Selective Classification via One-Sided Prediction

Aditya Gangrade
Boston University
gangrade@bu.edu

Anil Kag
Boston University
anilkag@bu.edu

Venkatesh Saligrama
Boston University
srv@bu.edu

Abstract

We propose a novel method for selective classification (SC), a problem which allows a classifier to abstain from predicting some instances, thus trading off accuracy against coverage (the fraction of instances predicted). In contrast to prior gating or confidence-set based work, our proposed method optimises a collection of class-wise decoupled one-sided empirical risks, and is in essence a method for explicitly finding the largest decision sets for each class that have few false positives. This one-sided prediction (OSP) based relaxation yields an SC scheme that attains near-optimal coverage in the practically relevant high target accuracy regime, and further admits efficient implementation, leading to a flexible and principled method for SC. We theoretically derive generalization bounds for SC and OSP, and empirically we show that our scheme strongly outperforms state of the art methods in coverage at small error levels.

1 Introduction

Selective Classification is a classical problem that goes back to the work of Chow 1957, 1970. The setup allows a learner to classify a query into a class, or to abstain from doing so (we also call this 'rejecting' the query). This abstention models real-world decisions to gather further data/features, or engage experts, all of which may be costly. Such considerations commonly arise in diverse settings, including healthcare[1], security, web search, and the internet of things (Xu et al. 2014; Zhu et al. 2019), all of which require very low error rates (lower even than the Bayes risk of standard classification). The challenge of SC is to attain such low errors while keeping coverage (i.e., the probability of not rejecting a point) high. This is a difficult problem because any choice of what points to reject is intimately coupled with the classifiers chosen for the remaining points.

The most common SC method is via 'gating,' in which rejection is explicitly modelled by a binary-valued function γ, and classification is handled by a function π. An instance, x, is predicted as $\pi(x)$ if $\gamma(x) = 1$, and otherwise rejected. Within this formulation, recent work has proposed a number of methods, ranging from alternating minimisation based joint training, to the design of new surrogate losses, and of new model classes to accommodate rejection. Despite this increased complexity, these methods lack power, as shown by the fact that they do not significantly outperform naïve schemes that rely on abstaining on the basis of post-hoc uncertainty estimates for a trained standard classifier. This represents a significant gap in the practical effectiveness of selective classification.

Our Contributions. We describe a new formulation for the SC problem, that comprises of directly learning *disjoint* classification regions $\{\mathcal{S}_k\}_{k\in\mathcal{Y}}$, each of which corresponds to labelling the instance as k respectively. Rejection is *implicitly defined* as the gap, i.e., the set $\mathcal{R} = \mathcal{X} \setminus \bigcup \mathcal{S}_k$. We show that this formulation is equivalent to earlier approaches, thus retaining expressivity.

The principal benefit of our formulation is that it admits a natural relaxation, via dropping the disjointness constraints, into *decoupled* 'one-sided prediction' (OSP) problems. We show that at design error ε, this relaxation has the coverage optimality gap bounded by ε itself, and so the relaxation is statistically efficient in the practically relevant high target accuracy regime.

We pose OSP as a standard constrained learning problem, and due to the decoupling property, they can be approached by standard techniques. We design a method that efficiently adjusts to inter-class heterogeneity by solving a minimax program, controlled by one

[1] For example, when deciding if a mammary mass is benign or malignant, a general physician may predict based on ultrasound imaging tests, and, in more subtle cases, abstain and refer the patient to a specialist.

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. Copyright 2021 by the author(s).

parameter that limits overall error rates. This yields a powerful SC training method that does not require designing of special losses or model classes, instead allowing use of standard discriminative tools.

To validate these claims, we implement the resulting SC methods on benchmark vision datasets - CIFAR-10, SVHN, and Cats & Dogs. We empirically find that the OSP-based scheme has a consistent advantage over SOTA methods in the regime of low target error. In particular, we show a clear advantage over the naïve scheme described above, which in our opinion is a significant first milestone in the practice of selective classification.

1.1 Related Work

State of the Art (SOTA) methods: The SOTA, in terms of performance, for SC is encapsulated by three methods. The Naïve method, i.e., rejecting when the output of a soft classifier is non-informative (e.g. classifier margin is too small), and this is surprisingly effective when implemented for modern model classes such as DNNs (Geifman and El-Yaniv 2017). The only other methods that can (marginally) beat this are due to Liu et al. 2019, who design a loss function for DNNs, and Geifman and El-Yaniv 2019, who design a new architecture for DNNs that incorporates gating.

Both Liu et al. 2019 and Geifman and El-Yaniv 2019 design methods are based on the **Gating formulation**, mentioned earlier. This formulation was popularised by Cortes et al. 2016, although similar proposals appeared previously (Wiener and El-Yaniv 2011; El-Yaniv and Wiener 2010). A number of papers have since extended this approach, e.g. designing training algorithms via alternating minimisation, (Nan and Saligrama 2017a,b), designing loss functions (Liu et al. 2019; Ni et al. 2019; Ramaswamy et al. 2018), and model classes, such as an architecturally augmented deep neural network (DNN) (Geifman and El-Yaniv 2019). In contrast, our work develops an alternate formulation that directly solves SC without use of specialised losses or model classes.

The naïve method has its roots in the **Direct SC** formulation, which is based on learning a function $f : \mathcal{X} \to \{1, \ldots, K, ?\}$ (where ? denotes rejection), and is pursued by Bartlett and Wegkamp 2008; Herbei and Wegkamp 2006; Wegkamp 2007; Wegkamp and Yuan 2011; Yuan and Wegkamp 2010. The main disadvantage of this formulation is that the methods emerging from it consider very restricted forms of rejection decisions, e.g. $\{|\phi - 1/2| < \delta\}$, where ϕ is a softmax output of a binary classifier.

An alternate **Confidence Set formulation** has been pursued in the statistics literature by Denis and Hebiri 2019; Lei 2014 (for the binary case), and involves learning sets $\{\mathcal{C}_k\}_{k\in[1:K]}$ such that $\bigcup \mathcal{C}_k = \mathcal{X}$, and each $\mathcal{C}_k$ covers class k in the sense $\mathbb{P}(\mathcal{C}_k|Y = k)$ is large[2]. Points which lie in two or more of the $\mathcal{C}_k$s are rejected, and otherwise points are labelled according to which $\mathcal{C}_k$ they lie in. Chzhen et al. 2019; Denis and Hebiri 2017; Sadinle et al. 2019 have subsequently extended this work to the multiclass setting, but they study a 'least ambiguous set-valued classification', which is a different problem from selective classification and does not express it well (see §D). A limitation of existing work in this framework is their reliance on estimating the regression function $\eta(x) := \mathbb{P}(Y = k|X = x)$. Proposals typically go via using non-parametric estimates of η, which are then filtered. On a practical level, this reliance on estimation reduces statistical efficiency, and on a principled level, this violates Vapnik's maxim of avoiding solving a more general problem as an intermediate step to solving a given problem (Vapnik 2000, §1.9).

While our formulation is most closely related to the confidence set formulation, and is equivalent to a change of variables of this (§2.3), it is directly motivated. Furthermore, our framework naturally leads to relaxations to OSP that let us study discriminative methods on high-dimensional datasets and large model classes, which are unexplored in these works.

In passing, we mention the *uncertainty estimation* (UE), and *budget learning* (BL) problems. UE involves estimating model uncertainty at any point (Gal and Ghahramani 2016; Lakshminarayanan et al. 2017), which can plug into both naïve classifiers, and the other methods. As such, UE is a vast generalisation of SC. BL is a restricted form of SC that aims at reaching the accuracy of a complex model using simple functions, and is relevant for efficient inference constraints.

We highlight a recent *decoupling-based* method for BL by Acar et al. 2020 that involves the first and last authors. This work can be seen as considerable extension of this paper to full SC. While the broad strategies of decoupling schemes are similar, significant differences arise since much of the structure developed by Acar et al. 2020 does not generalise to SC, and development of new forms is necessary. Additionally, our experiments study large multiclass models going beyond best achievable standard accuracy, while Acar et al. 2020 only study small binary models getting to standard accuracy achievable by larger models.

[2] More accurately, this precise formulation has not appeared for the multiclass setting, and only appears for the binary problem in work by Denis and Hebiri 2019; Lei 2014. Here we are expressing the natural multiclass extension of this, that turns out to be equivalent to selective classification (§2.3). The existing literature instead pursues the multiclass extension to LASV classification, as mentioned above. Please see §D for a detailed discussion

2 Formulation and Methods

Notation. Probabilities are denoted as $\mathbb{P}$, random variables are capitalised letters, while their realisations are lowercase (X and x). Sets are denoted as calligraphic letters, and classes of sets as formal script ($\mathcal{S} \in \mathscr{S}$). Parameters are denoted as greek letters. For a set $\mathcal{S} \subset \mathcal{X}$, $\mathbb{P}(\mathcal{S})$ is shorthand for $\mathbb{P}(X \in \mathcal{S})$.

We adopt the supervised learning setup - data is distributed according to an unknown joint law $\mathbb{P}$ on $\mathcal{X} \times \mathcal{Y}$, and we observe n i.i.d. points $(X_i, Y_i) \sim \mathbb{P}$. For K classes, we set $\mathcal{Y} = [1:K]$, where K is a constant independent of $|\mathcal{X}|$. We use $\mathscr{S}$ to denote the class of sets from which we learn classifiers.

2.1 Formulation of SC

We set up the SC problem (Fig. 2.3(top) illustrates binary case) as that of directly recovering disjoint classification regions, $\{\mathcal{S}_k\}_{k\in[1:K]}$ from a class of sets $\mathscr{S}$, under the constraint that the error rate is smaller than a given level ε, which we call the target error. Each such K-tuple of sets induces two events of interest - the rejection event, and the error event.

$$\mathcal{R}_{\{\mathcal{S}_k\}} := \left\{X \in \left(\bigcup \mathcal{S}_k\right)^c\right\}$$
$$\mathcal{E}_{\{\mathcal{S}_k\}} := \bigcup \{X \in \mathcal{S}_k, Y \neq k\}.$$

We will usually suppress the dependence of $\mathcal{R}, \mathcal{E}$ on $\{\mathcal{S}_k\}$. Notice further that $\mathcal{E}$ decomposes naturally into events that depend only on one of the $\mathcal{S}_k$s. We will call these 'one-sided' error events

$$\mathcal{E}^k_{\mathcal{S}_k} = \{X \in \mathcal{S}_k, Y \neq k\}.$$

With the above notation, we pose the problem as a maximisation program. The value of this is said to be the *coverage at target error level* ε, denoted $C(\varepsilon; \mathscr{S})$.

$$C(\varepsilon; \mathscr{S}) = \max_{\{\mathcal{S}_k\}_{k\in[1:K]} \in \mathscr{S}} \sum_{k=1}^K \mathbb{P}(\mathcal{S}_k) \qquad \text{(SC)}$$
$$\text{s.t.} \quad \mathbb{P}(\mathcal{E}_{\{\mathcal{S}_k\}}) \leq \varepsilon,$$
$$\mathbb{P}\Big(\bigcup_{k,k'\neq k} \mathcal{S}_k \cap \mathcal{S}_{k'}\Big) = 0,$$

where the final constraint is expressing the fact that the $\mathcal{S}_k$s must be pairwise disjoint. Note that if ε equals the Bayes risk of standard classification with $\mathscr{S}$, then (SC) recovers the standard solution and coverage 1.

Example. Consider the case of $K = 2$ where $\mathbb{P}_X$ is uniform on $[0,1]$, $\mathbb{P}(Y = 1|X = x) = x$, and $\mathscr{S}$ consists of single threshold sets $\{x > t\}, \{x \leq t\}$ for $t \in [0,1]$. The Bayes risk of standard classification is $1/4$. For any $\varepsilon < 1/4$, the coverage at level ε is $C(\varepsilon; \mathscr{S}) = 2\sqrt{\varepsilon}$, which is attained by $\mathcal{S}_1 = \{x > 1 - \sqrt{\varepsilon}\}, \mathcal{S}_2 = \{x \leq \sqrt{\varepsilon}\}$.

2.1.1 Design choices

We outline alternate ways to set up the SC problem that we don't pursue in this paper.

Form of constraints. In (SC), we maximise coverage, while controlling error, which is *error-constrained SC*. Alternately one can pursue *coverage constrained SC* - minimising $\mathbb{P}(\mathcal{E})$ subject to $\mathbb{P}(\mathcal{R}) \leq \varrho$. These are equivalent, and for brevity, we choose the former.

As illustrated in the starting example, our interest in SC is driven by the desire to attain very small error rates. We thus find the error constrained form of SC more natural, and we adopt it in the rest of the paper.[3] We note that our method is also effective for coverage constrained SC, and we show this empirically in §4.

Error criterion. In (SC), we constrain the raw error $\mathbb{P}(\mathcal{E})$. This has the benefit of being both natural, since it directly controls the standard error metric, and further, simple. Alternate forms of the error metric have been studied in the literature - e.g. Geifman and El-Yaniv 2019 condition on acceptance ($P(\mathcal{E}|\mathcal{R}^c)$); Lei 2014 separately constrain class conditionals ($P(\mathcal{E}|Y = k) \leq \varepsilon_k$). Most of the development below can be adapted to these settings with minimal changes, and we restrict attention to $\mathbb{P}(\mathcal{E})$ for concreteness.

2.2 Relaxation and One-sided Prediction

(SC) couples the $\mathcal{S}_k$s via the $\mathbb{P}$-a.s. disjointness constraint. We now develop a decoupling relaxation.

To begin, note that we may decouple the error constraint by introducing variables that trades off the one-sided error rates as below. This program is equivalent to (SC) in the sense that they have the same optimal value, and the same $\{\mathcal{S}_k\}$ achieve this value.

$$\max_{\{\mathcal{S}_k\}\in\mathscr{S}, \{\alpha_k\}\in[0,1]} \sum_{k=1}^K \mathbb{P}(\mathcal{S}_k) \qquad \text{(SC-expanded)}$$
$$\text{s.t.} \quad \forall k : \mathbb{P}(\mathcal{E}^k_{\mathcal{S}_k}) \leq \alpha_k \varepsilon, \quad \sum \alpha_k \leq 1,$$
$$\mathbb{P}\Big(\bigcup_{k,k'\neq k} \mathcal{S}_k \cap \mathcal{S}_{k'}\Big) = 0.$$

Our proposed relaxation is to simply drop the final constraint. The resulting program may be decoupled, via a search over the variables α_k into K *one-sided prediction* (OSP) problems:

$$L_k(\varepsilon_k; \mathscr{S}) = \max_{\mathcal{S}_k \in \mathscr{S}} \mathbb{P}(\mathcal{S}_k) \text{ s.t. } \mathbb{P}(\mathcal{E}^k_{\mathcal{S}_k}) \leq \varepsilon_k \quad \text{(OSP-k)}$$

[3] This is not to imply that the coverage constrained form cannot be more appropriate for some settings. Which one to use in practice is ultimately a problem specific choice.

Notice that the above OSPk problem demands finding the *largest* set $\mathcal{S}_k$ that has a low false alarm probability for the null hypothesis $Y = k$. Structurally this is the opposite to the more common Anomaly Detection problem, which demands finding the smallest set with a low missed detection probability.

We note that while we decouple the SC problem completely above, the main benefit is the removal of the intersection constraint, which is the principal difficulty in SC. The sum error constraint is benign, and for reasons of efficiency we will reintroduce it in §3.

Continuing, observe that the sets recovered from the above problems may overlap, which introduces an ambiguous region. This overlap region is necessarily of small mass (Prop. 1), and so may be dealt with in any convenient way. Theoretically we break ambiguities in the favour of the smallest label. These sets need not belong to $\mathscr{S}$ anymore, and so this is an (weakly) improper classification scheme.

Overall this gives the following infinite sample scheme:

- For each feasible $\alpha \in [0,1]^K$, solve for $\{L_k(\alpha_k \varepsilon)\}$ for each $k \in [1:K]$. Let $\{\mathcal{T}_k^\alpha\}$ be the recovered sets.
- Let $\mathcal{S}_k^\alpha = \mathcal{T}_k^\alpha \setminus \left(\bigcup_{k'<k} \mathcal{T}_{k'}^\alpha\right)$.
- Return the $\{\mathcal{S}_k^\alpha\}$ that maximises $\sum_k \mathbb{P}(\mathcal{S}_k^\alpha)$ over α.

At small target error levels, which is our intended regime of study, the resulting sets are guaranteed to not be too lossy, as in the following statement. The above is shown (in §A.1) by arguing that the mass of the overlap between the OSP solutions (the $\mathcal{T}_k$) is at most 2ε. Empirically this is even lower, see Table 4.

Proposition 1. *If $\{\mathcal{S}_k\}$ are the sets recovered by the procedure above, then these are feasible for (SC). Further, their optimality gap is at most 2ε, i.e.*

$$\sum_{k\in[1:K]} \mathbb{P}(\mathcal{S}_k) \geq C(\varepsilon; \mathscr{S}) - 2\varepsilon.$$

2.3 Equivalence of SC formulations

We show that the prior gating and confidence frameworks are equivalent to ours, based on transforming feasible solutions of one framework into an other.

Gating: Denote the acceptance set of gating as $\Gamma = \{\gamma = 1\}$, and let the predictions be $\Pi_k = \{\pi = k\}$. Taking $\mathcal{S}_k = \Pi_k \cap \Gamma$ yields disjoint sets that can serve for SC under our formulation that have the same decision regions for each class, and the same rejection region, since $(\bigcup \mathcal{S}_k)^c = \Gamma^c$. Conversely, for disjoint decision sets $\mathcal{S}_k$, the gate $\Gamma = \bigcup \mathcal{S}_k$, and the predictor $\Pi_k = \mathcal{S}_k$ form the corresponding gating solution.

Confidence set: Take confidence sets $\{\mathcal{C}_k\}$ which cover $\mathcal{X}$, and have the rejection set $\mathcal{B} = \bigcup_{k\neq k'} \mathcal{C}_k \cap \mathcal{C}_{k'}$. Then

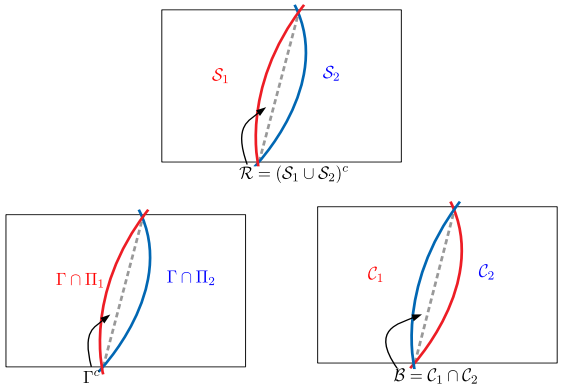

Figure 1: An illustration of the equivalence between the three formulations for binary classification. *Top:* our formulation; $\mathcal{S}_i$ denotes disjoint sets; *Bottom Left:* gating with Γ representing gated set; *Bottom Right:* $\mathcal{C}_i$ represents confidence sets, and their intersection representing the rejected set. In each case, the dashed line is the Bayes boundary.

we produce the disjoint sets $\mathcal{S}_k = \mathcal{C}_k \setminus (\bigcup_{k'\neq k} \mathcal{C}_{k'})$, which retain the same decision regions. These also have the same rejection region because we may express $\mathcal{S}_k = \mathcal{C}_k \cap \mathcal{B}^c$, and thus $\bigcap \mathcal{S}_k^c = (\bigcap_k \mathcal{C}_k^c) \cup \mathcal{B}$, and $\bigcap \mathcal{C}_k^c = \varnothing$ since the $\mathcal{C}_k$ cover the space. Conversely, for disjoint $\{\mathcal{S}_k\}$, the sets $\mathcal{C}_k = (\bigcup_{k'\neq k} \mathcal{S}_{k'})^c = \mathcal{S}_k \cup \mathcal{R}$ cover the space, and have the rejection region $\mathcal{R}$ since $\mathcal{C}_k \cap \mathcal{C}_{k'} = \mathcal{R}$ for any pair $k \neq k'$.

Figure 2.3 illustrates these equivalences. Notice that due to the simplicity of the reductions, these equivalences are fine-grained in that the joint complexity of the family of sets used is preserved in going from one to the other. Given these equivalences, we again distinguish our approach from the existing ones.

First, the structure of solutions is markedly different. The gating formulation takes both the rejection and the classification decisions explicitly via the two different sets. The confidence set formulation takes neither explicitly, and instead produces a 'list decoding' type solution. In contrast, we make the classification decisions explicit, and produce the rejection decision implicitly.

Consequently, the salient differences lies in the method. The gating based methods have concentrated on the design of surrogate losses and models, while for the confidence set, methods either go through estimating the regression function, or via a reduction to anomaly detection type problems (Denis and Hebiri 2017; Sadinle et al. 2019).[4]. In strong contrast, we develop a new relaxation that allows decoupled learning via 'one-sided prediction' problems. These OSP problems are almost opposite to anomaly detection - instead of finding small sets for each class that do not leave too much of its mass missing, we instead learn large sets that do not admit too much of the complementary class' mass.

[4] We refer to §D.4 for a deeper discussion of this point

2.4 Finite Sample Properties of OSP

Thus far we have spoken of the full information setting. This section gives basic generalisation analyses for an empirical risk minimisation (ERM) based finite sample approach. Since the one-sided problems are entirely symmetric, we concentrate only on OSP-1, that is (OSP-k) with $k = 1$, below. Note that the SC problem can directly be analysed in a similar way (Appx. A.2), but we focus on the OSP problem, since this underlies the method we pursue.

We show asymptotic feasibility of solutions, that is, we show that we can, with high probability, recover a set $\mathcal{S}$ for OSP such that $\mathbb{P}(\mathcal{S}) \geq L_1(\varepsilon) - o(1)$ and $\mathbb{P}(\mathcal{E}^1_{\mathcal{S}}) \leq \varepsilon + o(1)$, where the o are as the sample size diverges. This is in contrast to exact feasibility, i.e., insisting on $\mathcal{S}'$ such that $\mathbb{P}(\mathcal{E}^1_{\mathcal{S}'}) \leq \varepsilon$ with high probability. Exactly satisfying constraints via ERM whilst maintaining that the objective is also approaching the optimum is a subtle problem, and was shown to be impossible in certain cases by Rigollet and Tong 2011. On the other hand, plug-in methods along with an 'identifiability' condition which imposes that the law of $\eta(X)$ is not varying too fast at any point can be employed to give exact constraint satisfaction along with a small excess risk - the technique was developed by Tong 2013, and has been used in SC contexts by, e.g., Shekhar et al. 2019. However, since the applicability of plug-in methods to large datasets in high dimensions is limited, we do not pursue this avenue here.

One-Sided Learnability

Definition *We say that a class $\mathscr{S}$ is one-sided learnable if for every $\varepsilon \geq 0$ and $(\delta, \sigma, \nu) \in (0,1)^3$, there exists a finite $m(\delta, \sigma, \nu)$ and an algorithm $\mathfrak{A} : (\mathcal{X} \times [1 : K])^m \to \mathscr{S}$ such that for any law $\mathbb{P}$, given m i.i.d. samples from $\mathbb{P}$, $\mathfrak{A}$ produces a set $\mathcal{S}_1 \in \mathscr{S}$ such that with probability at least $1 - \delta$ over the data,*

$$\mathbb{P}(\mathcal{S}_1) \geq L_1(\varepsilon; \mathscr{S}) - \sigma, \qquad \text{and} \qquad \mathbb{P}(\mathcal{E}^1_{\mathcal{S}_1}) \leq \varepsilon + \nu.$$

The characterisation we offer is

Proposition 2. *A class $\mathscr{S}$ is one-sided learnable iff it has finite VC dimension. In particular, given n samples, we can obtain a set $\mathcal{S}_1$ that, with probability at least $1 - \delta$, satisfies*

$$\mathbb{P}(\mathcal{S}_1) \geq L_1(\varepsilon; \mathscr{S}) - \sqrt{C_K \frac{(\mathrm{VC}(\mathscr{S}) \log n + \log(C_K/\delta))}{n}}$$

$$\mathbb{P}(\mathcal{E}^1_{\mathcal{S}_1}) \leq \quad \varepsilon \quad + \sqrt{C_K \frac{(\mathrm{VC}(\mathscr{S}) \log n + \log(C_K/\delta))}{n}},$$

where C_K is a constant that depends only on the number of classes K.

The proof of the necessity of finite VC dimension is via a reduction to standard learning, while the upper bounds on rates above follow from uniform convergence due finite VC dimension. See §A.3. The scheme attaining these is a direct ERM that replaces all $\mathbb{P}$s in (OSP-k) by empirical distributions.

On the whole, applying the above result for each of the K OSP problems tells us that if we can solve the empirical OSP problems for the indicator losses and constraints, then we can recover a SC scheme that, with high probability, incurs error of at most $\varepsilon + O(1/\sqrt{n})$ and has coverage of at least $C(\varepsilon; \mathscr{S}) - 2\varepsilon - O(1/\sqrt{n})$.

3 Method

In this section, we derive an efficient scheme, first by replacing indicator losses with two differentiable surrogate variants, and then propose OSP relaxations. A summary of the method expressed as pseudo-code is included in Appx. B. Throughout, $\mathscr{S}$ is set to be level sets of the soft output of a deep neural network (DNN), i.e., $\mathscr{S} = \{f(\cdot;\theta) > t\}$, where $f(\cdot;\theta) : \mathcal{X} \to [0,1]$ is a DNN parametrised by θ. The bulk of the exposition concerns learning θs. In this and the following section, $\{(x_i, y_i)\}_{i=1}^n$ refers to a training dataset with n labelled data points.

Relaxed losses. To solve the OSP problem, we follow the standard approach of replacing indicator losses by differentiable ones. This sets up the relaxed problem

$$\min_{\theta_k} \frac{\sum_i \ell(f(x_i;\theta_k))}{n} \text{ s.t.} \frac{\sum_{i:y_i \neq k} \ell'(f(x_i;\theta_k))}{n_{\neq k}} \leq \varphi_k$$

where θ_k parametrises the DNN, φ_k denote relaxed values of the constraints, and ℓ, ℓ' are surrogate losses that are small for large values of their argument, and $n_{\neq k} = |\{i : y_i \neq k\}|$. In the experiments we use $\ell(z) = -\log(z)$ and $\ell'(z) = -\log(1-z)$, essentially giving a weighted cross entropy loss. We refer to the objective of the above problem as $\widetilde{L}_k(\theta_k)$, and the constraint as $\widetilde{C}_k(\theta_k)$.

A more stable loss. Practically, the loss $\widetilde{L}_k$ suffers from instability due to the fact that the first term sums over all instances. This can seen clearly when $\ell = -\log$, for which the objective includes the sum $\sum_{i:y_i \neq k} -\log(f(x_i;\theta_k))$. Since for negative examples we expect $f(x;\theta_k)$ to be small, this sum is very sensitive to perturbations in these values, which reduces the quality of the solutions. To ameliorate this, we formulate the following 'restricted' loss, where the objective instead sums over only the positively labelled samples

$$\min_{\theta_k} \frac{\sum_{i:y_i=k} \ell(f(x_i;\theta_k))}{n_k} \text{ s.t. } \widetilde{C}_k(\theta_k) \leq \varphi_k. \qquad (1)$$

Notice that the constraint $\widetilde{C}_k$ is the same as before. We refer to the restricted objective above as $\widetilde{L}_k^{\text{res.}}(\theta_k)$. This loss underlies all further methods, and §4.

Note that the above program remains sound w.r.t. the OSP task, since it is a surrogate for the following

$$\max_{\mathcal{S}_k \in \mathscr{S}} \mathbb{P}(X \in \mathcal{S}_k, Y = k) \text{ s.t. } \mathbb{P}(X \in \mathcal{S}_k, Y \neq k) \leq \varepsilon.$$

Comparing (OSP-k) and the above, the constraints are the same, and the objectives differ by $\mathbb{P}(\mathcal{S}_k) - \mathbb{P}(\mathcal{S}_k, Y = k) = \mathbb{P}(\mathcal{S}_k, Y \neq k)$, which, due to the constraint, is at most ε. Thus, the programs are equivalent up to a small gap (that is, optimal solutions for the above attain a value for (OSP-k) that is ε-close to the optimal value for it). For the same reason, we can use the solutions of the above one-sided problem in the scheme of §2.2 to yield solutions feasible for (SC) that satisfy an analogue of Prop. 1 with an optimality gap of 3ε instead of 2ε.

Joint Optimisation and normalisation. A naïve approach with the above relaxations in hand is to optimise the k OSP problems separately. However, this leads to an exponential in K rise in complexity in the model selection process, since different values of $(\varphi_1, \ldots, \varphi_K)$ need to be selected - if Φ such values are searched over for each φ_k, then this amounts to a prohibitive grid search over Φ^K values. In addition, due to class-wise heterogeneity, the values of φ_ks need not be calibrated across programs, and thus simple solutions like pinning all the φ_ks to the same value are not viable. A final issue is that a naïve implementation of this setup results in training K separate DNNs, which leads to a K-fold increase in model complexity.

We make two modifications to handle this situation. First, we normalise function outputs by adopting the following architecture: we consider DNNs with K output nodes, each representing one of the f_k. The backbone layers of the network are shared across all OSP problems. Further, we take

$$f(x) = (f_1(x), \ldots, f_K(x)) = \text{softmax}(\langle w_k, \xi_\theta(x) \rangle),$$

where ξ_θ denotes the backbone's output, and recall that $(\text{softmax}(v))_k = \exp(v_k)/\sum \exp(v_k)$. This normalisation and restricted model handles both the class-wise heterogeneity, and the blowup in model complexity.

For the sake of succinctness, we define $\mathbf{w} = (w_1, w_2, \ldots, w_K)$, and $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_K)$.

Next, in order to ameliorate the search, we propose jointly optimising the various OSP problems, by enforcing a joint constraint on the sum of the various constraint values via a single value φ. This mimics the structure of (SC), where the constraint limits the sum of the one-sided errors. The relaxation thus amounts to dropping the disjointness constraint, and softening the indicators in (SC). The resulting problem is

$$\min_{\theta, \mathbf{w}, \boldsymbol{\varphi}} \sum_k \widetilde{L}_k^{\text{res.}}(\theta, \mathbf{w}) \tag{2}$$
$$\text{s.t.} \quad \forall k : \widetilde{C}_k(\theta, \mathbf{w}) \leq \varphi_k, \quad \sum \varphi_k \leq \varphi,$$

where recall $\widetilde{L}^{\text{res.}}, \widetilde{C}_k$ from above, which are functions of $(\theta, \mathbf{w})$ since the backbone θ is shared, and since all f_k depend on all w_ks due to the softmax normalisation.

Finally, we propose optimising (2) via stochastic gradient ascent-descent. We note that one tunable parameter - μ - remains in the problem, corresponding to the sum constraint on the φ_ks, while λ_ks are multipliers for the $\widetilde{C}_k$ constraints. We again denote $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)$. The resulting Lagrangian is

$$\widetilde{M}^{\text{res.}}(\theta, \mathbf{w}, \boldsymbol{\varphi}, \boldsymbol{\lambda}, \mu) \tag{3}$$
$$= \sum_k \widetilde{L}_k^{\text{res.}}(\theta, \mathbf{w}) + \lambda_k(\widetilde{C}_k(\theta, \mathbf{w}) - \varphi_k) + \mu\varphi_k,$$

and we solve the problem

$$\min_{(\theta, \mathbf{w}, \boldsymbol{\varphi})} \max_{\boldsymbol{\lambda} : \forall k, \lambda_k \geq 0} \widetilde{M}^{\text{res.}}(\theta, \mathbf{w}, \boldsymbol{\varphi}, \boldsymbol{\lambda}, \mu), \tag{4}$$

treating μ as the single tunable parameter.

We note that the Lagrangian above bears strong resemblance to a one-versus-all (OVA) multiclass classification objective. The principal difference arises from the fact that the losses are weighted by the λ_k terms, and the optimisation trades these off, which are typically not seen in one-versus-all approaches (of course, we also use the resulting functions very differently).

Thresholding and resulting SC solution. The outputs of the classifiers learned with any given μ yield soft signals for the various OSP problems. To harden these into a decision, we threshold the outputs of the soft classifier at a common level $t \in [0, 1]$. This crucially relies on the earlier normalisation of the soft scores to make them comparable. Finally, to deal with ambiguous regions, we use the soft signals f_k, and assign the label to the one with the largest score. Overall, this leads to the SC solution

$$\mathcal{S}_k(\theta, \mathbf{w}, t) = \{x : f_k(x; \theta, \mathbf{w}) \geq t\} \tag{5}$$
$$\cap \ \{x : k = \arg\max_{k'} f_{k'}(x; \theta, \mathbf{w})\}.$$

Model Selection. The above setup has two scalar hyperparameters - μ from (4), and threshold t at which hard decisions are produced in (5), and each choice of these yields a different solution. Our final model is one that performs the best on the validation dataset among all hyperparameter tuples (μ, t). Concretely, let $\widehat{\mathbb{P}}_V$ denote the empirical law on a validation dataset. Denote the solutions from (4) with a choice of μ as $(\theta(\mu), \mathbf{w}(\mu))$. Let $\mathbf{M}, \mathbf{T}$ respectively be discrete sets of μ's and t's. The procedure is

- For each $(\mu, t) \in \mathbf{M} \times \mathbf{T}$, and each k, compute $\mathcal{S}_k(\mu, t) = \mathcal{S}_k(\theta(\mu), \mathbf{w}(\mu), t)$ as defined in (5).
- For each $(\mu, t) \in \mathbf{M} \times \mathbf{T}$, evaluate $\widehat{C}_V(\mu, t) = \sum_k \widehat{\mathbb{P}}_V(\mathcal{S}_k(\mu, t))$ and $\widehat{E}_V(\mu, t) = \sum_k \widehat{\mathbb{P}}_V(\mathcal{E}^k_{\mathcal{S}_k})$.

- Let $(\mu^*, t^*) = \arg\max_{\mathbf{M}\times\mathbf{T}} \widehat{C}_V(\mu, t)$ subject to $\widehat{E}_V(\mu, t) \leq \varepsilon$. Return $(\theta(\mu^*), \{w_k(\mu^*)\}, t^*)$.

4 Experiments

4.1 Experimental Setup and Baselines

Dataset	Num. of Samples Train.	Test	Val.	Std. Error
CIFAR-10	45K	10K	5K	9.58%
SVHN-10	65.9K	26K	7.3K	3.86%
Cats & Dogs	18K	5K	2K	5.72%

Table 1: Dataset sizes and standard classification error

Datasets and Model Class We evaluate all methods on three benchmark vision tasks: CIFAR-10 (Krizhevsky and Hinton 2009), SVHN-10 (Netzer et al. 2011) (10 classes), and Cats & Dogs[5] (binary). All models implemented below are DNNs with the RESNET-32 architecture (He et al. 2016), which is a standard model class in vision tasks. 20% of the training data is reserved for validation in each dataset. All models are implemented in the tensorflow framework. The samples sizes and the best standard classification performance is presented in Table 1.

Baselines: We benchmark against three state of the art methods. The 'selective net' and 'deep gamblers' methods also require hyperparameter and threshold tuning as in our setup, and we do this in a brute force way on validation data, as in ours.

Softmax Response Thresholding (SR) involves training a neural network for standard classification, and then thresholding its soft output to decide to reject. More formally, the decision is to reject if $\{\text{softmax}(f_1, \ldots, f_K) < t\}$, where f is the soft output, and t is tuned on validation data. This simple scheme is known to have near-SOTA performance (Geifman and El-Yaniv 2017, 2019).

Selective Net (SN) is a DNN meta-architecture for SC due to Geifman and El-Yaniv 2019. The network provides three soft outputs - (f, γ, π), where f is an auxiliary classifier used to aid featurisation during training, and γ, π is a gate-predictor pair. Selective net prescribes a loss function that trades off coverage and error via a multiplier c, and by fine-tuning a threshold on γ to reject. We use the publicly available code[6] to implement this, and a comprehensive sweep over the coverage and threshold hyper-parameters. We use 40 valued grid for the parameter c (with 10 equally spaced values in the range $[0.0, 0.65)$ and remaining 30 values in the range $[0.65, 1.0]$). For the gating threshold γ, we use 100 thresholds equally spaced in the range $[0, 1]$, the same as for our scheme.

Deep Gamblers (DG) is a loss function for SC within the gating framework due to Liu et al. 2019. The NNs have $K+1$ outputs - $f_1, \ldots, f_K, f_?$. The cross-entropy loss is modified to $\sum \log\left((f_{y_i}(x_i) + \varphi^{-1} f_?(x_i)\right)$, where $\varphi \in [1, K)$ is a hyperparameter that trades-off coverage and accuracy. Hard decisions are obtained by tuning the threshold of $f_?$ on a validation set. We adapt the public torch code[7] for this method to the Tensorflow framework. We used 40 values of φ spaced equally in the range $[1, 2)$[8], and 100 values of thresholds in $[0, 1]$.

4.2 Training One-Sided Classifiers

Loss Function We use the loss function $\widetilde{M}^{\text{res.}}$ developed in §3. In particular for $\widetilde{L}_k^{\text{res.}}$, we use $\ell(z) = -\log(z)$, and for $\widetilde{C}_k, \ell'(z) = -\log(1-z)$.

Training of Backbones As previously discussed, our models share a common backbone and have a separate output node for each OSC problem. We intialise this backbone with a base network trained using the cross-entropy loss (i.e. a 'warm start'). Note that this typically yields a strong featurisation for the data, and exploiting this structure requires us to not move too far away from the same. At the same time, due to the changed objective, it is necessary to at least adapt the final layer significantly. We attain this via a two-timescale procedure: the loss is set to the OSP Lagrangian, and the backbone is trained at a *slower rate* than the last layer. Concretely, the last layer is updated at every epoch, while the backbone is updated every 20 epochs. This stabilises the backbone, while still adapting it to the particular OSP problem that the network is now trying to solve.

Hyper-parameters. All of the methods were trained using the train split and the model selection was performed on the validation set. The results are reported on the separate test data (which is standard for all three of the models considered). The minimax program on the Lagrangian was optimised using a two-timescale

[5] https://www.kaggle.com/c/dogs-vs-cats

[6] https://github.com/geifmany/selectivenet

[7] https://github.com/Z-T-WANG/NIPS2019DeepGamblers/

[8] We initially made a mistake and scanned φ in $[1, 2)$ instead of $[1, 10)$. We then redid the experiment. with 40 values in $[1, 10)$, and found that performance deteriorated. This is because the optimal φ for these datasets lies in $[1, 2)$, and the wider grid leads to a less refined search in this domain. Thus, values from the original experiment are reported. See Tables 6, 7 in §C for the values with a scan over $[1, 10)$.

stochastic gradient descent-ascent, following the recent literature on nonconvex-concave minimax problems (Lin et al. 2019). In particular, we used Adam optimizer for training with initial learning rates of $(10^{-3}, 10^{-5})$ for the min and the max problems respectively for CIFAR-10 and SVHN-10, and of $(10^{-3}, 10^{-4})$ for Cats & Dogs.[9] These initial rates were reduced by a factor of 10 after 50 epochs, and training was run for 200 epochs. The batch size was set to 128.

We searched over 30 values of μ for each of our experiments - 10 values equally spaced in $[0.01, 1]$, and remaining 20 equally spaced in $[1, 16]$. We further used 100 values of thresholds equally spaced in $[0, 1]$.

4.3 Results

The key takeaway of our empirical results is the significant increase in performance of our SC scheme when compared to the baselines. We also include some observations about the structure of the solutions obtained.

4.3.1 Performance

Performance at Low Target Error is presented in Table 2, which reports coverage at three (small) targeted values of error - $1/2, 1$, and 2 percent - that are in line with the low target error regime that is the main focus of the paper. Notice that these target error values are far below the best error obtained for standard classification (Table 1). We observe that the performance of our SC methods is significantly higher than the SOTA methods, especially in the case of CIFAR-10 and Cats & Dogs, where we gain over 4% in coverage at the 0.5% design error. The effect is weaker in SVHN, which we suspect is due to saturation of performance in this simpler dataset.

Performance at High Target Coverage is presented in Table 3. This refers to the coverage constrained SC formulation discussed in §2.1.1. For these experiments, we use the same μ values (to avoid retraining), but choose thresholds such that the coverage of the resulting model exceeds the stated target, and the models with the lowest error at this threshold are chosen. We observe that at target coverage 100%, the SR solution outperfoms all others. This is expected, since 100% coverage corresponds to standard classification, and the SR objective is tuned to this, while the others are not. Surprisingly, for coverage below 100%, our OSP-based relaxations deliver stronger performance than the benchmarks. Note that this is not due to the low target error performance, because (besides SVHN), the errors attained at these coverage are signficantly above the low target errors investigated in Table 2. This shows that our formulation is also effective in the high-coverage regime.

Coverage-Error Curves for the CIFAR-10 dataset are shown in Fig. 2. These curves plot the best coverage obtained by training at a given target error level using each of the methods discussed.[10] We find that the coverage obtained by our method uniformly outperform DG and SN, and also outperform SR for the bulk of target errors, except those very close to the best standard error attainable. This illustrates that our scheme is effective across target error levels. We find this rather surprising since we designed our method with explicit focus on the low target error regime. Tables 2 and 3 can be seen as detailed looks at the left (error < 2) and the upper (coverage > 90) ends of these curves.

Observations regarding baselines. Tables 2,3, and Figure 2 all show that across regimes, DG and SN perform similarly to SR, and are frequently beaten by it. This observation is essentially consistent with the results presented in (Geifman and El-Yaniv 2019; Liu et al. 2019), and supports our earlier claims that the prior SOTA methods for selective classification do not meaningfully improve on naïve methods. To alleviate concerns about implementation, we emphasise that we performed a comprehensive hyperparameter search for both SN and DG, and the only change is to use RESNETs instead of VGG.

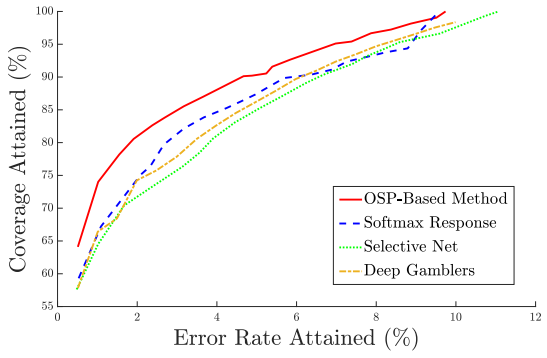

Figure 2: Coverage vs Error Curves for the CIFAR-10 dataset. Higher values of coverage are better. Notice the curious behaviour of SR in that the curve's slope sharply changes close to the best standard error rate.

[9] These rates were selected as follows: the standard classifier was trained with the rate 10^{-4}, which is a typical value in vision tasks. We then picked one value of μ, and trained models using rates in $(10^{-k}, 10^{-j})$ for $(j, k) \in [2:6] \times [2:6]$, tuned thresholds for models at 0.5% target accuracy using validation data, and chose the pair that yielded the best validation coverage. Performance tended to be similar as long as $j \neq k$, and curiously, we found it slightly better to use a smaller rate for the max problem, which goes against the suggestions of Lin et al. 2019.

[10] In particular, we train models at target errors $\varepsilon_i = (i/2)\%$ for $i \in [1:20]$. We then obtain the achieved test error rates $\widehat{\varepsilon}_i$ and coverages c_i for these models. The curves linearly interpolate between $(\widehat{\varepsilon}_i, c_i)$ and $(\widehat{\varepsilon}_{i+1}, c_{i+1})$.

Dataset	Target Error	OSP-based Cov.	OSP-based Error	SR Cov.	SR Error	SN Cov.	SN Error	DG Cov.	DG Error
CIFAR-10	2%	**80.6**	1.91	75.1	2.09	73.0	2.31	74.2	1.98
	1%	**74.0**	1.02	67.2	1.09	64.5	1.02	66.4	1.01
	0.5%	**64.1**	0.51	59.3	0.53	57.6	0.48	57.8	0.51
SVHN-10	2%	**95.8**	1.99	95.7	2.06	93.5	2.03	94.8	1.99
	1%	**90.1**	1.03	88.4	0.99	86.5	1.04	89.5	1.01
	0.5%	**82.4**	0.51	77.3	0.51	79.2	0.51	81.6	0.49
Cats & Dogs	2%	**90.5**	1.98	88.2	2.03	84.3	1.94	87.4	1.94
	1%	**85.4**	0.98	80.2	0.97	78.0	0.98	81.7	0.98
	0.5%	**78.7**	0.49	73.2	0.49	70.5	0.46	74.5	0.48

Table 2: Performance at Low Target Error. The OSP-based scheme is our proposal. SR, SN, DG correspond to softmax-response, selective net, deep gamblers. Errors are rounded to two decimals, and coverage to one.

Dataset	Target Coverage	OSP-based Cov.	OSP-based Error	SR Cov.	SR Error	SN Cov.	SN Error	DG Cov.	DG Error
CIFAR-10	100%	100	9.74	99.99	**9.58**	100	11.07	100	10.81
	95%	95.1	**6.98**	95.2	8.74	94.7	8.34	95.1	8.21
	90%	90.0	**4.67**	90.5	6.52	89.6	6.45	90.1	6.14
SVHN-10	100%	100	4.27	99.97	**3.86**	100	4.27	100	4.03
	95%	95.1	**1.83**	95.1	1.86	95.1	2.53	95.0	2.05
	90%	90.1	**1.01**	90.0	1.04	90.1	1.31	90.0	1.06
Cats & Dogs	100%	100	5.93	100	**5.72**	100	7.36	100	6.16
	95%	95.1	**2.97**	95.0	3.46	95.2	5.1	95.1	4.28
	90%	90.0	**1.74**	90.0	2.28	90.2	3.3	90.0	2.50

Table 3: Performance at High Target Coverage. Same notation as Table 2.

Dataset	Target Error	Overlap
CIFAR-10	2%	0.09%
	1%	0.01%
	0.5%	0.00%
SVHN-10	2%	0.05%
	1%	0.01%
	0.5%	0.00%
Cats & Dogs	2%	0.07%
	1%	0.01%
	0.5%	0.00%

Table 4: Size of overlap between OSP sets in Table 2

4.3.2 Structure of the Solutions

Overlap of OSP solutions is small. Table 4 shows the probability mass of the ambiguous regions for our raw OSP solutions (i.e., the raw sets $\{x : f_k > t\}$ without the max-assignment $\mathcal{S}_k = \{x : f_k > t\} \cap \{x : k = \arg\max f_k\}$) for the models of Table 2. We find that this overlap is very small - much smaller than the 2ε bound in Prop. 1. Empirically, these sets are essentially disjoint, and so the training process is close to tight for the SC problem. We believe that this effect is mainly due to the simple tuning enabled by the softmax normalisation of OSP problem outputs described in §3.

Consistency of rejection regions We say that a sequence of models trained at error levels ε_i have consistent rejection regions if for every $\varepsilon_i < \varepsilon_j$, if $\mathcal{R}_i, \mathcal{R}_j$ are the rejection regions for models trained at these errors, then $\mathbb{P}(\mathcal{R}_i \cap \mathcal{R}_j^c)$ is very small. This means that points that are rejected when designing at a higher error level continue to be rejected for stricter error control. Such consistency may be useful for building cascades of models, or for using the error level at which a point is rejected as a measure of uncertainty.

We found that the models obtained by our procedure are remarkably consistent in the high-accuracy regime. Concretely, for $\varepsilon_i = (i/2)\%$ for $\in [1:5]$, for both CIFAR-10 and Cats & Dogs test sets, the models were entirely consistent, i.e. $\mathcal{R}_j \subset \mathcal{R}_i$ for $j > i$[11], while for SVHN, the only violation was that $|\mathcal{R}_{2.5\%} \cap \mathcal{R}_{2\%}^c| = 2$. Since the test dataset for SVHN has size > 7000, this is a tiny empirical probability of inconsistency of $< 0.03\%$.

[11] Due to time constraints we only checked for higher values of i in the CIFAR-10 case, in which the trend continued until $i = 20$, that is, until full coverage. A curious observation in this case was that in all 20 models for the CIFAR-10 dataset, the same value of μ was best, and the models differed only in the thresholds (this did not occur for SVHN and Cats v/s Dogs). While this obviously implies consistency of the rejection regions, it is unexpected, and suggests that there may be room to improve in our training methodology.

Acknowledgements

This research was supported by National Science Foundation grants CCF-2007350 (VS), CCF-2022446(VS), CCF-1955981 (VS), the Data Science Faculty Fellowship from the Rafik B. Hariri Institute, the Office of Naval Research Grant N0014-18-1-2257 and by a gift from the ARM corporation.

We would like to thank Durmus Alp Emre Acar for helpful discussions regarding implementation of the methods.

References

Acar, Durmus Alp Emre, Aditya Gangrade, and Venkatesh Saligrama (2020). "Budget Learning via Bracketing". In: *International Conference on Artificial Intelligence and Statistics*, pp. 4109–4119.

Bartlett, Peter L and Marten Wegkamp (2008). "Classification with a reject option using a hinge loss". In: *Journal of Machine Learning Research* 9.Aug, pp. 1823–1840.

Blumer, Anselm, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth (1989). "Learnability and the Vapnik-Chervonenkis dimension". In: *Journal of the ACM (JACM)* 36.4, pp. 929–965.

Chandola, Varun, Arindam Banerjee, and Vipin Kumar (2009). "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* 41.3, pp. 1–58.

Chow, C (1957). "An optimum character recognition system using decision functions". In: *IRE Transactions on Electronic Computers* EC-6.4, pp. 247–254.

– (1970). "On optimum recognition error and reject tradeoff". In: *IEEE Transactions on Information Theory* 16.1, pp. 41–46.

Chzhen, Evgenii, Christophe Denis, and Mohamed Hebiri (2019). "Minimax semi-supervised confidence sets for multi-class classification". In: *arXiv preprint arXiv:1904.12527*.

Cortes, Corinna, Giulia DeSalvo, and Mehryar Mohri (2016). "Learning with rejection". In: *International Conference on Algorithmic Learning Theory*. Springer, pp. 67–82.

Denis, Christophe and Mohamed Hebiri (Jan. 2017). "Confidence Sets with Expected Sizes for Multiclass Classification". In: *J. Mach. Learn. Res.* 18.1, pp. 3571–3598. ISSN: 1532-4435.

– (2019). "Consistency of plug-in confidence sets for classification in semi-supervised learning". In: *Journal of Nonparametric Statistics*, pp. 1–31.

Gal, Yarin and Zoubin Ghahramani (2016). "Dropout as a bayesian approximation: Representing model uncertainty in deep learning". In: *international conference on machine learning*, pp. 1050–1059.

Geifman, Yonatan and Ran El-Yaniv (2017). "Selective classification for deep neural networks". In: *Advances in neural information processing systems*, pp. 4878–4887.

– (2019). "SelectiveNet: A Deep Neural Network with an Integrated Reject Option". In: *International Conference on Machine Learning*, pp. 2151–2159.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Herbei, Radu and Marten Wegkamp (2006). "Classification with reject option". In: *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, pp. 709–721.

Krizhevsky, Alex and Geoffrey Hinton (2009). "Learning multiple layers of features from tiny images". In:

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell (2017). "Simple and scalable predictive uncertainty estimation using deep ensembles". In: *Advances in neural information processing systems*, pp. 6402–6413.

Lei, Jing (Oct. 2014). "Classification with confidence". In: *Biometrika* 101.4, pp. 755–769. ISSN: 0006-3444. DOI: 10.1093/biomet/asu038. eprint: https://academic.oup.com/biomet/article-pdf/101/4/755/5029534/asu038.pdf. URL: https://doi.org/10.1093/biomet/asu038.

Lin, Tianyi, Chi Jin, and Michael I Jordan (2019). "On gradient descent ascent for nonconvex-concave minimax problems". In: *arXiv preprint arXiv:1906.00331*.

Liu, Ziyin, Zhikang Wang, Paul Pu Liang, Russ R Salakhutdinov, Louis-Philippe Morency, and Masahito Ueda (2019). "Deep Gamblers: Learning to Abstain with Portfolio Theory". In: *Advances in Neural Information Processing Systems*, pp. 10622–10632.

Mohri, M., A. Rostamizadeh, and A. Talwalkar (2018). *Foundations of Machine Learning*. Adaptive Computation and Machine Learning series. MIT Press. ISBN: 9780262039406.

Nan, Feng and Venkatesh Saligrama (2017a). "Adaptive classification for prediction under a budget". In: *Advances in Neural Information Processing Systems*, pp. 4727–4737.

– (2017b). "Dynamic model selection for prediction under a budget". In: *arXiv preprint arXiv:1704.07505*.

Netzer, Yuval, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng (2011). "Reading Digits in Natural Images with Unsupervised Feature Learning". In: *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*. URL: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Ni, Chenri, Nontawat Charoenphakdee, Junya Honda, and Masashi Sugiyama (2019). "On the Calibration of Multiclass Classification with Rejection". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 2586–2596. URL: http://papers.nips.cc/paper/8527-on-the-calibration-of-multiclass-classification-with-rejection.pdf.

Ramaswamy, Harish G., Ambuj Tewari, and Shivani Agarwal (2018). "Consistent algorithms for multiclass classification with an abstain option". In: *Electron. J. Statist.* 12.1, pp. 530–554. DOI: 10.1214/17-EJS1388. URL: https://doi.org/10.1214/17-EJS1388.

Rigollet, Philippe and Xin Tong (2011). "Neyman-pearson classification, convexity and stochastic constraints". In: *Journal of Machine Learning Research* 12.Oct, pp. 2831–2855.

Sadinle, Mauricio, Jing Lei, and Larry Wasserman (2019). "Least Ambiguous Set-Valued Classifiers With Bounded Error Levels". In: *Journal of the American Statistical Association* 114.525, pp. 223–234. DOI: 10.1080/01621459.2017.1395341. eprint: https://doi.org/10.1080/01621459.2017.1395341. URL: https://doi.org/10.1080/01621459.2017.1395341.

Shafer, Glenn and Vladimir Vovk (2008). "A Tutorial on Conformal Prediction." In: *Journal of Machine Learning Research* 9.3.

Shekhar, Shubhanshu, Mohammad Ghavamzadeh, and Tara Javidi (2019). "Binary Classification with Bounded Abstention Rate". In: *arXiv preprint arXiv:1905.09561*.

Tong, Xin (2013). "A Plug-in Approach to Neyman-Pearson Classification". In: *Journal of Machine Learning Research* 14.56, pp. 3011–3040. URL: http://jmlr.org/papers/v14/tong13a.html.

Vapnik, Vladimir N. (2000). *The Nature of Statistical Learning Theory*. Springer, New York. ISBN: 9781441931603. DOI: 10.1007/978-1-4757-3264-1.

Vovk, Vladimir, Alex Gammerman, and Glenn Shafer (2005). *Algorithmic learning in a random world*. Springer Science & Business Media.

Wegkamp, Marten (2007). "Lasso type classifiers with a reject option". In: *Electronic Journal of Statistics* 1, pp. 155–168.

Wegkamp, Marten and Ming Yuan (2011). "Support vector machines with a reject option". In: *Bernoulli* 17.4, pp. 1368–1385.

Wiener, Yair and Ran El-Yaniv (2011). "Agnostic selective classification". In: *Advances in neural information processing systems*, pp. 1665–1673.

Xu, Zhixiang (Eddie), Matt J. Kusner, Kilian Q. Weinberger, Minmin Chen, and Olivier Chapelle (2014). "Classifier Cascades and Trees for Minimizing Feature Evaluation Cost". In: *Journal of Machine Learning Research* 15, pp. 2113–2144. URL: http://jmlr.org/papers/v15/xu14a.html.

El-Yaniv, Ran and Yair Wiener (2010). "On the foundations of noise-free selective classification". In: *Journal of Machine Learning Research* 11.May, pp. 1605–1641.

Yuan, Ming and Marten Wegkamp (2010). "Classification Methods with Reject Option Based on Convex Risk Minimization". In: *Journal of Machine Learning Research* 11.5, pp. 111–130. URL: http://jmlr.org/papers/v11/yuan10a.html.

Zhu, Pengkai, Durmus Alp Emre Acar, Nan Feng, Prateek Jain, and Venkatesh Saligrama (2019). "Cost aware inference for iot devices". In: *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2770–2779.