A PAC-Bayesian Perspective on Structured Prediction with Implicit Loss Embeddings

Théophile Cantelobre, Benjamin Guedj, María Pérez-Ortiz, John Shawe-Taylor

A PAC-Bayesian Perspective on Structured Prediction with Implicit Loss Embeddings

Théophile Cantelobre
Mines ParisTech – PSL, Inria and Sorbonne Université
France

Benjamin Guedj
Inria and University College London
France and United Kingdom

María Pérez-Ortiz
University College London
United Kingdom

John Shawe-Taylor
University College London
United Kingdom

Abstract

Many practical machine learning tasks can be framed as Structured prediction problems, where several output variables are predicted and considered interdependent. Recent theoretical advances in structured prediction have focused on obtaining fast rates convergence guarantees, especially in the Implicit Loss Embedding (ILE) framework. PAC-Bayes has gained interest recently for its capacity of producing tight risk bounds for predictor distributions. This work proposes a novel PAC-Bayes perspective on the ILE Structured prediction framework. We present two generalization bounds, on the risk and excess risk, which yield insights into the behavior of ILE predictors. Two learning algorithms are derived from these bounds. The algorithms are implemented and their behavior analyzed, with source code available at https://github.com/theophilec/PAC-Bayes-ILE-Structured-Prediction.

Keywords: Statistical learning theory, PAC-Bayes theory, Structured output prediction, Implicit Loss Embeddings, Generalization bounds

1. Introduction

Structured prediction (also referred to as Structured output prediction) is a widely studied problem in machine learning of great theoretical and practical interest. Broadly, we define Structured prediction as the joint prediction of interdependent or constrained decision variables.

Although we make this definition precise below, let us first present an example of a recent application of Structured prediction: Scene Graph Generation. Given an image, a goal can be to extract the semantic information that describes the depicted scene, that is, to detect objects present in the image and the semantic relationships between them. Formally, the task consists in predicting a graph of all entities and the relations between then from an image (for example, *the man is wearing a hat*). Note that this graph is on the instance

level, so *the man wearing the hat*, is known to be *the same one holding a bucket* (which *the horse is known to be eating from*).

The decision variable for this task is the scene graph. Seen as a discrete object, its domain is combinatorially large. Although it would be possible to first predict instances independently, and then the relations between pairs of detected instances, this strategy clearly appears sub-optimal.

Imagine for example that the entities predicted are, for whatever reason, a dog and a lion. Jointly predicting the entities avoids predicting particularly improbable pairs. On the other hand, even if not all entity and relation triplets are represented in the training data, the correlations between objects should hopefully be taken into account. For example, even if the particular configuration of a horse eating from a bucket is not in the training data, all other domestic animal feeding situations are correlated with it (man feeding cow with a bucket, man feeding a horse with another object, etc.). We refer the interested reader to Liu et al. (2019) which provides a complete survey of the field of Visual Semantic Information Pursuit.

Beyond practical applications, prediction of joint or constrained variables is of great theoretical interest and encompasses tasks such as ordinal regression, multiclass classification, multilabel classification, or manifold regression, to cite but a few. In fact, we will see that most machine learning problems can be formulated as a form of Structured output prediction problem, from binary classification to ordinal regression. Furthermore, many theoretical and practical challenges of statistical learning are present in Structured output prediction:

1. Both training and inference are NP-Hard in general, making large-scale (in problem size, and dataset size) problems hard to solve. Indeed, many common approaches to optimization rely on repeatedly performing inference on (part of) the training set (for example, the Structured SVM algorithm, see Tsochantaridis et al., 2004).

2. Seen as a classification problem with a very large label space, Structured output prediction has few statistical or algorithmic guarantees with respect to standard approaches. In fact, as highlighted by Nowak-Vila et al. (2019b), many algorithms are known to be inconsistent.

In this work, we present a PAC-Bayes analysis of the Implicit Loss Embedding (ILE) approach to Structured prediction. In particular we concentrate of studying the generalization and consistency properties of the ILE approach.

Generalization. The former is a central problem in statistical learning theory and quantifies how well we can expect the performance of a predictor trained on a finite dataset to generalize to the entire data-generating distribution. There would be too many seminal contributions to the study of generalization to cite – we refer to Devroye et al. (1997) and Vapnik (1998) and references therein.

In this work, we consider the PAC-Bayes learning paradigm to study generalization. A generalization of the Probably Approximately Correct framework of Valiant (1984), PAC-Bayes bounds originate in the seminal papers of Shawe-Taylor and Williamson (1997); McAllester (1998, 1999) and further developed by Seeger (2002); Catoni (2004, 2007). In this framework, instead of considering the generalization qualities of a given predictor, we

study that of a distribution of predictors, learned from the training data. The bounds we aim to obtain make little to no hypotheses on the data-generating distribution. There has been a surge of interest in PAC-Bayes in the past few years, as it has re-emerged as a promising framework for assessing generalization performance in several settings, such as coherent risk measures (Mhammedi et al., 2020), deep neural networks (Dziugaite and Roy, 2017; Letarte et al., 2019), differential privacy (Dziugaite and Roy, 2018b,a), meta-learning (Amit and Meir, 2018) or contrastive learning (Nozawa et al., 2020) to name but a few. We refer to Guedj (2019) for a recent survey and to Guedj and Shawe-Taylor (2019) for a recent tutorial on PAC-Bayes. We outline some contributions of PAC-Bayesian theory to Structured prediction in the *Prior work* paragraph below.

Consistency. In statistical learning theory, much effort has been spent solving the intractability of minimizing the empirical $0-1$ loss. We will make the definition of consistency precise in what follows. Broadly, a surrogate method is consistent for a given task (for example, minimizing the empirical $0-1$ loss) if minimizing the surrogate loss (for example, the quadratic loss) solves the task. For general treatments about consistency, see for example Devroye et al. (1997) and Bartlett et al. (2006).

In the rest of the paper, we introduce and reason around Fisher consistency and its implications in practice.

Prior work. There have been many attempts to tame the Structured output prediction problem, both from a statistical and an algorithmic point of view. These include for example many *ad hoc* approaches such as Structural Support Vector Machines (Tsochantaridis et al., 2004) or Conditional Random Fields (Lafferty et al., 2001; Taskar et al., 2004). Nowozin and Lampert (2011) provides a survey of applications of Structured prediction for Computer vision. Wainwright and Jordan (2008) presents a unified approach to learning and inference over graphical models. Many ensemble methods based on building efficient subgraphs over graphical models have been developed: for example, Rakotomamonjy et al. (2008), Bach et al. (2004), Argyriou et al. (2008), Marchand et al. (2014).

Several quadratic methods have focused on (consistently) training quadratic surrogate methods for the Structured learning problem. These include Kernel Dependency Estimation (KDE, see Weston et al., 2003; Cortes et al., 2007), Input-Output Kernel (for example, Brouard et al., 2016) and ILE approaches (Ciliberto et al., 2016; Osokin et al., 2017; Rudi et al., 2018; Nowak-Vila et al., 2019a; Ciliberto et al., 2019). Non-quadratic methods also exist (Nowak-Vila et al., 2020). For instance, Nowak-Vila et al. (2019b) generalizes the ILE framework to a more general class of surrogates.

Several authors have proposed PAC-Bayes approaches to Structured prediction. Let us cite just three works which show the diversity of the PAC-Bayesian approach: Marchand et al. (2014) uses PAC-Bayes arguments to justify an approximation strategy for graphical models, while McAllester (2007) provides generalization bounds with a focus on tasks involving language models. Finally, Giguère et al. (2013) provides a PAC-Bayesian generalization bound for the KDE algorithm, which we will return to. For completeness we also mention recent works focusing on the decoding problem (that is, constructing a solution to the task from the surrogate solution), even though this falls outside of the scope of this paper (Mensch and Blondel, 2018; Blondel, 2019).

Contributions. In this work, we propose an original PAC-Bayes analysis of the ILE framework in Ciliberto et al. (2020). For most of the paper, we concentrate on the multi-label problem. In particular:

- We provide a PAC-Bayes generalization bound using classic PAC-Bayes arguments based on Zhang (2006).
- We extend the Comparison inequality in Ciliberto et al. (2020) to propose a new bound on the expected excess risk that depends on the performance of the predictor distribution on the training set. This result holds with high probability over the training set, thanks to recent work on PAC-Bayes generalization bounds in Haddouche et al. (2020) for unbounded loss functions, such as commonly used regression losses.
- We thoroughly analyze the bound and its behavior, allowing us to illustrate the importance of using a structured loss (for example, the Hamming loss) instead of the $0-1$ loss. This yields theoretical insights as well as practical advice for Structured prediction.
- Finally, we present learning algorithms derived from our novel PAC-Bayes bound and test their performance on Structured output prediction datasets. All our experiments are reproducible from the associated GitHub repository[1].

The rest of the paper is organized as follows: in Section 2, we introduce the Structured learning formulation and the ILE framework. In Section 3, we present our PAC-Bayes classification bound for ILE Structured output prediction. In Section 4, we derive and analyze a novel PAC-Bayes generalization bound providing insights into the consistency of ILE methodology. In Section 5, we solve the multi-label classification problem on Structured output prediction datasets using the derived result, through two methods. In Section 6 we put our results into perspective with respect to the ILE framework. Finally, we gather our experimental results in Section 7.

2. Structured output Learning with Implicit Loss Embeddings

We now introduce our notation and the Implicit Loss Embedding (ILE) framework for Structured output prediction.

2.1 Supervised Learning & Excess risk

Let $\mathcal{X}$ be an input space equipped with a kernel $k(x, x') = \langle \phi(x), \phi(x') \rangle$ for any $x, x' \in \mathcal{X}$. Let $\mathcal{Y}$ be the label space for the training data, and $\mathcal{Z}$ denotes the output space. In many cases, $\mathcal{Y} = \mathcal{Z}$, but the distinction is necessary, for example, in information retrieval. Indeed in this setting, $\mathcal{X}$ can be for example the space of search engine queries with $\mathcal{Z}$ an ordering over documents presented in response to a query and $\mathcal{Y}$ a relevance score for each result for the query (instead of a ranking). Let $\Delta : \mathcal{Z} \times \mathcal{Y} \to \mathbb{R}$ be a non-negative loss function.

We aim to solve the supervised learning problem, *i.e.*, given a dataset $\{(x_i, y_i)\}_{i=1}^m$ drawn i.i.d. from a data-generating distribution ρ over $\mathcal{X} \times \mathcal{Y}$, find a measurable function

1. https://github.com/theophilec/PAC-Bayes-ILE-Structured-Prediction

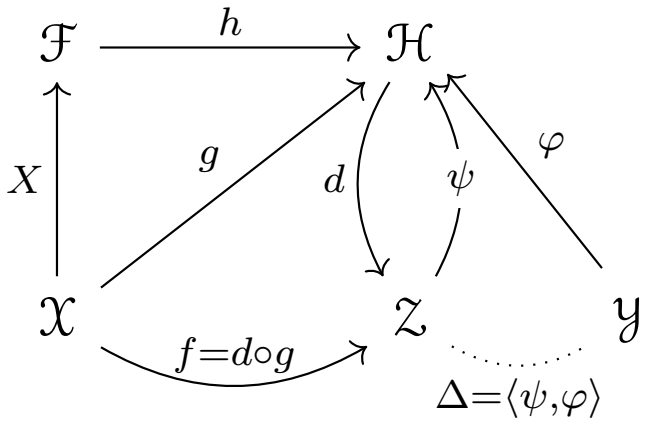

Figure 1: Summary illustration of the different sets and maps in the ILE framework.

$f_m : \mathcal{X} \to \mathcal{Z}$ that minimizes the expected task risk (or Bayes risk) defined by:

$$\mathcal{E}(f) = \int_{\mathcal{X}\times\mathcal{Y}} \Delta(f_m(x), y) \mathrm{d}\rho(x, y). \tag{1}$$

We define f^* the optimal predictor given by:

$$f^* = \arg \min_{f_m : \mathcal{X}\to\mathcal{Z}} \mathcal{E}(f_m), \tag{2}$$

where $\mathcal{X} \to \mathcal{Z}$ denotes all measurable functions from $\mathcal{X}$ to $\mathcal{Z}$.

We define the *expected excess risk* $\mathcal{E}(f_m) - \mathcal{E}(f^*)$ between f_m and f^*, which characterizes the sub-optimality of a predictor f_m with respect to the task loss Δ and data-generating distribution ρ.

In what follows, we present the ILE framework of Ciliberto et al. (2020) as well as some selected results. We refer to Ciliberto et al. (2020) and references therein for a more complete treatise.

2.2 Implicit Loss Embedding

The central idea to the ILE framework is the existence of an embedding of the output and label spaces such that the loss function Δ is the dot product of the embedding functions to this space (similarly to the idea behind the use of kernels). Formally, we define an Implicit Loss Embedding for Δ as follows. See Figure 1 for a summary illustration of the different involved sets and maps.

Definition 1 (Implicit Embedding, ILE)**.** *A loss function $\Delta : \mathcal{Z} \times \mathcal{Y} \to \mathbb{R}$ has an Implicit Loss Embedding (ILE) if there exists a Hilbert space $\mathcal{H}$ and two measurable maps $\varphi : \mathcal{Y} \to \mathcal{H}$ and $\psi : \mathcal{Z} \to \mathcal{H}$ such that*

$$\forall (z, y) \in \mathcal{Z} \times \mathcal{Y}, \ \Delta(z, y) = \langle \psi(z), \varphi(y) \rangle. \tag{3}$$

Let us motivate the introduction of this definition. First, notice that most structured (and indeed unstructured) loss functions fall into this formalism. For example, if $\mathcal{Y}$ and $\mathcal{Z}$

are finite, then the ILE is the matrix formed by the values of the loss function for (y, z) pairs, encoded as one-hot encodings e_y and e_z. In Example 1, we show that the Hamming loss for multi-label classification has an ILE. We offer a formal definition of the ℓ-multilabel binary classification problem.

Problem 1 (ℓ-multilabel binary classification problem)**.** *With $\ell \in \mathbb{N}^*$, let $\mathcal{Y} = \{0, 1\}^\ell$. Within the formalism described in Section 2.1, the problem is to learn a predictor $f : \mathcal{X} \to \mathcal{Y}$ from training data sampled iid from the data-generating distribution ρ over $\mathcal{X} \times \mathcal{Y}$.*

Some generic examples of ℓ-multilabel binary classification problems include for example detecting several characteristics of input datum x (i.e. different objects present in an image, topics of a text, presence of different defects). Using a similar example as in the introduction, detecting several objects in an image *should* be easier if the objects are jointly predicted.

Example 1. *In binary ℓ-multi-label learning, the Hamming loss H is ILE where*

$$H(z, y) = \frac{1}{\ell} \sum_{k=1}^{\ell} \mathbb{I}(z_k \neq y_k) = 1 - \frac{1}{\ell} \sum_{k=1}^{\ell} \mathbb{I}(z_k = 0)\mathbb{I}(y_k = 0) + \mathbb{I}(z_k = 1)\mathbb{I}(y_k = 1). \quad (4)$$

Furthermore, $|\mathcal{Y}| = 2^\ell$ and $\dim(\mathcal{H}) = 2\ell + 1$.

Different authors have shown that a wide variety of machine learning problems fall into the ILE framework. As a matter of fact, Ciliberto et al. (2020, Sec. 6) provide a collection of simple sufficient conditions for a learning problem to admit an ILE (the finite cardinality of $\mathcal{Y}$ and $\mathcal{Z}$ is the simplest of such conditions).

Second, the "separability" of Δ as illustrated in Eq. (6) ensures that the learning problem can be (Fisher-)consistently solved by solving a more computationally efficient quadratic regression problem, approximating $\varphi(y)$ from x. In particular, this ensures that, in structured contexts, the so-called pre-image problem is not solved during training as it is in Structured SVM methods.

In what follows, we assume that Δ is ILE with associated maps ψ and φ. We also assume that $\mathcal{Y}$ and $\mathcal{Z}$ are finite sets, and that $\mathcal{H}$ is finite dimensional. In specific examples, we will return to the ℓ-multi-label binary classification problem with the Hamming loss, as defined in Problem 1 and Eq. (4).

2.3 Quadratic surrogate and consistency

Under the assumptions that the task loss is ILE and that the data-generating distribution ρ can be factorized as $\rho(x, y) = \rho(x)\rho(y|x)$, Ciliberto et al. (2020) prove the following results, which naturally lead to a consistent surrogate method (in a sense we define below). We briefly survey this approach to set the context and demonstrate the utility of such a framework, in theory and in practice.

Ciliberto et al. (2020) first prove that the pointwise conditionally optimal predictor is identically equal to the minimizer of Eq. (1):

Lemma 1 (Ciliberto et al., 2020)**.** *Assume that the data-generating distribution ρ can be factorized as $\rho(x, y) = \rho(y|x)\rho_{\mathcal{X}}(x)$, then*

$$f^*(x) = \arg\min_{z \in \mathcal{Z}} \int_{\mathcal{Y}} \Delta(z, y) d\rho(y|x). \quad (5)$$

From this, because $\Delta(z, y) = \langle \psi(z), \varphi(y) \rangle$ is linear in $\varphi(y)$, the optimal predictor f^* is naturally written as a function of the conditional mean embedding of $\varphi(y)$ given x, which we denote $g^*(x) = \int_{\mathcal{Y}} \varphi(y) d\rho(y|x)$, *i.e.*,

$$f^*(x) = \arg\min_{z \in \mathcal{Z}} \langle \psi(z), g^*(x) \rangle. \tag{6}$$

Thus, in order to estimate f^*, it is natural to seek to estimate g^* by g and "plug-in" g to define f as

$$\forall x \in \mathcal{X}, f(x) = \arg\min_{z \in \mathcal{Z}} \langle \psi(z), g(x) \rangle. \tag{7}$$

It can be shown, see for example Grünewälder et al. (2012), that g^* is the minimizer of the following expected quadratic risk:

$$\mathcal{R}(g) = \int_{\mathcal{X} \times \mathcal{Y}} \|\varphi(y) - g(x)\|^2 \, d\rho(x, y). \tag{8}$$

A consistent empirical counterpart of Eq. (8) for a given sample is:

$$\mathcal{R}_n(g) = \frac{1}{n} \sum_{i=1}^{n} \|g(x_i) - \varphi(y_i)\|^2 + \lambda \|g\|^2, \tag{9}$$

where λ is a regularization parameter. There exist $\alpha_1, \ldots, \alpha_n$ n functions in $\mathcal{X} \to \mathcal{H}$ such that g_n minimizes $\mathcal{R}_n$ where g_n is defined by

$$g_n(x) = \sum_{i=1}^{n} \alpha_i(x) \varphi(y_i). \tag{10}$$

The associated classifier is thus defined by:

$$\forall x \in \mathcal{X}, \; f_n(x) = \arg\min_{z \in \mathcal{Z}} \langle \psi(z), g_n(x) \rangle \tag{11}$$

$$= \arg\min_{z \in \mathcal{Z}} \sum_{i=1}^{n} \alpha_i(x) \langle \psi(z), \varphi(y_i) \rangle. \tag{12}$$

The above computation is called the "loss trick" by Ciliberto et al. (2020) in analogy to the well-known "kernel trick". This proves that the existence of the ILE embedding is enough and that knowledge of φ and ψ are not required.

Efficiently solving the decoding problem. As we have outlined above, the general structure of the ILE learning method is to solve a quadratic surrogate regression problem, and then at inference solve a decoding problem (the arg min in Eq. (6)). A priori, this can be solved in $O(|\mathcal{Z}|)$ time. Because $|\mathcal{Z}|$ is combinatorial in nature (for example $|\mathcal{Z}| = 2^\ell$ for the ℓ-label binary classification problem) this can be costly and even impossible at large scale.

In this work, we focus on the link between the plug-in classifier f_n and the surrogate regressor g_n without worrying about how the arg min is computed. The problem of efficiently solving the decoding problem is the subject of some recent work in, for example, Blondel (2019); Mensch and Blondel (2018).

Fisher consistency Ciliberto et al. (2020) show that estimating g is *Fisher consistent*, *i.e.*, that if the optimal regressor g^* is found, then the associated predictor f^* is optimal for $\mathcal{E}$. Formally, there exists a map $d : \mathcal{H} \to \mathcal{Z}$ (for example, the arg min function over $\mathcal{Z}$) such that $\mathcal{E}(f^*) = \mathcal{E}(d \circ g^*)$.

They prove a stronger result, upper-bounding the expected excess prediction risk $\mathcal{E}(f) - \mathcal{E}(f^*)$ as a function of the expected excess regression risk $\mathcal{R}(g) - \mathcal{R}(g^*)$, their *comparison inequality*. This result will be the basis for our own analysis in Section 4. For easy cross-referencing, we present a stronger version of the inequality obtained by withholding the last step in the proof of the inequality in Ciliberto et al. (2020).

Theorem 2 (Strong Comparison Inequality)**.** *Assume $\mathcal{Z}$ is compact and Δ has an ILE. Let $g : \mathcal{X} \to \mathcal{H}$ be **measurable**, and f defined as in Eq. (7). Then,*

$$(f) - (f^*) \leq 2c_\Delta \int_{\mathcal{X}} \|g(x) - g^*(x)\| \, d\rho_{\mathcal{X}}(x),$$

where $c_\Delta = \sup_{z \in \mathcal{Z}} \|\psi(z)\|$ and $\rho_{\mathcal{X}}$ is the data-generating distribution marginalized over y.

In particular, this implies the Comparison inequality (Theorem 3).

Theorem 3 (Comparison inequality, Thm. 3, Ciliberto et al., 2020)**.** *Assume $\mathcal{Z}$ is compact and Δ has an ILE. Let $g : \mathcal{X} \to \mathcal{H}$ be **measurable**, and f defined as in Eq. (7). Then,*

$$\mathcal{E}(f) - \mathcal{E}(f^*) \leq 2c_\Delta \sqrt{\mathcal{R}(g) - \mathcal{R}(g^*)},$$

where $c_\Delta = \sup_{z \in \mathcal{Z}} \|\psi(z)\|$.

Note that such calibration functions exist for a wider range of surrogates, see Nowak-Vila et al. (2019b) for a more general approach. These results are analogous to the ψ-transform bounds in, for example, Bartlett et al. (2006).

In this section, we presented the supervised Structured learning problem and the ILE framework. We also introduced the relevant results for the rest of the paper. We now present the PAC-Bayes approach, as well as prove our first results.

3. A PAC-Bayes take on ILE classification

In this section, we offer a novel perspective on the learning problem described in Section 2 using tools from the PAC-Bayes framework. We first outline the main tenets of the PAC-Bayes approach, and a classic PAC-Bayes bound. Using these tools we prove a generalization bound for ILE classification.

Notation. In the rest of the paper, we define the empirical task risk as:

$$\mathcal{E}_m(f) = \frac{1}{m} \sum\nolimits_{i=1}^{m} \Delta(f(x_i), y_i). \tag{13}$$

The expectation of the measurable function f of a random variable X with domain $\mathcal{X}$ under distribution $\mathcal{Q}$ is defined as (when it exists):

$$\mathop{\mathbb{E}}_{X \sim \mathcal{Q}} f(x) = \int_{\mathcal{X}} f(x) d\mathcal{Q}(x).$$

The Kullback-Leibler divergence between distributions $\mathcal{Q}$ and $\mathcal{P}$ is denoted $\mathcal{K}(\mathcal{Q}, \mathcal{P})$ and defined as

$$\mathcal{K}(\mathcal{Q}, \mathcal{P}) = \int \log \frac{d\mathcal{Q}(x)}{d\mathcal{P}(x)} d\mathcal{Q}(x).$$

When there is no ambiguity, we identify linear maps between finite-dimensional vector spaces, associated matrices in the canonical bases, and the vectors of matrix coordinates.

Finally, in the rest of the paper, we will make the slight notation abuse of writing $f \sim \mathcal{Q}$ when f is the structured predictor associated to stochastic regressor $g \sim \mathcal{Q}$.

3.1 Canonical PAC-Bayes bounds

We introduce the PAC-Bayes setting and present a canonical PAC-Bayes generalization bound from Zhang (2006). In PAC-Bayes theory, one considers stochastic predictors sampled from a distribution $\mathcal{Q}$ over the space of measurable maps $\{f : \mathcal{X} \to \mathcal{Z}\}$. Intuitively, this can be understood as sampling a different predictor from a distribution $\mathcal{Q}$ for each inference task, for example for each data-point in a dataset.

Thus, if learning is generally the process of estimating an optimal predictor, in the PAC-Bayes approach, learning consists in estimating an optimal distribution of predictors. Similarly to the general learning setting, the definition of "optimality" depends on the performance of the predictor (distribution) on the training set and on regularization terms.

Because the optimized distribution $\mathcal{Q}$ depends on the training set used for "learning": in analogy to Bayesian statistics, this distribution is called the posterior. Conversely, we consider a data-independent prior distribution over predictors $\mathcal{P}$.

The result below bounds the expected risk over a posterior distribution $\mathcal{Q}$ of predictors from the empirical risk over the dataset, with high probability over the training set. Note that the bound holds for any data-generating distribution ρ. The generalization penalty terms contain the Kullback-Leibler divergence between $\mathcal{Q}$ and $\mathcal{P}$. Because the Exponential information inequality presented in Zhang (2006) uses an alternative so-called one-sample formulation, we refer to the corresponding result in Giguère et al. (2013):

Theorem 4 (Giguère et al., 2013, Thm. 10.3.)**.** *Let ζ be an arbitrary (in particular, not necessarily bounded) loss function.*

For any data-independent distribution $\mathcal{P}$, $\delta > 0$, and $a > 0$, with probability at least $1-\delta$ over the training set drawn from ρ^m, for any posterior distribution $\mathcal{Q}$ such that $\mathcal{Q} \ll \mathcal{P}$ and $\mathcal{P} \ll \mathcal{Q}$,

$$-\mathop{\mathbb{E}}_{g\sim Q} \log \mathop{\mathbb{E}}_{x,y\sim\rho} e^{-\zeta(g(x),y)} \leq \frac{1}{m}\left(\mathop{\mathbb{E}}_{g\sim Q} \sum_{i=1}^{m} \zeta(g(x_i), y_i) + \mathcal{K}(Q, \mathcal{P}) + \log\frac{1}{\delta}\right).$$

We use Theorem 4 to prove a first result below.

3.2 ILE generalization bound

In this section, we prove the following novel result, which applies Theorem 4 to ILE Structured prediction:

Theorem 5 (Classification bound). *Assume that Δ admits an ILE defined by $\Delta(z,y) = \langle \psi(z), \varphi(y) \rangle$. Let $g : \mathcal{X} \to \mathcal{H}$ a measurable function, and $f(x) = \arg\min_{z \in Z} \langle \psi(z), g(x) \rangle$.*

For any data-independent distribution $\mathcal{P}$, for any $\delta > 0$, for any $\lambda > 0$, with probability at least $1-\delta$ over the training set sampled iid from ρ^m, for any posterior distribution $\mathcal{Q}$ such that $\mathcal{Q} \ll \mathcal{P}$ and $\mathcal{P} \ll \mathcal{Q}$,

$$\mathop{\mathbb{E}}_{f\sim\mathcal{Q}} \mathcal{E}(f) \leq \frac{ae}{e-1}\left(1 - \exp\left(-\frac{1}{a}\mathop{\mathbb{E}}_{f\sim Q}\mathcal{E}_m(f) - \frac{\mathcal{K}(Q,\mathcal{P}) + \log\frac{1}{\delta}}{m}\right)\right). \tag{14}$$

The proof of Theorem 5 is inspired by the proof of Theorem 10.4 in (Giguère et al., 2013). We first prove an exponential bound on the identity, with a slack parameter a. Figure 2 illustrates the bound.

Proposition 6 (Exponential bound on the identity, with slack parameter). *For any $a > 1$,*

$$\forall x \in [0,1],\ x \leq \frac{ae}{e-1}\left(1 - e^{-\frac{x}{a}}\right). \tag{15}$$

Proof The function f defined by $f(x) = \frac{e}{e-1}(1-e^{-x})$ on $[0,1]$ is concave. Since, $f(0) = 0$ and $f(1) = 1$, $\forall x \in [0,1]$, $x \leq f(x)$ (f is over its chords). The result follows because $\frac{1}{a}[0,1] \subset [0,1]$ ($a > 1$). ■

Proof (Theorem 5) Let $\mathcal{P}$ be a prior distribution on regression functions. Let $\delta > 0$ and Q be a distribution over regression functions.

Let $a > 1$. Denote: $\zeta(f(x),y) = \frac{\Delta(f(x),y)}{a}$. Then, by Theorem 4, with probability at least $1-\delta$ on the training set sampled iid from ρ^m, we have

$$-\mathop{\mathbb{E}}_{g\sim Q}\log\mathop{\mathbb{E}}_{x,y\sim D} e^{-\zeta(g(x),y)} \leq \frac{1}{m}\left(\mathop{\mathbb{E}}_{g\sim Q}\sum_{i=1}^{m}\zeta(g(x_i),y_i) + \mathcal{K}(Q,\mathcal{P}) + \log\frac{1}{\delta}\right).$$

Because $x \mapsto -\log x$ is convex, by Jensen,

$$-\log\mathop{\mathbb{E}}_{g\sim Q}\mathop{\mathbb{E}}_{x,y\sim D} e^{-\zeta(g(x),y)} \leq \frac{1}{m}\left(\mathop{\mathbb{E}}_{g\sim Q}\sum_{i=1}^{m}\zeta(g(x_i),y_i) + \mathcal{K}(Q,\mathcal{P}) + \log\frac{1}{\delta}\right).$$

Because $x \mapsto -\exp(-x)$ is non-decreasing,

$$-\mathop{\mathbb{E}}_{g\sim Q}\mathop{\mathbb{E}}_{x,y\sim D} e^{-\zeta(g(x),y)} \leq -\exp\left(-\frac{1}{m}\left(\mathop{\mathbb{E}}_{g\sim Q}\sum_{i=1}^{m}\zeta(g(x_i),y_i) + \mathcal{K}(Q,\mathcal{P}) + \log\frac{1}{\delta}\right)\right). \tag{16}$$

On the other hand, from Proposition 6 applied to $\Delta(f(x),y)$,

$$\Delta(f(x),y) \leq \frac{ae}{e-1}\left(1 - e^{-\Delta(f(x),y)/a}\right). \tag{17}$$

We conclude by combining Eq. (16) and Eq. (17). ■

It is important to note that the value of the bound depends on the performance of a given distribution of predictors on the training set, independently of how the distribution is obtained. We return to this discussion in Section 6.

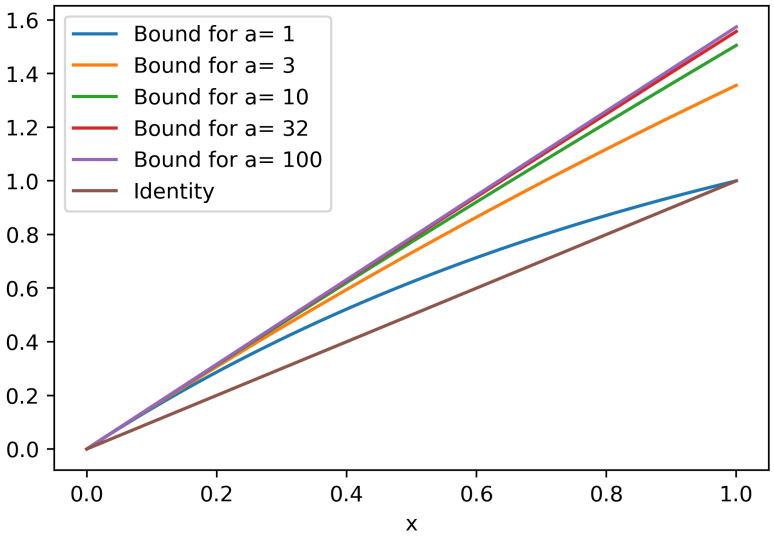

Figure 2: Illustration of Proposition 6 (the exponential bound on the identity) for different values of $a > 1$.

4. PAC-Bayes analysis of consistency

In the previous section, we proved a PAC-Bayes bound on the expected risk of the stochastic predictor $\mathbb{E}\Delta(f(x), y)$ (the expectation is taken with respect to $f \sim \mathcal{Q}$ and $x, y \sim \rho$). The bound depends on the empirical risk of the stochastic predictor f $\mathbb{E}\frac{1}{m}\sum_i \Delta(f(x_i), y_i)$ (where the expectation is taken with respect to $f \sim \mathcal{Q}$ only) and a penalty term. This can give a tight estimate of the generalization properties of a supervised classification problem.

However, minimizing the upper bound is at least as hard as solving the ILE Structured prediction problem directly. Thus, it does not provide an efficient way of learning: we cannot, given a model of posterior, estimate the parameters for which we minimize the bound. Furthermore, the results introduced in Section 3 pertain to generalization and do not quantify the sub-optimality of the predictor of f (compared to f^*) as a function of the sub-optimality of g.

We propose an approach to solving the latter problem in this section. In Section 5 we use these results to learn a posterior $\mathcal{Q}$ over predictors from a training set $(x_1, y_1), \ldots, (x_m, y_m)$.

In this section, we aim to bound the expected excess risk of a (stochastic) predictor f using the empirical excess quadratic surrogate risk. Our contributions are original in several respects: first, they incorporate the finite-sample performance of the predictor distribution in true PAC-Bayes fashion, giving actionable insight into the value of a predictor distribution; second, in the case of finite-dimensional vector spaces, they formalize certain trade-offs that are important in the practice of Structured prediction.

4.1 Augmented learning problem

To this end, we introduce an *augmented learning problem*. Simply put, we consider the learning problem of learning from the dataset $(x_i, g^*(x_i))$ instead of learning from (x_i, y_i). Note that g^* depends on the data-generating distribution ρ but not on the iid samples $x_1, \ldots, x_m$. Thus, the pairs $(x_i, g^*(x_i))$ are iid, just like (x_i, y_i).

This allows us to absorb the intractable $\mathcal{R}(g^*)$ in the Comparison inequality, and decompose $\mathcal{R}(g) - \mathcal{R}(g^*)$, using the Strong Comparison Inequality. We introduce the absolute deviation loss $\|g(x) - g^*(x)\|$, naturally interpretable as the pointwise surrogate excess risk. The empirical sum of these quantities can be seen as the $L_{2,1}$ norm of the residuals of the augmented regression.

Proving a PAC-Bayes bound for this augmented learning problem requires establishing a concentration inequality over the absolute deviation regression problem (an unbounded loss). In the following section, we present and adapt a result from Haddouche et al. (2020) and establish our result.

4.2 Absolute regression bound

In this section, we prove Theorem 9, which provides a finite-sample dependent consistency guarantee with high-probability. We first state Theorem 9, then introduce and extend results from Haddouche et al. (2020) needed to prove it.

In order to handle the unbounded absolute deviation loss, recall the following definition from Haddouche et al. (2020):

Definition 7 (Hypothesis-dependent range (`HYPE`) condition)**.** *Let $\mathcal{D} = \mathcal{X} \times \mathcal{H}$ be the data space and $\mathcal{G}$ be the space of predictors. A loss function $\ell : \mathcal{H} \times \mathcal{G}$ verifies the Hypothesis-dependent range (`HYPE`) condition for a function $K : \mathcal{G} \to \mathbb{R}_+^*$ such that $\forall g \in \mathcal{G}, \sup_{x,y \in \mathcal{D}} \ell(g, z) \leq K(g)$. In this case, we say that ℓ is `HYPE`(K)-compliant.*

The following theorem is extended to vector-valued regression from Haddouche et al. (2020, Thm. 5.1.).

Theorem 8 (extended from Haddouche et al., 2020, Thm. 6.1)**.** *Let $1 \geq \alpha > 0$ and $N \geq 6$, the dimension of the regression space $\mathcal{G}$. We define ℓ as:*

$$\ell : (h, (X(x), g^*(x))) \in \mathcal{G} \times (\mathcal{F} \times \mathcal{H}) \mapsto \|g^*(x) - h \circ X(x)\| . \tag{18}$$

Then, ℓ is `HYPE`(K) compliant for $K(h) = B\|h\|_F + C$ with $C = \|g^*\|_{2,\infty}$ and $B = \|X\|_{2,\infty}$.

Moreover, let $\mathcal{P} = \mathcal{N}(0, \sigma^2 I_N)$ with $\sigma^2 = t\frac{m^{1-2\alpha}}{B^2}$ with $0 < t < 1$, be a Gaussian prior. We have with probability at least $1 - \delta$ over $S \sim D^m$, for any posterior distribution $\mathcal{Q}$ such that $Q \ll P$ and $P \ll Q$:

$$\mathop{\mathbb{E}}_{h\sim Q} R(h) \leq \mathop{\mathbb{E}}_{h\sim Q} R_m(h) + \frac{\mathcal{K}(Q,P) + \log(2/\delta)}{m^\alpha} + \frac{C^2}{2m^{1-\alpha}}\left(1 + F(t)^{-1}\right) \tag{19}$$

$$+ \frac{N}{m^\alpha}\left[\log\left(1 + \frac{C}{\sqrt{2F(t)m^{1-2\alpha}}}\right) + \log\left(\frac{1}{\sqrt{1-t}}\right)\right]. \tag{20}$$

where $F(t) = \frac{1-t}{t}$.

Proof We first prove that ℓ is $\texttt{HYPE}(K)$-compliant for $K(h)$ of the form $K(h) = \|g^*\|_{2,\infty} + \|h\|_{op} \|X\|_{2,\infty}$. Indeed,

$$\|g^*(x) - h \circ X(x)\| \leq \|g^*\|_{2,\infty} + \|h\|_{op} \|X(x)\|_2 .$$

Furthermore, we have $\forall h, \|h\|_{op} \leq \|h\|_F$. Indeed, let $h = U\Sigma V^T$ be the singular value decomposition of h where U and V are unitary and Σ is diagonal with non-negative σ_i singular values of A. Because U and V are unitary, $\|\Sigma\|_F = \left\|U^T h V\right\|_F = \|h\|_F$. Thus,

$$\|h\|_F = \|\Sigma\|_F = \sqrt{\sum_i \sigma_i(A)^2} \geq \max_i \sigma_i(A) = \|h\|_{op} .$$

Thus,

$$\|g^*(x) - h \circ X(x)\| \leq \|g^*\|_{2,\infty} + \|h\|_{op} \|X\|_{2,\infty} .$$

The second part of the result is a generalization of the original result. Because the $\texttt{HYPE}$ condition is verified with the same form as in the scalar case. See Haddouche et al. (2020, Appendix F.2.) for a proof of the original result. ■

Finally, we specialize Theorem 8 into a PAC-Bayes generalization bound for the augmented regression problem:

Theorem 9 (Augmented linear regression excess risk bound)**.** *Let $\alpha > 0$ and $\dim(\mathcal{H} \otimes \mathcal{F}) = N \geq 6$. Let $\mathcal{P} = \mathcal{N}(0, \sigma_0^2 I_N)$ with $\sigma_0^2 = t\frac{m^{1-2\alpha}}{\kappa^2} = t\sigma^2$ be a Gaussian prior with $0 < t < 1$. We have with probability at least $1-\delta$ over the training set sampled from ρ^m, for any posterior distribution $\mathcal{Q} = \mathcal{N}(W, \sigma'^2 I_N)$,*

$$\mathop{\mathbb{E}}_{f\sim\mathcal{Q}} \mathcal{E}(f) - \mathcal{E}(f^*) \leq 2c_\Delta \left[\mathop{\mathbb{E}}_{g\sim\mathcal{Q}} \frac{1}{m} \sum_{i=1}^m \|g^*(x_i) - g(x_i)\| + \frac{\mathcal{K}(\mathcal{Q}, \mathcal{P}) + \log\frac{2}{\delta}}{m^\alpha} + \varepsilon(m, t, \alpha, \mathcal{P}) \right], \tag{21}$$

where

$$\varepsilon(m, t, \alpha, \mathcal{P}) = \frac{\|g^*\|^2}{2m^{1-\alpha}} \left(1 + F(t)^{-1}\right) + \frac{N}{m^\alpha} \left[\log\left(1 + \frac{\|g^*\|}{\sqrt{2F(t)m^{1-2\alpha}}}\right) + \log\left(\frac{1}{\sqrt{1-t}}\right) \right], \tag{22}$$

and $F(t) = \frac{1-t}{t}$.

Proof Recall the Strong Comparison Inequality (Theorem 2):

$$\mathcal{E}(f) - \mathcal{E}(f^*) \leq 2c_\Delta \underbrace{\int_{\mathcal{X}} \|g(x) - g^*(x)\|_{\mathcal{H}} \, d\rho_{\mathcal{X}}(x)}_{=:\, R(g)} .$$

We can then apply the Augmented regression formulation and Theorem 8 to bound the expectation of the lower-bound, by recognizing $R(g)$. We also specialize the result for Gaussian posteriors. ■

Theorem 9 expands on the general PAC-Bayes approach to learning presented in Section 3 and on the Comparison Inequality Theorem 3 by considering the consistency problem from the point of view of finite sample concentration: how suboptimal is f with respect to f^* over the data-generating distribution given the suboptimality of g with respect to g^* on the finite, real-world dataset at hand.

In the rest of this section we focus on the analysis of the dependencies of the bound in Theorem 9 in the different learning problem and bound parameters. By examining and discussing the influence of these parameters on the value of the bound, we show that our approach yields practical guidance on the performance of learning methods and formulations, in particular on the choice of loss for Structured prediction.

4.3 Posterior parametrization

We first examine the impact of the choice of posterior variance on the Kullback-Leibler divergence term. Different parametrizations yield different regularization behaviors, we present two alternatives here.

For completeness, we begin by recalling the Kullback-Leibler divergence of two multivariate Gaussian distributions, with a specialization for isotropic Gaussians.

Proposition 10 (Kullback-Leibler divergence of multivariate Gaussians)**.** *The Kullback-Leibler divergence of two N-dimensional multivariate Gaussians $\mathcal{P} = \mathcal{N}(\mu_2, \Sigma_2)$ and $\mathcal{Q} = \mathcal{N}(\mu_1, \Sigma_1)$ is:*

$$\mathcal{K}(\mathcal{Q}, \mathcal{P}) = \frac{1}{2}\left[\log \frac{\det \Sigma_2}{\det \Sigma_1} - N + tr(\Sigma_2^{-1}\Sigma_1) + (\mu_2 - \mu_1)^T \Sigma_2^{-1}(\mu_2 - \mu_1)\right]. \tag{23}$$

In particular, if $\mu_2 = 0$, $\Sigma_2 = \sigma_2^2 I_N$ and $\Sigma_1 = \sigma_1^2 I_N$, then Eq. (23) can be reduced to:

$$\mathcal{K}(\mathcal{Q}, \mathcal{P}) = \frac{1}{2}\left[2N \log \frac{\sigma_2}{\sigma_1} - N + N\frac{\sigma_1^2}{\sigma_2^2} + \frac{\|\mu_1\|^2}{\sigma_2^2}\right]. \tag{24}$$

The proof can be found in many references, including in Petersen and Pedersen (2008).

General Kullback-Leibler variations. For ease of comprehension, we consider the general case where the variance of $\mathcal{P}$ is fixed, and study the variations of the Kullback-Leibler divergence as the variance of $\mathcal{Q}$ changes. In Figure 3, we illustrate the dependence of the Kullback-Leilber divergence on posterior variance (especially relatively to the prior variance).

The main takeaway of such an illustration is that there is a global minimum of the Kullback-Leibler divergence (even though the means do not coincide), which justifies that the ensuing discussion in the section is justified.

Of course, the posterior variance can be optimized in all generality. However, simple rules are interesting because they can be applied in practice. Furthermore, in PAC-Bayes literature, it is common for the posterior variance to be fixed *a priori*, for example to 1. Two *a priori* examples are interesting because they yield insight into the impact of the problem parameters on generalization performance: the unit-variance model and the so-called, wide posterior. We first present both models then compare them.

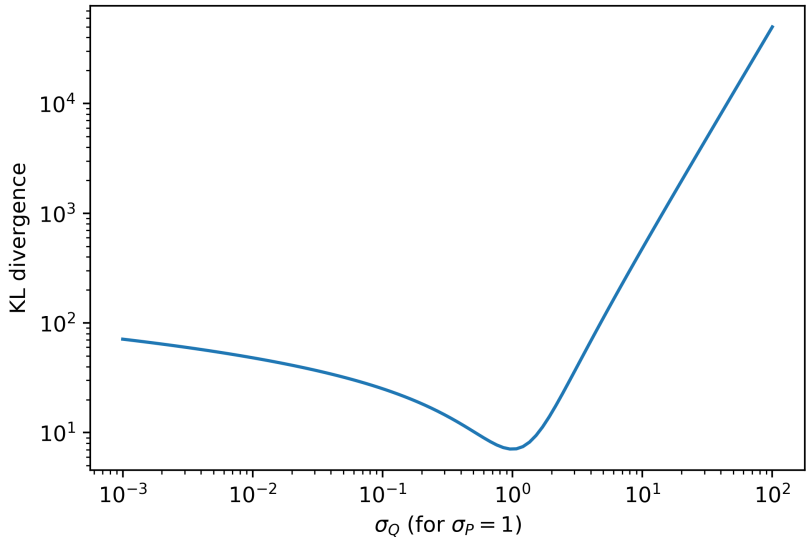

Figure 3: Kullback-Leibler divergence as a function of posterior covariance σ, where the prior covariance is unitary. Note the global minimum.

Unit variance posterior This posterior model is widely used in PAC-Bayes analysis. Here, $\sigma_1^2 = 1$ and $\sigma_2^2 = t\sigma^2$ where $0 < t < 1$, as defined in Theorem 9. Denote $\mathcal{K}_U(t)$ the Kullback-Leibler divergence in this model, which has the following expression:

$$\mathcal{K}_U(t) = \mathcal{K}(\mathcal{Q}, \mathcal{P}) = \frac{\|\mu_1\|^2}{2t\sigma^2} + \frac{n}{2}\left(\log t + \log \sigma^2 - 1 + \frac{1}{t\sigma^2}\right). \tag{25}$$

Wide posterior In this case, we set the posterior variance to the supremum of the set of admissible variances for the prior (i.e. corresponding to $t = 1$ in Theorem 9). Here, $\sigma_1^2 = \sigma^2$ and $\sigma_2^2 = t\sigma^2$, where $0 < t < 1$, and the Kullback-L divergence, noted $\mathcal{K}_W(t)$ becomes:

$$\mathcal{K}_W(t) = \mathcal{K}(\mathcal{Q}, \mathcal{P}) = \frac{\|\mu_1\|^2}{2t\sigma^2} + \frac{n}{2}\left(\log t - 1 + \frac{1}{t}\right). \tag{26}$$

Comparison We compare the two parametrizations, to better understand the implications of each hypothesis. The difference of the two divergences gives:

$$\mathcal{K}_U(t) - \mathcal{K}_W(t) = \frac{n}{2}\left(\log \sigma^2 + \frac{1}{t\sigma^2} - \frac{1}{t}\right). \tag{27}$$

This gap depends on t but only through the prior variance. It is positive if and only if

$$\log \sigma^2 + \frac{1}{t\sigma^2} - \frac{1}{t} > 0. \tag{28}$$

If $\sigma^2 < 1$, this is always verified. If $\sigma^2 > 1$, then there is a threshold behavior at $t_0(\sigma) = \frac{1-\frac{1}{\sigma^2}}{\log \sigma^2}$.

In Theorem 9, σ is not arbitrary, but determined by the parameters of the learning problem. Recall that $\sigma = \frac{m^{1-2\alpha}}{\kappa}$ where $\kappa^2 = \sup_{x \in \mathcal{X}} k(x, x)$. In other words, the magnitude of σ is inversely proportional to that of κ, with coefficient $m^{1-2\alpha}$.

In the special case of $\alpha = 1/2$, the above discussion yields insight into the impact of the magnitude of the input kernel on the generalization performance of the ILE method.

- If the input kernel is bounded by 1, then $\sigma > 1$ and the choice of parametrization (and the choice of t has an impact). In this case, t should be chosen larger than the threshold value when the wide parametrization is chosen, and smaller when the unitary parametrization is selected. This remark nuances the remark in Haddouche et al. (2020) that the limitation on the choice of prior is not limiting: choosing a smaller prior variance helps obtain good generalization bounds with a posterior with smaller variance.

- If the input kernel cannot be bounded by 1, then $\sigma < 1$ and the gap is always positive. In this case, the wide parametrization is better, whichever the choice of t. Note however that if the known bound for a given kernel is not tight, then the practitioner may find herself in the previous case without knowing it. This highlights the importance of having informative bounds when controlling the different ingredients to the learning problem.

The analysis in more subtle with $\alpha \neq \frac{1}{2}$, because the prior variance depends on m.

- In the case where $\alpha < \frac{1}{2}$, $\sigma \to +\infty$ when $m \to \infty$. So with enough data, we are in the first case above. Note that as σ becomes large, the threshold value above becomes small, meaning that the potential widening of the posterior is compensated by the choice of a small t (in the unitary posterior parametrization).

- If $1 > \alpha > \frac{1}{2}$, the prior becomes tighter as the quantity of data grows. This seems pathological in the large data limit: we want to maintain variance in the prior as the quantity of data grows.

The above discussion gives insight into the choice of posterior parametrization, giving simple rules of thumb for practical use (which eliminates a free parameter). Furthermore, we have shown that having informative bounds on the different ingredients of the learning problem has impact of the generalization guarantees we can obtain.

In the rest of this section, we qualitatively study the effect of different bound parameters on the bound value. We consider the unit-variance posterior parametrization. Furthermore, to evaluate the effects of the parameters on the bound values, we consider the mean regressor $\bar{g} = g^*$. The discussion can be formulated in the following way:

> If g^* was known, what PAC-Bayes guarantees could we expect on the resulting stochastic predictor?

In the rest of this section, we adopt the following shorthand, where we have regrouped terms in $\varepsilon' = \frac{\mathcal{K}(\mathcal{Q},\mathcal{P})}{m^\alpha} + \varepsilon$ to highlight their effect on the bound:

$$\varepsilon' = \frac{N}{m^\alpha}\left[\log\left(1 + \frac{\|g^*\|}{\sqrt{2F(t)m^{1-2\alpha}}}\right) + \log\left(\sqrt{F(t)^{-1}}\right) + \frac{1}{2}\log\left(\frac{m^{1-2\alpha}}{\kappa^2}\right) - \frac{1}{2}\right] + \frac{N}{2m^{1-\alpha}}\left[\frac{\kappa^2}{t}\left(1 + \frac{\|g^*\|_2^2}{N}\right) + \frac{\|g^*\|_2^2}{N}\left(1 + F(t)^{-1}\right)\right]. \quad (29)$$

4.4 Penalty dependence on N

The bound's dependence on N, the dimension of the regressor space, is a key contribution of our work. It justifies the practitioner's intuition in the choice of losses for Structured output prediction is important.

We can see in Eq. (29) that N contributes greatly to the generalization penalty, in $O(N)$. However, as written Eq. (29) does not materialize the dependence of the penalty on the "size" of the Structured output learning problem. We discuss below how the choice of loss determines the penalty value through N. Recall that $N = \dim(\mathcal{H}) \times \dim(\mathcal{F})$ is the dimension of the regressor. Indeed, the choice of loss impacts the ILE which in turn impacts the dimension of $\mathcal{H}$.

To illustrate this effect concretely, let us return to the multi-label binary classification problem in Example 1. For the $0-1$ loss, $\varphi(y)$ is a one-hot encoding of y, and $\dim(\mathcal{H}) = |\mathcal{Y}| = 2^\ell$. For the Hamming loss, $\dim(\mathcal{H}) = 2\ell + 1 \ll |\mathcal{Y}|$. This greatly reduces the number of samples needed to attain good generalization performance and, by Theorem 9, consistency.

In other words, the choice of a "structured loss" harnesses the power of the framework described above, instead of considering the Structured prediction problem as naïve classification.

4.5 Penalty dependence over m

Eq. (29) shows that apart from a light logarithmic dependence in m, the penalty decreases as $O\left(\frac{1}{m^\beta}\right)$, where $\beta = \min(\alpha, 1-\alpha)$. This gives an indication of a rate of convergence, which we discuss more generally in Section 6, and echoes that in Section 4.3.

4.6 Penalty dependence over t

In Figure 4, we visualize the dependence of Eq. (29) on t for different values of κ. Note the global minimum over t.

5. Learning from the Augmented Regression Bound

In this section, we present and analyze learning algorithms derived from Theorem 9. In PAC-Bayes theory, learning algorithms are derived from a bound when the bound is optimized with respect to the posterior for a given set of data. Because we use a fixed-variance posterior model, we can reduce optimizing the bound in Theorem 9 to minimizing the

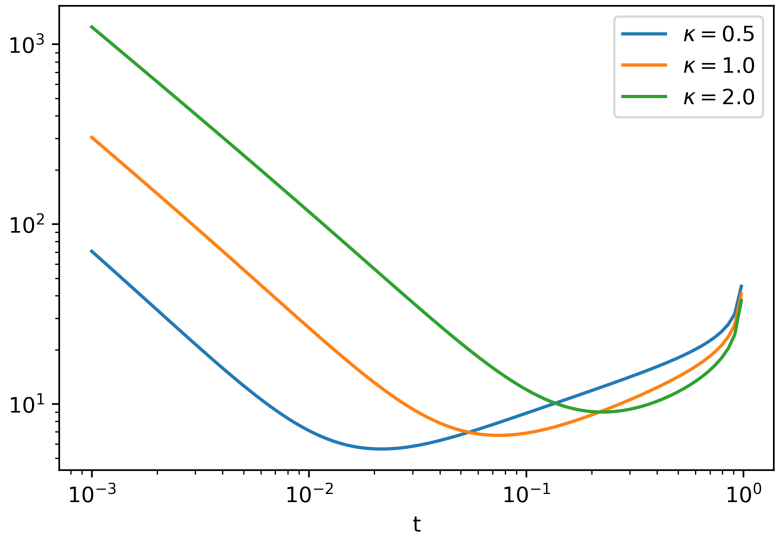

Figure 4: Dependence of ε' as a function of t, for different values of κ. For these curves: $N = 10^2$, $m = 10^4$, $\alpha = 0.3$ and $\|g^*\| = 10$. Note the global minimum over $0 < t < 1$.

following objective function, where $\mathcal{Q}(\bar{g})$ is the parameterized distribution with mean $\bar{g}$:

$$J(\bar{g}) = \mathop{\mathbb{E}}_{g\sim\mathcal{Q}(\bar{g})} \frac{1}{m}\sum_{i=1}^{m} \|g^*(x_i) - g(x_i)\| + \frac{\|\bar{h}\|^2}{2\sigma_0^2 m^\alpha}. \tag{30}$$

Because g^* is unknown, the above objective is intractable. By applying the Triangle Inequality $\|g^*(x) - g(x)\| \leq \|g^*(x) - \varphi(y)\| + \|\varphi(y) - g(x)\|$, we separate the approximation and estimation errors respectively and can concentrate on the latter.

Thus, we consider the optimization objective in Eq. (31) instead of in Eq. (30):

$$\hat{J}(\bar{g}) = \mathop{\mathbb{E}}_{g\sim\mathcal{Q}(\bar{g})} \frac{1}{m}\sum_{i=1}^{m} \|\varphi(y_i) - g(x_i)\| + \frac{\|\bar{h}\|^2}{2\sigma_0^2 m^\alpha}. \tag{31}$$

Although $\hat{J}$ is convex, because a practical closed-form expression of $\hat{J}$ does not exist[2], we present two approximate approaches: a deterministic relaxation strategy in Section 5.1, then an estimation strategy using the so-called "log-prob" trick, and apply variance reduction in Section 5.2.

2. Computing the above expectation reduces to computing the mean of the norm of an isotropic Gaussian vector. The norm of a Gaussian vector follows a Generalized Rayleigh Distribution (see for example, Park (1961) or Blumenson and Miller (1963)). Park (1961) shows that there exists a closed-form expression of the moments of the Generalized Rayleigh distribution. However, these expressions depend on the confluent hypergeometric function, which is not closed for in our setting: it is worth noting that in Park (1961) the words *closed-form* are in quotes!

5.1 Relaxation strategy

We can upper-bound the $\hat{J}$ with an expectation-free convex objective $\hat{J}_c$. In this section, after establishing the upper-bound, we discuss optimizing $\hat{J}_c$.

5.1.1 Relaxation

In order to prove the relaxation in Eq. (34) we first bound the expected value of the norm of the regression residual.

Proposition 11 (Expected deviation upper-bound)**.** *Let $\mathcal{Q} = \mathcal{N}(W, \sigma^2 I_N)$ a multivariate Gaussian distribution, where (by abuse of notation) $W \in \mathbb{R}^{d \times d'}$ and $d \times d' = N$. Let $x \in \mathbb{R}^{d'}$ and $y \in \mathbb{R}^d$. Then,*

(a) $\mathbb{E}_{V \sim \mathcal{Q}} \|y - Vx\|^2 = \|y - Wx\|^2 + \sigma^2 d \|x\|^2$,

(b) $\mathbb{E}_{V \sim \mathcal{Q}} \|y - Vx\| \leq \sqrt{\sigma^2 d \|x\|^2 + \|y - Wx\|^2}$

Proof *(a)* is a classic result: a detailed proof is given for example in Giguère et al. (2013, Appendix 10.7.2.). For *(b)*, we successively apply Jensen's inequality (Eq. (32)) then use *(a)* (Eq. (33)), as detailed below:

$$\mathop{\mathbb{E}}_{V \sim Q} \|y - Vx\| \leq \sqrt{\mathop{\mathbb{E}}_{V \sim \mathcal{Q}} \|y - Vx\|^2}, \tag{32}$$

$$= \sqrt{\|y - Wx\|^2 + \sigma^2 d \|x\|^2}. \tag{33}$$

■

Remark 12. *We note that there may in fact several interpretations of (b) in the previous lemma, such as Jensen or the positivity of the variance.*

We can apply this lemma to upper-bound $\hat{J}(\bar{g})$ in Eq. (31):

$$\hat{J}(\bar{g}) \leq \hat{J}_c(\bar{g}) := \frac{1}{m} \sum_{k=1}^{m} \sqrt{\beta(x_k) + \left\|\varphi(y_k) - \bar{h} \circ X(x_k)\right\|^2} + \lambda_m^\alpha(t) \left\|\bar{h}\right\|^2, \tag{34}$$

where $\beta(x) = \sigma^2 \dim(\mathcal{H}) \|X(x)\|^2 = \sigma^2 \dim(\mathcal{H}) k(x, x)$ and regularization parameter $\lambda_m^\alpha(t) = \frac{1}{2t\sigma^2 m^\alpha} = \frac{\kappa^2}{2tm^{1-2\alpha}m^\alpha}$.

Remark 13 (Slackness of the relaxation)**.** *Is the upper-bound in Proposition 11 reasonable? A first remark, is that Jensen's inequality is known to be tight. However, we cannot quantify what is lost before training using the bounds (see Section 7). We evaluated the slackness of Proposition 11 (b) numerically in Figure 5.*

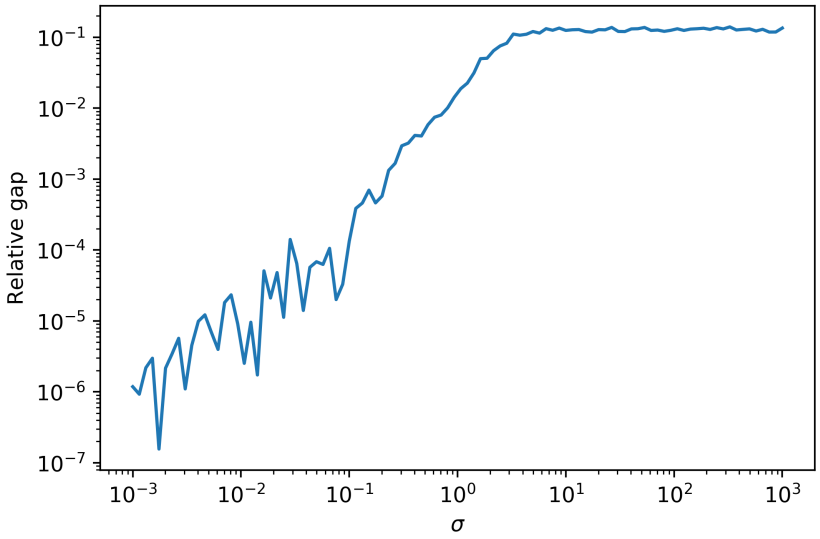

Figure 5: Relative gap in Proposition 11 (b). We plot $\frac{|L(\sigma)-R(\sigma)|}{R(\sigma)}$ where $L(\sigma) = \mathbb{E}_{V\sim\mathcal{Q}} \|y - Vx\|$ and $R(\sigma) = \sqrt{\sigma^2 d \|x\|^2 + \|y - Wx\|^2}$. Notice that the bound is at least tight to 10%. The simulation methodology can be found documented at https://github.com/theophilec/PAC-Bayes-ILE-Structured-Prediction in the source code associated to this paper (see Section 7).

5.1.2 Learning algorithm

In this section, we present a minimization method for $\hat{J}_c(W)$ over $W \in \mathcal{L}(\mathcal{F}, \mathcal{H})$. The resulting algorithm, based on gradient descent, is presented in Algorithm 1.

Proposition 14 (Properties of $\hat{J}_c$)**.** *$\hat{J}_c$ is differentiable and μ-strongly convex for any $\mu > 2\lambda_m^\alpha(t)$. Furthermore, $\nabla\hat{J}_c$ is given by*

$$\nabla\hat{J}_c(W) = \frac{1}{m}\sum_{k=1}^{m} \frac{(\varphi(y_k) - WX(x_k))\, X(x_k)^T}{\sqrt{\beta(x_k) + \|\varphi(y_k) - WX(x_k)\|^2}} + 2\lambda_m^\alpha(t)W,$$

thus $\hat{J}_c$ is twice-differentiable.

Proof Let $l(x) = \sqrt{a + x^2}$ where $a > 0$. l is increasing and convex: $l'(x) = \frac{2x}{\sqrt{a+x^2}} > 0$ and $l''(x) = \frac{4a}{2(a+x^2)^{3/2}} > 0$. Furthermore, $W \mapsto \|y - Wx\|$ is convex. Thus, $W \mapsto \sqrt{\beta(x) + \|y - Wx\|^2}$ is convex. The result follows by convex combination. Because the regularization penalty is μ strongly convex for any $\mu > 2\lambda_m^\alpha(t)$.

Algorithm 1: Gradient descent, `Relax-GD`

Input : Initial posterior mean W_0.
Parameters: Step-size parameters: $\nu \in (0.5, 1]$, $w \geq 0$.

Initialization $W^0 \leftarrow W_0$.
Pre-compute $\beta(x_k)$.
while *stopping criterion not met* **do**
  Compute $\nabla J_c(W_t)$ (see Proposition 14).
  Compute step-size: $\gamma_t = \frac{1}{(w+t)^\nu}$.
  Gradient step: $W^{t+1} \leftarrow W^t - \gamma_t \nabla J_c(W_t)$.
end
Result: W^{final}

Furthermore, if $f(W) = \|Y - WX\|^2$, for small H,

$$\begin{aligned} f(W+H) &= \left\|Y^T - X^T(W^T + H^T)\right\|^2, \\ &= \left\|Y^T - X^T W^T\right\|^2 + \left\|X^T H^T\right\|^2 - 2\langle Y^T - X^T W^T, X^T H^T\rangle, \\ &= \left\|Y - X^T W^T\right\|^2 + \left\|X^T H^T\right\|^2 - 2\operatorname{Tr}[(Y^T - X^T W^T)^T X^T H^T], \\ &= f(W) - 2\operatorname{Tr}[(Y - WX)X^T H^T] + o(H), \\ &= f(W) - 2\operatorname{Tr}[X(Y - WX)^T H] + o(H), \end{aligned}$$

where we used $\operatorname{Tr}(AB) = \operatorname{Tr}(BA)$ and $\operatorname{Tr}(A^T) = \operatorname{Tr}(A)$. The gradient expression follows by composition. ■

Because $\hat{J}_c$ is L-smooth (for some $L > 0$), it is well-known that we can minimize it by gradient descent Boyd and Vandenberghe (2004).

5.2 Stochastic Search Variational Minimization

In this section, we present a second approach to optimizing $\hat{J}$. Instead of relaxing the optimization problem, we seek to directly estimate $\nabla \hat{J}$. The intuition behind this approach is that $\hat{J}$ is convex, thus estimating its gradient could be sufficient to minimize it in practice. To this end, we apply the well-known "log-prob" trick. In order to reduce the variance of the resulting estimator, we apply techniques presented in Mohamed et al. (2020) and Paisley et al. (2012) with a well-chosen control variate.

These techniques are widely-used in different fields of mathematics and in particular machine learning: stochastic optimization, reinforcement learning, extreme-value statistics, among others. The interested reader can consult the recent survey Mohamed et al. (2020) and references therein.

5.2.1 Gradient estimation

Without loss of generality, we ignore the quadratic regularization in $\hat{J}$ and focus of the expectation term, using the following notation:

$$\hat{J}(W) = \mathop{\mathbb{E}}_{\mathcal{Q}(V|W)} [L(V)] ,$$

where $L(V) = \frac{1}{m} \sum_{k=1}^{m} \|\varphi(y_k) - VX(x_k)\|$ and $\mathcal{Q}(V|W)$ is the density of $\mathcal{Q}(W)$ at V.

The following proposition provides an expression of $\nabla \hat{J}$ at W and derives an unbiased estimator of $\nabla \hat{J}$.

Proposition 15 (Score-function gradient expression)**.** *$\nabla \hat{J}(W)$ can be expressed as the expectation of a function of $V \sim \mathcal{Q}(W)$ over $\mathcal{Q}(W)$:*

$$\nabla \hat{J}(W) = \mathop{\mathbb{E}}_{V \sim \mathcal{Q}(W)} [L(V) \nabla \log \mathcal{Q}(V|W)] .$$

Definition 16 (Score function estimator (SFE))**.** *Let $V_1, \ldots, V_M$ be M iid copies of $V \sim \mathcal{Q}(W)$. The score function estimator is defined as:*

$$\eta_M(W) = \frac{1}{M} \sum_{k=1}^{M} L(V_k) \nabla \log \mathcal{Q}(V_k|W). \tag{35}$$

Proposition 17. *η_M is an unbiased estimator of $\nabla \hat{J}(W)$.*

The proof of Proposition 17 stems directly from Proposition 15. Given the importance of the approach in our PAC-Bayes approach, we sketch out the main steps of the proof of Proposition 15.

As highlighted in Mohamed et al. (2020), proving Proposition 15 reduces essentially to interchanging the derivative and the expectation, and applying Lebesgue's dominated convergence theorem. Three conditions should be verified:

1. $\mathcal{Q}(v|W)$ is continuously differentiable with respect to W
2. $L(v)\mathcal{Q}(v|W)$ is integrable and differentiable with respect to W.
3. $L(v)\nabla\mathcal{Q}(v|W)$ is dominated by an integrable function g of v.

Proof The first two points are straight-forward for the multivariate Gaussian distribution. It is enough to verify the third over all bounded open sets of the space of linear functions. Let $\mathcal{C}$ be such a set and $C = \max\{\|W\| , W \in \mathcal{C}\}$.

Ignoring the normalization constant of $\mathcal{Q}$, we have:

$$\|L(V)\nabla\mathcal{Q}(V|W)\| = \|Y - VX\| \, \|V - W\| \exp\left(-\frac{1}{2\sigma^2} \|V - W\|^2\right). \tag{36}$$

Noting that $-\|V - W\|^2 = -\|V\|^2 - \underbrace{\|W\|^2}_{\geq 0} + 2\langle V, W\rangle \leq -\|V\|^2 + 2C\|V\|$ and $\|V - W\| \leq \|V\| + C$, we have:

$$\|L(V)\nabla\mathcal{Q}(V|W)\| \leq \|Y - VX\| \, (\|V\| + C) \exp\left(-\frac{1}{2\sigma^2}\left[\|V\|^2 - 2C\right]\right) =: \gamma(V).$$

Algorithm 2: Naïve score estimator gradient descent, `SF-GD`

Input : Initial posterior mean W_0.
Parameters: Step-size parameters: $\nu \in (0.5, 1]$, $w \geq 0$.

Initialization $W^0 \leftarrow W_0$.
while *stopping criterion not met* **do**
  Sample M predictors from $\mathcal{Q}$: $V_1, \ldots, V_M$.
  Estimate intractable gradient: $\eta_M \leftarrow \frac{1}{M} \sum_k f(V_k) \nabla \log \mathcal{Q}(V_K | W^t)$.
  Compute step-size: $\gamma_t = \frac{1}{(w+t)^\nu}$.
  Gradient step: $W^{t+1} \leftarrow W^t - \gamma_t \eta_M$.
end
Result: W^{final}

Because $\|V\|^2 \gamma(V) \to 0$ when $\|V\| \to \infty$, γ is integrable. ■

Remark 18. *In Proposition 15, L is not required to be differentiable.*

With this estimator of the gradient of $\hat{J}$, we can then minimize $\hat{J}$ using gradient descent. We present the associated algorithm in Algorithm 2.

The algorithm described above does not have the same theoretical guarantees as gradient descent, as the gradient estimate, while unbiased, can have large variance. There are several approaches to controlling the gradient's variance. We present an approach in Section 5.2.2.

5.2.2 Variance reduction

A general method of reducing the variance of an estimator is to introduce a *control variate*.

The main ingredient to this approach is the construction of a function $B(V)$ highly correlated with $L(V)$, so as to guarantee that $B(V)\nabla \log \mathcal{Q}(V|W)$ and $L(V)\nabla \log \mathcal{Q}(V|W)$ will also be correlated.

Let $B(V)$ be such a function, highly correlated with L. We define, for $a \in \mathbb{R}$,

$$\tilde{L}(V) := L(V)\nabla \log \mathcal{Q}(V) - a \left(B(V)\nabla \log \mathcal{Q}(V) - \mathop{\mathbb{E}}_{q} B(V)\nabla \log \mathcal{Q}(V) \right). \tag{37}$$

Note that $\mathbb{E}_{\mathcal{Q}} \tilde{L}(V) = \mathbb{E}_{\mathcal{Q}} L(V)\nabla \log \mathcal{Q}$. The goal is thus to choose a such that the variance of $\tilde{L}(V)$ is minimized.

Choice of a & estimation of a^*

Proposition 19 (Optimal baseline coefficient, Paisley et al., 2012)**.** *Given L and B, define $\tilde{L}$ as in Eq. (37). Then, $\mathbb{V}(\tilde{L})$ is minimized for $a^* \in \mathbb{R}$ given by:*

$$a^* = \frac{\sum_{k=1}^N \mathrm{Cov}\left(L \frac{\partial \log \mathcal{Q}}{\partial W_k}, B \frac{\partial \log \mathcal{Q}}{\partial W_k} \right)}{\sum_{k=1}^N \mathbb{V}\left(B \frac{\partial \log \mathcal{Q}}{\partial W_k} \right)}. \tag{38}$$

Of course, a^* is intractable in general. We use a plug-in estimator of it $\hat{a}$, by sampling from $\mathcal{Q}$, independently from the gradient samples. As recommended in

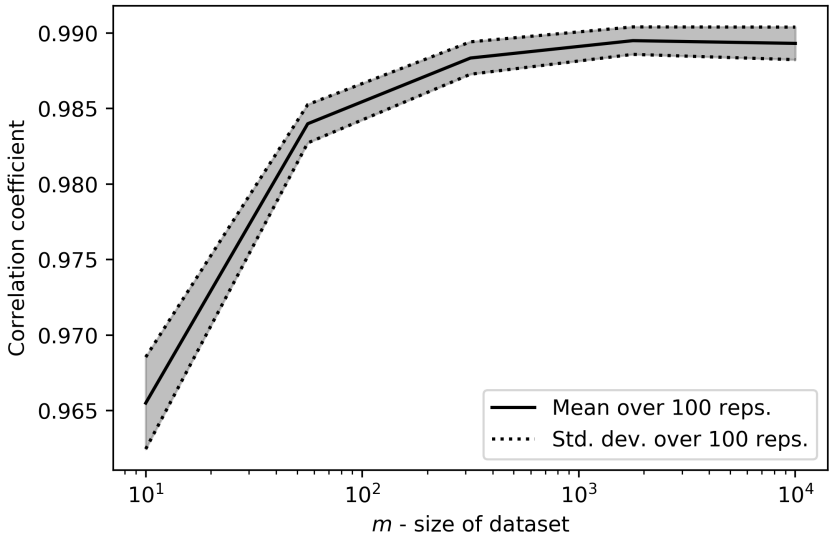

Figure 6: Numerical evaluation of correlation for $L(V) = \frac{1}{m}\sum_i^m \|y_i - Vx_i\|$ and $B(V) = \frac{1}{m}\sum_i^m \|y_i - Vx_i\|^2$. The correlation coefficient is given over $M = 500$ predictors sampled from a isotropic multivariate normal distribution. The coefficients are averaged over 100 experiments in order to estimate the variance of the estimate (which is materialized by the envelope and dotted lines). To reproduce, see https://github.com/theophilec/PAC-Bayes-ILE-Structured-Prediction.

Definition 20 (Optimal baseline coefficient estimator $\hat{a}$)**.** *Let $V^1, \dots, V^{M'}$ be M' iid samples from $\mathcal{Q}(V|W)$. Then,*

$$\hat{a} = \frac{\alpha}{\beta}, \tag{39}$$

where

$$\alpha = \frac{1}{M'}\sum_{j=1}^{M'}\sum_{k=1}^{N} \mathrm{Cov}\left(L\frac{\partial \log \mathcal{Q}(V^j)}{\partial W_k}, B\frac{\partial \log \mathcal{Q}(V^j)}{\partial W_k}\right), \tag{40}$$

$$\beta = \frac{1}{M'}\sum_{j=1}^{M'}\sum_{k=1}^{N} \mathbb{V}\left(B\frac{\partial \log \mathcal{Q}(V^j)}{\partial W_k}\right). \tag{41}$$

This is in particular the estimator proposed by Paisley et al. (2012) and Mohamed et al. (2020, associated code)[3].

Choice of B Of course, the variance reduction virtues of this approach rely on having a function B strongly correlated with L and for which we can compute the gradient of its

3. See: https://github.com/deepmind/mc_gradients/blob/master/monte_carlo_gradients/control_variates.py

expectation in closed form. Mohamed et al. (2020) present several alternative methods for choosing such a B. In particular, B does not need to be an upper-bound of L (contrary to the relaxation approach above).

We define $B(V)$ as:

$$B(V) = \frac{1}{m}\sum_{i=1}^{m} \|y_i - Vx\|^2 . \tag{42}$$

The intuition behind such a choice is that we have a closed-form expression of the gradient of B, and the norm and squared-norm are correlated as one is a non-trivial deterministic function of the other. Of course, L (B) are linear combinations of the norms (squared-norms) of residuals (respectively). However, the (x, y) pairs are drawn iid and because $x, y \mapsto \|y - Vx\|$ is deterministic, $\|y - Vx\|$ and $\|y' - Vx'\|$ are independent. Conditioned on the dataset, the correlation of L and B should remain strong.

Remark 21 (Numerical evaluation of correlation)**.** *We numerically evaluated our hypothesis that L and B as defined in Eq. (42) are well correlated, even when the correlation is reduced by summing over the dataset. We present the results in Figure 6. The correlation is strong, including for large datasets. The experiment is detailed in the associated codebase: https://github.com/theophilec/PAC-Bayes-ILE-Structured-Prediction.*

To summarize, the different approximations are as follows (again, ignoring the regularization term):

$$\begin{aligned}
\nabla \hat{J}(W) &= \mathop{\mathbb{E}}_{q}\left[L(V)\nabla \log \mathcal{Q}(V) + \nabla \ell(W)\right], \\
&= \mathop{\mathbb{E}}_{q}\left[L(V)\nabla \log \mathcal{Q}(V) - a\left(B(V)\nabla \log \mathcal{Q}(V) - \nabla \mathop{\mathbb{E}}_{q} B(V)\right)\right], \\
&= \mathop{\mathbb{E}}_{q}\left[(L(V) - aB(V))\nabla \log \mathcal{Q}(V)\right] + a\nabla \mathop{\mathbb{E}}_{q} B(V), \\
&\approx \underbrace{\mathop{\mathbb{E}}_{q}\left[(L(V) - \hat{a}B(V))\nabla \log \mathcal{Q}(V)\right]}_{\text{Sample}} + \hat{a}\underbrace{\nabla \mathop{\mathbb{E}}_{q} B(V)}_{\text{Tractable}}, \\
&\approx \underbrace{\frac{1}{M}\sum_{k=1}^{M}\left(L(V^k) - \hat{a}B(V^k)\right)\nabla \log \mathcal{Q}(V^k)}_{\hat{\eta}_M(W)} + \hat{a}\underbrace{\nabla \mathop{\mathbb{E}}_{q} B(V)}_{\eta_B(W)} .
\end{aligned} \tag{43}$$

where $V_1, \ldots, V^M$ are iid and independent from the M' samples used to estimate $\hat{a}$.

5.2.3 Algorithm

We present the complete algorithm using Stochastic Search Gradient Descent Paisley et al. (2012). To the best of our knowledge, this is a novel application of Stochastic Search Gradient Descent, the applications in the Paisley et al. (2012) being logistic regression and Hierarchical Dirichlet processes.

Algorithm 3: Quadratic Stochastic Search Gradient Descent iteration, `Q-SSGD`.

Input : Previous W^t.
Parameters: Step-size parameters: $\nu \in (0.5, 1]$, $w \geq 0$, M, M'.
Sample M' predictors $V_1, \ldots, V_{M'}$ from $\mathcal{Q}(W^t)$.
Estimate $\hat{a}$ according to Eq. (39).
Sample M predictors from $\mathcal{Q}(W^t)$, $V_1, \ldots, V_M$.
Estimate intractable gradient $\hat{\eta}_M \leftarrow \frac{1}{M_s} \sum_k (L(V_k) - \hat{a}B(V_k)) \nabla \log \mathcal{Q}(V_k|W^t)$.
Compute $\eta_B = \nabla \mathbb{E}\, B(V)$ under $\mathcal{Q}(W^t)$ according to Eq. (44).
Compute η_P the gradient of the penalty term.
Compute step-size $\gamma_t = \frac{1}{(w+t)^\nu}$.
Update $W^{t+1} \leftarrow W^t - \gamma_t \hat{\eta}_M(W^t) - \gamma_t \hat{a} \eta_B(W^t) - \gamma_t \eta_P$.
Result: W^{t+1}

As described above, two intractable quantities are sampled in order to estimate the gradient: the stochastic search correction $\hat{\eta}_M$ and $\hat{a}$ (as shown in Eq. (43)). Let M and M' be the number of samples used for each estimation, respectively[4].

We summarize the Stochastic Search Gradient Descent algorithm with the control variate described above in Algorithm 4. It repetitively calls Algorithm 3.

Proposition 22 provide expressions of the closed form gradients that are needed in the algorithm.

Proposition 22 (Control variate gradient η_B)**.**

$$\nabla \mathbb{E}_{\mathcal{Q}(W)} B(V) = \frac{2}{m} \sum_{k=1}^{m} \left(\varphi(y_k) - WX(x_k)\right) X(x_k)^T. \tag{44}$$

Proof

$$\begin{aligned} \nabla \mathbb{E}\, B(V) &= \nabla \left[\frac{1}{m} \sum_{k=1}^{m} \|\varphi(y_k) - WX(x_k)\|^2 + \sigma^2 \|X(x)\|^2 \dim(\mathcal{H}) \right], \\ &= \frac{1}{m} \sum_{k=1}^{m} \nabla \left[\|\varphi(y_k) - WX(x_k)\|^2 \right] = \frac{2}{m} \sum_{k=1}^{m} \left(\varphi(y_k) - WX(x_k)\right) X(x_k)^T. \end{aligned}$$

■

In this section, we presented two approaches to learning with Theorem 9 – bringing additional evidence that PAC-Bayes bounds are a principled way to derive new algorithms (see *e.g.* the ICML 2019 tutorial Guedj and Shawe-Taylor, 2019). In Section 7, we provide experimental results of to support theoretical results and discussion in Sections 3 to 5.

6. Discussion

In this paper, we provide a theoretical analysis of the properties of estimators in the Implicit Loss Embedding framework. Our analysis is theoretical and methodological in nature. On

4. Note that Paisley et al. (2012) consider $M' \ll M$.

Algorithm 4: Quadratic Stochastic Search Gradient Descent algorithm.

Input : Initial regressor W^0.
Parameters: Step-size parameters: $\nu \in (0.5, 1]$, $w \geq 0$, M, M'.
while *stopping criterion not met* **do**
  $W^{t+1} \leftarrow$ `Q-SSGD`(W^t)
  $t \leftarrow t + 1$
end
Result: W^t

one hand, we prove PAC-Bayes generalization bounds for ILE losses. These seek to upper-bound the task risk and the excess task risk, from the performance of a given predictor distribution on the dataset. In turn these theoretical results yield two important methodological contributions.

In this section, we briefly summarize our main contributions at a high-level, tying together the different sections and approaches presented therein. Furthermore, we put our results into perspective with respect to those presented in Ciliberto et al. (2020).

6.1 Generalization

In Theorem 5, we prove a generalization bound for the ILE problem stemming from a classic PAC-Bayes bound from Zhang (2006). It is important to note that in Theorem 5, the loss function Δ is ILE but the bound hold irrespective of how the predictor (distribution) $\mathcal{Q}$ is obtained. This differs from the approach adopted in Ciliberto et al. (2020) which consider convergence rates of specific algorithms.

The bound offers methodological guarantees. In the case where m is very large, the generalization penalty term is small, so with probability $1 - \delta$, the generalization gap (i.e. the difference between $\mathbb{E}_{\mathcal{Q}} \mathcal{E}(f)$ and $\mathbb{E}_{\mathcal{Q}} \mathcal{E}_m(f)$) reduces to the gap induced by exponential bound on the identity (Proposition 6). For small values of $\mathbb{E}_{\mathcal{Q}} \mathcal{E}_m(f)$, this bound is tight (see Figure 2). This is expected: in the infinite data limit, we expect the empirical loss to be close to the distribution loss.

Furthermore, generalization performance is limited by a penalization term formed of the Kullback-Leibler divergence of the posterior and prior distributions. In practice, this yields a sound way of penalizing predictor (distributions) that do not generalize well. Indeed, in the simple multivariate normal case, if the distributions have the same variance, the penalization coincides with the widely used Tikhonov regularization.

6.2 Comparison with KDE generalization bounds

Ciliberto et al. (2020) offers insight into the links between the Kernel Dependency Estimation approach and the ILE framework. In this section, we tie together the PAC-Bayes approach to generalization and the ILE approach to consistency, by recalling some results from Giguère et al. (2013).

We consider an output space $\mathcal{Y}$ associated to the reproducing kernel $k_{\mathcal{Y}}$ defined by $k_{\mathcal{Y}}(y, y') = \langle Y(y), Y(y') \rangle$, $\forall y, y' \in \mathcal{Y}^2$. We assume the task loss is defined from the output

kernel $k_{\mathcal{Y}}$ as follows:

$$L(y, y') = \frac{1}{2}\left\|Y(y) - Y(y')\right\|^2 = \frac{1}{2}k_{\mathcal{Y}}(y, y) + \frac{1}{2}k_{\mathcal{Y}}(y', y') - k_{\mathcal{Y}}(y, y'). \tag{45}$$

The KDE predictor f_k is the decoded linear regressor, defined by:

$$f_k(x) = \arg\min_{y \in \mathcal{Y}} \|Y(y) - WX(x)\|^2 . \tag{46}$$

One key insight in the KDE approach is to notice that:

$$L(y, f_k(x)) \leq 2\|Y(y) - WX(x)\|^2 , \tag{47}$$

where $W \in \mathcal{L}(\mathcal{F}, \mathcal{H})$. In this sense, the KDE framework is a quadratic surrogate approach, without any consistency guarantees as is.

Giguère et al. (2013) establishes a PAC-Bayes generalization bound in the KDE setting. The bound is based on the pointwise quadratic upper-bound of the KDE loss we presented in Eq. (47) and bounds the expected risk of the mean KDE predictor by the empirical regression risk of the mean regressor, $\mathcal{R}_m(\bar{g}) = \frac{1}{m}\sum_{i=1}^m \|Y(y_i) - \bar{g}(x_i)\|^2$ and $\bar{g} = \bar{w} \circ X$ where $w \in \mathcal{L}(\mathcal{F}, \mathcal{H})$.

Recall that we assume $\mathcal{F}$ (the RKHS associated to k) and $\mathcal{H}$ (the Hilbert space associated to the ILE) to be finite-dimensional. We consider regressors $g = w \circ X$ where $w \in \mathcal{L}(\mathcal{F}, \mathcal{H})$.

Theorem 23 (Giguère et al., 2013)**.** *Assume k is such that $k(x, x) = 1$, $\forall x \in \mathcal{X}$. For any $\delta > 0$, with probability at least $1 - \delta$ over training set sampled iid from ρ, for any $w \in \mathcal{L}(\mathcal{F}, \mathcal{H})$, with $g = w \circ X$,*

$$\mathcal{E}(d \circ g) \leq \frac{5e}{e-1}\left[1 - \exp\left(-2\mathcal{R}_m(g) - \frac{\frac{9}{8}\|w\|^2 + \log\left(\frac{1}{\delta}\right)}{m}\right)\right]. \tag{48}$$

Notice that in the above bound, the predictors are deterministic and not stochastic. Indeed, using Gaussian distributions for w, the expectations in Theorem 4 were explicitly computed, making the bound only depend on the mean predictor associated to $w = \bar{w}$.

As Ciliberto et al. (2020) point out, under certain hypotheses, we can show that KDE is a special case of ILE and transfer consistency guarantees of ILE predictors to KDE predictors. Thus the surrogate method in the ILE framework can also be seen as the minimization of a PAC-Bayes bound. Even though this is the case, minimizing the upper bound in Theorem 23 does not yield the same rate guarantees as in Ciliberto et al. (2020) which hold for a regularization parameter equal to $\frac{1}{\sqrt{m}}$.

6.3 Consistency

Existing approaches do not yield results about the gap between $\mathcal{E}(d \circ g) - \mathcal{E}(f^*)$, from a given predictor and a finite sample of data.

Indeed, Theorem 24 provides tight bounds of $\mathcal{E}(f_m) - \mathcal{E}(f^*)$ for finite m. However, these results depend on how f_m is obtained from the data sample. For example, kernel ridge regression can be used (with regularization parameter $\lambda_m = \frac{1}{\sqrt{m}}$, see Caponnetto and Vito, 2006). For completeness we recall the (abridged) result:

Theorem 24 (Ciliberto et al., 2020, from Theorem 5)**.** *If $\mathcal{Z}$ is compact and Δ admits an ILE. Assume that k is continuous and bounded by $\kappa^2 = \sup_{x \in \mathcal{X}} k(x, x)$. Let ρ be a data-generating distribution such that $g^* \in \mathcal{H} \otimes \mathcal{F}$ where $\mathcal{H}$ is associated to the ILE and $\mathcal{F}$ to k. Let $\delta > 0$ and m_0 large enough, then with probability at least $1 - \delta$ over the training set sampled from ρ^m for f_m (KRR with the same regularization as above),*

$$\mathcal{E}(f_m) - \mathcal{E}(f^*) \leq \frac{c_\Delta M q \log \frac{4}{\delta}}{m^{1/4}}, \tag{49}$$

where $M = 16 \left(\kappa \left(1 + \kappa \left\| g^* \right\| \right) + \kappa \sqrt{1 + \left\| g^* \right\|^2} + \left\| g^* \right\| \right)$ *and* $q \leq 3$.

In constrast, in Theorem 9, we bound a stochastic predictor analog to $\mathcal{E}(f_m) - (f^*)$. This approach is complementary to that of Ciliberto et al. (2020). It provides a theoretical guarantee bounding the suboptimality of a given predictor distribution, and dependent on the finite sample performance of the bound. In particular, Theorem 9 holds with high probability, independently of the learning method used.

Theorem 9 also yields methodological insight into the Structured prediction problem. Indeed, it materializes the gain obtained using "structured" losses such as the Hamming loss, over the $0 - 1$ loss, theoretically validating practitioner intuition. Furthermore, we show how prior information such as the bound on the input kernel κ can have an impact of the chosen learning method.

Theorem 9 is of particular interest because the upper-bound is differentiable. In Section 5, we derive two learning algorithms that minimize the bound. The experimental results of this approach is presented in Section 7.

7. Experimental results

In this paper, we prove a PAC-Bayes bound for Structured prediction using ILE losses. The bound provides an excess risk certificate that holds with high probability. Furthermore, we derive two learning algorithms from the bound, which both stem from minimizing a surrogate of the bound.

In this section, we present experimental results of three algorithms, on multi-label binary classification datasets. All experiments can be reproduced with source code available[5].

7.1 Algorithms

First, we recall the different implement algorithms and their variants. We compare three families of algorithms:

- `ILE(`λ`)`: minimize Eq. (9) by Kernel Ridge Regression with regularization parameter λ.
- `Relaxed-PB(`α, t`)`: minimize Eq. (34) with gradient descent (with constant learning rate γ). This is described in Algorithm 1.

5. https://github.com/theophilec/PAC-Bayes-ILE-Structured-Prediction

- `MC-PB(`α, t`)`: minimize Eq. (30) with score-function gradient descent (with constant learning rate γ).

For `Relaxed-PB` and `MC-PB`, α and t are parameters of the prior $\mathcal{P}$ so are, in general, data-independent. See Section 7.3 for more details on the choice of α and t.

7.2 Implementation

In our work, we consider the PAC-Bayes framework: predictors f are sampled from a posterior distribution $\mathcal{Q}$, itself learned from dataset $\mathcal{S} = \{(x_1, y_1), \dots, (x_m, y_m)\}$. Recall that prior to learning, we define a prior distribution over predictors $\mathcal{P}$, which is independent from $\mathcal{S}$. The main difference between the PAC-Bayes approach and more general approaches is thus that we do not learn a predictor through empirical risk minimization but seek to estimate an optimal posterior distribution which minimizes a bound. In Section 5 we showed how this can be reduced to minimizing a data-dependent penalized empirical risk objective.

Furthermore, in the ILE setting, learning a predictor $f : \mathcal{X} \to \mathcal{Z}$ (or predictor distribution $\mathcal{Q}$) is equivalent to learning a regressor $g : \mathcal{X} \to \mathcal{H}$ (or $\mathcal{Q}$). See Figure 1 for an overview. In Section 5, we present two alternative ways of learning $\mathcal{Q}$ over regressors : $g : \mathcal{X} \to \mathcal{H}$.

The combination of the ILE and PAC-Bayes settings in materialized in our implementation. Indeed, two main objects are implemented:

- `Regressor` encodes data in $\mathcal{X}$ to $\mathcal{H}$, akin to g (and to the mean of the posterior distribution $\mathcal{Q}$). The encoding is learned using one of the three families of algorithms mentioned above: `ILE`, `Relaxed-PB` and `MonteCarlo-PB`. As expected, in the case where `ILE` is employed, `Regressor` coincides with a Kernel ridge regressor.

- `StochasticPredictor` is a complete predictor from $\mathcal{X}$ to $\mathcal{Z}$, akin to $f \sim \mathcal{Q}$. It has a `Regressor` attribute and uses it to learn (`stochastic_predictor.regressor.fit` is called) and perform inference. However, inference can be performed in one of two ways: deterministically from the mean of the posterior distribution $\mathcal{Q}$ (this reproduces the behavior of an ordinary predictor) or stochastically by sampling M predictors $f_1, \dots, f_m$ from $\mathcal{Q}$ and returning the M corresponding predictions. Similarly, inference can be performed on regression by sampling $g_1, \dots, g_m$ from $\mathcal{Q}$.

Remark 25. *Given $x \in \mathcal{X}$, our stochastic predictor returns $f_1(x) = d \circ g_1(x), \dots, f_M(x) = d \circ g_M(x)$ where $g_1, \dots, g_M$ are sampled iid from $\mathcal{Q}$. It is important to note that it does not return $\frac{1}{M}\sum_k f_k(x)$ (in the structured setting, this quantity is not defined) nor $d \circ \left(\frac{1}{M}\sum_k g_k(x)\right)$.*

Remark 26 (Decoding)**.** *Recall that given an ILE loss $\Delta(z, y) = \langle \psi(y), \varphi(y) \rangle$ and a regressor g, the ILE framework defines f by $f = d \circ g$ where:*

$$d(g(x)) = \arg\min_{z \in \mathcal{Z}} \langle \psi(z), g(x) \rangle.$$

We mentioned in Section 2 that this decoder (or arg min *step) can be solved efficiently in certain cases. See for example Nowak-Vila et al. (2019b); Blondel (2019); Mensch and Blondel (2018). In our implementation, d is a naïve* arg max *over all possible labels in $\mathcal{Z}$. Note however, that d is not used during training.*

Both of these objects are implemented so as to be compatible with the popular `scikit-learn` library (Buitinck et al., 2013; Pedregosa et al., 2011), in order to maximize their familiarity.

In our implementations of gradient descent and score-function gradient descent, we make use of automatic differentiation and safe pseudo-random number generation from the `jax` library (Bradbury et al., 2018). This presents several advantages: (i) our code is entirely written in `Python`, meaning it can be easily read and extended all the whilst being fast and memory efficient; (ii) reproduction is easy despite multiple, nested sampling steps that take place at training and at inference; (iii) although not currently implemented, our code can be compiled to XLA and run on a GPU.

We use the `emotions` dataset from `scikit-multilearn`, a multi-label learning library for `Python` (Szymański and Kajdanowicz, 2017). The dataset presents $m = 593$ examples (we concatenate the training and test folds for our experiments), $\dim(\mathcal{X}) = 73$ and $\ell = 6$ binary labels. We use the linear kernel so $\mathcal{F} = \mathcal{H}$.

7.3 Choice of the prior $\mathcal{P}(\alpha, t)$

It is well-known that the choice of prior can help or hinder algorithm performance. In our case, the prior variance (we consider its mean to be 0), which varies with α and t determines the behaviors of the bound and derived algorithms.

Bound behavior We examined the variation of the penalty term ε' in Section 4. Here we consider the behavior of empirical risk term for the PAC-Bayes predictor associated to the ILE regressor. In this case, the mean of $\mathcal{Q}$ is independent of α and t. However, its variance σ^2 is chosen as a function of α and t. This impacts the bound value. Note that we cannot compute it explicitly (it depends on intractable quantities such as g^*). In Figure 7, we illustrate the dependence of the empirical risk part of the risk certificate $\hat{J}$, defined in Eq. (31), on α and t. Figure 7 helps to illustrate the behavior of the ILE predictor in our PAC-Bayes setting, and gives an idea of how α and t can impact the bound. Note that we are not offering this grid search experiment as a method of model selection. Indeed, the prior is assumed to be data independent and thus cannot be, without additional considerations, be optimized using bound values or predictor performance on the training set.

Algorithm behavior The choice of prior also impacts the behavior of the algorithms presented in this work: gradient descent on a relaxed objective or Monte Carlo gradient descent. Indeed, the relaxed objective presents terms which depend on the posterior variance. In the Monte Carlo approach, predictors are sampled from the posterior in order to optimize its mean. The behavior varies when σ is large or small.

Figures 8 and 9 illustrates bound surrogate values ($\hat{J}$ is computed for all three algorithms) for different values of t and α. A complementary picture is given by the dependence of σ and $\lambda_m^\alpha(t)$ on α and t, see Figures 10 and 11.

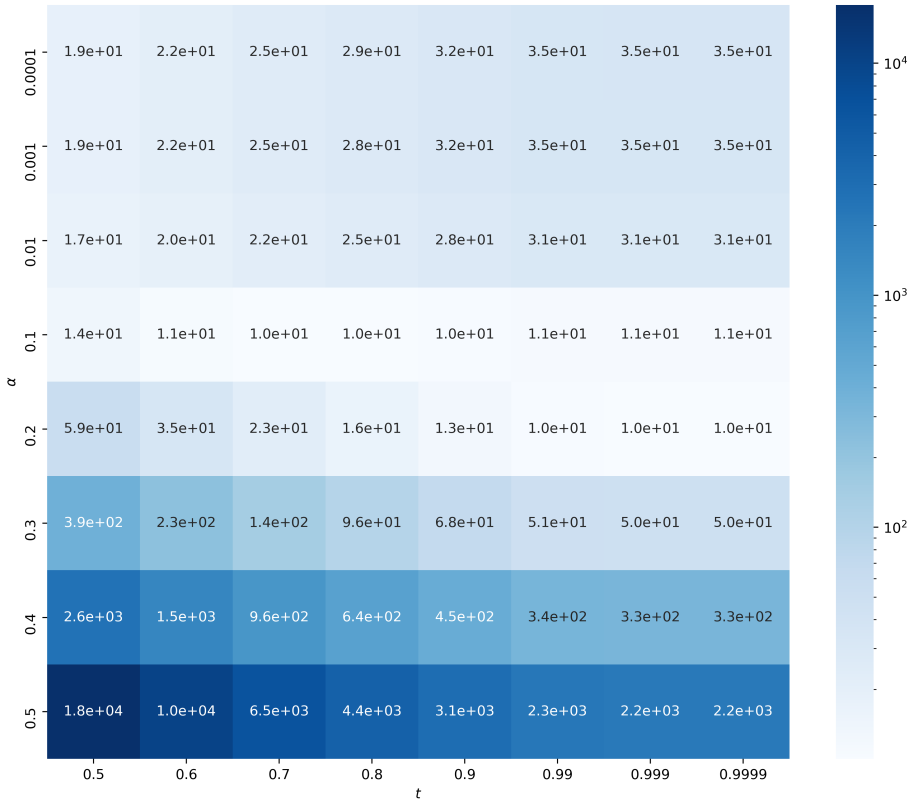

Figure 7: $\hat{J}$ as a function of α and t for the `ILE` algorithm. $\hat{J}$ is defined in Eq. (31). The regularization parameter λ is chosen to minimize $\hat{J}$.

References

R. Amit and R. Meir. Meta-learning by adjusting priors based on extended pac-bayes theory. In J. G. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 205–214. PMLR, 2018. URL http://proceedings.mlr.press/v80/amit18a.html.

A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, Dec. 2008.

F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. *Twenty-first international conference on Machine learning - ICML '04*, 2004.

P. Bartlett, M. I. Jordan, and J. D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 2006.

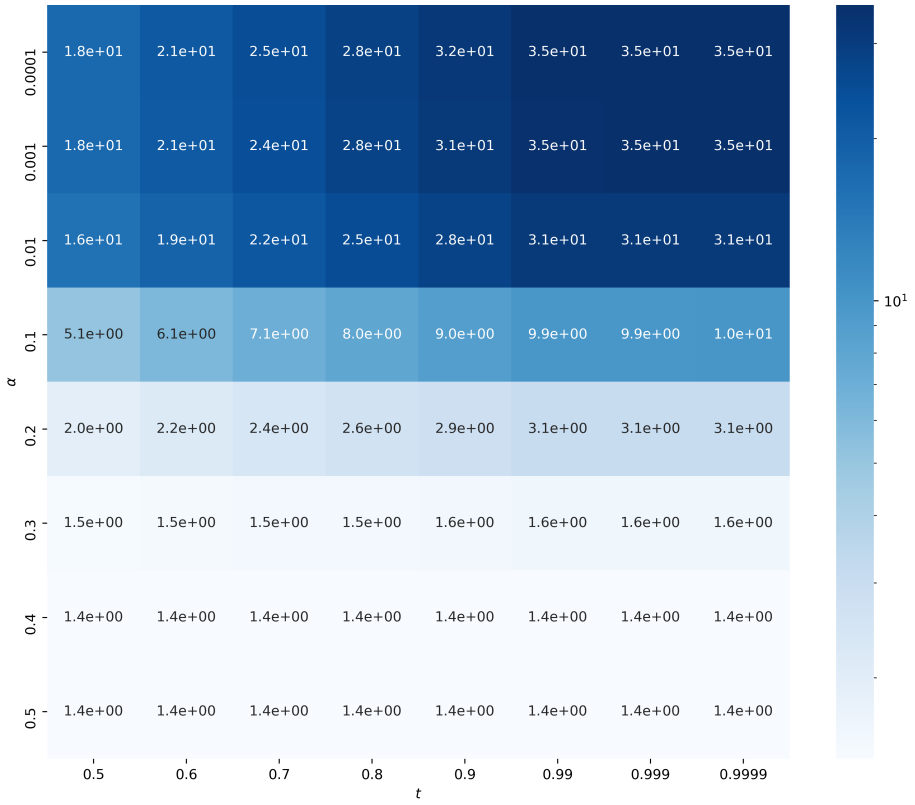

Figure 8: $\hat{J}$ as a function of α and t for the `Relaxed-PB` algorithm. $\hat{J}$ is defined in Eq. (31). The learning rate is chosen between $\{10^{-8}, 10^{-4}, 10^{-3}, 10^{-2}\}$ to minimize $\hat{J}$.

M. Blondel. Structured prediction with projection oracles. *Advances in Neural Information Processing Systems*, pages 12145–12156, 2019.

L. E. Blumenson and K. S. Miller. Properties of Generalized Rayleigh Distributions. *The Annals of Mathematical Statistics*, 34(3):903–910, 1963.

S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

C. Brouard, M. Szafranski, and F. d'Alché-Buc. Input output Kernel regression : supervised and semi-supervised structured output prediction with operator-valued kernels. *Journal of Machine Learning Research*, 17, 2016.

L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and

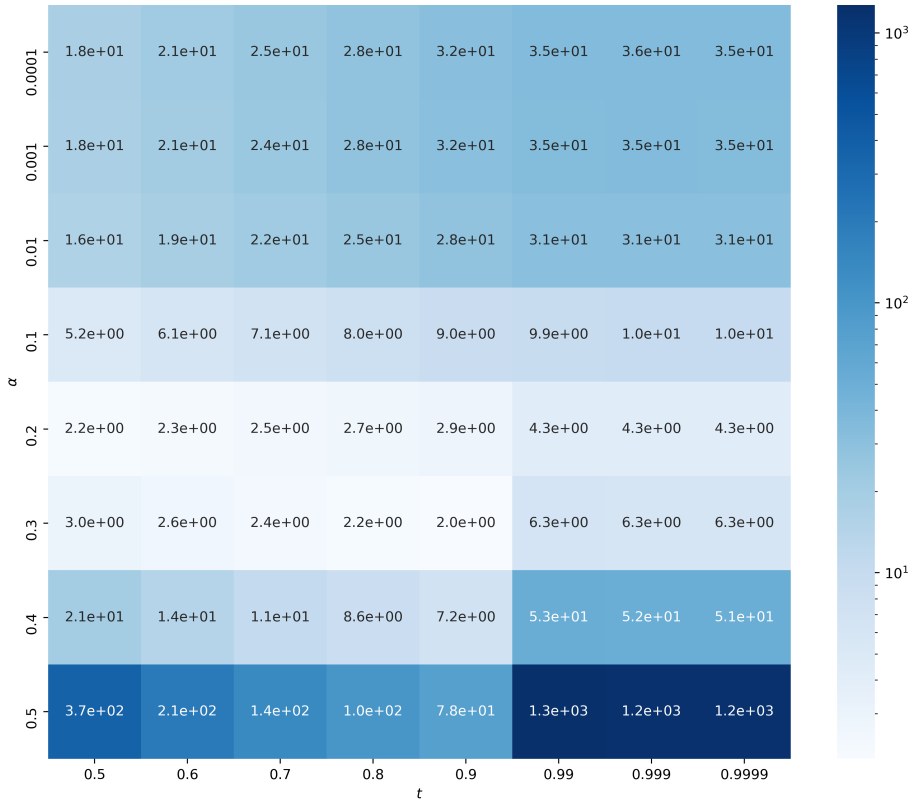

Figure 9: $\hat{J}$ as a function of α and t for the `MC-PB` algorithm. $\hat{J}$ is defined in Eq. (31). The learning rate is chosen between $\{10^{-5}, 10^{-4}, 10^{-3}\}$ to minimize $\hat{J}$. $M = 20$ samples where used at each gradient step (with $\hat{a} = 0$).

G. Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

A. Caponnetto and E. D. Vito. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2006.

O. Catoni. *Statistical learning theory and stochastic optimization: Ecole d'Eté de Probabilités de Saint-Flour XXXI-2001*. Springer, 2004.

O. Catoni. *PAC-Bayesian Supervised Classification: The Thermodynamics of Statistical Learning*. Institute of Mathematical Statistics lecture notes-monograph series. Institute of Mathematical Statistics, 2007. ISBN 9780940600720. URL https://books.google.fr/books?id=acnaAAAAMAAJ.

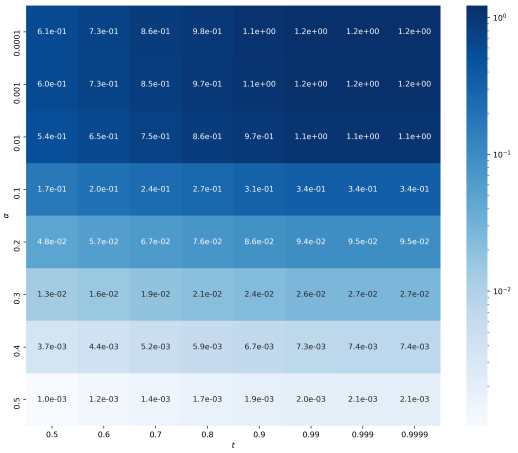

Figure 10: σ as a function of α and t.

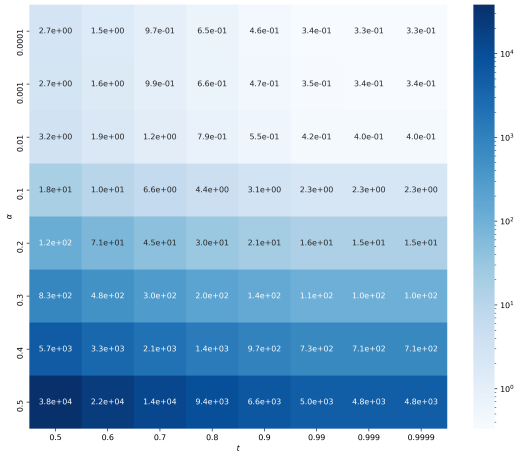

Figure 11: λ as a function of α and t.

C. Ciliberto, L. Rosasco, and A. Rudi. A consistent regularization approach for structured prediction. In *Advances in Neural Information Processing Systems*, pages 4412–4420, 2016.

C. Ciliberto, F. Bach, and A. Rudi. Localized structured prediction. In *Advances in Neural Information Processing Systems 32*, pages 7301–7311. Curran Associates, Inc., 2019.

C. Ciliberto, L. Rosasco, and A. Rudi. A general framework for consistent structured prediction with implicit loss embeddings. *Journal of Machine Learning Research (JMLR)*, 2020.

C. Cortes, M. Mohri, and J. Weston. A general regression framework for learning string-to-string mappings. In *Predicting Structured Data*. MIT Press, 2007.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Stochastic Modelling and Applied Probability. Springer New York, 1997.

G. K. Dziugaite and D. M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In G. Elidan, K. Kersting, and A. T. Ihler, editors, *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, August 11-15, 2017*. AUAI Press, 2017. URL http://auai.org/uai2017/proceedings/papers/173.pdf.

G. K. Dziugaite and D. M. Roy. Entropy-sgd optimizes the prior of a pac-bayes bound: Generalization properties of entropy-sgd and data-dependent priors. In J. G. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 1376–1385. PMLR, 2018a. URL http://proceedings.mlr.press/v80/dziugaite18a.html.

G. K. Dziugaite and D. M. Roy. Data-dependent pac-bayes priors via differential privacy. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and

R. Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 8440–8450, 2018b. URL http://papers.nips.cc/paper/8063-data-dependent-pac-bayes-priors-via-differential-privacy.

S. Giguère, F. Laviolette, M. Marchand, and K. Sylla. Risk Bounds and Learning Algorithms for the Regression Approach to Structured Output Prediction. *ICML*, page 8, 2013.

S. Grünewälder, G. Lever, L. Baldassarre, S. Patterson, A. Gretton, and M. Pontil. Conditional mean embeddings as regressors. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, page 1803–1810, 2012.

B. Guedj. A Primer on PAC-Bayesian Learning. In *Proceedings of the second congress of the French Mathematical Society*, 2019. URL https://arxiv.org/abs/1901.05353.

B. Guedj and J. Shawe-Taylor. A Primer on PAC-Bayesian Learning. https://bguedj.github.io/icml2019/material/main.pdf, 2019. ICML 2019 tutorial, talk given on 2019-06-10.

M. Haddouche, B. Guedj, O. Rivasplata, and J. Shawe-Taylor. PAC-Bayes unleashed: generalisation bounds with unbounded losses. *arXiv preprint arXiv:2006.07279*, 2020. URL https://arxiv.org/abs/2006.07279v2.

J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

G. Letarte, P. Germain, B. Guedj, and F. Laviolette. Dichotomize and generalize: Pac-bayesian binary activated deep neural networks. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 6869–6879, 2019. URL https://papers.nips.cc/paper/8911-dichotomize-and-generalize-pac-bayesian-binary-activated-deep-neural-networks.

D. Liu, M. Bober, and J. Kittler. Visual semantic information pursuit: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

M. Marchand, H. Su, E. Morvant, J. Rousu, and J. S. Shawe-Taylor. Multilabel Structured Output Learning with Random Spanning Trees of Max-Margin Markov Networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 873–881. Curran Associates, Inc., 2014.

D. A. McAllester. Some PAC-Bayesian theorems. In *Proceedings of the eleventh annual conference on Computational Learning Theory*, pages 230–234. ACM, 1998.

D. A. McAllester. PAC-Bayesian model averaging. In *Proceedings of the twelfth annual conference on Computational Learning Theory*, pages 164–170. ACM, 1999.

D. A. McAllester. Generalization Bounds and Consistency for Structured Labeling. In *Predicting Structured Data*. The MIT Press, 2007.

A. Mensch and M. Blondel. Differentiable dynamic programming for structured prediction and attention. *ICML*, 2018.

Z. Mhammedi, B. Guedj, and R. C. Williamson. PAC-Bayesian Bound for the Conditional Value at Risk. Submitted., 2020. URL https://arxiv.org/abs/2006.14763.

S. Mohamed, M. Rosca, M. Figurnov, and A. Mnih. Monte Carlo Gradient Estimation in Machine Learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020.

A. Nowak-Vila, F. Bach, and A. Rudi. Sharp analysis of learning with discrete losses. In K. Chaudhuri and M. Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89, pages 1920–1929. PMLR, 2019a.

A. Nowak-Vila, F. Bach, and A. Rudi. A General Theory for Structured Prediction with Smooth Convex Surrogates. *arXiv:1902.01958 [cs, stat]*, 2019b. URL https://arxiv.org/pdf/1902.01958.pdf.

A. Nowak-Vila, F. Bach, and A. Rudi. Consistent structured prediction with max-min margin markov networks. In *International Conference of Machine Learning (ICML)*, page to appear, 2020.

S. Nowozin and C. H. Lampert. Structured learning and prediction in computer vision. *Found. Trends. Comput. Graph. Vis.*, 6(3–4):185–365, 2011.

K. Nozawa, P. Germain, and B. Guedj. Pac-bayesian contrastive unsupervised representation learning. In *UAI*, 2020. URL https://arxiv.org/abs/1910.04464.

A. Osokin, F. Bach, and S. Lacoste-Julien. On structured prediction theory with calibrated convex surrogate losses. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 302–313. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/38db3aed920cf82ab059bfccbd02be6a-Paper.pdf.

J. Paisley, D. Blei, and M. Jordan. Variational Bayesian Inference with Stochastic Search. In J. Langford and J. Pineau, editors, *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, ICML '12, pages 1367–1374, New York, NY, USA, July 2012. Omnipress.

J. H. Park. Moments of the Generalized Rayleigh Distribution. *Quarterly of Applied Mathematics*, 19(1):45–49, 1961.

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.

K. B. Petersen and M. S. Pedersen. The matrix cookbook, vol. 7. 2008. URL https://www.math.uwaterloo.ca/~hwolkowi//matrixcookbook.pdf.

A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 2008.

A. Rudi, C. Ciliberto, G. Marconi, and L. Rosasco. Manifold structured prediction. In *Advances in Neural Information Processing Systems*, pages 5611–5622, 2018.

M. Seeger. PAC-Bayesian Generalization Error Bounds for Gaussian Process Classification. *Journal of Machine Learning Research*, 3, 08 2002.

J. Shawe-Taylor and R. C. Williamson. A PAC analysis of a Bayes estimator. In *Proceedings of the 10th annual conference on Computational Learning Theory*, pages 2–9. ACM, 1997.

P. Szymański and T. Kajdanowicz. A scikit-based Python environment for performing multi-label classification. *ArXiv e-prints*, Feb. 2017.

B. Taskar, C. Guestrin, and D. Koller. Max-Margin Markov Networks. In S. Thrun, L. K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 25–32. MIT Press, 2004.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In C. E. Brodley, editor, *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004.

L. G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, Nov. 1984.

V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1–2):1–305, Jan. 2008.

J. Weston, O. Chapelle, V. Vapnik, A. Elisseeff, and B. Schölkopf. Kernel dependency estimation. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 897–904. MIT Press, 2003.

T. Zhang. Information-theoretic upper and lower bounds for statistical estimation. *IEEE Transactions on Information Theory*, 52(4):1307–1321, 2006.