratio rejectors are more robust across noise rates than other models. This seems to be due to the default MNIST architecture being robust against higher noise rates (notice that the s.t.d. range is also quite small at 100% coverage). In OctMNIST and OrganMNIST, Figs. V and VI, we find that there is little to no change in performance, likely due to the Base classifier's own performance only changing slightly with the addition of noise.

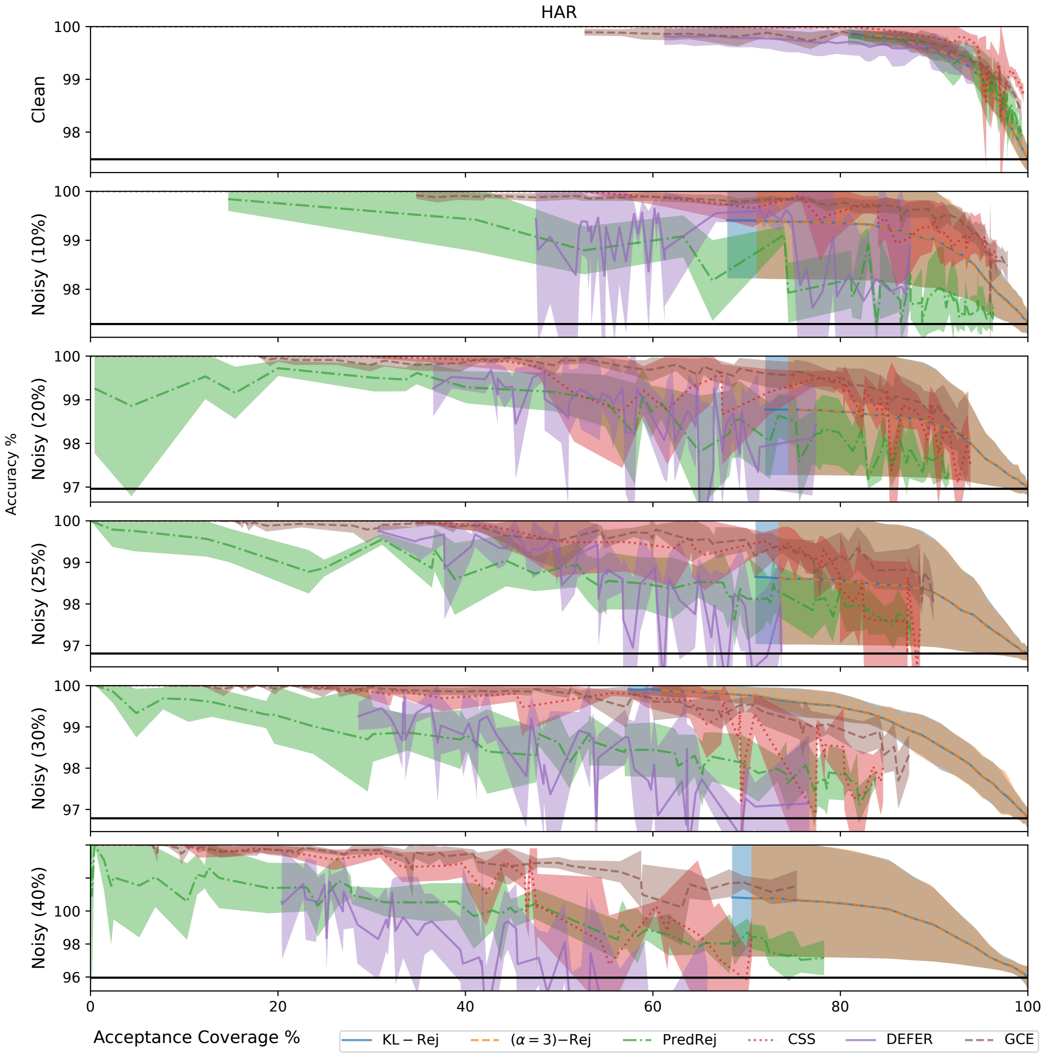

Figure I: Extended plots for HAR of Fig. 2.

Q.III Smaller models case study

In the following, we consider the Gas Drift dataset when models are switched to a base model with only a single hidden layer. First we make note of the original setting explored in the main text. In Table II, we take note of the number of tunable parameters in all approaches and baselines. Notably, these default parameter / architecture sizes are similar to [14], with the HAR and Gas Drift setting including an additional hidden layer than previously utilized in the literature.

In Table III, we note the setting we consider in this subsection. The parameter sizes of the Gas Drift dataset is reduced to the originally explored model sizes in [14]. Notice that both the base models and the baseline approaches have reduced parameter sizes. It should be noted that this smaller parameter

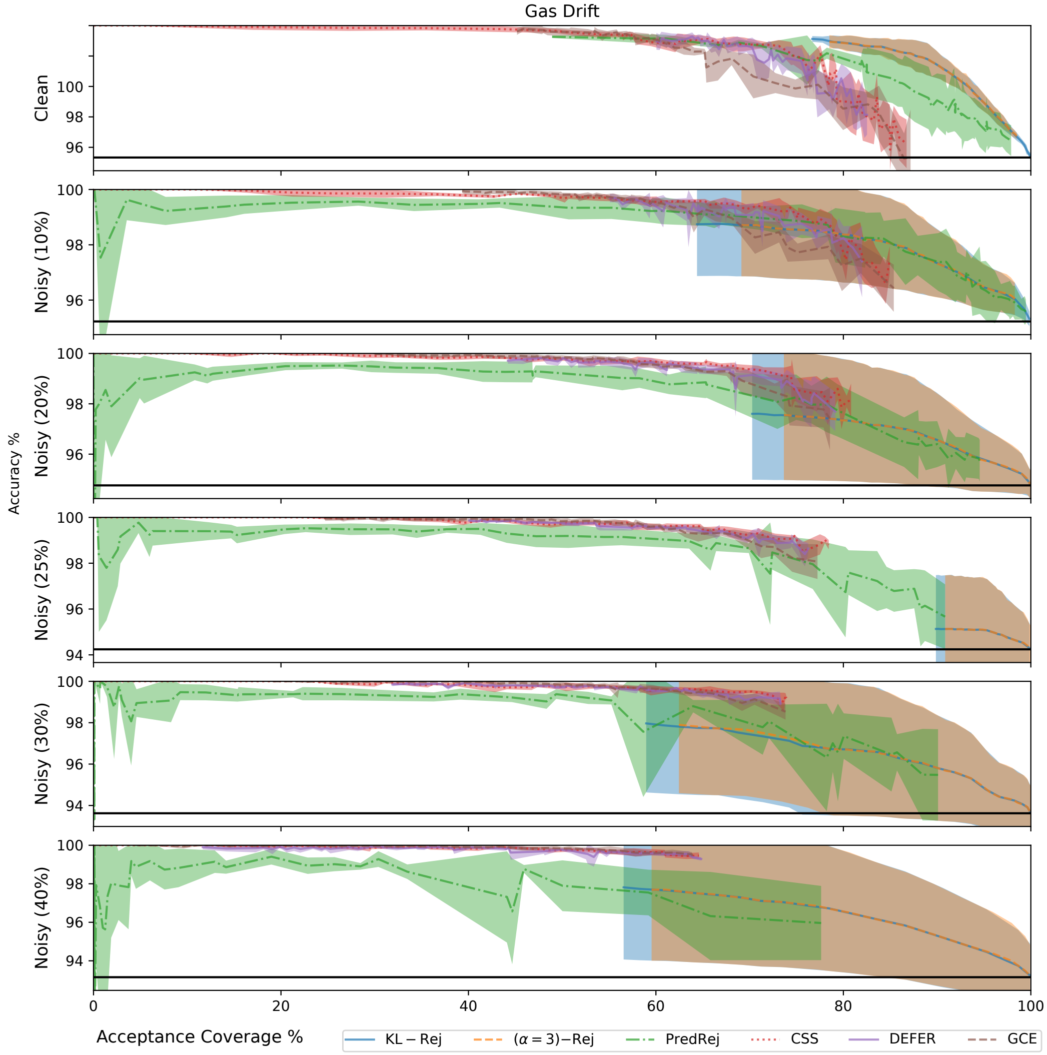

Figure II: Extended plots for Gas Drift of Fig. 2.

Table II: Default parameter sizes of experiments.

Dataset	BaseClf	α−Rej	DEFER	GCE	CSS	PredRej
HAR	40,647	40,648	40,646	40,711	40,711	80,968
Gas Drift	12,935	12,936	12,934	12,999	12,999	25,544
MNIST	1,199,883	1,199,884	1,200,138	1,200,011	1,200,267	2,398,860
CIFAR-10	21,282,123	21,282,124	21,283,146	21,282,635	21,283,659	42,560,652
OctMNIST	11,169,733	11,169,734	11,170,756	11,170,245	11,171,269	22,338,950
OrganMNIST	11,173,324	11,173,325	11,174,347	11,173,836	11,174,860	22,342,541

size setting can be useful in the related learning with deferral setting [52], where having a small model to defer to a larger model is needed.

The results are reported in Figs. VIII and IX. We can see that in this setting, PredRej and our density ratio approaches are more competitive. This might indicate that for simple base models, approaches which 'wrap' a base model for rejection can be quite effective (especially in higher coverage regimes). In general, it seems with this smaller architecture regime, the 'non-wrapping' baseline approaches only provide rejection options when the acceptance coverage is lower than 70%.

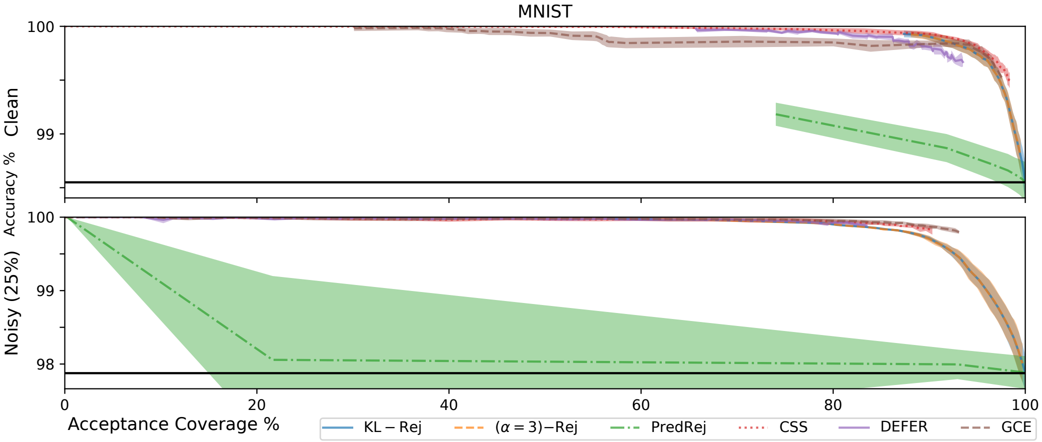

Figure III: Extended plots for MNIST of Fig. 2.

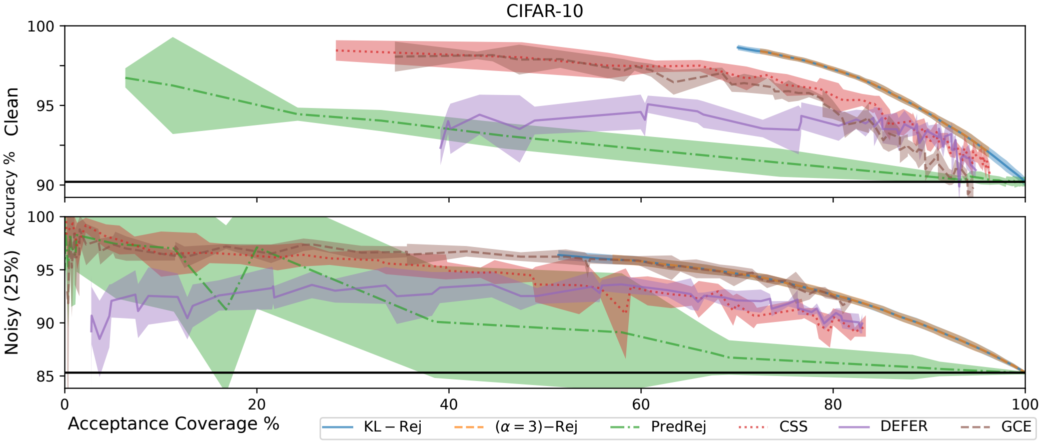

Figure IV: Deferred extended plots for CIFAR-10 of Fig. 2.

Q.IV Parameter sweeps over α and λ

The following shows parameter sweeps over α and λ for our density ratio rejectors. These are given by Figs. X and XI respectively. We find that increasing α compresses the trade-off curve from both sides. While decrease λ extends the trade-off curve on the left side.

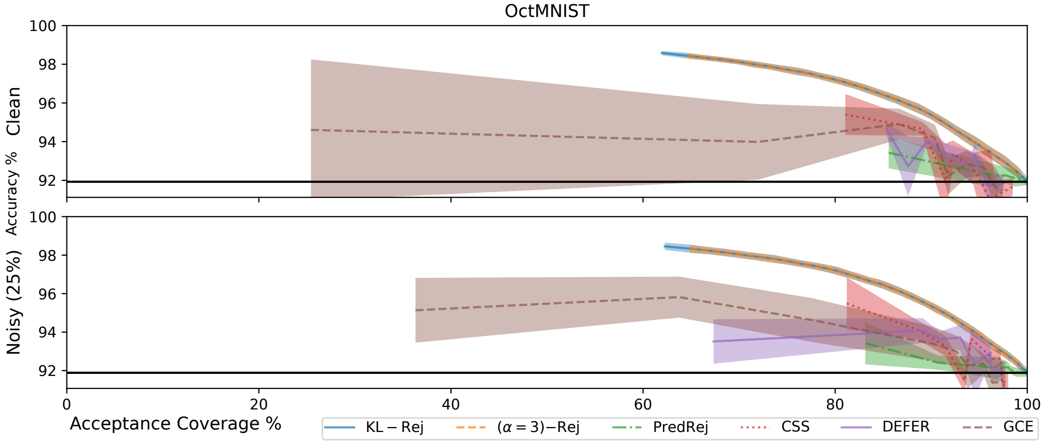

Figure V: Deferred extended plots for OctMNIST of Fig. 2.

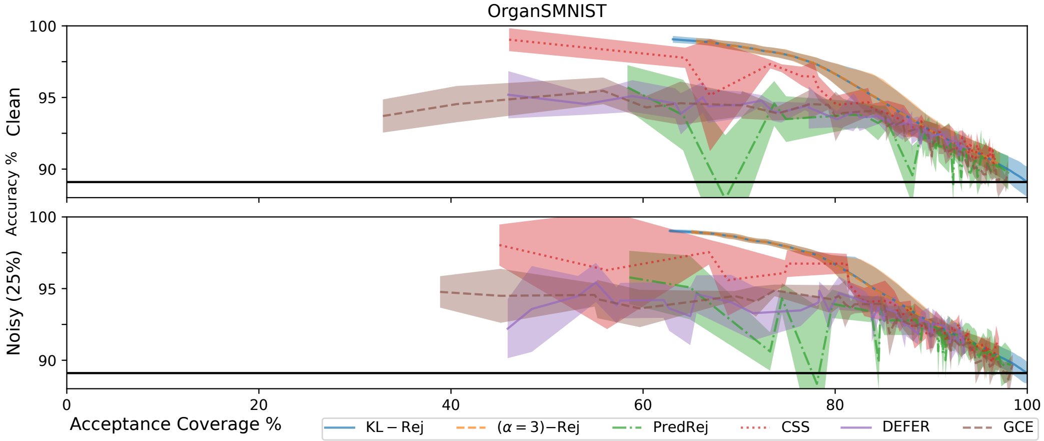

Figure VI: Deferred extended plots for OrganMNIST of Fig. 2.

Table III: Alternative parameter sizes of experiments.

Dataset	BaseClf	$\alpha-$Rej	DEFER	GCE	CSS	PredRej
HAR	36,487	36,488	36,486	36,551	36,551	72,648
Gas Drift	8,775	8,776	8,774	8,839	8,839	17,224

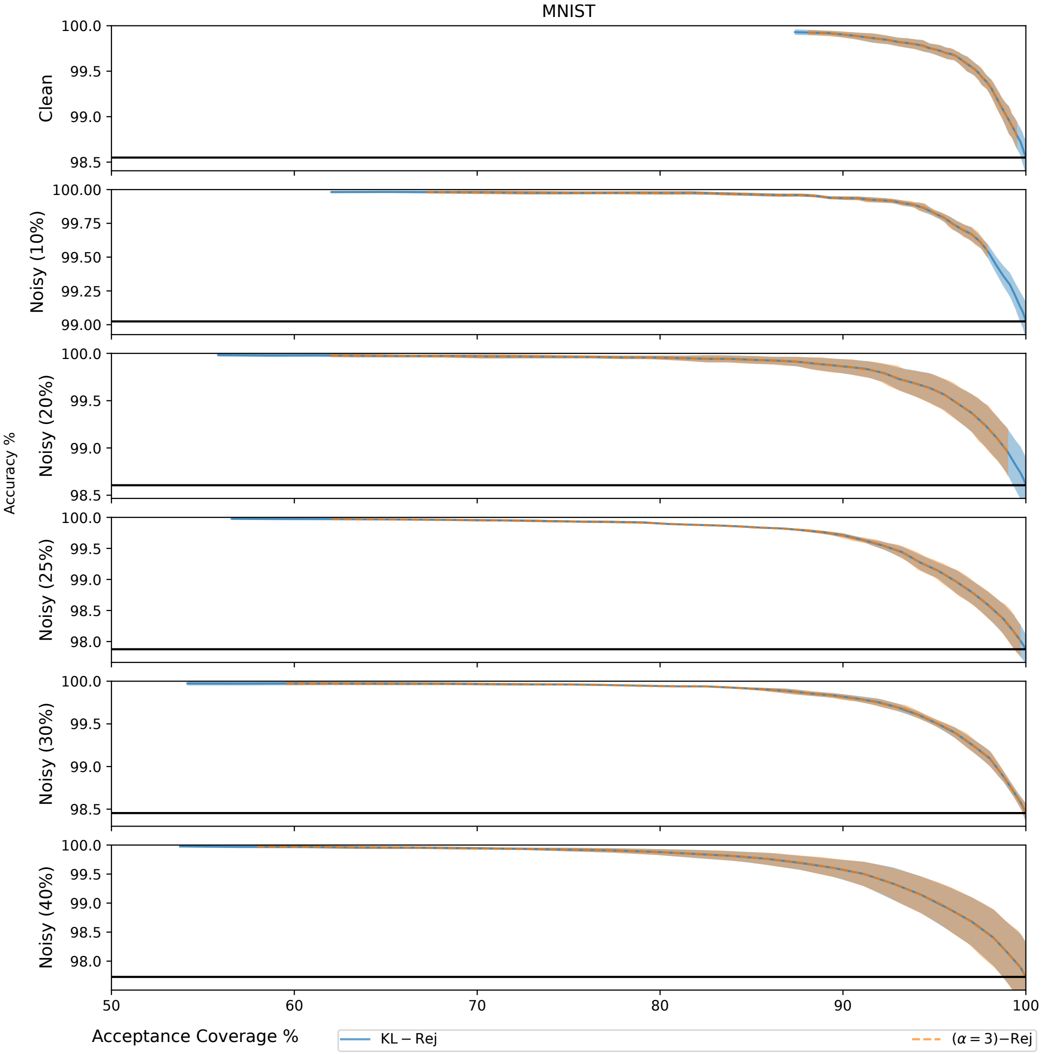

Figure VII: MNIST with different noises for density ratio approaches.

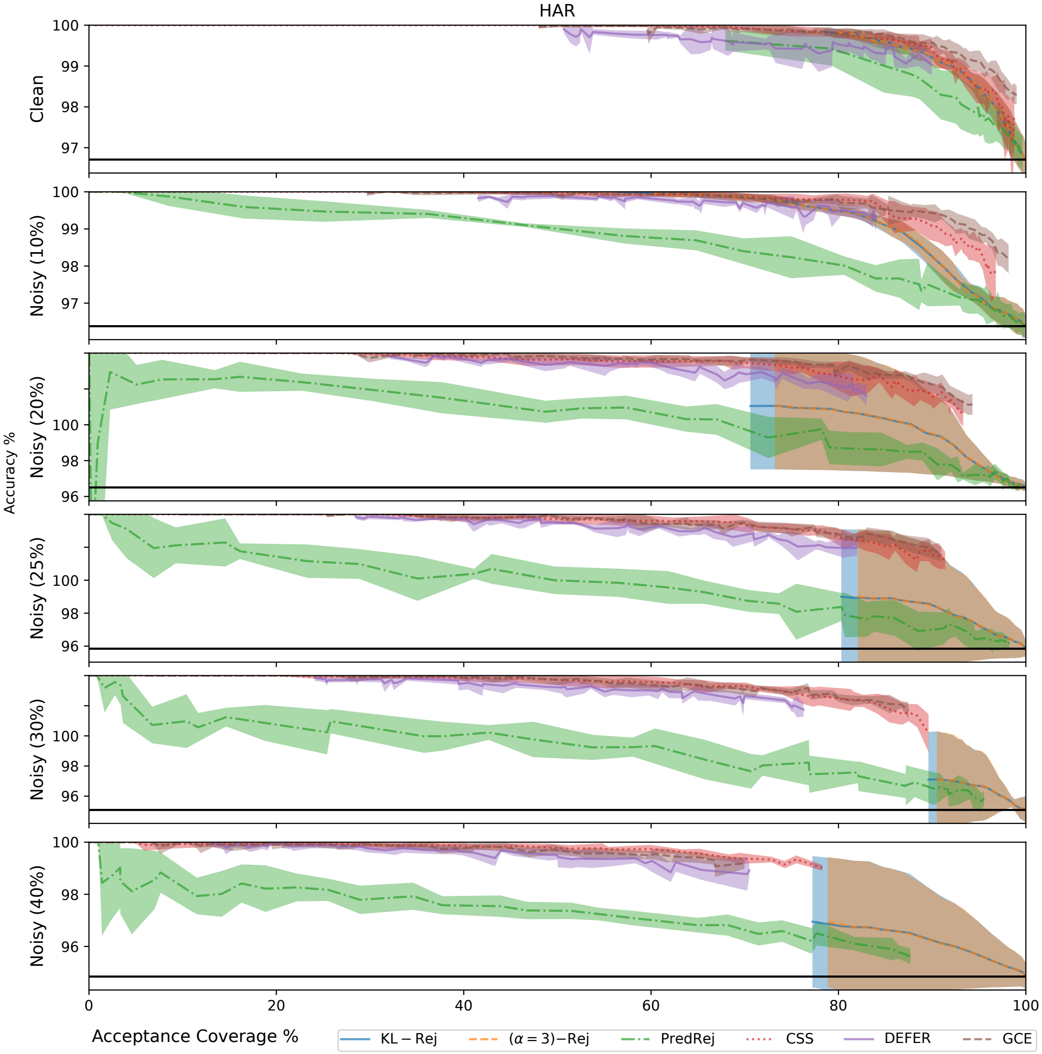

Figure VIII: Plots for HAR with smaller models.

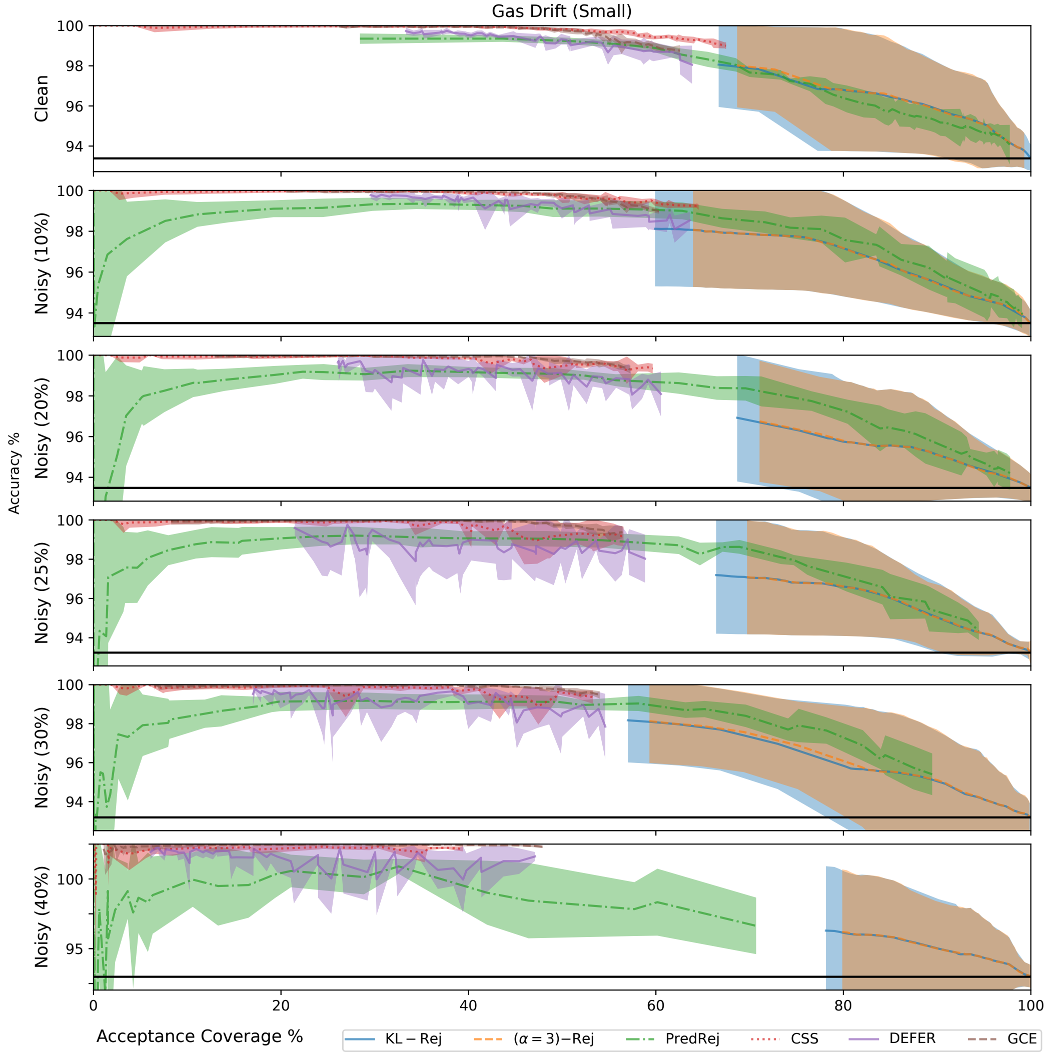

Figure IX: Plots for Gas Drift with smaller models.

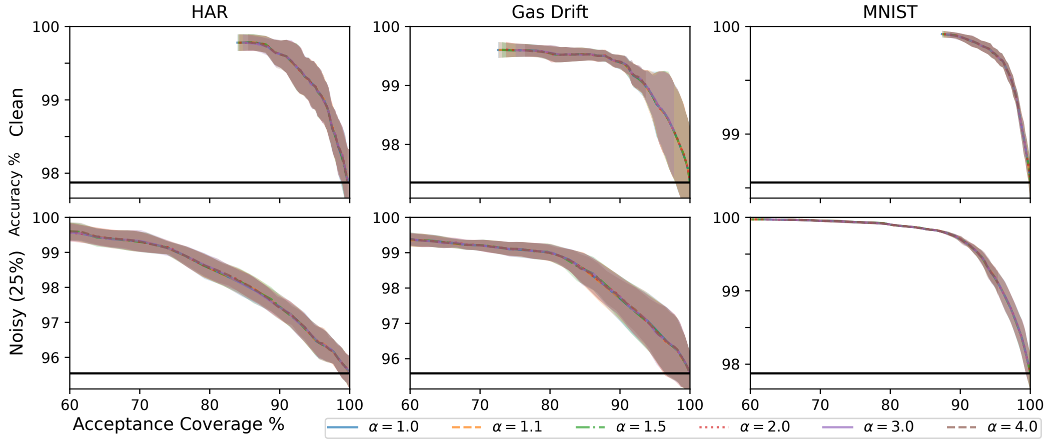

Figure X: α parameter sweep over dataset + noise combinations. S.t.d. shade is not utilized, but instead end points of the trade-off curves for $\tau \in (0, 1]$ are shown via vertical bars

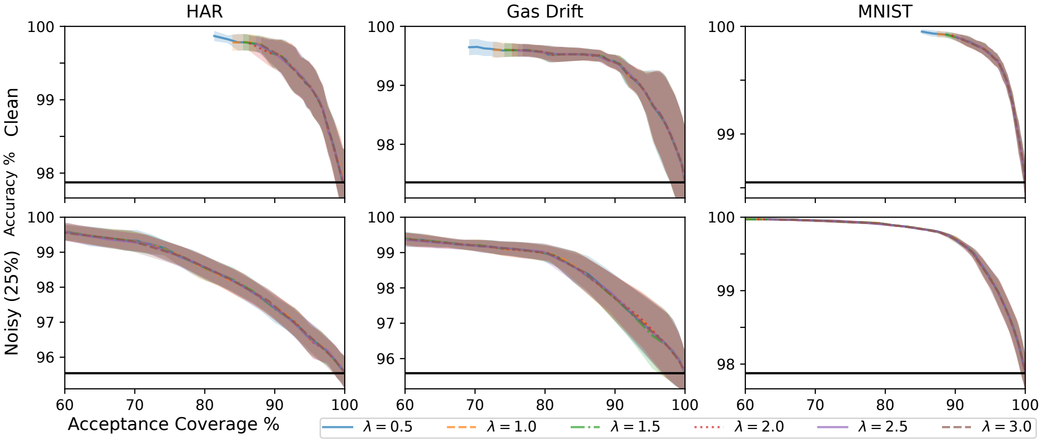

Figure XI: λ parameter sweep over dataset + noise combinations for KL rejector. S.t.d. shade is not utilized, but instead end points of the trade-off curves for $\tau \in (0, 1]$ are shown via vertical bars

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: Claims are justified via formal results and experiments.

   Guidelines:

   - The answer NA 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 NA 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: Limitations are appropriately discussed throughout the paper. Furthermore, the last section of the main-text also explicitly discusses limitations.

   Guidelines:

   - The answer NA 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: Assumptions are stated and complete proofs are provided in Appendix.

Guidelines:

- The answer NA 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: Detailed settings of experiments are provided in Appendix. Code is also included.

   Guidelines:

   - The answer NA 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: Code and scripts to create small scale experiments are provided.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.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://nips.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, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All included in the main-text and also the Appendix.

   Guidelines:

   - The answer NA 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: Error bars are given in plots.

   Guidelines:

   - The answer NA 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: We state the AWS resources utilized and estimated total compute time.

   Guidelines:

   - The answer NA 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: Research conforms with the Code of Ethics.

   Guidelines:

   - The answer NA 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: [Yes]

    Justification: We provide a broader impact section in the Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: No release.

Guidelines:

- The answer NA 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: Baselines and data are correctly credited (with licenses given in the Appendix).

Guidelines:

- The answer NA 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: [NA]

Justification: No new assets.

Guidelines:

- The answer NA 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: [NA]

    Justification: No crowdsourcing and research with human subjects.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: No research with human subjects.

    Guidelines:

    - The answer NA 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.