Bounding the Generalization Error: A Literature Review and Code Experiment for PAC-Bayesian Analysis

Yijiang Dong, Ruijie Mao, Jiarong Qi

ABSTRACT

In computational learning theory, probably approximately correct (PAC) learning purposes a framework that can provide generalization error bound with high probability. But in practice, PAC bound suffers from being vacuous and remains as a theoretical subject. PAC-Bayes approach obtains a tighter generalization error bound under the PAC framework by adding a Bayesian viewpoint and assuming a data-dependent prior distribution over the concept class. In this literature review, we will introduce theoretical PAC-Bayes bounds assuming previous knowledge of PAC learning. Then we will examine some recent works that apply PAC-Bayes bounds to neural network and achieve tight generalization error bound. We also discuss other applications of PAC-Bayes framework in adversarial learning, meta learning, etc. Lastly, we provide our own code experiments and our observations.

Keywords: PAC-Bayes Bounds, Optimization, Neural Networks, Meta Learning

PART I: INTRODUCING THE PAC-BAYES BOUNDS AND EMPIRICAL APPLICATIONS

1.1 Introduction

In order to introduce the PAC-Bayesian analysis, it is necessary to start off with PAC analysis and Bayesian analysis. Both PAC and Bayesian learning take data as input and output a concept from a hypothesis class. However, they differ in the setting of data. While the input distribution for PAC learning is unknown, Bayesian analysis requires input and test data to be generated from a prior distribution. PAC learning could be applied to a wide range of input data, regardless of the input distribution, but this generalization comes at a price of performance. On the other hand, while Bayesian analysis tend to outperform PAC analysis, when the data is distributed according to the specific prior, the output hypothesis fails to be applicable to other distributions. This is summarized as the "performance/generality trade off" by McAllester, one of the first researchers to prove PAC-Bayes bounds and theorems. [23]

PAC-Bayesian analysis combines the advantages of both PAC analysis and Bayesian analysis, and as a result oversteps the limits of both methods. PAC-Bayesian algorithms can be applied regardless of the input distribution, and they achieve good performance by utilizing the distributions of training and test data when such distributions are unknown.

The idea of PAC-Bayes bounds was first proposed by John Shawe-Taylor and Robert C. Williamson in their paper "A PAC Analysis of a Bayesian Estimator" in 1997. Taylor and Williamson combined the method of PAC analysis, which gives a priori estimates, and Bayesian analysis, which gives a posteriori estimates. [29] They provided generalization to the Bayesian analysis framework by placing a ball in a finite volume parameter space with uniform prior distribution, and the region of the ball represents correct classification. A larger region corresponds to a better bound on generalization. The authors also defined "luckiness" and "unluckiness" to parallel the prior encoding in Bayesian analysis and made the connection to PAC analysis by defining the concept "probably smooth". To lower generalization error, the authors lowered the unluckiness of the target function, which could be measured through the volume of the ball. The bound they ultimately derived was independent of the complexity of the hypothesis class, but dependent on the dimension of the parameter space. [29]

Then, David A. McAllester proved the first PAC-Bayes bounds in 1998 and 1999. While the concept space for the bound by Taylor and Williamson needs to be parameterized, McAllester developed bounds applicable to any concept space. Only a set of concepts is needed. Bounds by McAllester are also

tighter than the previous bound. In his paper "Some PAC-Bayesian Theorems", McAllester developed two preliminary theorems and two main theorems, where the preliminary theorems are applicable to a countable concept class, and the main theorems are generalized versions of preliminary theorems and concern all measurable subsets U of the concept space. The difference between the generalization error and the empirical error is bounded within an expression of a function $P(U)$ and the number of instances m. [23] McAllester developed a new theorem with a focus on model averaging in his paper "PAC-Bayesian Model Averaging". This new bound is capable of selecting a weighted subset of concepts instead of a uniformly weighted one. [24]

After that, more bounds are proven to reduce the difference between the empirical error and the generalization error. Some important bounds were given by Seeger and Langford, Maurer and Catoni. [28] [19] [22] [4] In the next section, we will introduce the key concepts and theorems of the PAC-Bayesian analysis, mainly following the notations and methods of Tim van Erven. [35]

1.2 Definition and Mathematics of PAC-Bayes Bounds

Notation: suppose the data $D = ((X_1,Y_1),(X_2,Y_2),...,(X_n,Y_n))$ is independent, identically distributed. For the i-th example and a hypothesis $h \in \mathscr{H}$, the loss function is defined as $l(X_i,Y_i,h)$, which, for example, could be $(Y_i - h(X_i))^2$ for the quadratic loss function. The empirical error measured from data D for hypothesis h is $R_n(D,h) = \frac{1}{n}\sum_{i=1}^{n} l(X_i,Y_i,h)$. The generalization error is $R(h) = \mathbb{E}[l(X,Y,h)]$. We would like to show that the generalization error $R(h)$ is close to the empirical error $R_n(D,h)$. van Erven also defined a term $M_\eta(h) = -\frac{1}{\eta} ln\mathbb{E}[e^{-\eta l(X,Y,h)}]$ (where $\eta > 0$) to substitute $R(h)$. The scheme is to first bound the difference between $R_n(D,h)$ and $M_\eta(h)$, and then relate $M_\eta(h)$ to $R(h)$. [35]

We will first introduce inequalities used in the proof for the bound: the Cramer-Chernoff Method, Hoeffding's inequality, and Markov's inequality.

The Cramer-Chernoff Method [36]

For a random variable X, for $r > 0$, the tail probability $P(X > x)$ is bounded by

$$P(X > x) = P(exp(rX) > exp(rx)) \leq exp(-rx)\mathbb{E}exp(rX)$$

Because this holds for all $r > 0$, $P(X > x) \leq \underset{r>0}{inf} exp(-rx)\mathbb{E}exp(rX)$.

Hoeffding's Inequality [8]

For independent, bounded random variables $Z_1,...,Z_n$, if all Z_i satisfies $a \leq Z_i \leq b$, where $-\infty < a < b < \infty$, then for all $t \geq 0$,

$$\mathbb{P}(\tfrac{1}{n}\textstyle\sum_{i=0}^{n}(Z_i - \mathbb{E}[Z_i]) \geq t) \leq exp(-\frac{2nt^2}{(b-a)^2})$$
$$\mathbb{P}(\tfrac{1}{n}\textstyle\sum_{i=0}^{n}(Z_i - \mathbb{E}[Z_i]) \leq -t) \leq exp(-\frac{2nt^2}{(b-a)^2})$$

From Hoeffding's Inequality, the following lemma can be proved. We will move on with a focus on deriving the PAC-Bayes bounds, but a detailed proof could be found in Appendix A.1.1 of Cesa-Bianchi and Lugosi's *Prediction, Learning, and Games*. [6]

Lemma 0

For a random variable X such that $a \leq X \leq b$, for any real number s,

$$ln\mathbb{E}e^{sX} \leq s\mathbb{E}X + \frac{s^2(b-a)^2}{8}$$

Markov's Inequality

For a nonnegative random variable X and any positive real number a,

$$\mathbb{P}(X \geq a) \leq \frac{\mathbb{E}[X]}{a}$$

Lemma 1

For any $\delta \in (0,1]$, with probability at least $1-\delta$, suppose $l(X,Y,h) \in [a,b]$,

$$R(h) \leq R_n(D,h) + \sqrt{\frac{ln(1/\delta)(b-a)^2}{2n}}$$

Proof: By the Cramer-Chernoff Method and Markov's Inequality,

$$\mathbb{P}(e^{-\eta n R_n(D,h)} \geq \mathbb{E}_{D'}[e^{-\eta n R_n(D',h)}]/\delta) \leq \frac{\mathbb{E}_{D'}[e^{-\eta n R_n(D',h)}]}{\mathbb{E}_{D'}[e^{-\eta n R_n(D',h)}]/\delta} = \delta$$

So with probability at least $1-\delta$,

$$e^{-\eta n R_n(D,h)} \leq \mathbb{E}_{D'}[e^{-\eta n R_n(D',h)}]/\delta$$

Because the instances are independent and identically distributed, RHS is $\frac{\mathbb{E}[e^{-\eta l(X,Y,h)}]^n}{\delta}$. Because the exponential function is increasing, we know that $-\frac{1}{\eta}ln(LHS) \geq -\frac{1}{\eta}ln(RHS)$

$$\begin{gathered}
-\frac{1}{\eta}ln(RHS) = -\frac{n}{\eta}ln\mathbb{E}[e^{-\eta l(X,Y,h)}] - \frac{1}{\eta}ln\frac{1}{\delta} = nM_\eta(h) - \frac{1}{\eta}ln\frac{1}{\delta} \\
-\frac{1}{\eta}ln(LHS) = nR_n(D,h) \\
nM_\eta(h) - \frac{1}{\eta}ln\frac{1}{\delta} \leq nR_n(D,h) \\
M_\eta(h) \leq R_n(D,h) + \frac{1}{\eta n}ln\frac{1}{\delta}
\end{gathered}$$

By Lemma 0, since $l(X,Y,h) \in [a,b]$, $-l(X,Y,h) \in [-b,-a]$,

$$\begin{gathered}
ln\mathbb{E}[e^{-\eta l(X,Y,h)}] \leq -\eta\mathbb{E}[l(X,Y,h)] + \frac{\eta^2(b-a)^2}{8} \\
-\frac{1}{\eta}ln\mathbb{E}[e^{-\eta l(X,Y,h)}] \geq E[l(X,Y,h)] - \frac{\eta(b-a)^2}{8} \\
R(h) \leq M_\eta(h) + \frac{\eta(b-a)^2}{8}
\end{gathered}$$

Therefore, with probability at least $1-\delta$,

$$R(h) \leq M_\eta(h) + \frac{\eta(b-a)^2}{8} \leq R_n(D,h) + \frac{1}{\eta n}ln\frac{1}{\delta} + \frac{\eta(b-a)^2}{8}$$

We could pick $\eta > 0$ for the tightest bound. To minimize RHS, pick $\eta = \sqrt{\frac{8ln(1/\delta)}{n(b-a)^2}}$, then we get the following result:

With probability at least $1-\delta$, $R(h) \leq R_n(D,h) + \sqrt{\frac{ln(1/\delta)(b-a)^2}{2n}}$

Lemma 2

Suppose the hypothesis class $\mathscr{H}$ is countable. We pick a hypothesis $\hat{h}$ from $\mathscr{H}$ based on current data. Let the "prior distribution" P be any function on $h \in \mathscr{H}$ such that $P(h) \geq 0$ for all h, and $\sum_{h\in\mathscr{H}} P(h) = 1$. Note that this prior distribution could be any function that satisfies the two conditions, instead of a distribution that reflects actual probabilities. Then, for any $\delta \in (0,1]$, for any $\eta > 0$, with probability at least $1-\delta$,

$$M_\eta(\hat{h}) \leq R_n(D,h) + \frac{1}{\eta n}ln\frac{1}{P(\hat{h})\delta}$$

Proof: We have already shown in the proof for Lemma 1 that $M_\eta(h) \leq R_n(D,h) + \frac{1}{\eta n}ln\frac{1}{\delta}$ with probability at least $1-\delta$ (1). Therefore, $\mathbb{P}(M_\eta(\hat{h}) > R_n(D,h) + \frac{1}{\eta n}ln\frac{1}{P(\hat{h})\delta})$ is not more that the probability that some hypothesis $h \in \mathscr{H}$ satisfies $M_\eta(h) > R_n(D,h) + \frac{1}{\eta n}ln\frac{1}{P(h)\delta}$, which is the sum of this probability for all h.

$$\mathbb{P}(M_\eta(\hat{h}) > R_n(D,h) + \frac{1}{\eta n} ln \frac{1}{P(\hat{h})\delta}) \leq \sum_{h \in \mathscr{H}} \mathbb{P}(M_\eta(h) > R_n(D,h) + \frac{1}{\eta n} ln \frac{1}{P(h)\delta})$$

$\mathbb{P}(M_\eta(h) > R_n(D,h) + \frac{1}{\eta n} ln \frac{1}{P(h)\delta}) < P(h)\delta$ for each h by the application of (1). Then,

$$\mathbb{P}(M_\eta(\hat{h}) > R_n(D,h) + \frac{1}{\eta n} ln \frac{1}{P(\hat{h})\delta}) \leq \sum_{h \in \mathscr{H}} P(h)\delta = \delta$$
$$\mathbb{P}(M_\eta(\hat{h}) \leq R_n(D,h) + \frac{1}{\eta n} ln \frac{1}{P(\hat{h})\delta}) > 1 - \delta$$

Although this is a rather satisfactory bound for a countable hypothesis class, the bounds we aim for are applicable for any continuous hypothesis class. To counter this drawback, using KL Divergence, the PAC-Bayes bounds could be derived.

KL Divergence

KL Divergence measures the difference between two probability distributions. If the probability distributions $p(x)$ and $q(x)$ are discrete, their KL Divergence is defined as

$$KL(p(x)||q(x)) = \sum_x p(x) ln \frac{p(x)}{q(x)}$$

For the continuous case,

$$KL(p(x)||q(x)) = \int_{-\infty}^{\infty} p(x) ln \frac{p(x)}{q(x)} dx$$

In the PAC-Bayesian analysis, the prior distribution P on the hypothesis class $\mathscr{H}$ is independent of the data, while the posterior distribution Q is data-dependent. Instead of picking a hypothesis by some decision rule, as in Lemma 2, in the PAC-Bayesian analysis, the output hypothesis is indeterminate and drawn from the posterior distribution Q. The KL Divergence serves as a measure for the difference between P and Q, and appears in the PAC-Bayes bound.

Lemma 3

For any $\eta > 0$, $\delta \in (0,1]$, with probability at least $1 - \delta$,

$$\mathbb{E}_{h \sim Q}[M_\eta(h)] \leq \mathbb{E}_{h \sim Q}[R_n(D,h)] + \frac{1}{\eta n}(KL(Q||pi) + ln \frac{1}{\delta})$$

Proof: We have shown in the proof for Lemma 1 that $\mathbb{E}_D[e^{-\eta n R_n(D,h)}] = \mathbb{E}[e^{-\eta l(X,Y,h)}]^n$,

$$e^{-\eta n M_\eta(h)} = e^{n ln \mathbb{E}[e^{-\eta l(X,Y,h)}]} = \mathbb{E}[e^{-\eta l(X,Y,h)}]^n = \mathbb{E}_D[e^{-\eta n R_n(D,h)}]$$
$$\mathbb{E}_{h \sim P}\mathbb{E}_D[\frac{e^{-\eta n R_n(D,h)}}{e^{-\eta n M_\eta(h)}}] = 1$$
$$\mathbb{E}_{h \sim P}\mathbb{E}_D[e^{-\eta n R_n(D,h) + \eta n M_\eta(h)}] = 1$$
$$\mathbb{E}_D\mathbb{E}_{h \sim P}[e^{-\eta n R_n(D,h) + \eta n M_\eta(h)}] = 1$$
$$\mathbb{E}_D\mathbb{E}_{h \sim Q}[(e^{-\eta n R_n(D,h) + \eta n M_\eta(h)})\frac{P(h)}{Q(h)}] = 1$$
$$\mathbb{E}_D\mathbb{E}_{h \sim Q}[e^{-\eta n R_n(D,h) + \eta n M_\eta(h) - ln \frac{Q(h)}{P(h)}}] = 1$$

Because the exponential function is convex, by Jensen's Inequality (also note that $\mathbb{E}_{h \sim Q} ln \frac{Q(h)}{P(h)} = KL(Q||P)$),

$$\mathbb{E}_D\mathbb{E}_{h \sim Q}[e^{-\eta n R_n(D,h) + \eta n M_\eta(h) - ln \frac{Q(h)}{P(h)}}] \geq \mathbb{E}_D[e^{-\eta n \mathbb{E}_{h \sim Q}[R_n(D,h) - M_\eta(h)] - KL(Q||P)}]$$
$$\mathbb{E}_D[e^{-\eta n \mathbb{E}_{h \sim Q}[R_n(D,h) - M_\eta(h)] - KL(Q||P)}] \leq 1$$

By Markov's Inequality,

$$\mathbb{P}(-\eta n \mathbb{E}_{h \sim Q}[R_n(D,h) - M_\eta(h)] - KL(Q||P) > ln \frac{1}{\delta}) = \mathbb{P}(e^{-\eta n \mathbb{E}_{h \sim Q}[R_n(D,h) - M_\eta(h)] - KL(Q||P)} > \frac{1}{\delta}) \leq$$
$$\mathbb{E}_D[e^{-\eta n \mathbb{E}_{h \sim Q}[R_n(D,h) - M_\eta(h)] - KL(Q||P)}]\delta \leq \delta$$

So with probability at least $1-\delta$,

$$-\eta n\mathbb{E}_{h\sim Q}[R_n(D,h)-M_\eta(h)]-KL(Q||P)\leq ln\frac{1}{\delta}$$
$$\mathbb{E}_{h\sim Q}[M_\eta(h)]\leq\mathbb{E}_{h\sim Q}[R_n(D,h)]+\frac{1}{\eta n}(KL(Q||pi)+ln\frac{1}{\delta})$$

Theorem 1

If the loss is bounded by $l(X,Y,h)\in[a,b]$, then for any $\alpha>1$, $v>0$, for any $\delta\in(0,1]$, for all $\eta\in(0,v]$ with probability at least $1-\delta$,

$$\mathbb{E}_{h\sim Q}[R(h)]\leq\mathbb{E}_{h\sim Q}[R_n(D,h)]+\eta\frac{(b-a)^2}{8}+\frac{\alpha}{\eta n}(KL(Q||P)+ln\frac{1}{\delta}+ln(\frac{1}{2}log_\alpha n+C))$$

Where $C=max\{log_\alpha(\frac{v(b-a)}{\sqrt{8\alpha}}),0\}+e$

This is an example of a PAC-Bayes bound given by van Erven. [35] To keep the focus of this literature review, we will not attempt to give the full proof for this theorem, but the derivation of this theorem is built on Lemma 3. The part $\eta\frac{(b-a)^2}{8}$ is obtained in the same way as in the proof for Lemma 1 (by relating $\mathbb{E}_{h\sim Q}[R(h)]$ to $\mathbb{E}_{h\sim Q}[M_\eta(h)]$). Bounding η within $(0,v]$ is an attempt to use the union bound to optimize η for the tightest bound.

Many more bounds are given by researchers and scholars, such as Catoni's bound, and many more bounds based on the Gibbs posterior that Alquier provides in his survey "User-friendly Introduction to PAC-Bayes Bounds". [4] [1] We will give a few examples of the bounds below.

Theorem 2(Catoni's bound) [4]

For any $\eta>0$, $\delta\in(0,1]$, with probability at least $1-\delta$,

$$\mathbb{E}_{h\sim Q}R(h)\leq\mathbb{E}_{h\sim Q}R_n(D,h)+\frac{\eta(b-a)^2}{8n}+\frac{KL(Q||P)+log\frac{1}{\delta}}{\eta}$$

Theorem 3(Thieman, Igel, Wintenberger and Seldin's bound) [33]

Suppose the data D has n instances, and r instances are selected independently from D to train hypothesis h. For any $\eta\in(0,2)$, for any $\delta\in(0,1]$, with probability at least $1-\delta$,

$$\mathbb{E}_{h\sim Q}R(h)\leq\frac{\mathbb{E}_{h\sim Q}R_n(D,h)}{1-\frac{\eta}{2}}+\frac{KL(Q||P)+ln\frac{2\sqrt{n-r}}{\delta}}{\eta(1-\frac{\eta}{2})(n-r)}$$

This is an example of a tight, non-vacuous PAC-Bayes bound. We will introduce non-vacuous PAC-Bayes bounds in the section below.

The bounds introduced above are empirical PAC-Bayes bounds. There is another category for PAC-Bayes bounds: the oracle PAC-Bayes bounds. While empirical PAC-Bayes bounds are based on empirical PAC bounds (bounds that can be calculate numerically with data), oracle PAC-Bayes bounds are based on oracle PAC bounds, bounds that cannot be computed numerically, but demonstrate the role of sample size and the set of predictors. Alquier has shown that although it is commonly perceived that empirical PAC-Bayes bounds are more applicable to practical problems while oracle PAC-Bayes bounds tend to be theoretical, there is no clear line between them. Empirical PAC-Bayes bounds could help develop theories on oracle PAC-Bayes bounds, and oracle PAC-Bayes bounds could help with the analysis of empirical PAC-Bayes bounds in practice. [1]

There is also a distinction between generic priors and oracle priors. The theorems above use generic priors, while oracle priors are able to optimize the expected value of the bound, thus optimizing the value of the KL Divergence term on the right hand side. [17] [9]

1.3 Non-Vacuous PAC-Bayes Bounds

According to Alquier, vacuous PAC-Bayes bounds are PAC-Bayes bounds that provide no additional information, i.e. the right hand side of the bound gets tremendously large. This could happen because some bounds are sensitive to the number of possible classifiers, so when there are multitudinous adjustable weights (as in neural networks), the right hand side of the bound could be uncontrollably large. On the

contrary, non-vacuous PAC-Bayes bounds are those that provide additional information. [1]

Therefore, the focus of researchers and scholars in the field is to reach tight non-vacuous PAC-Bayes bounds. Catoni's bound and Thieman, Igel, Wintenberger and Seldin's bound from above are both non-vacuous bounds. One more example of a famous non-vacuous bound is Tolstikhin and Seldin's bound:[34]

For any $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\mathbb{E}_{h\sim Q}[R(h)] \leq \mathbb{E}_{h\sim Q}[R_n(D,h)] + \sqrt{\frac{2\mathbb{E}_{h\sim Q}[R_n(D,h)](KL(Q||P)+ln\frac{2\sqrt{n}}{\delta})}{n}} + \frac{2(KL(Q||P)+ln\frac{2\sqrt{n}}{\delta})}{n}$$

PART II: APPLYING THE PAC-BAYES BOUNDS TO NEURAL NETWORK

Given the theoretical characterization of PAC-Bayes bound, the researchers started to apply the PAC-Bayes bound to a wide range of machine learning methods. In particular, [13] studied how PAC-Bayes bound can be applied to SVM and [5] studied how PAC-Bayes could provide risk certificates for linear regressions. In our literature review, we will focus on studying Neural Network, which is by far the most empirically effective yet poorly theoretically understood "black-box" model.

2.1 The Journey of Obtaining Tighter Generalization Bound for Neural Network

In section 2.1, we will focus on the research that provide tighter and tighter generalization bound on neural network. In Section 2.1.1, we introduce the first attempt of bounding the generalization error of (shallow) neural network by Langford and Caruana back in 2002. Then, in section 2.1.2, we present an important work by Dziugaite and Roy in 2017. They achieved non-vacuous generalization error bounds (around 0.2 on MNIST) by studying the role of stochastic gradient descent in modern massively overparametrized neural networks. Lastly, we present a recent work by Perez-Ortiz, Rivasplata, Shawe-Taylor and Szepesvari in 2021 in section 2.1.3. They achieved very tight bound (around 0.02 on MNIST) by studying the backward propagation algorithm.

2.1.1 The First Attempt of Bounding Generalization Errors of Neural Networks by Langford and Caruana [18]

This paper builds a construction algorithm based on the PAC-Bayes Relative Entropy Bound by Langford and Seeger in [19]. To implement the algorithm based on the above bounds, the authors adopt the traditional method of weight initialization and specify the prior P to a multidimensional gaussian with a mean of zero and variance in each dimension of b^2. The posterior Q is also set to be a multivariate normal distribution $N(w_i, \sigma_i)$. It's calculated by first training a neural network on the examples to get the w_is and then perform a sensitivity test to search for the largest variance that does not decrease the accuracy by a certain threshold. Plugging the prior and posterior to the theorem gives us the risk certificate of the generalization error.

The code experiment on a 100-examples synthetic dataset shows that this algorithm can achieve a "not completely tight but not vacuous" generalization bound. (The error is bounded by about 60%.) The authors also make a very remarkable observation that is further discussed in the following papers: the stochastic neural network can achieve much tighter (on the order of 2) generalization bound. This leads to the discussion of "flat minima", "entropy-SGD", "implicit regularization of SGD", etc.

2.1.2 Non-vacuous Generalization Error for Neural Networks by Dziugaite and Roy [10]

Extending the work by Langford and Caruana, Dziugaite and Roy studied stochastic gradient descent (SGD) in greater depth and achieved non-vacuous generalization error (around 0.2 on MNIST).

We first, as the authors did, diverge from PAC-Bayes and make some thought-provoking observations about SGD and generalization. In the famous paper "Understanding Deep Learning Requires Rethinking Generalization", Zhang et al. showed that modern highly overparametrized neural network optimized with SGD 1) can achieve near-zero training error even if the labels are random coin flips 2) can achieve near-zero training error and still generalize. These two experiments showed that although the modern

deep neural network has the capacity to overfit the data, it learns the pattern of the data rather than simply remembering it.

Some researchers conjecture that SGD performs some form of "implicit regularization" that forces the neural network to learn the structure of data. For example, Chaudhari et al. believe well-generalizable solutions lay in large flat regions of the energy landscape while poorly-generalizable parameters are located in the sharp valleys [7]. The intuition is that weights in sharp minima need to be highly precise to avoid overfitting while flat minimum does not require that level of precision. They claim that "without being explicitly tailored to do so", SGD implicitly optimizes for the flat regions as the SGD stops when the loss of several epochs of randomly selected mini-batches converges. Other works along this line of research include [3] [14] [15].

The hypothesis of "flat-minima" leads Dziugaite and Roy to believe that PAC-Bayes theorem could provide nonvacuous bound if the volume of "flat-minima" is large and not too far from the prior (which leads to small KL divergence in the PAC-Bayes relative Entropy Bound). They also mentioned that in modern overparameterized networks, each parameter has negligible effect on the training error and thus cannot effectively calculate the relative sensitivity as purposed in Langford and Caruana's algorithm. Instead, they use stochastic gradient descent to optimize the objective function: the sum of empirical loss plus the PAC-Bayes generalization error as a regularizer. The prior $N(w_0, \sigma^2 I)$ the authors choose is data-dependent while they did not optimize w_0 and just choose it randomly.

Their coding experiment on MNIST handwritten digits dataset shows that their purposed algorithm based on PAC-Bayes bound could bound the error rate by around 0.16 to 0.22 with probability 0.965. (They label digits 0 to 4 as 0 and digits 5 to 9 as 1 to transform the problem to a binary classification problem.) This is the first work known to us that could obtain nonvacuous (though still loose) generalization error bound for modern overparametrized neural networks.

2.1.3 Tighter Risk Certificates for Neural Networks by Perez-Ortiz et al. [26]

Mainly based on the papers discussed in section 2.1.1 and 2.1.2, Perez-Ortiz et al. exploit the back propagation of neural network and achieved even tighter generalization error (0.02 on MNIST) that can lead to self-certified learning. Under their "PAC-Bayes with Backprop" (PBB) framework, Perez-Ortiz et al. conducted thorough experiments with multiple different training objectives, prior distributions, optimizers, and training techniques including dropout and sample-splitting to find tighter PAC-Bayes bound. They also extend the network architecture from fully connected neural network (FCN) to Convolution neural network (CNN).

Building on the work of Dziugaite and Roy, which uses the classical PAC-Bayes relative entropy bound, Perez-Ortiz et al. used two other tighter PAC-Bayes bound as the training objectives: the PAC-Bayes-lambda bound from [33] and PAC-Bayes-quadratic bound from [27]. In addition to the Gaussian distribution to be considered by the the previous two papers as the prior and the posterior, Perez-Ortiz et al. adds the Laplacian distribution and considers both data-free and data-dependent priors. They also argue that the optimizer stochastic gradient Langevin dynamics (SGLD) that Dziugaite and Roy uses is not as good as they claim, so they used the standard vanilla SGD with momentum.

Their coding experiment achieved very tight risk certificate on MNIST and nonvacuous bound on CIFAR10. In particular, they achieved 1% test error and 1.5% risk certificate on MNIST with 2-layer CNN. By using 15-layer large CNN on CIFAR10, which is a more complex dataset, they achieved 14.6% test error and 18% risk certificate. These figures are tremendous improvement from previous works and show that self-certified learning may be possible.

2.2 PAC-Bayes Bounds and Neural Network Architecture

In section 2.2, we examine the application of PAC-Bayesian analysis to some specific types of neural networks: finite-width neural networks and neural networks with a binary activation function. In both papers, non-vacuous PAC-Bayes bounds were achieved for these variations of neural network architectures.

2.2.1 Transportation Map Estimation for Finite-Width Neural Networks by Suzuki [32]

Recently in 2020, in his research "Generalization Bound of Globally Optimal Non-Convex Neural Network Training: Transportation Map Estimation by Finite Dimensional Langevin Dynamics", Taiji Suzuki proposed to use transportation map estimation for finite-width neural network optimization. [32] The use of transportation map for neural networks was a novel idea, and the research obtained ground-breaking results of optimizing infinite and finite width neural networks and bounding their generalization gap and excess risk through the same process unifyingly.

Suzuki first discussed two conventional methods for optimization: the mean field theory and neural tangent kernel. Both mean field analysis and neural tangent kernel guarantee convergence to the global optimal. Mean field analysis has been practiced in biology and physics since decades ago [31], but recent convergence analysis began with a one-layer neural network by Sirignano in 2019. [30] The mean field theory deals with non-convexity of neural networks through analyzing neural network training as gradient flows, and requires taking limits of the number of nodes to infinity, as Sirignano put it, "a law of large numbers for neural networks." [30] Similarly, neural tangent kernels ensure convergence to 0 training loss with SGD, but the linearization of neural network also requires infinite width, and does not take advantage of the superiority of non-convex neural networks over linear methods. [12]

Therefore, Suzuki formulated transportation map estimation so that infinite-dimensional Langevin dynamics could be applied to finite-dimensional neural networks. Transportation map is a map from the particle's solution at time 0 to its solution at time t. The mapping is $w_0 \mapsto W(t, w_0)$, where W denotes the stochastic process, $w_0 = W(0)$ is the particle's location at time 0. Therefore, Suzuki transformed the optimization of neural network weights in to the optimization of transportation maps with $W(0)$ as an identity map. This induces convergence to a global optimal.

Optimization is achieved through minimizing the regularized empirical error $R_n(D,h) + \frac{\lambda}{2}||W||^2_{\mathscr{H}_K}$, where λ is a parameter of choice and $\mathscr{H}_K$ denotes a reproducing kernel Hilbert Space that Suzuki defined. Applying Langevin dynamics and an implicit Euler scheme that Suzuki defined in the infinite-dimensional RKHS, convergence is proven under a few assumptions that control the strength of the regularization term and the smoothness. Suzuki concluded that the generalization gap bound (corresponding to our PAC-Bayes bounds) is given as follows:

For any $\delta \in (0,1)$, if the loss function is bounded by $l(X,Y,h) \in [a,b]$ with probability at least $1-\delta$,

$$\mathbb{E}_{h\sim\hat{\pi}}[R(h)] \leq \mathbb{E}_{h\sim\hat{\pi}}[R_n(D,h)] + \frac{(b-a)^2}{\sqrt{n}}[2(1+\frac{2\beta}{\sqrt{n}}) + ln(\frac{1+e^{\frac{(b-a)^2}{2}}}{\delta})] + 2\varepsilon_K$$

Here ε_K denotes the optimization error, and β is a positive inverse temperature parameter. This shows that the generalization is dependent on both $O(\frac{1}{\sqrt{n}})$ and ε_K. Suzuki also bounded the excess risk (the difference between the expected risk of the output hypothesis and the expected risk of the best possible hypothesis) and demonstrates its fast learning rate.

2.2.2 A PAC-Bayesian Analysis of Binary Activated Deep Neural Networks by Letarte et al. [20]

Letarte et al. provided another application of PAC-Bayes Bounds to a specific neural network architecture: binary activated deep neural networks. Binary activation functions activate inputs above a certain value and deactivate inputs below that threshold, and for sign activation that the research concerns, the threshold is 0. Letarte et al. defines deep NN with a sign activation function as BAM networks (binary activation multilayer networks). The research discovers that such BAM networks obtain non-vacuous PAC-Bayes bounds with PBGNet when input instances are independent and identically distributed. [20]

In Letarte et al.'s experiment setting, there are L fully connected layers, the activation function is binary, and the classification problem predicts a label in $\{-1,1\}$. Their approach to BAM networks was built on Germain et al.'s application of PAC-Bayesian analysis to linear classifiers, where both prior and posterior of the weights are Gaussian. [11] Likewise, Letarte et al. assumed a Gaussian posterior, and minimized the generalization error through SGD.

By aggregating predictors, Letarte et al. brought the non-differentiability of binary functions under control. In this way the algorithm corresponds to a majority vote algorithm. For multiple number of layers, Letarte et al. proposed using a computation tree transformed from the BAM network. This leads to a deterministic output of the model. For training, Letarte et al. again built on Germain et al.'s PBGD (PAC-Bayesian Gradient Descent) algorithm. [11] The algorithm that Letarte et al. developed was

PBGNet (PAC-Bayesian Binary Gradient Network). The PAC-Bayes bound that Letarte et al. achieved is as follows:

For any $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\mathbb{E}_{h\sim\hat{\pi}}[R(h)] \leq \inf_{C>0}\{\frac{1}{1-e^{-C}}(1-exp(-C\mathbb{E}_{h\sim\hat{\pi}}[R_n(D,h)]-\frac{1}{n}[KL(\hat{\pi}||\pi)+ln\frac{2\sqrt{n}}{\delta}]))\}$$

In actual training, Letarte et al. explored both the MLP algorithm and four variants of the PBGNet algorithm. The number of hidden layers was $\{1,2,3\}$, and hidden size was picked from $\{10,50,100\}$. Six datasets were included: ads, adult, MNIST17, MNIST49, MNIST56, and MNISTLH. The experiment confirmed that the variants of the PBGNet algorithm achieved non-vacuous PAC-Bayes bounds.

2.3 application of PAC-Bayes Theory in Meta Learning

Given the basic structure of PAC-Bayesian framework, the researchers began to apply those frameworks into the real world. One of the most noticeable would be to provide the generalization bound for the meta learning. In section 2.3.1, we focus on the attempts brought by A. Pentina and C. Lambert who attempted to apply PAC-bayes bounds to lifelong learning, the prephase of meta learning. Then, in 2.3.2, we cover R. Amit and R. Meir's paper who took a step further to derive a tighter bounds that extends a single task PAC-bayes bound to meta learning setup. Further, in 2.3.3, we introduce the work proposed by S. T. Jose, G. Durisi who generalize prior works on transfer learning and quantify the impact of the meta-environment shift. Finally, in 2.3.4, we introduce how PAC-bayes bounds can be extended to data-dependent prior.

Note that in order to properly cover the content, which has a slightly different setting than the previous part, we adopt a different set of notations for posteriors and priors, as well as for generalization and sample error.

2.3.1 A PAC-Bayesian Bound for Lifelong Learning[25]

The first attempt of providing the generalization bound for the meta learning was done by A. Pentina and C. Lambert for life learning (prephase of meta learning). Recall that in lifelong learning, the objective was to learn the future scenario based on the past experiences and to do well in future, unobserved data. In this paper, it proposed PAC Bayesian generalization bound for lifelong learning that allows quantifying the relation between the expected loss on a future learning task to the average loss on the observed tasks.

It turned out that these bounds offer "unified view on existing paradigms for transfer learning." Also, these bounds can be used to derive two principled algorithms.

The paper explored the PAC Bayes bound under two assumptions: 1. the solution vector has a general form in that a single parameter vector plus a small-specific perturbation (parameter transfer); 2. the solutions can differ significantly but they all lie in the common subspace of low dimension (representation transfer).

According to Theorem 1:

$$\mathbb{E}_{h\sim Q}[R(h)] \leq \mathbb{E}_{h\sim Q}[R_n(D,h)] + \sqrt{\frac{KL(Q\|P)+\log\frac{1}{\delta}+\log m+2}{2m-1}}$$

where w_{pr} are the weights of the observed tasks so that we are to minimize w_{pr} and R is the risk of a hypothesis over a datasets.

This has a closed form solution:

$$\forall w_{\mathscr{Q}} \quad \frac{1}{2}\mathop{\mathbf{E}}_{(t,S_t)(x,y)\sim D_t}\min\left\{\left(y-\langle A_t w_{\mathscr{Q}}+b_t,x\rangle\right)^2,1\right\} \leq \frac{\sqrt{n\bar{m}}+1}{2\sigma n\sqrt{\bar{m}}}\left\|w_{\mathscr{Q}}\right\|^2+\frac{1}{2n\sqrt{\bar{m}}}\sum_{i=1}^{n}\left\|(A_i-I_d)\right.$$
$$\left.\mathrm{w}_{\mathscr{Q}}+b_i^2+\frac{1}{n}\sum_{i=1}^{n}\frac{1}{m_i}\sum_{j=1}^{m_i}\left(y_{ij}-\left\langle A_i w_{\mathscr{Q}}+b_i,x_{ij}\right\rangle\right)^2+\text{ const.}\right.$$

For the second assumption that all of the solution vectors lies in low dimensional subspace. In this case, we will try to minimize

$$w_Q = \underset{w}{\operatorname{argmin}} \left(\|w\|^2 + \frac{C}{m} \sum_{i=1}^{m} \left(y_i - \langle w, B^\top x_i \rangle \right)^2 \right)$$

where B is the matrix representing the subspace, such that $B > x$ is the projected representation of the training data in this subspace.

For P, we choose gaussian with zero mean and variance σI_k and for Q, we choose shifted gaussian with variance σI_k and mean w_Q. in this case, we will have $\mathrm{KL}(Q_i(S_i,P)\|P) = 1/2\sigma\|w_i(B)\|$. In combination, we will get

$$\mathrm{er}(M) \leq \widehat{\mathrm{er}}(M) + \frac{1}{2\sigma n\sqrt{\bar{m}}} \sum_{i=1}^{n} \underset{B\sim D(I_k,M)}{\mathbf{E}} \|w_i(B)\|^2 + \mathrm{const} = \frac{1}{n} \sum_{i=1}^{n} \underset{B\sim\mathscr{Q}}{\mathbf{E}} \left\{ \widehat{\mathrm{er}}\left(w_i(B)\right) + \frac{1}{2\sigma\sqrt{\bar{m}}} \|w_i(B)\|^2 \right\},$$

$$w_i(B) = \frac{C}{m_i} \left(I_k + \frac{C}{m_i} B^\top X_i X_i^\top B \right)^{-1} B^\top X_i Y_i$$

where M is the objective function for learning.

In this case, their main result is a generalization bound in PAC-Bayesian framework. On the one hand, those bounds can be applied to recover two cases of transfer learning: the transfer of classifier parameters and the transfer of subspace. On the other hand, it helps to derive principled algorithm for lifelong learning that achieve results with manually designed methods.

Furthermore, it can be applied to study the implicit assumptions of possible learning methods. One of the future directions would be, instead of the condition of the tasks are i.i.d, we could relax it to be other specialized conditions. For instance, the direction of learning tasks of continuously improving difficulty.

2.3.2 Meta-Learning by Adjusting Priors Based on Extended PAC-Bayes Theory[2]

Following on the step of [25], Ron Amit and Ron Meir, in [2], developed a gradient-based algorithm which minimizes an objective function derived from the bounds and demonstrate its effectiveness numerically with deep neural networks. In this case, they can derive a tighter bound that extends a single task PAC-Bayes bounds to meta learning setup.

For a single task problem, let expected error be $er(h,D) = E_{z\sim D} l(h,z)$ and the empirical error be $\hat{er}(h,S) = 1/m \sum_{j=1}^{m} l(h,z_i)$.

According to McAllester's single task bound, we have

$$\mathbb{E}_{h\sim Q}[R(h)] \leq \mathbb{E}_{h\sim Q}[R_n(D,h)] + \sqrt{\frac{KL(Q\|P) + \log\frac{m}{\delta}}{2(m-1)}}$$

According to Pentina & Lambert(2014), a meta learning formation is off the form $S_1, S_2, ...S_n$ corresponding to n different tasks. The goal of the meta learner is to extract information from the previous tasks and make a prediction on the new task.

To transfer the PAC-Bayes formula to the meta learning case, P would become a "hyper prior" $\mathscr{P}$ and Q will become "hyper posterior" $\mathscr{Q}$. When encountering a new task, the learner samples a prior from the hyper posterior Q(P).

Therefore, the transfer error would be:

$$er(Q,r) = E_{P\sim Q} er(P,r)$$

and the empirical multi-task error would be:

$$\widehat{er}(\mathscr{Q}, S_1, \ldots, S_n) \triangleq \underset{P\sim\mathscr{Q}}{\mathbb{E}} \frac{1}{n} \sum_{i=1}^{n} \widehat{er}\left(Q\left(S_i,P\right), S_i\right)$$

Combined with the above, the meta learning PAC-Bayes Bound would be:

$$\mathrm{er}(\mathscr{Q},\tau) \leq \frac{1}{n} \sum_{i=1}^{n} \underset{P\sim\mathscr{Q}}{\mathbb{E}} \widehat{er}_i(Q_i,S_i) + \frac{1}{n} \sum_{i=1}^{n} \sqrt{\frac{D(\mathscr{Q}\|\mathscr{P}) + \underset{P\sim\mathscr{Q}}{\mathbb{E}} D(Q_i\|P) + \log\frac{2nm_i}{\delta}}{2(m_i-1)}} + \sqrt{\frac{D(\mathscr{Q}\|\mathscr{P}) + \log\frac{2n}{\delta}}{2(n-1)}}$$

$Q(S_i, P)$ is the hyper posterior resulting from hyper-prior P and dataset S_i, D($Q||P$) is the KL divergence between P and Q.

This formula provided a tighter bound since they applied a single-task PAC-Bayes theorem to bound the expected error in each task separately, so that this formula takes into account of the number of sample in the observed task. In this case, they have attested rigorous performance bounds and demonstrated tighter bounds.

2.3.3 Transfer Meta-Learning: Information-Theoretic[16]

Then, in 2020, S. T. Jose, O. Simeone, G. Durisi proposed novel PAC-Bayesian bounds that generalize prior works on transfer learning and quantify the impact of the meta-environment shift through the log-likelihood ratio of the source and target task distributions. Through these bounds, they introduced a novel meta-learning algorithm, termed IMRM.

To start, the empirical weighted average of per-task test loss of meta-learning set was defined as:

$$\mathscr{L}_{t,g}\left(u \mid Z_{1:N}^{M}, T_{1:N}\right) = \frac{\alpha}{\beta N}\sum_{i=1}^{\beta N} L_g\left(u \mid Z_i^M, T_i\right) + \frac{1-\alpha}{(1-\beta)N}\sum_{i=\beta N+1}^{N} L_g\left(u \mid Z_i^M, T_i\right)$$

For some hyper parameter $\alpha, \beta \in [0, 1]$, where $T_{1:N}$ denote N different tasks, $Z_{1:N}^{M}$ denote N different tasks, $L_g(u\|Z_i^M, T_i)$ is defined as:

$$L_g(u\|T, Z^M) = E_{P_{W\|Z^M},u}[L_g(W\|T)]$$

Therefore, the transfer meta-generalization gap can be defined as:

$$\mathbb{E}_{P_{U|Z_{1:N}^{M}}}\left[\Delta\mathscr{L}'\left(U \mid Z_{1:N}^{M}\right)\right] =$$
$$\mathbb{E}_{P_{U|Z_{1:N}^{M}}}\left[\left(\mathscr{L}_g'(U) - \mathscr{L}_{t,g}\left(U \mid Z_{1:N}^{M}, T_{1:N}\right)\right) + \left(\mathscr{L}_{t,g}\left(U \mid Z_{1:N}^{M}, T_{1:N}\right) - \mathscr{L}_t\left(U \mid Z_{1:N}^{M}\right)\right)\right]$$

For any α, β in [0, 1] and σ is the variance of posterior, $P_{U|Z_{1:N}^{M}}$ is the hyper prior given the previous $Z_{1:N}^{M}$ tasks and datasets, U indicates any hyper parameters.

Then, they applied this PAC-Bayes meta generalization gap to invent a new algorithm called IMRM.

Denote as the meta-training loss regularized by the average KL divergence. The IMRM meta-learner is then defined as any algorithm that solves the optimization problem, with probability 1 - δ_{T_i}, M is the number of samples:

$$\left|\mathbb{E}_{P_{U|Z_{1:N}^{M}}}\left[\Delta\mathscr{L}'\left(U \mid Z_{1:N}^{M}\right)\right]\right| \leq \sqrt{2\sigma^2\left(\frac{\alpha^2}{\beta N} + \frac{(1-\alpha)^2}{(1-\beta)N}\right)\left(\sum_{i=1}^{\beta N}\log\frac{P_T(T_i)}{P_T'(T_i)} + D\left(P_{U|Z_{1:N}^{M}}\|Q_U\right) + \log\frac{2}{\delta}\right)}$$
$$+\frac{\alpha}{\beta N}\sum_{i=1}^{\beta N}\sqrt{\frac{2\delta_{T_i}^2}{M}\left(D\left(P_{U|Z_{1:N}^{M}}\|Q_U\right) + \mathbb{E}_{P_{U|Z_{1:N}^{M}}}\left[D\left(P_{W|U,Z_i^M}\|Q_{W|U}\right)\right] + \log\frac{4\beta N}{\delta}\right)}$$
$$+\frac{1-\alpha}{(1-\beta)N}\sum_{i=\beta N+1}^{N}\sqrt{\frac{2\delta_{T_i}^2}{M}\left(D\left(P_{U|Z_{1:N}^{M}}\|Q_U\right) + \mathbb{E}_{P_{U|Z_{1:N}^{M}}}\left[D\left(P_{W|U,Z_i^M}\|Q_{W|U}\right)\right] + \log\frac{4(1-\beta)N}{\delta}\right)}$$

where $U = argmin L_t(u\|Z_{1:N}^{M})$, for any hyper parameter U and W.

They leveraged the derived PAC-Bayesian bound to propose a new meta-learning algorithm for transfer meta-learning.

2.3.4 PAC-Bayes bounds for with data-dependent prior[21]

Following on this trace, Tianyu Liu, Jie Lu and Guangquan Zhang developed three noval generalization error bounds for meta-learning based on PAC-Bayes relative entropy bound. Also, PAC-Bayes boudn for meta-learning with data-dependent prior, using the empirical risk minimization method.

First, for the classical bayes bound:

For the three novel bound that they proposed, they are meta-learning PAC-Bayes λ bound:

$$\mathrm{er}(\mathscr{Q},\tau) \leq \frac{1}{n}\sum_{i=1}^{n}\frac{1}{(1-\lambda/2)^2}\mathop{\mathbb{E}}_{P\sim\mathscr{Q}}\hat{er}(Q,S_i) + \frac{1}{n}\sum_{i=1}^{n}\frac{D(\mathscr{Q}\|\mathscr{P})+\mathop{\mathbb{E}}_{P\sim\mathscr{Q}}D(Q\|P)+\log\frac{4n\sqrt{m_i}}{\delta}}{} + \sqrt{\frac{1}{2(n-1)}\left(D(\mathscr{Q}\|\mathscr{P})+\log\frac{2n}{\delta}\right)}$$

For any λ in [0, 1], τ appears in the classical bayes bound, *er* are the expected error, while $\hat{er}$ is the empirical error.

Meta-Learning PAC-Bayes quadratic bound:

$$\mathrm{er}(\mathscr{Q},\tau) \leq \frac{1}{n}\sum_{i=1}^{n}\left(\sqrt{\mathop{\mathbb{E}}_{P\sim\mathscr{Q}}\hat{er}(Q,S_i)+\varepsilon_i}+\sqrt{\varepsilon_i}\right)^2 + \sqrt{\frac{1}{2(n-1)}\left(D(\mathscr{Q}\|\mathscr{P})+\log\frac{2n}{\delta}\right)}$$

where $\varepsilon_i = \frac{1}{m_i}$ and meta-learning PAC-Bayes variational bound:

$$\mathrm{er}(\mathscr{Q},\tau) \leq \frac{1}{n}\sum_{i=1}^{n}\mathop{\mathbb{E}}_{P\sim\mathscr{Q}}\hat{er}(Q,S_i) + \frac{1}{n}\sum_{i=1}^{n}\min\left(\varepsilon_i+\sqrt{\varepsilon_i\left(\varepsilon_i+2\mathop{\mathbb{E}}_{P\sim\mathscr{Q}}\hat{er}(Q,S_i)\right)},\sqrt{\frac{\varepsilon_i}{2}}\right) + \sqrt{\frac{1}{2(n-1)}\left(D(\mathscr{Q}\|\mathscr{P})+\log\frac{2n}{\delta}\right)}$$

for any $\delta \in (0,1]$, $\lambda \in (0,2]$ and m_i is number of sample.

For the PAC-Bayes bound theory, the generalization error upper bound depends mainly on the regularization item involving the distance between prior distribution P and posterior distribution Q. However, the prior distribution is chosen randomly, with a view to measuring the parameter space. In order to solve this issue, they proposed an ERM approach on part of the training samples.

In practise, one can train probability neural network by minimizing the generalization error upper bound. That is, during the training process, the learner aimed to minimize the PAC-bayes generalization error bound.

The above are the three bounds derived from the PAC-Bayes relative entropy bound. First, it was the meta-learning PAC-Bayes λ bound and meta-learning quandratic bounds and meta-learning PAC-Bayes variational bounds. Next, in order for a better convergence ability, meta-learning PAC-Bayes bounds with data-dependent prior are also introduced.

For future work, more different prior distributions can be done and other ways to optimize KL can be done.

PART III: SIMULATION FOR PAC-BAYES BACKPROP

During this section, we proposed to replace the datasets applied by the paper "Tighter Risk Certificates for Neural Networks." That is, in order test whether those datasets working in more general level, we replaced MNIST to fashion-MNIST, a larger and more complex dataset about clothes. [37] fashion-MNIST contain fashion images from 10 fashion categories, compared to 10 numbers from MNIST. In that case, by running this dataset, we can examine how the generalization error bound in the PBB algorithm could generalize to other datasets.

3.1 Experiment Setup

We set the kl-penalty = 0.1, learning-rate δ = 0.001, Monte Carlo model samples m=150.000 as the oringinal paper. Then, we ran the experiment for $f_{quad}, f_{lambda}, f_{classic}$, three different objective functions.

3.2 Metric Reported

We applied the same trick proposed in the paper that compress the fashion mnist datasets into 0-1 label. In our experiment, we reported both "learnt" and "random" prior of the new datasets. We reported Kullback–Leibler penalty (KL-penalty), 0-1 loss, risk certificate (risk-CE), risk for 0-1 loss (risk-01). Also, we reported for 0-1 loss and cross entropy for the stochastic predictor.

3.3 Architecture

Our experiment composed of a fully convolutional Neural Network composed of 9 layers, 6 CNN layers plus 3 fully connected linear layers. Between the layer, we applied ReLU as the activation function, PNN that composed of 2 probability convolutional networks and 2 probability linear layers. For all of those layers, we ran 10 epochs.We trained fashion-MNIST datasets with train-test split of 6:1. (60000 samples as our training datasets and 10000 samples for testing).

3.4 Experimental Results

Below are the results from our experiments:
Random fully connected neural network

Objective	l^{x-e}	l^{01}	L_S^{x-e}	L_S^{01}	Stc x-e	Stc 01
f_{lambda}	0.03125	0.11265	0.34310	0.06329	0.28080	0.00220
$f_{classic}$	0.01067	0.15997	0.52280	0.09413	0.36350	0.00326
f_{PBB}	0.14315	0.07206	0.20850	0.04465	0.18600	0.00143
f_{quad}	0.03000	0.14650	0.47250	0.08529	0.34370	0.00297

Learnt fully connected neural network

Objective	l^{x-e}	l^{01}	L_S^{x-e}	L_S^{01}	Stc x-e	Stc 01
f_{lambda}	0.00101	0.05643	0.17533	0.03958	0.14950	0.03366
$f_{classic}$	0.00002	0.06073	0.04218	0.15290	0.03668	0.13920
f_{PBB}	0.01372	0.05175	0.03579	0.14270	0.03032	0.13270
f_{quad}	0.00007	0.05974	0.04134	0.15220	0.03601	0.13900

Random convolutional neural network

Objective	l^{x-e}	l^{01}	L_S^{x-e}	L_S^{01}	Stc x-e	Stc 01
f_{lambda}	0.02637	0.10882	0.08104	0.32400	0.06251	0.26750
$f_{classic}$	0.00863	0.15440	0.12291	0.50860	0.09306	0.37020
f_{PBB}	0.12404	0.06590	0.04491	0.18110	0.03803	0.15260
f_{quad}	0.01206	0.13945	0.10946	0.44670	0.08101	0.33110

Learnt convolutional neural network

Objective	l^{x-e}	l^{01}	L_S^{x-e}	L_S^{01}	Stc x-e	Stc 01
f_{lambda}	0.00092	0.04728	0.03024	0.11730	0.01987	0.10500
$f_{classic}$	0.00002	0.05205	0.03357	0.12060	0.02776	0.11040
f_{PBB}	0.01372	0.05175	0.03579	0.14270	0.03032	0.13270
f_{quad}	0.00010	0.05052	0.03225	0.11920	0.02592	0.10880

L^{x-e} is cross entropy training loss, L^{01} is the 01 error, l^{x-e} is the cross entropy risk certificate and l^{01} is the 0-1 risk certificate.

3.5 Discussion and Conclusion

After making a careful comparison between our results and that of the original paper, we made following conclusion: 1) the improvements brought by PBB are the best among these four objectives in this datasets, which is consistent with the result brought by the paper. Therefore, among these four objectives, PBB works the best and achieved the greatest improvements. 2) the risk certificate of both 0-1 loss and cross entropy loss are achieved by $f_{classics}$, which shows an inconsistency with that shown in the original paper. To my guess, objective $f_{classics}$ can be more generalized with random data than f_{PBB} 3) The best training cross entropy loss was achieved by PBB, which is consistent with the results given by the original paper, but there has been a large gap between fashion-MNIST and MNIST, the primary reason would be fashion-MNIST are more complicated and it is more understandable that the error or loss of Fashion-MNIST datasets will be greater; 4) for 0-1 error, PBB would still lead the way, and it was consistent with the result for x-e loss; 5) for testing error presented by stochastic loss, it further shows that PBB achieved the top in these experiments, which adds to the credibility of the results of the original paper since the results can be generalized.

In conclusion, we explored how to PAC-Bayes backdrop optimization to train the Convolutional Neural Network and Probability Neuron Network. One of the take-home message is that we have seen

PBB the greatest improvement among these four objectives in a more generalized dataset than MNIST—-Fashion-MNIST that consists of image data closer to real life. This is a one step further from the original paper that we successfully showed the conclusion from the paper can be generalized, which further adds to the credibility of the paper's conclusion.

REFERENCES

[1] Alquier, P. (2021). User-friendly introduction to pac-bayes bounds. *ArXiv*, abs/2110.11216.

[2] Amit, R. and Meir, R. (2019). Meta-learning by adjusting priors based on extended pac-bayes theory.

[3] Baldassi, C., Ingrosso, A., Lucibello, C., Saglietti, L., and Zecchina, R. (2015). Subdominant dense clusters allow for simple learning and high computational performance in neural networks with discrete synapses. *Physical Review Letters*, 115(12).

[4] Catoni, O. (2009). A pac-bayesian approach to adaptive classification.

[5] Catoni, O. and Picard, J. (2004). Statistical learning theory and stochastic optimization.

[6] Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, Learning, and Games*.

[7] Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. (2019). Entropy-SGD: biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018.

[8] Duchi, J. (2015). Cs229 supplemental lecture notes hoeffding's inequality.

[9] Dziugaite, G. K., Hsu, K., Gharbieh, W., Arpino, G., and Roy, D. M. (2020). On the role of data in pac-bayes bounds.

[10] Dziugaite, G. K. and Roy, D. M. (2017). Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*.

[11] Germain, P., Lacasse, A., Laviolette, F., and Marchand, M. (2009). 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. Association for Computing Machinery.

[12] Hayakawa, S. and Suzuki, T. (2020). On the minimax optimality and superiority of deep neural network learning over sparse parameter spaces. *Neural Networks*, 123:343–361.

[13] Herbrich, R. and Graepel, T. (2001). A pac-bayesian margin bound for linear classifiers: Why svms work. *Advances in neural information processing systems*, pages 224–230.

[14] Hinton, G. E. and van Camp, D. (1993). Keeping the neural networks simple by minimizing the description length of the weights.

[15] Hochreiter, S. and Schmidhuber, J. (1997). Flat minima. *Neural computation*, 9(1):1–42.

[16] Jose, S. T., Simeone, O., and Durisi, G. (2020). Transfer meta-learning: Information-theoretic bounds and information meta-risk minimization.

[17] Langford, J. and Blum, A. (1999). Microchoice bounds and self bounding learning algorithms. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, COLT '99, page 209–214, New York, NY, USA. Association for Computing Machinery.

[18] Langford, J. and Caruana, R. (2002). (not) bounding the true error. *Advances in Neural Information Processing Systems*, 2:809–816.

[19] Langford, J. and Seeger, M. (2001). Bounds for averaging classifiers.

[20] Letarte, G., Germain, P., Guedj, B., and Laviolette, F. (2020). Dichotomize and generalize: Pac-bayesian binary activated deep neural networks.

[21] Liu, T., Lu, J., Yan, Z., and Zhang, G. (2021). Pac-bayes bounds for meta-learning with data-dependent prior.

[22] Maurer, A. (2004). A note on the pac bayesian theorem.

[23] McAllester, D. A. (1998). Some pac-bayesian theorems. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, COLT' 98, page 230–234, New York, NY, USA. Association for Computing Machinery.

[24] McAllester, D. A. (1999). Pac-bayesian model averaging. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, COLT '99, page 164–170, New York, NY, USA. Association for Computing Machinery.

[25] Pentina, A. and Lampert, C. H. (2014). A pac-bayesian bound for lifelong learning.

[26] Pérez-Ortiz, M., Rivasplata, O., Shawe-Taylor, J., and Szepesvári, C. (2021). Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22.

[27] Rivasplata, O., Tankasali, V. M., and Szepesvari, C. (2019). Pac-bayes with backprop.

[28] Seeger, M. (2002). Pac-bayesian generalization error bounds for gaussian process classification. *Journal of Machine Learning Research*, 3.

[29] Shawe-Taylor, J. and Williamson, R. C. (1997). A pac analysis of a bayesian estimator. In *COLT '97*.

[30] Sirignano, J. and Spiliopoulos, K. (2019). Mean field analysis of neural networks: A law of large numbers.

[31] Sompolinsky, H., Crisanti, A., and Sommers, H. J. (1988). Chaos in random neural networks. *Phys. Rev. Lett.*, 61:259–262.

[32] Suzuki, T. (2020). Generalization bound of globally optimal non-convex neural network training: Transportation map estimation by infinite dimensional langevin dynamics.

[33] Thiemann, N., Igel, C., Wintenberger, O., and Seldin, Y. (2017). A strongly quasiconvex pac-bayesian bound.

[34] Tolstikhin, I. and Seldin, Y. (2013). Pac-bayes-empirical-bernstein inequality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'13, page 109–117, Red Hook, NY, USA. Curran Associates Inc.

[35] van Erven, T. (2014). Pac-bayes mini-tutorial: A continuous union bound.

[36] Wellner, J. A. (2018). The cramer-chernoff method... and some exponential bounds.

[37] Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.