Post-hoc Estimators for Learning to Defer to an Expert

Harikrishna Narasimhan
Google Research, Mountain View
hnarasimhan@google.com

Wittawat Jitkrittum
Google Research, New York
wittawat@google.com

Aditya Krishna Menon
Google Research, New York
adityakmenon@google.com

Ankit Singh Rawat
Google Research, New York
ankitsrawat@google.com

Sanjiv Kumar
Google Research, New York
sanjivk@google.com

Abstract

Many practical settings allow a classifier to *defer* predictions to one or more costly *experts*. For example, the *learning to defer* paradigm allows a classifier to defer to a human expert, at some monetary cost. Similarly, the *adaptive inference* paradigm allows a base model to defer to one or more large models, at some computational cost. The goal in these settings is to learn classification and deferral mechanisms to optimise a suitable accuracy-cost tradeoff. To achieve this, a central issue studied in prior work is the design of a coherent loss function for both mechanisms. In this work, we demonstrate that existing losses can *underfit* the training set when there is a non-trivial deferral cost, owing to an implicit application of a high level of label smoothing. To resolve this, we propose two *post-hoc estimators* that fit a deferral function on top of a base model, either by threshold correction, or by learning when the base model's error rate exceeds the cost of deferring to the expert. Both approaches are equipped with theoretical guarantees, and empirically yield effective accuracy-cost tradeoffs on learning to defer and adaptive inference benchmarks.

1 Introduction: Learning to Defer to an Expert Model

Supervised classification conventionally considers learning a *single* model with good average-case test performance. However, many practical settings allow for choosing amongst *multiple* models to classify a given sample. For example, in the *learning to defer* (*L2D*) paradigm [18], a *base* classifier may be used in conjunction with a human *expert*. Specifically, the classifier has the option of *deferring* to the expert, at the expense of incurring some additional cost (e.g., monetary expense). The goal is to learn a classifier and deferral function achieving a suitable accuracy-cost tradeoff. This problem has received a surge of interest [26, 19, 22, 34] given the increasing use of machine learning in high-stakes decision making. Interestingly, similar considerations are also found in the problem of *adaptive inference* for resource-constrained prediction [13]: here, a base model can choose to defer to one or more experts, which are themselves learning models with higher capacity and inference cost. Similar to L2D, the goal is to achieve a suitable tradeoff between accuracy and (inference) cost.

A key problem in such settings is deciding when the base model should defer its prediction to the expert(s). An intuitive strategy is to defer when the model has low *prediction confidence* (e.g., classification margin). While theoretically grounded [8], this strategy can under-perform when the base model has low capacity [9]. An alternate strategy is to explicitly formulate and optimise a joint loss function for the base classifier and deferral function. This has been followed with success in the L2D literature, particularly with the cost-sensitive softmax cross-entropy loss (`CSS`) of Mozannar and Sontag [19], and one-versus-all (`OvA`) loss of Verma and Nalisnick [34].

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

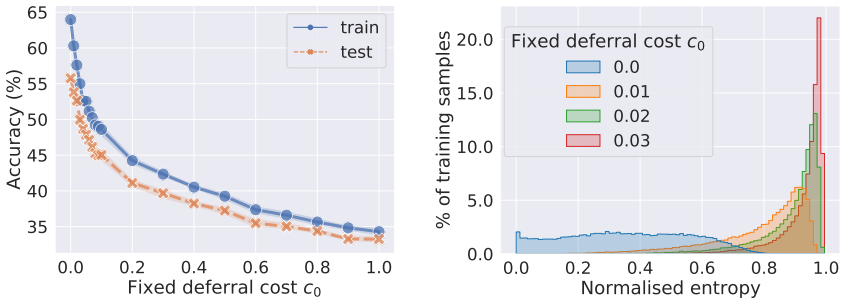

Figure 1: Illustration of underfitting of the cost-sensitive softmax cross-entropy (CSS) loss of Mozannar and Sontag [19]. On CIFAR-100, we consider a learning to defer setting comprising a ResNet-8 base model and a ResNet-32 "expert" h_{exp}. We assume a cost $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$ of deferring to the expert (see §2 for details on notation), where the *fixed cost* c_0 is varied from $[0, 1]$. For each c_0, we train the base model using the CSS loss (4), and report the resulting accuracy. For $c_0 > 0$, the base model exhibits *underfitting*, evidenced by significant degradation in the *training* accuracy. In §3.1, we trace this behaviour to the loss applying a high level of *label smoothing* [31] to incorrect labels. Consequently, the *entropy* of the base model probabilities steadily increase with c_0 (right panel).

While the above losses have proven effective, they are not without limitations. Indeed, both focus on the setting where the cost of deferring is only given by the probability that the expert makes a mistake on a sample; this does not consider the additional *fixed cost* of querying an expert in the first place. As we shall see (§3), incorporating any non-zero fixed cost makes these losses prone to *underfitting*; cf. Figure 1. This behaviour is owing to the fact that the losses implicitly apply a high level of *label smoothing*. The setting of non-zero fixed cost is particularly important in adaptive inference settings, where querying an expert inherently involves paying an increased computational cost.

In this work, we resolve this issue via deferral schemes based on *post-hoc estimators*. In a nutshell, we follow a two-step procedure wherein we fit a deferral function on top of a base model, either by threshold correction, or by learning to predict when the base model's error rate exceeds the deferral cost. This allows us to build on top of the standard (zero fixed cost) CSS and OvA solutions, resulting in an empirically effective strategy with theoretical guarantees. In sum, our contributions are:

(i) we identify limitations in prior losses for L2D, particularly when the cost of deferring to an expert involves a non-zero fixed cost in addition to the probability of the expert making a mistake (§3)

(ii) we propose new post-hoc estimators for learning a deferral function, which overcome the above limitation while enjoying theoretical guarantees (§4, Theorem 2, Theorem 4)

(iii) we empirically show that the new post-hoc estimators work well on synthetic and real-world benchmark datasets for both L2D and adaptive inference settings (§5).

In the course of our analysis, we additionally generalise the OvA loss to reliably account for general deferral costs ((10), Lemma 1), and to reliably estimate label probabilities in practical settings (12).

2 Background and Notation

Our interest is in learning settings where a classifier is equipped with the option of deferring to an expert model. For concreteness, we shall focus our exposition on the learning to defer paradigm.

2.1 Multi-class Classification

Fix an instance space $\mathcal{X}$ and label space $\mathcal{Y} = [L] \doteq \{1, 2, \ldots, L\}$, and let $\mathbb{P}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$. Given a training sample $S = \{(x_n, y_n)\}_{n \in [N]}$ drawn from $\mathbb{P}$, multi-class classification seeks a *classifier* $h \colon \mathcal{X} \to \mathcal{Y}$ with low *misclassification error* $R(h) = \mathbb{P}(y \neq h(x))$. We may parameterise h as $h(x) = \operatorname{argmax}_{y' \in \mathcal{Y}} f_{y'}(x)$ for a *scorer* $f \colon \mathcal{X} \to \mathbb{R}^L$. Such a scorer may be learned by minimising the *empirical surrogate risk* $\hat{R}(f) = \frac{1}{N} \sum_{n \in [N]} \ell(x_n, y_n, f(x_n))$ for some *loss* $\ell \colon \mathcal{X} \times \mathcal{Y} \times \mathbb{R}^L \to \mathbb{R}_+$, such as the *softmax cross-entropy* $\ell(x, y, f(x)) = \log \left[\sum_{y' \in \mathcal{Y}} e^{f_{y'}(x)} \right] - f_y(x)$. For binary classification problems with $L = 2$, it is common to enforce $f_2(x) = -f_1(x)$ and set $\ell(x, y, f(x)) = \phi(f_y(x))$ for some *margin loss* $\phi \colon \mathbb{R} \to \mathbb{R}_+$, such as $\phi(z) = \log(1 + e^{-z})$.

2.2 Learning to Defer (L2D) Problem

In the *learning to defer* (*L2D*) problem [18], one seeks a classifier that can either make a standard prediction in $\mathcal{Y}$ via some *base* model, or *defer* its prediction to an *expert* model $h_{\text{exp}} \colon \mathcal{X} \to \mathcal{Y}$, at the expense of incurring a sample-dependent *cost* $c_{\text{exp}}(x, y) > 0$. Typically, c_{exp} takes into account both the *fixed cost* of deferring to the expert, which we will assume to be a constant $c_0 \in [0, 1]$ (e.g., normalised monetary cost), and an estimate of the probability that the expert makes a mistake on the example. Canonical examples include $c^{(1)}_{\text{exp}}(x, y) = c_0$ and $c^{(2)}_{\text{exp}}(x, y) = c_0 + \mathbf{1}\left(y \neq h_{\text{exp}}(x)\right)$. The choice $c^{(1)}_{\text{exp}}(x, y)$ corresponds to learning with a *reject option* or *abstention* [8, 2, 9, 6].

Given a suitable $c_{\text{exp}}(x, y)$, the learning problem may be formalised as follows. Consider a classifier $\bar{h} \colon \mathcal{X} \to \mathcal{Y} \cup \{\perp\}$ equipped with a "defer" option $\perp$. Our goal is to minimise

$$R_{\text{def}}(\bar{h}) = \mathbb{E}\left[\mathbf{1}(y \neq \bar{h}(x)) \cdot \mathbf{1}(\bar{h}(x) \neq \perp) + c_{\text{exp}}(x, y) \cdot \mathbf{1}(\bar{h}(x) = \perp)\right]. \tag{1}$$

Intuitively, when $\bar{h}$ chooses not to defer, we incur the usual misclassification error; otherwise, we incur the cost $c_{\text{exp}}(x, y)$ of invoking the expert h_{exp}. One can cast (1) as an instantiation of *cost-sensitive learning* [11]; see Appendix B. We remark that Okati et al. [22] considered a slightly different setting, wherein one enforces a *hard* constraint on the expert cost $c_{\text{exp}}(x, y)$. We focus on a *soft* additive penalty in (1), but we may handle hard constraints via standard Lagrangian theory; see Appendix G.

In practice, it is useful to parameterise $\bar{h}$ in terms of a scorer $\bar{f} \colon \mathcal{X} \to \mathbb{R}^{L+1}$ with label logits $\{\bar{f}_1(x), \ldots, \bar{f}_L(x)\}$, and "defer" logit $\bar{f}_\perp(x)$. We then predict via

$$\bar{h}(x) = \operatorname{argmax}_{y \in \mathcal{Y}} \bar{f}_y(x) \text{ if } \max_{y \in \mathcal{Y}} \bar{f}_y(x) > \bar{f}_\perp(x); \quad \bar{h}(x) = \perp \text{ otherwise.} \tag{2}$$

In practice, we may parameterise $\bar{f}$ via neural network logits: for any $y' \in \mathcal{Y} \cup \{\perp\}$, $\bar{f}_{y'}(x) = w_{y'}^\top \Phi(x)$ for weights $w_{y'}$ and embedding Φ. This parameterisation induces implicit sharing amongst all logits.

One conceptually useful quantity is the *Bayes-optimal* $\bar{h}^* = \operatorname{argmin}_{\bar{h} \colon \mathcal{X} \to \mathcal{Y} \cup \{\perp\}} R_{\text{def}}(\bar{h})$, which is

$$\bar{h}^*(x) = \begin{cases} \operatorname*{argmax}_{y \in \mathcal{Y}} \mathbb{P}(y \mid x) & \text{if } 1 - \max_{y \in \mathcal{Y}} \mathbb{P}(y \mid x) < \mathop{\mathbb{E}}_{y \mid x}\left[c_{\text{exp}}(x, y)\right] \\ \perp & \text{else.} \end{cases} \tag{3}$$

Intuitively, we defer iff the expected cost of querying the expert is smaller than the expected error of the base model. For the constant cost function $c^{(1)}_{\text{exp}}(x, y)$, this is known as *Chow's rule* [8].

2.3 Losses for Learning to Defer

We review some popular recent strategies to learn $\bar{f}$ (and thus $\bar{h}$) below.

Confidence thresholding. For $c_{\text{exp}}(x, y) = c_0$, one strategy is to learn $\bar{f}_1, \ldots, \bar{f}_L$ by minimising the softmax cross-entropy, and fix $\bar{f}_\perp(x) = 1 - c_0$. This *confidence thresholding* approach [21] mimics the Bayes-optimal classifier (3). A related series of approaches [27, 38, 1] make deferral decisions by comparing the base model confidence with the confidence in the expert being correct. When the base model has low-capacity, such approaches may be sub-optimal as the base model may sub-optimally allocate its finite capacity by classifying examples that the expert can easily predict.

Cost-sensitive softmax cross-entropy (`CSS`). To overcome the limitations of confidence thresholding, it is desirable to explicitly learn $\bar{f}_\perp$ as well. To this end, Mozannar and Sontag [19] proposed a *cost-sensitive softmax cross-entropy* (`CSS`): for constant upper bound $c_{\max} \geq c(x, y)$, this is

$$\ell_{\text{CSS}}(x, y, \bar{f}(x)) = -\sum_{y' \in \mathcal{Y}} (c_{\max} - \mathbf{1}(y \neq y')) \cdot \log\left(\bar{p}_{y'}(x)\right) - (c_{\max} - c_{\text{exp}}(x, y)) \cdot \log\left(\bar{p}_\perp(x)\right), \tag{4}$$

where $\bar{p}_y(x) = \exp(\bar{f}_y(x)) / \sum_{y' \in \mathcal{Y} \cup \{\perp\}} \exp(\bar{f}_{y'}(x))$ is the softmax distribution over the standard *and* the defer label. For $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}\left(y \neq h_{\text{exp}}(x)\right)$, we have $c_{\max} = 1 + c_0$. The loss ℓ_{CSS} is Fisher-consistent [19, Proposition 1], with Bayes-optimal prediction

$$(\forall y' \in \mathcal{Y})\ \bar{p}^*_{y'}(x) \propto c_{\max} - (1 - \mathbb{P}(y' \mid x)) \qquad \bar{p}^*_\perp(x) \propto c_{\max} - \mathbb{E}_{y \mid x}[c_{\text{exp}}(x, y)]. \tag{5}$$

One-versus-all (`OvA`). Despite the Fisher consistency of ℓ_{CSS}, Verma and Nalisnick [34] observed that empirically, the loss may not produce *calibrated* estimates of the expert's error probability. Verma and

Nalisnick thus proposed a *one-versus-all* (OvA) loss for the cost function $c_{\text{exp}}(x, y) = \mathbf{1}(y \neq h_{\text{exp}}(x))$:

$$\ell_{\text{OVA}}(x, y, \bar{f}(x)) = \phi(\bar{f}_y(x)) + \sum_{y' \in \mathcal{Y} - \{y\}} \phi(-\bar{f}_{y'}(x)) + (1 - c_{\text{exp}}(x, y)) \cdot \phi(\bar{f}_\perp(x)) + c_{\text{exp}}(x, y) \cdot \phi(-\bar{f}_\perp(x)). \tag{6}$$

Here, $\phi : \mathbb{R} \to \mathbb{R}_+$ is a binary *proper composite* loss [28], e.g., $\phi(z) = \log(1 + e^{-z})$. Such losses have an associated *inverse link function* $\psi \colon \mathbb{R} \to [0, 1]$, and produce Bayes-optimal scorer

$$(\forall y' \in \mathcal{Y})\, \psi(\bar{f}^*_{y'}(x)) = \mathbb{P}(y' \mid x) \qquad \psi(\bar{f}^*_\perp(x)) = 1 - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x, y)\right]. \tag{7}$$

Intuitively, both (4) and (6) seek to have the first L logits $\{\bar{f}_1(x), \ldots, \bar{f}_L(x)\}$ estimate $\mathbb{P}(y \mid x)$, and the reject logit $\bar{f}_\perp(x)$ model the cost of querying the expert. With this, the highest scoring logit provides us with an estimate of the Bayes-optimal deferral decision (3).

3 Limitations of Existing Learning to Defer Losses

We now study existing losses for learning to defer more carefully, and show that they may *underfit* in an important practical setting. Specifically, recall that a canonical choice of deferral cost function in (1) is $c_{\text{exp}}(x, y) = \mathbf{1}\left(y \neq h_{\text{exp}}(x)\right) + c_0$, for constant $c_0 \geq 0$. Here, c_0 represents a *fixed cost* of querying the expert, which is independent of whether the expert misclassifies the sample.

The setting of *non-zero fixed cost* $c_0 > 0$ is of practical interest; e.g., this is intrinsic to adaptive inference settings, wherein querying the expert model fundamentally involves increasing a computational overhead (see §4.3 for more discussion). Thus, an ideal surrogate loss for (1) should be performant for $c_0 > 0$. Unfortunately, we now show that this does *not* hold for the CSS and OvA losses.

3.1 Underfitting of Cost-Sensitive Softmax Cross-Entropy when $c_0 > 0$

The cost-sensitive softmax cross-entropy (CSS) is in principle applicable for any deferral cost $c_{\text{exp}}(x, y)$, and in particular $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$ for any $c_0 \geq 0$. However, the focus in Mozannar and Sontag [19] was the case $c_0 = 0$. In fact, we now show that when $c_0 > 0$, the loss can *underfit*. To see this, note that the tightest possible maximal cost is $c_{\max} = 1 + c_0$. With this choice, (4) becomes

$$\ell_{\text{CSS}}(x, y, \bar{f}(x)) \;=\; -\log\left(\bar{p}_y(x)\right) + c_0 \cdot \sum_{y'} -\log(\bar{p}_{y'}(x)) \;-\; \mathbf{1}(y = h_{\text{exp}}(x)) \cdot \log\left(\bar{p}_\perp(x)\right), \tag{8}$$

When $c_0 = 0$, the second term vanishes, and we have a conic combination of the log-loss on the true and deferral labels. When $c_0 > 0$, the second term is active, and applies a form of *label smoothing* [31]: we allow for *all* labels $y' \in \mathcal{Y} - \{y\}$ to be potential "positives" for x, with a weight of $c_0 \in [0, 1]$. Put differently, we treat the ground-truth label *distribution* as a mixture of a one-hot distribution on y, and a uniform distribution over all labels. As $c_0 \to 1^-$, we apply a large amount of smoothing on labels other than y; this can be problematic. Concretely, consider an x where the expert gives the wrong prediction, i.e., $\mathbb{P}(y = h_{\text{exp}}(x)) = 0$. Then, the Bayes-optimal solution (5) will be

$$(\forall y' \in \mathcal{Y})\, \bar{p}^*_{y'}(x) = (\mathbb{P}(y \mid x) + c_0)/(1 + L \cdot c_0) \qquad \bar{p}^*_\perp(x) = 0. \tag{9}$$

Observe that for $c_0 > 0$, as L increases, the range of values in $\bar{p}^*$ shrinks: indeed, the gap between the optimal probability for true label y and a competing label y' will be $\mathcal{O}(1/L)$. This small *probability margin* makes learning challenging: given a finite sample, it will be easy for the model to mistakenly swap the true label y with some competing label y'. Thus, in settings where L is moderately large, and where there is a non-zero fixed deferral cost $c_0 > 0$, ℓ_{CSS} may underperform. Such issues have also been observed [7] for a similar loss developed for label noise [25].

As an illustration, we consider in Figure 1 a learning to defer setting comprising a ResNet-8 base model and a ResNet-32 "expert" on CIFAR-100. We assume a deferral cost $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$, where c_0 is varied in $[0, 1]$. For each c_0, we train the base model using (4), and report the train and test accuracy. As c_0 increases, the base model shows significant *training* accuracy degradation, i.e., it underfits. The high level of label smoothing plays a role: in the right panel, we see that the *entropy* of the model's predicted distribution over labels increases with c_0. (The entropy is normalised by $\log L$ to lie in $[0, 1]$.) See Appendix I for additional plots, and Appendix I.3 for results when $c_{\text{exp}}(x, y) = c_0$.

3.2 Underfitting of One-Versus-All Loss when $c_0 > 0$

The one-versus-all (OvA) loss of Verma and Nalisnick [34] was presented for the specific case of $c_0 = 0$; as stated, the loss is not applicable when $c_0 > 0$. We may however extend this loss to a general $c_{\text{exp}}(x, y)$ with upper bound $c_{\max} \geq 1$ (see Appendix D for a more general derivation):

$$\ell_{\text{OVA}}(x, y, \bar{f}(x)) = c_{\max} \cdot \phi(\bar{f}_y(x)) + \sum_{y' \neq y} \Big((c_{\max} - 1) \cdot \phi(\bar{f}_{y'}(x)) \; + \; \phi(-\bar{f}_{y'}(x)) \Big) \\ + (c_{\max} - c_{\text{exp}}(x, y)) \cdot \phi(\bar{f}_{\perp}(x)) + c_{\text{exp}}(x, y) \cdot \phi(-\bar{f}_{\perp}(x)). \tag{10}$$

Compared to (6), we have an additional weighted sum over each $\phi_{y'}(x)$. As with OvA, this loss results in a coherent Bayes-optimal solution when ϕ is a strictly proper composite loss [5, 28]. Recall that such losses are characterised by an inverse link function $\psi \colon \mathbb{R} \to [0, 1]$, such as the logistic loss $\phi(z) = \log(1 + e^{-z})$ with sigmoid inverse link $\psi(v) = (1 + \exp(-v))^{-1}$.

Lemma 1. *Suppose ϕ is a strictly proper composite loss with inverse link function ψ. Then, the one-versus-all loss in (10) is calibrated for the learning to defer risk (1), with Bayes-optimal scorer*

$$(\forall y' \in \mathcal{Y})\, \psi(\bar{f}^*_{y'}(x)) = \frac{c_{\max} - (1 - \mathbb{P}(y' \mid x))}{c_{\max}} \quad \psi(\bar{f}^*_{\perp}(x)) = \frac{c_{\max} - \mathbb{E}_{y|x}\left[c_{exp}(x, y)\right]}{c_{\max}}.$$

Unlike the optimal scorer for the CSS loss, here $\sum_{y' \in [L]} \psi(\bar{f}^*_{y'}(x)) \neq 1$. Nonetheless, the highest scoring logit agrees with the Bayes-optimal classifier (3) for learning to defer.

Reviewing (10), we see that when $c_{\max} > 1$, we apply a form of label smoothing by additional terms of the form $\sum_{y' \neq y} \phi(\bar{f}_{y'}(x))$, i.e., we effectively treat all $y' \neq y$ as potential positives. As with the cost-sensitive softmax entropy, this can result in underfitting; recall that with $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$, we have an upper bound $c_{\max} = 1 + c_0 > 1$. We empirically confirm in Appendix I.2 that we indeed observe analogous underfitting behaviour as per the CSS loss in Figure 1.

The above shows that the CSS and OvA losses have a subtle limitation when $c_0 > 0$. This is not in conflict with the Fisher-consistency of these losses: the latter establishes that the asymptotic minimiser of these losses agrees with the Bayes-optimal classifier (3). Our analysis explicates that the precise *form* of these Bayes-optimal solutions may not be amenable to learning from a *finite sample*.

Since both losses seamlessly handle $c_0 = 0$, a natural question is whether one can leverage this solution to help guide the one for $c_0 > 0$. We now present post-approaches that do precisely this.

4 Post-hoc Approaches to Learning to Defer

We now present two simple *post-hoc* strategies for learning to defer, which build on the CSS and OvA solutions for $c_0 = 0$ via *threshold correction* and *re-training* the deferral function.

4.1 Post-hoc Threshold Correction

Our first post-hoc strategy (Figure 2) exploits the form of the Bayes-optimal classifier in (3): to mimic this classifier for the cost function $c_{\text{exp}}(x, y) = \mathbf{1}(y \neq h_{\text{exp}}(x)) + c_0$, it suffices to obtain estimates of $\mathbb{P}(y \mid x)$ and the expected deferral cost $\mathbb{E}[c_{\text{exp}}(x, y)] = \mathbb{P}(y \neq h_{\text{exp}}(x)) + c_0$. Consequently, suppose we learn models $\pi \colon \mathcal{X} \to [0, 1]^L$ and $\epsilon \colon \mathcal{X} \to [0, 1]$ estimating $\mathbb{P}(y \mid x)$ and $\mathbb{P}(y \neq h_{\text{exp}}(x))$ respectively. Then, for any $c_0 \geq 0$, we can approximate (3) via the classifier

$$\bar{h}(x) = \text{argmax}_{y \in \mathcal{Y}}\, \pi_y(x) \text{ if } 1 - \max_{y \in \mathcal{Y}} \pi_y(x) < \epsilon(x) + c_0; \quad \bar{h}(x) = \perp \text{ else.} \tag{11}$$

Intuitively, this procedure ought to perform well when $\pi(x)$ is close to $\eta(x) \doteq [\mathbb{P}(y \mid x)]_{y \in \mathcal{Y}}$, and $\epsilon(x)$ is close to $\mathbb{P}(y \neq h_{\text{exp}}(x))$. The following *excess risk bound* for $\bar{h}$ formalises this intuition.

Theorem 2. *Pick any $\pi \colon \mathcal{X} \to [0, 1]^L$ and $\epsilon \colon \mathcal{X} \to [0, 1]$. Let $\bar{h}$ be the corresponding classifier in (11). Then, the excess risk of $\bar{h}$ compared to the Bayes-optimal classifier $\bar{h}^*$ (3) is:*

$$R_{\text{def}}(\bar{h}) - R_{\text{def}}(\bar{h}^*) \leq c_{\max} \cdot \mathop{\mathbb{E}}_{x} \left[\|\eta(x) - \pi(x)\|_1 \right] \; + \; 2 \cdot \mathop{\mathbb{E}}_{x} \left[|\mathbb{P}(y \neq h_{\text{exp}}(x)) - \epsilon(x)| \right].$$

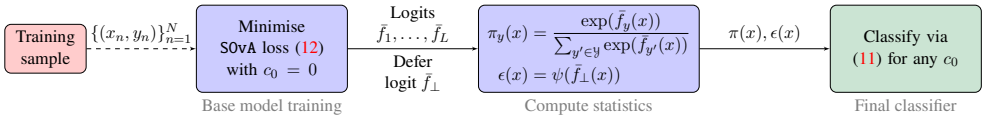

Figure 2: Summary of post-hoc threshold correction procedure (§4.1).

Crucially, observe that we can learn π and ϵ *once*, and then sweep across different c_0 values in (11). Thus, we can attempt to leverage the solutions for either the CSS or OvA losses with $c_0 = 0$ — recalling that these do *not* exhibit underfitting, unlike when $c_0 > 0$ — and then simply vary c_0 post-hoc. Indeed, both losses' Bayes-optimal solutions (5), (7) for $c_0 = 0$ suggest that a transform of $\bar{f}_1, \ldots, \bar{f}_L$ provide estimates of $\mathbb{P}(y \mid x)$, while a transform of $\bar{f}_\perp$ provides an estimate of $\mathbb{P}(y \neq h_{\text{exp}}(x))$.

Unfortunately, in practice, both losses are potentially suboptimal to learn π, ϵ. For the CSS loss, Verma and Nalisnick [34] showed it does not provide *calibrated* estimates of $\mathbb{P}(y \neq h_{\text{exp}}(x))$. On the other hand, while the OvA loss provides calibrated estimates of $\mathbb{P}(y \neq h_{\text{exp}}(x))$, its probability estimates of $\mathbb{P}(y \mid x)$ may underperform when L is large (see Appendix J). We overcome this with a simple solution: we estimate π via the softmax cross-entropy, and ϵ via the OvA loss. This is achieved via a *hybrid* softmax cross-entropy plus OvA loss (SOvA):

$$\ell_{\text{SOVA}}(x, y, \bar{f}(x)) = -\log p_y(x) + (c_{\max} - c_{\text{exp}}(x, y)) \cdot \phi(\bar{f}_\perp(x)) + c_{\text{exp}}(x, y) \cdot \phi(-\bar{f}_\perp(x)), \quad (12)$$

where $p_y(x) = \exp(\bar{f}_y(x)) / \sum_{y' \in \mathcal{Y}} \exp(\bar{f}_{y'}(x))$. Following the Bayes-optimal solution, we set $\pi(x) = (p_1(x), \ldots, p_L(x))$ and $\epsilon(x) = \psi(\bar{f}_\perp(x))$. As with the CSS and OvA losses, we may parameterise $\bar{f}_{y'}(x) = w_{y'}^\top \Phi(x)$ for shared Φ, and thus induce information sharing amongst all logits.

The above post-hoc approach crucially relies on estimating the expert's error rate. A potentially simpler task is learning to predict whether it is *beneficial* to invoke the expert, i.e., whether the expert's error is lower than that of the base model. We now consider an alternate strategy that implements this.

4.2 Post-hoc Rejector Training

Our second post-hoc strategy revisits the fundamental risk (1) underpinning the learning to defer problem. While defined in terms of a classifier $\bar{h} \colon \mathcal{X} \to \mathcal{Y} \cup \{\perp\}$, we may rewrite this in terms of a *base classifier* $h \colon \mathcal{X} \to \mathcal{Y}$ and *rejector* $r \colon \mathcal{X} \to \mathbb{R}$, with the latter denoting the confidence in deferring:

$$R_{\text{def}}(h, r) = \mathbb{E}_{(x,y)}[\mathbf{1}(r(x) < 0) \cdot c_{\text{mod}}(x, y) + \mathbf{1}(r(x) > 0) \cdot c_{\text{exp}}(x, y)], \quad (13)$$

where $c_{\text{mod}}(x, y) = \mathbf{1}(y \neq h(x))$. Given a scorer $\bar{f} \colon \mathcal{X} \to \mathbb{R}^{L+1}$, we may set $h(x) = \operatorname{argmax}_{y' \in \mathcal{Y}} \bar{f}_{y'}(x)$ and $r(x) = \bar{f}_\perp(x) - \max_{y' \in \mathcal{Y}} \bar{f}_{y'}(x)$. Both CSS and OvA construct surrogate losses that are applied *jointly* over all logits $\bar{f}_1, \ldots, \bar{f}_L, \bar{f}_\perp$; equally, they construct a joint surrogate loss for both the base classifier h and rejector r. As established earlier, the solution obtained from these losses is reasonable under fixed deferral cost $c_0 = 0$, but tends to suffer when $c_0 > 0$.

We may resolve this issue via a two-step procedure (see Figure 3): in the first step, we obtain standard logits $\bar{f}_1(x), \ldots, \bar{f}_L(x)$ by minimising either the CSS or OvA loss with $c_0 = 0$. In the second step, we train *only* the deferral logit, by constructing a *partial* surrogate for only the rejector $r(x)$ in (13). Concretely, for a binary surrogate loss $\phi \colon \mathbb{R} \to \mathbb{R}_+$, we may define the following surrogate for r:

$$\ell_{\text{rej}}(x, y, r(x); h) = c_{\text{mod}}(x, y) \cdot \phi(r(x)) + c_{\text{exp}}(x, y) \cdot \phi(-r(x)), \quad (14)$$

and minimize the resulting partial surrogate risk over r:

$$R_{\text{def}}^{\phi}(h, r) = \mathbb{E}\left[\ell_{\text{rej}}(x, y, r(x); h)\right]. \quad (15)$$

With this loss, c_0 only plays a role in the $c_{\text{exp}}(x, y)$ weighting term, and thus only influences the deferral decision; it does not induce any underfitting of the standard logits $\bar{f}_1, \ldots, \bar{f}_L$. Observe also that $c_{\text{mod}}(x, y)$ will involve the *frozen* logits $\bar{f}_1, \ldots, \bar{f}_L$ obtained from the first stage. We may further leverage these logits by parameterising $r(x) = \bar{f}_\perp(x) - \max_{y' \in \mathcal{Y}} \bar{f}_{y'}(x)$, and optimising only the $\bar{f}_\perp(x)$ term. In practice, exploiting the *probabilities* from the first stage model can offer a closer approximation to the Bayes-optimal classifier; see Appendix F for more discussion.

Theoretically, the surrogate loss in (14) provides a calibrated deferral rule for any base classifier h. For strictly proper composite ϕ with inverse link $\psi \colon \mathbb{R} \to [0, 1]$, the optimal rejector is as follows.

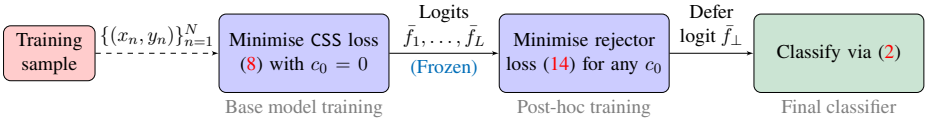

Figure 3: Summary of post-hoc training procedure (§4.2).

Lemma 3. *The Bayes-optimal rejector for the loss* (14) *is* $\psi(r^*(x)) = \frac{\mathbb{E}_{y|x}[c_{mod}(x,y)]}{\mathbb{E}_{y|x}[c_{mod}(x,y)] + \mathbb{E}_{y|x}[c_{exp}(x,y)]}$.

Note that when the first stage logits agree with the Bayes-optimal solution, $\mathbb{E}_{y|x}[c_{\text{mod}}(x,y)] = 1 - \max_{y \in \mathcal{Y}} \mathbb{P}(y \mid x)$, and so deferring when $\psi(r^*(x)) > 0.5$ produces the optimal decision.

We next provide a bound an *excess risk bound* for a classifier learned using the post-hoc training procedure, under the assumption that the surrogate loss ϕ used is classification calibrated [3].

Theorem 4. *Suppose ϕ is classification calibrated and $c_{\text{exp}}(x,y) \in [c_{\min}, c_{\max}]$ for some $c_{\min} > 0$. Denote the misclassification risk for classifier h by $R_{\text{err}}(h) = \mathbb{E}[\mathbf{1}(y \neq h(x))]$. Let $\hat{r}$ be the rejector obtained by minimizing the partial surrogate risk in* (15) *for base classifier $\hat{h}$. Then the excess* L2D *risk for the resulting classifier $(\hat{h}, \hat{r})$ is bounded in terms of the excess surrogate risk for the rejector $\hat{r}$ and the excess misclassification risk for the base classifier $\hat{h}$:*

$$R_{\text{def}}(\hat{h}, \hat{r}) - \min_{\bar{h}: \mathcal{X} \to [L] \cup \{\perp\}} R_{\text{def}}(\bar{h})$$
$$\leq \Psi\left(R^{\phi}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r: \mathcal{X} \to \mathbb{R}} R^{\phi}_{\text{def}}(\hat{h}, r) \right) + R_{\text{err}}(\hat{h}) - \min_{h: \mathcal{X} \to [L]} R_{\text{err}}(h),$$

for some increasing function $\Psi : \mathbb{R}_+ \to \mathbb{R}_+$ with $\Psi(0) = 0$.

4.3 Discussion and Extensions

Relation to existing work. Post-hoc threshold correction generalises Chow's rule [8] for learning to reject, wherein $c_{\text{exp}}(x,y) = c_0$. It also generalises the approaches of Raghu et al. [27], Wilder et al. [37], Bansal et al. [1] for learning to defer: these assume $c_0 = 0$, and use *separate* models to estimate $\pi(x)$ and $\epsilon(x)$. By contrast, we estimate these quantities with a *single* neural model, by minimising the `SOvA` loss in (12) with $\bar{f}_{y'}(x) = w_{y'}^\top \Phi(x)$ for $y' \in \mathcal{Y} \cup \{\perp\}$. As with the standard `OvA` loss, this induces implicit information sharing between the defer logit $\bar{f}_\perp$ and the standard logits $\bar{f}_1, \ldots, \bar{f}_L$; thus, the estimates of the expert error probability will be influenced by the base model. Note that the confidence approach of Raghu et al. [27] was shown to be outperformed by the `CSS` and `OvA` losses in Mozannar and Sontag [19], Verma and Nalisnick [34]; our proposed threshold correction further achieves competitive or superior performance to these losses (§5).

Verma and Nalisnick [34, Appendix E] considered a post-hoc scheme for the setting where there is a hard constraint on the expert cost. Here, one trains a probabilistic model $\bar{p}$ using the `OvA` loss, and defers on the samples with the largest values of $\bar{p}_\perp(x) - \max_{y' \in \mathcal{Y}} \bar{p}_{y'}(x)$. While derived for a different setting, this is similar to (11), as varying the cost c_0 effectively changes the fraction of samples that are deferred. Note however that we use the `SOvA` rather than `OvA` loss to obtain π, ϵ, which can have a notable impact on performance when L is large (see Appendix J).

Cortes et al. [9] proposed a *joint* surrogate over $r, \bar{f}$ for (13) in the special case of constant $c_{\text{exp}}(x,y) = c_0$, and a scalar scorer $\bar{f} \colon \mathcal{X} \to \mathbb{R}$ for binary labels. By contrast, the surrogate loss (14) handles multi-class $\bar{f}$, generic cost functions c_{exp}, and provides a partial surrogate over only r.

Adaptive inference and multiple experts. Post-hoc training is naturally amenable to settings where there are *multiple* experts. Concretely, suppose there are K experts with logits $\bar{f}^{(1)}(x), \ldots, \bar{f}^{(K)}(x)$, and our goal is to learn which expert to employ for a given sample. One may learn *multiple* deferral functions $\bar{f}_\perp^{(1)}(x), \ldots, \bar{f}_\perp^{(K)}(x)$ in a post-hoc manner, and choose the smallest k with $\bar{f}_\perp^{(k)}(x) < 0$.

This setting arises naturally in *adaptive inference* [13], where our goal is to reduce the *inference cost* of a learning system by allowing for variable compute across samples: intuitively, we would like to spend less computation on "easy" samples. To achieve this, one may construct several models ("experts") of increasing inference cost, and defer to a suitable expert. Such models include *early-exit*

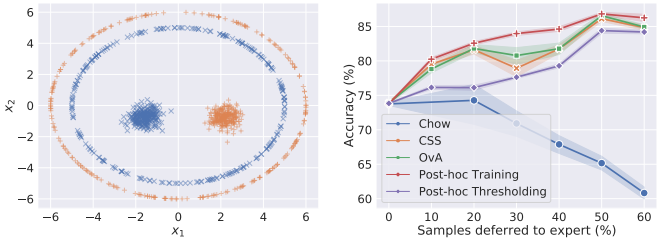

Figure 4: Results on synthetic dataset (left panel) comprising two subgroups, each with positive (+) and negative (×) samples. The subgroup of concentric circles corresponds to a latent variable $a = 1$, while the subgroup of adjacent circles corresponds to $a = 0$. The base learner is a linear classifier on the raw features $x = (x_1, x_2)$, and can query an expert that trains a classifier on *quadratic* features. We train classifiers using different losses, for varying fixed deferral cost c_0. Posthoc training is competitive or superior to the cost-sensitive softmax (`CSS`) and one-versus-all loss (`OvA`) (right panel).

classification heads at intermediate nodes of a network [24, 32, 36, 15, 29], or *sequential cascades* of models of increasing complexity [4, 30, 35]. Conceptually, adaptive inference involves the same core problem as learning to defer: at any given intermediate model, one has the option to defer prediction to the next model, at the expense of increasing the overall computation cost. Interestingly, similar ideas to post-hoc training have been explored in adaptive inference [33]. Compared to this work, we present a unified view of the learning to defer and adaptive inference problems, demonstrate limitations of existing losses for the former, and leverage generic logits $\bar{f}_1, \ldots, \bar{f}_L$ from any base classifier.

5 Experimental Results

We now present empirical results illustrating the efficacy of both our proposed posthoc estimators.

5.1 Results on Synthetic Data

We begin with a synthetic problem where $\mathcal{X} \subset \mathbb{R}^2$. We consider a distribution $\mathbb{P}(x, y) = \sum_{a \in \{0,1\}} \mathbb{P}(x, y, a)$, where $a \in \{0, 1\}$ denotes some latent subgroup. Figure 4 shows a sample draw from $\mathbb{P}(x, y)$: in a nutshell, $\mathbb{P}(x, y \mid a = 0)$ is a mixture of isotropic Gaussians, while $\mathbb{P}(x, y \mid a = 1)$ comprises two concentric circles. See Appendix H.1 for a precise specification. We consider learners that have access to a sample $\{(x_n, y_n)\}_{n=1}^N$ from $\mathbb{P}(x, y)$, and to expert model predictions $\{h_{\text{exp}}(x_n)\}_{n=1}^N$. The expert is taken to be a linear classifier that is trained on only samples with $a = 1$, using *quadratic* features: i.e., for $x = (x_1, x_2)$, we construct $\Phi(x) = (x_1, x_2, x_1^2, x_2^2, x_1 \cdot x_2)$. We set the expert deferral cost function to be $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$.

We train linear classifiers using confidence thresholding based on a constant deferral cost c_0 (`Chow`), the `CSS` and `OvA` losses, and our proposed post-hoc approaches. For each learner, we vary the fixed deferral cost c_0 from $\{0.0, 0.05, 0.10, \ldots, 0.50\}$. For each setting, we compute the fraction of samples deferred to the expert, and the overall accuracy of the base model plus expert. We conduct 250 independent trials, each with a different draw of the training sample and expert predictions.

We highlight a few trends from the results in Figure 4. First, `Chow` strongly underperforms post-hoc thresholding when many samples are deferred. This is because the latter does not consider the expert's error (which is high on $a = 1$) when deferring. Second, post-hoc training is competitive or superior to `CSS` and `OvA` at most operating points. Third, post-hoc thresholding underperforms post-hoc training. This is by construction, as it is challenging to reliably estimate $\mathbb{P}(y \mid x)$ with a linear model. However, we shall now see that on real-world benchmarks, thresholding is far more competitive.

5.2 Results on Real-World Data

We now report results on the CIFAR-10, CIFAR-100 [16], and ImageNet [10] datasets.

Learning to defer. Inspired by Mozannar and Sontag [19, Section 6.2], we consider a setting where there is a "specialist" expert: we train an expert model on only samples with label belonging to some subset $\mathcal{Y}_{\text{sub}}$. Intuitively, this model "specialises" to only samples from $\mathcal{Y}_{\text{sub}}$. We then consider a base model that is trained on all samples, but is allowed to defer to the "specialist" expert. On

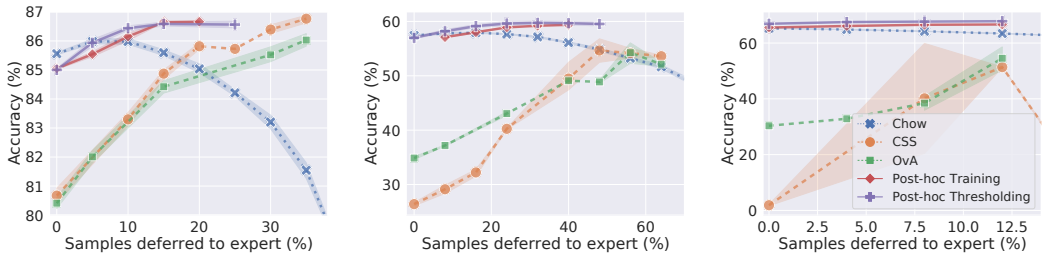

Figure 5: Results on CIFAR-10 (left), CIFAR-100 (middle), and ImageNet (right) in a learning to defer setting, where a base model is allowed to defer to a "specialist" expert. The latter is trained on only samples from the first 5 classes on CIFAR-10, the 50 classes from the first 10 "coarse labels" on CIFAR-100, and only classes corresponding to the "dog" synset on ImageNet. Our posthoc schemes offers gains over the existing cost-sensitive softmax (`CSS`) and one-versus-all (`OvA`) at most operating points, particularly when only a few samples are deferred to the expert (corresponding to fixed deferral cost $c_0 \gg 0$).

CIFAR-10 and CIFAR-100, we use a ResNet-8 base model and ResNet-56 expert; on ImageNet, we use a MobileNet-v2 base model and an EfficientNet-B0 expert. The label subset $\mathcal{Y}_{\text{sub}}$ comprises the first 5 labels on CIFAR-10, the 50 labels corresponding to the first 10 "coarse labels" on CIFAR-100, and all labels corresponding to the "dog" synset on ImageNet.

We consider a cost $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$ of deferring to the expert model. Per Figures 2 and 3, we apply posthoc threshold correction on top of the `SOvA` solution with $c_0 = 0$, and posthoc training on top of the `CSS` solution with $c_0 = 0$. Figure 5 compares all approaches as we vary the fixed cost c_0, reporting the fraction of samples deferred to the expert, and the overall accuracy of the base model plus expert. Our posthoc estimators offer gains over the existing approaches at most operating points. We particularly see gains in the regime where only a small fraction of samples are deferred to the expert, which corresponds to larger values of c_0. Between the two post-hoc methods, we see that the threshold correction approach tends to have a (slight) edge. This suggests that on these benchmarks, it is possible to obtain sufficiently reliable estimates of $\mathbb{P}(y \mid x)$ and $\mathbb{P}(y \neq h_{\text{exp}}(x))$.

We reiterate that when c_0 is large, the existing `CSS` and `OvA` losses tend to degrade owing to underfitting. Indeed, their performance in this regime is significantly lower than simply using the student model trained with the softmax cross-entropy (corresponding to `Chow` with 0% of samples deferred). Note also that in comparison to the synthetic dataset, here L is larger and thus further exacerbates underfitting. Finally, as in the synthetic dataset, `Chow` strongly underperforms when a large fraction of samples are deferred to the expert; this is because it does not take into account the expert's error rate, and thus potentially defers when the expert performs akin to random guessing.

Adaptive inference. We conclude with results on the related problem of adaptive inference. We consider a setting where there is a cascade of two models of differing inference costs, as measured in floating point operations (*FLOPs*), and we seek to learn a deferral function that intelligently forwards a small subset of samples to the latter model. This allows for an inference scheme with a favourable accuracy-compute tradeoff. Compared to the previous section, we now train the experts on *all* labels.

We report results on CIFAR-100 and ImageNet, with the former using ResNet-8 and ResNet-56 models, and the latter using a MobileNet-v2 and EfficientNet model. As before, we consider the deferral cost function $c_{\text{exp}}(x, y) = c_0 + \mathbf{1}(y \neq h_{\text{exp}}(x))$. Figure 6 compares the FLOPs versus the overall accuracy of the base model plus expert. Here, the `CSS` and `OvA` methods again degrade in the low FLOPs (high c_0) regime. However, the `Chow` baseline is more competitive, since the expert models tend to be highly accurate on all samples. Our posthoc estimators are seen to strongly outperform `CSS` and `OvA`, while being competitive with or slightly superior to `Chow`.

6 Discussion and Future Work

While post-hoc training resolves issues identified with previous joint surrogate losses, it is not without limitation: it relies crucially on reliable base classifiers from the first phase. As formalised in Theorem 2, any degradation in performance for the latter is also transferred to the final post-hoc predictor. As this may be further compounded with estimation errors in the second phase, it is natural to ask whether there may be *other* joint surrogates that do not exhibit underfitting. One

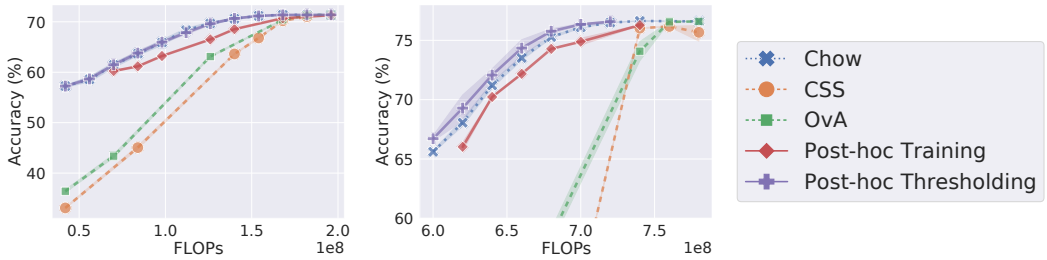

Figure 6: Results on CIFAR-100 (left), and ImageNet (right) in an adaptive inference setting, where a computationally cheap base model is allowed to defer to a more expensive expert. Our posthoc schemes offers gains over the existing cost-sensitive softmax (`CSS`) and one-versus-all (`OvA`) at most operating points, particularly in the low-FLOPs regime (corresponding to fixed deferral cost $c_0 \gg 0$).

candidate is the cost-sensitive loss of Lee et al. [17], which is Fisher-consistent and is suitable when $c_0 > 0$. This loss imposes the constraint that $\sum_{\hat{y} \in \mathcal{Y} \cup \{\perp\}} \bar{f}_{\hat{y}}(x) = 0$, and takes the form $\ell(x, y, \bar{f}(x)) = \sum_{y' \in \mathcal{Y} - \{y\}} \left[\frac{1}{L-1} + \bar{f}_{y'}(x)\right]_+ + c_{\exp}(x, y) \cdot \left[\frac{1}{L-1} + \bar{f}_{\perp}(x)\right]_+$. One may show that under this loss, the gap between the Bayes-optial scores for the highest scoring label and any competing label is $\mathcal{O}\left(1 + \frac{1}{L}\right)$, and thus is not adversely affected as L increases. As a qualifying comment, the loss poses challenges due to being non-differentiable, and requiring constraints on the logits. Nonetheless, further study of the viability of this loss would be worthwhile.

References

[1] Gagan Bansal, Besmira Nushi, Ece Kamar, Eric Horvitz, and Daniel S. Weld. Is the most accurate AI the best teammate? optimizing AI for teamwork. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 11405–11414. AAAI Press, 2021.

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

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

[4] Tolga Bolukbasi, Joseph Wang, Ofer Dekel, and Venkatesh Saligrama. Adaptive neural networks for fast test-time prediction. In *International Conference on Machine Learning*, 2017.

[5] Andreas Buja, Werner Stuetzle, and Yi Shen. Loss functions for binary class probability estimation and classification: Structure and applications. Technical report, UPenn, 2005.

[6] Nontawat Charoenphakdee, Zhenghang Cui, Yivan Zhang, and Masashi Sugiyama. Classification with rejection based on cost-sensitive classification. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1507–1517. PMLR, 18–24 Jul 2021.

[7] Yu-Ting Chou, Gang Niu, Hsuan-Tien Lin, and Masashi Sugiyama. Unbiased risk estimators can mislead: A case study of learning with complementary labels. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1929–1938. PMLR, 2020.

[8] C. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.

[9] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *ALT*, 2016.

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[11] Charles Elkan. The foundations of cost-sensitive learning. In *In Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 973–978, 2001.

[12] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[13] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *CoRR*, abs/2102.04906, 2021.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[15] Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Weinberger. Multi-scale dense networks for resource efficient image classification. In *International Conference on Learning Representations*, 2018.

[16] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[17] Yoonkyung Lee, Yi Lin, and Grace Wahba. Multicategory support vector machines. *Journal of the American Statistical Association*, 99(465):67–81, 2004. doi: 10.1198/016214504000000098.

[18] David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 6150–6160, Red Hook, NY, USA, 2018. Curran Associates Inc.

[19] Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7076–7087. PMLR, 13–18 Jul 2020.

[20] Harikrishna Narasimhan, Harish Ramaswamy, Aadirupa Saha, and Shivani Agarwal. Consistent multiclass algorithms for complex performance measures. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2398–2407, Lille, France, 07–09 Jul 2015. PMLR.

[21] Chenri Ni, Nontawat Charoenphakdee, Junya Honda, and Masashi Sugiyama. On the calibration of multiclass classification with rejection. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 2582–2592, 2019.

[22] Nastaran Okati, Abir De, and Manuel Rodriguez. Differentiable learning under triage. *Advances in Neural Information Processing Systems*, 34, 2021.

[23] Shreyas Padhy, Zachary Nado, Jie Ren, Jeremiah Z. Liu, Jasper Snoek, and Balaji Lakshminarayanan. Revisiting one-vs-all classifiers for predictive uncertainty and out-of-distribution detection in neural networks. *CoRR*, abs/2007.05134, 2020. URL https://arxiv.org/abs/2007.05134.

[24] Priyadarshini Panda, Abhronil Sengupta, and Kaushik Roy. Conditional deep learning for energy-efficient and enhanced pattern recognition. In *Proceedings of the 2016 Conference on Design, Automation & Test in Europe*, DATE '16, page 475–480, San Jose, CA, USA, 2016. EDA Consortium. ISBN 9783981537062.

[25] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: a loss correction approach. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2233–2241, 2017.

[26] Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*, 2019.

[27] Maithra Raghu, Katy Blumer, Rory Sayres, Ziad Obermeyer, Bobby Kleinberg, Sendhil Mullainathan, and Jon Kleinberg. Direct uncertainty prediction for medical second opinions. In *International Conference on Machine Learning*, pages 5281–5290. PMLR, 2019.

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

[29] Roy Schwartz, Gabriel Stanovsky, Swabha Swayamdipta, Jesse Dodge, and Noah A. Smith. The right tool for the job: Matching model and instance complexities. In *Proc. of ACL*, 2020.

[30] Matthew Streeter. Approximation algorithms for cascading prediction models. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4752–4760. PMLR, 10–15 Jul 2018.

[31] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826, 2016.

[32] Surat Teerapittayanon, Bradley McDanel, and H. T. Kung. Branchynet: Fast inference via early exiting from deep neural networks. In *23rd International Conference on Pattern Recognition, ICPR 2016, Cancún, Mexico, December 4-8, 2016*, pages 2464–2469. IEEE, 2016.

[33] Kirill Trapeznikov and Venkatesh Saligrama. Supervised sequential classification under budget constraints. In Carlos M. Carvalho and Pradeep Ravikumar, editors, *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, pages 581–589, Scottsdale, Arizona, USA, 29 Apr–01 May 2013. PMLR.

[34] Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. *arXiv preprint arXiv:2202.03673*, 2022.

[35] Xiaofang Wang, Dan Kondratyuk, Eric Christiansen, Kris M. Kitani, Yair Movshovitz-Attias, and Elad Eban. Wisdom of committees: An overlooked approach to faster and more accurate models. In *International Conference on Learning Representations*, 2022.

[36] Xin Wang, Yujia Luo, Daniel Crankshaw, Alexey Tumanov, Fisher Yu, and Joseph E. Gonzalez. IDK cascades: Fast deep learning by learning not to overthink. In Amir Globerson and Ricardo Silva, editors, *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pages 580–590. AUAI Press, 2018.

[37] Bryan Wilder, Eric Horvitz, and Ece Kamar. Learning to complement humans. *CoRR*, abs/2005.00582, 2020.

[38] Bryan Wilder, Eric Horvitz, and Ece Kamar. Learning to complement humans. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021. ISBN 9780999241165.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]

(c) Did you discuss any potential negative societal impacts of your work? [N/A] The paper is primarily analytical in nature: it identifies mathematical and empirical limitations of existing methods, and proposes new methods for a problem where a model can query an "expert"s.

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...

   (a) Did you state the full set of assumptions of all theoretical results? [Yes]

   (b) Did you include complete proofs of all theoretical results? [Yes]

3. If you ran experiments...

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No]

   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]

   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]

   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] The newly proposed methods in this paper involve simple post-hoc operations over existing models. Thus, they do not add significant computational overhead.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

   (a) If your work uses existing assets, did you cite the creators? [Yes]

   (b) Did you mention the license of the assets? [No]

   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Supplementary material for "Post-hoc estimators for learning to defer to an expert"

A Proofs

A.1 Proof of Lemma 1

Proof. To show that our proposed loss is calibrated for the cost function $c_{\text{exp}}(x, y)$, we write out the conditional risk for the loss:

$$\begin{aligned}
\mathbb{E}_{y|x}\left[\ell(y, f(x))\right] &= \sum_{y\in[L]} \eta_y(x) \left(c_{\max} \cdot \phi(f_y(x)) + \sum_{y'\neq y} \left((c_{\max}-1)\cdot\phi(f_{y'}(x)) + \phi(-f_{y'}(x))\right)\right) \\
&\qquad + \left(c_{\max} - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\right)\cdot\phi(f_\perp(x)) + \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\cdot\phi(-f_\perp(x)) \\
&= \sum_{y\in[L]} \eta_y(x) \left(\phi(f_y(x)) + \sum_{y'\in[L]} (c_{\max}-1)\cdot\phi(f_{y'}(x)) + \sum_{y'\neq y} \phi(-f_{y'}(x))\right) \\
&\qquad + \left(c_{\max} - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\right)\cdot\phi(f_\perp(x)) + \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\cdot\phi(-f_\perp(x)) \\
&= \sum_{y\in[L]} \left(\eta_y(x)\cdot\phi(f_y(x)) + (1-\eta_y(x))\cdot\phi(-f_y(x))\right) + \sum_{y\in[L]} (c_{\max}-1)\cdot\phi(f_y(x)) \\
&\qquad + \left(c_{\max} - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\right)\cdot\phi(f_\perp(x)) + \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\cdot\phi(-f_\perp(x)) \\
&= \sum_{y\in[L]} (\eta_y(x) + c_{\max} - 1)\cdot\phi(f_y(x)) + (1-\eta_y(x))\cdot\phi(-f_y(x)) \\
&\qquad + \left(c_{\max} - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\right)\cdot\phi(f_\perp(x)) + \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]\cdot\phi(-f_\perp(x)).
\end{aligned}$$

For any $y \in \mathcal{Y}$, the optimal scorer is thus

$$\begin{aligned}
f_y^*(x) &= \operatorname*{argmin}_{v\in\mathbb{R}} a(x)\cdot\phi(v) + (1-a(x))\cdot\phi(-v) \\
a(x) &\doteq \frac{\eta_y(x) + c_{\max} - 1}{c_{\max}}.
\end{aligned}$$

Similarly, for the defer logit,

$$\begin{aligned}
f_\perp^*(x) &= \operatorname*{argmin}_{v\in\mathbb{R}} b(x)\cdot\phi(v) + (1-b(x))\cdot\phi(-v) \\
b(x) &\doteq \frac{c_{\max} - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right]}{c_{\max}}.
\end{aligned}$$

By definition of a strictly proper composite loss [28], the optimal scorer f^* then takes the form:

$$\psi(f_{y'}^*(x)) = \begin{cases} \frac{1}{c_{\max}}(\eta_{y'}(x) + c_{\max} - 1) & \text{if } y' \in [L] \\ \frac{1}{c_{\max}}\left(c_{\max} - \mathbb{E}_{y|x}\left[c(x,y)\right]\right) & \text{if } y' = \perp, \end{cases}$$

where ψ is the inverse link function associated with ϕ, and the corresponding classifier $\bar{h}^*(x) \in \operatorname{argmax}_{y'\in[L]\cup\{\perp\}} f_{y'}^*(x)$ is given by:

$$\bar{h}^*(x) = \begin{cases} \operatorname{argmax}_{y'\in[L]} \eta_{y'}(x) & \text{if } \max_{y'\in[L]} \eta_{y'}(x) < 1 - \mathbb{E}_{y|x}\left[c_{\text{exp}}(x,y)\right] \\ \perp & \text{otherwise} \end{cases},$$

as desired.

□

A.2 Proof of Theorem 2

Proof. The proof is similar to that of a regret bound in Narasimhan et al. [20, Lemma 14] for a general-cost-sensitive risk. As noted in §4.1, we denote $\eta_y(x) = \mathbb{P}(y \mid x)$ and $\eta(x) \doteq [\mathbb{P}(1 \mid x), \ldots, \mathbb{P}(L \mid$

$x)]^\top$. We will denote the expected cost of deferring to the expert using $C_{\text{exp}}(x) \doteq \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x,y)\right]$ and use $\hat{C}_{\text{exp}}(x)$ to denote an estimate of this quantity.

We then re-write the learning to defer risk in (1) in terms of $\eta(x)$ and $C_{\text{exp}}(x)$:

$$\begin{aligned} R_{\text{def}}(h) &= \mathop{\mathbb{E}}_{x}\left[\sum_{y=1}^{L} \eta_y(x)\left(\mathbf{1}(h(x)\neq y) \,+\, C_{\text{exp}}(x)\cdot \mathbf{1}(h(x)=\perp)\right)\right] \\ &= \mathop{\mathbb{E}}_{x}\left[\langle \eta(x), e^{\text{mod}}_{h(x)} + e^{\text{exp}}_{h(x)}(x)\rangle\right], \end{aligned}$$

where $\langle\cdot,\cdot\rangle$ denotes standard inner-product, $e^{\text{mod}}_{\hat{y}} = [\mathbf{1}(\hat{y}\neq 1),\ldots,\mathbf{1}(\hat{y}\neq L)]^\top \in \{0,1\}^L$ and $e^{\text{exp}}_{\hat{y}}(x) = [C_{\text{exp}}(x)\cdot\mathbf{1}(\hat{y}\neq\perp),\,\ldots,\,C_{\text{exp}}(x)\cdot\mathbf{1}(\hat{y}\neq\perp)]^\top \in \mathbb{R}^L_+$.

We also define an empirical version of the learning to defer risk, in which $\eta(x)$ is replaced by an estimate $\pi(x)$ and the cost $C_{\text{exp}}(x)$ is replaced by $\hat{C}_{\text{exp}}(x)$:

$$\hat{R}_{\text{def}}(h) = \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), e^{\text{mod}}_{h(x)} + \hat{e}^{\text{exp}}_{h(x)}(x)\rangle\right],$$

where we denote $\hat{e}^{\text{exp}}_{\hat{y}}(x) = \left[\hat{C}_{\text{exp}}(x)\cdot\mathbf{1}(\hat{y}\neq\perp),\,\ldots,\,\hat{C}_{\text{exp}}(x)\cdot\mathbf{1}(\hat{y}\neq\perp)\right]^\top \in \mathbb{R}^L_+$.

Finally, consider an empirical version of the Bayes-optimal classifier in (3):

$$\bar{h}(x) = \begin{cases} \operatorname*{argmax}_{y\in\mathcal{Y}} \pi_y(x) & \text{if } 1-\max_{y\in[L]}\pi_y(x) < \hat{C}_{\text{exp}}(x) \\ \perp & \text{else} \end{cases}.$$

From the definition of $\bar{h}$,we have that $\bar{h}(x)$ minimizes $\langle \pi(x), e^{\text{mod}}_{h(x)} + \hat{e}^{\text{exp}}_{h(x)}(x)\rangle$ for each x, and as a result, for any classifier h:

$$\hat{R}_{\text{def}}(\bar{h}) \leq \hat{R}_{\text{def}}(h). \tag{16}$$

The excess risk can then be expanded as:

$$\begin{aligned}
&R_{\text{def}}(\bar{h}) - R_{\text{def}}(\bar{h}^*) \\
&= R_{\text{def}}(\bar{h}) - \hat{R}_{\text{def}}(\bar{h}) + \hat{R}_{\text{def}}(\bar{h}) - R_{\text{def}}(\bar{h}^*) \\
&\leq R_{\text{def}}(\bar{h}) - \hat{R}_{\text{def}}(\bar{h}) + \hat{R}_{\text{def}}(\bar{h}^*) - R_{\text{def}}(\bar{h}^*) && (17)\\
&= \mathop{\mathbb{E}}_{x}\left[\langle \eta(x), e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x)\rangle\right] - \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), e^{\text{mod}}_{\bar{h}(x)} + \hat{e}^{\text{exp}}_{\bar{h}(x)}(x)\rangle\right] \\
&\qquad + \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), e^{\text{mod}}_{\bar{h}^*(x)} + \hat{e}^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] - \mathop{\mathbb{E}}_{x}\left[\langle \eta(x), e^{\text{mod}}_{\bar{h}^*(x)} + e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] \\
&= \mathop{\mathbb{E}}_{x}\left[\langle \eta(x), e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x)\rangle\right] - \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x)\rangle\right] \\
&\qquad + \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), e^{\text{mod}}_{\bar{h}^*(x)} + e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] - \mathop{\mathbb{E}}_{x}\left[\langle \eta(x), e^{\text{mod}}_{\bar{h}^*(x)} + e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] \\
&\qquad + \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), e^{\text{exp}}_{\bar{h}(x)}(x) - \hat{e}^{\text{exp}}_{\bar{h}(x)}(x)\rangle\right] + \mathop{\mathbb{E}}_{x}\left[\langle \pi(x), \hat{e}^{\text{exp}}_{\bar{h}^*(x)}(x) - e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] \\
&\leq \mathop{\mathbb{E}}_{x}\left[\langle \eta(x)-\pi(x), e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x) - e^{\text{mod}}_{\bar{h}^*(x)} - e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] \\
&\qquad + \mathop{\mathbb{E}}_{x}\left[\|\pi(x)\|_1 \cdot \|e^{\text{exp}}_{\bar{h}(x)}(x) - \hat{e}^{\text{exp}}_{\bar{h}(x)}(x)\|_\infty\right] + \mathop{\mathbb{E}}_{x}\left[\|\pi(x)\|_1 \cdot \|e^{\text{exp}}_{\bar{h}^*(x)}(x) - \hat{e}^{\text{exp}}_{\bar{h}^*(x)}(x)\|_\infty\right] && (18)\\
&\leq \mathop{\mathbb{E}}_{x}\left[\langle \eta(x)-\pi(x), e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x) - e^{\text{mod}}_{\bar{h}^*(x)} - e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\right] + 2\cdot\mathop{\mathbb{E}}_{x}\left[\left|C_{\text{exp}}(x) - \hat{C}_{\text{exp}}(x)\right|\right] && (19)\\
&\leq \mathop{\mathbb{E}}_{x}\left[\|\eta(x)-\pi(x)\|_1 \cdot \|e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x) - e^{\text{mod}}_{\bar{h}^*(x)} - e^{\text{exp}}_{\bar{h}^*(x)}(x)\|_\infty + 2\left|C_{\text{exp}}(x) - \hat{C}_{\text{exp}}(x)\right|\right], && (20)
\end{aligned}$$

where the inequality in (17) follows from (16) with $h = h^*$; the inequality in (18) follows from Hölder's inequality; and the inequality in (19) uses the fact that $\|\pi(x)\|_1 = 1$ and $\mathbf{1}(\bar{h}(x) \neq \perp) \leq 1$. The last step uses Hölder's inequality.

Further expanding the inner ∞-norm term in (20), we have for any x:

$$\begin{aligned}
&\|e^{\text{mod}}_{\bar{h}(x)} + e^{\text{exp}}_{\bar{h}(x)}(x) - e^{\text{mod}}_{\bar{h}^*(x)} - e^{\text{exp}}_{\bar{h}^*(x)}(x)\rangle\|_\infty \\
&\leq \max_{y \in [L]} \left|\mathbf{1}(\bar{h}(x) \neq y) + C_{\text{exp}}(x) \cdot \mathbf{1}(\bar{h}(x) = \perp) \; - \; \mathbf{1}(\bar{h}^*(x) \neq y) - C_{\text{exp}}(x) \cdot \mathbf{1}(\bar{h}^*(x) = \perp)\right| \\
&\leq \max\{0,\, 1,\, 1 - C_{\text{exp}}(x),\, C_{\text{exp}}(x) - 1,\, C_{\text{exp}}(x)\} \\
&\leq \max\{1, C_{\text{exp}}(x)\} \\
&\leq c_{\max},
\end{aligned}$$

where we use the fact that $C_{\text{exp}}(x) \leq c_{\max}$ and $c_{\max} \geq 1$. Substituting this back in (20), we get:

$$R_{\text{def}}(\bar{h}) - R_{\text{def}}(\bar{h}^*) \leq c_{\max} \cdot \mathop{\mathbb{E}}_{x}\left[\|\eta(x) - \pi(x)\|_1\right] \; + \; 2\mathop{\mathbb{E}}_{x}\left[\left|C_{\text{exp}}(x) - \hat{C}_{\text{exp}}(x)\right|\right].$$

Setting $C_{\text{exp}}(x) = \mathbb{P}(y \neq h_{\text{exp}}(x)) + c_0$ and $\hat{C}_{\text{exp}}(x) = \epsilon(x) + c_0$ completes the proof. □

A.3 Proof of Lemma 3

Proof. The conditional risk for the loss is

$$\begin{aligned}
\mathop{\mathbb{E}}_{y|x}\left[\ell(x, y, r(x))\right] &= \mathop{\mathbb{E}}_{y|x}\left[c_{\text{mod}}(x, y)\right] \cdot \phi(r(x)) + \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right] \cdot \phi(-r(x)) \\
&\propto a(x) \cdot \phi(r(x)) + (1 - a(x)) \cdot \phi(-r(x)),
\end{aligned}$$

where

$$a(x) \doteq \frac{\mathop{\mathbb{E}}_{y|x}\left[c_{\text{mod}}(x, y)\right]}{\mathop{\mathbb{E}}_{y|x}\left[c_{\text{mod}}(x, y)\right] + \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]}.$$

Consequently, since ϕ is strictly proper composite, the Bayes-optimal must satisfy [28] $\psi(r^*(x)) = a(x)$. Note that $a(x) > 0.5 \iff \mathop{\mathbb{E}}_{y|x}\left[c_{\text{mod}}(x, y)\right] > \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]$, i.e., the cost of querying the model exceeds that of querying the expert. □

A.4 Proof of Theorem 4

Let $\bar{h}^*$ denote the Bayes-optimal classifier that minimizes the L2D risk in (13). We can re-write $\bar{h}^*$ in terms of a base classifier and a rejector:

$$\bar{h}^*(x) = \begin{cases} h^*(x) & \text{if } r^*(x) < 0 \\ \perp & \text{else.} \end{cases},$$

where $h^*(x) \in \operatorname*{argmax}_{y \in \mathcal{Y}} \mathbb{P}(y \mid x)$ and $\text{sign}(r^*(x)) = \text{sign}\left(1 - \max_{y \in \mathcal{Y}} \mathbb{P}(y \mid x) - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]\right)$. Notice that h^* is also the Bayes-optimal classifier for the standard misclassification error $R_{\text{err}}(h)$.

We will find it useful to first define a normalized version of the L2D risk:

$$\tilde{R}_{\text{def}}(h, r) = \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L(x, h, r)\right],$$

where

$$L(x, h, r) = \mathop{\mathbb{E}}_{y|x}\left[\mathbf{1}(y \neq h(x))\right] \cdot \mathbf{1}(r(x) < 0) + \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right] \cdot \mathbf{1}(r(x) > 0)$$

and

$$Z(x) = \mathop{\mathbb{E}}_{y|x}\left[\mathbf{1}(y \neq h(x)) + c_{\text{exp}}(x, y)\right].$$

Similarly, we define the normalized partial surrogate risk as:

$$\tilde{R}^{\phi}_{\text{def}}(h, r) = \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L^{\phi}(x, h, r)\right].$$

where

$$L^{\phi}(x,h,r) = \mathop{\mathbb{E}}_{y|x}\left[\mathbf{1}(y \neq h(x))\right] \cdot \phi(r(x)) + \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x,y)\right] \cdot \phi(-r(x)).$$

We will also find the following lemmas useful for proving the theorem.

Lemma 5. *Suppose $c_{\text{exp}}(x,y) \in [c_{\min}, c_{\max}]$ for some $c_{\min} > 0$. Then for any fixed base classifier $\hat{h}$, the normalized excess risks can be bounded in terms of the unnormalized excess risks as follows:*

$$\tilde{R}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} \tilde{R}_{\text{def}}(\hat{h},r) \geq \frac{1}{1+c_{\max}} \cdot \left(R_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h},r)\right),$$

$$\tilde{R}^{\phi}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} \tilde{R}^{\phi}_{\text{def}}(\hat{h},r) \leq \frac{1}{c_{\min}} \cdot \left(R^{\phi}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h},r)\right).$$

Proof. Expanding the normalized excess misclassification risk, we have:

$$\begin{aligned}
\tilde{R}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} \tilde{R}_{\text{def}}(\hat{h},r) &= \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L(x,\hat{h},\hat{r})\right] - \min_{r:\mathcal{X}\to\mathbb{R}} \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L(x,\hat{h},r)\right] \\
&= \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L(x,\hat{h},\hat{r})\right] - \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot \min_{z\in\mathbb{R}} L(x,\hat{h},z)\right] \\
&= \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot \left(L(x,\hat{h},\hat{r}) - \min_{z\in\mathbb{R}} L(x,\hat{h},z)\right)\right] \\
&\geq \mathbb{E}_x\left[\frac{1}{1+c_{\max}} \cdot \left(L(x,\hat{h},\hat{r}) - \min_{z\in\mathbb{R}} L(x,\hat{h},z)\right)\right] \\
&= \frac{1}{1+c_{\max}} \cdot \left(R_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h},r)\right),
\end{aligned}$$

where in the second step, we use the fact that the optimal rejector is point-wise optimal, and in the penultimate step, we use $Z(x) = \mathop{\mathbb{E}}_{y|x}\left[\mathbf{1}(y \neq h(x)) + c_{\text{exp}}(x,y)\right] \leq 1 + c_{\max}$.

Similarly, expanding the normalized excess surrogate risk, we have:

$$\begin{aligned}
\tilde{R}^{\phi}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} \tilde{R}^{\phi}_{\text{def}}(\hat{h},r) &= \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L^{\phi}(x,\hat{h},\hat{r})\right] - \min_{r:\mathcal{X}\to\mathbb{R}} \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot L^{\phi}(x,\hat{h},r)\right] \\
&= \mathbb{E}_x\left[\frac{1}{Z(x)} \cdot \left(L(x,\hat{h},\hat{r}) - \min_{z\in\mathbb{R}} L(x,\hat{h},z)\right)\right] \\
&\leq \frac{1}{c_{\min}} \cdot \left(R^{\phi}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h},r)\right),
\end{aligned}$$

where we have used the fact $Z(x) \geq c_{\min}$. □

The next lemma builds on Lemma 5 to provide a excess risk bound for the rejector.

Lemma 6. *Suppose ϕ is classification calibrated and $c_{\text{exp}}(x,y) \in [c_{\min}, c_{\max}]$ for some $c_{\min} > 0$. Then for any fixed base classifier $\hat{h}$,*

$$R_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h},r) \leq \Psi\left(R^{\phi}_{\text{def}}(\hat{h},\hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h},r)\right),$$

for some increasing function $\Psi : \mathbb{R}_+ \to \mathbb{R}_+$ with $\Psi(0) = 0$.

Proof. Using the fact that ϕ is classification calibrated, one can directly apply the surrogate excess risk bounds from Bartlett et al. [3] to the normalized surrogate risk. Specifically, we first note that for the label distribution $q(x) = \frac{1}{Z(x)} \cdot \mathop{\mathbb{E}}_{y|x}\left[\mathbf{1}(y \neq h(x)\right]$, the normalized risks can be written as:

$$\tilde{R}_{\text{def}}(h,r) = q(x) \cdot \phi(r(x)) + (1 - q(x)) \cdot \phi(-r(x));$$

$$\tilde{R}^{\phi}_{\text{def}}(h, r) = q(x) \cdot \mathbf{1}(r(x) < 0) + (1 - q(x)) \cdot \mathbf{1}(r(x) > 0).$$

We can then bound the excess normalized surrogate risk in terms of the excess normalized misclassification risk:

$$\tilde{R}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} \tilde{R}_{\text{def}}(\hat{h}, r) \leq \zeta\left(\tilde{R}^{\phi}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} \tilde{R}^{\phi}_{\text{def}}(\hat{h}, r)\right),$$

for some increasing function $\zeta : \mathbb{R}_+ \to \mathbb{R}_+$ with $\zeta(0) = 0$.

We then apply Lemma 5 to lower bound the LHS and upper bound the RHS:

$$\frac{1}{1 + c_{\max}} \cdot \left(R_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h}, r)\right) \leq \zeta\left(\frac{1}{c_{\min}} \cdot \left(R^{\phi}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h}, r)\right)\right)$$

from which we have:

$$\left(R_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h}, r)\right) \leq (1 + c_{\max}) \cdot \zeta\left(\frac{1}{c_{\min}} \cdot \left(R^{\phi}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h}, r)\right)\right).$$

Setting $\Psi(z) = (1 + c_{\max}) \cdot \zeta\left(\frac{1}{c_{\min}} \cdot z\right)$ completes the proof. □

We are now ready to prove Theorem 4.

Proof of Theorem 4. We have:

$$R_{\text{def}}(\hat{h}, \hat{r}) - R_{\text{def}}(h^*, r^*) = \underbrace{R_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h}, r)}_{\text{term}_1} + \underbrace{\min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h}, r) - R_{\text{def}}(h^*, r^*)}_{\text{term}_2}. \tag{21}$$

We first bound the second term below:

$$\begin{aligned}
\min_{r:\mathcal{X}\to\mathbb{R}} R_{\text{def}}(\hat{h}, r) - R_{\text{def}}(h^*, r^*) &\leq R_{\text{def}}(\hat{h}, r^*) - R_{\text{def}}(h^*, r^*) \\
&\leq \mathbb{E}\left[\mathbf{1}(r^*(x) < 0) \cdot \left(\mathbf{1}(y \neq \hat{h}(x)) - \mathbf{1}(y \neq h^*(x))\right)\right] \\
&= \mathbb{E}_x\left[\mathbf{1}(r^*(x) < 0) \cdot \left(\mathbb{E}_{y|x}\left[\mathbf{1}(y \neq \hat{h}(x))\right] - \mathbb{E}_{y|x}\left[\mathbf{1}(y \neq h^*(x))\right]\right)\right] \\
&\leq \mathbb{E}_x\left[\mathbb{E}_{y|x}\left[\mathbf{1}(y \neq \hat{h}(x))\right] - \mathbb{E}_{y|x}\left[\mathbf{1}(y \neq h^*(x))\right]\right].
\end{aligned}$$

where the last step uses the fact that $\mathbf{1}(r^*(x) < 0) \leq 1$ and by definition of h^*, $\mathbb{E}_{y|x}\left[\mathbf{1}(y \neq h(x))\right] \geq \mathbb{E}_{y|x}\left[\mathbf{1}(y \neq h^*(x))\right]$, for all classifiers h.

Substituting this back into (21) to bound the second term, and applying Lemma 6 to bound the first term, we have:

$$\begin{aligned}
&R_{\text{def}}(\hat{h}, \hat{r}) - R_{\text{def}}(h^*, r^*) \\
&\leq \Psi\left(R^{\phi}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h}, r)\right) + \mathbb{E}\left[\mathbf{1}(y \neq \hat{h}(x))\right] - \mathbb{E}\left[\mathbf{1}(y \neq h^*(x))\right] \\
&= \Psi\left(R^{\phi}_{\text{def}}(\hat{h}, \hat{r}) - \min_{r:\mathcal{X}\to\mathbb{R}} R^{\phi}_{\text{def}}(\hat{h}, r)\right) + \mathbb{E}\left[\mathbf{1}(y \neq \hat{h}(x))\right] - \min_{h:\mathcal{X}\to[L]} \mathbb{E}\left[\mathbf{1}(y \neq h^*(x))\right],
\end{aligned}$$

where Ψ is some increasing function with $\Psi(0) = 0$, and in the second inequality, we use the fact that h^* is the Bayes-optimal classifier for the misclassification error. □

B Learning to defer as cost-sensitive learning

For a classifier $h\colon \mathcal{X} \to \mathcal{Y}$, the misclassification error $R(h) = \mathbb{P}(y \neq h(x))$ assumes that all mispredictions are equally undesirable, and that the classifier can only output labels from $\mathcal{Y}$. More generally, *cost-sensitive classification* [11] seeks $h\colon \mathcal{X} \to \hat{\mathcal{Y}}$ that minimises

$$R_{\text{cs}}(h) = \mathbb{E}_{(x,y)}\left[\alpha(x, y, h(x))\right],$$

for *cost function* $\alpha\colon \mathcal{X}\times\mathcal{Y}\times\hat{\mathcal{Y}} \to \mathbb{R}_+$ and *prediction space* $\hat{\mathcal{Y}}$. When $\hat{\mathcal{Y}} = \mathcal{Y}$ and $\alpha(x, y, \hat{y}) = \mathbf{1}(y \neq \hat{y})$, $R_{\text{cs}}(h) = R(h)$. The Bayes-optimal cost-sensitive classifier is

$$(\forall x \in \mathcal{X})\, h^*(x) = \operatorname*{argmin}_{\hat{y}\in\hat{\mathcal{Y}}} \mathop{\mathbb{E}}_{y|x} \left[\alpha(x, y, \hat{y})\right]. \tag{22}$$

In the *learning to defer* (*L2D*) problem [18], one seeks a classifier that can either make a standard prediction in $\mathcal{Y}$, or *defer* its prediction to an *expert* model $h_{\text{exp}}\colon \mathcal{X} \to \mathcal{Y}$. Invoking an expert carries an associated sample-dependent *cost* $c_{\text{exp}}(x, y) > 0$, so as to prevent the classifier from deferring on all samples. Concretely, let $\hat{\mathcal{Y}} = \mathcal{Y} \cup \{\perp\}$, where $\perp$ denotes the "defer" option. Consider a classifier $\bar{h}\colon \mathcal{X} \to \hat{\mathcal{Y}}$ equipped with a "defer" option $\perp$. Our goal is then to minimise

$$R_{\text{def}}(\bar{h}) = \mathbb{E}_{(x,y)}\left[\alpha(x, y, \bar{h}(x))\right] \text{ for } \alpha(x, y, \hat{y}) = \begin{cases} \mathbf{1}(y \neq \hat{y}) & \text{if } \hat{y} \neq \perp \\ c_{\text{exp}}(x, y) & \text{else.} \end{cases} \tag{23}$$

C On the Bayes-optimal deferral logit and probability

We now detail the form of the Bayes-optimal defer logit $\bar{f}^*_\perp(x)$ for the cost-sensitive softmax cross-entropy, assuming that all other logits $\bar{f}_1(x), \ldots, \bar{f}_L(x)$ are fixed, and chosen arbitrarily. This is slightly more general than (5), which considered the joint Bayes-optimal solution for *all* logits. This analysis highlights two points: first, even when the other logits $\bar{f}_1(x), \ldots, \bar{f}_L(x)$ are chosen arbitrarily, there is no dependence of the optimal defer *probability* $\bar{p}^*_\perp(x)$ on the logits $\bar{f}_1(x), \ldots, \bar{f}_L(x)$. Second, as a qualifier to the above, there is however a dependence of the optimal defer *logit* $\bar{f}^*_\perp(x)$ on the logits $\bar{f}_1(x), \ldots, \bar{f}_L(x)$.

From (4), the cost-sensitive softmax cross-entropy has conditional risk

$$\begin{aligned}
L(x, \bar{f}(x)) &= \mathop{\mathbb{E}}_{y|x}\left[\ell_{\text{CSS}}(x, y, \bar{f}(x))\right] \\
&= \mathop{\mathbb{E}}_{y|x}\left[\sum_{y'\in\mathcal{Y}} (c_{\max} - \mathbf{1}(y \neq y')) \cdot -\log \bar{p}_{y'}(x)\right] + \mathop{\mathbb{E}}_{y|x}\left[(c_{\max} - c_{\text{exp}}(x, y)) \cdot -\log \bar{p}_\perp(x)\right] \\
&= \sum_{y'\in\mathcal{Y}} \mathop{\mathbb{E}}_{y|x}\left[(c_{\max} - \mathbf{1}(y \neq y'))\right] \cdot -\log \bar{p}_{y'}(x) + \mathop{\mathbb{E}}_{y|x}\left[(c_{\max} - c_{\text{exp}}(x, y))\right] \cdot -\log \bar{p}_\perp(x) \\
&= \sum_{y'\in\mathcal{Y}} (c_{\max} - (1 - \mathbb{P}(y' \mid x))) \cdot -\log \bar{p}_{y'}(x) + (c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]) \cdot -\log \bar{p}_\perp(x) \\
&= \sum_{y'\in\mathcal{Y}} (c_{\max} - (1 - \mathbb{P}(y' \mid x))) \cdot -\log \frac{\exp(\bar{f}_{y'}(x))}{\bar{Z}(x)} \\
&\qquad + (c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]) \cdot -\log \frac{\exp(\bar{f}_\perp(x))}{\bar{Z}(x)},
\end{aligned}$$

where $\bar{Z}(x) \doteq Z(x) + \exp(\bar{f}_\perp(x))$ and $Z(x) \doteq \sum_{y'\in\mathcal{Y}} \exp(\bar{f}_{y'}(x))$. Thus, the terms involving $\bar{f}_\perp(x)$ are

$$\begin{aligned}
&\bar{L}(x, \bar{f}_\perp(x)) \\
&= \left[\sum_{y'\in\mathcal{Y}} (c_{\max} - (1 - \mathbb{P}(y' \mid x)))\right] \cdot \log\left[\bar{Z}(x)\right] + (c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]) \cdot -\log \frac{\exp(\bar{f}_\perp(x))}{\bar{Z}(x)} \\
&= \left[L \cdot (c_{\max} - 1) + 1\right] \cdot \log\left[\bar{Z}(x)\right] + (c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]) \cdot -\log \frac{\exp(\bar{f}_\perp(x))}{\bar{Z}(x)} \\
&= \left[L \cdot (c_{\max} - 1) + 1 + c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]\right] \cdot \log\left[Z(x) + \exp(\bar{f}_\perp(x))\right] \\
&\qquad - (c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\text{exp}}(x, y)\right]) \cdot \bar{f}_\perp(x).
\end{aligned}$$

At optimality, for any choice of $\bar{f}_1, \ldots, \bar{f}_L$, we thus have

$$\left[L \cdot (c_{\max} - 1) + 1 + c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]\right] \cdot \frac{\exp(\bar{f}^*_\perp(x))}{Z(x) + \exp(\bar{f}^*_\perp(x))} = c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]$$
$$\iff \bar{p}^*_\perp(x) = \frac{c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]}{L \cdot (c_{\max} - 1) + 1 + c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]}.$$

Thus, for *any* $\bar{f}_1, \ldots, \bar{f}_L$, the *probability* assigned to the deferral label will be independent of these logits. Note however that the deferral *logit* $\bar{f}^*_\perp$ itself will necessarily depend on these logits: intuitively, it serves to re-normalise the softmax probabilities of these logits to ensure $\bar{p}_\perp$ takes the desired form. Indeed, some simple algebra reveals that

$$\begin{aligned}\bar{f}^*_\perp(x) &= \log Z(x) + \log \frac{c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]}{L \cdot (c_{\max} - 1) + 1} \\ &= \log \left[\sum_{y' \in \mathcal{Y}} \exp(\bar{f}_{y'}(x))\right] + \log \frac{c_{\max} - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]}{L \cdot (c_{\max} - 1) + 1}. \end{aligned} \tag{24}$$

Thus, the Bayes-optimal defer logit has a non-linear dependence on the logits $\bar{f}_1, \ldots, \bar{f}_L$. We further remark that when the model class is $\bar{f}$ is of insufficiently high capacity, it may not be feasible to achieve this Bayes-optimal solution (even though $\bar{p}^*_\perp(x)$ is relatively simple). For example, suppose we parameterise $\bar{f}_{y'}(x) = w_{y'}^\top \Phi(x) + b_{y'}$. Then, it is not possible to express $\bar{f}^*_\perp(x)$ from (24) in the form $w_\perp^\top \Phi(x) + b_\perp$, since $\log \left[\sum_{y' \in \mathcal{Y}} \exp(\bar{f}_{y'}(x))\right]$ is non-linear in $\Phi(x)$.

As a final remark, we note that the picture changes when considering the one-versus-all loss: here, regardless of the choice of $\bar{f}_1, \ldots, \bar{f}_L$ the Bayes-optimal solution $\bar{f}^*_\perp$ will be $\psi^{-1}\left(1 - \mathop{\mathbb{E}}_{y|x}\left[c_{\exp}(x, y)\right]\right)$. This is owing to the decoupled nature of the loss; consequently, there is no influence of the other logits on $\bar{f}^*_\perp$.

D General one-versus-all loss

We now present a more general form of the one-versus-all loss, where we allow for a generic cost function $\alpha(x, y, \bar{h}(x))$ following (23). Consider a loss of the form

$$\ell(x, y, \bar{f}(x)) = \sum_{y' \in \hat{\mathcal{Y}}} \left[a_{yy'}(x) \cdot \phi(f_y(x)) + b_{yy'}(x) \cdot \phi(f_{y'}(x))\right] \tag{25}$$

for $a_{yy'}(x), b_{yy'}(x) \in \mathbb{R}$. The conditional risk is

$$\mathop{\mathbb{E}}_{y|x}\left[\ell(x, y, \bar{f}(x))\right] = \sum_{y' \in \hat{\mathcal{Y}}} \left[A_{y'}(x) \cdot \phi(f_y(x)) + B_{y'}(x) \cdot \phi(f_{y'}(x))\right],$$

where $A_{y'}(x) \doteq \mathop{\mathbb{E}}_{y|x}\left[a_{yy'}(x)\right]$ and $B_{y'}(x) \doteq \mathop{\mathbb{E}}_{y|x}\left[b_{yy'}(x)\right]$. The Bayes-optimal scorer for a strictly proper composite loss with inverse link ψ is thus

$$(\forall x \in \mathcal{X})\, \psi(f^*_{y'}(x)) = \frac{A_{y'}(x)}{B_{y'}(x) + B_{y'}(x)} \tag{26}$$

Now observe that the Bayes-optimal solution for the cost-sensitive problem (22) is expressible as

$$\begin{aligned} h^*(x) &= \mathop{\mathrm{argmax}}_{\hat{y} \in \hat{\mathcal{Y}}} f^*_{y'}(x) \\ f^*_{y'}(x) &= 1 - \frac{\mathop{\mathbb{E}}_{y|x}\left[\alpha(x, y, y')\right]}{\alpha_{\max}}, \end{aligned}$$

where $\alpha_{\max}$ is an upper bound on $\alpha(x, y, y')$. Thus, if we set

$$a_{yy'}(x) = 1 - \frac{\alpha(x, y, y')}{\alpha_{\max}}$$
$$b_{yy'}(x) = \frac{\alpha(x, y, y')}{\alpha_{\max}},$$

the Bayes-optimal scorer in (26) will precisely yield $\psi(f^*_{y'}(x)) = 1 - \frac{\mathbb{E}_{y|x}\left[\alpha(x,y,y')\right]}{\alpha_{\max}}$.

As a sanity check, when α corresponds to the learning to defer risk, we have

$$a_{yy'}(x) = \begin{cases} \frac{c_{\max} - \mathbf{1}(y \neq y')}{c_{\max}} & \text{if } y' \neq \perp \\ \frac{c_{\max} - c_{\exp}(x,y)}{c_{\max}} & \text{if } y' = \perp \end{cases}$$
$$b_{yy'}(x) = \begin{cases} \frac{\mathbf{1}(y \neq y')}{c_{\max}} & \text{if } y' \neq \perp \\ \frac{c_{\exp}(x,y)}{c_{\max}} & \text{if } y' = \perp, \end{cases}$$

which is seen to yield the `OvA` loss in (10).

E Summary of losses for learning to defer

Table 1 summarises various approaches for learning to defer. Each method employs a base loss $\ell(y, x, f(x); c_{\text{base}})$ to train the base model with a base deferral cost c_{base}, and optionally optimizes a rejector model using the rejector loss provided in (14) with the user-specified fixed deferral cost c_0:

$$\ell_{\text{rej}}(x, y, r(x); c_0) = c_{\text{mod}}(x, y) \cdot \phi(r(x)) + (c_0 + \mathbf{1}\left(y \neq h_{\text{exp}}(x)\right)) \cdot \phi(-r(x)).$$

Prior methods by Mozannar and Sontag [19] and Verma and Nalisnick [34] include the user-specified cost c_0 in the base loss, i.e., set $c_{\text{base}} = c_0$, and do not train an explicit rejector. The proposed methods in this paper set $c_{\text{base}} = 0$ in the base loss, and include c_0 either as a part of a *post-hoc thresholding* step, or as a part of the rejector loss in an explicit *post-hoc training* step.

We also include a variant where one minimizes the standard softmax cross-entropy loss $\ell_{\text{CE}}(y, f(x)) = \log\left[\sum_{y' \in \mathcal{Y}} e^{f_{y'}(x)}\right] - f_y(x)$ to train the base model, and trains the rejector in a post-hoc manner.

F Parameterising the post-hoc rejector

In constructing the rejector model $r : \mathcal{X} \to \mathbb{R}$, it may be beneficial to paramterize it in terms of the base model's probabilities. For example, if the base loss is the `CSS` loss in (8), then we may want to use a rejector of the form:

$$r(x) = g_{\perp}(x) + \frac{\bar{p}_{\perp}(x)}{1 - \bar{p}_{\perp}(x)} - \max_{y \in [L]} \frac{\bar{p}_y(x)}{1 - \bar{p}_{\perp}(x)},$$

where $\bar{p}$ denotes a softmax transformation of the $L + 1$ logits from the fixed base model $\bar{f}$, and $g_{\perp} : \mathcal{X} \to \mathbb{R}$ is a example-dependent bias term that we explicitly train.

Notice that if the first L probabilities from the base model accurately estimate the underlying conditional-class probabilities, i.e., $\bar{p}_y(x) \propto \mathbb{P}(y \mid x), \forall y \in [L]$, and the reject probability is an accurate estimate of the expert's error, i.e., $\bar{p}_{\perp}(x) \propto \mathbb{P}(y \neq h_{\text{exp}}(x))$, then $r(x) > 0$ matches the Bayes-optimal deferral decision when $g_{\perp}(x) = c_0$. In practice, the base model may not accurately estimate either $\mathbb{P}(y \mid x)$ or the expert's error, and therefore the additional term $g_{\perp}(x)$ that we train can be seen as correcting errors in the base model's estimates.

Table 1 also presents the rejector parameterisation for other base losses, such as the `OvA` loss and the `SOvA` loss. In each case, the form of the rejector is chosen to recover the Bayes-optimal decision rule when the base model provides calibrated probability estimates for $\mathbb{P}(y \mid x)$ and $\mathbb{P}(y \neq h_{\text{exp}}(x))$. We employ these parameterisations in all our experiments in §5.

Table 1: Summary of old and new approaches for learning to defer. We use $\bar{f}(x)$ and $f(x)$ to denote scoring functions with and without defer option $\perp$. We use $\bar{p}(x)$ to denote a softmax transformation on all $L+1$ logits of $\bar{f}$, and $p(x)$ to denote a softmax transformation on the first L logits of either f or $\bar{f}$. We use $\sigma(z) = \frac{1}{1+e^{-z}}$ to denote the sigmoid function. Rows 2–5 describe the post-hoc thresholding approach in §4.1. Rows 6–8 describe the post-hoc training approach in §4.2.

Base Loss	**Rejector Loss**	**Deferral rule**	**Description**
$\ell_{\text{CSS}}(y, x, \bar{f}(x); c_0)$	—	$\max_y \bar{p}_y(x) < \bar{p}_\perp(x)$	Cost-sensitive CE [19]
$\ell_{\text{OVA}}(y, x, \bar{f}(x); c_0)$	—	$\max_y \bar{f}_y(x) < \bar{f}_\perp(x)$	Cost-sensitive OVA [34]
$\ell_{\text{CSS}}(y, x, \bar{f}(x); 0)$	—	$\max_y \bar{p}_y(x) < (1+c_0) \cdot \bar{p}_\perp(x) - c_0$	CSS + Post-thresholding
$\ell_{\text{OVA}}(y, x, \bar{f}(x); 0)$	—	$\max_y \sigma\left(\bar{f}_y(x)\right) < \sigma\left(\bar{f}_\perp(x)\right) - c_0$	OVA + Post-thresholding
$\ell_{\text{SOVA}}(y, x, \bar{f}(x); 0)$	—	$\max_y p_y(x) < \sigma\left(\bar{f}_\perp(x)\right) - c_0$	SOVA + Post-thresholding
$\ell_{\text{CE}}(y, f(x))$	$\ell_{\text{rej}}(y, x, r(x); c_0)$, where $r(x) = g_\perp(x) - \max_{y \in [L]} p_y(x)$	$r(x) > 0$	CE + Post-hoc training
$\ell_{\text{CSS}}(y, x, \bar{f}(x); 0)$	$\ell_{\text{rej}}(y, x, r(x); c_0)$, where $r(x) = g_\perp(x) + \frac{\bar{p}_\perp(x)}{1-\bar{p}_\perp(x)} - \max_{y \in [L]} \frac{\bar{p}_y(x)}{1-\bar{p}_\perp(x)}$	$r(x) > 0$	CSS + Post-hoc training
$\ell_{\text{OVA}}(y, x, \bar{f}(x); 0)$	$\ell_{\text{rej}}(y, x, r(x); c_0)$, where $r(x) = g_\perp(x) + \sigma\left(\bar{f}_\perp(x)\right) - \max_{y \in [L]} \sigma\left(\bar{f}_y(x)\right)$	$r(x) > 0$	OVA + Post-hoc training
$\ell_{\text{SOVA}}(y, x, \bar{f}(x); 0)$	$\ell_{\text{rej}}(y, x, r(x); c_0)$, where $r(x) = g_\perp(x) + \sigma\left(\bar{f}_\perp(x)\right) - \max_{y \in [L]} p_y(x)$	$r(x) > 0$	SOVA + Post-hoc training

G Relation to differentiable triage paradigm

Okati et al. [22] consider a variant of the L2D setup where one is given a *budget* on the fraction of examples on which the model is allowed to defer. The goal is to now learn a base model and deferral rule satisfying this budget constraint. An analogous constrained setting has also been considered in the adaptive inference literature [15]. One potential approach to incorporating post-hoc techniques into this framework is to formulate an equivalent Lagrangian saddle-point problem, and perform alternating *stochastic* updates on $\bar{f}$, and updates on the multiplier associated with the budget constraint. Further exploring this would be of interest.

H Details of experiment parameters

H.1 Synthetic dataset

We construct a distribution $\mathbb{P}(x, y) = \sum_{a \in \{0,1\}} \mathbb{P}(x, y, a)$ as follows. We fix $\mathbb{P}(a = 1) = 0.5$, $\mathbb{P}(y = 1 \mid a) = 0.5$. For subgroup $a = 0$, we set $\mathbb{P}(x \mid y, a = 0) = \mathcal{N}(\mu^{(y,a)}, \Sigma^{(y,a)})$. We draw the elements of $\mu_0^{(0,0)} \sim \text{Unif}(-2, -1)$, $\mu_1^{(0,0)} \sim \text{Unif}(-1, -1)$, and set $\mu_0^{(1,0)} = \mu_0^{(0,0)} + z$ for $z \sim \text{Unif}(3, 4)$. We set $\Sigma^{(0,0)} = \Sigma^{(1,0)}$ to be the identity matrix scaled by a random constant $s \sim \text{Unif}(0, 1)$. Observe that the Gaussians for $a = 0$ both have identical isotropic covariance. The Bayes-optimal solution for the subgroup $\mathbb{P}(x, y \mid a = 0)$ is a linear classifier over the raw features x.

For subgroup $a = 1$, we draw $\theta \sim \text{Unif}(0, 2\pi)$, and set $r = 5 + y$. We then collect samples of the form $(r \cos\theta, r \sin\theta)$.

H.2 Real-world datasets

For all datasets, we perform minibatch SGD optimisation with momentum 0.9. Dataset specific settings are given below.

Quantity	CIFAR-*	ImageNet
Weight decay	10^{-4}	5×10^{-4}
Epochs	256	90
Batch size	1024	512
Learning rate: peak	1.0	0.4
Learning rate: warmup epochs	15	5
Learning rate: annealing	Decay by 0.1 after 96th, 192nd, 224th epoch	Cosine
Data augmentation	[14]	[12]

Table 2: Details of hyperparameters for various datasets.

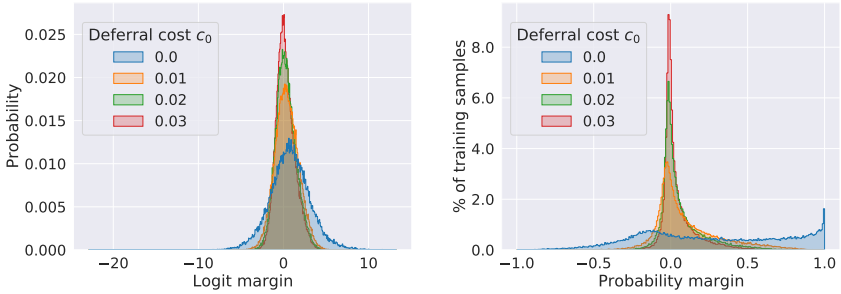

Figure 7: Illustration of underfitting of the cost-sensitive softmax cross-entropy (CSS) loss of Mozannar and Sontag [19] on CIFAR-100. For the setup in Figure 1, we plot the margins in logit and probability space as c_0 is varied. These confirm that the range of scores shrinks as c_0 increases.

I Additional experiments: underfitting of existing losses

I.1 Underfitting of cost-sensitive softmax cross-entropy

For the setting in Figure 1, Figure 7 shows plots of the *logit margins* $f_y(x) - \max_{y' \neq y} f_{y'}(x)$, and the *probability* margins $p_y(x) - \max_{y' \neq y} p_{y'}(x)$. Both of these degrade as c_0 increases, indicating that the probability of the true label is increasingly confusable with that of other labels.

I.2 Underfitting of one-versus-all loss

For the setting in Figure 1, Figure 7 shows analogous plots for the one-versus-all loss of Verma and Nalisnick [34]. From the top left panel, we see that as c_0 increases, the training accuracy degrades. There is a sharper transition here in the margin distributions as c_0 becomes non-zero. Nonetheless, we again see that while $c_0 = 0$ produces highly confident predictions on the training set (e.g., from the entropy plot), the introduction of $c_0 > 0$ makes the predictions much less confident (e.g., as evident from the entropy and margin plots).

I.3 Impact of expert cost function on underfitting

We emphasise here that the underfitting issues above are a consequence of incorporating a fixed deferral cost $c_0 > 0$ in addition to the expert error probability. In settings where $c_0 = 0$, *or* where the expert is assumed to be perfect, such an issue is not present. The former has been considered as part of the plots in the previous section. The latter is potentially applicable in some settings, and indeed corresponds to the classic learning to reject setting [2, 9, 6]. Here, we may set $c_{\max} = 1$, and the cost-sensitive softmax cross-entropy simplifies to

$$\ell_{\text{CSS}}(x, y, \bar{f}(x)) = -\log(\bar{p}_y(x)) - (1 - c_0) \cdot \log(\bar{p}_\perp(x)),$$

which is similar in nature to (8). As with (8), we will not exhibit severe underfitting as a result of the loss only considering $\bar{p}_y$ and $\bar{p}_\perp$, as opposed to all possible labels.

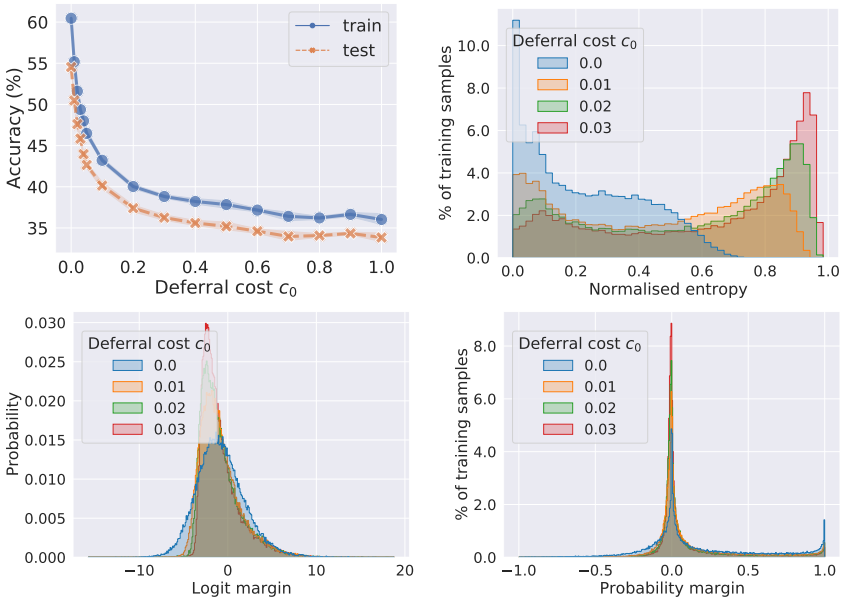

Figure 8: Illustration of underfitting of the **one-versus-all** (`OvA`) loss of Verma and Nalisnick [34] on CIFAR-100. These confirm that as with the CSS loss, the model underfits as c_0 increases, which is a consequence of shrinking margins and increased entropy.

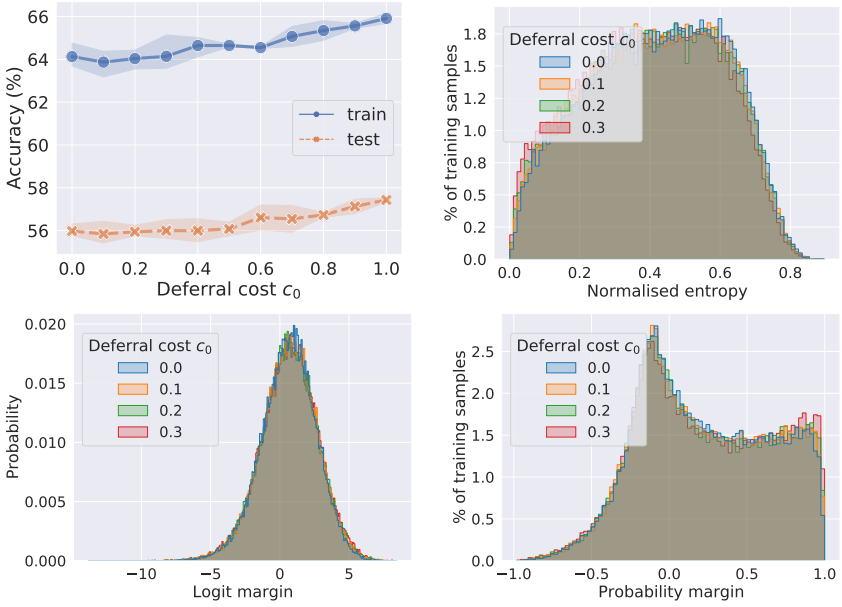

Figure 9: Illustration of impact of choice of expert cost function on underfitting. We follow the same setup as Figure 1 on CIFAR-100, but employ a **constant expert cost** $c_{\text{exp}}(x, y) = c_0 \leq 1$, with $c_{\max} = 1$. Here, we see that as c_0 increases, there is **no degradation** of training accuracy. This is owing to this choice of cost only corresponding to a mild form of label smoothing.

To confirm this, we repeat the CIFAR-100 experiment considered in Figure 1 with a constant cost of $c_{\text{exp}}(x, y) = c_0$.

I.4 Impact of number of labels on underfitting

Figure 10 shows an analogue of Figure 1 for the CIFAR-10 dataset. We see largely similar trends, although the magnitude of degradation is not as severe as CIFAR-100. This is consistent with our

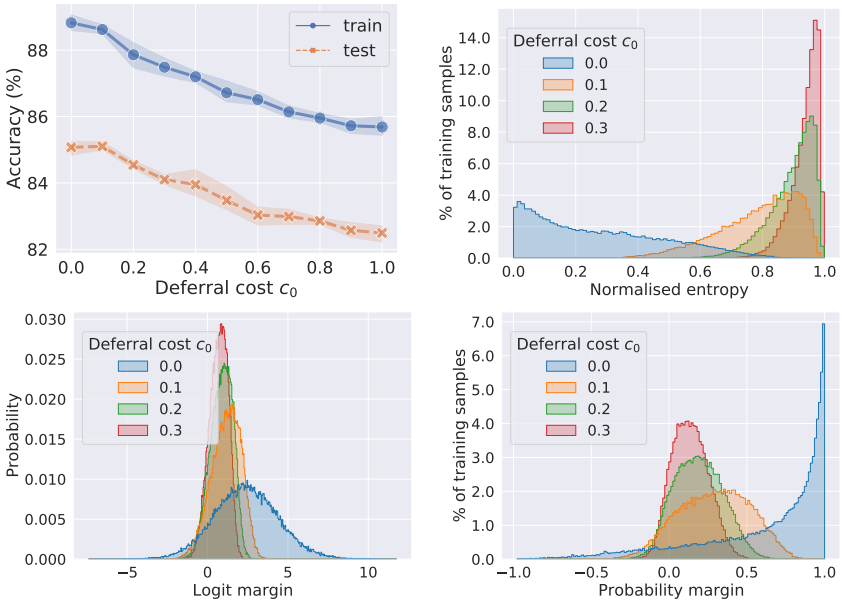

Figure 10: Illustration of underfitting of the cost-sensitive softmax cross-entropy (CSS) loss of Mozannar and Sontag [19]. We repeat the setup in Figure 1 for the **CIFAR-10** dataset. We see largely similar trends as in CIFAR-100.

analysis of the large L setting being problematic, since the amount of label smoothing on all labels scales with L.

J Additional experiments: quality of one-versus-all probability estimates

We demonstrate that the probabilities $\bar{p}_1, \ldots, \bar{p}_L$ for the standard labels obtained from the one-versus-all loss (OvA) may underperform those obtained from the hybrid softmax cross-entropy plus OvA loss. Figure 11 compares the distribution of the *log-loss* $-\log \bar{p}_y(x)$, the entropy normalised by $\log L$, and the logit and probability margins. We see that the hybrid softmax cross-entropy plus OvA loss results in consistently lower log-loss values, as well as entropies and margins. Intuitively, the latter suggests that in the regular OvA loss, there are *multiple* labels y' for which $\bar{f}_{y'}$ is large, resulting in multiple large $\bar{p}_{y'}$ values. As a qualifying comment, the value of OvA probabilities has been demonstrated in *out-of-domain* settings [23]; it is of interest to further study the intersection of this with the deferral setup considered in the present work.

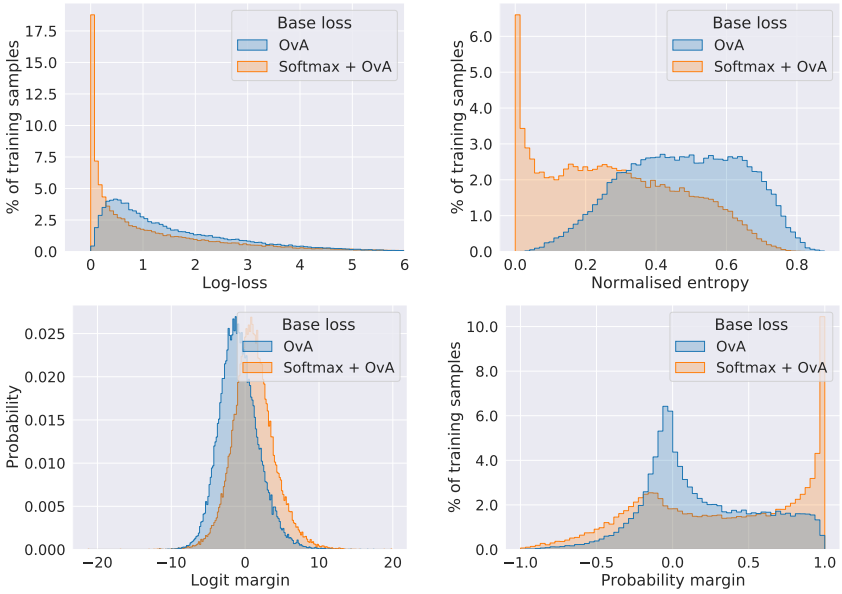

Figure 11: Comparison of standard `OvA` loss with hybrid softmax cross-entropy plus `OvA`, **CIFAR-100** dataset. We see that the hybrid softmax cross-entropy plus `OvA` loss results in consistently lower log-loss values, as well as entropies and margins.

K Additional experiments: OvA and SOvA losses under different post-hoc schemes

We explained in §4.1 why the hybrid softmax cross-entropy plus OvA loss is well-suited for use with post-hoc thresholding approach, and provided empirical evidence in the previous section to showcase why it is a better alternative to the standard OvA loss. For the sake of completeness, we compare the performance of these losses when combined with both the post-hoc schemes proposed in this paper.

Figure 12 presents results for all four combinations (OvA and SOvA with $c_0 = 0$ coupled with either post-hoc thresholding and post-hoc training) on the CIFAR-10 and CIFAR-100 datasets in a learning to defer setting where the base models can defer to "specialist" experts (§5.2). On both datasets, it is one of the SOvA-based post-hoc approaches that performs the best for most operating points. Between the two post-hoc thresholding approaches, the one that uses the SOvA loss is clearly seen to yield significantly higher accuracies than the OvA loss.

We also plot the results from two additional baselines. The first is post-hoc training on a base model trained with the standard softmax cross-entropy (CE) loss. The fact that this baseline is seen to under-perform for most deferral costs highlights the benefit of using a loss that explicitly allows the base model to defer to the expert. The second is the confidence-based approach of Raghu et al. [26], where one trains a base model using the CE loss, then trains a model to get confidence estimates for the expert, and compares the base model's confidence scores with those estimated for the expert to decide whether to defer on an example. For the confidence estimation, we adopt the same architecture as the post-hoc rejector we use in our post-hoc training scheme, namely a linear model on top of the embeddings learned by the first model, and train it with the logistic loss to predict whether the expert classifies a given training example correctly. As seen, at least one of the proposed post-hoc schemes outperforms the baseline on almost all operating points, highlighting the benefit of training the defer logit jointly with the base model.

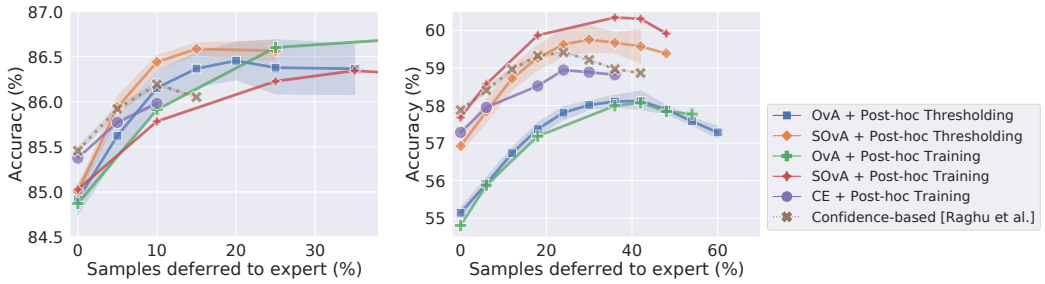

Figure 12: Additional results on CIFAR-10 (left), and CIFAR-100 (middle) in a learning to defer setting where a base model is allowed to defer to a "specialist" expert (same setting as in Figure 5). We compare the post-hoc thresholding and post-hoc training approaches with either the one-vs-all (OvA) loss or the hybrid softmax cross-entropy plus OvA loss (SOvA) loss used to train the base model. We additionally include post-hoc training on a base model trained with the softmax cross-entropy (CE) loss and the confidence-based approach of Raghu et al. [26]. The post-hoc training methods use the rejector parameterisations in Table 1.

L Additional experiments: OvA and CSS losses with different post-hoc thresholding rules

In another set of experiments, we train the base model using the OvA and CSS losses (with $c_0 = 0$), and apply the post-hoc thresholding rule in (11) to defer on an example. We also include a variant of this approach, where we train using the OvA and CSS losses (with $c_0 = 0$), ignore the defer logit, and apply Chow's rule [8] with $c_{\text{exp}}(x, y) = c_0$ to defer on an example. The latter approach does not explicitly take the expert's error into account. Figure 13 presents comparisons of all four combinations OvA and CSS losses with two the different thresholding schemes. We additionally include the standard Chow's rule for learning to reject with a constant rejection cost c_0 (CE + Chow), and the proposed scheme of training the base model using the SOvA loss (with $c_0 = 0$) and applying our post-hoc thresholding rule (SOvA + Post-hoc Thresholding). The proposed post-hoc thresholding approach is seen to yield a higher accuracy than all the baselines at most operating points. Also, we see that, as expected, the use

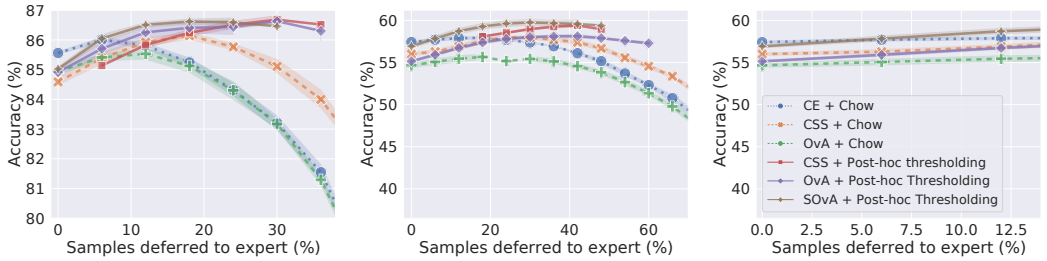

Figure 13: Additional results on CIFAR-10 (left), CIFAR-100 (middle), and ImageNet (right) in a learning to defer setting where a base model is allowed to defer to a "specialist" expert (same setting as in Figure 5). We compare for all four combinations of `OvA` and `CSS` losses (with $c_0 = 0$) coupled with either Chow's rule or the post-hoc thresholding rule in (11). We also include the standard Chow's rule where the base model is trained with the `CE` loss and our proposed post-hoc thresholding approach where the base model is trained with the `SOvA` loss.

of Chow's thresholding rule generally under-performs a post-hoc thresholding scheme that explicitly takes the expert's error into account.

M Additional plots: deferral risk in (1) vs c_0

The plots in the main body show the tradeoff between the fraction of samples deferred to the experts, and the accuracy. In these plots, the role of the fixed cost c_0 is *implicit*, with high fixed costs translating to a small fraction of samples deferred to the expert. We may also make the role of c_0 *explicit*, by studying how varying c_0 affects the deferral risk in (1). These are presented in Figures 14 and 15. As with Figures 5 and 6, we see that the `CSS` and `OvA` losses tend to underperform at high fixed costs c_0. Figures 16 and 17 further show the accuracy of the final classifier as a function of c_0. Again, we observe that the `CSS` and `OvA` accuracy tends to suffer as c_0 is increased. With the adaptive inference experiments, `CSS` and `OvA` tend to defer a large proportion of the examples to the expert even at higher costs (Figure 18), and as a result yield a higher accuracy, but with poorer deferral risk.

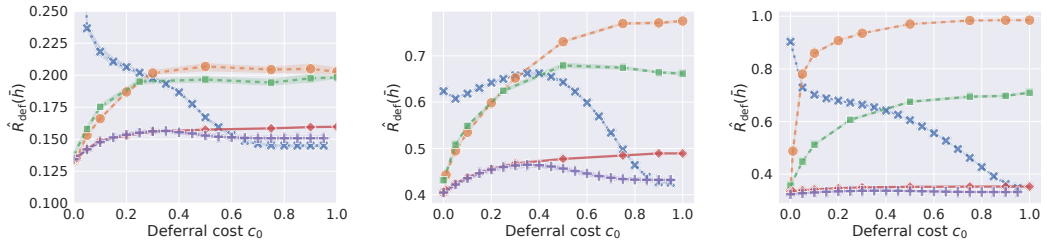

Figure 14: Results on CIFAR-10 (left), CIFAR-100 (middle), and ImageNet (right) in a learning to defer setting, where a base model is allowed to defer to a "specialist" expert. Here, we vary the fixed cost c_0 and study the resulting learning to defer risk (1).

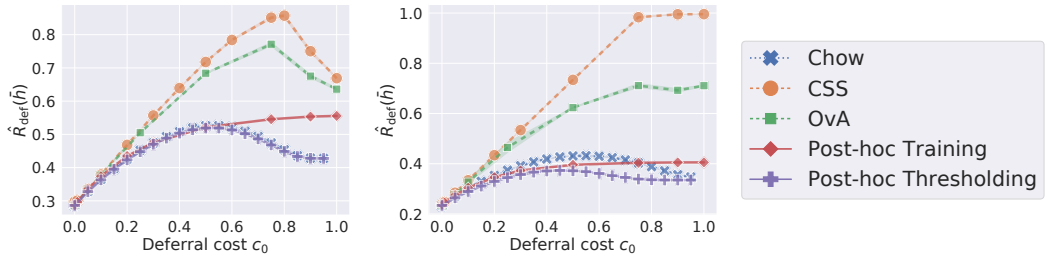

Figure 15: Results on CIFAR-100 (left), and ImageNet (right) in an adaptive inference setting, where a computationally cheap base model is allowed to defer to a more expensive expert. Here, we vary the fixed cost c_0 and study the resulting learning to defer risk (1).

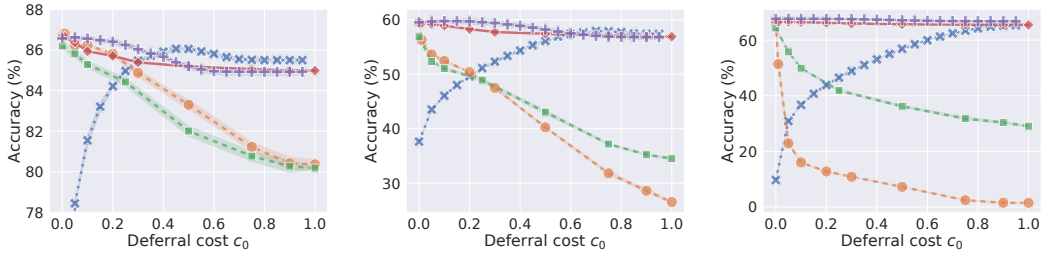

Figure 16: Results on CIFAR-10 (left), CIFAR-100 (middle), and ImageNet (right) in a learning to defer setting, where a base model is allowed to defer to a "specialist" expert. Here, we vary the fixed cost c_0 and study the resulting accuracy of the final classifier.

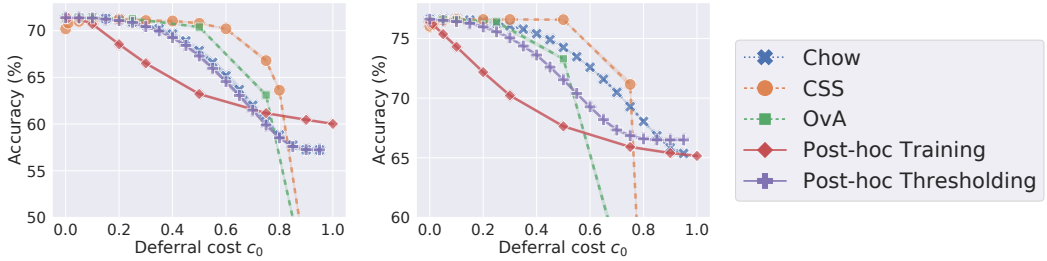

Figure 17: Results on CIFAR-100 (left), and ImageNet (right) in an adaptive inference setting, where a computationally cheap base model is allowed to defer to a more expensive expert. Here, we vary the fixed cost c_0 and study the resulting accuracy of the final classifier.

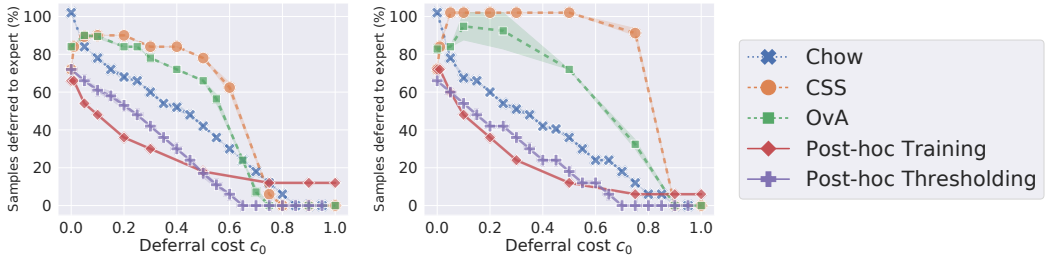

Figure 18: Results on CIFAR-100 (left), and ImageNet (right) in an adaptive inference setting, where a computationally cheap base model is allowed to defer to a more expensive expert. We vary the fixed cost c_0 and plot the resulting proportion of samples deferred to the expert.