Optimal PAC-Bayes Bounds and their Variational Approximations

Szilvia Ujváry

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Master of Philosophy in Machine Learning and Machine Intelligence

Emmanuel College

October 2023

Dedicated to my loving parents and late grandfather.

Declaration

I, Szilvia Ujváry of Emmanuel College, being a candidate for the MPhil in Machine Learning and Machine Intelligence, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

The software used throughout this thesis relies on the open-source bnn-hmc JAX codebase [Izmailov et al., 2021] for HMC sampling. The bnn-hmc source code[1] is modified in the following ways

- Added prior mean initialization from truncated Gaussian
- Modified target log density to the Gibbs log joint
- Added likelihood tempering
- Added new models and datasets
- Added scripts to evaluate ESS, MCMC SE and $\hat{R}$ and ECE diagnostics from the individual samples
- Added scripts for sample aggregation and bound computation

I further used the open-source PBB codebase[2][Perez-Ortiz et al., 2020] to carry out the comparison with MFVI. Only new models, datasets, and small scripts to calculate standard errors were added to this source code.

Word count: 14981

Szilvia Ujváry
October 2023

[1] bnn-hmc Github: https://github.com/google-research/google-research/tree/master/bnn_hmc
[2] PBB Github: https://github.com/mperezortiz/PBB

Acknowledgements

I would like to thank my supervisor, José Miguel Hernández-Lobato for giving feedback on my work and nudging me in the right directions. I am also grateful to my second supervisor, Gergely Flamich for his many interesting ideas and encouragement throughout this project. I would like to thank Vincent Fortuin and Ferenc Huszár for insightful conversations.

I would like to thank my family and friends who have supported me through a very challenging year.

Last but not least, I am grateful to Emmanuel College for my scholarship, which made my studies in Cambridge possible.

Abstract

Understanding generalization in neural networks is a key question in the theory of machine learning. PAC-Bayes theory formalizes this question and allows us to empirically compute upper bounds on the generalization error of specific neural networks, that hold with high probability. One of the goals of this framework is to eliminate the need for test data, since the probabilistic guarantees are computed from the train data alone. However, many of these bounds are loose in that they can only guarantee performance that is much worse than the network's performance on test datasets and hence are not yet practically useful.

In this thesis, we study a potential cause of this looseness: the variational approximation that is most commonly used to estimate PAC-Bayes bounds. We propose a method to evaluate the accuracy of variational approximations of bounds by comparing them to the true minimizer of the bound, given by the generalised PAC-Bayes posterior, a Gibbs distribution.

Our approach to estimating this true minimizer of PAC-Bayes bounds consists of three steps. Firstly, we approximately sample from the target Gibbs distribution using Hamiltonian Monte Carlo, a Markov chain Monte Carlo technique. Then, we estimate its KL divergence from the PAC-Bayes prior, which is needed to form the bound. We reduce this task to estimating the log normalizing constant of the Gibbs distribution, which we achieve via thermodynamic integration. Our final step is ensuring that our estimate is valid with high probability.

We apply our method to estimate optimal PAC-Bayes bounds on MNIST and related datasets and compare these to the bounds supplied by the variational approximation. We extensively test our estimates with Markov chain convergence diagnostics. Our optimal bound estimates improve on neural network accuracy guarantees obtained by variational approximations, in all the setups considered. However, variational approximations approach our estimates of the optimal bound as model depth is increased. We further find that adding more data improves the tightness of our optimal bound estimates more than the variational approximations.

Table of contents

List of figures

List of tables

Nomenclature

Abbreviations

AIS Annealed Importance Sampling

BNN Bayesian Neural Network

CLT, MCCLT Central Limit Theorem and Markov chain Central Limit Theorem

ECE Expected Calibration Error

ESS Effective sample size

HMC Hamiltonian Monte Carlo

MCMC SE Markov chain Monte Carlo standard error

MCMC Markov chain Monte Carlo

MFVI Mean-field variational inference

RC Risk certificate

Notation

$\mathcal{M}_1(\mathcal{W})$ Probability measures on space $\mathcal{W}$

$\hat{L}_S^{\mathrm{CE}}, \hat{L}_S^{0-1}$ Empirical risk of cross-entropy and 0-1 loss, on data sample S

$\log Z$ Log normalizing constant of a Gibbs posterior

π_β A (tempered) Gibbs posterior of form $Q^*_{\lambda\beta}$

$\widetilde{L}_S^{\mathrm{CE}}$ Bounded version of the cross-entropy loss, used for training

L^{CE}, L^{0-1} Risk functionals of cross-entropy and 0-1 loss, respectively

Q^*_λ Optimal Gibbs posterior corresponding to the PAC-Bayes λ bound (Theorem. 2.2.2)

Chapter 1

Introduction

1.1 Introduction

The empirical success of deep learning has challenged classical statistical learning theory, and turned research focus towards generalization [Zhang et al., 2017]. PAC-Bayes offers a framework for studying generalization, using the tools of statistical learning theory.

PAC-Bayes allows us to relate the empirical performance of a model to future expected performance in terms of a high-probability bound. The main challenge is to produce bounds that are nonvacuous even in complex settings such as neural networks. The current best method for obtaining nonvacuous bounds is to train a Bayesian Neural Network (BNN) with an objective derived from a PAC-Bayes bound. However, the crucial limitation is that the current bounds are only tight for relatively simple datasets (e.g. MNIST, CIFAR-10) and they require some part of the data to be used in forming the prior. An explanation for part of the looseness could lie in the approximations utilized by current methods. Most prominently, the PAC-Bayes posterior is approximated by a mean-field Gaussian [Dziugaite and Roy, 2017, Perez-Ortiz et al., 2020]. This variational approximation is known to not work well in general in Bayesian Neural Networks [Foong et al., 2020], and hence may result in loose approximations of the optimal value of the bound.

Our work is interested in measuring how tight specific PAC-Bayes bounds can be, compared to those obtained by the mean-field approximation. For many bounds, there exists a (posterior) probability measure that minimizes the bound. The bound can be estimated in this measure to obtain an estimate of the optimal, that is, the lowest possible value of the bound. This problem is interesting since it allows us to simultaneously test the power of mean-field approximations and the theoretical limit of specific PAC-Bayes bounds.

The mentioned optimal probability measure is a Gibbs distribution, hence only asymptotic sampling is possible. We use Hamiltonian Monte Carlo (HMC), a Markov chain Monte Carlo

(MCMC) technique to approximately sample from this distribution and then estimate the bound. The key steps apart from sampling are estimating the normalizing constant of our Gibbs posterior and ensuring that the resulting estimate holds with high probability.

Uniting PAC-Bayes and MCMC is a challenging task due to the inherently different nature of the two fields. In PAC-Bayes, we care about *certainty*, and the closest to certainty we can achieve in statistics is claims that hold with high probability. MCMC, on the other hand, relies on approximations. Although asymptotic convergence theorems exist for MCMC algorithms, their conditions are difficult, if not practically impossible to establish. However, we will see that there are claims we can make under reasonable assumptions on our MCMC samples, which expose some of the limitations of variational approximations and allow us to obtain tighter PAC-Bayes bounds in the setups we consider.

1.2 Thesis Contributions

The contributions of our thesis are as follows:

- A thorough **review of the foundations of the key areas used in this thesis**: statistical learning theory, PAC-Bayes, and Markov chain Monte Carlo methods.
- The development of a **method to estimate optimal PAC-Bayes bounds** that brings together tools from PAC-Bayes, Markov chain Monte Carlo and probability theory.
- The application of our method to **calculate optimal PAC-Bayes bounds** for various datasets.
- A **review of existing approaches** related to each aspect of our method uniting separate areas of literature.

1.3 Thesis Outline

Chapter 2 establishes the background knowledge needed for the rest of this thesis. We begin by motivating our topic from the angle of generalization. Then we discuss the foundations of statistical learning theory, PAC-Bayes and PAC-Bayes bound optimization. Lastly, we introduce Hamiltonian Monte Carlo and discuss tools to test the convergence of Markov chain Monte Carlo methods.

Chapter 3 discusses our method to calculate PAC-Bayes bounds for optimal Gibbs posteriors. After setting up the task, we provide a theoretical analysis of the gap between optimal

PAC-Bayes bounds and their variational approximations. Then we describe our method in the following three steps: (i) sampling from the Gibbs posterior (Section 3.4), (ii) estimating the KL divergence from the prior (Section 3.5) and (iii) providing a high-probability upper bound on our estimates (Section 3.6).

Chapter 4 describes our experiments to calculate optimal PAC-Bayes bounds and their results. We begin with a toy example that visually illustrates the effect of several experimental choices (Section 4.1) and then move on to test the effect of the key hyperparameters in our experiments (Section 4.3). In Section 4.4, we test our obtained HMC samples using a range of convergence diagnostics. Finally, Section 4.5 presents our estimates for optimal PAC-Bayes bounds.

Chapter 5 discusses works related to each aspect of our method. We present these works in unifying notation and discuss their connections to our results.

Chapter 2

Background

2.1 Motivation

The underlying motivation of this work is the problem of generalization in neural networks. In this section, we describe and motivate our research topic from the angle of generalization.

Why care about generalization?

Much of modern machine learning focuses on deep learning, and much of the theory of machine learning aims to explain the success of deep learning. Generalization has become the key question in understanding deep learning [Zhang et al., 2017].

A natural approach to studying generalization is to find measures that allow us to predict, or, put otherwise, explain generalization. Past research has considered optimization-based measures, such as gradient noise [Jiang et al., 2020, Roberts, 2021, Smith et al., 2021], the flatness of the loss surface [Jiang et al., 2020, Foret et al., 2021, Ju et al., 2022], and in this context its information geometry [Kim et al., 2022, Jang et al., 2022]. However, no definitive answer was found. What is clear, is that a plausible answer needs to combine many fields including optimization, geometry, and information theory. We will see that PAC-Bayes tackles this question from the angle of statistical learning theory and optimization.

PAC-Bayes as a way to understand generalization. PAC-Bayes theory uses the toolkit of statistical learning theory to explain generalization. It provides probabilistic guarantees on the future performance of a predictor based on its current performance.

What is a PAC-Bayes bound?

When designing a model to predict a certain quantity based on some (train) data, our goal is to perform well on any test dataset that comes from the same distribution as the train data. In other words, we wish to keep the empirical error (or some other measure of performance) of the predictor low on any test dataset. A way to achieve this is to ensure that the expected value of the error under the data-generating distribution is low, since

$$\boxed{\text{Expected error of a predictor}} \approx \boxed{\text{Empirical error of predictor on any test dataset}}$$

PAC-Bayes bounds upper bound the expected error[1] of a so-called *randomized predictor* (defined formally in Section 2.2.1). This means that we care about models that are parametrized by a distribution over their parameters, rather than a single parameter vector. This allows us to capture the complexity of a model better. A typical PAC-Bayes bound has the following form.

$$\boxed{\text{Expected error}} \leq \boxed{\text{Empirical error}} + \boxed{\text{Complexity of the predictor}}$$

One can interpret this bound as follows. The "expected error" depends on the error observed on the train dataset and the extent to which the predictor has overfitted to the train dataset. The latter is expressed by the "complexity" of the predictor. The bound guarantees that a model of low complexity that also has low train error will generalize well. Since we work with randomized predictors, an appropriate notion of complexity is the KL divergence (defined in Section 2.2.1) between the distribution parametrizing the randomized predictor, and some reference prior distribution P. This can be interpreted as the description length of the model, via the *bits-back argument* [Hinton and van Camp, 1993].

[1] In the case of randomized predictors, the expectation is taken both over the data generating distribution and the distribution over the parameters.

The aim of PAC-Bayes. Studying these bounds has three main motivations:

1. **Explaining generalization.**
   Tight bounds can hint at the ingredients of generalization.

2. **Supply loss functions for optimization algorithms.**
   A PAC-Bayes bound can be directly minimized (in the posterior) to yield stochastic neural networks with automatic performance guarantees (a "risk certificate").

3. **Self-certified learning.**
   A tight bound can supply performance guarantees on the test error from the training data only, thus there is no need to withhold some of the data as test data.

The key step in reaching the above goals is to make the bounds *tight*, that is, the empirical value of the bound should be close to the "expected error". We can empirically measure tightness by comparing the value of the bound to the test error. A tight bound can be interpreted as an "explanation of generalization" in the following sense. Any PAC-Bayes bound gives *sufficient conditions* for generalization: the terms of the upper bound should each be small to guarantee a tight risk certificate. A tight bound can also be informally interpreted as a *necessary condition*, i.e. it is unlikely that other terms play a role since the bound is tight. Our work studies how tight can specific PAC-Bayes bounds be, and thus fits into the line of research on understanding generalization.

Chapter Roadmap

This chapter contains the background needed for our solution. Section 2.2 introduces the concepts from statistical learning theory, including PAC-Bayes. Section 2.3 provides a background on Hamiltonian Monte Carlo (HMC) and techniques to test MCMC convergence.

2.2 Statistical Learning Theory

This section introduces basic concepts from statistical learning theory. In Section 2.2.1, we introduce our statistical framework. PAC-Bayesian theory was originally introduced to provide bounds for Bayesian learning algorithms. It was subsequently extended to a wider range of algorithms termed as *generalised* Bayesian learning. Section 2.2.2 summarizes and compares Bayesian learning and generalised Bayesian learning. Finally, Sections 2.2.3, 2.2.4, 2.2.5 and 2.2.5 introduce PAC-Bayes theory and bound estimation.

2.2.1 Setup

This section largely follows Dziugaite [2018] in the order of introducing the relevant concepts. Let $\mathcal{Z}$ be a measurable space and let $\mathcal{M}_1(\mathcal{Z})$ denote the set of probability measures on $\mathcal{Z}$. Let D be an unknown distribution $D \in \mathcal{M}_1(\mathcal{Z})$ A learning algorithm receives a set of n samples $S = (z_1, ..., z_n)$, and we assume that $S \sim D^n$ i.i.d. Throughout this thesis, we consider supervised learning. There, $\mathcal{Z}$ takes the form $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} \subset \mathbb{R}^d$ is the space of data samples, and $\mathcal{Y} \subset \mathbb{R}$ is the space of labels. Further, we fix a weight space $\mathcal{W} \subseteq \mathbb{R}^p$ containing all possible weights. Each weight vector $\mathbf{w}$ maps to a *predictor function* $h_{\mathbf{w}} : \mathcal{X} \to \mathcal{Y}$ that assigns a label $y \in \mathcal{Y}$ to any input $x \in \mathcal{X}$. We seek to find a predictor function that minimizes the *risk* (expected loss)

$$L(\mathbf{w}) = \mathbb{E}_{z \sim D}\left[l(\mathbf{w}, z)\right], \tag{2.1}$$

where $l : \mathcal{W} \times \mathcal{Z} \to [0, \infty)$ is a measurable loss function. Since the data-generating distribution is unknown, $L(\mathbf{w})$ is not observable. Hence in practice, we compute the *empirical risk* functional, which depends on our sample set S and is defined as

$$\hat{L}_S(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} l(\mathbf{w}, z_i).$$

Randomized predictors

PAC-Bayes bounds will be stated for *randomized predictors*. Given a data sample x, a randomized predictor makes a prediction at its label using a random sample of weights $\mathbf{w}$, drawn from a distribution $Q \in \mathcal{M}_1(\mathcal{W})$. We may identify the randomized predictor with its distribution Q. The risk of a randomized predictor Q is defined via averaging

$$L(Q) = \mathbb{E}_{\mathbf{w} \sim Q}[\mathbb{E}_{z \sim D}\left[l(\mathbf{w}, z)\right]]. \tag{2.2}$$

The empirical risk of the randomized predictor is given by

$$\hat{L}_S(Q) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\mathbf{w}\sim Q}\left[l(\mathbf{w}, z_i)\right]]. \tag{2.3}$$

Loss functions

In PAC-Bayes, we measure model performance in terms of loss functions. In this thesis, we consider classification tasks only. There, the label set $\mathcal{Y}$ is discrete. Let $\mathcal{Y} = K$ with $|K| = k$. The natural loss function is the *0-1 loss*, defined as

$$l^{0-1}(\mathbf{w}, z) = \mathbb{I}(y \neq h_{\mathbf{w}}(x)). \tag{2.4}$$

As this function is not differentiable everywhere, we will use as a surrogate the *cross-entropy loss*. To define this, we note that $h_{\mathbf{w}}(x)$ determines a probability distribution $(p_1, ..., p_k)$ on K.

$$l^{\mathrm{CE}}(\mathbf{w}, y) = -\sum_{i=1}^{k} y_i \log(p_i) \tag{2.5}$$

The cross-entropy loss is bounded from below and unbounded from above. Depending on which loss function is used, we will use notations $L^{\mathrm{CE}}(Q), \hat{L}_S^{\mathrm{CE}}(Q), L^{0-1}(Q)$ and $\hat{L}_S^{0-1}(Q)$ for our risk functionals. We will omit the superscrips when discussing risk functionals in general.

KL divergence

The notion of KL divergence will be used in the "complexity term" of PAC-Bayes bounds. Given two probability distributions $Q, P \in \mathcal{M}_1(W)$, we define the *Kullback-Leibler (KL) divergence* of Q from P as

$$D_{KL}(Q||P) = \int_W \log\left(\frac{dQ}{dP}\right) dQ,$$

whenever $\frac{dQ}{dP}$, the Radon-Nikodym derivative of Q with respect to P is defined, and $D_{KL}(Q||P) = \infty$ otherwise. We will also make use of the *Bernoulli KL divergence*, i.e. for $Q = \mathrm{Bern}(q)$ and $P = \mathrm{Bern}(p)$, $q, p \in [0, 1]$,

$$D_{KL}(Q||P) := \mathrm{kl}(q||p) = q\log\left(\frac{q}{p}\right) + (1-q)\log\left(\frac{1-q}{1-p}\right).$$

2.2.2 Bayesian and Generalised Bayesian Learning

PAC-Bayes theory applies to so-called *generalized Bayesian learning algorithms*. In order to introduce these, we now recall the setup of Bayesian learning and inference, in order to contrast it to generalized Bayesian learning.

Bayesian inference is a principled way of incorporating prior beliefs into probabilistic inference. Given the data, we construct a probability model $\{P_{\mathbf{w}} : \mathbf{w} \in \mathcal{W}\}$, where the data generating process is governed by a set of parameters (weights) $\mathbf{w} \in \mathcal{W}$. The model can be written as $p(z|\mathbf{w})$ and, when viewed as a function of $\mathbf{w}$, is termed the *likelihood*. We fix a prior measure $P \in \mathcal{M}_1(W)$ with density $p(\mathbf{w})$ that represents our knowledge of the true parameter in $\mathcal{W}$. Then the *Bayesian posterior measure Q* has density

$$p(\mathbf{w}|S) = \frac{p(\mathbf{w}, S)}{\int p(\mathbf{w}, S) \mathrm{d}\mathbf{w}}.$$

We will refer to this simply as the posterior. The core to Bayesian inference/learning lies in the interpretation of the above constituents: the prior captures knowledge about the parameter, and the likelihood describes our probabilistic model. The posterior then gives the distribution of the parameters given the data, under the assumption that our model is correct.

However, in machine learning, the emphasis is on *prediction*, rather than on inference. We seek to build models that generalise well, that is, give accurate predictions on unseen (but in-distribution) data, but do not care about the parameters generating those predictions. Generalised Bayesian learning was developed to turn the focus to prediction [Bissiri et al., 2016]. Most approaches relax the constraints on the interpretation of the likelihood and the prior. The likelihood is allowed to be any, perhaps model-agnostic measure of performance. We will refer to generalised negative log likelihoods as *loss functions*. Further, the prior is no longer restricted to incorporate prior knowledge but is viewed as a way to guide predictions.

2.2.3 PAC-Bayes Bounds

PAC-Bayes theory was first introduced in McAllester [1998], as a combination of PAC and Bayesian correctness theorems. In the PAC-learning framework [Valiant, 1984], the learner receives some data samples, and its goal is to select a hypothesis such that with high probability ("probably"), the selected function will have low generalization error ("approximately correct"). PAC-Bayes is the combination of this framework with Bayesian learning, where we are constrained to experimental settings consistent with a pre-specified prior.

We distinguish two types of PAC-Bayes inequalities: empirical and oracle inequalities, and consequently, empirical and oracle PAC-Bayes bounds [Alquier, 2023]. In empirical bounds, the right-hand side is directly computable. Numerically evaluating it for a given dataset and model (hypothesis), we can obtain a probabilistic guarantee (a *risk certificate*) for the future performance of the model. On the other hand, oracle bounds are purely theoretical objects that allow us to reason about estimators in general. In this thesis, we will exclusively focus on empirical bounds as our goal is to give numerical generalization guarantees for neural networks. We now state two common empirical PAC-Bayes bounds.

Some PAC-Bayes bounds

The *PAC-Bayes relative entropy bound* is given in the following theorem, originally due to Langford and Seeger [2001] and improved by Maurer [2004].

Theorem 2.2.1. *(PAC-Bayes relative entropy bound) Fix a triple $(\mathcal{W}, \mathcal{Z}, l)$ to stand for a weight space $\mathcal{W} \subset \mathbb{R}^p$, an example space $\mathcal{Z}$ and a loss function $l : \mathcal{W} \times \mathcal{Z} \to [0,1]$. For any randomized predictor identified with any distribution $Q \in \mathcal{M}_1(\mathcal{W})$, let $L(Q)$ and $\hat{L}_S(Q)$ denote the risk and empirical risk on some set S of n i.i.d. data samples, respectively. Fix arbitrary $\delta \in (0,1)$. Then, for any data-free distribution P over $\mathcal{W}$, simultaneously for all distributions Q over $\mathcal{W}$, we have*

$$\mathbb{P}\left(\mathrm{kl}(\hat{L}_S(Q)||L(Q)) \leq \frac{D_{KL}(Q||P) + \log\frac{2\sqrt{n}}{\delta}}{n}\right) > 1 - \delta. \tag{2.6}$$

Proof. Please find the proof in Maurer [2004] (Theorem 5). □

Note on the proofs of PAC-Bayes bounds. PAC-Bayes bounds are derived from concentration inequalities. The most common tools are Hoeffding's inequality and the Chernoff bound [Alquier, 2023].

To compute the bound in Equation 2.6, inversion of the Bernoulli kl is required, hence this bound cannot directly be used for optimization. Various relaxations have been proposed. Of particular interest are linear PAC-Bayes bounds. An example is below.

Theorem 2.2.2. *(PAC-Bayes λ bound) In the same setup as in Theorem 2.2.1, for any $\delta \in (0,1)$ with probability of at least $1-\delta$ over S, simultaneously for all $Q \in \mathcal{W}$ and $\lambda \in (0,2)$, we have*

$$L(Q) \leq \frac{\hat{L}_S(Q)}{1-\frac{\lambda}{2}} + \frac{D_{KL}(Q||P) + \log(\frac{2\sqrt{n}}{\delta})}{n\lambda(1-\frac{\lambda}{2})}. \tag{2.7}$$

Proof. (Sketch) This bound is derived from Equation 2.6 using a refined version of Pinsker's inequality $\mathrm{kl}(q||p) \geq (p-q)^2/(2p)$, which holds for all $q < p$, and the inequality $\sqrt{ab} \leq \frac{1}{2}(\lambda a + \frac{b}{\lambda})$, valid for all $\lambda > 0$. For a full proof, please refer to Thiemann et al. [2017]. □

This bound is special in that it holds *simultaneously* for all $\lambda \in (0,2)$. For many bounds which include a free parameter λ, this is not the case, and hence grid search over λ requires adding a penalty term to the bound for each value checked. For details, see Theorem 2.4 in Alquier [2023]. The PAC-Bayes λ bound can be optimized in both Q and λ in an alternated fashion, without incurring a penalty term.

As stated in the theorems, PAC-Bayes bounds are given in terms of an arbitrary, but fixed prior probability measure P. The choice of P affects the tightness of the bound. We now explain the considerations behind choosing the PAC-Bayes prior.

2.2.4 The PAC-Bayes Prior

The choice of P affects the tightness of the bound most strongly through the "complexity term" $D_{\mathrm{KL}}(Q||P)$. This limits how tight the bound can be made, hence we aim to find a prior we can reasonably believe to be close to well-performing posteriors, in the KL sense. The typical choice of prior is a mean-field Gaussian [Dziugaite and Roy, 2017], but Laplace distributions have also been used in the literature [Perez-Ortiz et al., 2020].

In Bayesian inference and learning, the prior is chosen independently of the data. However, in many cases a pre-chosen prior may be far from any well-performing posterior, causing bounds to be loose or even vacuous. As a consequence, the PAC-Bayes community developed data-dependent priors. In this thesis, our focus will be on data-independent priors, because these lead to loose bounds even for simple datasets. However, we briefly discuss data-dependent priors too, as they are a key part of modern PAC-Bayes theory.

Data-independent priors

Data-independent priors are chosen before observing the data. In the typical case of a mean-field Gaussian prior, the mean and variance parameters need to be selected. For the former, one may be tempted to initialize the mean at zero, since neural network weights are typically near zero. However, in Dziugaite and Roy [2017], the authors give an argument for favoring a random mean initialization. They observe that a zero initialization of the prior mean preserves the network's symmetries. They prove that in this case, it is possible to construct a posterior that incorporates the symmetries and is closer to the prior in KL divergence. Since computing such a posterior for an exponential number of symmetries is intractable, the alternative is to *break* the symmetries by sampling the mean randomly. In order to maintain a plausible magnitude, the mean initialization is drawn from a truncated Gaussian with zero mean and small variance.

The choice of the prior variance is based on two main, contradictory considerations. On the one hand, when set large, the prior influences the risk less, hence in general, better-performing (in terms of the risk) PAC-Bayes posteriors can be found. On the other hand, large variance means that when computing the empirical risk $\hat{L}_S(Q) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{\mathbf{w}\sim Q}[l(\mathbf{w}, z_i)]$, we are sampling in a large radius of the posterior mean, causing many samples to have large loss.

Data-dependent priors

Data-dependent priors have come into focus when they were used to compute the first non-vacuous PAC-Bayes bounds on MNIST neural network classifiers [Dziugaite and Roy, 2017]. The main difficulty is that a PAC-Bayes bound is not valid on data that has been used to build the prior. Many workarounds were proposed, see [Dziugaite and Roy, 2017, 2018, Perez-Ortiz et al., 2020]. In the most common case, stochastic gradient descent (SGD) is used to learn the prior mean on a separated subset of the train dataset.

Data-dependent priors can result in remarkably tight bounds, such as 1.4-1.5% on MNIST [Perez-Ortiz et al., 2020, Lotfi et al., 2022]. However, Lotfi et al. [2022] argues that data-dependent PAC-Bayes bounds do not explain generalization any further than the prior alone, since they separate out the process of learning the prior. Lotfi et al. [2022] demonstrates that using elementary (i.e. non-PAC-Bayes) bounds on a data-dependent prior can result in performance competitive with that of sophisticated PAC-Bayes bounds. Hence data-independent bounds are more informative for understanding generalization.

Having discussed the choice of the prior, we now turn to the methods available for optimizing a PAC-Bayes bound.

2.2.5 Optimizing PAC-Bayes Bounds

Given our prior and model architecture, we wish to select the PAC-Bayes posterior in a way that the resulting bound is small. Depending on the form of the bound, there may exist an exact minimizer. We will use (asymptotically exact) samples from this exact minimizer. The competitor method is to find an approximation to the exact minimizer using variational inference. We now explain both methods.

Exact minimisation: Gibbs posteriors

For linear PAC-Bayes bounds, the theoretical minimiser is a Gibbs distribution, also called Gibbs (generalised) posterior. We state the precise form of the Gibbs posterior corresponding to the PAC-Bayes λ bound.

Proposition 2.2.1. *The minimizer of the PAC-Bayes λ bound over $\mathcal{M}_1(\mathcal{W})$ (the unconstrained set of probability measures over $\mathcal{W}$) is a Gibbs distribution over the weights* $\mathbf{w}$ *with Radon-Nikodym derivative with respect to a fixed prior P given by*

$$\frac{dQ^*_\lambda}{dP}(\mathbf{w}) = \frac{e^{-n\lambda \hat{L}_S(\mathbf{w})}}{\mathbb{E}_{\mathbf{w}\sim P}\left[e^{-n\lambda \hat{L}_S(\mathbf{w})}\right]}. \tag{2.8}$$

Proof. Please refer to Corollary 2.3 in Alquier et al. [2016]. □

We will make use of the unnormalized Gibbs posterior $\frac{d\tilde{Q}^*_\lambda}{dP}(\mathbf{w}) = e^{-n\lambda \hat{L}_S(\mathbf{w})}$. The density of the Gibbs posterior measure is given by

$$q^*_\lambda(\mathbf{w}|z) = \frac{e^{-n\lambda \hat{L}_S(\mathbf{w})}p(\mathbf{w})}{\mathbb{E}_{\mathbf{w}\sim P}\left[e^{-n\lambda \hat{L}_S(\mathbf{w})}\right]}. \tag{2.9}$$

One way to estimate the optimal PAC-Bayes bound given a prior is to evaluate it in the optimal Gibbs posterior Q^*_λ. However, the Gibbs posterior is intractable in practice, only asymptotic sampling methods are available. This thesis focuses on using Hamiltonian Monte Carlo (HMC) to simulate from the Gibbs posterior given in Equation 2.8. The basics of this MCMC technique will be presented in Section 2.3.

Non-exact minimisation: Variational Inference

The most common option for non-exact minimisation of PAC-Bayes bounds is variational inference. In the case of the PAC-Bayes λ bound, variational inference aims to find the best approximation to the Gibbs distribution given in Equation 2.8, among a parametrised family $\mathcal{F}_\theta$. The most common choice is to take $\mathcal{F}_\theta$ to be the set of mean-field Gaussian distributions with parameter $\theta = (\mu, \sigma^2 \mathrm{I})$. Classical variational inference minimizes the negative ELBO. However, a more direct way to obtain tight bounds is to minimise the PAC-Bayes bound as a training objective. Since the prior is also a (mean-field) Gaussian, the KL-divergence is available analytically, and the empirical risk $\hat{L}_S(Q)$ can be approximated using a Monte Carlo average over exact samples from the Gaussian approximate posterior Q. In what follows, we will refer to this technique as mean-field variational inference (MFVI).

A perhaps unusual aspect of this method that is nevertheless consistently used in the literature is the use of the stochastic gradient descent (SGD) optimizer. This is motivated by the fact that SGD tends to converge to flat minima [Keskar et al., 2017]. For a mean μ around a flat (local) minimum $\mathbf{w}^*$, we have the property that $\mathbb{E}_{\mathbf{w}\sim\mathcal{N}(\mu,s^2)}[L(\mathbf{w})] \approx L(\mathbf{w}^*)$ even for large values of the variance s^2 [Alquier, 2023]. This means that we are allowed a more spread-out posterior while still retaining low loss and thereby keeping term one of PAC-Bayes bounds low.

Technicalities of the optimisation and risk certificate estimation

We now explain all details of the optimisation process. Our discussion closely follows the method outlined in Perez-Ortiz et al. [2020]. The key technical steps are (i) making the loss bounded and (ii) constructing a risk certificate from posterior samples.

PAC-Bayes bounds assume a loss function bounded in $[0,1]$. However, the cross-entropy loss (defined in Equation 2.4) is unbounded from above. Hence we transform $l^{\mathrm{CE}}(\mathbf{w},y) = \sum_{i=1}^{k} y_i \log(p_i)$ as follows.

$$\widetilde{l^{\mathrm{CE}}}(\mathbf{w},y) := \frac{-\sum_{i=1}^{k} y_i \log(\max(p_i, p_{\min}))}{-\log(p_{\min})}, \tag{2.10}$$

with some $p_{\min} > 0$, which now falls into $[0,1]$. We will take $p_{\min} = 10^{-4}$. The corresponding risk functionals will be denoted by $\widetilde{L}^{\mathrm{CE}}(Q)$ and $\widetilde{L}_S^{\mathrm{CE}}(Q)$.

After arriving at a posterior distribution Q either by sampling from the Gibbs posterior or MFVI, we wish to compute a risk certificate on the *error* (0-1 loss) $L^{0-1}(Q)$ of the stochastic predictor given by Q. The transformed cross-entropy loss is only used for sampling. To compute the risk certificate, we use the PAC-Bayes relative entropy bound (Equation 2.6), since it is the tightest. To invert the Bernoulli KL, we define

$$\mathrm{kl}^{-1}(x,b) := \sup\{y \in [x,1] : \mathrm{kl}(x||y) \leq b\}. \tag{2.11}$$

This can be seen as a proper definition of the inverse Bernoulli KL. Our RC is then

$$L^{0-1}(Q) \leq \mathrm{kl}^{-1}\left(\hat{L}_S^{0-1}(Q), \frac{D_{\mathrm{KL}}(Q||P) + \log\frac{2\sqrt{n}}{\delta}}{n}\right). \tag{2.12}$$

Finally, the empirical risk $\hat{L}_S^{0-1}(Q)$ can be approximated with Monte-Carlo average (from either the Gibbs posterior or the Gaussian approximate posterior). Here, we seek more than an unbiased estimator. In order to produce a valid PAC-Bayes bound estimate, we need an *upper bound with high probability*. One may be tempted to use the Central Limit Theorem (CLT) to give a confidence interval. However, as we have a finite sample, the obtained confidence intervals will only be asymptotic. A non-asymptotic bound is supplied by the following Theorem from Langford and Caruana [2001].

Theorem 2.2.3. *Suppose $W_1, W_2, \dots, W_m \sim Q$ are i.i.d., and $\hat{Q}_m = \sum_{j=1}^{m} \delta_{W_j}$ is their empirical distribution. Then for any $\delta' \in (0,1)$ with probability $1-\delta'$*

$$\hat{L}_S(Q) \leq \mathrm{kl}^{-1}\left(\hat{L}_S(\hat{Q}_m), \frac{1}{m}\log(\frac{2}{\delta'})\right). \tag{2.13}$$

Due to a union bound argument, using this bound on $\hat{L}_S^{0-1}(Q)$ will modify the resulting bound to hold with probability $1-\delta-\delta'$ only. This completes all the required steps to compute the bound.

Next, we briefly review the Markov Chain Monte Carlo (MCMC) methods available to sample from distributions with properties similar to the Gibbs posterior. Our special focus will be on Hamiltonian Monte Carlo (HMC).

2.3 Markov Chain Monte Carlo

Markov chain Monte Carlo (MCMC) stands for a class of algorithms for asymptotically exact sampling from probability distributions. They construct a Markov chain that has the desired target distribution as its equilibrium distribution, hence one can (asymptotically) obtain a sample of the desired distribution by recording states from the chain. We wish to use MCMC methods to sample from a Gibbs distribution. Since the normalizing constant of the Gibbs distribution is intractable, our method needs to apply to unnormalized densities. Hamiltonian Monte Carlo is such a method.

2.3.1 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo (HMC, [Duane et al., 1987]) is the "gold standard" for approximate sampling from complicated distributions such as the (Gibbs) posterior of a neural network [Izmailov et al., 2021]. It is superior to other MCMC algorithms due to its ability to take into account the geometry of the target distribution through the use of gradients and thereby generate better step proposals. This motivates our use of HMC for sampling from the Gibbs posterior that minimizes the PAC-Bayes λ bound.

HMC originally appeared under the name "Hybrid Monte Carlo" as a unification of MCMC and molecular dynamics methods [Duane et al., 1987]. Hamiltonian Monte Carlo uses the Metropolis–Hastings algorithm, together with a discretized Hamiltonian dynamics evolution, which proposes a move to a new point in the state space. The Hamiltonian dynamics evolution is simulated by a leapfrog integrator that updates the parameters and the momentum of the process. The HMC algorithm is given in Algorithm 1, and the leapfrog step supplied separately in Algorithm 2 [Izmailov et al., 2021].

Hamiltonian dynamics operates on a p-dimensional position vector $\mathbf{w}$ and a p-dimensional momentum vector, $\mathbf{m}$, and is described by the Hamiltonian, $H(\mathbf{w},\mathbf{m})$. A commonly used Hamiltonian function is $H(\mathbf{w},\mathbf{m}) := U(\mathbf{w}) + K(\mathbf{m})$, the sum of the *potential energy* and the *kinetic energy*, respectively. The unnormalized target density, $\tilde{q}(\mathbf{w})$ satisfies $U(\mathbf{w}) = -\log\tilde{q}(\mathbf{w})$. The change of $\mathbf{w}$ and $\mathbf{m}$ in time is described by Hamilton's equations

$$\frac{\mathrm{d}\mathbf{w}_i}{\mathrm{d}t} = \frac{\partial H}{\partial \mathbf{m}_i}, \quad \frac{\mathrm{d}\mathbf{m}_i}{\mathrm{d}t} = -\frac{\partial H}{\partial \mathbf{w}_i}. \tag{2.14}$$

The three key properties of Hamiltonian dynamics are leveraged in constructing a Markov chain updates with Metropolis-Hastings. These properties are (i) time reversibility, (ii) conservation of the Hamiltonian, and (iii) volume preservation.

The proposed state is accepted as the next state of the Markov chain with probability p_{accept}, given in Algorithm 1. If the proposed state is not accepted, the next state is the same as the current state. When estimating statistics from the Markov chain, all states are used, hence those in place a non-accepted proposal are counted multiple times. For more detailed introductions, see Betancourt [2017] and Brooks et al. [2011a].

We note that HMC is connected to Gibbs distributions[2] in the following sense. HMC formulates the target density $\tilde{q}(\mathbf{w})$ as a certain Gibbs distribution. The kinetic energy $K(\mathbf{m})$ is usually taken to be a bilinear form $K(\mathbf{m}) = \frac{1}{2}\mathbf{m}^\top\hat{M}^{-1}\mathbf{m}$, and HMC constructs a Gibbs distribution of form

$$e^{-\beta(U(\mathbf{w})+K(\mathbf{m}))} = \tilde{q}(\mathbf{w})^\beta \cdot e^{-\beta\frac{1}{2}\mathbf{m}^\top\hat{M}^{-1}\mathbf{m}}. \tag{2.15}$$

In this work, we will use HMC in order to sample from the Gibbs posterior corresponding to the PAC-Bayes λ bound on neural networks.

[2] Note that Gibbs distributions are called Boltzmann distributions in the physics literature.

Algorithm 1 Hamiltonian Monte Carlo

```
Input: Trajectory length $\tau$, number of burn-in iterations $N_{\text{burnin}}$, initial parameters $\mathbf{w}_{\text{init}}$, step size $\Delta$, number of samples $K$, unnormalized posterior log density function $f(\mathbf{w}) = \log p(z|\mathbf{w}) + \log p(\mathbf{w})$.
Output: Set $S$ of samples $\mathbf{w}$ of the parameters.
$\mathbf{w} \leftarrow \mathbf{w}_{\text{init}}$; $N_{\text{leapfrog}} \leftarrow \frac{\tau}{\Delta}$;
# Burn-in stage
for $i \leftarrow 1...N_{\text{burnin}}$ do
    $\mathbf{m} \sim \mathcal{N}(0,\mathrm{I})$;
    $(\mathbf{w},\mathbf{m}) \leftarrow \text{Leapfrog}(\mathbf{w},\mathbf{m},\Delta,N_{\text{leapfrog}},f)$;
end for
# Sampling
$S \leftarrow \emptyset$;
for $i \leftarrow 1...K$ do
    $\mathbf{m} \sim \mathcal{N}(0,\mathrm{I})$;
    $(\mathbf{w},\mathbf{m}) \leftarrow \text{Leapfrog}(\mathbf{w},\mathbf{m},\Delta,N_{\text{leapfrog}},f)$;
    # Metropolis-Hastings correction
    $p_{\text{accept}} \leftarrow \min\left\{1, \frac{f(\mathbf{w}')}{f(\mathbf{w})} \cdot \exp(\frac{1}{2}||\mathbf{m}||^2 - ||\mathbf{m}'||^2)\right\}$ ;
    $u \sim \text{Uniform}[0,1]$;
    if $u \leq p_{\text{accept}}$ then
        $\mathbf{w} \leftarrow \mathbf{w}'$;
    end if
    $S \leftarrow S \cup \{\mathbf{w}\}$;
end for
```

Algorithm 2 Leapfrog Integration

```
Input: Parameters $\mathbf{w}_0$, initial momentum $\mathbf{m}_0$, step size $\Delta$, number of leapfrog steps $N_{\text{leapfrog}}$, posterior log density function $f(\mathbf{w}) = \log p(z|\mathbf{w})$
Output: New parameters $\mathbf{w}$; new momentum $\mathbf{m}$.
$\mathbf{w} \leftarrow \mathbf{w}_0$; $\mathbf{m} \leftarrow \mathbf{m}_0$;
for $i \leftarrow 1...N_{\text{leapfrog}}$ do
    $\mathbf{m} \leftarrow \mathbf{m} + \frac{\Delta}{2} \cdot \nabla f(\mathbf{w})$;
    $\mathbf{w} \leftarrow \mathbf{w} + \Delta \cdot \mathbf{m} f(\mathbf{w})$;
    $\mathbf{m} \leftarrow \mathbf{m} + \frac{\Delta}{2} \nabla f(\mathbf{w})$;
end for
$\text{Leapfrog}(\mathbf{w}_0,\mathbf{m}_0,\Delta,N_{\text{leapfrog}},f) \leftarrow (\mathbf{w},\mathbf{m})$
```

2.3.2 MCMC Convergence and Diagnostics

Having introduced HMC, this section discusses statistics commonly used to evaluate the quality of samples produced by MCMC algorithms, including HMC. This step is crucial for us in ensuring that the bounds we calculate using HMC samples are valid. Although it is not possible to prove Markov chain convergence in practical settings, we will use diagnostics to verify that there are no apparent signs of problematic behavior. We now define these.

Let $(\mathbf{X}^{(t)})_{t=1}^{\infty}$ be a Markov chain on state space $\mathscr{A}$, whose initial distribution equals its stationary distribution, hence $X_1, X_2, ...$ are identically distributed. Consider a square-integrable function $f : \mathscr{A} \to \mathbb{R}$, whose expected value $\mathbb{E}_{P_{X_1}}[f(X_1)]$ we wish to estimate. Note that square integrability impies $\mathrm{Var}[f(X_1)] \leq \infty$. First, we define the Effective sample size (ESS), which quantifies the loss of information caused by correlation.

Definition 2.3.1. *(Effective Sample Size) The Effective sample size of $(\mathbf{X}^{(t)})_{t=1}^{N}$ is given by*

$$\mathrm{ESS}[f] = \frac{N}{1 + 2\sum_{\tau=1}^{\infty} \rho(\tau)}, \tag{2.16}$$

where $\rho(\tau) = \mathrm{Cor}(f(X_1), f(X_{1+\tau}))$ denotes the autocorrelation at lag τ.

The ESS is typically lower than the number of samples, indicating the presence of positive correlations among the samples. In MCMC, where states are typically positively correlated, an ESS equaling the number of samples signals uncorrelated samples.

Definition 2.3.2. *(MCMC standard error) The MCMC standard error of $(\mathbf{X}^{(t)})_{t=1}^{N}$ is*

$$\mathrm{MCMC\_SE}[f] = \sqrt{\frac{\mathrm{Var}[f(X_1)]}{\mathrm{ESS}[f]}}. \tag{2.17}$$

The MCMC SE measures the concentration of a sample mean around the true mean, in the sense of the Markov chain Central Limit theorem (MCCLT). It requires the above conditions on the Markov chain, namely (i) stationary chain and (ii) square-integrable f and states that, for large N, approximately

$$\frac{1}{N}\sum_{i=1}^{N} f(X_i) \sim \mathcal{N}(\mathbb{E}_{P_{X_1}}[f(X_1)], \mathrm{MCMC\_SE}[f]). \tag{2.18}$$

Since in practice we don't have access to $\rho(\tau)$ and $\mathrm{Var}[f(X_1)]$, we estimate them from our finite set of samples.

Since ESS requires stationarity to be well-defined, we will also use the $\hat{R}$ (known also as potential scale reduction) statistic to check convergence by comparing multiple independent

Markov chains [Gelman and Rubin, 1992]. $\hat{R}$ measures how much the variance of the means between multiple chains exceeds that of identically distributed chains. We have that $\hat{R} \geq 1$, where $\hat{R} = 1$ means perfect convergence. For more discussion on $\hat{R}$, please refer to Gelman and Rubin [1992], and for general discussion on MCMC and convergence diagnostics, see Jones and Qin [2022] and Brooks et al. [2011b].

Summary. This chapter presented the background required to discuss our method for estimating optimal PAC-Bayes bounds. We introduced basic concepts in statistical learning theory, including PAC-Bayes bounds and their optimization. We also discussed MCMC methods and convergence diagnostics, with special focus on HMC.

Chapter 3

Method

In this chapter, we present the methods used to tackle our goal: approximately sampling from the optimal Gibbs posterior in order to estimate the tightness gap between the optimal value of the bound, and estimates obtained by MFVI. We begin by a precise setup of our problem (Section 3.1). In Section 3.2, we give a simple lower bound for the tightness gap, which is, to the best of our knowledge, novel. Then, in Section 3.3 we define the specific Gibbs posterior that minimizes our target PAC-Bayes bound.

Sections 3.4, 3.5, and 3.6 contain our method for estimating PAC-Bayes bounds for the Gibbs posterior. This will require three major steps: (i) sampling from the Gibbs posterior, (ii) estimating its KL divergence from the prior P and (iii) computing high-probability upper bounds to ensure that the bound we put together holds with high probability.

Solution (i) relies on Hamiltonian Monte Carlo and is presented in Section 3.4. The core technique used for Solution (ii) is *thermodynamic integration*. We supply an introduction to the method (Section 3.5.1), and an explanation of its use in our context (Section 3.5.2). In Section 3.5.3, we present two properties of thermodynamic integration, one of which was proven by us. For step (iii), we discuss several options for high-probability bounds and compare them in terms of their assumptions (Section 3.6).

Finally, Section 3.7 summarizes the method and its underlying assumptions.

3.1 Task Setup

To begin our discussion, we first precisely define the objects of interest, namely, the tightness gap and the PAC-Bayes objective (bound) to be used. Let $C : \mathcal{M}_1(\mathcal{Z}) \to \mathbb{R}^+$, $Q \mapsto C(Q)$ denote any PAC-Bayes objective.

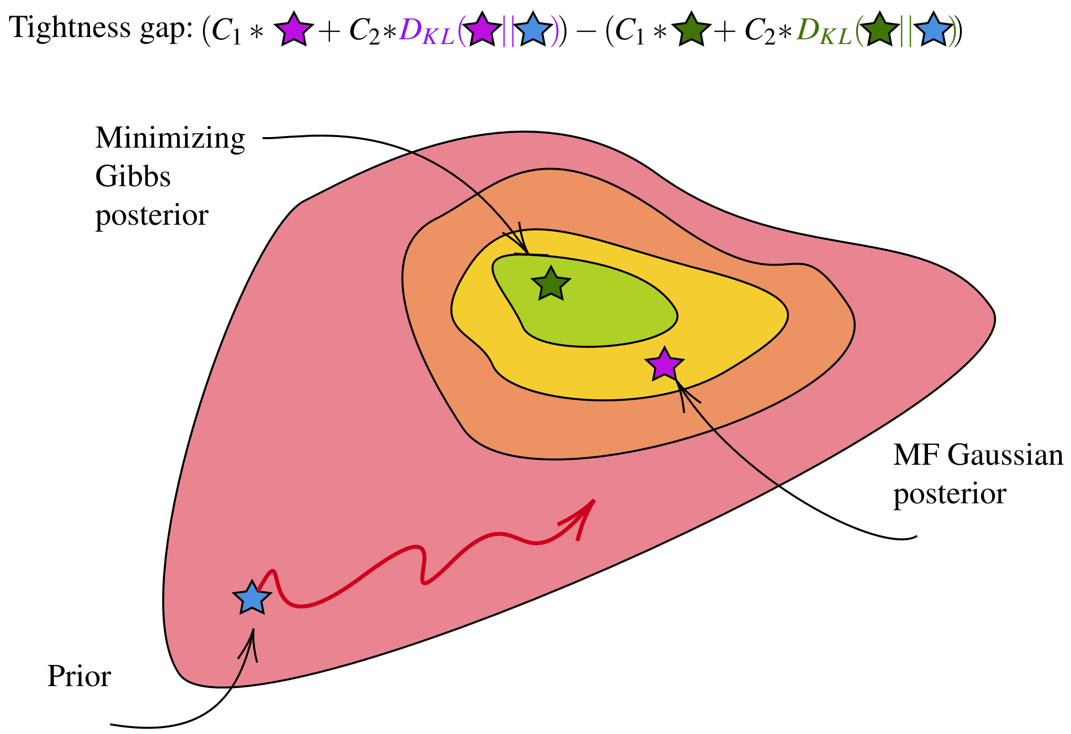

Tightness gap: $(C_1 * ★ + C_2 * D_{KL}(★||★)) - (C_1 * ★ + C_2 * D_{KL}(★||★))$
Minimizing Gibbs posterior
MF Gaussian posterior
Prior

Fig. 3.1 Illustration of the contours of the expected loss of a stochastic classifier as a function of the underlying parameter distribution. The red arrow shows a possible optimization trajectory from the prior. The tightness gap is evaluated by computing a PAC-Bayes bound in both the optimal Gibbs posterior and the optimal mean-field Gaussian posterior. The bound shown here is linear.

The tightness gap

We can define the *tightness gap* as follows.

Definition 3.1.1. *The tightness gap between the optimal value of a PAC-Bayes objective and the optimal mean-field Gaussian approximation is defined as*

$$C(G^*) - C(Q^*), \tag{3.1}$$

where $G^* = \inf_{G \text{ is a mean-field Gaussian}} C(G)$ and Q^* is the optimal Gibbs posterior corresponding to the PAC-Bayes bound C.

The tightness gap and the underlying relations between the prior, G^* and Q^* are illustrated in Figure 3.1. The figure should be interpreted informally and shows the expected loss of a stochastic classifier as a function of its underlying parameter distribution. Euclidean distance between two points of the figure should be understood as the KL divergence from the prior.

The mean-field Gaussian posterior either achieves higher expected loss, is farther from the prior than the Gibbs posterior, or both. In the following sections, we will estimate G^* with MFVI and Q^* with HMC.

Used bounds

We will use the following PAC-Bayes bounds. For sampling, we will use the PAC-Bayes λ bound (recalled below), and for evaluation, we will use both the PAC-Bayes relative entropy bound (Equation 2.6) and the PAC-Bayes λ bound. Both of these bounds were introduced in Section 2.2.3. Recall that the PAC-Bayes λ bound is defined for any (posterior) distribution Q as

$$C(Q,\lambda) := \frac{\hat{L}_S(Q)}{1-\frac{\lambda}{2}} + \frac{D_{KL}(Q\,||\,P) + \log(2\sqrt{n}/\delta)}{n\lambda(1-\frac{\lambda}{2})}. \tag{3.2}$$

The inequality $L(Q) \leq C(Q,\lambda)$ holds with probability at least $1-\delta$ uniformly over any Q and $\lambda \in (0,2)$.

The choice of using this bound for sampling was based on two related considerations. Firstly, the fact that the bound holds uniformly over $\lambda \in (0,2)$ means that we are allowed to cross-validate over λ. By controlling all hyperparameters in the bound, this bound is easier to optimize as an objective function. Secondly, in Perez-Ortiz et al. [2020], the authors find that optimizing this bound performs close-to best in most experiments. Its competitor is a bound in quadratic form. To the best of our knowledge, the minimizing Q corresponding to this bound is not known.

The PAC-Bayes relative entropy bound is tighter than the PAC-Bayes λ bound but is not convenient to use for sampling, due to needing to invert the Bernoulli kl divergence. Further, the specific Gibbs posterior given in Section 2.2, Equation 2.8 is only the minimizer of the PAC-Bayes λ bound, hence we will obtain a valid, but not the *optimal* risk certificate for the PAC-Bayes relative entropy bound. We will report risk certificates with both the PAC-Bayes relative entropy bound and the PAC-Bayes λ bound. For the latter, the Gibbs posterior gives the optimal risk certificate. We can now formulate our task as follows.

Estimate the tightness gap $C(G^*) - C(Q^*)$, where C is the PAC-Bayes relative entropy bound and the PAC-Bayes λ bound.

Before proceeding to our solution method, we first examine the object $C(G^*) - C(Q^*)$ theoretically. It is clear that $C(G^*) - C(Q^*) > 0$. However, using a result from Large Deviation Theory, we can supply a stronger lower bound for *linear* PAC-Bayes bounds. We state this lower bound next.

3.2 A Lower Bound on the Tightness Gap

Let us define a generic linear PAC-Bayes bound as

$$C(Q) := a\mathbb{E}_{\mathbf{w}\sim Q}[r(\mathbf{w})] + bD_{KL}(Q||P) + c \tag{3.3}$$

for risk function $\hat{L} : \mathcal{Z} \to \mathbb{R}$, fixed prior measure P with density $p(\mathbf{w})$ and any probability measure Q with density $q(\mathbf{w})$. The PAC-Bayes λ bound is a special case with $a = 1/(1-\frac{\lambda}{2})$, $b = 1/(n\lambda(1-\frac{\lambda}{2}))$ and $c = \log(2\sqrt{n}/\delta)/b$. We have that

$$\inf_Q C(Q) = C(Q^*), \text{ where } q^*(\mathbf{w}) = \frac{e^{-\frac{a}{b}\hat{L}(\mathbf{w})}p(\mathbf{w})}{\mathbb{E}_{\mathbf{w}\sim P}[e^{-\frac{a}{b}\hat{L}(\mathbf{w})}]}. \tag{3.4}$$

Proposition 3.2.1. *Let G be any distribution such that $G \neq Q^*$ and $G \neq P$. Then the tightness gap satisfies*

$$C(G) - C(Q^*) \geq bD_{KL}(G||Q^*).$$

Proof. The proof can be found in Appendix A. □

Note. Substituting $b = 1/(n\lambda(1-\frac{\lambda}{2}))$, we can see that the lower bound decays with $\frac{1}{n}$. We now turn to explaining our method for estimating $C(Q^*)$.

3.3 The Gibbs Posterior and its Properties

Recall from Section 2.2.5 that the density of the Gibbs posterior measure that minimises the PAC-Bayes λ bound, when stated in terms of $\widetilde{L}_S^{\text{CE}}(\mathbf{w})$ is,

$$q_\lambda^*(\mathbf{w}|z) = \frac{e^{-n\lambda\widetilde{L}_S^{\text{CE}}(\mathbf{w})}p(\mathbf{w})}{\mathbb{E}_{\mathbf{w}\sim P}\left[e^{-n\lambda\widetilde{L}_S^{\text{CE}}(\mathbf{w})}\right]}. \tag{3.5}$$

We give the Gibbs posterior in terms of $\widetilde{L}_S^{\text{CE}}(\mathbf{w})$ because we can only sample from this form as we require the risk to be differentiable and bounded. The logarithm of the joint distribution of $\mathbf{w} \sim Q_\lambda^*$ and $z \sim D$ is

$$J(\lambda) := \log p(\mathbf{w}, z) = -\lambda n\widetilde{L}_S^{\text{CE}}(\mathbf{w}) + \log p(\mathbf{w}). \tag{3.6}$$

Note that we suppressed the dependence of $\widetilde{L}_S^{\text{CE}}(\mathbf{w})$ on the data z. We will make use of the following simple fact.

Proposition 3.3.1. *Fix a prior measure P with density $p(\mathbf{w})$ and define a (possibly generalized) likelihood $p(z|\mathbf{w})$. For any distribution Q that is dependent on the data z, with density $q(\mathbf{w}|z)$ the following holds*

$$D_{KL}[q(\mathbf{w}|z)||p(\mathbf{w})] = \mathbb{E}_q\left[\log p(z|\mathbf{w})\right] - \log p(z) + D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w}|z)], \tag{3.7}$$

where $p(\mathbf{w}|z)$ is the density of the posterior measure with respect to P and $p(z|\mathbf{w})$.

Proof. The proof can be found in Appendix A. □

Notice that if the (generalized) likelihood is defined as $p(z|\mathbf{w}) = e^{-n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})}$ and $Q = Q_\lambda^*$, then $D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w}|z)] = 0$ since Q_λ^* is precisely the (generalized) posterior. In this case, the equation reads as

$$D_{\mathrm{KL}}(Q_\lambda^*||P) = \mathbb{E}_{Q_\lambda^*}\left[-n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right] - \log Z, \tag{3.8}$$

where $Z = \mathbb{E}_P\left[e^{-n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})}\right]$.

Recall that our end goal is to reliably estimate $C(Q_\lambda^*, \lambda)$. The PAC-Bayes framework requires producing an estimate of the bound that holds with probability $1-\delta$ for arbitrary $\delta > 0$. Since we can only estimate $\widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*)$ and $D_{\mathrm{KL}}(Q_\lambda^*||P)$ from a finite number of samples, we will need to provide a high-probability upper bound on each of these estimates in order to control the end probability of the bound.

The first term $\widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*)$ is approximated by drawing samples from Q_λ^* using HMC and forming a Monte Carlo average. From Proposition 3.3.1, we see that as long as we have samples from Q_λ^*, estimating $D_{\mathrm{KL}}(Q_\lambda^*||P)$ boils down to estimating the log marginal likelihood $\log Z$. Notice that $\mathbb{E}_{Q_\lambda^*}\left[-n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right] = -n\lambda \widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*)$, hence this term cancels with the first term of the PAC-Bayes λ bound, and the bound becomes a constant multiple of $-\log Z$. This connection between PAC-Bayes and Bayesian inference has been pointed out in Germain et al. [2016]. However, this convenient property holds only for linear PAC-Bayes bounds, and for bounds on the cross-entropy loss L^{CE} as the Gibbs posterior was defined in terms of this function. In summary, we have the following three objectives.

1. Sample from the Gibbs posterior corresponding to the PAC-Bayes λ bound
2. Estimate $D_{KL}(Q_\lambda^* \,||\, P)$ via estimating $\log Z$
3. Obtain high-probability upper bounds on our estimates for $\widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*)$ and $D_{KL}(Q_\lambda^* \,||\, P)$.

We discuss objective 1 in the next section. Objectives 2 and 3 will be discussed in Sections 3.5 and 3.6, respectively.

3.4 Step I: Sampling from the Gibbs Posterior

We now explain how we set up HMC. As the target unnormalized density, we will use the log joint of the Gibbs posterior $J(\lambda)$, given in Equation 3.6. The Gibbs posterior depends on the prior. We use a data-independent prior. The loss function used in the exponent of the Gibbs density will be the bounded version of the cross-entropy loss, $\widetilde{l^{\mathrm{CE}}}(\mathbf{w},z) \in [0,1]$, hence the risk in the exponent becomes $\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$. We will evaluate risk certificates both for $\widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*)$ and $\hat{L}_S^{0-1}(Q_\lambda^*)$. The risk certificate will only be optimal for $\widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*)$, since our Gibbs posterior uses this risk function. However, we expect our method to give close-to-optimal risk certificates for $\hat{L}_S^{0-1}(Q_\lambda^*)$ too.

We explain the used hyperparameter setups in Section 4.2. We argue for the validity of our samples using the Effective Sample Size, the MCMC Standard Error and the $\hat{R}$ diagnostics. Please find the details in Section 4.4.

Having obtained samples from the Gibbs posterior, the next step is estimating $D_{KL}(Q_\lambda^* \mid\mid P)$ via estimating $\log Z$. An efficient method for estimating log-marginal likelihoods is *thermodynamic integration*. This technique gives a lower bound on log marginal likelihoods, hence an upper bound for $D_{KL}(Q_\lambda^* \mid\mid P) = \mathbb{E}_{Q_\lambda^*}\left[-n\lambda\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right] - \log Z$. We will use this technique in the next sections.

3.5 Step II: Estimating the KL Divergence

In this section, we describe our method to estimate the KL divergence term in PAC-Bayes bounds, using thermodynamic integration. We first give a general review of this technique, before applying it to our problem in Section 3.5.2.

3.5.1 A Primer on Thermodynamic Integration

Thermodynamic integration (TI) is a physics-inspired method that allows us to approximate intractable normalizing constants of high dimensional distributions [Masrani et al., 2019]. The main insight is to transform the problem into estimating the difference of two log normalizing constants. Since we are required to estimate $\log Z$, this framework suits our purposes.

Consider two probability measures $\Pi_1, \Pi_2 \in \mathcal{M}_1(\mathcal{W})$ with corresponding densities $\pi_1(\mathbf{w}), \pi_2(\mathbf{w})$ and their unnormalized versions

$$\pi_i(\mathbf{w}) = \frac{\tilde{\pi}_i(\mathbf{w})}{W_i}, \qquad W_i = \int \tilde{\pi}(\mathbf{w}) \mathrm{d}\mathbf{w}, \quad i \in \{0,1\}. \tag{3.9}$$

To apply TI, we form a geometric path between $\pi_0(\mathbf{w})$ and $\pi_1(\mathbf{w})$ via a scalar parameter $\beta \in [0,1]$:

$$\pi_\beta(\mathbf{w}) = \frac{\tilde{\pi}_\beta(\mathbf{w})}{W_\beta} = \frac{\tilde{\pi}_1(\mathbf{w})^\beta \tilde{\pi}_0(\mathbf{w})^{1-\beta}}{W_\beta}, \quad W_\beta = \int \tilde{\pi}_\beta(\mathbf{w}) \mathrm{d}\mathbf{w}, \quad \beta \in [0,1]. \tag{3.10}$$

The central identity of thermodynamic integration is as follows. The right-hand side of Equation 3.11 is referred to as the *thermodynamic integral*.

Proposition 3.5.1. *Define the potential as* $U_\beta(\mathbf{w}) = \log \tilde{\pi}_\beta(\mathbf{w})$ *and let* $U'_\beta(\mathbf{w}) := \frac{\partial}{\partial \beta} U_\beta(\mathbf{w})$. *Then,*

$$\log(W_1) - \log(W_0) = \int_0^1 \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'_\beta(\mathbf{w})\right] \mathrm{d}\beta. \tag{3.11}$$

Proof. Please refer to Appendix A in Masrani et al. [2019]. □

We can now explain how to use TI in our context of estimating $D_{\mathrm{KL}}(Q^*_\lambda || P)$.

3.5.2 Estimating the KL with Thermodynamic Integration

As seen in Section 3.5.1, thermodynamic integration requires defining two probability measures with unnormalized densities, $\tilde{\pi}_i(\mathbf{w})$ and calculates the difference of the log normalizing constants $\log W_1 - \log W_0$. If $\tilde{\pi}_0$ is defined such that $W_0 = 1$, then TI calculates the log normalizing constant of $\tilde{\pi}_1$. We thus define

$$\tilde{\pi}_0(\mathbf{w}) := p(\mathbf{w}), \quad \tilde{\pi}_1(\mathbf{w}) := p(z, \mathbf{w}), \tag{3.12}$$

thus $W_0 = 1$ and $W_1 = \int p(z,\mathbf{w}) d\mathbf{w} = Z$. In TI, we then define the geometric path for $\beta \in [0,1]$:

$$\tilde{\pi}_\beta(\mathbf{w}) := p(z,\mathbf{w})^\beta p(\mathbf{w})^{1-\beta}. \tag{3.13}$$

Then we have that $U_\beta(\mathbf{w}) = \log \tilde{\pi}_\beta(\mathbf{w}) = \beta \log p(z,\mathbf{w}) + (1-\beta)\log p(\mathbf{w})$ and thus

$$\frac{\partial}{\partial \beta} U_\beta(\mathbf{w}) = \log \frac{p(z,\mathbf{w})}{p(\mathbf{w})} = \log \frac{e^{-n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})} p(\mathbf{w})}{p(\mathbf{w})} = -n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}). \tag{3.14}$$

Then, the thermodynamic integration formula (Equation 3.11 yields the following form for the log normalizing constant

$$\log Z = \int_0^1 \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[-n\lambda \hat{L}_S(\mathbf{w})\right] \mathrm{d}\beta. \tag{3.15}$$

We will estimate $\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[-n\lambda \widetilde{L}_S^{\text{CE}}(\mathbf{w})\right]$ with a Monte Carlo average, using samples $\mathbf{w} \sim \pi_\beta$. To sample from each π_β, we can use HMC again to draw from the log joint

$$\begin{aligned}
\log \tilde{\pi}_\beta(\mathbf{w}) &= \beta \log p(z, \mathbf{w}) + (1-\beta)\log p(\mathbf{w}) && (3.16)\\
&= -\beta\lambda n \widetilde{L}_S^{\text{CE}}(\mathbf{w}) + \beta \log p(\mathbf{w}) + (1-\beta)\log p(\mathbf{w}) && (3.17)\\
&= -\beta\lambda n \widetilde{L}_S^{\text{CE}}(\mathbf{w}) + \log p(\mathbf{w}). && (3.18)
\end{aligned}$$

Defining $\tilde{\lambda} = \beta\lambda$ and comparing to Eq 3.6, we see that $\log \tilde{\pi}_\beta(\mathbf{w}) = J(\tilde{\lambda})$. Hence sampling from π_β for various β is equivalent to sampling from $Q^*_{\beta\lambda} = Q^*_{\tilde{\lambda}}$. In principle, we can thus reuse samples obtained for λ to estimate bounds for $\beta\lambda$ too. In our experiments with Gibbs posteriors, we fix $\lambda = 1$ and leave this direction to future work.

We now turn to estimating the integrand and show that calculating the left Riemann sums and even the trapezium rule guarantees an upper bound.

3.5.3 Properties of the Integrand

In this section, we state our results for the *negative* log normalizing constant, $-\log Z$ for convenience. A naive strategy for calculating the integrand for $\log Z$ is to use a Monte Carlo average for the integral.

$$\begin{aligned}
-\log Z &= \int_0^1 \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda \widetilde{L}_S^{\text{CE}}(\mathbf{w})\right] \mathrm{d}\beta && (3.19)\\
&\approx \frac{1}{B}\sum_{b=0}^{B-1}\left[\mathbb{E}_{\mathbf{w}\sim\pi_{\frac{b}{B}}}\left[n\lambda \widetilde{L}_S^{\text{CE}}(\mathbf{w})\right]\right] && (3.20)
\end{aligned}$$

Notice that the right-hand-side in Equation 3.19 is a left Riemann sum (i.e. the integral is approximated at the left end of each subinterval of $[0,1]$) on the function $\mathbf{w} \mapsto \mathbb{E}_{\mathbf{w}\sim\pi_{b/B}}\left[n\lambda \widetilde{L}_S^{\text{CE}}(\mathbf{w})\right]$, where each subinterval has length $\frac{1}{B}$.

The fact that this quantity upper bounds the KL divergence is established in Masrani et al. [2019] by showing that the integrand is a decreasing function of β. Let $g(\beta) = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[-U'_\beta(\mathbf{w})\right] = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda \widetilde{L}_S^{\text{CE}}(\mathbf{w})\right]$, our integrand. We notice that $U'_\beta(\mathbf{w}) = -n\lambda \widetilde{L}_S^{\text{CE}}(\mathbf{w})$

is independent of β, hence we may abandon the subscript and use notation $U'(\mathbf{w}) = -n\lambda\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}))$. Masrani et al. [2019] show that

Proposition 3.5.2. $\frac{\partial g(\beta)}{\partial \beta} = -\mathrm{Var}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right] \leq 0.$

Proof. Please refer to Appendix A in Masrani et al. [2019]. Please note that they define $g(\beta)$ to be the *negative* of our $g(\beta)$. □

Hence $\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[-U'(\mathbf{w})\right]$ is a monotonically decreasing function, and this shows that calculating the left sums upper bounds our integral. In fact, we can show more. Below we show that $g(\beta)$ is convex, hence the trapezium rule can also be used to upper bound this integral.

Proposition 3.5.3. *Let* $g(\beta) = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[-U'(\mathbf{w})\right] = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda\widetilde{L}_S^{CE}(\mathbf{w})\right]$. *Then we have that* $\frac{\partial^2 g(\beta)}{\partial \beta^2} \geq 0$, *hence* $g(\beta)$ *is convex.*

Proof. The full proof is given in Appendix A. □

We offer an intuitive argument for the proofs of Propositions 3.5.2 and 3.5.3. Since $g(\beta) = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$ is defined as the expectation of the loss under a Gibbs distribution with temperature β, by increasing β, the Gibbs density concentrates increasingly around the minima of the loss landscape. This implies that $g(\beta)$ decreases as β increases. Further, the variance of π_β decreases, and consequently so does $\mathrm{Var}_{\mathbf{w}\sim\pi_\beta}\left[\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$. Thus $g'(\beta)$ is increasing, and hence $g(\beta)$ is convex.

The above result implies that the trapezium rule can be used to obtain an estimate of the integral which is an upper bound, as long as we are able to accurately estimate the values of the function, $\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda\widehat{L}_S^{\mathrm{CE}}\right]$. Using a discretization $\beta_1, \beta_2, \ldots\beta_K$ on [0, 1], and m_k samples from each π_{β_k}, our estimator for $-\log Z$ with the trapezium rule becomes

$$-\widehat{\log Z} := \sum_{k=1}^{K-1} \frac{\left(\frac{1}{m_k}\sum_{i=1}^{m_k}\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}_{i,k})\right) + \left(\frac{1}{m_{k+1}}\sum_{i=1}^{m_{k+1}}\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}_{i,k+1})\right)}{\beta_{k+1} - \beta_k}, \mathbf{w}_{i,k} \sim \pi_{\beta_k}. \tag{3.21}$$

Finally, we estimate the KL divergence as

$$D_{\mathrm{KL}}(Q_\lambda^* || P) \approx -\frac{1}{m_1}\sum_{i=1}^{m_1}\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}_{i,1}) - \widehat{\log Z} \tag{3.22}$$

However, in the end, we wish to obtain a high-probability bound, and hence we need to correct for the fact that these estimates are obtained from a finite number of samples $\mathbf{w}_i$. This correction is also required for $\widetilde{L}_S^{\mathrm{CE}}(Q_\lambda^*) \approx \frac{1}{m}\sum_{i=1}^{m}\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}_i)$. The correction is done by computing a high-probability upper bound for each of these and is explained in the next section.

3.6 Step III: High-Probability Upper Bounds

We now discuss the third, final step of our method, which is to ensure that the computed PAC-Bayes risk certificate holds with probability at least $1-\delta$, for arbitrary $\delta > 0$. How we achieve this will be different for MFVI and for the Gibbs posterior.

In the case of MFVI, the steps are as follows: (i) we use the kl inversion bound (Theorem 2.2.3 in Section 2.2.5) to upper bound our Monte Carlo estimate for $\widetilde{L}_S^{\mathrm{CE}}(Q)$ with probability $1-\delta_1$, and then (ii) compute the PAC-Bayes bound (risk certificate) with $\delta_2 = 1-\delta+\delta_1$, where we chose δ_1 and δ_2 such that $\delta_1+\delta_2=\delta$. This will ensure that the risk certificate holds with probability at least $1-\delta$. However, in (i) we used the assumption that our sample from the approximate posterior is i.i.d., which is required for Theorem 2.2.3.

For the Gibbs posterior (ii) remains the same, but for (i), we cannot directly use Theorem 2.2.3 as our HMC sample is likely not i.i.d. In addition, since our estimate for the KL divergence is also not exact (based on a finite number of samples), a high-probability upper bound is required for this term too. There are many options to obtain confidence intervals for MCMC samples, but there is no perfect choice. We now discuss three options, which will all be computed in this thesis. In general, the two large categories of bounds are (i) concentration inequalities and (ii) asymptotic confidence intervals. The former are appealing because they are non-asymptotic, but their assumptions are very hard to verify for MCMC samples. Asymptotic confidence intervals are easier to verify, but they only provide a high-probability bound as $n \to \infty$. We discuss an option in each category.

The bound requiring independence

The bound most commonly used with MFVI is Theorem 2.2.3. In order to compare our Gibbs posterior bound to this, we would like to use Theorem 2.2.3. We thus need to argue for our samples being i.i.d. In Chapter 4.4, we compute $\hat{R}$ statistics across four chains both in weight- and function space to argue for convergence to (at least) a mode of the Gibbs posterior. In order to achieve approximate independence, we thin our chains with rates high enough to bring the ESS close to the (remaining) sample size. Since this justification is very informal, we also compare the Gibbs posterior bound to MFVI using other confidence intervals.

The asymptotic bound

The Markov chain Central Limit Theorem (MCCLT) provides a confidence interval, but the conditions of this theorem are very strong. Fortunately, Rosenthal [2017] provide a similar, but weaker confidence interval that does not require the MCCLT. The assumptions on our estimator $\frac{1}{m}\sum_{i=1}^{m}\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}_i)$, $\mathbf{w}_i \sim Q$ are as follows.

1. $O\left(\frac{1}{n}\right)$ variance of the estimator
2. Bias of order smaller than $O\left(\frac{1}{\sqrt{n}}\right)$
3. An estimator $\hat{\sigma}_m$ for the *asymptotic variance* $\sigma := \lim_{m\to\infty} m\mathrm{Var}_{\mathbf{w}\sim Q}\left[\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$ that converges in probability.

Assumptions 1. and 2. are typical in MCMC applications [Rosenthal, 2017]. For assumption 3, we estimate the asymptotic variance using the fact that the MCMC standard error converges to $\frac{\sigma}{\sqrt{m}}$, following one of the suggestions of Rosenthal [2017]. The (one-sided) version on the confidence interval in Rosenthal [2017] has form

$$I_{m,\varepsilon} = \left[0, \frac{1}{m}\sum_{i=1}^{m}\hat{L}_S(\mathbf{w}_i) + m^{-\frac{1}{2}}\hat{\sigma}_m(2\alpha)^{-\frac{1}{2}}(1+\varepsilon)\right) \quad \text{with prob. at least } 1-\alpha, \tag{3.23}$$

where m is the number of samples, $\hat{\sigma}_m$ is an estimate of the asymptotic variance, and ε appears in the proof when formalizing Assumption 3. We take $\varepsilon = 0.01$. Fair comparison to MFVI demands that we use the classical CLT to obtain an asymptotic confidence interval in this case. This is valid as both the transformed cross-entropy and the $0-1$ losses are bounded. The CLT confidence interval has form $I_\alpha = \left[0, \frac{1}{m}\sum_{i=1}^{m}\hat{L}_S(\mathbf{w}_i) + q_\alpha\frac{\hat{\sigma}_m}{\sqrt{n}}\right)$, where q_α is the appropriate quantile of the standard Gaussian distribution[1].

The bound under milder assumptions

Given the difficulties in verifying assumptions for MCMC samples, the reader may wonder if we can give, perhaps looser, estimates for $C(Q_\lambda^*, \lambda)$ with milder assumptions. Let Q denote the underlying distribution of our HMC samples, which may not be Gibbs posterior. The following simple proposition can be used to give a (loose) bound for $D_{\mathrm{KL}}(Q||P)$ only requiring that Q is not much further from the Gibbs posterior than the mean-field approximation. We expect that Q is much closer to the Gibbs posterior than MFVI, hence this is a very mild assumption.

[1] Although the population variance is unknown, we decided to use this instead of the Student-t distribution to ensure comparability to $I_{m,\varepsilon}$, and because we have a large sample size, $m = 20,000$ across the four chains.

Proposition 3.6.1. *Let P be the prior, and let Q^*_λ denote the corresponding (Gibbs) posterior $Q^*_\lambda \propto e^{-n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})} p(\mathbf{w})$. Suppose that we are able to simulate from a distribution Q. Let G be another distribution (in our case, a Gaussian) such that $D_{\mathrm{KL}}(Q||Q^*_\lambda) \leq D_{\mathrm{KL}}(G||Q^*_\lambda) + \mathbb{E}_Q\left[\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$. Then,*

$$D_{\mathrm{KL}}(Q||P) \leq n\lambda \mathbb{E}_G\left[\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right] + D_{\mathrm{KL}}(G||P). \tag{3.24}$$

Proof. The proof van be found in Appendix A. □

We can use this estimate to bound the PAC-Bayes λ and PAC-Bayes relative entropy bounds to obtain an upper bound on $C(Q,\lambda)$, which we denote by $C^{\mathrm{UB}}(Q,\lambda)$. For this, we need an estimate for $\widetilde{L}_S^{\mathrm{CE}}(Q)$, for which we use the kl inverse bound (Theorem 2.2.3). It can be shown easily that we have $C^{\mathrm{UB}}(Q,\lambda) \geq C(G,\lambda)$ if we use the λ bound and the cross-entropy loss. However, we are most interested in risk certificates in terms of accuracy. For the $0-1$ loss, our upper bound on the λ bound becomes

$$L^{0\text{-}1}(Q) \leq \frac{\hat{L}_S^{0\text{-}1}(Q)}{1-\frac{\lambda}{2}} + \frac{D_{\mathrm{KL}}(G||P) + \log(2\sqrt{n}/\delta)}{n\lambda(1-\frac{\lambda}{2})} + \frac{\widetilde{L}_S^{\mathrm{CE}}(G)}{1-\frac{\lambda}{2}}. \tag{3.25}$$

This can result in $C^{\mathrm{UB}}(Q,\lambda) \leq C(G,\lambda)$ if Q has lower $0-1$ loss than G. The difference between this and the true λ bound on $L^{0\text{-}1}(Q)$ is precisely $D_{\mathrm{KL}}(G||Q^*_\lambda) - D_{\mathrm{KL}}(Q||Q^*_\lambda) + \widetilde{L}_S^{\mathrm{CE}}(Q)$. We will use this bound for a sanity check on our results. If our computed value for $C(Q,\lambda)$ is much higher than this bound, then we likely overestimated the true value, even if our bound is smaller than the one with MFVI. If, on the contrary, our estimate is much lower than this upper bound, that means that either $D_{\mathrm{KL}}(G||Q^*_\lambda) - D_{\mathrm{KL}}(Q||Q^*_\lambda) + \widetilde{L}_S^{\mathrm{CE}}(Q)$ is high or our chains did not even manage to achieve $D_{\mathrm{KL}}(Q||Q^*_\lambda) \leq D_{\mathrm{KL}}(G||Q^*_\lambda) + \widetilde{L}_S^{\mathrm{CE}}(Q)$.

Having obtained high-probability upper bounds on our estimates for both terms of PAC-Bayes bounds, both the PAC-Bayes relative entropy bound and the λ bound can now be computed. In the next, final section of the present chapter, we briefly summarize our method together with the assumptions required at each step.

3.7 Summary of Method and Assumptions

In summary, our method for estimating PAC-Bayes bounds for the Gibbs posterior consists of the following steps.

1. Drawing (approximate) samples from the Gibbs posterior Q^*_λ using HMC
2. Estimating $\hat{L}_S(Q^*_\lambda)$ as $\hat{L}_S(Q^*_\lambda) \approx \frac{1}{m}\sum_{i=1}^{m}\hat{L}_S(\mathbf{w}_i)$, $\mathbf{w}_i \sim Q^*_\lambda$ using $\hat{L}_S = \widetilde{L}_S^{\text{CE}}, \hat{L}_S^{\text{0-1}}$
3. Estimating $D_{\text{KL}}(Q^*_\lambda||P)$ using thermodynamic integration.
   (a) Sampling from $Q^*_{\lambda\beta}$ for several $\beta \in [0,1]$
   (b) Estimating $\hat{L}_S(Q^*_{\lambda\beta})$ as in Step 2
   (c) Estimating the thermodynamic integral with the trapezium rule, i.e.
   $$-\widehat{\log Z} := \sum_{k=1}^{K-1} \frac{\left(\frac{1}{m_k}\sum_{i=1}^{m_k}\widetilde{L}_S^{\text{CE}}(\mathbf{w}_{i,k})\right) + \left(\frac{1}{m_{k+1}}\sum_{i=1}^{m_{k+1}}\widetilde{L}_S^{\text{CE}}(\mathbf{w}_{i,k+1})\right)}{\beta_{k+1}-\beta_k}, \mathbf{w}_{i,k} \sim \pi_{\beta_k}$$
   (d) Estimating $D_{\text{KL}}(Q^*_\lambda||P)$ as
   $$D_{\text{KL}}(Q^*_\lambda||P) \approx -\frac{1}{m_1}\sum_{i=1}^{m_1}\widetilde{L}_S^{\text{CE}}(\mathbf{w}_{i,1}) - \widehat{\log Z}$$
4. Constructing a high-probability upper bound on each $\frac{1}{m_k}\sum_{i=1}^{m_k}\hat{L}_S(\mathbf{w}_{ik})$, $\mathbf{w}_{ik} \sim Q^*_{\lambda\beta_k}$ to ensure that our estimates are high-probability upper bounds for $\hat{L}_S(Q^*_\lambda)$ and $D_{\text{KL}}(Q^*_\lambda||P)$.
5. Computing the two PAC-Bayes bounds using the upper bounds from Step 4.

We now recall the assumptions required by our method. Assume that the true distribution we are sampling from is Q. Ideally, for Steps 2, 3 to supply accurate results, the Markov chain Central Limit Theorem is required, i.e. that our chain converges in function space. However, since PAC-Bayes bounds can be computed for any Q, we do not need to require the MCCLT to compute the bound *for* Q. Hence the only assumptions we have are those in our high-probability upper bounds.

The high-probability bounds of Step 4 have been chosen to have assumptions on varying strength levels, allowing the reader to choose according to their taste. The bound based on Theorem 2.2.3 requires that $\mathbf{w} \sim Q$ are i.i.d. The confidence interval from Rosenthal [2017] required conditions on the order of the bias and variance of our estimators for $\mathbb{E}_{\mathbf{w}\sim Q}[\hat{L}_S(\mathbf{w})]$ and an estimate for the asymptotic variance that converges in probability. These two bounds additionally require $D_{\text{KL}}(Q||Q^*_\lambda)$ to be small so that the approximation in Step 3d that neglects term $D_{\text{KL}}(Q||Q^*_\lambda)$ is reasonably accurate. The third bound only requires

$D_{\mathrm{KL}}(Q||P) \leq D_{\mathrm{KL}}(G||P) + \mathbb{E}_Q\left[\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$ for a mean-field Gaussian approximate posterior we are comparing to.

We emphasize that in Step 3c, the approximation of $-\log Z$ via the trapezium rule supplies an upper bound. Hence possible violations of our assumptions may be (partly) masked by this approximation.

Summary. In this chapter, we presented our method to estimate optimal PAC-Bayes bounds. We also proved simple results including (i) a lower bound on the tightness gap, (ii) the convexity of the thermodynamic integral, and (iii) a bound under mild assumptions we will use as a “sanity check” on our estimates. In the next chapter, we apply our method to compute (close-to) optimal PAC-Bayes bounds and compare these to MFVI.

Chapter 4

Experiments and Results

Chapter Roadmap

In this chapter, we apply our method for estimating optimal PAC-Bayes bounds on the MNIST dataset and some of its variations. As the first step of our method, sampling with HMC is challenging to implement, we first gain some intuition on HMC and Gibbs posteriors with some small experiments. Section 4.1 provides a two-dimensional toy example, where we can visualize Gibbs-like posteriors. In Section 4.2, we provide our experimental setup. Then, in Section 4.3, we explore some calibration choices for HMC and the prior variance. In Section 4.4, we assess our HMC samples with the diagnostics measures introduced in Section 2.3.2, before presenting our RC estimates in Section 4.5.

4.1 A Motivating Toy Example

We begin by illustrating the benefits of approximating posteriors with HMC instead of MFVI with a toy example. We consider the task of inferring a parameter $\mathbf{w} \in [15,55] \times [0,40]$ using a prior $p(\mathbf{w})$ some risk function (generalised negative log likelihood) $R : [15,55] \times [0,40] \to [0,1]$, $\mathbf{w} \mapsto R(\mathbf{w})$. The parameter $\mathbf{w}$ may govern the generation of some data source. However, as we seek to visualize the model in two dimensions, we consider the data fixed and only treat the risk as a function of $\mathbf{w}$. Note that generalised Bayesian inference is model-free, hence we do not require an explicit data modelling setup.

In order to illustrate the multi-modality typical of Gibbs posteriors, we construct a two-dimensional surface with multiple local minima. Taking inspiration from [Kim et al., 2022], we define $R(\mathbf{w})$ as the negative log-mixture of four KL-divergence-driven energy models.

Specifically, we define the risk as

$$R(\mathbf{w}) = \frac{-\log(\alpha_1 e^{-E_1/\beta_1^2+E} + \alpha_2 e^{-E_2/\beta_2^2+E} + \alpha_3 e^{-E_3/\beta_3^2+E} + \alpha_4 e^{-E_4/\beta_4^2+E} + E) - 0.1218}{1.6592},$$

where $E_i = \frac{(\mathbf{w}_0-\mu_i)^2+\mathbf{w}_1^2}{2s_i^2} - \frac{1}{2} + \log s_i - \log(|\mathbf{w}_1|)$, $i \in \{1,2,3,4\}$ and E is the element-wise maximum of the E_i. We chose the numerical constants in order to ensure that $R(\mathbf{w}) \in [0,1]$. We also set

$$(\mu_1,\mu_2,\mu_3,\mu_4) = (32,20,48,32),$$

$$(s_1,s_2,s_3,s_4) = (8.5,3.5,4.5,2.3),$$

$$(\alpha_1,\alpha_2,\alpha_3,\alpha_4) = (0.75,0.5,0.5,0.3),$$

$$(\beta_1,\beta_2,\beta_3,\beta_4) = (1.9,1.6,1.6,1.6).$$

The resulting surface has three local minima.

We wish to visualize the effect of some of our future experimental choices on this toy problem. We focus on (i) the effect of the prior variance on multi-modality, (ii) the comparison of the generalised posterior with the PAC-Bayes objectives, and finally (iii) a comparison between the result of running HMC and MFVI on this problem.

The results of (i) are given in Figure 4.1. The multi-modality of the risk is only reserved at the highest prior variance setup (bottom row). Figure 4.2 carries out (ii). This comparison is important, since the use of the PAC-Bayes objectives as loss functions, including the ELBO can be motivated by the fact that their corresponding surface "looks" similar to that of the true (generalised) posterior. Hence (successful) optimization of the PAC-Bayes objective using MFVI will result in an approximate posterior over **w** that matches the high-probability regions of the (generalised) posterior.

In Figure 4.3, we illustrate the advantages of using HMC instead of MFVI to describe the true (generalised) posterior (iii). The underlying prior distribution is a Gaussian $p(\mathbf{w}) := N(\mathbf{w};[36,10]^\top, 2000\cdot \mathrm{I})$. As we treat our data fixed, during both generation of the HMC samples and optimization of a mean-field approximate posterior, the only variable is **w**. For HMC, the unnormalized log joint is defined as $J(\mathbf{w}) := -R(\mathbf{w}) + \log p(\mathbf{w})$. A single HMC chain was generated, starting from $[32,20]^\top$. The same point was used to initialize the mean of the approximate posterior, while its covariance matrix was initialized to $3\cdot\mathrm{I}$.

HMC was run for 500 iterations after a burn-in phase of 100 iterations. We used 10 leapfrog steps with step-size 2.0. The described setup resulted in an acceptance probability of 58%. The generated samples are shown in green in Figure 4.3. MFVI was run for 1000 iterations with a learning rate of 0.01. The KL divergence term was scaled by 0.005. This

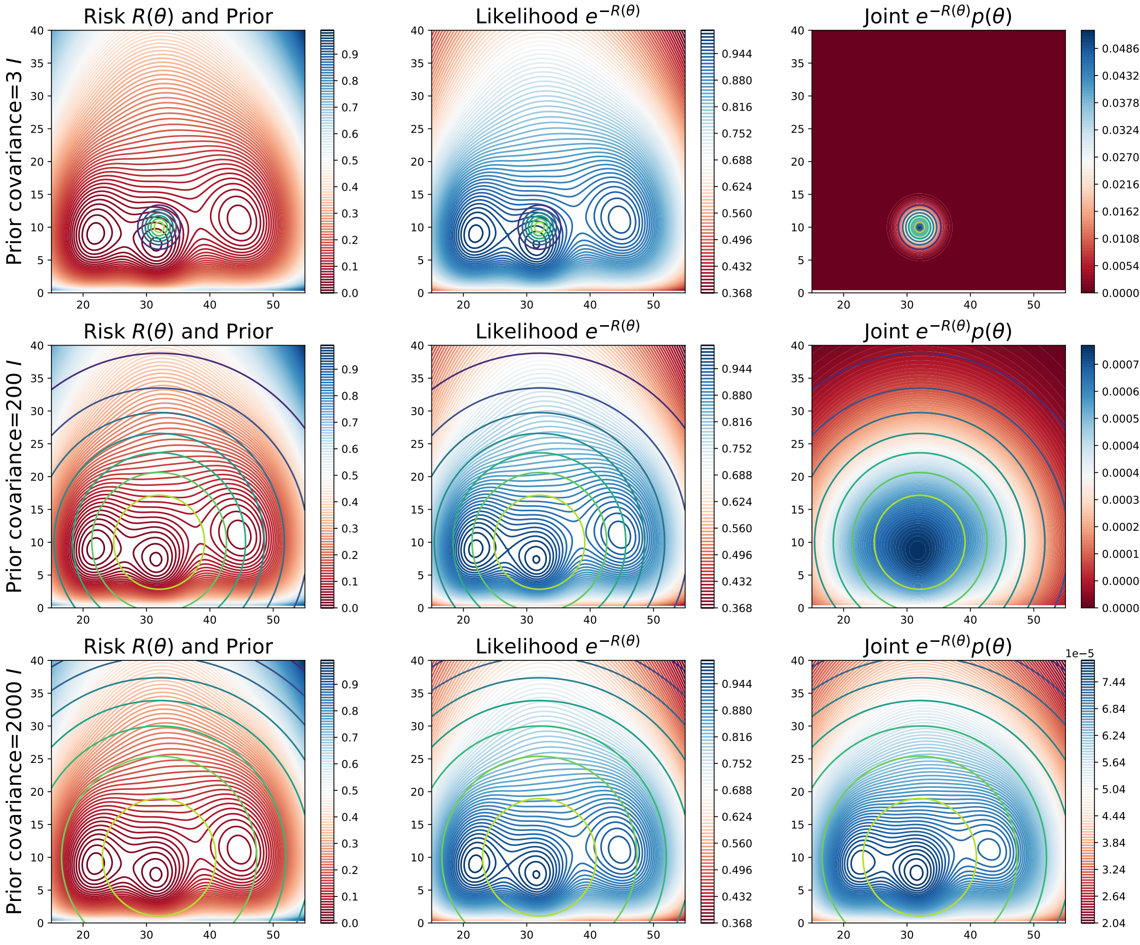


Fig. 4.1 Illustration of the effect of the prior variance on the posterior corresponding to a multi-modal generalised likelihood. The left panels show the risk $R(\mathbf{w})$ together with the contours of a Gaussian prior with mean $[36,10]$ and isotropic covariance $c \cdot \mathrm{I}$ for $c \in \{3,200,2000\}$, where I is the two-dimensional identity matrix. The middle panels show the generalised likelihood with the same priors, while the right panels show the joint, which is proportional to the generalised posterior. Only the largest covariances, $2000 \cdot \mathrm{I}$ preserved the multi-modality of the risk function.

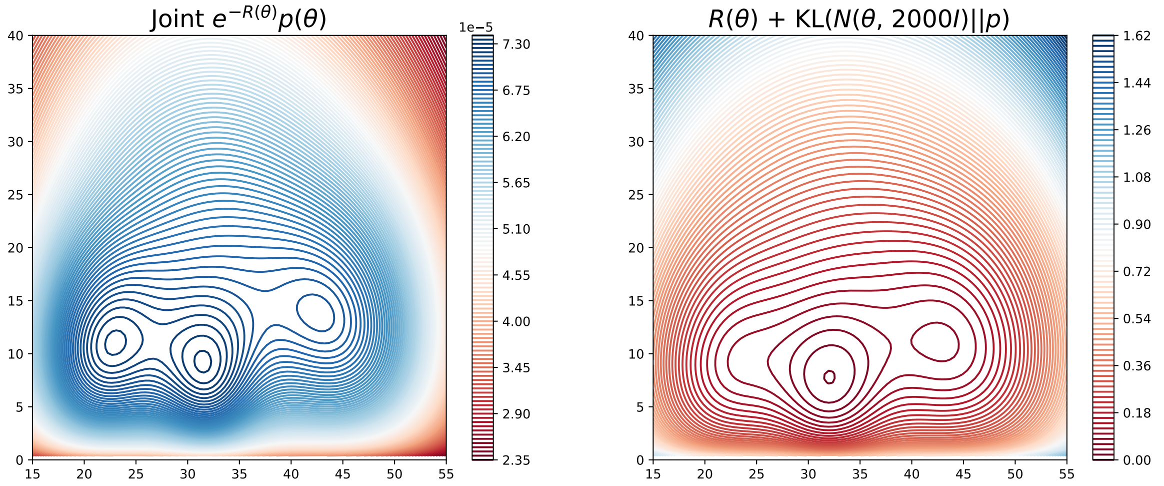


Fig. 4.2 Comparison of the generalised posterior with a generic linear PAC-Bayes objective (also equivalent to the ELBO for generalized likelihood). The left panel shows the contours of the joint distribution (proportional to the generalised posterior), while the right shows the surface of the PAC-Bayes objective. The surfaces look similar, which confirms the validity of using the PAC-Bayes objective to reach good approximate posteriors with MFVI.

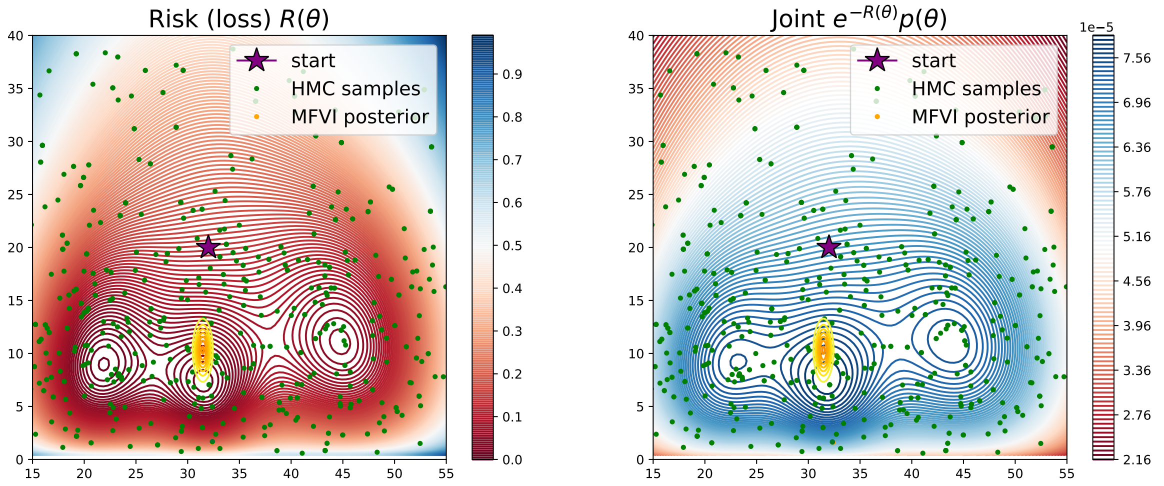


Fig. 4.3 Comparison of using HMC and MFVI to describe the posterior surface. The left panel shows the risk, while the right one shows the surface of the joint (proportional to the generalised posterior). The prior was taken to be a Gaussian $N(\mathbf{w};[36,10]^\top, 2000\cdot \mathrm{I})$ The HMC chain and the mean-field Gaussian approximate mean was initialized at the purple star $[32,20]^\top$. The HMC chain explores the high-probability regions of the posterior including the three modes, while the mean-field Gaussian approximate posterior only attempts to fit onto one of them.

was necessary in order to balance the magnitudes of the risk and the KL divergence term and avoid instability. The contours of the end posterior are shown in yellow and orange.

The inability of the Gaussian posterior to capture the multi-modal structure is clearly shown. HMC, on the other hand, explores the whole surface, allowing us to store information from around all three modes. Despite the multi-modality, the surface is relatively flat, hence our HMC samples cover the high-probability region evenly, with some concentration around the modes.

4.2 Experimental Setup

The previous section investigated the effect of experimental choices in a toy problem. We now leverage those findings in our experiments using neural networks. This section outlines the common setup of our experiments including datasets, architectures, and hyperparameters.

Datasets

Or two key consideration in selecting our datasets were

- *Efficiency.* Complicated datasets require large models, but it is challenging to use HMC with very large model dimensions. Accurate marginal likelihood estimation is intractable in very high dimensions. In addition, the size of the dataset impacts the time required to run HMC.
- *Relevance to the Deep Learning community.* Small datasets that can be learned without using neural networks are of less interest to the deep learning community. The simplest dataset for which PAC-Bayes bounds are commonly calculated is MNIST.

Based on these considerations, we chose to work with the following datasets, all of which are versions of MNIST.

- *(Full) MNIST.* The regular MNIST dataset with $60,000$ train and $10,000$ test datapoints.
- *Binary MNIST.* We mapped the labels of MNIST to binary using the map $y \mapsto y \bmod 2$.
- 14×14 *MNIST.* We reduced the size of each MNIST image from 28×28 to 14×14 by removing every second pixel. The characters remain identifiable.

To test the effect of the dataset size, we also experiment with using only half of the $60,000$ training datapoints in each of the datasets above. For each dataset, we use the training data to sample from the Gibbs posterior. The risk certificates are also evaluated in the training

Dataset	Architecture				# params
Binary MNIST	784	20	-	2	15742
14×14 MNIST	196	128	128	10	43018
MNIST	784	128	128	10	118282

Table 4.1 Our neural network architectures. The input dimension of each layer is shown, together with the final output dimension.

data, as is customary in PAC-Bayes [Perez-Ortiz et al., 2020]. The test data is only used to check the validity of our bounds by comparing them to the test error (which they must upper bound).

Architectures

We use simple Multi-Layer Perceptron (MLP) architectures in our sampling experiments. Due to the scalability limits of marginal likelihood estimation, the number of parameters are kept small. The architectures along with the corresponding number of parameters are given in Table 4.1.

Although MLPs may seem too limited and not parameter-efficient, their use is motivated by the trade-off between how complex our model is and the amount of time it takes to generate samples. Small CNN architectures were trialed, but sampling was unreasonably slow with them. Further, we do not wish to compete with the state-of-the-art in producing tight risk certificates. We only aim to compare (our estimate for) the optimal bound to the bounds obtained by MFVI. This comparison is interesting for any deep learning model on MNIST.

Hyperparameters

We first discuss the hyperparameters related to the prior. We initialize the prior means randomly from a truncated Gaussian distribution of mean 0 and separate variance for each layer, given by $\frac{1}{\sqrt{n_{\text{in}}}}$, where n_{in} is the input dimensionality of the layer. The constants of the truncation are ± 2 standard deviations. The prior covariance is set to $0.03 \cdot \mathrm{I}$, matching Perez-Ortiz et al. [2020]. We use the cross-entropy loss as our generalized likelihood function, in the transformed version (defined in Section 2.2.5, Equation 2.10), with $p_{\text{min}} = 10^{-4}$. This ensures that our loss is bounded.

We now explain the hyperparameters used with MFVI. As optimizer, we used SGD with learning rate 0.005 and momentum 0.95 and train for 150 epochs. The motivations for

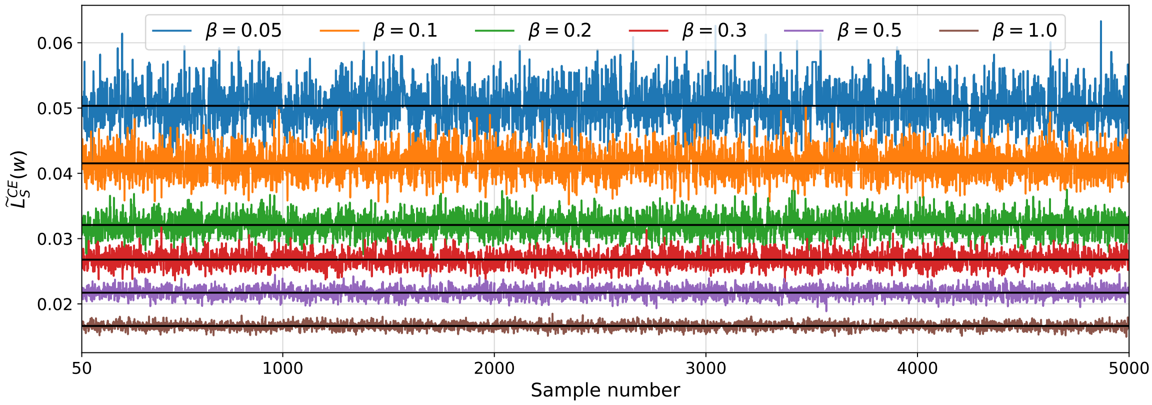


Fig. 4.4 HMC sample traces for a single chain without burn-in (50), for various β values. The black lines show the mean of each chain. The dataset is Binary MNIST - Half.

choosing SGD were discussed in Section 2.2.5. Since our objective function, the λ bound allows for cross-validation in λ, we optimize for this parameter while training and initialize it to 1. Following Perez-Ortiz et al. [2020], we use SGD with the same learning rate and momentum. This results in very small decrease in λ, at most 0.02 across our experiments. We have run initial experiments with higher learning rates for λ, but these resulted in looser risk certificates using the PAC-Bayes relative entropy bound.

We choose our risk certificate to hold with probability $1-\delta$ with $\delta = 0.05$. We will use $\delta_1 = 0.025$ to upper bound $\frac{1}{m}\sum_{i=1}^m L_S(\mathbf{w}_i)$, $\mathbf{w}_i \sim Q$ and then $\delta_2 = 0.025$ to compute bounds. This will ensure a risk certificate that holds with probability 0.95, via a union bound argument.

Next, we discuss the hyperparameters for HMC. We use full-batch HMC to sample from Gibbs posteriors of form $\propto e^{-\beta \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})} p(\mathbf{w})$ for $\beta \in (0,1)$. We run HMC with Metropolis-Hastings correction. We use constant step-sizes for the discretization of Hamiltonian dynamics, which we calibrate individually by testing the values $\{2,3,4,5,6,7,8,9,10,20,30\} \times 10^{-3}$ for each dataset and β. The step-sizes were chosen from this set by running HMC for around 100 sampling iterations for each step size, targeting the ideal acceptance probability for HMC given by 0.65 [Beskos et al., 2010]. Longer trial runs were not possible due to computational constraints. Lower step sizes were found suitable for higher β values. In every experiment, we use a trajectory length of 1.5. The authors of Izmailov et al. [2021] recommend a trajectory length of at least $\frac{\pi\sigma_{\text{prior}}}{2}$, which is approximately 0.27 in our case, but we choose higher than this in order to allow for more leapfrog steps. The chosen step-sizes resulted in leapfrog steps between 50 and 750. The difference in leapfrog steps is motivated by the fact that for lower β values, the generalized posterior is more similar to the prior and hence is easier to sample from. Each chain was run for 5000 iterations including a burn-in

Dataset	Discretization for $\beta \in [0,1]$
Binary MNIST	0.005, 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0, 8, 0.9, 1.0
14×14 MNIST	0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0, 8, 0.9, 1.0
MNIST	0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1.0

Table 4.2 Discretization values of $\beta \in$ [0, 1] for our datasets.

of 50 iterations. A low burn-in was chosen since we observed that the chains converge very quickly in function space $(\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w}))$. To support this choice, Figure 4.4 shows burned-in chains for the Binary MNIST - Half dataset for various values of β. To be able to test the convergence of the chains, we run four chains started from independent seeds for each β.

Depending on the size of the dataset, the interval $[0,1]$ was discretized to 7 or 12 values. The exact discretizations are reported in Table 4.2. We note that our discretizations are slightly denser near 0, which is where we lose the most with the trapezium rule approximation. There are also some more principled ways to select a good discretization, see for example Huber [2015]. Assuming a discretization of size K, we need to set $K+1$ δ values (one for term one of the PAC-Bayes bound, $K-1$ for the KL divergence estimate, and one for computing the bound). The δ values corresponding to the KL divergence term are set to $0.01/(K-1)$ and the others to 0.02, making a total of 0.05, consistent with MFVI.

4.3 Introductory Experiments

In this section, we test the effect of the prior variance and the HMC step-size on the quality of the samples produced.

Effect of the prior variance

We experiment with 10 prior variance parameters linearly spaced between 0.03 (used in Perez-Ortiz et al. [2020] and in our subsequent experiments) and 1.0. Figure 4.5 shows the results. The train cross-entropy, $0-1$, and test $0-1$ losses all tend to decrease as the variance is increased. The only exception is $s^2 = 0.89$. This exception may show that high prior variances can reinforce regions of the likelihood landscape that have problematic structures, making it harder for our Markov chain to explore the posterior. A smaller variance with an appropriate mean may be a more suitable choice.

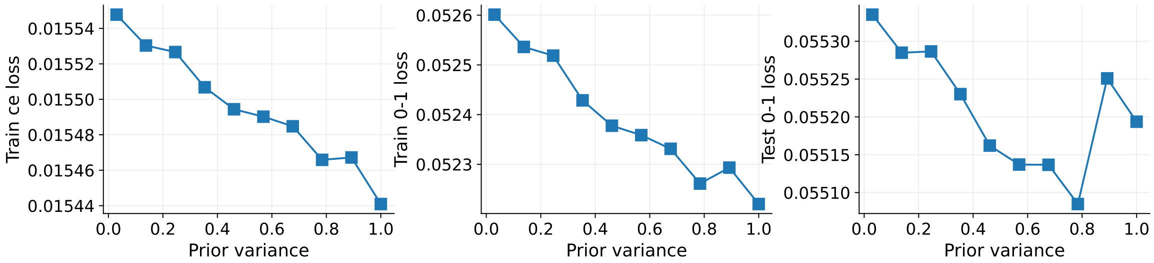


Fig. 4.5 Effect of the variance of the prior on the train cross-entropy loss $\frac{1}{m}\sum_{i=1}^{m}\widehat{L}_S^{\mathrm{CE}}(\mathbf{w}_i)$ (left), the train $0-1$ loss $\frac{1}{m}\sum_{i=1}^{m}\widehat{L}_S^{0-1}(\mathbf{w}_i)$ (middle) and the test $0-1$ loss $\frac{1}{m}\sum_{i=1}^{m}\widehat{L}_{\mathrm{Test}}^{0-1}(\mathbf{w}_i)$ (right). The dataset used was the $30,000$ samples from Binary MNIST, and the architecture was a one hidden layer MLP with 20 hidden units.

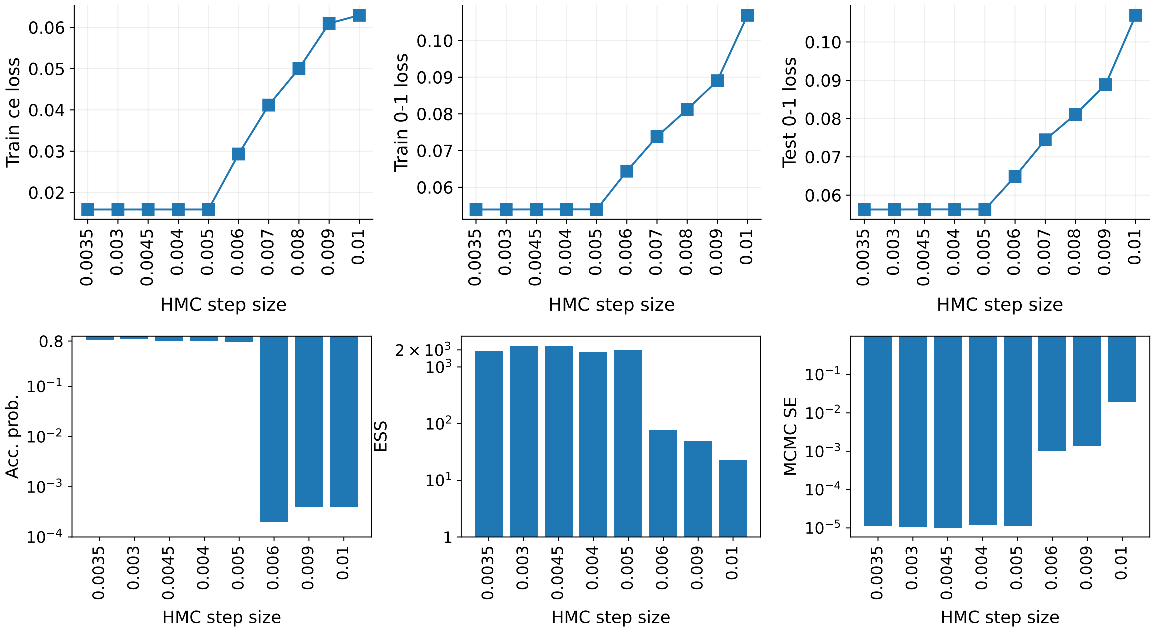


Fig. 4.6 Top row: effect of the HMC step size on the train cross-entropy loss $\frac{1}{m}\sum_{i=1}^{m}\widehat{L}_S^{\mathrm{CE}}(\mathbf{w}_i)$ (left), the train $0-1$ loss $\frac{1}{m}\sum_{i=1}^{m}\widehat{L}_S^{0-1}(\mathbf{w}_i)$ (middle) and the test $0-1$ loss $\frac{1}{m}\sum_{i=1}^{m}\widehat{L}_{\mathrm{Test}}^{0-1}(\mathbf{w}_i)$ (right). Bottom row: effect of the HMC step size on the acceptance probability (left), the mean Effective Sample Size (middle) and the mean MCMC Standard Error (right). The dataset used was the $30,000$ samples from Binary MNIST, and the architecture was a one-hidden-layer MLP with 20 hidden units.

Effect of the HMC step size

In this experiment, we tested 10 step size lengths between 0.003 and 0.01 on $30,000$ images selected from Binary MNIST, using our small, 1-hidden-layer architecture with 20 hidden units. We report the mean train cross-entropy, train $0-1$ and test $0-1$ losses in the top row of Figure 4.6, and the acceptance probability, ESS and MCMC SE in the bottom row.

The quality of the samples drop as the step size is increased from 0.005, as shown by the diagnostics. Results for step sizes of 0.07 and 0.08 were not reported as in these chains, no samples were accepted. It is also visible that the well-behaving step sizes produced acceptance probabilities over 0.7 and ESS values over 2000 (with a burn-in of 50). This illustrates that very high acceptance probabilities (e.g. above 0.8) can lead to good performance according to these most common metrics. This is contrary to the behavior of simpler MCMC methods, where too high acceptance results in poor exploration of the posterior and very low ESS values.

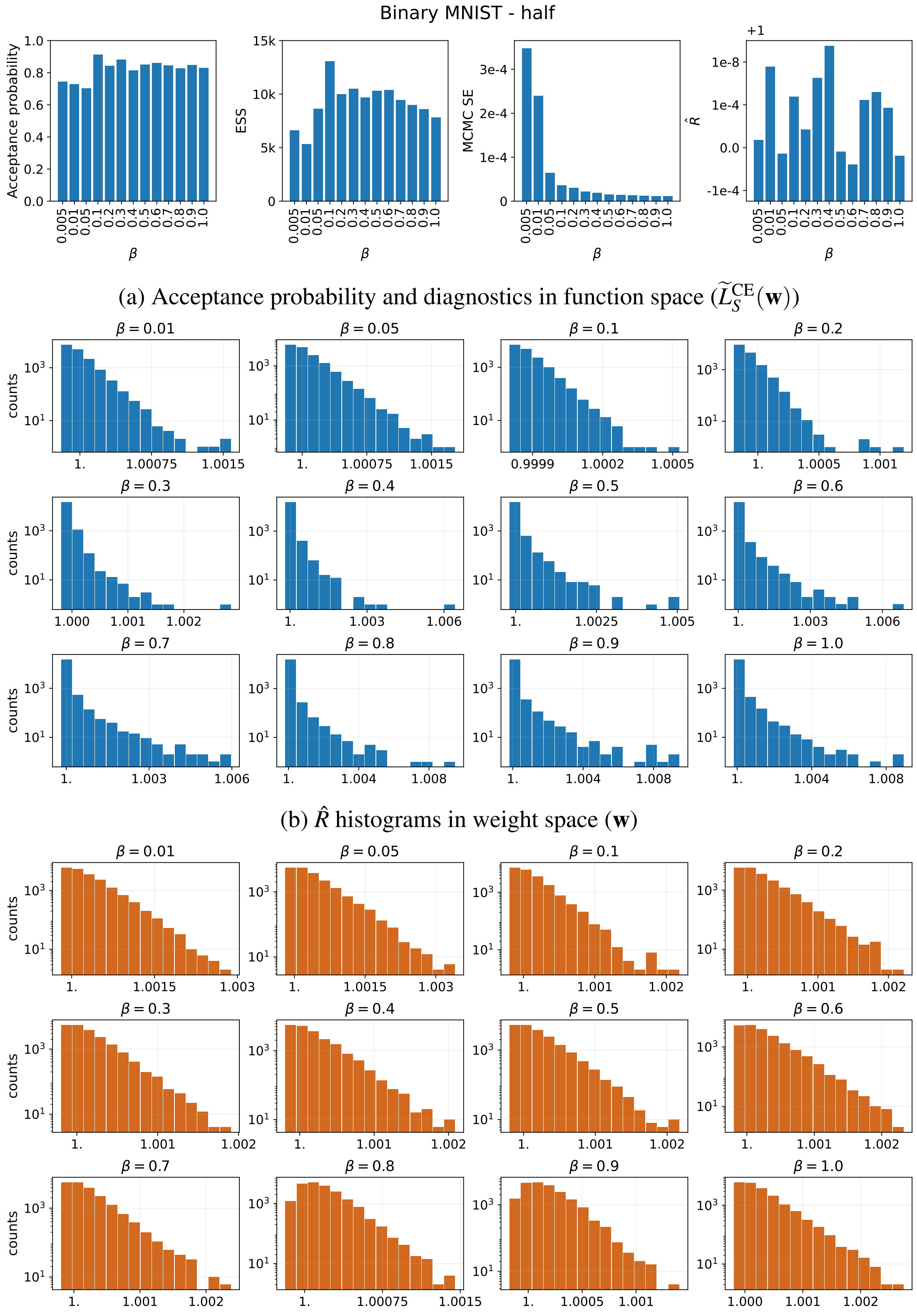
4.4 HMC Diagnostics

This section presents diagnostic results for all the HMC samples that will be used to compute the PAC-Bayes bound. Figures 4.7, 4.8, 4.9 4.10, 4.11 and 4.12 show our diagnostics for each dataset, respectively. Each figure is structured as follows. The top row (a) shows the acceptance probabilities of the Metropolis-Hastings step of HMC. We also report ESS, MCMC SE and $\hat{R}$ statistics, which as computed in function space, for $\tilde{L}_S^{\mathrm{CE}}(\mathbf{w})$. This is because we estimate expectations in function space. Values are reported for each β in our discretization, and the burn-in phase was not included. Figures (b) and (c) show histograms for $\hat{R}$ for each β. The histograms in (b) are computed in weight space, while in (c), $\hat{R}$ is evaluated on the test softmax predictions of the samples.

We make the following observations. The target acceptance probability of 0.65 was not always achieved on average. On Binary MNIST, acceptance probabilities were higher, around $70-80\%$. In these cases, increasing the step-size resulted in unstable behaviour, with an acceptance probability close to 0. We have observed the same phenomenon in our introductory experiments on HMC learning rates (Figure 4.6). We followed the usual recommendation for these cases, which is to keep the step-size slightly lower than the highest stable value [Betancourt, 2017]. On 14×14 MNIST, acceptance probabilities were roughly as desired. However, the MNIST experiments resulted in some acceptance probabilities below 0.4. In these cases, the ideal step-size is likely smaller. However, for MNIST, decreasing the step-size would have meant the largest increase in running time: from 5 to 12 hours for a single chain (5000 samples) for $\beta = 1$. Furthermore, running short chains with smaller step-sizes showed very similar values for the train and test losses. Hence a decision was made to use larger step-sizes to be able to produce a longer chain.

The ESS values are relatively high, almost always retaining at least 5000 out of the 19800 samples (burn-in removed) across the four chains. The MCMC SE values are also low. The sudden decrease as β is increased is explained by the fact that the true variance of the Gibbs posterior decreases as β increases, since the posterior becomes more concentrated around the minima of the loss landscape. All $\hat{R}$ statistics are very small, showing that our chains are approximately stationary. This finding motivates our use of the kl inversion bound (Theorem 2.2.3) which required an identically distributed sample.

Following Izmailov et al. [2021], in Figure 4.13, we supply expected calibration error (ECE) estimates averaged over all estimates for a given β. ECE measures model calibration by quantifying how well a model's output pseudo-probabilities match the true (observed) probabilities [Naeini et al., 2015]. We observe low ECE values implying that our samples correspond to well-calibrated models.

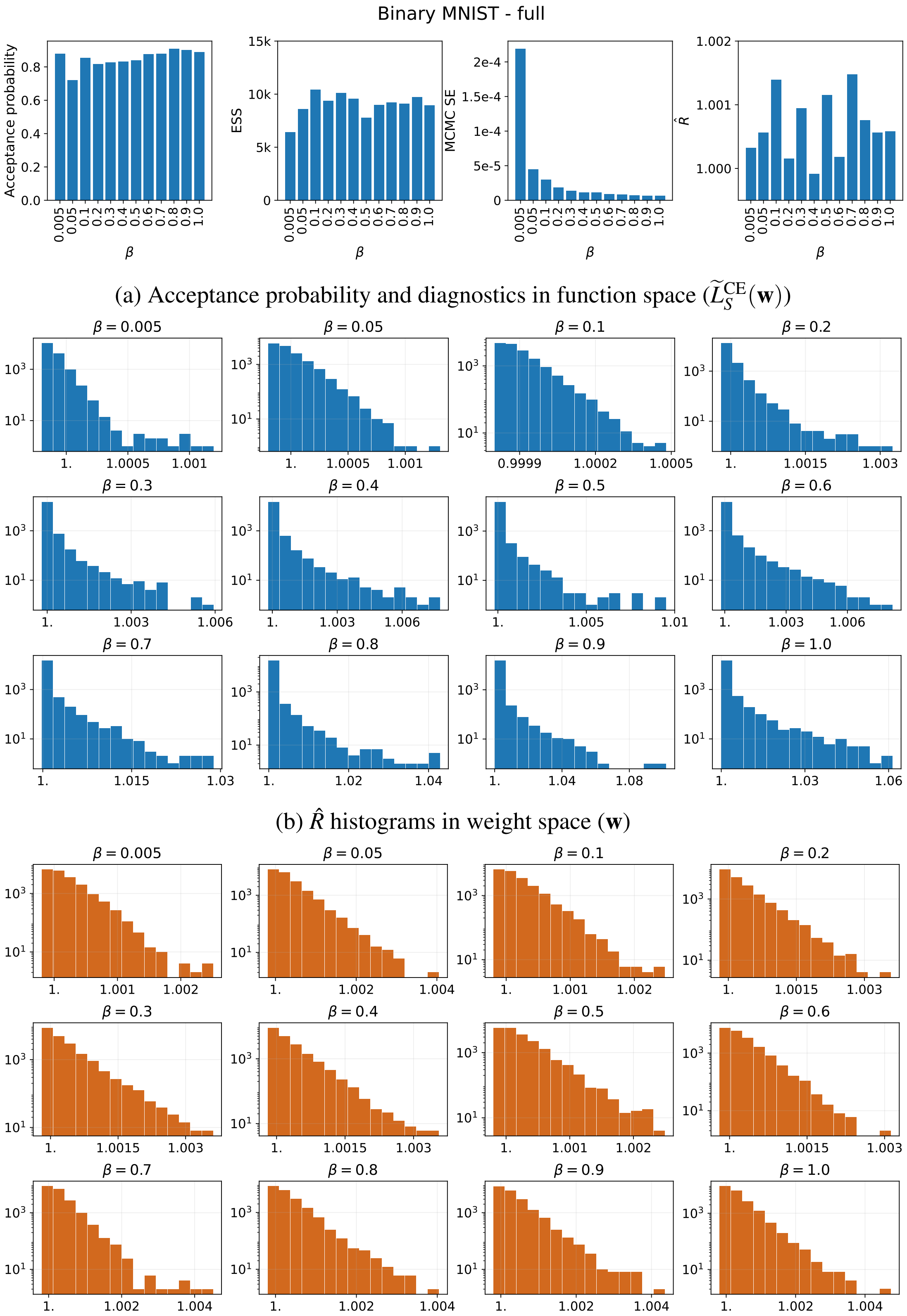


(a) Acceptance probability and diagnostics in function space ($\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$)

(b) $\hat{R}$ histograms in weight space ($\mathbf{w}$)

(c) $\hat{R}$ histograms in function space (evaluated on the test softmax predictions).

Fig. 4.7 HMC diagnostic plots for the Binary MNIST - Half dataset.

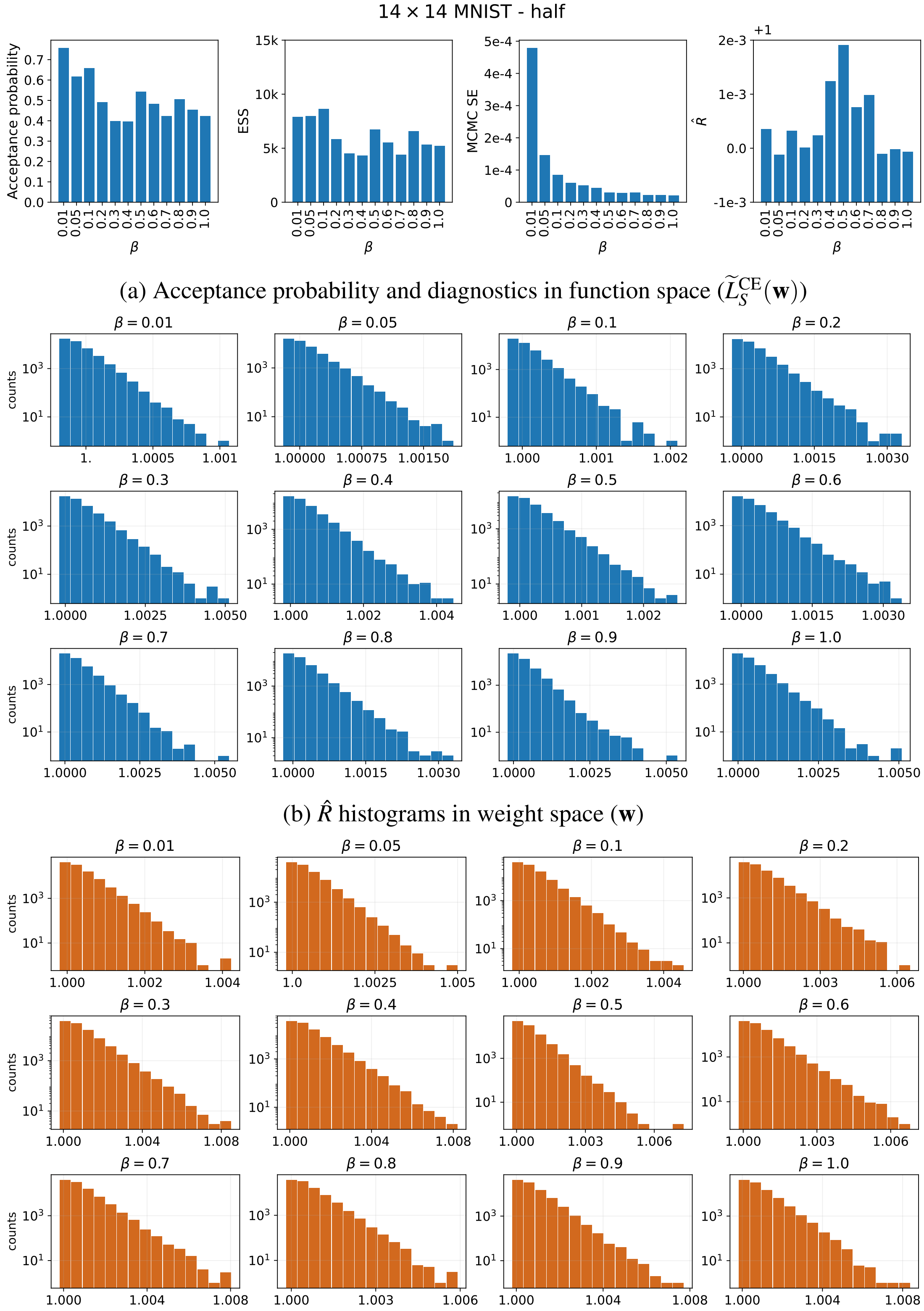


(a) Acceptance probability and diagnostics in function space ($\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$)

(b) $\hat{R}$ histograms in weight space ($\mathbf{w}$)

(c) $\hat{R}$ histograms in function space (evaluated on the test softmax predictions).

Fig. 4.8 HMC diagnostic plots for the Binary MNIST - Full dataset.

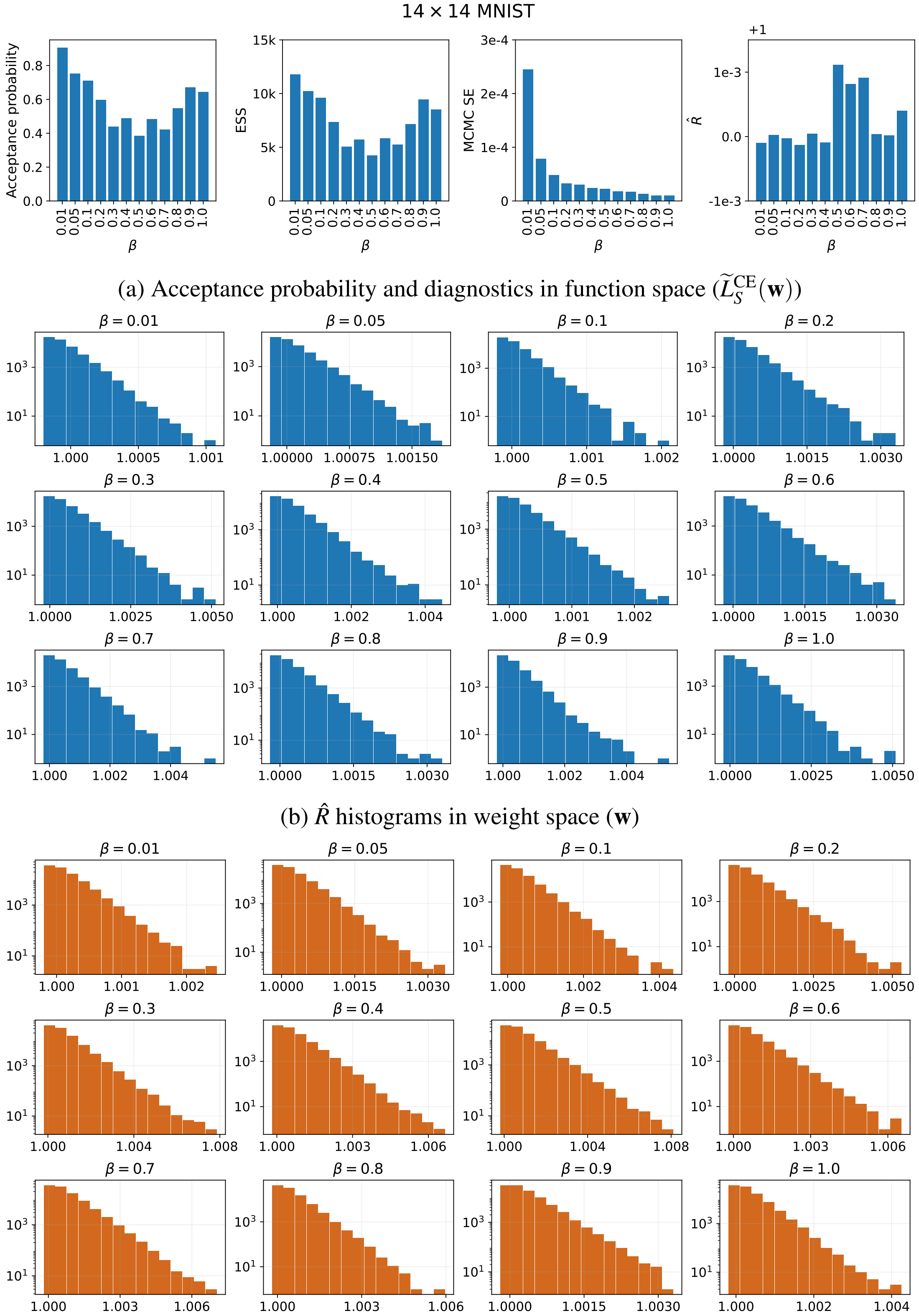


(a) Acceptance probability and diagnostics in function space ($\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$)

(b) $\hat{R}$ histograms in weight space ($\mathbf{w}$)

(c) $\hat{R}$ histograms in function space (evaluated on the test softmax predictions).

Fig. 4.9 HMC diagnostic plots for the 14 × 14 MNIST - Half dataset.

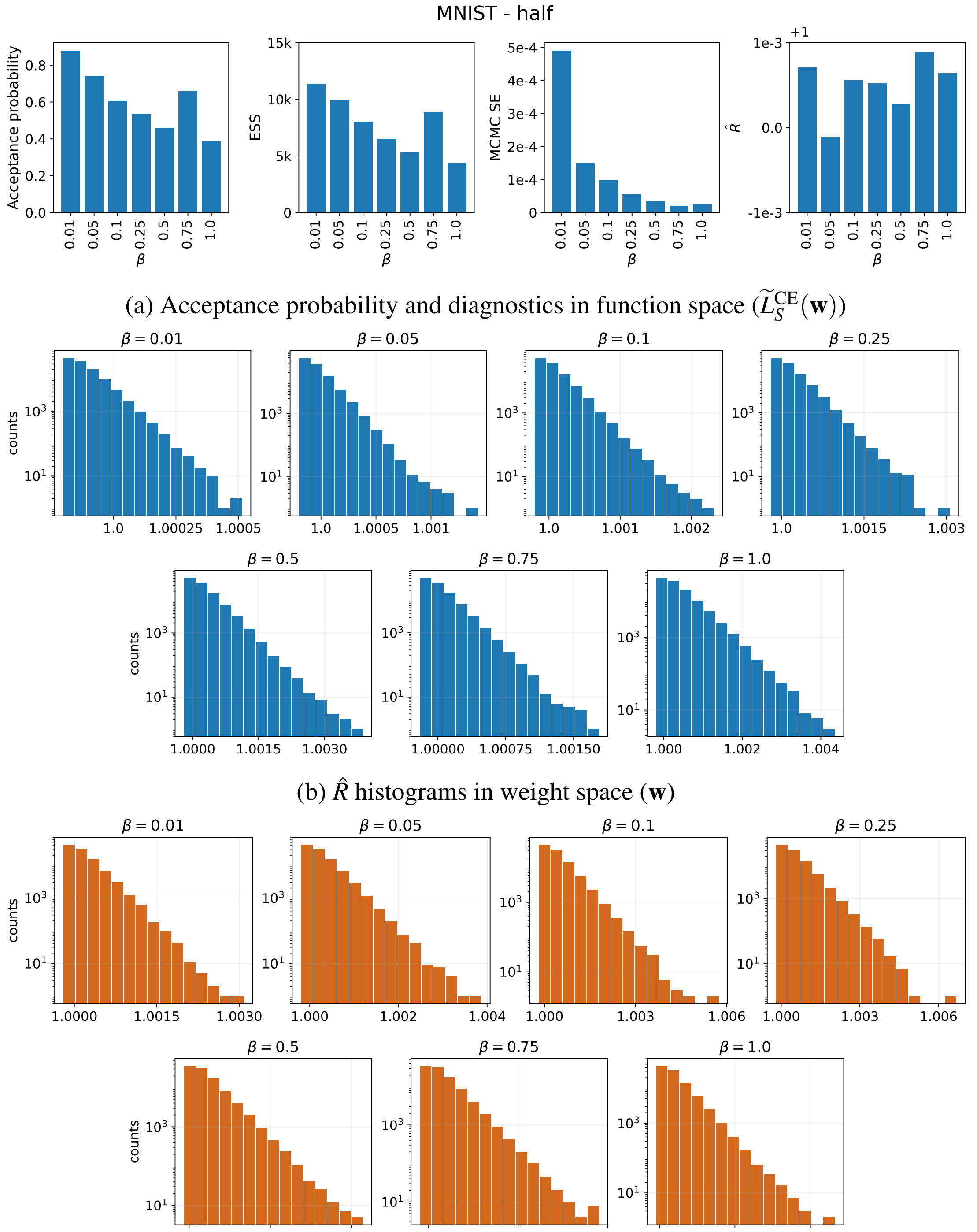


(a) Acceptance probability and diagnostics in function space ($\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$)

(b) $\hat{R}$ histograms in weight space ($\mathbf{w}$)

(c) $\hat{R}$ histograms in function space (evaluated on the test softmax predictions).

Fig. 4.10 HMC diagnostic plots for the 14 × 14 MNIST - Full dataset.

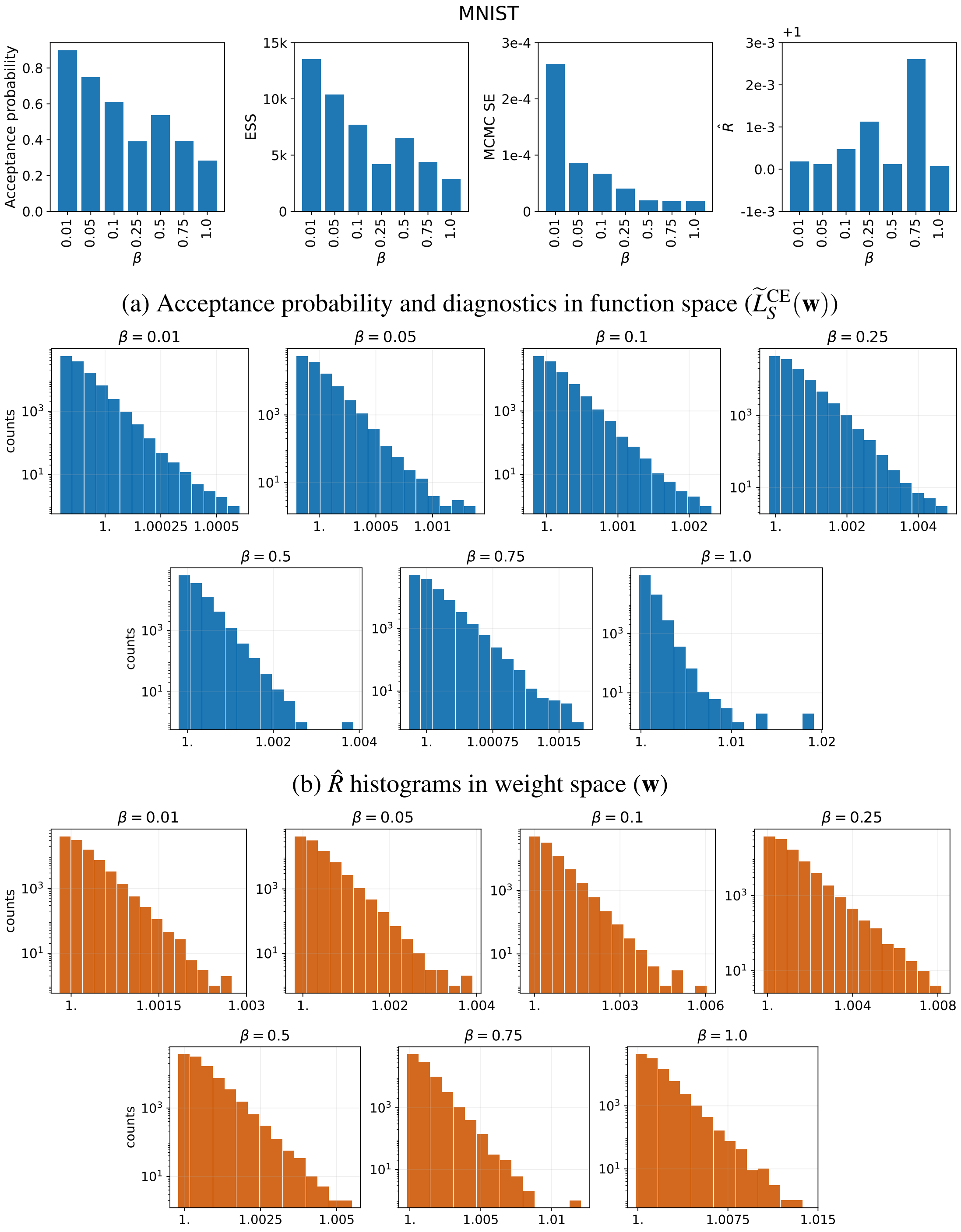


(a) Acceptance probability and diagnostics in function space ($\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$)

(b) $\hat{R}$ histograms in weight space ($\mathbf{w}$)

(c) $\hat{R}$ histograms in function space (evaluated on the test softmax predictions).

Fig. 4.11 HMC diagnostic plots for the MNIST - Half dataset.



(a) Acceptance probability and diagnostics in function space ($\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})$)

(b) $\hat{R}$ histograms in weight space ($\mathbf{w}$)

(c) $\hat{R}$ histograms in function space (evaluated on the test softmax predictions).

Fig. 4.12 HMC diagnostic plots for the MNIST - Full dataset.

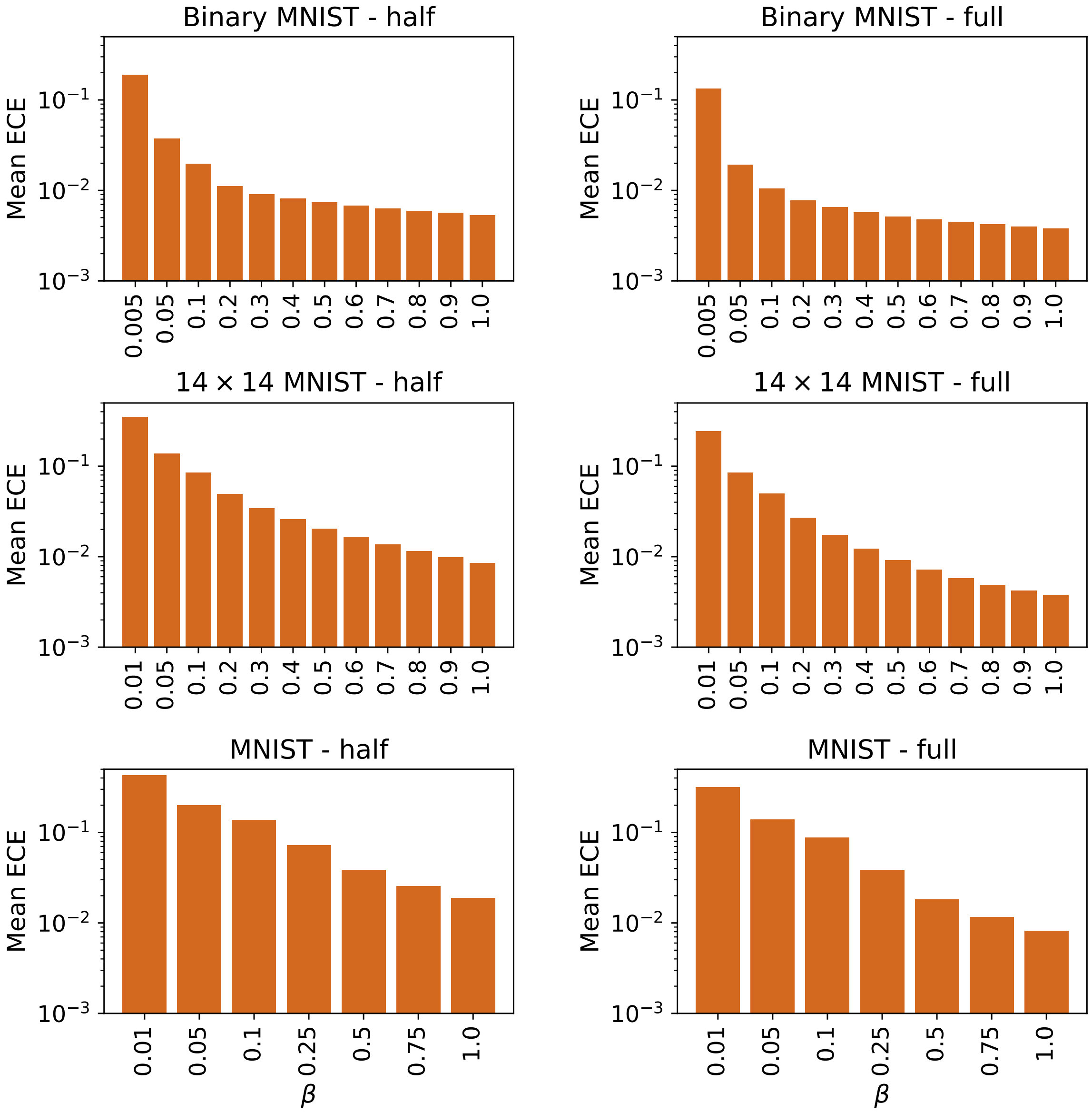


Fig. 4.13 Mean expected calibration error (ECE) values for each $\beta \in [0,1]$. The ECEs were averaged over all samples from all four chains.

Setup			Train stats			Test stats		RC with rel. entropy bound		RC with λ bound		
Method	Dataset	Model	$\widetilde{\text{CE}}$	0-1	KL/n	$\widetilde{\text{CE}}$	0-1	$\widetilde{\text{CE}}$	0-1	$\widetilde{\text{CE}}$	0-1	λ
Gibbs p.	Binary Half	1L	0.0166	0.0562	0.0205	0.0166	0.0561	0.0706	**0.1342**	**0.0915**	**0.1820**	1.0
MFVI	Binary Half	1L	0.0265	0.0924	0.0106	0.0281	0.0982	**0.0703**	0.1600	0.0934	0.2385	0.99
Gibbs p.	Binary	1L	0.0122	0.0404	0.0195	0.0125	0.0415	**0.0598**	**0.1080**	**0.0777**	**0.1435**	1.0
MFVI	Binary	1L	0.0270	0.0960	0.0105	0.0272	0.0928	0.0707	0.1640	0.0941	0.2452	0.99
Gibbs p.	14×14 Half	2L	0.0335	0.0954	0.0477	0.0361	0.1010	0.1410	**0.2376**	0.1888	**0.3273**	1.0
MFVI	14×14 Half	2L	0.0481	0.1348	0.0148	0.0481	0.1319	**0.1238**	0.2449	**0.1710**	0.3595	0.99
Gibbs p.	14×14	2L	0.0246	0.0695	0.0381	0.0259	0.0723	**0.1118**	**0.1855**	**0.1484**	**0.2507**	1.0
MFVI	14×14	2L	0.0463	0.1389	0.0140	0.0460	0.1313	0.1194	0.2379	0.1631	0.3597	0.98
Gibbs p.	MNIST Half	2L	0.0324	0.0898	0.0428	0.0365	0.0970	0.1347	**0.2248**	0.1792	**0.3091**	1.0
MFVI	MNIST Half	2L	0.0430	0.1256	0.0199	0.0437	0.1264	**0.1176**	0.2302	**0.1570**	0.3387	0.99
Gibbs p.	MNIST	2L	0.0233	0.0653	0.0334	0.0253	0.0691	**0.1065**	**0.1759**	**0.1401**	**0.2381**	1.0
MFVI	MNIST	2L	0.0423	0.1236	0.0196	0.0419	0.1200	0.1172	0.2287	0.1556	0.3342	0.99

Table 4.3 Training and test set metrics and risk certificate estimates using the kl inversion bound (Theorem 2.2.3) to ensure risk certificates hold with probability at least 0.95. Train loss refers to the transformed cross-entropy loss ($\widetilde{\text{CE}}$). Risk certificates are compared for the approximate Gibbs posterior samples and MFVI for all datasets. **Bold** numbers indicate the tighter certificate out of the Gibbs and MFVI ones for the same dataset. Models in the three dataset categories have $15742, 43018$, and 118282 parameters, respectively (twice this many for MFVI).

4.5 Bound Calculation Results

In this section, we present and discuss the results of our risk certificate calculation. Table 4.3 contains training and test statistics, as well as RCs computed using the PAC-Bayes relative entropy and λ bounds. The 0.95 probability is enforced using the kl inverse bound (Theorem 2.2.3). To ensure approximate independence, the bounds in this table were calculated on a thinned version of the Gibbs samples, which ensured that the ESS of the thinned sample is close to the remaining sample size. This resulted in retaining 1000-3000 samples out of 19800, depending on the dataset. For comparison, MFVI was also evaluated using the same amount of (exact) Gaussian samples.

We first observe that there are no bound violations, i.e. the test loss/error is always smaller than the corresponding RC. Further, the RCs for MFVI and the Gibbs posterior have the same magnitude. This suggests that there are no apparent problems with our estimates. Next, we compare our RCs to RCs obtained with different high-probability bounds from Section 3.6. This allows us to further assess the plausibility of our results.

Setup			RC with rel. entropy bound				RC with λ bound			
Method	Dataset	Model	kl inverse		asymptotic		kl inverse		asymptotic	
			$\widetilde{\text{CE}}$	0-1	$\widetilde{\text{CE}}$	0-1	$\widetilde{\text{CE}}$	0-1	$\widetilde{\text{CE}}$	0-1
Gibbs p.	Binary Half	1L	0.0706	**0.1342**	**0.0511**	**0.1065**	**0.0915**	**0.1820**	**0.0663**	**0.1428**
MFVI	Binary Half	1L	**0.0703**	0.1600	0.0581	0.1426	0.0934	0.2385	0.0756	0.2090
Gibbs p.	Binary	1L	**0.0598**	**0.1080**	**0.0315**	**0.0702**	**0.0777**	**0.1435**	0.0402	**0.0969**
MFVI	Binary	1L	0.0707	0.1640	0.0580	0.1452	0.0941	0.2452	**0.0756**	0.2136
Gibbs p.	14×14 Half	2L	0.1410	**0.1888**	**0.0946**	**0.1805**	0.1888	**0.3273**	**0.1237**	**0.2481**
MFVI	14×14 Half	2L	**0.1238**	0.2449	0.0947	0.2059	**0.1710**	0.3595	0.1263	0.3070
Gibbs p.	14×14	2L	**0.1118**	**0.1855**	**0.0702**	**0.1335**	**0.1484**	**0.2507**	**0.0910**	**0.1810**
MFVI	14×14	2L	0.1196	0.2379	0.0906	0.1991	0.1631	0.3597	0.1194	0.2930
Gibbs p.	MNIST Half	2L	0.1347	**0.2248**	**0.0935**	**0.1740**	0.1792	**0.3091**	**0.1223**	**0.2377**
MFVI	MNIST Half	2L	**0.1176**	0.2302	0.0966	0.2025	**0.1570**	0.3387	0.1263	0.2911
Gibbs p.	MNIST	2L	0.1065	**0.1759**	**0.0673**	**0.1269**	0.1401	**0.2381**	**0.0872**	**0.1714**
MFVI	MNIST	2L	**0.1010**	0.2070	0.0947	0.1987	**0.1317**	0.2977	0.1226	0.2822

Table 4.4 Comparison of risk certificates using the kl inversion bound (Theorem 2.2.3) and the asymptotic bound [Rosenthal, 2017]. Train loss refers to the transformed cross-entropy loss ($\widetilde{\text{CE}}$). Risk certificates are estimated to hold with probability at least 0.95. Risk certificates are compared for the approximate Gibbs posterior samples and MFVI for all datasets. **Bold** numbers indicate the tighter certificate out of the Gibbs and MFVI ones for the same dataset. Models in the three dataset categories have $15742, 43018$, and 118282 parameters, respectively (twice this many for MFVI).

Dataset	RC with rel. entropy bound	RC with λ bound
Binary Half	0.1417	0.1877
Binary	0.1184	0.1566
14×14 Half	0.2324	0.3180
14×14	0.1920	0.2600
MNIST Half	0.2247	0.3068
MNIST	0.1880	0.2553

Table 4.5 Risk certificates for the $0-1$ loss using the approximate Gibbs posterior samples. RCs were calculated using the "sanity check" bound with mild assumptions and were calibrated to hold with probability at least 0.95. Only $0-1$ RC-s are reported since it does not make sense to calculate this bound for cross-entropy loss.

4.5.1 Comparison of High-probability Bounds

In Table 4.4, we compare RCs computed with two different confidence intervals needed to guarantee that our RCs hold with probability at least 0.95. The kl inversion bound (Theorem 4.3) and the asymptotic bound (from Rosenthal [2017]) is compared using both MFVI and (approximate) Gibbs posterior samples. For the asymptotic bound, 19800 samples were used since this does not require independence. The asymptotic bound gives lower risk certificates, which is explained by the fact that (i) more samples were used, and (ii) that this interval only guarantees a probability of 0.95 as the number of samples tends to infinity.

For readers skeptical about assumptions, Table 4.5 contains $0-1$ loss RCs for the underlying distribution our HMC samples come from, Q (not necessarily the Gibbs posterior), requiring the sole assumption that $D_{\mathrm{KL}}(Q||Q_\lambda^*) \leq D_{\mathrm{KL}}(G||Q_\lambda^*) + \mathbb{E}_Q\left[n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$ to estimate the KL. We use the kl inversion bound to upper bound our estimate for the risk. We do not show these RCs for the cross-entropy loss as these are not smaller than the corresponding MFVI RCs, by construction. We can see that these RCs almost always fall within 1% of the kl inverse bounds. Hence our kl inverse RCs are likely not lower bounds of the true 95% confidence bound. Our asymptotic estimates are always significantly smaller. This sanity check shows strong support for our kl inverse bounds, hence in the rest of this section, we use these bounds as RC estimates.

Having supported their validity, our estimates allow us to assess hypotheses on RCs and their variational approximations. We interpret our results in Question-Answer (Q & A) style.

4.5.2 Interpretation of Results - Q & A

Q1: How much better is the Gibbs posterior than MFVI?

A1: Interestingly, the computed RCs for $\widetilde{L}_S^{\mathrm{CE}}(Q)$ ($\widetilde{\mathrm{CE}}$) are almost always better for MFVI than for the approximate Gibbs posterior samples, despite the Gibbs posterior being the minimizer of the λ bound. However, the RC-s on the $0-1$ losses show considerable improvement with respect to MFVI, especially for Binary MNIST. This discrepancy in cross-entropy and $0-1$ losses may indicate that we have considerably overestimated the true KL divergence term. Since our KL divergences are small compared to the $0-1$ losses, they affect the $0-1$ RCs less. We are mainly interested in $0-1$ RCs, hence this is not a problem for us.

Model	#params	Train stats			Test stats		Best RC	
		$\widetilde{\text{CE}}$	0-1	KL/n	$\widetilde{\text{CE}}$	0-1	$\widetilde{\text{CE}}$	0-1
Perez-Ortiz et al. [2020], MLP	2×1.2M	0.0277	0.0951	0.1383	0.0268	0.0921	0.2033	0.3155
Perez-Ortiz et al. [2020], CNN	2×1.2M	0.0157	0.0535	0.1039	0.0143	0.0513	0.1453	0.2165
ours (MFVI), MLP	2×118k	0.0419	0.1236	0.0196	0.0419	0.1200	0.1172	0.2287
ours (Gibbs p.), MLP	118k	0.0233	0.0653	0.0334	0.0253	0.0691	0.1065	0.1759

Table 4.6 Comparison of our (full) MNIST RC results to Perez-Ortiz et al. [2020]. Full MNIST classification is the only shared task. Please note that Perez-Ortiz et al. [2020] uses $200,000$ samples from the approximate posterior, while we used ≈ 2000 to ensure that the Gibbs samples are approximately independent. The PAC-Bayes bound used for RC computation was the PAC-Bayes relative entropy bound.

Q2: How does the tightness gap change as the data size and /or model size is increased?

A2: The gap between the $0-1$ RCs for the Gibbs posterior and MFVI increases as the amount of train data is increased. MFVI gives similar RCs in both cases while Gibbs RCs improve by $3-7\%$ when the amount of data is increased. This is reasonable since Gibbs posteriors concentrate more around minima of the loss as n is increased. However, our results show that MFVI is not able to utilize the extra data very well.

When increasing model size, the gap between the $0-1$ RCs decreases as we move from Binary MNIST to 14×14 MNIST and remains roughly the same as we move to MNIST. This supports the hypothesis of variational approximations become more accurate as model depth is increased.

Q3: Does the size of the architecture affect the RC on a fixed dataset?

A3: We compare our results to Perez-Ortiz et al. [2020] in Table 4.6. Surprisingly, even with much fewer samples (≈ 2000 compared to $200,000$) we obtain better RCs both for cross-entropy and $0-1$ loss. The only difference in the MFVI experiments between Perez-Ortiz et al. [2020] and us is the size of the architecture: our MLPs have a magnitude fewer parameters. The main contributor to the difference in RCs is the KL divergence per datapoint term, which is higher in Perez-Ortiz et al. [2020]. A possible explanation for this phenomenon is as follows. Since the number of datapoints is given, a smaller architecture allows for more variation per parameter while keeping the same KL/n value and hence can explore better trade-offs between loss and KL terms. Hence it can be beneficial to consider small architectures.

Q4: How good can data-independent bounds be?

A4: Our results show that data-independent bounds may hold more promise than what most of the current literature assumes. Although the data-independent prior does constrain the bound through $D_{\mathrm{KL}}(Q||P)$, they can be made smaller than as reported in previous literature: our improvement was 17.6% (with the Gibbs posterior) error compared to 21.7% in Perez-Ortiz et al. [2020]. It is likely that better-chosen architectures could achieve even better RCs. As Lotfi et al. [2022] suggests, bounds based on data-independent priors explain generalization better, hence their study should not be neglected.

Summary. In this chapter, we presented our RC estimates for the Gibbs posterior. As preparation, we investigated properties of the Gibbs posterior and HMC via small experiments. We then assessed our HMC samples using various diagnostics. We argued for the validity of our RC estimates by comparing to RCs obtained with other high-probability bounds. We then used our RC estimates to discuss hypotheses in PAC-Bayes. Our Gibbs RC estimate always improved on MFVI in $0-1$ loss. The tightness gap between MFVI and Gibbs RCs was found largest for Binary MNIST - Full. In general, it was found that the tightness gap increases with dataset size and decreases as network depth is increased. Finally, comparing to Perez-Ortiz et al. [2020], we demonstrated the benefits of using small architectures and argued for data-independent priors.

Chapter 5

Related Works

Chapter Roadmap

Having presented our results in the previous chapter, we now discuss works related to each aspect of our method. We first explain how our topic fits into the current PAC-Bayes literature, before turning to the main threads of related work: (i) tightness PAC-Bayes bounds, (ii) evaluating MFVI, (iii) MCMC for neural networks and (iv) literature on high-probability bounds. For a rough categorization of most of the mentioned works, see Table 5.1.

5.1 Recent Developments and Trends in PAC-Bayes

Since the establishment of non-vacuous PAC-Bayes bounds for over-parametrized neural networks [Dziugaite and Roy, 2017], the main focus of the literature has been to (empirically) achieve the tightest bounds possible. To this end, there has been a lot of work on data-dependent priors, which are considered to be the only route to produce tight bounds [Dziugaite and Roy, 2018, Dziugaite et al., 2021]. Currently, the tightest empirical bounds exist for compressed models [Lotfi et al., 2022], and were achieved using a bound that upper-bounds the KL divergence with the compressed size of the model [Zhou et al., 2019]. Table 5.2 shows the current state-of-the-art on typical deep learning datasets.

Another thread of current research is concerned with extending the applicability of PAC-Bayes beyond its usual assumptions and to a variety of models. The two key assumptions of the PAC-Bayes framework are a bounded risk (loss) function and i.i.d. data. To relax the former, PAC-Bayes bounds have been developed for unbounded losses [Haddouche et al., 2021] and heavy-tailed losses [Haddouche and Guedj, 2023a, Rodriguez Gálvez et al., 2023]. There also exist some bounds for non i.i.d. data [Ralaivola et al., 2009]. Recently, PAC-Bayes bounds have been formulated in terms of the Wasserstein distance (instead of the

	PAC-Bayes	Bayesian Deep Learning
MCMC	Dziugaite and Roy [2018] (SGLD)	Izmailov et al. [2021]
comparison	Alquier et al. [2016] Our work	Foong et al. [2020] Farquhar et al. [2020]
MFVI	Pitas [2020] Perez-Ortiz et al. [2020]	Most BNN papers

Table 5.1 Comparison of recent works in the PAC-Bayes and Bayesian Deep Learning communities, focusing on MFVI, MCMC and their comparison. Blue shows mainly theoretical works, red shows mainly empirical works, while green shows work with significant theoretical and empirical components.

	MNIST	CIFAR-10	CIFAR-100	ImageNet
Data-independent	11.6	58.2	94.6	93.5
Data-dependent	1.4	16.6	44.4	40.9

Table 5.2 State of the art in empirical PAC-Bayes bounds. All bounds were obtained for compressed models, in Lotfi et al. [2022].

KL divergence), which is easier to estimate [Haddouche and Guedj, 2023b, Viallard et al., 2023]. Finally, PAC-Bayes bounds have been developed recently to provide reconstruction guarantees for generative models, such as VAEs [Chérief-Abdellatif et al., 2022] and GANs [Mbacke et al., 2023].

In comparison, our work goes back to considering smaller neural network models on MNIST and its simplified versions only. This is necessary to be able to (approximately) sample from the Gibbs posterior and for (generalized) marginal likelihood estimation. Our main interest is to empirically evaluate the accuracy of MFVI compared to the optimal bound, and not to obtain even tighter generalization bounds. However, as a byproduct, we do obtained tighter risk certificates than those of MFVI on the small models considered.

5.2 Tightness of PAC-Bayes Bounds

Perez-Ortiz et al. [2020] is a recent contribution to achieving tight empirical PAC-Bayes bounds in a general setting (i.e. non-compressed models). The authors compare MFVI on several training objectives derived from PAC-Bayes bounds, including the PAC-Bayes λ bound (Equation 3.2). As explained in Section 2.2.5, we follow their approach in optimizing

PAC-Bayes objective for MFVI.

Foong et al. [2021] shares our interest in the tightness limits of PAC-Bayes. However, they consider the small data regime (30-60 datapoints), arguing that this regime benefits most from eliminating the need for test data with a PAC-Bayes risk certificate. Their setting also differs from ours in that they study the tightness limits of the whole PAC-Bayes framework, not those of a specific bound. They do this by considering a theorem from Germain et al. [2009], that unites many PAC-Bayes bounds in a general form.

Theorem 5.2.1. *(Generic PAC-Bayes theorem) Fix $P \in \mathcal{M}_1(\mathcal{W})$, $l \in [0,1]$, $\delta \in (0,1)$ and Δ a proper*[1]*, convex and lower semicontinuous*[2] *function $\Delta : [0,1]^2 \to \mathbb{R} \cup \{+\infty\}$. Then,*

$$\mathrm{P}\left((\forall Q)\ \Delta(\hat{L}_S(Q), L(Q)) \leq \frac{1}{n} \left[D_{\mathrm{KL}}(Q||P) + \log \frac{\mathcal{I}_\Delta(n)}{\delta} \right] \right) \geq 1 - \delta, \tag{5.1}$$

where $\mathcal{I}_\Delta(n) := \sup_{r \in [0,1]} \sum_{k=0}^{n} \binom{n}{k} r^k (1-r)^{n-k} e^{n\Delta(k/n,r)}$.

We note that our PAC-Bayes relative entropy bound (Equation 2.6) is almost equivalent to this with $\Delta(q,p) = \mathrm{kl}(q,p)$, and the difference is a slight loosening with an upper bound on $\mathcal{I}_\Delta(n)$.

This generic bound can be optimized in the function Δ, giving us a way to compare many PAC-Bayes bounds and also characterize the limits of this general form of PAC-Bayes bounds. Among their findings, the most relevant to us is the following. When Δ is chosen independently of the dataset, they lower bound the tightest possible bound in Δ and find that this lower bound coincides with the PAC-Bayes relative entropy bound *without* the $\log(2\sqrt{n})$ term.

This finding supports our choice to use the PAC-Bayes relative entropy bound to compute our risk certificates from the Gibbs posterior samples. We note, however, that the tightness of a PAC-Bayes bound in general does not guarantee that the bound can be used efficiently for empirical risk certificate computation. In the context of exact minimization, the optimizing posterior may not exist or have a suitable form, and for MFVI, the bound may not be a good objective for optimization. This is why, most of the time, linear PAC-Bayes bounds are used as PAC-Bayes objectives.

The authors also compare PAC-Bayes bounds to test set bounds in a synthetic 1D classification experiment and find that their PAC-Bayes cannot be made competitive with

[1] A proper convex function Δ is a convex function on $\mathbb{R} \cup \{\infty\}$ with a non-empty domain X such that $f(x) < \infty$ for some $x \in X$.

[2] Δ is lower semicontinuous if $\Delta(x) \leq \liminf_{k \to \infty} \Delta(x_k)$ for all sequences $x_k \to x$ [Bertsekas, 2009].

their best test set bound. This shows that we have more powerful techniques in the small data regime, hence PAC-Bayes should be used for larger datasets.

5.3 Evaluating MFVI

This section collects works that aim to evaluate MFVI, both in the Bayesian deep learning and PAC-Bayes communities.

5.3.1 In Bayesian Deep Learning

A debate on the accuracy of MFVI arose from the works of Foong et al. [2020] and Farquhar et al. [2020]. Both study the performance of MFVI in function space, that is, the performance of the approximate posterior predictive. This setting is relevant for us since PAC-Bayes bounds involve functions of $\hat{L}_S(Q)$ and $D_{\mathrm{KL}}(Q||P)$. The former is an object in function space, and the latter, in itself, has nothing to do with the true posterior. Hence an approximate posterior Q that is far from the true posterior but performs well in function space could in principle have a tight risk certificate.

Foong et al. [2020] study single-hidden-layer neural networks and identify the following limitation of mean-field Gaussian approximate posteriors in function space. Their predictive cannot represent *in-between uncertainty*, that is, higher uncertainty between well-separated regions of low uncertainty. This is because the variance on any line segment in the input space is bounded by the maximum of the variance at its endpoints. However, they show empirically that exact inference does not suffer from this problem, showing that MFVI fails to capture this property. For deeper BNNs of at least two hidden layers, they prove a universality result for their mean and variance functions but find experimentally (using small BNNs of multiple hidden layers) that optimization does not necessarily discover these sets of optimal parameters.

Farquhar et al. [2020], in contrast, study deep neural networks of at least two hidden layers. In this setting, they provide a stronger universality result than in Foong et al. [2020], proving the closeness of the MFVI predictive to the true predictive in distribution, rather than in the first two moments as in Foong et al. [2020]. In their experiments on the “two moons” dataset, they compare full-covariance approximate posteriors to mean-field ones via (i) comparing their Wasserstein distance from the *true posterior*, computed using HMC samples and (ii) comparing their KL divergence directly. They find that errors quickly diminish as depth is increased, with six layers being enough for negligible error. Their larger-scale experiments on CIFAR-100 and ImageNet using structured covariances [Maddox et al., 2019] and MFVI also suggest that MFVI performs well in deep models.

The above articles together identify the limitations of MFVI as a function of the data size and model depth. The picture seems simple: for small datasets and shallow networks, MFVI approximates the true posterior poorly, but performs well in deep networks.

The above is relevant to our context in the following way. We study small neural networks (one or two hidden layers), where, by the above, MFVI is less accurate[3]. Thus, we can "win" more by using the optimal PAC-Bayes posterior. Interestingly, based on the universality results, the "line" between poor and good performance seems to be at two hidden layers. This is in accordance with our results: we also observed a drop in the tightness gap as model depth was increased. However, we used separate datasets for the different depths, hence verifying this hypothesis in our setting requires further experiments.

5.3.2 In PAC-Bayes

We now discuss related works assessing the performance of MFVI in providing approximate posteriors with tight generalization guarantees.

Alquier et al. [2016] is the closest to our work. They study the concentration of Gibbs posteriors and of their Gaussian approximations. Recall that the goal in statistical learning theory is to find the best possible predictor, which is the one that satisfies $\bar{\mathbf{w}} = \arg\inf_{\mathbf{w}\in\mathcal{W}} \mathbb{E}_{z\sim D}[L(\mathbf{w}, z)]$. Let us denote $L(\bar{\mathbf{w}})$ by $\bar{L}$. When we form a randomized predictor (e.g. any posterior Q) after observing a data sample S, we want this randomized predictor to converge to a distribution that concentrates around $\bar{L}$ with low variance (as the number of data samples tends to infinity). The contribution of Alquier et al. [2016] is proving in certain settings that the best Gaussian variational approximation to the Gibbs posterior, G^*_λ has the same rate of convergence to $\bar{L}$ as the Gibbs posterior.

They achieve this using *oracle* bounds (briefly discussed in Section 2.2.5) on the true risks $L(Q^*_\lambda)$ and $L(G^*_\lambda)$ of the optimal Gibbs posterior Q^*_λ and its best variational approximation G^*_λ, respectively. Their key result is a characterization of the difference between the two upper bounds. They supply a general result and then specify their theorem to common task settings of which the closest to us is binary linear classification. We only discuss our specific result and refer the reader to Alquier et al. [2016] for the assumptions required. Roughly, these require conditions on the boundedness of the risk and the data-generating distribution.

[3] Although by all means insightful, the exact theorems in the above works may not be directly applicable to our PAC-Bayes setting as our setting constrains the negative log (pseudo) likelihood to be bounded.

Let $\mathcal{F}$ denote the set of mean-field Gaussians. The authors upper bound the right-hand side of their oracle bounds and thus show that the Gibbs posterior and its variational approximation have the same rate of convergence to $\bar{L}$, given below.

Theorem 5.3.1. *Let $\bar{L} = \inf_{\mathbf{w}\in\mathcal{W}} L(\mathbf{w})$ and G^*_λ and Q^*_λ denote the optimal mean-field Gaussian approximate posterior and the Gibbs posterior, respectively, for some data sample S. Then, for any $\varepsilon > 0$, with probability at least $1-\varepsilon$, simultaneously,*

$$\left.\begin{array}{l} L(Q^*_\lambda) \\ L(G^*_\lambda) \end{array}\right\} \leq \bar{L} + \sqrt{\frac{d}{n}} \log\left(4ne^2\right) + \frac{c}{\sqrt{n}} + \sqrt{\frac{d}{4n^3}} + \frac{2\log\left(\frac{2}{\varepsilon}\right)}{\sqrt{nd}}, \tag{5.2}$$

where d is the data dimensionality and the number of model parameters, i.e. $\mathcal{X} = \mathcal{W} = \mathbb{R}^d$ (in this setup of binary linear classification).

We now discuss the implications of this result to our setting. We compute *empirical* PAC-Bayes bounds, which upper bound the true risks $L(Q^*_\lambda)$ and $L(G^*_\lambda)$. The above theorem studies how powerful the true risks are at the two different posteriors (Gibbs and Gaussian), and the finding that they are similar suggests that their corresponding empirical PAC-Bayes bounds could in principle be made comparably small. However, our setting bears the following differences:

1. We perform nonlinear classification, since we use MLPs with nonlinear activation functions. The above result is only proven for binary linear classification.

2. The mean-field Gaussian found via MFVI optimization may not be the *optimal* mean-field Gaussian, i.e. $\text{argmin}_{\text{MF Gaussians}} D_{\text{KL}}(G||Q^*_\lambda)$. In particular, the theorem requires the approximate posterior mean to be $\bar{\mathbf{w}}$.

To expand on point 2, we note that gradient-based optimizers tend to converge to local minima, hence in practice, MFVI will likely only supply a good, but non-optimal posterior mean. However, as discussed in Section 5.3.1, in the context of MFVI in Bayesian deep learning, this result may hint at the hypothesis that variational approximations to PAC-Bayes bounds become accurate as the data and/or model size is increased to infinity[4]. In our experiments, we observed a decrease of the tightness gap as model size was increased, but not when dataset size was increased.

[4] For readers interested in the implications of this in Bayesian deep learning, we note that the above theorems are only valid under bounded losses, and the negative log likelihood used in Bayesian deep learning is unbounded. For a study of the topic in the classical Bayesian setting, see Alquier and Ridgway [2020].

Pitas [2020] considers the general method of Dziugaite and Roy [2017] for computing nonvacuous PAC-Bayes bounds with MFVI. The author studies the significance of many steps in the method in achieving nonvacuity, using a bound from Catoni [2007]. The posterior mean is kept fixed at a minimum of the loss landscape. The main finding is that the effect of the optimization of the posterior covariance in MFVI is minimal compared to using an isotropic posterior covariance with a pre-specified parameter. The author argues that the key step in ensuring nonvacuity is setting the prior mean to a random initialization instead of zero (see more discussion in Section 2.2.4). The author also derives theoretical optimal values for general Gaussian covariance, which do not result in a valid bound, but (at least informally), characterize the limits of MFVI. The optimal covariance is then approximated with a K-FAC Laplace approximation, which improves the bounds significantly compared to the optimal (invalid) diagonal covariance. It is argued that this demonstrates an inherent limitation of the mean-field covariance structure.

After discussing the literature around MFVI, we now sketch the related work around our method.

5.4 MCMC for Neural Networks

In this section, we describe the MCMC algorithms that are typically used to sample from a neural network posterior.

The most important algorithm for us is HMC. HMC was first used for neural networks in Neal [1996]. More recent studies allowed us to scale up HMC sampling to large architectures [Wenzel et al., 2020, Izmailov et al., 2021]. In particular, Izmailov et al. [2021] use ResNet-20 [He et al., 2016] networks with 10^5 leapfrog steps in their large-scale experiments using hundreds of TPUs. Since HMC requires careful calibration of the stepsize and trajectory length, automatic approaches were developed, the most popular being the No-U-Turn Sampler (NUTS) [Homan and Gelman, 2014].

Two key limitations of full-batch MCMC methods, such as HMC, are that they scale poorly to (i) high-dimensional distributions and (ii) large datasets. To mitigate the high-dimensionality issue, reversible-jump MCMC methods can be used (also known as trans-dimensional MCMC) [Green, 1995]. In these algorithms, the dimension of the parameter space can vary between iterates of the Markov chain, hence we avoid sampling from very high dimensional spaces. Transdimensional algorithms have been used in for Gibbs posterior sampling in non-neural network PAC-Bayes tasks [Guedj and Alquier, 2013, Guedj and Robbiano, 2018, Li et al., 2018].

To mitigate the large datasets issue, batch MCMC methods have been developed. They introduce a trade-off between computational speed-up and accuracy. Among them are stochastic gradient-based methods, which are derived from discrete-time approximations to continuous-time diffusion processes [Nemeth and Fearnhead, 2019]. Stochastic gradient HMC uses second-order Langevin dynamics with a friction term, which mitigates the problem of noisy gradients [Chen et al., 2014]. Arguably, the most popular stuchastic gradient MCMC method is Stochastic Gradient Langevin Dynamics (SGLD) [Welling and Teh, 2011], which combines characteristics of stochastic gradient descent, Robbins–Monro optimization and Langevin dynamics.

5.5 (Generalised) Marginal Likelihood and KL Estimation

In our method, we made the choice to reduce KL estimation to generalized log marginal likelihood estimation, via Theorem 3.3.1. Hence, we now discuss alternative methods for computing the log of the normalizing constant of a Gibbs posterior[5], $Z = \mathbb{E}_{\mathbf{w}\sim P}[e^{-\lambda \hat{L}_S(\mathbf{w})}]$. Interestingly, most methods involve the tempering of the generalized likelihood, $\left(e^{-\lambda \hat{L}_S(\mathbf{w})}\right)^{\beta}$, and they require us to sample from the corresponding (tempered) Gibbs posterior.

Our used method, thermodynamic integration, directly estimates $\log Z$. It is also possible to first estimate Z, and then use the logarithm to give a high-probability lower bound on $\log Z$ using Markov's inequality, as long as $\hat{Z}$ is an unbiased estimator of Z [Grosse et al., 2015]:

$$\mathrm{P}(\log \hat{Z} > \log Z + b) < e^{-b}. \tag{5.3}$$

From this, we obtain a high-probability upper bound on $-\log Z$. Methods that take this route include annealed importance sampling (AIS), the paired-product estimator, sequential Monte Carlo, and nested sampling methods [Neal, 2001, Huber, 2015, Moral et al., 2006, Buchner, 2023]. We note the following interesting connection of the first two methods to thermodynamic integration. They all approximate the area under the integrand $\beta \mapsto \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$ for each random sample *from the function*, at $(\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_m) \sim \pi_{\beta_1} \times \pi_{\beta_2} ... \times \pi_{\beta_m}$. However, thermodynamic integration averages these random areas immediately to estimate the area below $\beta \mapsto \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda \widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$, while AIS and the paired-product estimator average the individual random areas in exponential space before taking the logarithm. AIS and the paired-product estimator differ in that AIS calculates these random areas using the left sums on the random function estimates, whereas the paired-product estimator uses the trapezium rule.

[5] Some works in the mathematics and physics literature refer to this as the log partition function.

Since each method can be used together with HMC, these techniques are all viable alternatives to our method, with the added convenience of the high-probability upper bound on $-\log Z$. In our method, we chose to give high-probability bounds for each individual expectation $\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$, so that we use the same bound for both $\hat{L}_S(Q)$ and the KL divergence term.

Finally, we note that there exist works that estimate KL divergences directly from samples, see for example Zhao and Lai [2020] and Ghimire et al. [2021].

5.6 High-probability Bounds

As mentioned, the two main categories are concentration inequalities and asymptotic bounds. In the context of PAC-Bayes, the former are more desirable but are harder to establish for MCMC methods. Concentration inequalities for general Markov chains usually involve some spectral property of the Markov chain, most often the (absolute) spectral gap [Fan et al., 2021], which quantifies the convergence speed of the chain to its stationary distribution [Rudolf, 2012]. These error bounds are usually quite complicated and involve the estimation of spectral quantities from the samples. Out of asymptotic error bounds, the most simple ones we are aware of are either based on the MCCLT, or on the weaker assumptions of Rosenthal [2017]. We used the latter method to obtain asymptotic error bounds on our estimates.

In the next, final section, we discuss a piece of past research that attempts to carry out a very similar task to ours, and compare it to our approach.

5.7 Estimating a PAC-Bayes Bound in the Gibbs Posterior

Dziugaite and Roy [2018] is the closest to our work in that they attempt to estimate a PAC-Bayes bound in a Gibbs posterior but in the context of data-dependent priors. Their goal is to use a Gaussian prior, where the mean parameter depends on the data sample. This mean parameter is chosen to be the mean of the Gibbs posterior of a simple Gaussian prior. They use SGLD to sample from this Gibbs posterior and choose the resulting mean as the mean of their Gaussian prior, which thus becomes data-dependent. They justify this by the fact that SGLD optimization results in a (mean) vector that is close to an approximately differentially private vector. This property allows them to use the resulting mean in their data-dependent prior and obtain valid bounds by slightly loosening the PAC-Bayes relative entropy bound (Theorem 2.6) to account for the approximation to differential privacy. For a background in differential privacy, please refer to Dwork [2006].

This piece of research is also the only source we are aware of, that attempts to estimate the negative log normalizing constant/marginal likelihood $-\log Z = -\log \mathbb{E}_{\mathbf{w}\sim P}[e^{\lambda \hat{L}_S(\mathbf{w})}]$ in order to compute PAC-Bayes bounds with the Gibbs posterior. They propose the simple $\log \hat{Z}$ upper-bound we discussed above in Equation 5.3. Their estimator for Z, $\hat{Z}$ is a simple Monte Carlo average using samples from their data-dependent prior P. Their bound is as follows.

$$-\log \mathbb{E}_{\mathbf{w}\sim P}\left[e^{\lambda \hat{L}_S(\mathbf{w})}\right] = -\log \mathbb{E}_{\mathbf{w}\sim P}\left[\frac{1}{m}\sum_{i=1}^{m} e^{-\lambda \hat{L}_S(\mathbf{w}_i)}\right] \leq \mathbb{E}_{\mathbf{w}\sim P}\left[-\log\left(\frac{1}{m}\sum_{i=1}^{m} e^{-\lambda \hat{L}_S(\mathbf{w}_i)}\right)\right]$$

The authors observe that their bound is very loose in practice. This is expected since they don't use any tempering. Compared to AIS and the paired-product estimator, they estimate the random areas with a single rectangle that is the left sum with $\beta = 0, 1$. Even with a data-dependent prior, this can be a very crude approximation. We only estimate expectations under tempered Gibbs posteriors, not Gaussians and hence obtain tighter bounds on the KL divergence.

Summary. The present chapter brought together areas of literature related to our method. We presented and compared approaches to obtain tight PAC-Bayes bounds, debates around the effectivity of MFVI, and alternatives to each step of our method. Our discussion touched on some of the many connections between results in these largely separate communities, which, the author hopes, illustrates the beauty of this area.

Chapter 6

Conclusion

6.1 Discussion

In this thesis, we have attempted to add to our understanding of generalization in neural networks, by studying generalization guarantees (risk certificates) in the PAC-Bayes framework. Existing methods for calculating PAC-Bayes risk certificates involve various approximations, and an evaluation of the strength of these is much needed in the literature. Therefore, our central question was the effect of one of the most prominent approximation steps: approximating the optimal value of PAC-Bayes bounds using mean-field Gaussians.

We have proposed a method to estimate the optimal value of a specific PAC-Bayes bound, the PAC-Bayes λ bound. Our approach brings together MCMC techniques, marginal likelihood estimation, and elements of probability theory. We have applied our method to small neural networks on versions of the MNIST dataset. Our optimal bound estimates improve on some existing risk certificates for model accuracy, but the values we guarantee under reasonable assumptions are often close to the MFVI approximations. We have found that the gap is largest for small models and that adding more data tightens our optimal bound estimates more than their MFVI approximations.

Limitations The chief limitation of our method lies in the task: PAC-Bayes bound estimates are valid under assumptions that are not possible to completely verify for MCMC samples. The alternative could be a theoretical treatment, but this becomes very challenging for complex (classification) tasks. Hence most of what is possible are empirical estimates accompanied by robust diagnostics. To achieve this, we make claims under different sets of assumptions of varying strength. There are many ways to further refine and improve our methods, which we discuss next.

6.2 Future Work

There following future directions can be investigated.

Improving sampling and diagnostics Since HMC is challenging to calibrate, automatic approaches, such as NUTS could be used to guarantee more robust performance [Homan and Gelman, 2014]. The evaluation of the obtained samples could also be extended by adding new tests and statistics, or potentially developing convergence diagnostics specifically for tempered Gibbs posteriors. An option for this could be to estimate the derivative of $\beta \to \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[\widetilde{L}_S^{\mathrm{CE}}(\mathbf{w})\right]$ and compare to the (estimated) negative variance which it should be equal to (Proposition 3.5.2).

Improving KL estimation The choice of discretization for $\beta \in [0,1]$ could be automated, for example as in Huber [2015].

Experiments on additional datasets and models Given that we found that the tightness gap is larger for smaller neural networks, it would be interesting to calculate tightness gaps on even smaller models and datasets. In addition, under a fixed model size, we futher test the effect of the number of layers and other architectural elements on the optimal risk certificate, possibly enabling us to validate hypotheses about architecture design.

Achieving tighter estimates on optimal PAC-Bayes bounds Although this was not our primary interest, our Gibbs posterior-based risk certificates could be made tighter in various ways. More parameter-efficient architectures could be considered, which likely produce a better trade-off between model accuracy and loss surface dimensionality. In particular, CNNs could be a good choice. In addition, data-dependent priors and cross-validation in λ could be implemented.

Incorporating convergence properties into PAC-Bayes bounds To reach full control on the accuracy of estimates of PAC-Bayes bounds in the Gibbs posterior, novel PAC-Bayes bounds could be developed that account for the convergence behavior of the underlying MCMC algorithm via the presence of MCMC statistics (see a related approach in Dziugaite and Roy [2018] for SGLD). In particular, it would be most useful to estimate $D_{\mathrm{KL}}(Q||Q^*)$ from a chain. However, we think that this is a challenging area of research.

References

Pierre Alquier. User-friendly introduction to pac-bayes bounds, 2023.

Pierre Alquier and James Ridgway. Concentration of tempered posteriors and of their variational approximations. *The Annals of Statistics*, 48(3):1475 – 1497, 2020. doi: 10.1214/19-AOS1855. URL https://doi.org/10.1214/19-AOS1855.

Pierre Alquier, James Ridgway, and Nicolas Chopin. On the properties of variational approximations of gibbs posteriors. *Journal of Machine Learning Research*, 17(236):1–41, 2016. URL http://jmlr.org/papers/v17/15-290.html.

D. Bertsekas. *Convex Optimization Theory*. Athena Scientific optimization and computation series. Athena Scientific, 2009. ISBN 9781886529311. URL https://books.google.co.uk/books?id=0H1iQwAACAAJ.

Alexandros Beskos, Natesh S. Pillai, Gareth O. Roberts, Jesus M. Sanz-Serna, and Andrew M. Stuart. Optimal tuning of the hybrid monte-carlo algorithm, 2010.

Michael Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv: Methodology*, 2017. URL https://api.semanticscholar.org/CorpusID:88514713.

P. G. Bissiri, C. C. Holmes, and S. G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78 (5):1103–1130, feb 2016. doi: 10.1111/rssb.12158. URL https://doi.org/10.1111%2Frssb.12158.

Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng, editors. *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, may 2011a. doi: 10.1201/b10905. URL https://doi.org/10.1201%2Fb10905.

Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo*. CRC press, 2011b.

Johannes Buchner. Nested sampling methods. *Statistics Surveys*, 17(none), jan 2023. doi: 10.1214/23-ss144. URL https://doi.org/10.1214%2F23-ss144.

Olivier Catoni. Pac-bayesian supervised classification: The thermodynamics of statistical learning. *Lecture Notes-Monograph Series*, 56:i–163, 2007. ISSN 07492170. URL http://www.jstor.org/stable/20461499.

Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient hamiltonian monte carlo. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1683–1691, Bejing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/cheni14.html.

Badr-Eddine Chérief-Abdellatif, Yuyang Shi, Arnaud Doucet, and Benjamin Guedj. On PAC-Bayesian reconstruction guarantees for VAEs. In *25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022*, Valencia / Virtual, Spain, March 2022. URL https://inria.hal.science/hal-03587178. 14 pages.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, July 2006. ISBN 0471241954.

Simon Duane, A.D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, 1987. ISSN 0370-2693. doi: https://doi.org/10.1016/0370-2693(87)91197-X. URL https://www.sciencedirect.com/science/article/pii/037026938791197X.

Cynthia Dwork. Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-35908-1.

Gintare Dziugaite. *Revisiting Generalization for Deep Learning: PAC-Bayes, Flat Minima, and Generative Models*. PhD thesis, 12 2018.

Gintare Karolina Dziugaite and Daniel M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *CoRR*, abs/1703.11008, 2017. URL https://arxiv.org/abs/1703.11008.

Gintare Karolina Dziugaite and Daniel M Roy. Data-dependent pac-bayes priors via differential privacy. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/9a0ee0a9e7a42d2d69b8f86b3a0756b1-Paper.pdf.

Gintare Karolina Dziugaite, Kyle Hsu, Waseem Gharbieh, Gabriel Arpino, and Daniel Roy. On the role of data in PAC-Bayes bounds. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 604–612. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/karolina-dziugaite21a.html.

Jianqing Fan, Bai Jiang, and Qiang Sun. Hoeffding's inequality for general markov chains and its applications to statistical learning. *Journal of Machine Learning Research*, 22(139): 1–35, 2021. URL http://jmlr.org/papers/v22/19-479.html.

Sebastian Farquhar, Lewis Smith, and Yarin Gal. Liberty or depth: Deep bayesian neural nets do not need complex weight posterior approximations. In *Advances in Neural Information Processing Systems*, volume 33, pages 4346–4357, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/2dfe1946b3003933b7f8ddd71f24dbb1-Paper.pdf.

Andrew Foong, Wessel Bruinsma, David Burt, and Richard Turner. How tight can pac-bayes be in the small data regime? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 4093–4105. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/214cfbe603b7f9f9bc005d5f53f7a1d3-Paper.pdf.

Andrew Y. K. Foong, David R. Burt, Yingzhen Li, and Richard E. Turner. On the expressiveness of approximate inference in bayesian neural networks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, 2020. ISBN 9781713829546.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=6Tm1mposlrM.

Andrew Gelman and Donald B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472, 1992. ISSN 08834237. URL http://www.jstor.org/stable/2246093.

Pascal Germain, Alexandre Lacasse, François Laviolette, and Mario Marchand. Pac-bayesian learning of linear classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 353–360, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553419. URL https://doi.org/10.1145/1553374.1553419.

Pascal Germain, Francis Bach, Alexandre Lacoste, and Simon Lacoste-Julien. Pac-bayesian theory meets bayesian inference. 05 2016.

Sandesh Ghimire, Aria Masoomi, and Jennifer G. Dy. Reliable estimation of kl divergence using a discriminator in reproducing kernel hilbert space. In *Neural Information Processing Systems*, 2021. URL https://api.semanticscholar.org/CorpusID:238227158.

Peter J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 12 1995. ISSN 0006-3444. doi: 10.1093/biomet/82.4.711. URL https://doi.org/10.1093/biomet/82.4.711.

Roger B. Grosse, Zoubin Ghahramani, and Ryan P. Adams. Sandwiching the marginal likelihood using bidirectional monte carlo. *CoRR*, abs/1511.02543, 2015. URL http://arxiv.org/abs/1511.02543.

Benjamin Guedj and Pierre Alquier. Pac-bayesian estimation and prediction in sparse additive models. *Electronic Journal of Statistics*, 7:264–291, 01 2013. doi: 10.1214/13-EJS771.

Benjamin Guedj and Sylvain Robbiano. Pac-bayesian high dimensional bipartite ranking. *Journal of Statistical Planning and Inference*, 196:70–86, 2018. ISSN 0378-3758. doi: https://doi.org/10.1016/j.jspi.2017.10.010. URL https://www.sciencedirect.com/science/article/pii/S0378375817301945.

Maxime Haddouche and Benjamin Guedj. PAC-bayes generalisation bounds for heavy-tailed losses through supermartingales. *Transactions on Machine Learning Research*, 2023a. ISSN 2835-8856. URL https://openreview.net/forum?id=qxrwt6F3sf.

Maxime Haddouche and Benjamin Guedj. Wasserstein pac-bayes learning: Exploiting optimisation guarantees to explain generalisation, 2023b.

Maxime Haddouche, Benjamin Guedj, Omar Rivasplata, and John Shawe-Taylor. PAC-bayes unleashed: Generalisation bounds with unbounded losses. *Entropy*, 23(10):1330, oct 2021. doi: 10.3390/e23101330. URL https://doi.org/10.3390%2Fe23101330.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 06 2016. doi: 10.1109/CVPR.2016.90.

Geoffrey E. Hinton and Drew van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, COLT '93, page 5–13, New York, NY, USA, 1993. Association for Computing Machinery. ISBN 0897916115. doi: 10.1145/168304.168306. URL https://doi.org/10.1145/168304.168306.

Matthew D. Homan and Andrew Gelman. The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, jan 2014. ISSN 1532-4435.

Mark Huber. Approximation algorithms for the normalizing constant of Gibbs distributions. *The Annals of Applied Probability*, 25(2):974 – 985, 2015. doi: 10.1214/14-AAP1015. URL https://doi.org/10.1214/14-AAP1015.

Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are bayesian neural network posteriors really like? In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4629–4640. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/izmailov21a.html.

Cheongjae Jang, Sungyoon Lee, Frank Park, and Yung-Kyun Noh. A reparametrization-invariant sharpness measure based on information geometry. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27893–27905. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b2ba568effcc3ab221912db2fb095ea9-Paper-Conference.pdf.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=SJgIPJBFvH.

Galin L. Jones and Qian Qin. Markov chain monte carlo in practice. *Annual Review of Statistics and Its Application*, 9(1):557–578, 2022. doi: 10.1146/annurev-statistics-040220-090158. URL https://doi.org/10.1146/annurev-statistics-040220-090158.

Haotian Ju, Dongyue Li, and Hongyang R Zhang. Robust fine-tuning of deep neural networks with hessian-based generalization guarantees. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings* *of Machine Learning Research*, pages 10431–10461. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ju22a.html.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=H1oyRlYgg.

Minyoung Kim, Da Li, Shell X Hu, and Timothy Hospedales. Fisher SAM: Information geometry and sharpness aware minimisation. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 11148–11161. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/kim22f.html.

John Langford and Rich Caruana. (not) bounding the true error. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001. URL https://proceedings.neurips.cc/paper_files/paper/2001/file/98c7242894844ecd6ec94af67ac8247d-Paper.pdf.

John Langford and Matthias Seeger. Bounds for averaging classifiers. Technical report, CMU-CS-01-102. Carnegie Mellon University, 2001.

Le Li, Benjamin Guedj, and Sé bastien Loustau. A quasi-bayesian perspective to online clustering. *Electronic Journal of Statistics*, 12(2), jan 2018. doi: 10.1214/18-ejs1479. URL https://doi.org/10.1214%2F18-ejs1479.

Sanae Lotfi, Marc Anton Finzi, Sanyam Kapoor, Andres Potapczynski, Micah Goldblum, and Andrew Gordon Wilson. PAC-bayes compression bounds so tight that they can explain generalization. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=o8nYuR8ekFm.

Wesley J. Maddox, Timur Garipov, Pavel Izmailov, Dmitry Vetrov, and Andrew Gordon Wilson. *A Simple Baseline for Bayesian Uncertainty in Deep Learning*. Curran Associates Inc., Red Hook, NY, USA, 2019.

Vaden Masrani, Tuan Anh Le, and Frank Wood. The thermodynamic variational objective. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/618faa1728eb2ef6e3733645273ab145-Paper.pdf.

Andreas Maurer. A note on the pac bayesian theorem, 2004.

Sokhna Diarra Mbacke, Florence Clerc, and Pascal Germain. Pac-bayesian generalization bounds for adversarial generative models. *CoRR*, abs/2302.08942, 2023. doi: 10.48550/arXiv.2302.08942. URL https://doi.org/10.48550/arXiv.2302.08942.

David A. McAllester. Some pac-bayesian theorems. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, COLT' 98, page 230–234, New York, NY, USA, 1998. Association for Computing Machinery. ISBN 1581130570. doi: 10.1145/279943.279989. URL https://doi.org/10.1145/279943.279989.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 68(3):411–436, 2006. ISSN 13697412, 14679868. URL http://www.jstor.org/stable/3879283.

Mahdi Naeini, Pakdaman, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1), Feb. 2015. doi: 10.1609/aaai.v29i1.9602. URL https://ojs.aaai.org/index.php/AAAI/article/view/9602.

Radford M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, Berlin, Heidelberg, 1996. ISBN 0387947248.

R.M. Neal. Annealed importance sampling. *Statistics and Computing*, 11, 01 2001.

Christopher Nemeth and Paul Fearnhead. Stochastic gradient markov chain monte carlo, 2019.

Maria Perez-Ortiz, Omar Rivasplata, John Shawe-Taylor, and Csaba Szepesvari. Tighter risk certificates for neural networks. *J. Mach. Learn. Res.*, 22:227:1–227:40, 2020. URL https://api.semanticscholar.org/CorpusID:220793611.

Konstantinos Pitas. Dissecting non-vacuous generalization bounds based on the mean-field approximation. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7739–7749. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/pitas20a.html.

Liva Ralaivola, Marie Szafranski, and Guillaume Stempfel. Chromatic pac-bayes bounds for non-iid data. In David van Dyk and Max Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 416–423, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL https://proceedings.mlr.press/v5/ralaivola09a.html.

Daniel A. Roberts. Sgd implicitly regularizes generalization error, 2021.

Borja Rodriguez Gálvez, Ragnar Thobaben, and Mikael Skoglund. More pac-bayes bounds: From bounded losses, to losses with general tail behaviors, to anytime-validity, 06 2023.

Jeffrey S. Rosenthal. Simple confidence intervals for MCMC without CLTs. *Electronic Journal of Statistics*, 11(1):211 – 214, 2017. doi: 10.1214/17-EJS1224. URL https://doi.org/10.1214/17-EJS1224.

Daniel Rudolf. Explicit error bounds for markov chain monte carlo. *Dissertationes Mathematicae*, 485:1–93, 2012. doi: 10.4064/dm485-0-1. URL https://doi.org/10.4064%2Fdm485-0-1.

Samuel L. Smith, Benoit Dherin, David G. T. Barrett, and Soham De. On the origin of implicit regularization in stochastic gradient descent, 2021.

Niklas Thiemann, Christian Igel, Olivier Wintenberger, and Yevgeny Seldin. A strongly quasiconvex pac-bayesian bound. In Steve Hanneke and Lev Reyzin, editors, *Proceedings of the 28th International Conference on Algorithmic Learning Theory*, volume 76 of *Proceedings of Machine Learning Research*, pages 466–492. PMLR, 15–17 Oct 2017. URL https://proceedings.mlr.press/v76/thiemann17a.html.

L. G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, nov 1984. ISSN 0001-0782. doi: 10.1145/1968.1972. URL https://doi.org/10.1145/1968.1972.

Paul Viallard, Maxime Haddouche, Umut Simsekli, and Benjamin Guedj. Learning via wasserstein-based high probability generalisation bounds. *ArXiv*, abs/2306.04375, 2023.

Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 681–688, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Florian Wenzel, Kevin Roth, Bastiaan S. Veeling, Jakub undefinedwiątkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy8gdB9xx.

Puning Zhao and Lifeng Lai. Minimax optimal estimation of kl divergence for continuous distributions. *IEEE Transactions on Information Theory*, 66(12):7787–7811, 2020. doi: 10.1109/TIT.2020.3009923.

Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P. Adams, and Peter Orbanz. Non-vacuous generalization bounds at the imagenet scale: a PAC-bayesian compression approach. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJgqqsAct7.

Appendix A

Proofs

Proposition 3.3.1. *Fix a prior measure P with density $p(\mathbf{w})$ and define a (possibly generalized) likelihood $p(z|\mathbf{w})$. For any distribution Q that is dependent on the data z, with density $q(\mathbf{w}|z)$ the following holds*

$$D_{KL}[q(\mathbf{w}|z)||p(\mathbf{w})] = \mathbb{E}_q[\log p(z|\mathbf{w})] - \log p(z) + D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w}|z)], \tag{3.7}$$

where $p(\mathbf{w}|z)$ is the density of the posterior measure with respect to P and $p(z|\mathbf{w})$.

Proof. we can write the ELBO in two ways:

$$\mathbb{E}_q\left[\log \frac{p(z,\mathbf{w})}{q(\mathbf{w}|z)}\right] = \mathbb{E}_q\left[\log \frac{p(\mathbf{w})p(z|\mathbf{w})}{q(\mathbf{w}|z)}\right] = \mathbb{E}_q\left[\log \frac{p(z)p(\mathbf{w}|z)}{q(\mathbf{w}|z)}\right] \tag{A.1}$$

We can expand both terms as:

$$\mathbb{E}_q[\log p(z|\mathbf{w})] + \mathbb{E}_q\left[\log \frac{p(\mathbf{w})}{q(\mathbf{w}|z)}\right] = \mathbb{E}_q[\log p(z)] + \mathbb{E}_q\left[\log \frac{p(\mathbf{w}|z)}{q(\mathbf{w}|z)}\right], \tag{A.2}$$

which we can rewrite as

$$\mathbb{E}_q[\log p(z|\mathbf{w})] - \mathbb{E}_q\left[\log \frac{q(\mathbf{w}|z)}{p(\mathbf{w})}\right] = \mathbb{E}_q[\log p(z)] - \mathbb{E}_q\left[\log \frac{q(\mathbf{w}|z)}{p(\mathbf{w}|z)}\right]. \tag{A.3}$$

We can now rewrite this as KL divergences and notice that $p(z)$ is independent of $\mathbf{w}$, hence

$$\mathbb{E}_q[\log p(z|\mathbf{w})] - D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w})] = \log p(z) - D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w}|z)]. \tag{A.4}$$

We can reorder this as:

$$D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w})] = \mathbb{E}_q[\log p(z|\mathbf{w})] - \log p(z) + D_{\mathrm{KL}}[q(\mathbf{w}|z)||p(\mathbf{w}|z)]. \tag{A.5}$$

□

Proposition 3.2.1. *Let G be any distribution such that $G \neq Q^*$ and $G \neq P$. Then the tightness gap satisfies*

$$C(G) - C(Q^*) \geq bD_{KL}(G||Q^*).$$

Proof. The proof mostly follows the proof of the Pythagorean inequality for the KL divergence. See Theorem 11.6.1 in Cover and Thomas [2006]. For Equation A.6 to hold, we require the existence of a convex set $\mathcal{S} \subset \mathcal{M}_1(\mathcal{W})$ such that $Q^*, G \in \mathcal{S}$ and $P \notin \mathcal{S}$. Since $G \neq P$, such a set clearly exists. Since Q^* is the unique minimizer of $C(Q)$, by defining $P_\lambda := \lambda G + (1-\lambda)Q^* \in \mathcal{S}$ for $\lambda \in [0,1]$, we have that

$$\begin{aligned}
0 &\leq \frac{d}{d\lambda} C(P_\lambda)\Big|_{\lambda=0+} && \text{(A.6)}\\
&= \frac{d}{d\lambda}\left[a\mathbb{E}_{P_\lambda}[\hat{L}(\mathbf{w})] + bD_{KL}(P_\lambda||P)\right]\Big|_{\lambda=0+} && \text{(A.7)}\\
&= \frac{d}{d\lambda}\left[a\left[\lambda\mathbb{E}_G[\hat{L}(\mathbf{w})] + (1-\lambda)\mathbb{E}_{Q^*}[\hat{L}(\mathbf{w})]\right] + bD_{KL}(P_\lambda||P)\right]\Big|_{\lambda=0+} && \text{(A.8)}\\
&= a\left[\mathbb{E}_G[\hat{L}(\mathbf{w})] - \mathbb{E}_{Q^*}[\hat{L}(\mathbf{w})]\right] + b\int [G(\mathbf{w}) - Q^*(\mathbf{w})]\log\frac{p_\lambda(\mathbf{w})}{p(\mathbf{w})}\Big|_{\lambda=0+} && \text{(A.9)}\\
&+ b\int \frac{p_\lambda(\mathbf{w})p(\mathbf{w})}{p_\lambda(\mathbf{w})}\frac{g(\mathbf{w}) - q^*(\mathbf{w})}{p(\mathbf{w})}\mathrm{d}\mathbf{w}\Big|_{\lambda=0+} && \text{(A.10)}\\
&= a\left[\mathbb{E}_G[\hat{L}(\mathbf{w})] - \mathbb{E}_{q^*}[\hat{L}(\mathbf{w})]\right] + b\int g(\mathbf{w})\log\frac{q^*(\mathbf{w})}{p(\mathbf{w})}\frac{g(\mathbf{w})}{g(\mathbf{w})}\mathrm{d}\mathbf{w} - bD_{KL}(Q^*||P) && \text{(A.11)}\\
&= a\left[\mathbb{E}_G[\hat{L}(\mathbf{w})] - \mathbb{E}_{Q^*}[\hat{L}(\mathbf{w})]\right] + bD_{KL}(G||P) - bD_{KL}(G||Q^*) - bD_{KL}(Q^*||P). && \text{(A.12)}
\end{aligned}$$

Rearranging, we obtain

$$C(G) - C(Q^*) \geq bD_{KL}(G||Q^*). \tag{A.13}$$

In the slightly patological case when $G = P$, since $D_{KL}(G||P) = 0$, we obtain a simple bound

$$a[\mathbb{E}_P[\hat{L}(\mathbf{w})] - \mathbb{E}_{Q^*}[\hat{L}(\mathbf{w})]] \geq b[D_{KL}(P||Q^*) + D_{KL}(Q^*||P)] = 2b\ \mathrm{JSD}(P||Q^*). \tag{A.14}$$

□

Proposition 3.6.1. *Let P be the prior, and let Q^*_λ denote the corresponding (Gibbs) posterior $Q^*_\lambda \propto e^{-n\lambda \widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})} p(\mathbf{w})$. Suppose that we are able to simulate from a distribution Q. Let G be another distribution (in our case, a Gaussian) such that $D_{\mathrm{KL}}(Q||Q^*_\lambda) \leq D_{\mathrm{KL}}(G||Q^*_\lambda) + \mathbb{E}_Q\left[\widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})\right]$. Then,*

$$D_{\mathrm{KL}}(Q||P) \leq n\lambda \mathbb{E}_G\left[\widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})\right] + D_{\mathrm{KL}}(G||P). \tag{3.24}$$

Proof. We use Proposition 3.3.1 with P having density $p(\mathbf{w})$, G having density $q(\mathbf{w}|z)$ and Q^*_λ having density $p(\mathbf{w}|z)$ and $Z = \mathbb{E}_P\left[e^{-n\lambda \widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})}\right]$ being the marginal likelihood. We get

$$D_{\mathrm{KL}}(G||P) = -n\lambda \mathbb{E}_G\left[\widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})\right] - \log Z + D_{\mathrm{KL}}(G||Q^*_\lambda). \tag{A.15}$$

Reordering this, we obtain an estimate for $-\log Z$:

$$-\log Z = D_{\mathrm{KL}}(G||P) - D_{\mathrm{KL}}(G||Q^*_\lambda) + n\lambda \mathbb{E}_G\left[\widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})\right]. \tag{A.16}$$

This estimate can be used to calculate $D_{\mathrm{KL}}(Q||P)$, invoking Proposition 3.3.1 again.

$$\begin{aligned} D_{\mathrm{KL}}(Q||P) = -n\lambda \mathbb{E}_Q\left[\widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})\right] + n\lambda \mathbb{E}_G\left[\widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})\right] + D_{\mathrm{KL}}(G||P) \\ + \left(D_{\mathrm{KL}}(Q||Q^*_\lambda) - D_{\mathrm{KL}}(G||Q^*_\lambda)\right) \end{aligned} \tag{A.17}$$

Since we assumed that $D_{\mathrm{KL}}(Q||Q^*_\lambda) \leq D_{\mathrm{KL}}(G||Q^*_\lambda)$, the last term is negative. By ignoring it, we thus obtain an upper bound on $D_{\mathrm{KL}}(Q||P)$.

We note that for $G \equiv P$, Equation A.15 becomes Jensen's inequality for the function $-\log$, i.e.,

$$-\log \mathbb{E}_G\left[e^{-n\lambda \widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})}\right] \leq -\mathbb{E}_G\left[\log\left(e^{-n\lambda \widetilde{L}^{\mathrm{CE}}_S(\mathbf{w})}\right)\right]. \tag{A.18}$$

□

Proposition 3.5.3. *Let $g(\beta) = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}[-U'(\mathbf{w})] = \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[n\lambda\widetilde{L}_S^{CE}(\mathbf{w})\right]$. Then we have that $\frac{\partial^2 g(\beta)}{\partial\beta^2} \geq 0$, hence $g(\beta)$ is convex.*

Proof. We make use of the fact that $\frac{\partial}{\partial\beta}\mathbb{E}_{\mathbf{w}\sim\pi_\beta}[-U'(\mathbf{w})] = -\mathrm{Var}_{\mathbf{w}\sim\pi_\beta}[U'(\mathbf{w})]$, proved in Masrani et al. [2019] (note that they define $g(\beta)$ to be the negative of our $g(\beta)$, hence the two statements differ by a minus sign). Plugging this in and expanding the variance, we obtain

$$\begin{aligned}
\frac{\partial^2 g(\beta)}{\partial\beta^2} &= \frac{\partial^2}{\partial\beta^2}\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[-U'(\mathbf{w})\right] && \text{(A.19)}\\
&= -\frac{\partial}{\partial\beta}\left[\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^2\right] - \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^2\right] && \text{(A.20)}\\
&= -\frac{\partial}{\partial\beta}\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^2\right] + 2\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]\frac{\partial}{\partial\beta}\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right] && \text{(A.21)}\\
&= A + B. && \text{(A.22)}
\end{aligned}$$

For B we can use the above, i.e.

$$\begin{aligned}
B &= 2\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]\mathrm{Var}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right] && \text{(A.23)}\\
&= 2\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^2\right] - 2\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^3. && \text{(A.24)}
\end{aligned}$$

For A, note that $U'(\mathbf{w})$ is *independent of* β, hence we can directly plug into equation (30) in Masrani et al. [2019], noting that we have an extra $U'(\mathbf{w})$ term and we need the negative of their expression:

$$\begin{aligned}
A &= \int (U'(\mathbf{w}))^2\pi_\beta(\mathbf{w})\mathbb{E}_{\mathbf{w}\sim\pi_\beta(\mathbf{w})}\left[U'(\mathbf{w})\right]\mathrm{d}\mathbf{w} - \int (U'(\mathbf{w}))^3\pi_\beta(\mathbf{w})\mathrm{d}\mathbf{w} && \text{(A.25)}\\
&= \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^2\right] - \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^3\right]. && \text{(A.26)}
\end{aligned}$$

Hence $A+B$ is

$$A+B = -2\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^3 - \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^3\right] + 3\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^2\right]. \tag{A.27}$$

Since $U'(\mathbf{w}) = -n\lambda\hat{L}_S(\mathbf{w})$, it is negative. The function $x \mapsto x^3$ is concave on $(-\infty, 0)$, while the function $x \mapsto x^2$ is convex. Hence Jensen's inequality gives:

$$\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^3 \geq \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^3\right] \tag{A.28}$$

$$\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^2 \leq \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[(U'(\mathbf{w}))^2\right]. \tag{A.29}$$

Thus,

$$A+B = \frac{\partial^2 g(\beta)}{\partial\beta^2} \geq 3\left[-\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^3 + \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^2\right] \tag{A.30}$$

$$= 3\left[-\mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^3 + \mathbb{E}_{\mathbf{w}\sim\pi_\beta}\left[U'(\mathbf{w})\right]^3\right] = 0, \tag{A.31}$$

thus $g(\beta)$ is convex. □