Bound to Disagree: Generalization Bounds via Certifiable Surrogates

Mathieu Bazinet[1] **Valentina Zantedeschi**[1,2] **Pascal Germain**[1]

[1]Département d'informatique et génie logiciel, Université Laval, Québec, Qc, Canada
[2]ServiceNow Research, Montreal, Qc, Canada

Abstract

Generalization bounds for deep learning models are typically vacuous, not computable or restricted to specific model classes. In this paper, we tackle these issues by providing new disagreement-based certificates for the gap between the true risk of any two predictors. We then bound the true risk of the predictor of interest via a surrogate model that enjoys tight generalization guarantees, and by evaluating our disagreement bound on an unlabeled dataset. We empirically demonstrate the tightness of the obtained certificates and showcase the versatility of the approach by training surrogate models leveraging three different frameworks: sample compression, model compression and PAC-Bayes theory. Importantly, such guarantees are achieved without modifying the target model, nor adapting the training procedure to the generalization framework.

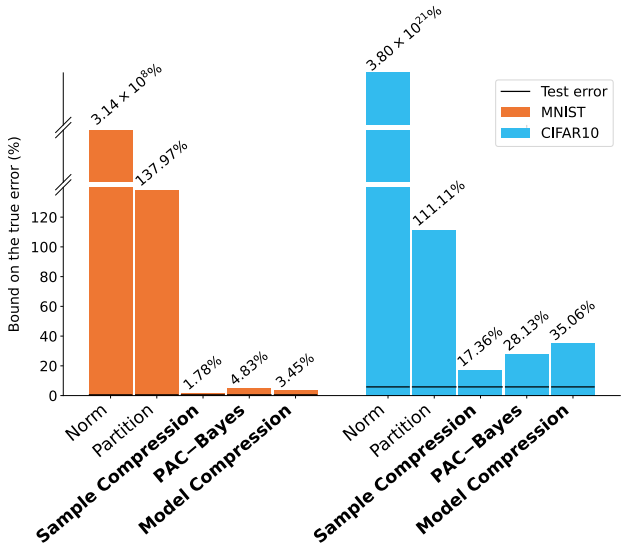


Figure 1: **Generalization bounds on the true risk of the model.** Comparison between bounds from the literature (Norm-based and Partition-based) and our new disagreement-based bounds, using surrogates from sample compression, PAC-Bayes and model compression theory.

1 INTRODUCTION

Deep neural networks have been the stars of the machine learning field for the last decade. Their empirical performance challenges the once-common wisdom that over-parameterized models should overfit the training data [Zhang et al., 2017]. This gap between practice and theory calls for refined theoretical frameworks for studying generalization. The community has mainly turned to statistical learning theory to understand this phenomenon, producing generalization bounds based on the VC dimension [Vapnik and Chervonenkis, 1971], the Rademacher and Gaussian complexities [e.g., Bartlett and Mendelson, 2002, Pinto et al., 2025], information theory [e.g., Hellström et al., 2025], PAC-Bayes theory [McAllester, 1998], sample compression theory [Littlestone and Warmuth, 1986] and model compression theory [e.g., Zhou et al., 2019]. Despite this breadth of approaches, progress has remained limited: most bounds are vacuous when applied to medium-to-large neural networks or apply only to a modified version of the model rather than the original predictor.

A recent line of work sidesteps these issues by leveraging a simpler surrogate model with similar behavior to the network of interest. The strategy is then to bound the true risk of the original model through its link with the surrogate. In particular, Hsu et al. [2021] and Suzuki et al. [2020] leverage a smaller compressed neural network as a surrogate model, and derive generalization bounds based either on distillation or on the Minkowski difference between the two models.

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:491–520.

Table 1: Certification frameworks comparison. We seek deep neural network bounds that are computable, non-vacuous, and do not require assumptions about the target model. The last column shows whether an additional dataset beyond the train set is required (and should it be labeled or unlabeled).

	Computable	Non-vacuous	No assumptions	No additional data
PAC-Bayes	✓	✓	×	✓
Sample Compression	✓	✓	×	✓
Model Compression	✓	✓	×	✓
VC dimension	×	×	✓	✓
Information-theoretic	✓	✓	×	× (need labeled data)
Norm-based	✓	×	✓	✓
Partition-based	✓	×	✓	✓
Our bound (Zero-one loss)	✓	✓	✓	×(need unlabeled data)
Our bound (Lipschitz loss)	✓	✓	✓	× (need unlabeled data)
Our bound (Non-Lipschitz loss)	✓	✓	✓	× (need labeled data)

However, both results depend on universal constants that cannot be computed. In contrast, Dziugaite and Roy [2025] provide PAC-Bayes generalization bounds by pruning neural networks and finding a "teacher" model of smaller size with small risk within them. While the resulting bounds are computable, this approach is restricted to neural networks with gated activations and residual connections, limiting its applicability. Finally, Leblanc and Germain [2025] recently presented a new approach for stochastic surrogates via a conditional decomposition of the loss. Unlike our approach, their method doesn't require a supplementary subset of the data, but cannot be used with stochastic neural networks for classification problems with more than two classes.

In this paper, we present a framework that fulfills all desiderata simultaneously: we provide fully computable, non-vacuous bounds that hold with high probability and apply to any predictor, with no restrictions on architecture, activation functions, training objectives, or optimizer (see Table 1 for an overview of desiderata fulfilled by each framework). Like prior work, we leverage a surrogate model that is simple enough to enjoy tight generalization guarantees on its own, while remaining close enough to the target network that their predictions align on most inputs. The performance gap between the two models is measured through a disagreement bound, which we evaluate on a small unlabeled dataset not used during training, the only additional requirement of our approach. In comparison to labeled data, acquiring unlabeled data is much cheaper and faster to obtain [Liao et al., 2021], making this requirement much less stringent than labeled data.

Our key theoretical contributions are reported in Section 3, where we present a sketch of our disagreement bound, before specializing this result to the zero-one loss and to Lipschitz-continuous losses. We further provide generalization certificates for the case where the disagreement term is optimized on the available unlabeled data.

The resulting certificates are much stronger than the other bounds in the literature that are computable and that hold for the target model without modification, as illustrated in Figure 1. Moreover, unlike previous approaches, our guarantees can be derived for virtually any proxy model. We showcase this versatility in Section 4, where we apply three different approaches (sample compression, model compression, and PAC-Bayes theory) to train and certify the proxy model, and study various families of neural networks: a CNN on MNIST [LeCun et al., 1998], a ResNet18 [He et al., 2016] on CIFAR10 [Krizhevsky et al., 2009], and a DistilBERT [Sanh et al., 2019] and a GPT2 [Radford et al., 2019] on Amazon polarity [Zhang et al., 2015].

2 BACKGROUND AND NOTATION

Let $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$ be a sequence of n independently and identically distributed (*i.i.d.*) data points sampled from an unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. Let $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ be the dataset that contains these data points. In this paper, we consider a feature space $\mathcal{X} \subseteq \mathbb{R}^d$ and a label space $\mathcal{Y} = \{1, 2, \ldots, C\}$ for C-class classification tasks.

The hypothesis class $\mathcal{H}$ contains predictors $h : \mathcal{X} \to \mathbb{R}^C$ with $h(\boldsymbol{x}) = (h(\boldsymbol{x})_1, \ldots, h(\boldsymbol{x})_C)$. A learning algorithm $A : \bigcup_{k=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^k \to \mathcal{H}$ outputs a predictor $A(S) \in \mathcal{H}$ when applied to S. Let $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ denote a loss function with a range $\lambda_\ell = T_\ell - B_\ell$. Given a loss function ℓ, the true loss of a predictor h is

$$\mathcal{L}_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \ell(h(\boldsymbol{x}), y).$$

The true loss cannot be computed, as the distribution $\mathcal{D}$ is unknown. However, given a dataset $S \sim \mathcal{D}^n$, we can compute the empirical loss of a predictor $\widehat{\mathcal{L}}_S(h) = \frac{1}{n}\sum_{i=1}^n \ell(h(\boldsymbol{x}_i), y_i)$. When the choice of $h \in \mathcal{H}$ is dependent on the dataset S, the empirical risk becomes a biased estimator of the true risk. Thus, we rely on generalization bounds to upper-bound the true risk.

Definition 1. *A generalization bound is defined as a high-probability upper-bound on the true risk of a hypothesis $h \in \mathcal{H}$. With a confidence parameter δ and a complexity function $\epsilon(n, \delta)$, for some hypothesis $h \in \mathcal{H}$, we have*

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^n} \Big(\mathcal{L}_{\mathcal{D}}(h) \leq \widehat{\mathcal{L}}_S(h) + \epsilon(n, \delta) \Big) \geq 1 - \delta.$$

In this paper, we are mainly interested in two loss functions: the zero-one loss $\ell^{0\text{-}1}$ and the cross-entropy loss $\ell^{\text{x-e}}$. Given a predictor h and a pair $(\boldsymbol{x}, y)$, the zero-one loss is defined as $\ell^{0\text{-}1}(h(\boldsymbol{x}), y) = \mathbb{I}[\operatorname{argmax}_{j\in\{1,\ldots,C\}} h(\boldsymbol{x})_j \neq y]$, where the indicator function $\mathbb{I}[a]$ takes the value of 1 if the predicate a is true and 0 otherwise. This loss has its own notation for the true risk and empirical risk, respectively denoted $R_{\mathcal{D}}(h)$ and $\widehat{R}_S(h)$.

The cross-entropy loss is widely used to train neural networks and has recently become an object of interest in statistical learning theory [e.g., Pérez-Ortiz et al., 2021,

Lotfi et al., 2024]. The cross-entropy loss is defined as $\ell^{\text{x-e}}(h(\boldsymbol{x}), y) = -\ln(\sigma(h(\boldsymbol{x}), y))$ with the softmax function

$$\sigma(h(\boldsymbol{x}), y) = \frac{\exp(h(\boldsymbol{x})_y)}{\sum_{c=1}^{C} \exp(h(\boldsymbol{x})_c)}.$$

We denote $\boldsymbol{\sigma}(f(\boldsymbol{x})) = (\sigma(f(\boldsymbol{x}), 1), \ldots, \sigma(f(\boldsymbol{x}), C))$ the softmax vector. By forcing the softmax to always be greater than some positive constant, it is possible to bound the cross-entropy loss, which otherwise tends to infinity when the softmax tends to zero. To do so, we consider the clamped softmax of Pérez-Ortiz et al. [2021] (see Definition S1) and the smoothed softmax of Lotfi et al. [2024] (see Definition S2).

We now review the frameworks we leverage to derive the results in Section 3 and experiment with in Section 4.

2.1 SAMPLE COMPRESSION THEORY

Introduced by Littlestone and Warmuth [1986], sample compression theory provides generalization guarantees for data-dependent predictors that can be fully described by a small subset of the training data. Intuitively, if a model's parameters depend on only a few training points, the model is unlikely to have overfit, and sample compression theory formalizes this intuition by providing an upper-bound of the true risk of the model. Some examples of algorithms compatible with the sample compression framework include the support vector machine [Boser et al., 1992], the perceptron [Rosenblatt, 1958, Moran et al., 2020], the decision tree [Shah, 2007], the set covering machine [Marchand and Shawe-Taylor, 2002, Marchand et al., 2003, Marchand and Sokolova, 2005, Laviolette et al., 2005] and Pick-To-Learn [Paccagnan et al., 2024, Marks and Paccagnan, 2025, Paccagnan et al., 2025]. The latter was shown to be very effective at deriving tight deep learning bounds [Bazinet et al., 2025, Comeau et al., 2025].

We denote the compression set $S_{\mathbf{i}} = \{(\boldsymbol{x}_i, y_i)\}_{i \in \mathbf{i}}$, which is defined using a strictly increasing sequence of $|\mathbf{i}|$ indices $\mathbf{i} = (i_1, \ldots, i_{|\mathbf{i}|})$. All sequences $\mathbf{i}$ belong to $\mathcal{P}(n)$, the set of all the 2^n strictly increasing sequences composed of the numbers 1 through n. We denote the complement set $S_{\mathbf{i}^c} = S \setminus S_{\mathbf{i}}$ with $|\mathbf{i}^c| = n - |\mathbf{i}|$. The subset of sample-compressed predictors is denoted $\mathcal{H}_S \subseteq \mathcal{H}$. A predictor $h = A(S)$ is called sample-compressed if there exists a function $\mathcal{R} : \bigcup_{m \leq n} (\mathcal{X} \times \mathcal{Y})^m \to \mathcal{H}_S$ and a sequence $\mathbf{i} \in \mathcal{P}(n)$ such that $\mathcal{R}(S_{\mathbf{i}})$ would return the same predictor as the original algorithm ($A(S) = \mathcal{R}(S_{\mathbf{i}})$). For any such predictor, the sample-compression bound of Bazinet et al. [2025] can be used to upper-bound the true risk of the predictor.

Theorem 2 (Bazinet et al. [2025])**.** *For a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, a loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ and $\delta \in (0, 1]$, with probability at least $1 - \delta$ over the draw of $S \sim \mathcal{D}^n$, simultaneously for all compression sets $\mathbf{i} \in \mathcal{P}(n)$, we have*

$$\mathcal{L}_{\mathcal{D}}(\mathcal{R}(S_{\mathbf{i}})) \leq \widehat{\mathcal{L}}_{S_{\mathbf{i}^c}}(\mathcal{R}(S_{\mathbf{i}})) + \lambda_\ell \sqrt{\frac{1}{2|\mathbf{i}^c|} \log\left(\frac{2\sqrt{|\mathbf{i}^c|}}{P_n(|\mathbf{i}|)\delta}\right)},$$

with $P_n(|\mathbf{i}|) = \frac{6}{\pi^2}(|\mathbf{i}|+1)^{-2}\binom{n}{|\mathbf{i}|}^{-1}$.

This result is versatile and can be applied to any sample-compressed predictor. In practice, one needs to prove that an algorithm produces sample-compressed models in order to compute the bound. A notable example for deep neural networks is the aforementioned Pick-To-Learn algorithm. Interestingly, the coreset methods (see Moser et al. [2025] for an introduction) also fit this framework: a coreset can be viewed as the compression set in Theorem 2 to obtain generalization bounds. To the best of our knowledge, the connection between coreset methods and sample compression has never been investigated.

2.2 MODEL COMPRESSION THEORY

The code length of a model is exploited by Zhou et al. [2019] to measure its complexity: if it is possible to reduce the code length of a model, by either decreasing the number of trained parameters or quantizing its weights without performance degradation, then the model should generalize well. In a similar fashion, Arora et al. [2018] provided computable bounds for compressed models by studying the robustness of the layers to perturbations of an uncompressed model and relating it to the description length of the model. Following Zhou et al. [2019], Lotfi et al. [2022] presented a tight bound for model compression based on the universal prior [Solomonoff, 1964], which introduces the Kolmogorov complexity [Kolmogorov, 1965] of the model into the bound. In practice, they perform subspace compression [Li et al., 2018] and adaptative quantization [Han et al., 2016] to minimize the complexity of the model. Finally, Lotfi et al. [2024] proposed a hybrid approach between subspace compression and LoRA [Hu et al., 2022] named SubLoRA and extended the generalization bound to real-valued losses, a result that we now present. Let $l_{\mathcal{C}}(\hat{h})$ denote the code length in bits of a model $\hat{h}$ according to a code $\mathcal{C}$, and $\mathcal{H}_{\mathcal{C}} \subseteq \mathcal{H}$ be the subset of compressed predictors that can be encoded by $\mathcal{C}$.

Theorem 3 (Lotfi et al. [2024])**.** *For a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, a loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ and $\delta \in (0, 1]$, with probability at least $1 - \delta$ over the draw of $S \sim \mathcal{D}^n$, simultaneously for all $\hat{h} \in \mathcal{H}_{\mathcal{C}}$, we have*

$$\mathcal{L}_{\mathcal{D}}(\hat{h}) \leq \widehat{\mathcal{L}}_S(\hat{h}) + \lambda_\ell \sqrt{\frac{1}{2n}\left[l_{\mathcal{C}}(\hat{h})^2 \log 2^{l_{\mathcal{C}}(\hat{h})} + \log\frac{1}{\delta}\right]}.$$

2.3 PAC-BAYES THEORY

While the previous sections presented inequalities bounding the true risk of a single model, the PAC-Bayes frame-

work principally studies the true risk of stochastic predictors expressed as a distribution over multiple models. Built on the work of McAllester [1998] and Shawe-Taylor and Williamson [1997], PAC-Bayes theory was popularized for stochastic neural networks by Dziugaite and Roy [2017] and Pérez-Ortiz et al. [2021]. Recent developments include applications to meta-learning [e.g., Guan and Lu, 2022, Zakerinia et al., 2024, Leblanc et al., 2025], deep learning generative networks [e.g., Chérief-Abdellatif et al., 2022, Mbacke et al., 2023a,b, Mbacke and Rivasplata, 2024] and large language models [Su et al., 2024].

In this setting, one is interested in learning a distribution Q over the set of predictors $\mathcal{H}$. Starting with a data-independent prior distribution P, an algorithm A' takes as input the dataset S and the prior distribution P and then outputs the posterior distribution $Q = A'(S,\ P)$. PAC-Bayes bounds control generalization by penalizing posteriors that deviate from the prior, as typically measured by the Kullback-Leibler (KL) divergence:

$$\mathrm{KL}(Q||P) = \mathop{\mathbb{E}}_{h\sim Q} \ln \frac{Q(h)}{P(h)}.$$

The following PAC-Bayesian theorem was first presented by McAllester [2003] and tightened by Maurer [2004].

Theorem 4 (McAllester [2003])**.** *For a distribution $\mathcal{D}$ over $\mathcal{X}\times\mathcal{Y}$, a data-independent prior distribution P over $\mathcal{H}$, a loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ and $\delta \in (0,1]$, with probability at least $1-\delta$ over the draw of $S\sim\mathcal{D}^n$, simultaneously for all Q over $\mathcal{H}$, we have*

$$\mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h) \le \mathop{\mathbb{E}}_{h\sim Q}\widehat{\mathcal{L}}_S(h) + \lambda_\ell\sqrt{\frac{\mathrm{KL}(Q||P) + \log\frac{2\sqrt{n}}{\delta}}{2n}}.$$

2.4 OTHER THEORETICAL FRAMEWORKS

There exist several other approaches in statistical learning theory beyond those discussed above. Notably, norm-based generalization bounds [Bartlett et al., 2017] are based on the norm of the network's weight matrices as a measure of complexity. However, these bounds are generally vacuous [Galanti et al., 2023b], sometimes negatively correlated with generalization [Jiang et al., 2020], or cannot be computed, as they either require the input space to be bounded [Golowich et al., 2018] or depend on intractable universal constants [Bartlett and Mendelson, 2002, Lin and Zhang, 2019, Long and Sedghi, 2020, Wei and Ma, 2020, Ledent et al., 2021, Pinto et al., 2025]. Similarly, VC dimension bounds are intractable or vacuous for deep neural networks [Vapnik and Chervonenkis, 1971, Bartlett et al., 1998, 2019].

In recent years, bounds based on information theory gained interest, as the conditional mutual information framework [Steinke and Zakynthinou, 2020, Hellström and Durisi, 2020] tends to yield non-vacuous bounds for stochastic gradient descent [Hellström and Durisi, 2022]. However, these bounds either hold in expectation or use a super-sample of size $2n$, where the model is trained on a first half of the super-sample and the guarantee upper-bounds the risk on the second half, instead of the true risk.

Finally, Galanti et al. [2023a] use the effective depth of the neural network as complexity measure. However, their bound is in expectation and depends on multiple assumptions that are not always satisfied in practice. Than and Phan [2025] provide vacuous generalization guarantees for trained models that depend on a partition of the input space.

3 DISAGREEMENT-BASED BOUNDS

As discussed in the previous section, most bounds for deep learning are either vacuous, not computable or hold only for a restricted class of hypotheses. In this section, we present new theoretical results applicable to any machine learning model. We first formalize the general ideas underpinning our approach by stating a BOUND SKETCH, which presents the form of our proposed disagreement-based bounds over the gap between the losses of two models, and show how this can be leveraged for three use cases. We then instantiate our general scheme to three different types of losses: the zero-one loss (Theorem 5), Lipschitz-continuous losses (Theorem 6) and non-Lipschitz-continuous losses (Theorem 7), a weaker result which requires a labeled dataset for computing the disagreement.

The following BOUND SKETCH provides the general scheme of the forthcoming high-probability bounds on the gap between the true losses of two predictors. We denote $U = \{(\boldsymbol{x}_i, \cdot)\}_{i=1}^m$ an unlabeled dataset sampled *i.i.d.* from the distribution $\mathcal{D}$.

BOUND SKETCH. *Given two predictors $f, h \in \mathcal{H}$, a loss function ℓ, and a proper (to be defined) disagreement measure $D_U(f,h;\ell,\delta)$ between f and h over a dataset U. The proposed disagreement-based bounds are such that, with probability at least $1-\delta$ over the sampling of $U\sim\mathcal{D}^m$, we have*

$$\mathcal{L}_{\mathcal{D}}(f) \le \mathcal{L}_{\mathcal{D}}(h) + D_U(f,h;\ell,\delta). \tag{1}$$

From now on, we take f to be the target model, i.e., a model that does not enjoy tight generalization bounds, either because its complexity is too large or because it doesn't fit the required assumptions, and h to be the surrogate model for which tight generalization bounds exist. Let us describe three use cases of the proposed disagreement-based bounds.

Use case #1 : Certifying the target model. Let $\overline{\mathcal{H}} \subseteq \mathcal{H}$ be a class of certifiable surrogate models, such as the class of sample-compressed predictors $\mathcal{H}_S$ or the class of compressed models $\mathcal{H}_{\mathcal{C}}$. Let Q be a distribution over $\overline{\mathcal{H}}$. Suppose that there exists a function $\epsilon(n,\delta)$ bounding the generalization gap of the proxy model h, that is, the following generalization bound on its true risk holds with probability at least

$1-\delta$ over the sampling of $S \sim \mathcal{D}^n$:

$$\forall h \in \overline{\mathcal{H}} : \mathcal{L}_{\mathcal{D}}(h) \leq \widehat{\mathcal{L}}_S(h) + \epsilon(n, Q(h)\delta)\,. \qquad (2)$$

Examples of the function $\epsilon(n, Q(h)\delta)$ would be the square-root terms in Theorems 2, 3 and 4, respectively for surrogate models from sample compression, model compression and PAC-Bayes theory.

Then, given a chosen model $h^\star \in \overline{\mathcal{H}}$, combining Equations (1) and (2) gives that, with probability at least $1-2\delta$ over the sampling of $S \sim \mathcal{D}^n$ and $U \sim \mathcal{D}^m$:

$$\mathcal{L}_{\mathcal{D}}(f) \leq \widehat{\mathcal{L}}_S(h^\star) + \epsilon(n, Q(h^\star)\delta)) + D_U(f, h^\star; \ell, \delta)\,. \quad (3)$$

Note that this type of bounds does not hold simultaneously for all $h \in \overline{\mathcal{H}}$, but the model $h^\star$ can be dependent on the dataset S, notably via an optimization algorithm. Furthermore, the model f can also depend on S, as its complexity does not appear in the bound of Equation (3). When a surrogate model $h^\star$ with tight generalization guarantees is obtainable, then this strategy renders a certificate for f whose tightness depends on the disagreement measured by the chosen function $D_U(f, h; \ell, \delta)$.

Use case #2 : Minimizing the disagreement. The second use case of this disagreement bound is to certify the true risk of the target model whilst minimizing the disagreement between the two models. With probability at least $1-2\delta$ over the sampling of $S \sim \mathcal{D}^n$ and $U \sim \mathcal{D}^m$, simultaneously for all $h \in \overline{\mathcal{H}}$, we have

$$\mathcal{L}_{\mathcal{D}}(f) \leq \widehat{\mathcal{L}}_S(h) + \epsilon(n, Q(h)\delta) + D_U(f, h; \ell, Q(h)\delta)\,. \quad (4)$$

We obtain this result by applying Equation (1) with a union bound over all $h \in \overline{\mathcal{H}}$, followed by a union bound with Equation (3). Although this added step loosens the bound in principle, it allows the surrogate model to be selected and trained on both the datasets S and U. One way to train the surrogate model on the unlabeled set is via model distillation [Schmidhuber, 1992, Hinton et al., 2015], using pseudo-labels produced by the target model.

Use case #3 : Bounding the true risk gap. The final use case of this bound is to guarantee that replacing a model h with a new model f with better qualities does not lead to a significant performance drop. Such qualities could be that the model is fairness-aware [e.g., Dwork et al., 2012, Hardt et al., 2016], differentially private [e.g., Dwork, 2006, Dwork and Roth, 2014] or achieves faster inference, either via model distillation [e.g., Buciluǎ et al., 2006, Hinton et al., 2015] or model quantization [e.g., Han et al., 2016, Nagel et al., 2020]. Note that bounding the risk gap can lead to useful insights even when both models are too complex to enjoy non-vacuous generalization bounds.

3.1 DISAGREEMENT WITH THE ZERO-ONE LOSS

We present our first result for the zero-one loss. To do so, we build on the work of Yang et al. [2024], who presented a theoretical result to compare the reconstruction error of a coreset. We show that this result can be upper-bounded with high probability via the test set bound of Langford [2005] (see Theorem S6), which is essentially perfectly tight for binomial distributions. With these two results, we can state a disagreement measure for the zero-one loss, which is defined as

$$d_U^{0\text{-}1}(f,h) = \frac{1}{m}\sum_{i=1}^{m} \mathbb{I}\left[\operatorname*{argmax}_j f(\boldsymbol{x}_i)_j \neq \operatorname*{argmax}_k h(\boldsymbol{x}_i)_k\right].$$

The disagreement measure $d_U^{0\text{-}1}$ compares the labels predicted by each model for a given data point. If the disagreement bound between f and h tends to zero, then the models achieve the same true risk. We now present our first result: a disagreement-based bound for the zero-one loss. The proof of all results are given in Appendix B.

Theorem 5. *For two predictors $h \in \overline{\mathcal{H}}$ and $f \in \mathcal{H}$, with probability at least $1-\delta$ over the sampling of $U \sim \mathcal{D}^m$, we have*

$$R_{\mathcal{D}}(f) \leq R_{\mathcal{D}}(h) + \overline{\mathrm{Bin}}\big(m d_U^{0\text{-}1}(f,h), m, \delta\big),$$

with $\overline{\mathrm{Bin}}(k,m,\delta) = \operatorname{argsup}_{p\in[0,1]}\{\mathrm{Bin}(k,m,p) \geq \delta\}$ *and* $\mathrm{Bin}(k,m,p) = \sum_{i=0}^{k} \binom{m}{i} p^i (1-p)^{m-i}$.

This result provides generalization guarantees for the zero-one loss of any target model f by training a surrogate predictor h with tight generalization bounds and a small disagreement with f; In the experiments of Section 4, we do so by bounding the true risk $R_{\mathcal{D}}(h)$ of the surrogate model using sample compression and model compression guarantees.

By definition, the binomial tail inversion $\overline{\mathrm{Bin}}(k,m,\delta)$ [Langford, 2005] gives the tightest upper-bound on the error rate of a binomial random variable. To do so, the binomial tail inversion returns the greatest error rate p such that $\mathrm{Bin}(k,m,p)$ the probability of making k or fewer errors out of m samples is greater than δ. As it computes the exact worst-case error rate, there cannot be a tighter bound.

3.2 DISAGREEMENT WITH LIPSCHITZ LOSSES

We now generalize the previous result for Lipschitz-continuous bounded losses. An intuitive measure of disagreement between two models is to compare the decision boundary of the models by comparing their output distributions :

$$d_U^{K_\ell}(f,h) = \frac{1}{m}\sum_{i=1}^{m} \|\boldsymbol{\sigma}(f(\boldsymbol{x}_i)) - \boldsymbol{\sigma}(h(\boldsymbol{x}_i))\|_1\,.$$

When this measure of disagreement tends to zero, the models predict the same class for each data point sampled from $\mathcal{D}$, which means that the models generalize similarly. The following Theorem 6 uses the Chernoff bound for random variables in the interval unit of Foong et al. [2022] (see Theorem S7) in place of the zero-one loss-based test set bound exploited by Theorem 5. Furthermore, Theorem 6 provides a bound on a target predictor f given a distribution Q over a class of surrogate models. This allows us to apply PAC-Bayes bounds (such as Theorem 4) to the true risk of the surrogate model. Still, the following disagreement-based bound for Lipschitz-continuous $[B_\ell, T_\ell]$-losses can be applied to a single surrogate model by setting Q to a Dirac distribution over a single hypothesis.

Theorem 6. *For a distribution Q over $\mathcal{H}$, a predictor $f \in \mathcal{H}$, a Lipschitz loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ with a Lipschitz constant K_ℓ, with probability at least $1-\delta$ over the sampling of $U \sim \mathcal{D}^m$, we have*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + 2K_\ell \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{d_U^{K_\ell}(f,h)}{2}, \frac{\log\frac{1}{\delta}}{m} \right)$$

with $\mathrm{kl}^{-1}(q,\epsilon) = \mathrm{argsup}_{p\in[0,1]} \{\mathrm{kl}(q,p) \leq \epsilon\}$ *and* $\mathrm{kl}(q,p) = q\log\frac{q}{p} + (1-q)\log\frac{1-q}{1-p}$.

The above result can be applied to common losses such as the logistic loss, the hinge loss and the Huber loss, which are known to be Lipschitz functions [Chinot et al., 2020]. The cross-entropy loss is Lipschitz with respect to the softmax output of the model when constraining the entries of the softmax output to be greater than a given positive constant, which is achieved with the clamped or the smoothed softmax. However, the obtained Lipschitz constant K_ℓ is typically large and renders vacuous bounds. In the following subsection, we present a weaker result that bounds the loss of a model using disagreement on a labeled dataset.

3.3 DISAGREEMENT WITH NON-LIPSCHITZ LOSSES

Let $L = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^m \sim \mathcal{D}^m$ be a small held-out labeled set of data points removed before training f. Then, we provide the following disagreement bound for any bounded non-Lipschitz loss.

Theorem 7. *For a distribution Q over $\mathcal{H}$, a predictor $f \in \mathcal{H}$, a loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ with range $\lambda_\ell = T_\ell - B_\ell$, with probability at least $1-\delta$ over the sampling of $L \sim \mathcal{D}^m$:*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \lambda_\ell \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{d_L(f,h)}{\lambda_\ell}, \frac{\log\frac{1}{\delta}}{m} \right),$$

with

$$d_L(f,h) = \frac{1}{m}\sum_{i=1}^m \left|\ell(f(\boldsymbol{x}_i), y_i) - \ell(h(\boldsymbol{x}_i), y_i)\right|.$$

In contrast with Theorems 5 and 6, the disagreement d_L is not a disagreement between the output of the models. However, we show in Corollary S16 that, for Lipschitz losses with large K_ℓ constants, d_L is a proxy for disagreement between the models, as it is upper-bounded by the disagreement between the probability distributions outputted by the models. We instantiate this result for the clamped and smoothed cross-entropy respectively in Appendix C.1.

For both Theorems 6 and 7, the expectation of the true loss of the surrogate might not be tractable, which is the case for stochastic neural networks [Pérez-Ortiz et al., 2021]. Following the Monte Carlo sampling approach of Pérez-Ortiz et al. [2021], we use a Chernoff bound [Foong et al., 2022] to upper-bound the expectation of both the risks of the surrogate and the disagreement between the surrogate and target models, as described in Appendix C.2.

When used with the zero-one loss, d_L is upper-bounded by $d_U^{0\text{-}1}$, leading to a bound on the true risk of the target using PAC-Bayes theory and an unlabeled dataset U, which wasn't possible with the test set bound of Langford [2005].

4 EXPERIMENTS

In this section, we demonstrate the versatility and efficacy of our framework by training and certifying a variety of deep neural networks for which tight generalization bounds were previously unavailable.[1] In Section 4.1, we focus on use case #1 and thus leverage Theorem 5 to bound the zero-one loss and Theorem 7 to bound the cross-entropy loss of the target network, obtaining a surrogate via sample compression, model compression or PAC-Bayes training. In Section 4.2, we illustrate use case #2 by applying Theorem 6 to the true Huber loss of the target model, while optimizing a compressed surrogate via model distillation. Finally, in Section 4.3, we consider use case #3 and evaluate the certificate on the gap between the true risks of the target model and its quantized version.

For all experiments, we report the mean and standard deviation over five training-evaluation runs. We randomly sample 10% of the training set to form a validation set. Of the remaining training set, we further select a disagreement set, corresponding to 20% of data for MNIST and CIFAR10, and 85% for Amazon polarity [2]. We optimize the smoothed softmax with $\alpha = 10^{-3}$, following the result of an ablation study on coreset methods in Appendix D.4.1. Thus, in all tables and figures, the cross-entropy loss is bounded by $\ln(10^3 C)$, which is approximately 9.21 with $C = 10$. All the bounds presented hold with probability $1 - \delta = 0.99$. Other hyperparameters are detailed in Appendix D.

[1] Our code is available at `https://github.com/GRAAL-Research/Bound-to-Disagree`.

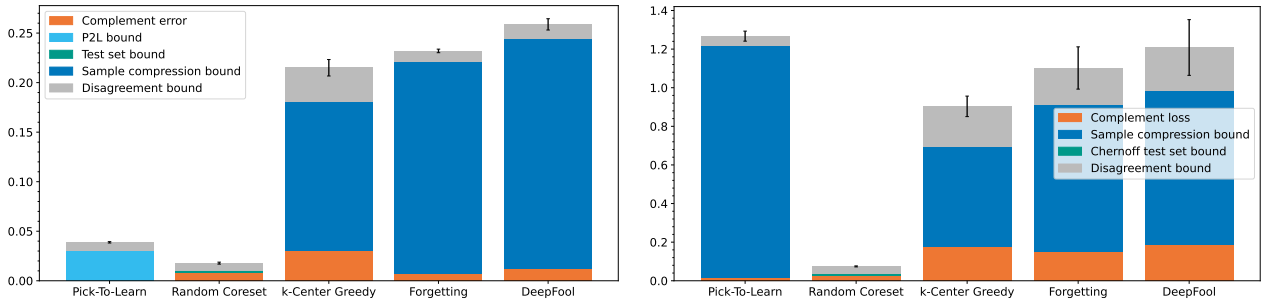
[2] 15% corresponds to roughly 500000 data points, more than enough data to train the models on Amazon polarity.

Table 2: Generalization bounds for the zero-one loss of the target models according to the approach (and theorem) used to certify the surrogate. The bound values are reported in %.

	Bound (Theorem)	MNIST	CIFAR10
Baselines	Partition-based (S4)	86.55±0.53	90.58±0.47
	Norm-based (S5)	$(3.14\pm0.37)\times10^{8}$	$(3.80\pm0.30)\times10^{21}$
Our approach	Random Coreset (S8)	**1.79±0.19**	**17.36±0.20**
	Best Coreset (S9)	21.50±0.83	81.72±0.69
	Pick-To-Learn (S10)	3.90±0.09	57.13±2.71
	Model compression (S11)	3.45±0.11	35.06±0.40
	PAC-Bayes (S15)	4.83±0.32	28.13±1.55

Table 3: Generalization bounds for the smoothed cross-entropy loss of the target models according to the approach (and theorem) used to certify the surrogate.

	Bound (Theorem)	MNIST	CIFAR10
Baselines	Partition-based (S4)	7.9546±0.0475	8.3437±0.0431
	Norm-based	N/A	N/A
Our approach	Random Coreset (S12)	**0.0744±0.0026**	**0.6858±0.0089**
	Best Coreset (S13)	0.9035±0.0531	5.7076±0.0970
	Pick-To-Learn (S13)	1.2673±0.0248	7.4941±0.2277
	Model compression (S14)	0.1756±0.0041	1.7851±0.0223
	PAC-Bayes (S15)	0.2592±0.0087	1.4204±0.1377



(a) Zero-one loss

(b) Smoothed cross-entropy loss

Figure 2: Illustration of the behavior of our disagreement bounds on MNIST using sample-compression methods.

4.1 CERTIFYING THE TARGET MODEL

For these experiments, we first train a CNN on MNIST [LeCun et al., 1998] and a ResNet18 [He et al., 2016] on CIFAR10 [Krizhevsky et al., 2009] as target models. Because these models are deterministic, trained on the whole training set and not quantized, the only generalization bounds of the literature that we can compare with are: the **partition-based bound** of Than and Phan [2025] (see Theorem S4), and **norm-based bounds**, among which we present the results of Galanti et al. [2023b] (see Theorem S5), which is only defined for the zero-one loss. Note that for the partition-based bound, contrary to the original paper, we compute the partition on the disagreement set, to obtain non-vacuous results. See Appendix D.2.1 for an ablation study on the way of computing the partition. Tables 2 and 3 report a summary of the value of our new disagreement bounds for all the experiments, where we specify the generalization framework from the literature used to certify the surrogate model. The disagreement bounds displayed were obtained without considering the disagreement by choosing the surrogates with the tightest generalization bounds. We observe that both norm-based bounds and partition-based bounds are unsuited to certify the target model both on MNIST and on CIFAR10: norm-based bounds are vacuous as cross-entropy maximizes the network's weight norms [Galanti et al., 2023b], and partition-based bounds are almost trivial, as they estimate a generalization performance close to a random predictor (90%). Moreover, the latter cannot differentiate between two problems of varying complexities, such as MNIST and CIFAR10.

Sample compression In our first experiment, we use sample-compression bounds to certify the target models on MNIST and CIFAR10. To do so, we train neural networks with the same architecture as the targets using Pick-To-Learn (P2L) [Paccagnan et al., 2024] and coreset methods such as Random Coreset, k-Center Greedy [Sener and Savarese, 2018], Forgetting [Toneva et al., 2019] and DeepFool [Ducoffe and Precioso, 2018].

Figure 2 presents the contribution of each term of the bound on MNIST for the different sample-compression methods. The results for CIFAR10 are presented in Figure S2 in Appendix D. The complement error is added as complementary information, but the bound is not linear in the complement error. We observe that each method is able to achieve tight generalization bounds for our target models. The tightest bound is achieved by the Random Coreset approach, which is expected as it enjoys a test set bound, which means that the complexity of the random coreset is never accounted for in the bound. In comparison, the other bounds are train set bounds, meaning the coreset method must find a compromise between accuracy and coreset size. Moreover, the Random Coreset approach was found to be a very competitive baseline by Guo et al. [2022], which can explain that this method achieves a small train and test error. For the zero-one loss, P2L achieves the second-tightest bound. For the cross-entropy loss, it actually achieves the worst bound

Table 4: Generalization bounds on the zero-one loss achieved on MNIST and CIFAR10 using model compression (MC) bounds. We present the metrics for the surrogate model used to certify the target model. All metrics are in percents (%), except the size.

Dataset	Surrogate model			Target model	
	Test error	Size (KB)	MC Bound	Test error	Our bound
MNIST	0.89±0.08	0.04±0.00	2.45±0.13	0.50±0.07	3.45±0.11
CIFAR10	14.95±0.42	0.03±0.00	20.15±0.35	5.84±0.16	35.06±0.40

Table 5: Generalization bounds on the zero-one loss achieved on MNIST and CIFAR10 using PAC-Bayes (PB) bounds. We present the metrics for the surrogate model used to certify the target model. All metrics presented are in percents (%), except the KL.

Dataset	Surrogate model			Target model	
	Test error	KL	PB Bound	Test error	Our bound
MNIST	0.82±0.07	4.68±0.22	2.28±0.13	0.50±0.07	4.83±0.32
CIFAR10	11.33±0.55	1.62±0.92	14.04±0.91	5.84±0.16	28.13±1.55

of all methods. The complexity of the model learned using P2L is much higher than the coreset methods, however, the P2L bound (Theorem S10) still achieves significantly tighter bounds. In Tables 2 and 3, we report the bound for P2L, Random Coreset and the best (non-random) coreset method. Of all the methods presented, Random Coreset achieves the tightest disagreement bound for our target models. All of the sample compression bounds are significantly tighter than both the norm-based bound and the partition-based bound.

Model compression. For the model compression experiments, we use the SubLoRA method of Lotfi et al. [2024]. To showcase a different approach that leads to tight certificates, we start by pretraining a CNN on 20% of the training set for MNIST and a ResNet18 on 50% of the training set for CIFAR10. We then add a low-rank adapter [Hu et al., 2022] and use the subspace compression method [Li et al., 2018] implemented by Lotfi et al. [2022] on the adapter. After training the adapter on the remaining training data, the subspace vector is quantized using adaptative quantization [Han et al., 2016] and encoded using arithmetic encoding [Rissanen and Langdon, 1979]. The bound on the surrogate's loss is computed over the set on which the adapter was trained.

In Table 4, we report the bound on the zero-one loss for the SubLoRA approach on MNIST and CIFAR10. We report the bound on the cross-entropy loss in Table S2. On MNIST, the model compression bounds are the second-tightest bounds reported in Table 2, whilst they are the third-tightest bounds for CIFAR10. It was expected that these bounds would be much tighter than the sample-compression bounds, as pretraining the models tightens significantly the guarantees [Ambroladze et al., 2006, Parrado-Hernández et al., 2012, Dziugaite and Roy, 2018]. However, we note that even after an extensive hyperparameter search on CIFAR10, the bound favors a smaller subspace vector, where the model only marginally improves after the pretraining, if at all.

PAC-Bayes. For the PAC-Bayes experiments, we train stochastic neural networks with the implementation of Pérez-Ortiz et al. [2021]. We start by pretraining a deterministic neural network, either a CNN on 20% of the training set for MNIST or a ResNet18 on 60% or 70% on the training set of CIFAR10. We then add Gaussian distributions over the weights of the neural network, with the mean and the standard deviation of the distributions as trainable parameters. To compute the bound of Theorem S15, we approximate the expectation via Monte Carlo sampling. The full statement of the disagreement loss with the Monte Carlo sampling is presented in Theorem S20.

In Table 5, we report the bound on the zero-one loss for the stochastic neural networks on MNIST and CIFAR10. The results on the cross-entropy loss are reported in Table S3. As the models were pretrained, they achieve small test errors even with a small KL divergence, similarly to the model compression bound. In contrast with the previous approaches, the disagreement bounds are actually larger than the compressed model bounds. The approximation via the Monte Carlo sampling and the variance in the stochastic predictor leads to a larger disagreement between the model and its surrogate. However, this approach provides tight generalization bounds and achieves the second-best disagreement bound on CIFAR10.

4.2 MINIMIZING THE DISAGREEMENT

For the second use case of our disagreement bounds, we provide model distillation experiments on Amazon polarity dataset [Zhang et al., 2015]. Our target models are a DistilBERT [Sanh et al., 2019] and a GPT2 [Radford et al., 2019]. Using the target models, we create pseudo-labels for the unlabeled disagreement set and train the surrogate models on both the labeled set S and the unlabeled set U. We freeze the pretrained weights of the model (from the Transformers library [Wolf et al., 2020]) and add SubLoRA weights [Lotfi et al., 2024] on the top half of the model. As the model is trained on both datasets, the bound holds simultaneously for all possible models, following Equation (4). We use the Huber loss (see Definition S3) with $\delta_{\mathrm{H}} = 0.2$ to compute Theorem 6, which means that we can train the model with vanilla softmax.

We present the results for the Huber loss in Table 6, where both models achieved non-vacuous generalization bounds. The DistilBERT surrogate achieved a test loss very similar to the one of the target model, while the GPT2 surrogate's loss is much higher. This could be explained by the fact that the model is twice as big, although the SubLoRA adapter that achieved the best disagreement bound is the same size

Table 6: Generalization bounds on the Huber loss via model distillation on Amazon polarity using model compression bounds. With $\delta_{\text{H}} = 0.2$, the loss is less than $\delta_{\text{H}} - \frac{1}{2}\delta_{\text{H}}^2 = 0.18$.

Dataset	Surrogate model			Target model	
	Test loss	Size (KB)	MC Bound	Test loss	Our bound
DistilBERT	.015±.000	2.02±0.01	.027±.000	.012±.000	.059±.001
GPT2	.035±.004	2.04±0.43	.052±.004	.010±.000	.152±.015

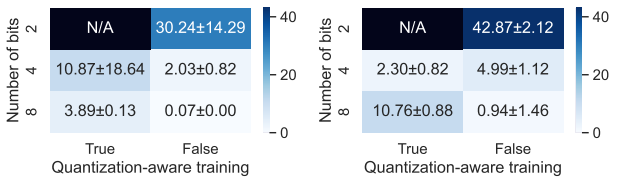
(in KB) as the one of DistilBERT. Indeed, as the distribution Q is present twice in Equation (4), the size of the model becomes much more important in the bound minimization than the loss of the model, leading to higher train and test losses. Although the bound is only available for the Huber loss instead of the cross-entropy loss, the model's softmax doesn't have to be modified, the dataset does not need to be labeled and the bound minimizes the actual disagreement between the two models. We report the results for the zero-one loss in Table S4. For both architectures, we are able to provide non-vacuous and non-trivial (less than 50% for binary classification) generalization bounds for the targets.

4.3 BOUNDING THE TRUE RISK GAP

To demonstrate our third use case, we provide quantization experiments on Amazon polarity, with the same target models as the previous section. We quantize the models and compare their disagreement with the unquantized target models. In contrast to the previous sections, the surrogates are still too large to enjoy non-vacuous model compression bounds, even after quantization. We can use our disagreement bounds to provide an upper-bound on the gap between the true risk of two models. This will guarantee that although the surrogate is quantized, the models will perform similarly on unseen data points. We present results for models with and without quantization-aware (QA) training. For QA training, we quantize to 4 bits using TorchAO [Or et al., 2025] and 8 bits using AO from PyTorch [Ansel et al., 2024]. For the other experiments, we use half-quadratic quantization (HQQ) [Badri and Shaji, 2023] to quantize the models.

We report the results in Figure 3. Without a surprise, fully quantizing a model to 2 bits leads to large performance loss. The tightest disagreement bounds were obtained by using 8 bits with HQQ for both models. The QA training with 4 bits achieves a better test error for both models than 8 bit training, leading, however, the DistilBERT model to overfit on the training set. For both DistilBERT and GPT2, respectively without and with quantization-aware training, the 4-bits models are able to achieve disagreement bound of about 2%, whilst achieving faster inference and using less memory.

Remark. Finding a good surrogate model is the most important step to use our disagreement framework. We provide some recommendations derived from our experience



(a) DistilBERT (b) GPT2

Figure 3: Behavior of the disagreement bound according to the number of bits and the use of QA training.

in training the models presented in this section. In general, choosing the same model architecture for both the target and surrogate models leads to very small disagreement between the models, although choosing a smaller model with model compression theory may yield tighter bounds. For sample compression surrogates, we recommend using the random coreset approach or Pick-To-Learn. The random coreset achieved the tightest bound by far with the smallest computational cost, while Pick-To-Learn gives control on the complement error via early stopping. For model compression surrogates, we found that using SubLoRA [Lotfi et al., 2024] with a base model pretrained on 20 to 50% of the dataset works best. Finally, stochastic neural networks with Gaussian distributions are generally harder to train than the other surrogates, but can yield tight generalization bounds when the base model is pretrained on 50 to 70% of the data.

5 CONCLUSION

We proposed new disagreement-based bounds that can be leveraged for any machine learning predictor. These bounds require no assumptions about the model's architecture or the training method. We showed that these bounds can be used with multiple statistical learning theory frameworks to obtain tight generalization bounds, by experimenting on datasets of varying complexity such as MNIST, CIFAR10 and Amazon polarity.

Despite achieving eloquent improvement over existing bounds in many contexts, our proposed framework also has limitations that deserve to be addressed in future work. Notably, the disagreement-based bounds are only as good as the guarantees used to certify the surrogate models, and finding a surrogate for a very complex model can be challenging. Moreover, the tightest bounds are obtained without minimizing the disagreement on the unlabeled set, giving no control on the disagreement between the target and the surrogate. In particular, the random coreset approach provides little control over the resulting bound, yet yields the tightest bounds of all the approaches. Finally, we believe that this approach may be generalized for regression problems and even possibly to text generation by leveraging the token-level bound of Lotfi et al. [2025].

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. We wish to thank Jacob Comeau and Benjamin Leblanc for helping review the paper, Benjamin Guedj, Paul Viallard, Romaric Gaudel and Omar Rivasplata for the insightful discussions that helped elevate this paper and Dario Paccagnan for pointing an error in the P2L bound code before the submission.

Mathieu Bazinet is supported by a FRQNT B2X scholarship (343192, `https://doi.org/10.69777/343192`). Pascal Germain is supported by the NSERC Discovery grant RGPIN-2020-07223.

References

Amiran Ambroladze, Emilio Parrado-Hernández, and John Shawe-Taylor. Tighter PAC-Bayes bounds. *Advances in neural information processing systems*, 19, 2006.

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, Will Constable, Alban Desmaison, Zachary DeVito, Elias Ellison, Will Feng, Jiong Gong, Michael Gschwind, Brian Hirsh, Sherlock Huang, Kshiteej Kalambarkar, Laurent Kirsch, Michael Lazos, Mario Lezcano, Yanbo Liang, Jason Liang, Yinghai Lu, CK Luk, Bert Maher, Yunjie Pan, Christian Puhrsch, Matthias Reso, Mark Saroufim, Marcos Yukio Siraichi, Helen Suk, Michael Suo, Phil Tillet, Eikan Wang, Xiaodong Wang, William Wen, Shunting Zhang, Xu Zhao, Keren Zhou, Richard Zou, Ajit Mathews, Gregory Chanan, Peng Wu, and Soumith Chintala. PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, 2024.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In *Proceedings of the 35th International Conference on Machine Learning 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 254–263. PMLR, 2018.

Hicham Badri and Appu Shaji. Half-quadratic quantization of large machine learning models, November 2023. URL `https://dropbox.github.io/hqq_blog/`.

Peter Bartlett, Vitaly Maiorov, and Ron Meir. Almost linear VC dimension bounds for piecewise polynomial networks. *Advances in neural information processing systems*, 11, 1998.

Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of machine learning research*, 3(Nov):463–482, 2002.

Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

Peter L Bartlett, Nick Harvey, Christopher Liaw, and Abbas Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *Journal of Machine Learning Research*, 20(63):1–17, 2019.

Mathieu Bazinet, Valentina Zantedeschi, and Pascal Germain. Sample compression unleashed: New generalization bounds for real valued losses. In *AISTATS*, 2025.

Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com.

Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

Badr-Eddine Chérief-Abdellatif, Yuyang Shi, Arnaud Doucet, and Benjamin Guedj. On PAC-Bayesian reconstruction guarantees for vaes. In *International conference on artificial intelligence and statistics*, pages 3066–3079. PMLR, 2022.

Geoffrey Chinot, Guillaume Lecué, and Matthieu Lerasle. Robust statistical learning with Lipschitz and convex loss functions. *Probability Theory and Related Fields*, 176(3): 897–940, 2020.

Jacob Comeau, Mathieu Bazinet, Pascal Germain, and Cem Subakan. Sample compression for self certified continual learning. *arXiv preprint arXiv:2503.10503*, 2025.

Aaron Defazio, Xingyu Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The road less scheduled. *Advances in Neural Information Processing Systems*, 37:9974–10007, 2024.

Melanie Ducoffe and Frederic Precioso. Adversarial active learning for deep networks: a margin based approach. *arXiv preprint arXiv:1802.09841*, 2018.

Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. Springer, 2006.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and trends® in theoretical computer science*, 9(3-4):211–487, 2014.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

Gintare Karolina Dziugaite and Daniel M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *Proceedings of the 33rd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, 2017.

Gintare Karolina Dziugaite and Daniel M. Roy. Data-dependent PAC-Bayes priors via differential privacy. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018*, pages 8440–8450, 2018.

Gintare Karolina Dziugaite and Daniel M Roy. The size of teachers as a measure of data complexity: PAC-Bayes excess risk bounds and scaling laws. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025.

William Falcon and The PyTorch Lightning team. PyTorch Lightning, 2019.

Andrew YK Foong, Wessel P Bruinsma, and David R Burt. A note on the Chernoff bound for random variables in the unit interval. *ArXiv preprint*, abs/2205.07880, 2022.

Tomer Galanti, Liane Galanti, and Ido Ben-Shaul. Comparative generalization bounds for deep neural networks. *Transactions on Machine Learning Research*, 2023a.

Tomer Galanti, Mengjia Xu, Liane Galanti, and Tomaso Poggio. Norm-based generalization bounds for sparse neural networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Pascal Germain, Alexandre Lacasse, François Laviolette, and Mario Marchand. PAC-Bayesian learning of linear classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML*, volume 382 of *ACM International Conference Proceeding Series*, pages 353–360. ACM, 2009.

Noah Golowich, Alexander Rakhlin, and Ohad Shamir. Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pages 297–299. PMLR, 2018.

Jiechao Guan and Zhiwu Lu. Fast-rate PAC-Bayesian generalization bounds for meta-learning. In *ICML*, pages 7930–7948, 2022.

Chengcheng Guo, Bo Zhao, and Yanbing Bai. Deepcore: A comprehensive library for coreset selection in deep learning. In *International Conference on Database and Expert Systems Applications*, pages 181–195. Springer, 2022.

Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding. In *ICLR*, 2016.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020.

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

Fredrik Hellström and Giuseppe Durisi. Generalization bounds via information density and conditional information density. *IEEE Journal on Selected Areas in Information Theory*, 1(3):824–839, 2020.

Fredrik Hellström and Giuseppe Durisi. A new family of generalization bounds using samplewise evaluated CMI. In *Advances in Neural Information Processing Systems*, 2022.

Fredrik Hellström, Giuseppe Durisi, Benjamin Guedj, and Maxim Raginsky. Generalization bounds: Perspectives from information theory and PAC-Bayes. *Foundations and Trends in Machine Learning*, 18(1):1–223, 2025.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Daniel Hsu, Ziwei Ji, Matus Telgarsky, and Lan Wang. Generalization bounds via distillation. In *International Conference on Learning Representations*, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

Andrei N Kolmogorov. Three approaches to the quantitative definition of information. *Problems of information transmission*, 1(1):1–7, 1965.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

John Langford. Tutorial on practical prediction theory for classification. *Journal of machine learning research*, 6 (3), 2005.

John Langford and Rich Caruana. (Not) bounding the true error. *Advances in Neural Information Processing Systems*, 14, 2001.

François Laviolette, Mario Marchand, and Mohak Shah. Margin-Sparsity Trade-Off for the Set Covering Machine. In *Machine Learning: ECML 2005*, volume 3720, pages 206–217. Springer Berlin Heidelberg, 2005. ISBN 978-3-540-29243-2 978-3-540-31692-3.

Benjamin Leblanc and Pascal Germain. A framework for bounding deterministic risk with PAC-Bayes: Applications to majority votes. *arXiv preprint arXiv:2510.25569*, 2025.

Benjamin Leblanc, Mathieu Bazinet, Nathaniel D'Amours, Alexandre Drouin, and Pascal Germain. Generalization bounds via meta-learned model representations: PAC-Bayes and sample compression hypernetworks. In *Forty-second International Conference on Machine Learning*, 2025.

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

Antoine Ledent, Waleed Mustafa, Yunwen Lei, and Marius Kloft. Norm-based generalisation bounds for deep multi-class convolutional neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 8279–8287, 2021.

Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. In *International Conference on Learning Representations*, 2018.

Yuan-Hong Liao, Amlan Kar, and Sanja Fidler. Towards good practices for efficiently annotating large-scale image classification datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4350–4359, 2021.

Shan Lin and Jingwei Zhang. Generalization bounds for convolutional neural networks. *arXiv preprint arXiv:1910.01487*, 2019.

Nick Littlestone and Manfred Warmuth. Relating data compression and learnability. 1986.

Philip M. Long and Hanie Sedghi. Generalization bounds for deep convolutional neural networks. In *International Conference on Learning Representations*, 2020.

Sanae Lotfi, Marc Finzi, Sanyam Kapoor, Andres Potapczynski, Micah Goldblum, and Andrew G Wilson. PAC-Bayes compression bounds so tight that they can explain generalization. *Advances in Neural Information Processing Systems*, 35:31459–31473, 2022.

Sanae Lotfi, Marc Anton Finzi, Yilun Kuang, Tim G. J. Rudner, Micah Goldblum, and Andrew Gordon Wilson. Non-vacuous generalization bounds for large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 32801–32818. PMLR, 2024.

Sanae Lotfi, Yilun Kuang, Marc Finzi, Brandon Amos, Micah Goldblum, and Andrew G Wilson. Unlocking tokens as data points for generalization bounds on larger language models. *Advances in Neural Information Processing Systems*, 37:9229–9256, 2025.

Mario Marchand and John Shawe-Taylor. The set covering machine. *Journal of Machine Learning Research*, 3(4-5): 723–746, 2002.

Mario Marchand and Marina Sokolova. Learning with decision lists of data-dependent features. *Journal of Machine Learning Research*, 6(4), 2005.

Mario Marchand, Mohak Shah, John Shawe-Taylor, and Marina Sokolova. The set covering machine with data-dependent half-spaces. In *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003)*, pages 520–527. AAAI Press, 2003.

Daniel Marks and Dario Paccagnan. Pick-to-Learn and self-certified gaussian process approximations. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025.

Andreas Maurer. A note on the PAC Bayesian theorem. *arXiv preprint cs/0411099*, 2004.

Sokhna Diarra Mbacke and Omar Rivasplata. A note on the convergence of denoising diffusion probabilistic models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Sokhna Diarra Mbacke, Florence Clerc, and Pascal Germain. PAC-Bayesian generalization bounds for adversarial generative models. In *International Conference on Machine Learning*, pages 24271–24290. PMLR, 2023a.

Sokhna Diarra Mbacke, Florence Clerc, and Pascal Germain. Statistical guarantees for variational autoencoders using PAC-Bayesian theory. *Advances in Neural Information Processing Systems*, 36:56903–56915, 2023b.

David A McAllester. Some PAC-Bayesian theorems. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 230–234, 1998.

David A McAllester. PAC-Bayesian stochastic model selection. *Machine Learning*, 51(1):5–21, 2003.

Shay Moran, Ido Nachum, Itai Panasoff, and Amir Yehudayoff. On the perceptron's compression. In *Beyond the Horizon of Computability: 16th Conference on Computability in Europe, CiE 2020*, pages 310–325. Springer, 2020.

Brian B Moser, Arundhati S Shanbhag, Stanislav Frolov, Federico Raue, Joachim Folz, and Andreas Dengel. A coreset selection of coreset selection literature: Introduction and recent advances. *arXiv preprint arXiv:2505.17799*, 2025.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International conference on machine learning*, pages 7197–7206. PMLR, 2020.

Andrew Or, Apurva Jain, Daniel Vega-Myhre, Jesse Cai, Charles David Hernandez, Zhenrui Zhang, Driss Guessous, Vasiliy Kuznetsov, Christian Puhrsch, Mark Saroufim, et al. Torchao: Pytorch-native training-to-serving model optimization. In *Championing Open-source DEvelopment in ML Workshop@ ICML25*, 2025.

Francesco Orabona and Tatiana Tommasi. Training deep networks without learning rates through coin betting. *Advances in neural information processing systems*, 30, 2017.

Dario Paccagnan, Marco Campi, and Simone Garatti. The pick-to-learn algorithm: Empowering compression for tight generalization bounds and improved post-training performance. *Advances in Neural Information Processing Systems*, 36, 2024.

Dario Paccagnan, Daniel Marks, Marco C Campi, and Simone Garatti. Pick-to-learn for systems and control: Data-driven synthesis with state-of-the-art safety guarantees. *arXiv preprint arXiv:2512.04781*, 2025.

Emilio Parrado-Hernández, Amiran Ambroladze, John Shawe-Taylor, and Shiliang Sun. PAC-Bayes bounds with data dependent priors. *The Journal of Machine Learning Research*, 13(1):3507–3531, 2012.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

María Pérez-Ortiz, Omar Rivasplata, John Shawe-Taylor, and Csaba Szepesvári. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22 (227):1–40, 2021.

Andrea Pinto, Akshay Rangamani, and Tomaso A Poggio. On generalization bounds for neural networks with low rank layers. In *36th International Conference on Algorithmic Learning Theory*, 2025.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

J. Rissanen and G. G. Langdon. Arithmetic coding. *IBM J. Res. Dev.*, 23(2):149–162, 1979. ISSN 0018-8646.

Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv preprint*, abs/1910.01108, 2019.

Jürgen Schmidhuber. Learning complex, extended sequences using the principle of history compression. *Neural Computation*, 4(2):234–242, 1992.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Mohak Shah. Sample compression bounds for decision trees. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007)*, volume 227 of *ACM International Conference Proceeding Series*, pages 799–806. ACM, 2007.

John Shawe-Taylor and Robert C Williamson. A PAC analysis of a Bayesian estimator. In *Proceedings of the tenth annual conference on Computational learning theory*, pages 2–9, 1997.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning*

for multi-domain operations applications, volume 11006, pages 369–386. SPIE, 2019.

Ray J Solomonoff. A formal theory of inductive inference. part i. *Information and control*, 7(1):1–22, 1964.

Thomas Steinke and Lydia Zakynthinou. Reasoning about generalization via conditional mutual information. In *Conference on Learning Theory*, pages 3437–3452. PMLR, 2020.

Jingtong Su, Julia Kempe, and Karen Ullrich. Mission impossible: A statistical perspective on jailbreaking llms. *Advances in Neural Information Processing Systems*, 37: 38267–38306, 2024.

Taiji Suzuki, Hiroshi Abe, and Tomoaki Nishimura. Compression based bound for non-compressed network: unified generalization error analysis of large compressible deep neural network. In *International Conference on Learning Representations*, 2020.

Khoat Than and Dat Phan. Non-vacuous generalization bounds for deep neural networks without any modification to the trained models. *arXiv preprint arXiv:2503.07325*, 2025.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. In *ICLR*, 2019.

V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

Paul Viallard, Pascal Germain, Amaury Habrard, and Emilie Morvant. Self-bounding majority vote learning algorithms by the direct minimization of a tight PAC-Bayesian C-bound. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 167–183. Springer, 2021.

Colin Wei and Tengyu Ma. Improved sample complexities for deep neural networks and robust classification via an all-layer margin. In *International Conference on Learning Representations*, 2020.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics, 2020.

Shuo Yang, Zhe Cao, Sheng Guo, Ruiheng Zhang, Ping Luo, Shengping Zhang, and Liqiang Nie. Mind the boundary: Coreset selection via reconstructing the decision boundary. In *Forty-first International Conference on Machine Learning*, 2024.

Hossein Zakerinia, Amin Behjati, and Christoph H. Lampert. More flexible PAC-bayesian meta-learning by learning learning algorithms. In *Forty-first International Conference on Machine Learning*, 2024.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pages 649–657, 2015.

Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P. Adams, and Peter Orbanz. Non-vacuous generalization bounds at the imagenet scale: a PAC-Bayesian compression approach. In *International Conference on Learning Representations*, 2019.

Bound to Disagree: Generalization Bounds via Certifiable Surrogates (Supplementary Materials)

Mathieu Bazinet[1] **Valentina Zantedeschi**[1,2] **Pascal Germain**[1]

[1]Département d'informatique et génie logiciel, Université Laval, Québec, Qc, Canada
[2]ServiceNow Research, Montreal, Qc, Canada

A DEFINITIONS AND THEORETICAL RESULTS FROM THE LITERATURE

Definition S1 (Pérez-Ortiz et al. [2021])**.** *Given a parameter $\alpha \in (0,1)$ and a number of classes C, the clamped softmax is defined as*

$$\sigma_{\max}(h(\boldsymbol{x}), y) = \max\left(\frac{\alpha}{C}, \frac{\exp\left(h(\boldsymbol{x})_y\right)}{\sum_{c=1}^{C}\exp(h(\boldsymbol{x})_c)}\right),$$

and the cross-entropy loss is bounded with $B_\ell = 0$, $T_\ell = \ln\frac{C}{\alpha}$ and $\lambda_\ell = \ln\frac{C}{\alpha}$.

Definition S2 (Lotfi et al. [2024])**.** *Given a parameter $\alpha \in (0,1)$ and a number of classes C, the smoothed softmax is defined as*

$$\sigma_{\text{smooth}}(h(\boldsymbol{x}), y) = (1-\alpha)\left(\frac{\exp\left(h(\boldsymbol{x})_y\right)}{\sum_{c=1}^{C}\exp(h(\boldsymbol{x})_c)}\right) + \frac{\alpha}{C},$$

and the cross-entropy loss bounded with $B_\ell = \ln\left(1-\alpha+\frac{\alpha}{C}\right)$, $T_\ell = \ln\frac{C}{\alpha}$ and $\lambda_\ell = \ln\left(1+(1-\alpha)\frac{C}{\alpha}\right)$.

Definition S3. *Given $h : \mathcal{X} \to \mathbb{R}^C$, $y \in \mathbb{R}^C$ and $\delta_H > 0$, the Huber loss is defined as*

$$\ell^{\text{Huber}}(h(\boldsymbol{x}), y) = \frac{1}{C}\sum_{i=1}^{C}\begin{cases}\frac{1}{2}(\sigma(h'(\boldsymbol{x}), i) - y_i)^2 & \text{if } |\sigma(h'(\boldsymbol{x}), i) - y_i| \le \delta_H \\ \delta_H|\sigma(h'(\boldsymbol{x}), i) - y_i| - \frac{1}{2}\delta_H^2 & \text{otherwise.}\end{cases} \tag{5}$$

As we apply a softmax over the output of the predictor, this loss is bounded by $\delta_H - \frac{1}{2}\delta_H^2$. This loss is Lipschitz-continuous with $K^\ell = \delta_H$ [Chinot et al., 2020].

Theorem S4 (Partition-based bound of Than and Phan [2025])**.** *Let $\mathcal{Z} = \mathcal{X}\times\mathcal{Y}$ and let $\Gamma(\mathcal{Z}) = \bigcup_{i=1}^{K}\mathcal{Z}_i$ be a partition of $\mathcal{Z}$ into K subsets. Let $n_i = |S \cap \mathcal{Z}_i|$ be the number of samples of a dataset S grouped into $\mathcal{Z}_i$. Let $\mathcal{T} = \sum_{i=1}^{k}\mathbb{I}[n_i \neq 0]$ be the number of subsets of $\Gamma(\mathcal{Z})$ in which data points from S fall into. For any partition Γ into K subsets, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [0, T_\ell]$, for any constants $\gamma \ge 1$ and $\eta \in \left[0, \frac{\gamma n(K+\gamma n)}{K(4n-3)}\right]$, with probability at least $1 - \gamma^{-\eta} - \delta$ over the draw of $S \sim \mathcal{D}^n$, for a given model h_S trained on S, we have*

$$\mathcal{L}_{\mathcal{D}}(h_S) \le \widehat{\mathcal{L}}_S(h_S) + T_\ell\sqrt{\eta\ln\gamma}\sqrt{\frac{\gamma}{2n} + \frac{\gamma^2}{2}\sum_{i=1}^{K}\left(\frac{n_i}{n}\right)^2 + \gamma^2\sqrt{\frac{2}{n}\ln\frac{2K}{\delta}}} + T_\ell(\sqrt{2}+1)\sqrt{\frac{\mathcal{T}\ln\frac{4K}{\delta}}{n}} + \frac{2T_\ell\mathcal{T}\ln\frac{4K}{\delta}}{n}.$$

Theorem S5 (Norm-based bound of Galanti et al. [2023b])**.** *Given a fixed neural network architecture G that defines a directed acyclic graph. Let the architecture have L layers and d_l neurons at each layer $l \in \{1,\dots,L\}$. Denote z_i^l is the i-th neuron of the l-th layer. Let z_i^{l-1} be a predecessor of the neuron z_j^l if there exists an edge between the two neurons and denote $pred(l,j)$ the set of predecessors of the neuron z_j^l. Let $\mathcal{H}_G$ be the class of neural networks with architecture*

G. Denote $\rho(h)$ the norm of the model $h \in \mathcal{H}_G$. Given images with c_0 channels and d_0 values such that the images are in $\mathbb{R}^{c_0 d_0}$ and labels in $\mathcal{Y} \subset \mathbb{R}^C$ the set of C-dimensional one-hot vectors. Let $z_j^0(x)$ be the j^{th} pixel of the image. For any distribution $\mathcal{D}$ over $\mathbb{R}^{c_0 d_0} \times \{-1, +1\}$, with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, we have

$$\forall h \in \mathcal{H}_G : R_{\mathcal{D}}(h) \leq \widehat{R}_S^{\gamma}(h) + \frac{2\sqrt{2}(\rho(h)+1)}{\gamma n} \cdot \Lambda(h) + 3\sqrt{\frac{\log(2(\rho(h)+2)^2)/\delta}{2n}},$$

with $\widehat{R}_S^{\gamma}(h) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}[\max_{j \neq y_i} h(\boldsymbol{x}_i)_j + \gamma \geq h(\boldsymbol{x}_i)_{y_i}]$ and

$$\Lambda(h) = \left(1 + \sqrt{2(L\log 2 + \sum_{l=1}^{L-1} \log(\max_{j \in [d_l]} |pred(l,j)|) + \log C)}\right) \cdot \sqrt{\max_{j_0,\ldots,j_L} \prod_{l=1}^{L-1} |pred(l,j_l)| \cdot \sum_{i=1}^n \|z_{j_0}^0(x_i)\|_2^2}.$$

Theorem S6 (Test set bound of Langford [2005])**.** *Let $X_1, \ldots, X_n$ be* i.i.d. *random variables $X_i \in \{0,1\}$ and $p = \mathbb{E}[X_i]$. Then, with probability at least $1-\delta$, we have*

$$p \leq \overline{\text{Bin}}\left(\sum_{i=1}^n X_i, n, \delta\right),$$

with $\text{Bin}(k,m,p) = \sum_{i=0}^k \binom{m}{i} p^i (1-p)^{m-i}$ and $\overline{\text{Bin}}(k,m,\delta) = \text{argsup}_{p \in [0,1]}\{\text{Bin}(k,m,p) \geq \delta\}$.

Theorem S7 (Chernoff bound of Foong et al. [2022])**.** *Let $X_1, \ldots, X_n$ be* i.i.d. *random variables $X_i \in [0,1]$ and $p = \mathbb{E}[X_i]$. Then, with probability at least $1-\delta$, we have*

$$p \leq \text{kl}^{-1}\left(\frac{1}{n}\sum_{i=1}^n X_i, \frac{1}{n}\log\frac{1}{\delta}\right).$$

with $\text{kl}(q,p) = q\log\frac{q}{p} + (1-q)\log\frac{1-q}{1-p}$ and $\text{kl}^{-1}(q,\epsilon) = \text{argsup}_{p \in [0,1]}\left\{\text{kl}(q,p) \leq \epsilon\right\}$.

Theorem S8 (Test set bound of Langford [2005], adapted for random coresets)**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any predictor $h \in \mathcal{H}$, for any random sequence $\mathbf{i} \in \mathcal{P}(n)$ and for any $\delta \in (0,1]$, with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$R_{\mathcal{D}}(A(S_{\mathbf{i}})) \leq \overline{\text{Bin}}\Big((n - |\mathbf{i}|)\widehat{R}_{S_{\mathbf{i}^c}}(A(S_{\mathbf{i}})), n - |\mathbf{i}|, \delta\Big).$$

Theorem S9 (Sample compression bound of Laviolette et al. [2005])**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any deterministic reconstruction function $\mathcal{R}$ that outputs sample-compressed predictors $h \in \mathcal{H}_S$ and for any $\delta \in (0,1]$, with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$\forall\, \mathbf{i} \in \mathcal{P}(n) : R_{\mathcal{D}}(\mathcal{R}(S_{\mathbf{i}})) \leq \overline{\text{Bin}}\Big(|\mathbf{i}^c|\widehat{R}_{S_{\mathbf{i}^c}}(\mathcal{R}(S_{\mathbf{i}})), |\mathbf{i}^c|, \binom{n}{|\mathbf{i}|}^{-1} \tfrac{6}{\pi^2}(|\mathbf{i}|+1)^{-2}\delta\Big).$$

Theorem S10 (P2L bound of Paccagnan et al. [2025])**.** *For a fixed compression set size M, let $\mathcal{R}(S_{\mathbf{i}})$ be the output of P2L which stops when $|\mathbf{i}| = M$. For any $\delta \in (0,1)$, with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$R_{\mathcal{D}}(\mathcal{R}(S_{\mathbf{i}})) \;\leq\; \overline{\varepsilon}\Big(M + |\,\mathbf{i}^c\,|\widehat{R}_{S_{\mathbf{i}^c}}(\mathcal{R}(S_{\mathbf{i}})),\; \delta\Big),$$

where $\overline{\varepsilon}(n,\delta) = 1$ and for $k = 0, 1, \ldots, n-1$, $\overline{\varepsilon}(k,\delta)$ is the unique solution to the equation $\Psi_{k,\delta}(\varepsilon) = 1$ in the interval $[\frac{k}{n}, 1]$, with

$$\Psi_{k,\delta}(\varepsilon) = \frac{\delta}{n}\sum_{m=k}^{n-1} \frac{\binom{m}{k}}{\binom{n}{k}}(1-\varepsilon)^{-(n-m)}.$$

Theorem S11 (Model compression bound, adapted from Lotfi et al. [2022])**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any prefix-free code $\mathcal{C}$, for any hypothesis class $\mathcal{H}_{\mathcal{C}}$ such that any $\hat{h} \in \mathcal{H}_{\mathcal{C}}$ can be defined using the code $\mathcal{C}$ and for any $\delta \in (0,1]$, with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$\forall \hat{h} \in \mathcal{H}_{\mathcal{C}} : R_{\mathcal{D}}(\hat{h}) \leq \overline{\text{Bin}}\Big(n\widehat{R}_S(\hat{h}), n, 2^{-l_{\mathcal{C}}(\hat{h})}2^{-2\log_2 l_{\mathcal{C}}(\hat{h})}\delta\Big).$$

Proof. Using the prior $p(h) = \frac{1}{Z} 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})}$ of Lotfi et al. [2022], the test set bound of Langford [2005] and the union bound, we get

$$\forall \hat{h} \in \mathcal{H}_{\mathcal{C}} : R_{\mathcal{D}}(\hat{h}) \leq \overline{\text{Bin}}\left(\kappa, n, \frac{1}{Z} 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})} \delta\right). \tag{6}$$

The binomial tail inversion is decreasing with respect to δ, as a smaller δ means a looser bound. By definition, $Z \leq 1$, which means that

$$\begin{aligned} &\frac{1}{Z} 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})} \geq 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})} \\ \implies &\overline{\text{Bin}}\left(\kappa, n, \frac{1}{Z} 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})} \delta\right) \leq \overline{\text{Bin}}\left(\kappa, n, 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})} \delta\right). \end{aligned}$$

□

Theorem S12 (Chernoff test-set bound of Foong et al. [2022], adapted for random coresets)**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any predictor $h \in \mathcal{H}$, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$, for any random sequence $\mathbf{i} \in \mathcal{P}(n)$ and for any $\delta \in (0, 1]$, with probability at least $1 - \delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$\mathcal{L}_{\mathcal{D}}(A(S_{\mathbf{i}})) \leq B_\ell + \lambda_\ell \text{kl}^{-1}\left(\frac{\widehat{\mathcal{L}}_{S_{\mathbf{i}^c}}(A(S_{\mathbf{i}})) - B_\ell}{\lambda_\ell}, \frac{1}{n - |\mathbf{i}|} \log \frac{1}{\delta}\right).$$

Theorem S13 (Sample-compression bound of Bazinet et al. [2025])**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any deterministic reconstruction function $\mathcal{R}$ that outputs sample-compressed predictors $h \in \mathcal{H}_S$, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ and for any $\delta \in (0, 1]$, with probability at least $1 - \delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$\forall \mathbf{i} \in \mathcal{P}(n) : \mathcal{L}_{\mathcal{D}}(\mathcal{R}(S_{\mathbf{i}})) \leq B_\ell + \lambda_\ell \text{kl}^{-1}\left(\frac{\widehat{\mathcal{L}}_{S_{\mathbf{i}^c}}(\mathcal{R}(S_{\mathbf{i}})) - B_\ell}{\lambda_\ell}, \frac{1}{|\mathbf{i}^c|}\left[\log \binom{n}{|\mathbf{i}|} + \log\left(\frac{2\sqrt{|\mathbf{i}^c|}}{\frac{6}{\pi^2}(|\mathbf{i}|+1)^{-2}\delta}\right)\right]\right).$$

Theorem S14 (Model compression bound, adapted from Lotfi et al. [2024])**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any prefix-free code $\mathcal{C}$, for any hypothesis class $\mathcal{H}_{\mathcal{C}}$ such that any $\hat{h} \in \mathcal{H}_{\mathcal{C}}$ can be defined using the code $\mathcal{C}$, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ and for any $\delta \in (0, 1]$, with probability at least $1 - \delta$ over the draw of $S \sim \mathcal{D}^n$, we have*

$$\forall \hat{h} \in \mathcal{H}_{\mathcal{C}} : \mathcal{L}_{\mathcal{D}}(\hat{h}) \leq B_\ell + \lambda_\ell \text{kl}^{-1}\left(\frac{\widehat{\mathcal{L}}_S(\hat{h}) - B_\ell}{\lambda_\ell}, \frac{1}{n}\left[l_{\mathcal{C}}(\hat{h}) \log 2 + 2\log l_{\mathcal{C}}(\hat{h}) + \log\left(\frac{2\sqrt{n}}{\delta}\right)\right]\right).$$

Proof. We use the same proof technique as Bazinet et al. [2025], but we replace the sample-compression prior with $p(h) = \frac{1}{Z} 2^{-l_{\mathcal{C}}(\hat{h})} 2^{-2\log_2 l_{\mathcal{C}}(\hat{h})}$. Similarly to the proof of Theorem S11, we can avoid computing the normalizing constant Z as $\text{kl}^{-1}(q, \epsilon)$ is increasing in ϵ and, with $Z \leq 1$, we have

$$-\log p(h) = l_{\mathcal{C}}(\hat{h}) \log 2 + 2\log l_{\mathcal{C}}(\hat{h}) + \log Z \leq l_{\mathcal{C}}(\hat{h}) \log 2 + 2\log l_{\mathcal{C}}(\hat{h}).$$

□

Theorem S15 (PAC-Bayes bound of Pérez-Ortiz et al. [2021])**.** *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any set $\mathcal{H}$ of predictors $h : \mathcal{X} \to \mathcal{Y}$, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$, for any dataset-independent prior distribution P on $\mathcal{H}$, for any $\delta, \delta' \in (0, 1]$, with probability at least $1 - \delta - \delta'$ over the draw of $S \sim \mathcal{D}^n$ and a set of V predictors $h_1, \ldots, h_V \sim Q$, where Q is a dataset-dependent posterior distribution over $\mathcal{H}$, we have*

$$\mathop{\mathbb{E}}_{h \sim Q} \mathcal{L}_{\mathcal{D}}(h) \leq B_\ell + \lambda_\ell \text{kl}^{-1}\left(\text{kl}^{-1}\left(\frac{1}{V}\sum_{i=1}^{V} \frac{\widehat{\mathcal{L}}_S(h_i) - B_\ell}{\lambda_\ell}, \frac{1}{V} \log \frac{2}{\delta'}\right), \frac{1}{n}\left[\text{KL}(Q||P) + \ln\left(\frac{2\sqrt{n}}{\delta}\right)\right]\right).$$

B PROOFS OF THE MAIN RESULTS

Theorem 5. *For two predictors $h \in \overline{\mathcal{H}}$ and $f \in \mathcal{H}$, with probability at least $1-\delta$ over the sampling of $U \sim \mathcal{D}^m$, we have*

$$R_{\mathcal{D}}(f) \leq R_{\mathcal{D}}(h) + \overline{\text{Bin}}\big(m d_U^{0\text{-}1}(f,h), m, \delta\big),$$

with $\overline{\text{Bin}}(k,m,\delta) = \text{argsup}_{p\in[0,1]}\{\text{Bin}(k,m,p) \geq \delta\}$ *and* $\text{Bin}(k,m,p) = \sum_{i=0}^{k} \binom{m}{i} p^i (1-p)^{m-i}$.

Proof. Given two predictors $h, f \in \mathcal{H}$ such that $h(\boldsymbol{x}) = (h(\boldsymbol{x})_1, \ldots, h(\boldsymbol{x})_C)$ and $f(\boldsymbol{x}) = (f(\boldsymbol{x})_1, \ldots, f(\boldsymbol{x})_C)$. We use $f_{\max}(\boldsymbol{x}) = \text{argmax}_j\, f(\boldsymbol{x})_j$ and $h_{\max}(\boldsymbol{x}) = \text{argmax}_k\, h(\boldsymbol{x})_k$ to simplify the notation. Then, the result of Yang et al. [2024] states that

$$|R_{\mathcal{D}}(f) - R_{\mathcal{D}}(h)| \leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})\right]. \tag{7}$$

For completeness, we restate the proof of this result before continuing on to prove our result.

$$\begin{aligned}
|R_{\mathcal{D}}(f) - R_{\mathcal{D}}(h)| &= \left| \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq y\right] - \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \mathbb{I}\left[h_{\max}(\boldsymbol{x}) \neq y\right] \right| \\
&= \left| \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq y\right] - \mathbb{I}\left[h_{\max}(\boldsymbol{x}) \neq y\right]\right] \right| \\
&\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left| \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq y\right] - \mathbb{I}\left[h_{\max}(\boldsymbol{x}) \neq y\right] \right| \\
&\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})\right].
\end{aligned}$$

The first inequality is the triangle inequality. The second inequality can easily be shown with a truth table (see Table S1).

$\mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq y\right]$	$\mathbb{I}\left[h_{\max}(\boldsymbol{x}) \neq y\right]$	$\left\lvert \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq y\right] - \mathbb{I}\left[h_{\max}(\boldsymbol{x}) \neq y\right] \right\rvert$	$\mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})\right]$
0	0	0	0
1	0	1	1
0	1	1	1
1	1	0	0 or 1

Table S1: Truth table (0 for false, 1 for true). Let $\mathcal{Y} = \{1, \ldots, C\}$ the set of classes. First line: if both $f_{\max}(\boldsymbol{x}) = y$ and $h_{\max}(\boldsymbol{x}) = y$, then $f_{\max}(\boldsymbol{x}) = h_{\max}(\boldsymbol{x})$. Second line (and similarly for the third line): if $f_{\max}(\boldsymbol{x}) \neq y$ and $h_{\max}(\boldsymbol{x}) = y$, then $f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})$. Without loss of generality, suppose $y = 1$. Then, for the last line, if $f_{\max}(\boldsymbol{x}) \neq y$ and $h_{\max}(\boldsymbol{x}) \neq y$, then we have $f_{\max}(\boldsymbol{x}) \in \{2, \ldots, C\}$ and $h_{\max}(\boldsymbol{x}) \in \{2, \ldots, C\}$. It is then both possible to have $f_{\max}(\boldsymbol{x}) = h_{\max}(\boldsymbol{x}) \implies \mathbb{I}[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})] = 0$ and $f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x}) \implies \mathbb{I}[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})] = 1$.

Equation 7 is not computable because it requires knowledge of the true data distribution $\mathcal{D}$. We now extend this result and upperbound the disagreement. To do so, we start by applying the binomial test-set bound of Langford (2005) (see Theorem S6) on the right-hand side of Equation (7), with $p = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}} \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})\right]$.

$$\begin{aligned}
1-\delta &\leq \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m} \left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \mathbb{I}\left[f_{\max}(\boldsymbol{x}) \neq h_{\max}(\boldsymbol{x})\right] \leq \overline{\text{Bin}}\left(\sum_{i=1}^{m} \mathbb{I}\left[f_{\max}(\boldsymbol{x}_i) \neq h_{\max}(\boldsymbol{x}_i)\right], m, \delta \right) \right) \\
&\leq \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m} \left(|R_{\mathcal{D}}(f) - R_{\mathcal{D}}(h)| \leq \overline{\text{Bin}}\left(\sum_{i=1}^{m} \mathbb{I}\left[f_{\max}(\boldsymbol{x}_i) \neq h_{\max}(\boldsymbol{x}_i)\right], m, \delta \right) \right) \qquad \text{(Equation 7)} \\
&= \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m} \left(|R_{\mathcal{D}}(f) - R_{\mathcal{D}}(h)| \leq \overline{\text{Bin}}\big(m d_U^{0\text{-}1}(f,h), m, \delta\big) \right) \\
&\leq \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m} \left(R_{\mathcal{D}}(f) \leq R_{\mathcal{D}}(h) + \overline{\text{Bin}}\big(m d_U^{0\text{-}1}(f,h), m, \delta\big) \right)
\end{aligned}$$

with

$$d_U^{0\text{-}1}(f,h) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{I}\left[\mathop{\text{argmax}}_j\; f(\boldsymbol{x}_i)_j \neq \mathop{\text{argmax}}_k\; h(\boldsymbol{x}_i)_k \right].$$

□

For the following theorem, we make the assumption that the loss natively applies a softmax over the output of the predictors. Otherwise, we simply consider $h'(\cdot) = \boldsymbol{\sigma}(h(\cdot))$ in the proof.

Theorem 6. *For a distribution Q over $\mathcal{H}$, a predictor $f \in \mathcal{H}$, a Lipschitz loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ with a Lipschitz constant K_ℓ, with probability at least $1-\delta$ over the sampling of $U \sim \mathcal{D}^m$, we have*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + 2K_\ell \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{d_U^{K_\ell}(f,h)}{2}, \frac{\log\frac{1}{\delta}}{m}\right)$$

with $\mathrm{kl}^{-1}(q,\epsilon) = \mathrm{argsup}_{p\in[0,1]}\{\mathrm{kl}(q,p) \leq \epsilon\}$ *and* $\mathrm{kl}(q,p) = q\log\frac{q}{p} + (1-q)\log\frac{1-q}{1-p}$.

Proof of Theorem 6. With $\boldsymbol{\sigma}(f(\boldsymbol{x})) = (\sigma(f(\boldsymbol{x}),1),\ldots,\sigma(f(\boldsymbol{x}),C))$, we have

$$\begin{aligned}
\left|\mathcal{L}_{\mathcal{D}}(f) - \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h)\right| &= \left|\mathop{\mathbb{E}}_{h\sim Q}[\mathcal{L}_{\mathcal{D}}(f) - \mathcal{L}_{\mathcal{D}}(h)]\right| \\
&= \left|\mathop{\mathbb{E}}_{h\sim Q}\left[\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\ell(f(\boldsymbol{x}),y) - \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\ell(h(\boldsymbol{x}),y)\right]\right| \\
&= \left|\mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)]\right| \\
&\leq \mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)| \\
&\leq \mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} K_\ell\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \\
&= K_\ell \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1.
\end{aligned} \tag{8}$$

The first inequality stems from the triangle inequality. The second inequality comes from the K_ℓ-Lipschitz continuity of ℓ.

The 1-norm of the difference of two softmax distributions is always positive and always bounded by 2, which can easily be proved using the triangle inequality.

$$\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \leq \|\boldsymbol{\sigma}(f(\boldsymbol{x}))\|_1 + \|\boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \leq 2.$$

From Chernoff's bound for random variables in the unit interval [Foong et al., 2022], with $p = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathbb{E}_{h\sim Q}\frac{1}{2}\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1$, we have

$$\mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{2}\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \leq \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{m}\sum_{i=1}^m\frac{1}{2}\|\boldsymbol{\sigma}(f(\boldsymbol{x}_i)) - \boldsymbol{\sigma}(h(\boldsymbol{x}_i))\|_1, \frac{1}{m}\log\frac{1}{\delta}\right)\right) \geq 1-\delta.$$

Let the difference between the losses be denoted by

$$d_U^{K_\ell}(f,h) = \|\boldsymbol{\sigma}(f(\boldsymbol{x}_i)) - \boldsymbol{\sigma}(h(\boldsymbol{x}_i))\|_1.$$

We finish the proof.

$$\begin{aligned}
1-\delta &\leq \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{2}\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \leq \mathrm{kl}^{-1}\left(\frac{1}{2}\mathop{\mathbb{E}}_{h\sim Q}d_U^{K_\ell}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&= \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \leq 2\mathrm{kl}^{-1}\left(\frac{1}{2}\mathop{\mathbb{E}}_{h\sim Q}d_U^{K_\ell}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&= \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}K_\ell\|\boldsymbol{\sigma}(f(\boldsymbol{x})) - \boldsymbol{\sigma}(h(\boldsymbol{x}))\|_1 \leq 2K_\ell\mathrm{kl}^{-1}\left(\frac{1}{2}\mathop{\mathbb{E}}_{h\sim Q}d_U^{K_\ell}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&\leq \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)| \leq 2K_\ell\mathrm{kl}^{-1}\left(\frac{1}{2}\mathop{\mathbb{E}}_{h\sim Q}d_U^{K_\ell}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&\leq \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\left|\mathcal{L}_{\mathcal{D}}(f) - \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h)\right| \leq 2K_\ell\mathrm{kl}^{-1}\left(\frac{1}{2}\mathop{\mathbb{E}}_{h\sim Q}d_U^{K_\ell}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right)
\end{aligned}$$

$$= \mathop{\mathbb{P}}_{U\sim\mathcal{D}^m}\left(\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h) + 2K_\ell \mathrm{kl}^{-1}\left(\frac{1}{2}\mathop{\mathbb{E}}_{h\sim Q} d_U^{K_\ell}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right).$$

□

Theorem 7. *For a distribution Q over $\mathcal{H}$, a predictor $f \in \mathcal{H}$, a loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [B_\ell, T_\ell]$ with range $\lambda_\ell = T_\ell - B_\ell$, with probability at least $1-\delta$ over the sampling of $L \sim \mathcal{D}^m$:*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h) + \lambda_\ell \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{d_L(f,h)}{\lambda_\ell}, \frac{\log\frac{1}{\delta}}{m}\right),$$

with

$$d_L(f,h) = \frac{1}{m}\sum_{i=1}^m \left|\ell(f(\boldsymbol{x}_i), y_i) - \ell(h(\boldsymbol{x}_i), y_i)\right|.$$

Proof of Theorem 7.

$$\begin{aligned}
\left|\mathcal{L}_{\mathcal{D}}(f) - \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h)\right| &= \left|\mathop{\mathbb{E}}_{h\sim Q}\left[\mathcal{L}_{\mathcal{D}}(f) - \mathcal{L}_{\mathcal{D}}(h)\right]\right| \\
&= \left|\mathop{\mathbb{E}}_{h\sim Q}\left[\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\ell(f(\boldsymbol{x}),y) - \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\ell(h(\boldsymbol{x}),y)\right]\right| \\
&= \left|\mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)\right]\right| \\
&\leq \mathop{\mathbb{E}}_{h\sim Q}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)\right| \\
&= \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\left|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)\right|
\end{aligned}$$

The inequality stems from the triangle inequality.

To use Chernoff's bound, we need a loss in the unit interval, so we normalize it.

$$0 \leq \frac{|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)|}{\max_{y,y'}\ell(y',y) - \min_{y,y'}\ell(y',y)} = \frac{|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)|}{T_\ell - B_\ell} \leq 1.$$

We set $\lambda_\ell \coloneqq T_\ell - B_\ell$. Then, from Chernoff's bound for random variables in the unit interval [Foong et al., 2022], with $p = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathbb{E}_{h\sim Q}\frac{1}{\lambda_\ell}\left|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)\right|$, we have

$$\mathop{\mathbb{P}}_{L\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\frac{|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)|}{\lambda_\ell} \leq \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{m}\sum_{i=1}^m\frac{|\ell(f(\boldsymbol{x}_i),y_i) - \ell(h(\boldsymbol{x}_i),y_i)|}{\lambda_\ell}, \frac{1}{m}\log\frac{1}{\delta}\right)\right) \geq 1-\delta.$$

Let the difference between the losses be denoted by

$$d_L(f,h) = \frac{1}{m}\sum_{i=1}^m \left|\ell(f(\boldsymbol{x}_i), y_i) - \ell(h(\boldsymbol{x}_i), y_i)\right|.$$

We finish the proof.

$$\begin{aligned}
1-\delta &\leq \mathop{\mathbb{P}}_{L\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}\frac{|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)|}{\lambda_\ell} \leq \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{\lambda_\ell}d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&= \mathop{\mathbb{P}}_{L\sim\mathcal{D}^m}\left(\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{h\sim Q}|\ell(f(\boldsymbol{x}),y) - \ell(h(\boldsymbol{x}),y)| \leq \lambda_\ell\mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{\lambda_\ell}d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&\leq \mathop{\mathbb{P}}_{L\sim\mathcal{D}^m}\left(\left|\mathcal{L}_{\mathcal{D}}(f) - \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h)\right| \leq \lambda_\ell\mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{\lambda_\ell}d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right) \\
&= \mathop{\mathbb{P}}_{L\sim\mathcal{D}^m}\left(\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q}\mathcal{L}_{\mathcal{D}}(h) + \lambda_\ell\mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q}\frac{1}{\lambda_\ell}d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\right).
\end{aligned}$$

□

Corollary S16. *In the setting of Theorem 7, for a Lipschitz continuous loss function ℓ with constant K_ℓ, with probability at least $1-\delta$ over the sampling of $L \sim \mathcal{D}^m$, we have :*

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}(f) &\leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \lambda_\ell \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\lambda_\ell} d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right) \\ &\leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \lambda_\ell \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{K_\ell}{\lambda_\ell} \widehat{d_L}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\end{aligned}$$

with

$$\widehat{d_L}(f,h) = \frac{1}{m}\sum_{i=1}^{m} \|\sigma(f(\boldsymbol{x}_i), y_i) - \sigma(h(\boldsymbol{x}_i), y_i)\|_1.$$

Proof. Following from the Lipschitz-continuity of ℓ, we have

$$d_L(f,h) = \frac{1}{m}\sum_{i=1}^{m} |\ell(f(\boldsymbol{x}_i), y_i) - \ell(h(\boldsymbol{x}_i), y_i)| \leq \frac{K_\ell}{m}\sum_{i=1}^{m} \|\sigma(f(\boldsymbol{x}_i), y_i) - \sigma(h(\boldsymbol{x}_i), y_i)\|_1.$$

Moreover, as $\mathrm{kl}^{-1}(q,\epsilon)$ is monotonically increasing in q, we have :

$$\mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\lambda_\ell} d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right) \leq \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{K_\ell}{\lambda_\ell}\frac{1}{m}\sum_{i=1}^{m} \|\sigma(f(\boldsymbol{x}_i), y_i) - \sigma(h(\boldsymbol{x}_i), y_i)\|_1, \frac{1}{m}\log\frac{1}{\delta}\right).$$

□

C ADDITIONAL RESULTS

C.1 SPECIAL CASES OF THE MAIN RESULTS

Corollary S17. *In the setting of Theorem 7, with $\alpha \in (0,1)$, $\beta = \frac{C}{\alpha}$, the clamped softmax (see Definition S1) and the clamped cross-entropy loss $\ell(g(\boldsymbol{x}), y) = -\ln(\sigma_{\max}(g(\boldsymbol{x}), y))$, with probability at least $1-\delta$ over the sampling of $L \sim \mathcal{D}^m$, we have*

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}(f) &\leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \ln(\beta)\, \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\ln(\beta)} d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right) \\ &\leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \ln(\beta)\, \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{\beta}{\ln(\beta)} \widehat{d_L}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\end{aligned}$$

with

$$\widehat{d_L}(f,h) = \frac{1}{m}\sum_{i=1}^{m} \|\sigma_{\max}(f(\boldsymbol{x}_i), y_i) - \sigma_{\max}(h(\boldsymbol{x}_i), y_i)\|_1.$$

Proof. This corollary is an application of Theorem 7 and Corollary S16 to the clamped cross-entropy loss. In Corollary S16, we want to highlight the difference between the clamped softmax of f and h. Thus, we consider the loss ℓ as simply $-\ln(u)$, with $u \in (\frac{\alpha}{C}, 1)$.

We now find the Lipschitz constant K_ℓ by computing the gradient of ℓ.

$$K_\ell = \sup_{u\in(\frac{\alpha}{C},1)} \left|\frac{\mathrm{d}(-\ln(u))}{\mathrm{d}u}\right| = \sup_{u\in(\frac{\alpha}{C},1)} \left|\frac{\mathrm{d}\ln(u)}{\mathrm{d}u}\right| = \sup_{u\in(\frac{\alpha}{C},1)} \left|\frac{1}{u}\right| = \frac{C}{\alpha}.$$

To simplify the notation in the theorem, we choose $\beta = \frac{C}{\alpha}$. □

Corollary S18. *In the setting of Theorem 7, with $\alpha \in (0,1)$, $\beta = \frac{C}{\alpha}$, the smoothed softmax (see Definition S2) and the smoothed cross-entropy loss $\ell(g(\boldsymbol{x}), y) = -\ln(\sigma_{\text{smooth}}(g(\boldsymbol{x}), y))$, with probability at least $1-\delta$ over the sampling of $L \sim \mathcal{D}^m$, we have*

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}(f) &\leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \ln\left(1+(1-\alpha)\beta\right) \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\ln\left(1+(1-\alpha)\beta\right)} d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right) \\ &\leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \ln\left(1+(1-\alpha)\beta\right) \mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{\beta}{\ln\left(1+(1-\alpha)\beta\right)} \widehat{d_L}(f,h), \frac{1}{m}\log\frac{1}{\delta}\right)\end{aligned}$$

with

$$\widehat{d_L}(f,h) = \frac{1}{m}\sum_{i=1}^{m} \|\sigma_{\text{smooth}}(f(\boldsymbol{x}_i), y_i) - \sigma_{\text{smooth}}(h(\boldsymbol{x}_i), y_i)\|_1.$$

Proof. This corollary is an application of Theorem 7 and Corollary S16 to the clamped cross-entropy loss. In Corollary S16, we want to highlight the difference between the smoothed softmax of f and h. Thus, we consider the loss ℓ as simply $-\ln(u)$, with $u \in \left(\frac{\alpha}{C}, 1-\alpha+\frac{\alpha}{C}\right)$.

We now find the Lipschitz constant K_ℓ by computing the gradient of ℓ.

$$K_\ell = \sup_{u\in\left(\frac{\alpha}{C}, 1-\alpha+\frac{\alpha}{C}\right)} \left|\frac{\mathrm{d}(-\ln(u))}{\mathrm{d}u}\right| = \sup_{u\in\left(\frac{\alpha}{C}, 1-\alpha+\frac{\alpha}{C}\right)} \left|\frac{\mathrm{d}\ln(u)}{\mathrm{d}u}\right| = \sup_{u\in\left(\frac{\alpha}{C}, 1-\alpha+\frac{\alpha}{C}\right)} \left|\frac{1}{u}\right| = \frac{C}{\alpha}.$$

To simplify the notation in the theorem, we choose $\beta = \frac{C}{\alpha}$. □

C.2 COMPUTABLE PAC-BAYESIAN DISAGREEMENT BOUNDS

To compute the bounds exactly, we need to use the same trick as in Theorem S15, that is we approximate the sampling of the hypothesis from the posterior Q via Monte Carlo Sampling. This result can be applied to the zero-one loss, as we have $\lambda_\ell = 1$ and $d_L(\cdot,\cdot) \leq d_U^{0\text{-}1}(\cdot,\cdot)$ by Equation (7), which doesn't require a labeled set L.

Lemma S19. *For any distribution $\mathcal{D}$ over $\mathcal{X}\times\mathcal{Y}$, for any predictor $f \in \mathcal{H}$, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [0,1]$, for any $\delta \in (0,1]$, with probability at least $1-2\delta$ over the draw of $L \sim \mathcal{D}^m$ and a set of V predictors $h_1, \ldots, h_V \sim Q$, where Q is a dataset-dependent posterior distribution over $\mathcal{H}$, we have*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathop{\mathbb{E}}_{h\sim Q} \mathcal{L}_{\mathcal{D}}(h) + \lambda_\ell \mathrm{kl}^{-1}\left(\mathrm{kl}^{-1}\left(\frac{1}{V}\sum_{i=1}^{V} \frac{1}{\lambda_\ell} d_L(f,h_i), \frac{1}{V}\log\frac{1}{\delta}\right), \frac{1}{m}\log\frac{1}{\delta}\right).$$

Proof. When computing Theorem 7 for a Gaussian distribution Q over the predictors, we cannot compute exactly the term

$$\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\lambda_\ell} d_L(f,h)$$

as the expectation doesn't have a closed form. However, similarly to Theorem S15, we can upper-bound it. Instead of using the two-sided Chernoff bound as in Langford and Caruana [2001], Pérez-Ortiz et al. [2021], we use the one-sided Chernoff bound [Foong et al., 2022], as we are explicitly interested in an upper-bound.

With $X_i = \frac{1}{\lambda_\ell} d_L(f,h_i)$ and $p = \mathbb{E}[X_i] = \mathbb{E}_{h\sim Q} \frac{1}{\lambda_\ell} d_L(f,h)$, with probability at least $1-\delta$ over the sampling of $h_1, \ldots, h_V \sim Q$, we have

$$\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\lambda} d_L(f,h) \leq \mathrm{kl}^{-1}\left(\frac{1}{V}\sum_{i=1}^{V} \frac{1}{\lambda_\ell} d_L(f,h_i), \frac{1}{V}\log\frac{1}{\delta}\right).$$

Using the fact that the inverse of the kl is increasing in its first argument, we have

$$\mathrm{kl}^{-1}\left(\mathop{\mathbb{E}}_{h\sim Q} \frac{1}{\lambda_\ell} d_L(f,h), \frac{1}{m}\log\frac{1}{\delta}\right) \leq \mathrm{kl}^{-1}\left(\mathrm{kl}^{-1}\left(\frac{1}{V}\sum_{i=1}^{V} \frac{1}{\lambda_\ell} d_L(f,h_i), \frac{1}{V}\log\frac{1}{\delta}\right), \frac{1}{m}\log\frac{1}{\delta}\right). \tag{9}$$

We finish the proof with a union bound with Theorem 7 and Equation 9. □

Theorem S20. *For any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, for any predictor $f \in \mathcal{H}$, for any set $\mathcal{H}$ of predictors $h : \mathcal{X} \to \mathcal{Y}$, for any loss $\ell : \mathbb{R}^C \times \mathcal{Y} \to [0,1]$, for any dataset-independent prior distribution P on $\mathcal{H}$, for any $\delta \in (0,1]$, with probability at least $1-4\delta$ over the draw of $S \sim \mathcal{D}^n$, $L \sim \mathcal{D}^m$ and a set of $2V$ predictors $h_1, \ldots, h_{2V} \sim Q$, where Q is a dataset-dependent posterior distribution over $\mathcal{H}$, we have*

$$\mathcal{L}_{\mathcal{D}}(f) \leq B_\ell + \lambda_\ell \mathrm{kl}^{-1}\left(\mathrm{kl}^{-1}\left(\frac{1}{V}\sum_{i=1}^{V}\frac{\widehat{\mathcal{L}}_S(h_i) - B_\ell}{\lambda_\ell}, \frac{1}{V}\log\frac{1}{\delta}\right), \frac{1}{n}\left[\mathrm{KL}(\mathcal{Q}_S \,||\, \mathcal{P}) + \ln\left(\frac{2\sqrt{n}}{\delta}\right)\right]\right)$$
$$+ \lambda_\ell \mathrm{kl}^{-1}\left(\mathrm{kl}^{-1}\left(\frac{1}{V}\sum_{i=V+1}^{2V}\frac{1}{\lambda_\ell}d_L(f,h_i), \frac{1}{V}\log\frac{1}{\delta}\right), \frac{1}{m}\log\frac{1}{\delta}\right).$$

Proof. This result simply stems from the union bound of Theorem S15 and Lemma S19. We remove the factor of 2 in the $\frac{1}{m}\log\frac{2}{\delta}$ by using the one-sided Chernoff bound, similarly to the proof of Lemma S19. □

C.3 USING OTHER COMPARATOR FUNCTIONS

Up until now, all the results are presented with the inverse of the kl. This function gives the tightest bounds, but the bounds obtained with this function are not intuitive. For example, it might be beneficial to use a bound with an analytical upper-bound such as the quadratic loss $\Delta_2(q,p) = 2(q-p)^2$ or Catoni's distance $\Delta_C(q,p) = -\ln(1-p(1-e^{-C})) - Cq$. Both are upper-bounded by $\mathrm{kl}(q,p)$, which can be proven by Pinsker's inequality for the quadratic loss and Proposition 2.1 of [Germain et al., 2009] for Catoni's distance. We present the following result, which is very intuitive and has probably been used before. However, we were not able to find it in the literature, which is why we prove it formally here.

Lemma S21. *If $\Delta(q,p) \leq \mathrm{kl}(q,p) \forall q,p \in [0,1]$, for any $\epsilon > 0$, we have $\mathrm{kl}^{-1}(q,\epsilon) \leq \Delta^{-1}(q,\epsilon)$.*

Proof. For any comparator function, we have

$$\begin{aligned}\Delta^{-1}(q,\epsilon) &= \arg\sup_{0\leq p\leq 1}\{\Delta(q,p) \leq \epsilon\} \\ &= \arg\sup_{q\leq p\leq 1}\{\Delta(q,p) \leq \epsilon\}.\end{aligned}$$

Indeed, for most Δ, there exists two solutions, one where $p \leq q$ and one where $q \leq p$. Obviously, as we take the arg-supremum, the second solution is always chosen. Thus, we consider only $q \leq p \leq 1$.

Let $p_1 = \mathrm{kl}^{-1}(q,\epsilon)$ be the solution such that $\mathrm{kl}(q,p_1) = \epsilon$. Let $p_2 = \Delta^{-1}(q,\epsilon)$ be the solution such that $\Delta(q,p_2) = \epsilon$. From the assumption that $\Delta(q,p) \leq \mathrm{kl}(q,p) \forall q,p \in [0,1]$, we have

$$\mathrm{kl}(q,p_1) = \Delta(q,p_2) = \epsilon \wedge \Delta(q,p_1) \leq \mathrm{kl}(q,p_1) \implies \Delta(q,p_1) \leq \Delta(q,p_2).$$

As $\Delta(q,p)$ is increasing for $p \geq q$, we have that :

$$\Delta(q,p_1) \leq \Delta(q,p_2) \implies p_1 \leq p_2 \implies \mathrm{kl}^{-1}(q,\epsilon) \leq \Delta^{-1}(q,\epsilon).$$

□

D EXPERIMENTS

Devices. The experiments were run on three different devices. The target neural networks, the experiments with Pick-To-Learn and the quantization experiments were computed on a computer with Python 3.12.3 and a NVIDIA GeForce RTX 4090. The coreset experiments and the model compression experiments were computed with Python 3.12.4 and a NVIDIA H100 SXM5. Finally, the PAC-Bayesian experiments and the model distillation experiments were computed with Python 3.12.4 and a NVIDIA A100 SXM4.

Libraries. All libraries used can be found within the code here : `https://github.com/GRAAL-Research/Bound-to-Disagree`. Notably, we use DeepCore [Guo et al., 2022] (MIT license), PACTL [Lotfi et al., 2022] (Apache 2.0 License), PAC-Bayes with Backprop [Pérez-Ortiz et al., 2021] (CC-BY 4.0 License), PyTorch [Ansel et al., 2024] (BSD 3-Clause License), Lightning [Falcon and The PyTorch Lightning team, 2019] (Apache 2.0 license), Loralib [Hu et al., 2022] (MIT License), Transformers [Wolf et al., 2020] (Apache 2.0 License), Scikit-Learn [Pedregosa et al., 2011] (BSD 3-Clause License), NumPy [Harris et al., 2020] (NumPy license), Weight and Biases [Biewald, 2020] (MIT License), ScheduleFree [Defazio et al., 2024] (Apache 2.0 License), TorchAO [Or et al., 2025] (BSD 3-Clause License). We also use the KL inversion function of [Viallard et al., 2021], which is distributed under MIT License.

Datasets. The datasets used are the MNIST dataset [LeCun et al., 1998] (MIT License), the CIFAR10 dataset [Krizhevsky et al., 2009] and the Amazon polarity dataset [Zhang et al., 2015] (Apache 2.0 License). There is no explicit license for CIFAR10, but the authors simply ask the user to cite this technical report : Krizhevsky et al. [2009].

D.1 TARGET MODEL

For MNIST, the models were trained using data augmentation for 200 epochs or until the validation error hasn't decreased for 10 epochs. For CIFAR10, the model was trained for 200 epochs. The models on Amazon polarity are trained for 10 epochs or until the validation error hasn't decreased for 2 epochs. The models for MNIST and CIFAR10 are randomly initialized, but for Amazon polarity the models are initialized with the pretrained weights from the Transformers library [Wolf et al., 2020]. In some experiments, to remove the need for learning rate scheduling, we try the optimizers SGDFree and AdamFree from the ScheduleFree library [Defazio et al., 2024] or the COCOB optimizer from Orabona and Tommasi [2017]. Other times, we use the OneCycle learning rate scheduler [Smith and Topin, 2019] implemented in PyTorch.

We present the hyperparameter grids used for each problem.

Hyperparameters for MNIST

- Dropout probability : $\{0.2, 0.5\}$
- Optimizer : { Adam, AdamFree, COCOB }
- Training learning rate : $\{10^{-3}, 10^{-4}\}$
- Weight decay : $\{0.1, 0.01, 0.001\}$

Hyperparameters for CIFAR10

- Model type : { CNN, ResNet18 }
- Dropout probability : 0.0
- Optimizer : SGD
- Learning rate scheduler : OneCycle
- Training learning rate : $\{0.05, 0.01, 0.005\}$
- Weight decay : $\{10^{-3}, 10^{-4}\}$

Hyperparameters for Amazon polarity

- Model type : { DistilBERT, GPT2}
- Max epochs : 2
- Dropout probability : 0.0
- Optimizer : SGD
- Learning rate scheduler : OneCycle
- Training learning rate : 2×10^{-5}

D.2 PARTITION-BASED BOUNDS

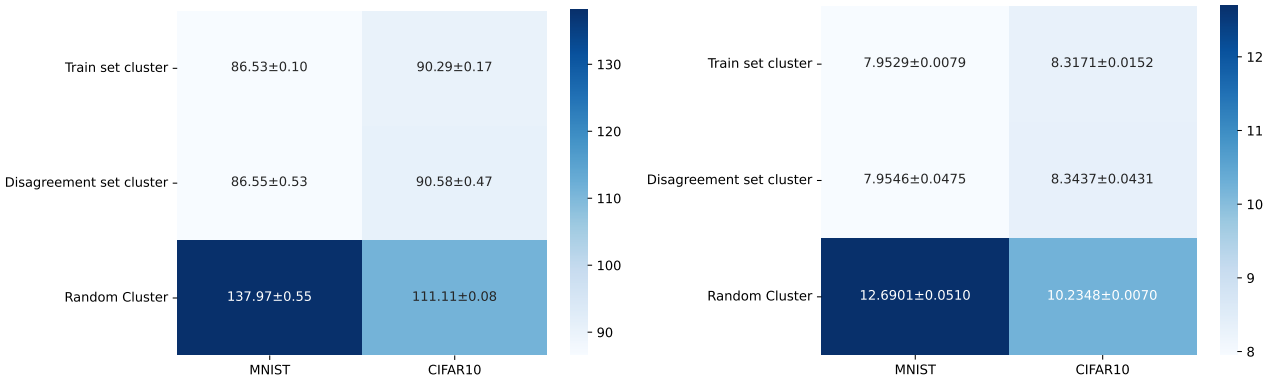
We compute the bound both for the bounded cross-entropy loss and the zero-one loss for all of our target models. For each model and for each loss, we do a grid search over the following hyperparameters :

- Number of subsets K : [5, 10, 20, 50, 100, 200]
- α = $[20, 50, 100, \frac{n(K+n)}{K(4n-3)}]$

Given one hyperparameter combination, we choose $\gamma = (\frac{\delta}{2})^{\frac{-1}{\alpha}}$ such that the bound holds in $1-\gamma^{-\alpha}-\frac{\delta}{2} = 1-((\frac{\delta}{2})^{\frac{-1}{\alpha}})^{-\alpha}-\frac{\delta}{2} = 1-\delta$ with $\delta = 0.01$. We use a union bound over the 24 combinations of the grid search to get a bound that holds simultaneously for all possibilities.

D.2.1 Ablation Study on the Partitioning Method

To compute their bound, Than and Phan [2025] partitions the space using a K-means clustering applied to the training set. However, to the best of our knowledge, this choice of data-dependent partition seems to violate the statement of their theorem. To compute the bound, we try two techniques : computing the partition on the unlabeled disagreement set and using random clusters. As their bound doesn't consider unlabeled data points, the disagreement approach is completely valid. However, this is not the setting of the original paper. For the random clusters, we sample K centroids from a uniform distribution and assign the data points to the cluster of the closest centroid. We repeat this experiments five times and consider the best cluster.



(a) Zero-one loss

(b) Smoothed cross-entropy loss

Figure S1: Comparison of the different approaches to compute the partition-based bounds. We consider 5 different seeds for the random clusters.

Although the train set cluster and the disagreement set cluster were obtained using different datasets, they are almost identical. Moreover, we can see that in their setting (without a disagreement set), the only valid option is to use random clusters, which leads to vacuous generalization bounds. Surprisingly enough, the bounds are actually worse on MNIST than CIFAR10.

D.3 NORM-BASED BOUNDS

We compute the bound of Theorem S5 for each target model by adapting the code provided by [Galanti et al., 2023b]. As suggested in the paper, we use $\gamma = 1$.

D.4 SAMPLE COMPRESSION EXPERIMENTS

We use the Pick-To-Learn with Early Stopping [Marks and Paccagnan, 2025] implementation of Bazinet et al. [2025]. We use the coreset methods implemented by the DeepCore library [Guo et al., 2022]. All approach use Theorem S9 for the zero-one loss and Theorem S13 for the cross-entropy loss. The exceptions include Pick-To-Learn with the zero-one loss, as Pick-To-Learn enjoys the tight generalization bounds of Theorem S10, for which it was designed. The other exception is the Random Coreset approach, which can be used with the following test set bounds : Theorem S8 for the zero-one loss and Theorem S12 for the cross-entropy loss.

We now prove that coreset methods are sample-compression algorithms. Let $\mathcal{C} : \cup_{1\leq n\leq\infty}(\mathcal{X}\times\mathcal{Y})^n \to \cup_{m\leq n}(\mathcal{X}\times\mathcal{Y})^m$ denote some coreset method that takes a dataset S and returns a compression set $S_{\mathbf{i}}$. Let SGD denote stochastic gradient descent. Then, we can apply the sample compression bounds for the predictors outputted by $A : \mathrm{SGD}\circ\mathcal{C}$ with the compression set $S_{\mathbf{i}} = \mathcal{C}(S)$ and the reconstruction function $\mathcal{R} = \mathrm{SGD}$. Then, we have proven that $A(S) = \mathcal{R}(S_{\mathbf{i}}) = \mathrm{SGD}(\mathcal{C}(S))$.

We present the hyperparameter grids.

Hyperparameters for P2L on MNIST

- Dropout probability : $\{0.2, 0.5\}$
- Optimizer : { Adam, AdamFree }
- Training learning rate : $\{10^{-3}, 10^{-4}\}$
- Weight decay : $\{0.01, 0.1\}$
- Smoothing parameter α : $\{10^{-3}, 10^{-4}, 10^{-5}\}$
- Softmax : { Smoothed, Clamped}

Hyperparameters for P2L on CIFAR10

- Model type : { CNN, ResNet18 }
- Dropout probability : 0.5
- Optimizer : { SGD, SGDFree }
- Training learning rate : $\{0.05, 0.01, 0.005\}$
- Weight decay : $\{10^{-3}, 10^{-4}\}$
- Smoothing parameter α : $\{10^{-3}, 10^{-4}, 10^{-5}\}$
- Softmax : { Smoothed, Clamped}

Hyperparameters for coresets on MNIST

- Dropout probability : $\{0.2, 0.5\}$
- Coreset method : { Random Coreset, k-Center Greedy, Forgetting, DeepFool }
- Percentage of dataset used as coreset : $\{0.1\%, 0.5\%, 1\%, 5\%, 10\%, 15\%, 20\%\}$
- Optimizer : { Adam, AdamFree }
- Training learning rate : $\{10^{-3}, 10^{-4}\}$
- Weight decay : $\{0.01, 0.1\}$
- Smoothing parameter α : $\{10^{-3}, 10^{-4}, 10^{-5}\}$
- Softmax : { Smoothed, Clamped}

Hyperparameters for coresets on CIFAR10

- Model type : ResNet18
- Dropout probability : 0.5
- Coreset method : { Random Coreset, k-Center Greedy, Forgetting, DeepFool }
- Percentage of dataset used as coreset : $\{0.1\%, 0.5\%, 1\%, 5\%, 10\%, 15\%, 20\%, 30\%, 40\%, 50\%\}$
- Optimizer : { SGD, SGDFree }
- Training learning rate : $\{0.01, 0.05\}$
- Weight decay : 5×10^{-4}
- Smoothing parameter α : $\{10^{-3}, 10^{-4}\}$
- Softmax : { Smoothed, Clamped}

We report the results for the sample compression experiments on CIFAR10 in the following Figure S2.

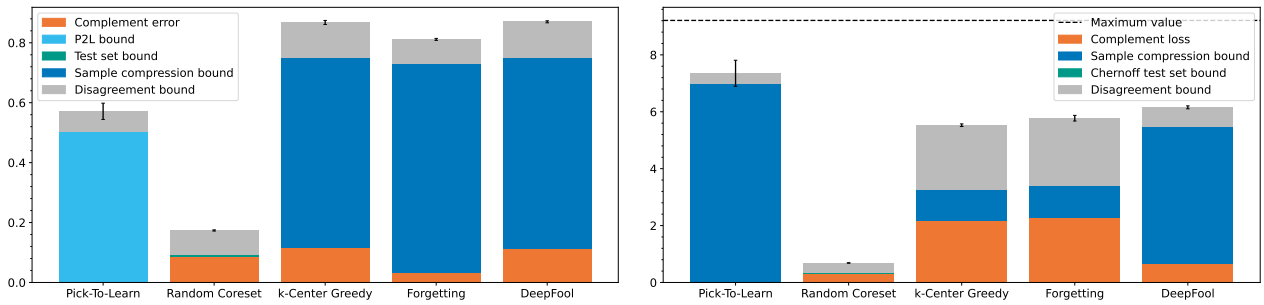
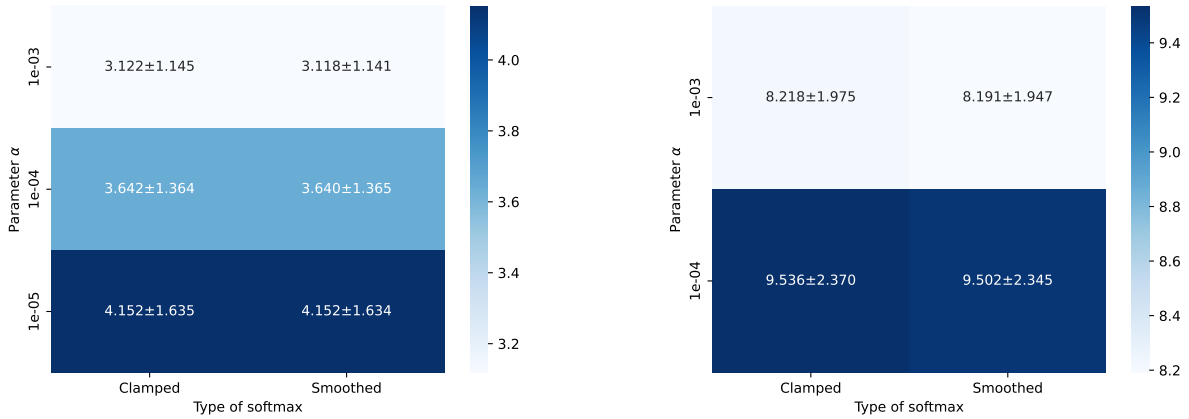


Figure S2: Illustration of the behavior of our disagreement bounds on CIFAR10 using sample-compression methods.

D.4.1 Ablation Study for the Bounded Cross-Entropy Loss

To choose which hyperparameter to use for the bounded cross-entropy loss, we compare the average cross-entropy loss obtained over all the combinations of hyperparameters for the coreset methods. After computing all possible values, we take the mean over all dimensions except the smoothing parameter α and the softmax type : clamped or smoothed. We report the results in Figure S3.

For both experiments, the optimal parameter seems to be $\alpha = 10^{-3}$. The gap between all values of α are not significant, as their standard deviation overlaps. However, on the 448 combinations of hyperparameters for MNIST (with 5 seeds per combinations) and 320 combinations for CIFAR10 (with 5 seeds per combinations), it seems like the parameter $\alpha = 10^{-3}$ leads to the best result. As for the type of softmax, the smoothed softmax seems to lead to slightly better results. Thus, we choose the smoothed softmax with $\alpha = 10^{-3}$ in all other experiments.



(a) MNIST

(b) CIFAR10

Figure S3: Ablation study on the hyperparameters of the bounded cross-entropy loss. We report the mean of the cross-entropy loss over all coreset methods experiments.

D.4.2 Ablation Study for the Size of the Disagreement Set

Although acquiring unlabeled data is generally easier and cheaper than acquiring labeled data, the cost of generating a large unlabeled set of data can still be prohibitive. Thus, in Figure S4, we study the effect of the size of the disagreement set on the disagreement bound of Theorem 5 across 20 subset sizes and five seeds. To do so, we train CNNs on MNIST with random coresets. The original disagreement set contains 10800 data points. The smallest subset contains 1% of the disagreement set, comprising 108 data points.

We observe that when the disagreement set is smaller than 1000 data points, the bound has high variance. However, when the number of data points becomes greater than a thousand, the improvement in the tightness of the bound becomes marginal. From this small experiment, we argue that the minimum number of data points needed to obtain tight bounds would be between a thousand and two thousand data points.

D.5 MODEL COMPRESSION EXPERIMENTS

For the model compression experiments, we pretrain randomly initialized models on a subset of the training set, over which we then add LoRA adapters implemented by Hu et al. [2022] and the subspace compression approach implemented by Lotfi et al. [2022]. This SubLoRA model is then trained on the remaining data points, where the bound is also computed.

Hyperparameters for MNIST

- Portion of training set for pretraining : 20%
- Number of pretraining epochs : 100
- Pretraining learning rate : 10^{-3}
- Dropout probability : 0.0
- Optimizer : AdamFree
- Training learning rate : $\{10^{-2}, 10^{-4}\}$
- Weight decay : 0.1
- Random subspace projector : Dense
- Dimension of the subspace : $\{100, 500, 1000\}$
- Level of quantization : $\{5, 20\}$
- Rank of LoRA : $\{0, 2, 4\}$

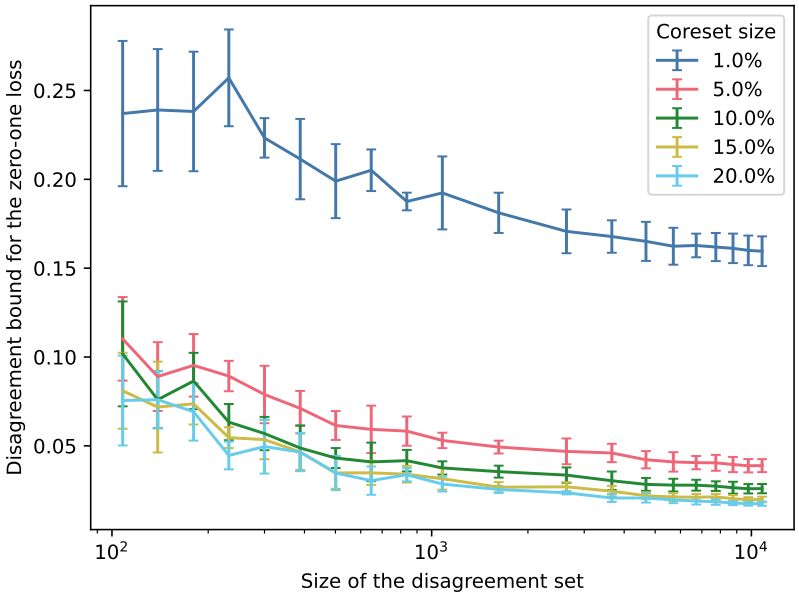


Figure S4: Illustration of the behavior of the disagreement bound with respect to the size of the disagreement set.

Hyperparameters for CIFAR10

- Portion of pretraining set : 50%
- Number of pretraining epochs : 150
- Pretraining learning rate : 0.05
- Dropout probability : 0.2
- Optimizer : SGD
- Learning rate scheduler : OneCycle
- Training learning rate : $\{0.01, 0.005\}$
- Max epochs : $\{500, 1000\}$
- Weight decay : 10^{-4}
- Random subspace projector : { Dense, Rounded Double Kronecker }
- Dimension of the subspace : $\{100, 500\}$
- Level of quantization : $\{5, 10, 20\}$
- Rank of LoRA : $\{0, 2, 4\}$

We report the results for the cross-entropy loss on both MNIST and CIFAR10 in Table S2.

Table S2: Generalization bounds on the smoothed cross-entropy loss achieved on MNIST and CIFAR10 using model compression bounds.

Dataset	Surrogate model			Target model	
	Test loss	Size (KB)	MC Bound	Test loss	Our bound
MNIST	0.0273±0.0015	0.0356±0.0008	0.1338±0.0040	0.0150±0.0006	0.1756±0.0041
CIFAR10	0.7005±0.0243	0.0336±0.0014	1.0978±0.0195	0.2247±0.0060	1.7851±0.0223

D.6 PAC-BAYES EXPERIMENTS

For the PAC-Bayes experiments, we pretrain randomly initialized models on a subset of the training set. We then add Gaussian distributions over each weight, with learnable mean and standard deviation parameters. The stochastic model is then trained on the remaining data points, where the bound is also computed.

Hyperparameters for MNIST

Table S3: Generalization bounds on the smoothed cross-entropy loss achieved on MNIST and CIFAR10 using PAC-Bayes bounds.

Dataset	Surrogate model			Target model	
	Test loss	KL	PB Bound	Test loss	Our bound
MNIST	0.0286±0.0030	0.7425±0.1188	0.1235±0.0059	0.0150±0.0006	0.2592±0.0087
CIFAR10	0.5440±0.0568	1.6239±0.9202	0.7404±0.0744	0.2247±0.0060	1.4204±0.1377

Table S4: Generalization bounds on the zero-one loss achieved via model distillation on Amazon polarity using model compression (MC) bounds. All metrics presented are in percents (%), except the size.

Model	Surrogate model			Target model	
	Test error	Size (KB)	MC Bound	Test error	Our bound
DistilBERT	7.61±0.07	2.02±0.01	14.37±0.08	6.19±0.15	21.35±0.32
GPT2	16.37±0.94	2.04±0.43	25.05±0.91	5.51±0.11	43.34±1.52

- Portion of pretraining set : 20%
- Number of pretraining epochs : 100
- Pretraining learning rate : 10^{-3}
- Dropout probability : 0.0
- Optimizer : Adam
- Training learning rate : $\{10^{-3}, 5 \times 10^{-3}, 10^{-4}\}$
- Weight decay : $\{0.01, 0.1\}$
- Smoothing parameter α : 10^{-3}
- Softmax : { Smoothed, Clamped}
- Standard deviation of prior : $\{10^{-2}, 5 \times 10^{-2}, 10^{-3}, 10^{-4}\}$
- Number of Monte Carlo sampling : 2000

Hyperparameters for CIFAR10

- Portion of pretraining set : $\{60\%, 70\%\}$
- Number of pretraining epochs : 150
- Pretraining learning rate : 0.05
- Dropout probability : 0.2
- Optimizer : SGD
- Training learning rate : $\{10^{-3}, 5 \times 10^{-3}, 10^{-4}\}$
- Weight decay : 10^{-4}
- Smoothing parameter α : 10^{-3}
- Softmax : { Smoothed, Clamped}
- Standard deviation of prior : $\{10^{-2}, 5 \times 10^{-2}, 10^{-3}, 10^{-4}\}$
- Number of Monte Carlo sampling : 5000

D.7 MODEL DISTILLATION ON AMAZON POLARITY

We start with DistilBERT and GPT2 with the pretrained weights of the Transformers library. We freeze the model and apply a SubLoRA model on the top half of the model. For the adaptative quantization, we use the implementation of Lotfi et al. [2022], a learning rate of 10^{-2} for DistilBERT and a learning rate of 10^{-4} for GPT2. We produce pseudo-labels for the unlabeled using the target model and train the model to classify correctly both the labeled set S and the unlabeled set U with the pseudo-labels.

- Max epochs : 5
- Dropout probability : 0.0
- Optimizer : Adam
- Training learning rate : 10^{-6}
- Weight decay : 0.01
- Dimension of the subspace : $\{5000, 10000\}$
- Level of quantization : 250
- Rank of LoRA : 4

D.8 QUANTIZATION EXPERIMENTS ON AMAZON POLARITY

We use the target models and quantize them following this hyperparameter grid. We do not consider the quantization-aware training with 2 bits, as both TorchAO and AO did not consider this setting.

- Max epochs : 2
- Dropout probability : 0.0
- Optimizer : Adam
- Training learning rate : 10^{-6}
- Weight decay : 0.01
- Number of bits : $\{2, 4, 8\}$
- Quantization-aware training : { Yes, No }
- Delta for the Huber loss : 0.2