Generalization Bound via PAC-Bayes

A Refined Hierarchy of Hypothesis Class

Pingbang Hu

University of Illinois Urbana-Champaign

July 24, 2024

Table of Content

Table of Content

If we look back at what we have done:

As previously seen (Throughout the book [SB14]. . .)

We characterize the notion of learnability by uniform convergence of a hypothesis class $\mathcal{H}$.[1]

[1] There are other notions like stability (omitted), compressibility (last time), etc.

Recap: Uniform Convergence and Beyond

If we look back at what we have done:

As previously seen (Throughout the book [SB14]...)

We characterize the notion of learnability by uniform convergence of a hypothesis class $\mathcal{H}$.[1]

However, this requirement might be too strong:

Observe

All examples given are simple (even finite) hypothesis classes! What about neural networks?

[1] There are other notions like stability (omitted), compressibility (last time), etc.

Recap: Uniform Convergence and Beyond

If we look back at what we have done:

As previously seen (Throughout the book [SB14]...)

We characterize the notion of learnability by uniform convergence of a hypothesis class $\mathcal{H}$.[1]

However, this requirement might be too strong:

Observe

All examples given are simple (even finite) hypothesis classes! What about neural networks?

Intuition (Going beyond uniformality)

What if we know some hypotheses are unlikely to appear? I.e., how to encode biases in $\mathcal{H}$?

- *Minimum Description Length (MDL) and Occam's razor principles do exactly this.*

[1] There are other notions like stability (omitted), compressibility (last time), etc.

The Structure of this Talk

Goal

Introduce non-vacuous generalization bounds for neural networks.

The Structure of this Talk

Goal

Introduce non-vacuous generalization bounds for neural networks.

To achieve this, we will focus on:

- *Naive PAC-Bayes Bound* [SB14]: See how does the classical PAC-Bayes theory work.
- *PAC-Bayes Bound on NNs* [NBS18]: Get a generalization bound of NNs the first time!!

The Structure of this Talk

Goal

Introduce non-vacuous generalization bounds for neural networks.

To achieve this, we will focus on:

- *Naive PAC-Bayes Bound* [SB14]: See how does the classical PAC-Bayes theory work.
- *PAC-Bayes Bound on NNs* [NBS18]: Get a generalization bound of NNs the first time!!

If we get time, we will go beyond the above and see (glance):

- *Rademacher Bound on NNs*: Learn about the classical approaches.
- *Remove the Blow-Up* [GRS19]: First class of NNs with **independent-size** error.

Table of Content

From the Bayesian perspective, the prior knowledge can be described as *prior distribution* P.

- *Prior*: Consider a probability distribution P over $\mathcal{H}$;
- *Posterior*: The learning algorithm updates P to produce a posterior distribution Q on $\mathcal{H}$.

Example (Minimum Description Length)

The probability (density) $P(h)$ of $h \in \mathcal{H}$ is proportional to its minimum description length.

From the Bayesian perspective, the prior knowledge can be described as *prior distribution* P.

- *Prior*: Consider a probability distribution P over $\mathcal{H}$;
- *Posterior*: The learning algorithm updates P to produce a posterior distribution Q on $\mathcal{H}$.

Example (Minimum Description Length)

The probability (density) $P(h)$ of $h \in \mathcal{H}$ is proportional to its minimum description length.

For supervised learning, where $\mathcal{H} = \{h \colon \mathcal{X} \to \mathcal{Y}\}$, one can interpret Q as:

1. whenever a new instance $\boldsymbol{x} \in \mathcal{X}$ arrives,
2. pick $h \sim Q$, and output $h(\boldsymbol{x})$.

Given a data distribution $\mathcal{D}$, a sampled dataset $S \sim \mathcal{D}^m$, and a hypothesis class $\mathcal{H}$, consider

- *Prior* and *Posterior*: P and Q over $\mathcal{H}$, where Q comes from some learning algorithms.

[2] There is a typo in [SB14]: this should be the correct form. Hence, the constant later will vary.

Basic Notation and Lemma

Given a data distribution $\mathcal{D}$, a sampled dataset $S \sim \mathcal{D}^m$, and a hypothesis class $\mathcal{H}$, consider

- *Prior* and *Posterior*: P and Q over $\mathcal{H}$, where Q comes from some learning algorithms.
- *Loss*: the loss of Q on an example z is defined as $\ell(Q, z) \coloneqq \mathbb{E}_{h\sim Q}[\ell(h, z)]$:
  - *Generalized Loss*: $L_{\mathcal{D}}(Q) \coloneqq \mathbb{E}_{h\sim Q}[L_{\mathcal{D}}(h)]$, where $L_{\mathcal{D}}(h) \coloneqq \mathbb{E}_{z\sim\mathcal{D}}[\ell(h, z)]$.
  - *Empirical Loss*: $L_S(Q) \coloneqq \mathbb{E}_{h\sim Q}[L_S(h)]$, where $L_S(h) = \frac{1}{m}\sum_{z\in S}\ell(h, z)$.

[2] There is a typo in [SB14]: this should be the correct form. Hence, the constant later will vary.

Basic Notation and Lemma

Given a data distribution $\mathcal{D}$, a sampled dataset $S \sim \mathcal{D}^m$, and a hypothesis class $\mathcal{H}$, consider

- *Prior* and *Posterior*: P and Q over $\mathcal{H}$, where Q comes from some learning algorithms.
- *Loss*: the loss of Q on an example z is defined as $\ell(Q, z) \coloneqq \mathbb{E}_{h\sim Q}[\ell(h, z)]$:
  - *Generalized Loss*: $L_{\mathcal{D}}(Q) \coloneqq \mathbb{E}_{h\sim Q}[L_{\mathcal{D}}(h)]$, where $L_{\mathcal{D}}(h) \coloneqq \mathbb{E}_{z\sim \mathcal{D}}[\ell(h, z)]$.
  - *Empirical Loss*: $L_S(Q) \coloneqq \mathbb{E}_{h\sim Q}[L_S(h)]$, where $L_S(h) = \frac{1}{m}\sum_{z\in S} \ell(h, z)$.
- *KL-divergence*: $D_{\mathrm{KL}}(P_1 \| P_2) \coloneqq \mathbb{E}_{h\sim P_1}[\ln(P_1(h)/P_2(h))]$ for two distributions P_1, P_2.

That's all notations and definitions we need.

[2] There is a typo in [SB14]: this should be the correct form. Hence, the constant later will vary.

Basic Notation and Lemma

Given a data distribution $\mathcal{D}$, a sampled dataset $S \sim \mathcal{D}^m$, and a hypothesis class $\mathcal{H}$, consider

- *Prior* and *Posterior*: P and Q over $\mathcal{H}$, where Q comes from some learning algorithms.
- *Loss*: the loss of Q on an example z is defined as $\ell(Q, z) \coloneqq \mathbb{E}_{h\sim Q}[\ell(h, z)]$:
  - *Generalized Loss*: $L_{\mathcal{D}}(Q) \coloneqq \mathbb{E}_{h\sim Q}[L_{\mathcal{D}}(h)]$, where $L_{\mathcal{D}}(h) \coloneqq \mathbb{E}_{z\sim\mathcal{D}}[\ell(h, z)]$.
  - *Empirical Loss*: $L_S(Q) \coloneqq \mathbb{E}_{h\sim Q}[L_S(h)]$, where $L_S(h) = \frac{1}{m}\sum_{z\in S}\ell(h, z)$.
- *KL-divergence*: $D_{\mathrm{KL}}(P_1 \| P_2) \coloneqq \mathbb{E}_{h\sim P_1}[\ln(P_1(h)/P_2(h))]$ for two distributions P_1, P_2.

That's all notations and definitions we need. One additional lemma we need is the following.

Lemma (Two-sided bound[2])

Let X be a random variable with $\mathbb{P}(|X| \geq \epsilon) \leq e^{-2m\epsilon^2}$ for $\epsilon > 0$. Then $\mathbb{E}[e^{2(m-1)X^2}] \leq 2m$.

[2] There is a typo in [SB14]: this should be the correct form. Hence, the constant later will vary.

Table of Content

Main Theorem I

Theorem (PAC-Bayes bound)

Consider a loss ℓ bounded in $[0, 1]$ and let $\delta \in (0, 1)$. With probability at least $1 - \delta$ over $S = \{z_i\}_{i=1}^m \sim \mathcal{D}^m$, for all distribution Q over $\mathcal{H}$, we have

$$L_{\mathcal{D}}(Q) \leq L_S(Q) + \sqrt{\frac{D_{KL}(Q \| P) + \ln(2m/\delta)}{2(m-1)}}.$$

We observe the following:

Problem (How useful is it?)

- *It doesn't care about the learning algorithm;*
- *It depends on our prior knowledge P...*

Main Theorem II

Proof.

We want to bound $\Delta(h) \coloneqq L_{\mathcal{D}}(h) - L_S(h)$. Consider

$$f(S) \coloneqq \sup_Q \left(2(m-1)\mathbb{E}_{h\sim Q}[\Delta^2(h)] - D_{\mathrm{KL}}(Q\|P)\right).$$

From Markov's inequality, for any $f(S)$, as $e^{f(S)} \geq 0$,

$$\mathbb{P}_S(f(S) \geq \epsilon) = \mathbb{P}_S(e^{f(S)} \geq e^{\epsilon}) \leq \frac{\mathbb{E}_S[e^{f(S)}]}{e^{\epsilon}}.$$

If we can show[3] $\mathbb{E}_S[e^{f(S)}] \leq 2m$, we get $\mathbb{P}_S(f(S) \geq \epsilon) \leq 2m/e^{\epsilon} =: \delta$, i.e., $\epsilon \coloneqq \ln(2m/\delta)$.

$\Rightarrow$ W.p. $\geq 1-\delta$, for all Q, $2(m-1)\mathbb{E}_{h\sim Q}[\Delta^2(h)] - D_{\mathrm{KL}}(Q\|P) \leq \epsilon = \ln(2m/\delta)$.

The proof is complete by noticing $(\mathbb{E}[\Delta(h)])^2 \leq \mathbb{E}[\Delta^2(h)]$ from Jensen's inequality. $\square$

Main Theorem III

Next, we show $\mathbb{E}_S[e^{f(S)}] \leq 2m$. Recall that $f(S) = \sup_Q(2(m-1)\mathbb{E}_{h\sim Q}[\Delta^2(h)] - D_{\mathrm{KL}}(Q\|P))$.

Proof.

Fix some S, then by definition, $2(m-1)\mathbb{E}_{h\sim Q}[\Delta^2(h)] - D_{\mathrm{KL}}(Q\|P)$ is just

$$\mathop{\mathbb{E}}_{h\sim Q}\left[\ln(e^{2(m-1)\Delta^2(h)}P(h)/Q(h))\right] \leq \ln \mathop{\mathbb{E}}_{h\sim Q}[e^{2(m-1)\Delta^2(h)}P(h)/Q(h)] = \ln \mathop{\mathbb{E}}_{h\sim P}[e^{2(m-1)\Delta^2(h)}],$$

hence $\mathbb{E}_S[e^{f(S)}] \leq \mathbb{E}_S[\mathbb{E}_{h\sim P}[e^{2(m-1)\Delta^2(h)}]] = \mathbb{E}_{h\sim P}[\mathbb{E}_S[e^{2(m-1)\Delta^2(h)}]]$. Finally, for all $h \in \mathcal{H}$,

$$\mathbb{P}_S(|\Delta(h)| \geq \epsilon) \leq e^{-2m\epsilon^2} \Rightarrow \mathbb{E}_S[e^{2(m-1)\Delta^2(h)}] \leq 2m$$

from the Hoeffding's inequality and the two-sided bound lemma (with $X := \Delta(h)$). □

[3] The goal is to get rid of $\sup_Q$, i.e., bounding $\mathbb{E}_S[e^{f(S)}]$ by an expression without Q.

Some Consequences

The naive PAC-Bayes bound suggests how we should design our learning algorithm:

Remark (Regularization)

Given a prior P, return a posterior Q that minimizes

$$L_S(Q) + \sqrt{\frac{D_{KL}(Q \| P) + \ln(2m/\delta)}{2(m-1)}}.$$

This rule is similar to the regularized risk minimization principle. That is, we jointly minimize the empirical loss of Q on the sample and the KL-divergence between Q and P.

Table of Content

Setup and Notations

Consider the *k-class classification task* with a d-layer MLP *model* $f_{\boldsymbol{w}} \colon \mathcal{X} \to \mathbb{R}^k$ where

- *Parameter*: $\boldsymbol{w} = \operatorname{vec}(\{W_i\}_{i=1}^d)$ such that $f_{\boldsymbol{w}}(x) = W_d \phi(W_{d-1} \phi(\ldots \phi(W_1 x)))$:
  - ϕ is the ReLU.
  - $f_{\boldsymbol{w}}^i(x)$ is the output of layer i before activation.

Setup and Notations

Consider the *k-class classification task* with a d-layer MLP *model* $f_{\boldsymbol{w}} \colon \mathcal{X} \to \mathbb{R}^k$ where

- *Parameter*: $\boldsymbol{w} = \operatorname{vec}(\{W_i\}_{i=1}^d)$ such that $f_{\boldsymbol{w}}(x) = W_d\phi(W_{d-1}\phi(\ldots\phi(W_1x)))$:
  - ϕ is the ReLU.
  - $f_{\boldsymbol{w}}^i(x)$ is the output of layer i before activation.
- *Width*: maximum number h of output units in each layer.

Setup and Notations

Consider the *k-class classification task* with a d-layer MLP *model* $f_{\boldsymbol{w}}\colon \mathcal{X} \to \mathbb{R}^k$ where

- *Parameter*: $\boldsymbol{w} = \operatorname{vec}(\{W_i\}_{i=1}^d)$ such that $f_{\boldsymbol{w}}(x) = W_d \phi(W_{d-1}\phi(\dots \phi(W_1 x)))$:
  - ϕ is the ReLU.
  - $f_{\boldsymbol{w}}^i(x)$ is the output of layer i before activation.
- *Width*: maximum number h of output units in each layer.
- *Input domain*: $\mathcal{X} \coloneqq \mathcal{X}_B \coloneqq \{x \in \mathbb{R}^n \colon \sum_{i=1}^n x_i^2 \leq B^2\}$.

Setup and Notations

Consider the *k-class classification task* with a d-layer MLP *model* $f_{\boldsymbol{w}}\colon \mathcal{X} \to \mathbb{R}^k$ where

- *Parameter*: $\boldsymbol{w} = \operatorname{vec}(\{W_i\}_{i=1}^d)$ such that $f_{\boldsymbol{w}}(x) = W_d \phi(W_{d-1}\phi(\dots\phi(W_1 x)))$:
  - ϕ is the ReLU.
  - $f_{\boldsymbol{w}}^i(x)$ is the output of layer i before activation.
- *Width*: maximum number h of output units in each layer.
- *Input domain*: $\mathcal{X} \coloneqq \mathcal{X}_B \coloneqq \{x \in \mathbb{R}^n \colon \sum_{i=1}^n x_i^2 \leq B^2\}$.
- *Output domain*: $\mathcal{Y} \coloneqq [k]$, where the predicted class of x by $f_{\boldsymbol{w}}$ is $\arg\max_{i\in[k]} f_{\boldsymbol{w}}(x)[i]$.

Setup and Notations

Consider the *k-class classification task* with a d-layer MLP *model* $f_{\boldsymbol{w}} \colon \mathcal{X} \to \mathbb{R}^k$ where

- *Parameter*: $\boldsymbol{w} = \operatorname{vec}(\{W_i\}_{i=1}^d)$ such that $f_{\boldsymbol{w}}(x) = W_d \phi(W_{d-1}\phi(\dots\phi(W_1 x)))$:
  - ϕ is the ReLU.
  - $f_{\boldsymbol{w}}^i(x)$ is the output of layer i before activation.
- *Width*: maximum number h of output units in each layer.
- *Input domain*: $\mathcal{X} \coloneqq \mathcal{X}_B \coloneqq \{x \in \mathbb{R}^n \colon \sum_{i=1}^n x_i^2 \leq B^2\}$.
- *Output domain*: $\mathcal{Y} \coloneqq [k]$, where the predicted class of x by $f_{\boldsymbol{w}}$ is $\arg\max_{i\in[k]} f_{\boldsymbol{w}}(x)[i]$.

Margin loss: Given a margin $\gamma > 0$, we define

$$\ell(f_{\boldsymbol{w}}, (x, y)) \coloneqq \mathbb{1}\left\{ f_{\boldsymbol{w}}(x)[y] \leq \gamma + \max_{j\neq y} f_{\boldsymbol{w}}(x)[j] \right\}.$$

Table of Content

First Step: Generalization Bound of Deterministic Models I

Lemma (Key lemma)

Let $f_{\boldsymbol{w}}\colon \mathcal{X} \to \mathbb{R}^k$ be a model with parameters $\boldsymbol{w}$, and P be any distribution on $\boldsymbol{w}$, independent of S. For any $\boldsymbol{w}$, consider the posterior $Q(\boldsymbol{w} + \boldsymbol{u})$ where $\boldsymbol{u}$ is random such that

$$\mathbb{P}\left(\max_{x \in \mathcal{X}} \| f_{\boldsymbol{w}+\boldsymbol{u}}(\boldsymbol{x}) - f_{\boldsymbol{w}}(x) \|_\infty < \frac{\gamma}{4}\right) > \frac{1}{2}.$$

Then, for any $\gamma, \delta > 0$, with probability at least $1 - \delta$ over $S \sim \mathcal{D}^m$, for any $\boldsymbol{w}$,

$$L_D^{(0)}(f_{\boldsymbol{w}}) \leq L_S^{(\gamma)}(f_{\boldsymbol{w}}) + \sqrt{\frac{2 D_{KL}(Q \| P) + \ln \frac{8m}{\delta}}{2(m-1)}}.$$

This basically forms our theorem. If we get this, the only job left is to calculate $\mathrm{D}_{\mathrm{KL}}(Q \| P)$.

First Step: Generalization Bound of Deterministic Models II

Proof.

Let $\boldsymbol{w}' := \boldsymbol{w} + \boldsymbol{u} \sim Q(\boldsymbol{w}')$, and consider $\mathcal{C}$ be the set of *perturbation*:

$$\mathcal{C} := \left\{ \boldsymbol{w}' \colon \max_{x \in \mathcal{X}} \| f_{\boldsymbol{w}'}(x) - f_{\boldsymbol{w}}(x) \|_\infty < \frac{\gamma}{4} \right\}.$$

Then, we consider two distributions conditioned on $\mathcal{C}$ and $\mathcal{C}^c$:

$$\widetilde{Q}(\widetilde{w}) := \begin{cases} Q(\widetilde{\boldsymbol{w}})/Z, & \text{if } \widetilde{\boldsymbol{w}} \in \mathcal{C}; \\ 0, & \text{if } \widetilde{\boldsymbol{w}} \in \mathcal{C}^c, \end{cases} \quad \widetilde{Q}^c(\widetilde{w}) := \begin{cases} 0, & \text{if } \widetilde{\boldsymbol{w}} \in \mathcal{C}; \\ Q(\widetilde{\boldsymbol{w}})/(1-Z), & \text{if } \widetilde{\boldsymbol{w}} \in \mathcal{C}^c, \end{cases}$$

and we will primarily work with $\widetilde{Q}$. Note that $Z = \mathbb{P}(\widetilde{w} \in \mathcal{C}) > 1/2$. From the definition of $\mathcal{C}$:

Observe

Perturbation can change the margin between two output units of $f_{\boldsymbol{w}}$ by at most $\gamma/2$.

Proof (Continued).

Rigorously, we have $\max_{i,j\in[k],x\in\mathcal{X}} \left| |f_{\widetilde{\boldsymbol{w}}}(x)[i] - f_{\widetilde{\boldsymbol{w}}}(x)[j]| - |f_{\boldsymbol{w}}(x)[i] - f_{\boldsymbol{w}}(x)[j]| \right| < \gamma/2$. Using this fact, we can conclude that for any perturbation $\widetilde{w} \sim \widetilde{Q}$,

$$L_D^{(0)}(f_{\boldsymbol{w}}) \leq L_D^{(\gamma/2)}(f_{\widetilde{\boldsymbol{w}}}), \quad L_S^{(\gamma/2)}(f_{\widetilde{w}}) \leq L_S^{(\gamma)}(f_{\boldsymbol{w}}).$$

Hence, with probability at least $1-\delta$ over S, from the PAC-Bayes bound,

$$\begin{aligned} L_D^{(0)}(f_{\boldsymbol{w}}) &\leq \mathop{\mathbb{E}}_{\widetilde{\boldsymbol{w}}\sim\widetilde{Q}}[L_D^{(\gamma/2)}(f_{\widetilde{\boldsymbol{w}}})] \\ &\leq \mathop{\mathbb{E}}_{\widetilde{\boldsymbol{w}}\sim\widetilde{Q}}[L_S^{(\gamma/2)}(f_{\widetilde{\boldsymbol{w}}})] + \sqrt{\frac{D_{\mathrm{KL}}(\widetilde{Q}\|P) + \ln\frac{2m}{\delta}}{2(m-1)}} \leq \mathop{\mathbb{E}}_{\boldsymbol{w}\sim Q}[L_S^{(\gamma)}(f_{\boldsymbol{w}})] + \sqrt{\frac{D_{\mathrm{KL}}(\widetilde{Q}\|P) + \ln\frac{2m}{\delta}}{2(m-1)}}. \end{aligned}$$

The only thing left is to replace $\widetilde{Q}$ with Q in D_{KL}.

Proof (Continued).

Recall that $Z := \mathbb{P}(\widetilde{\boldsymbol{w}} \in \mathcal{C})$, and $\widetilde{Q} := Q/Z$ with $\widetilde{Q}^c := Q/(1-Z)$, we have

$$\begin{aligned} D_{\mathrm{KL}}(Q \| P) &= \int_{\widetilde{\boldsymbol{w}} \in \mathcal{C}} Q \ln \frac{Q}{P} \, \mathrm{d}\widetilde{\boldsymbol{w}} + \int_{\widetilde{\boldsymbol{w}} \in \mathcal{C}^c} Q \ln \frac{Q}{P} \, \mathrm{d}\widetilde{\boldsymbol{w}} \\ &= \int_{\widetilde{\boldsymbol{w}} \in \mathcal{C}} \frac{QZ}{Z} \ln \frac{Q}{ZP} + Q \ln Z \, \mathrm{d}\widetilde{\boldsymbol{w}} + \int_{\widetilde{\boldsymbol{w}} \in \mathcal{C}^c} \frac{Q(1-Z)}{1-Z} \ln \frac{Q}{(1-Z)P} + Q \ln(1-Z) \, \mathrm{d}\widetilde{\boldsymbol{w}} \\ &= Z D_{\mathrm{KL}}(\widetilde{Q} \| P) + (1-Z) D_{\mathrm{KL}}(\widetilde{Q}^c \| P) - H(Z), \end{aligned}$$

where $H(Z) = -Z \ln Z - (1-Z) \ln(1-Z)$ is the *entropy* of $\mathrm{Ber}(Z)$. Finally, since $D_{\mathrm{KL}} \geq 0$, and with $Z \in [1/2, 1]$, we have $1 - Z \geq 0$ and $H(Z) \in [0, \ln 2]$,

$$D_{\mathrm{KL}}(\widetilde{Q} \| P) = \frac{1}{Z} \left(D_{\mathrm{KL}}(Q \| P) + H(Z) - (1-Z) D_{\mathrm{KL}}(\widetilde{Q}^c \| P) \right) \leq 2 D_{\mathrm{KL}}(Q \| P) + 2 \ln 2.$$

This completes the proof. □

Table of Content

Second Step: Perturbation Bound for NNs I

Lemma (Perturbation bound)

For any $B, d > 0$, let $f_{\mathbf{w}} \colon \mathcal{X}_B \to \mathbb{R}^k$ be a d-layer MLP. Then for any $\mathbf{w}$, and $x \in \mathcal{X}_{B,n}$, and any perturbation $\boldsymbol{u} = \operatorname{vec}(\{U_i\}_{i=1}^d)$ such that $\|U_i\|_2 \leq \|W_i\|_2/d$, the change in the output of the network satisfies

$$\|f_{\mathbf{w}+\boldsymbol{u}}(x) - f_{\mathbf{w}}(x)\|_2 \leq eB\left(\prod_{i=1}^d \|W_i\|_2\right)\sum_{i=1}^d \frac{\|U_i\|_2}{\|W_i\|_2}.$$

Intuition

This characterizes the change in the output of a network w.r.t. perturbation of its weight, helping us calculate the KL-divergence term in the previous bound given a margin budgets γ.

Proof.

Let $\Delta_i := \|f^i_{\boldsymbol{w}+\boldsymbol{u}}(x) - f^i_{\boldsymbol{w}}(x)\|_2$. It suffices to show that for all $i \geq 0$,

$$\Delta_i \leq \left(1 + \frac{1}{d}\right)^i \left(\prod_{j=1}^{i} \|W_j\|_2\right) \|x\|_2 \sum_{j=1}^{i} \frac{\|U_j\|_2}{\|W_j\|_2}.$$

For $i = 0$, this is trivial. For any $i \geq 1$, note that $\phi_i(0) = 0$, and it's 1-Lipschitz,

$$\begin{aligned}
\Delta_{i+1} &= \left\|(W_{i+1} + U_{i+1})\phi_i(f^i_{\boldsymbol{w}+\boldsymbol{u}}(x)) - W_{i+1}\phi_i(f^i_{\boldsymbol{w}}(x))\right\|_2 \\
&= \left\|(W_{i+1} + U_{i+1})\left(\phi_i(f^i_{\boldsymbol{w}+\boldsymbol{u}}(x)) - \phi_i(f^i_{\boldsymbol{w}}(x))\right) + U_{i+1}\phi_i(f^i_{\boldsymbol{w}}(x))\right\|_2 \\
&\leq (\|W_{i+1}\|_2 + \|U_{i+1}\|_2)\|\phi_i(f^i_{\boldsymbol{w}+\boldsymbol{u}}(x)) - \phi_i(f^i_{\boldsymbol{w}}(x))\|_2 + \|U_{i+1}\|_2\|\phi_i(f^i_{\boldsymbol{w}}(x))\|_2 \\
&\leq (\|W_{i+1}\|_2 + \|U_{i+1}\|_2)\|f^i_{\boldsymbol{w}+\boldsymbol{u}}(x) - f^i_{\boldsymbol{w}}(x)\|_2 + \|U_{i+1}\|_2\|f^i_{\boldsymbol{w}}(x)\|_2 \\
&= \Delta_i(\|W_{i+1}\|_2 + \|U_{i+1}\|_2) + \|U_{i+1}\|_2\|f^i_{\boldsymbol{w}}(x)\|_2.
\end{aligned}$$

Second Step: Perturbation Bound for NNs III

Proof (Continued).

By the assumption, $\|U_{i+1}\|_2 \leq \|W_{i+1}\|_2/d$, we have

$$
\begin{aligned}
\Delta_{i+1} &\leq \Delta_i(\|W_{i+1}\| + \|U_{i+1}\|_2) + \|U_{i+1}\|_2 \|f_{\boldsymbol{w}}^i(x)\|_2 \\
&\leq \Delta_i \left(1 + \frac{1}{d}\right) \|W_{i+1}\|_2 + \|U_{i+1}\|_2 \|x\|_2 \prod_{j=1}^{i} \|W_j\|_2 \\
&\leq \left(1 + \frac{1}{d}\right)^{i+1} \left(\prod_{j=1}^{i+1} \|W_j\|_2\right) \|x\|_2 \sum_{j=1}^{i} \frac{\|U_j\|_2}{\|W_j\|_2} + \frac{\|U_{i+1}\|_2}{\|W_{i+1}\|_2} \|x\|_2 \prod_{j=1}^{i+1} \|W_j\|_2 \qquad \text{(induction)} \\
&\leq \left(1 + \frac{1}{d}\right)^{i+1} \left(\prod_{j=1}^{i+1} \|W_j\|_2\right) \|x\|_2 \sum_{j=1}^{i+1} \frac{\|U_j\|_2}{\|W_j\|_2}. \qquad \text{(multiply } 2^{\text{nd}} \text{ term with } (1 + 1/d)^{i+1})
\end{aligned}
$$

This concludes the proof as $(1 + 1/d)^d \leq e$ and $x \in \mathcal{X}_B$ (i.e., $\|x\|_2 \leq B$). □

Table of Content

With all the build-up, we can finally prove the following.

Theorem (Generalization Bound for MLPs)

For any $B, d, h > 0$, let $f_{\mathbf{w}} \colon \mathcal{X}_B \to \mathbb{R}^k$ be a d-layer MLP. Then, for any $\delta, \gamma > 0$, with probability at least $1 - \delta$ over $S \sim \mathcal{D}^m$, for any $\mathbf{w}$,

$$L_{\mathcal{D}}^{(0)}(f_{\mathbf{w}}) \leq L_S^{(\gamma)}(f_{\mathbf{w}}) + O\left(\sqrt{\frac{B^2 d^2 h \ln(dh) \prod_{i=1}^d \lVert W_i \rVert_2^2 \sum_{i=1}^d \frac{\lVert W_i \rVert_F^2}{\lVert W_i \rVert_2^2} + \ln \frac{dm}{\delta}}{\gamma^2 m}}\right).$$

We divide the proof into two steps:

1. First, calculate the *maximum allowed perturbation* of parameters to satisfy a given γ.
2. Second, calculate the D_{KL} for this value of the perturbation.

Before we start, we make the following observation:

Observe

Let $\beta := (\prod_{i=1}^{d} \|W_i\|_2)^{1/d}$, *consider a network with “normalized weights”* $\widetilde{W}_i := \beta W_i / \|W_i\|_2$.

$\Rightarrow$ *From homogeneity of ReLU,* $f_{\widetilde{\mathbf{w}}} = f_{\mathbf{w}}$, *hence (empirical & expected) losses are the same.*

Moreover, observe that $\prod_{i=1}^{d} \|W_i\|_2 = \prod_{i=1}^{d} \|\widetilde{W}_i\|_2$, *and* $\|W_i\|_F / \|W_i\|_2 = \|\widetilde{W}_i\|_F / \|\widetilde{W}_i\|_2$,

$\Rightarrow$ *Excess risk is invariant under this transformation:*

$$L_{\mathcal{D}}^{(0)}(f_{\mathbf{w}}) - L_S^{(\gamma)}(f_{\mathbf{w}}) = O\left(\sqrt{\frac{B^2 d^2 h \ln(dh) \prod_{i=1}^{d} \|W_i\|_2^2 \sum_{i=1}^{d} \frac{\|W_i\|_F^2}{\|W_i\|_2^2} + \ln \frac{dm}{\delta}}{\gamma^2 m}}\right).$$

Hence, it suffices to consider normalized weights $\widetilde{\mathbf{w}}$, *i.e.,* $\|\widetilde{W}_i\|_2 = \beta$ *for all* i.

First Step: Applying Perturbation Bound I

Proof.

Let $P = \mathcal{N}(0, \sigma^2 I)$ and $\boldsymbol{u} \sim \mathcal{N}(0, \sigma^2 I)$ with the same σ to be determined, depends on β.

Intuition

However, β is determined by $\boldsymbol{w}$, which is unknown before the training. Hence, we will set σ based on an approximation $\widetilde{\beta}$. I.e., we pre-determine a grid of $\widetilde{\gamma}$'s and their σ, such that

- *each relevant value of β is covered by some $\widetilde{\beta}$ on the grid:*
  - *Covered: $|\beta - \widetilde{\beta}| \leq \beta / d$.*

Finally, we take a union bound over all $\widetilde{\beta}$ on the grid.

For now, consider a fixed $\widetilde{\beta}$ and some $\boldsymbol{w}$ such that $|\beta - \widetilde{\beta}| \leq \beta / d$, hence

$$\frac{1}{e}\beta^{d-1} \leq \widetilde{\beta}^{d-1} \leq e\beta^{d-1}.$$

First Step: Applying Perturbation Bound II

Proof (Continued).

Since $\boldsymbol{u} \sim \mathcal{N}(0, \sigma^2 I)$, the following concentration for the spectral norm of U_i is known:

$$\mathbb{P}_{U_i \sim \mathcal{N}(0,\sigma^2 I)}(\|U_i\|_2 > t) \leq 2he^{-t^2/2h\sigma^2}.$$

Taking a union bound over layers, with probability $\geq 1/2$, $\|U_i\|_2 \leq \sigma\sqrt{2h\ln(4dh)} =: t$. Then

$$\begin{aligned}\max_{x \in \mathcal{X}_B} \|f_{\boldsymbol{w}+\boldsymbol{u}}(x) - f_{\boldsymbol{w}}(x)\|_2 &\leq eB\beta^d \sum_{i=1}^{d} \frac{\|U_i\|_2}{\beta} \qquad \text{(Perturbation bound)} \\ &= eB\beta^{d-1} \sum_{i=1}^{d} \|U_i\|_2 \leq e^2 dB\widetilde{\beta}^{d-1}\sigma\sqrt{2h\ln(4dh)} \leq \frac{\gamma}{4}\end{aligned}$$

where we let $\sigma := \frac{\gamma}{42dB\widetilde{\beta}^{d-1}\sqrt{h\ln(4hd)}}$. Now, we appeal to the key lemma. $\square$

Proof.

With $Q := \boldsymbol{w} + \boldsymbol{u}$, we can already apply the key lemma to get

$$L_D^{(0)}(f_{\boldsymbol{w}}) \leq L_S^{(\gamma)}(f_{\boldsymbol{w}}) + \sqrt{\frac{2D_{\mathrm{KL}}(\boldsymbol{w}+\boldsymbol{u} \| P) + \ln \frac{8m}{\delta}}{2(m-1)}}.$$

Hence, we just need to calculate $D_{\mathrm{KL}}(\boldsymbol{w}+\boldsymbol{u} \| P) = D_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{w}, \sigma^2 I) \| \mathcal{N}(0, \sigma^2 I))$. By a direct calculation, it's bounded above by (you will need to believe me for this one)

$$\begin{aligned} \frac{\|w\|_2^2}{2\sigma^2} &= \frac{42^2 d^2 B^2 \widetilde{\beta}^{2d-2} h \ln(4hd)}{2\gamma^2} \sum_{i=1}^{d} \|W_i\|_F^2 \\ &\leq O\left(B^2 d^2 h \ln(dh) \frac{\beta^{2d}}{\gamma^2} \sum_{i=1}^{d} \frac{\|W_i\|_F^2}{\beta^2} \right) = O\left(B^2 d^2 h \ln(dh) \frac{\prod_{i=1}^{d} \|W_i\|_2^2}{\gamma^2} \sum_{i=1}^{d} \frac{\|W_i\|_F^2}{\|W_i\|_2^2} \right). \end{aligned}$$

Second Step: Applying PAC-Bayes Bound II

Proof (Continued).

Hence, for any $\widetilde{\beta}$, with probability $\geq 1-\delta$, and for all $\boldsymbol{w}$ such that $|\beta-\widetilde{\beta}| \leq \beta/d$, we have

$$L_D^{(0)}(f_{\boldsymbol{w}}) \leq L_S^{(\gamma)}(f_{\boldsymbol{w}}) + O\left(\sqrt{\frac{B^2 d^2 h \ln(dh) \prod_{i=1}^d \|W_i\|_2^2 \sum_{i=1}^d \frac{\|W_i\|_F^2}{\|W_i\|_2^2} + \ln\frac{m}{\delta}}{\gamma^2 m}}\right).$$

Remark

Compared to the theorem, the only difference is $\ln\frac{m}{\delta}$ *v.s.* $\ln\frac{dm}{\delta}$.

To fix this, recall that we still need to take a union bound over $\widetilde{\beta}$'s.

Second Step: Applying PAC-Bayes Bound III

Proof (Continued).

Observe (Non-trivial range)

We only need to consider β *in the range of* $\left(\frac{\gamma}{2B}\right)^{1/d} \leq \beta \leq \left(\frac{\gamma\sqrt{m}}{2B}\right)^{1/d}$*, so to satisfy* $|\beta - \widetilde{\beta}| \leq \beta/d$*, we only need* $|\widetilde{\beta} - \beta| \leq \frac{1}{d}\left(\frac{\gamma}{2B}\right)^{1/d}$ *for* β *in this range.*

This observation leads to the following simple calculation of the cover size:

$$\left(\frac{\gamma\sqrt{m}}{2B}\right)^{1/d} \Big/ \frac{1}{d}\left(\frac{\gamma}{2B}\right)^{1/d} = d \cdot m^{\frac{1}{2d}}.$$

Taking a union bound, the corresponding probability is $\delta' := \delta \cdot d \cdot m^{1/2d}$. Expressing everything in terms of δ', we have $\ln \frac{m}{\delta} = \ln \frac{dm^{1+1/2d}}{\delta'} \approx \ln \frac{dm}{\delta'}$, which completes the proof. $\square$

Table of Content

Road Map

Although the proof is a bit long, but here's the takeaway:

- *PAC-Bayes bound* is applicable to any loss $\ell \in [0, 1]$, independent of *learning algorithms*:

$$L_{\mathcal{D}}(Q) \leq L_S(Q) + \sqrt{\frac{D_{\mathrm{KL}}(Q \| P) + \ln(m/\delta)}{2(m-1)}}.$$

- *Generalization bound* for a d-layer, h-width MLP:

$$L_{\mathcal{D}}^{(0)}(f_{\boldsymbol{w}}) \leq L_S^{(\gamma)}(f_{\boldsymbol{w}}) + O\left(\sqrt{\frac{B^2 d^2 h \ln(dh) \prod_{i=1}^{d} \|W_i\|_2^2 \sum_{i=1}^{d} \frac{\|W_i\|_F^2}{\|W_i\|_2^2} + \ln \frac{dm}{\delta}}{\gamma^2 m}}\right).$$

Road Map

Although the proof is a bit long, but here's the takeaway:

- *PAC-Bayes bound* is applicable to any loss $\ell \in [0, 1]$, independent of *learning algorithms*:

$$L_{\mathcal{D}}(Q) \leq L_S(Q) + \sqrt{\frac{D_{\mathrm{KL}}(Q \| P) + \ln(m/\delta)}{2(m-1)}}.$$

- *Generalization bound* for a d-layer, h-width MLP:

$$L_{\mathcal{D}}^{(0)}(f_{\boldsymbol{w}}) \leq L_S^{(\gamma)}(f_{\boldsymbol{w}}) + O\left(\sqrt{\frac{B^2 d^2 h \ln(dh) \prod_{i=1}^{d} \|W_i\|_2^2 \sum_{i=1}^{d} \frac{\|W_i\|_F^2}{\|W_i\|_2^2} + \ln \frac{dm}{\delta}}{\gamma^2 m}}\right).$$

  - *Key lemma*: For a "robust model" (w.r.t. margin γ), *PAC-Bayes bound* applies.
  - *Perturbation bound*: Provides an analytical bound for the perturbation.
    - $\Rightarrow$ With normal prior and perturbation, the MLP is "robust" enough from *perturbation bound*.
    - $\Rightarrow$ *Key lemma* applies.

A Detailed Look of the Generalization Bound for NNs

Observe (Generalization Bound for NNs)

Let's take a closer look at the bound we get finally:

$$L_{\mathcal{D}}^{(0)}(f_{\mathbf{w}}) \leq L_S^{(\gamma)}(f_{\mathbf{w}}) + O\left(\sqrt{\frac{B^2 d^2 h \ln(dh) \prod_{i=1}^{d} \|W_i\|_2^2 \sum_{i=1}^{d} \frac{\|W_i\|_F^2}{\|W_i\|_2^2} + \ln \frac{dm}{\delta}}{\gamma^2 m}}\right).$$

- *It's independent of the feature dimensions n, as long as $x \in \mathcal{X}$ is bounded (by B).*
- *As $m \to \infty$, if d is fixed, then we're in a good shape: the bounds shrinks linearly.*
- *If d grows, it's likely that $\prod_{i=1}^{d} \|W_i\|_2^2$ dominants $1/m$ since it's an exponential blow-up.*

The last point is why the generalization theory doesn't seem to be useful for *deep* learning.

Table of Content

Several natural questions should arise if you are still awake:

Problem (Natural questions. . .)

1. *Can the PAC-Bayes approach be applied to other tasks?*

What's Next?

Several natural questions should arise if you are still awake:

Problem (Natural questions. . .)

1. *Can the PAC-Bayes approach be applied to other tasks?*
2. *Are there other methods to get a similar bound?*

What's Next?

Several natural questions should arise if you are still awake:

Problem (Natural questions. . .)

1. *Can the PAC-Bayes approach be applied to other tasks?*
2. *Are there other methods to get a similar bound?*
3. *Is the exponential blow-up avoidable?*

What's Next?

Several natural questions should arise if you are still awake:

Problem (Natural questions. . .)

1. *Can the PAC-Bayes approach be applied to other tasks?*
2. *Are there other methods to get a similar bound?*
3. *Is the exponential blow-up avoidable?*

It turns out that, for these questions:

1. ***Yes***! It is extended to *graph neural networks* in particular:
   - Graph classification [LUZ20] and semi-supervised node classification [MDM21].

What's Next?

Several natural questions should arise if you are still awake:

Problem (Natural questions...)

1. *Can the PAC-Bayes approach be applied to other tasks?*
2. *Are there other methods to get a similar bound?*
3. *Is the exponential blow-up avoidable?*

It turns out that, for these questions:

1. ***Yes***! It is extended to *graph neural networks* in particular:
   - Graph classification [LUZ20] and semi-supervised node classification [MDM21].
2. ***Yes***! Classical approach in the context of *statistical learning theory* is well-developed.
   - First formalizes the generalization error as an *empirical process* $\mathbb{P}_n h - \mathbb{P} h$;
   - Then bounds $S_n := \mathbb{E}[\sup_{h\in\mathscr{H}} \sqrt{n}(\mathbb{P}_n h - \mathbb{P} h)]$, leading to a high concentration bound.
   - Bounding S_n often reduces to bounding *VC-dimension* or *Rademacher complexity* of $\mathscr{H}$.

What's Next?

Several natural questions should arise if you are still awake:

Problem (Natural questions. . .)

1. *Can the PAC-Bayes approach be applied to other tasks?*
2. *Are there other methods to get a similar bound?*
3. *Is the exponential blow-up avoidable?*

It turns out that, for these questions:

1. ***Yes***! It is extended to *graph neural networks* in particular:
   - Graph classification [LUZ20] and semi-supervised node classification [MDM21].
2. ***Yes***! Classical approach in the context of *statistical learning theory* is well-developed.
   - First formalizes the generalization error as an *empirical process* $\mathbb{P}_n h - \mathbb{P} h$;
   - Then bounds $S_n := \mathbb{E}[\sup_{h \in \mathscr{H}} \sqrt{n}(\mathbb{P}_n h - \mathbb{P} h)]$, leading to a high concentration bound.
   - Bounding S_n often reduces to bounding *VC-dimension* or *Rademacher complexity* of $\mathscr{H}$.
3. ***Yes*** and ***No***. . .

Dealing with Exponential Blow-Up I

Now, let's formalize the last question regarding the exponential blow-up:

Problem

Under what norm-based constraints of NNs, can we avoid the exponential blow-up?

Intuition (Why norm-based constraints?)

For linear hypothesis class, if $\|\mathbf{w}\| \leq M$ *and* $\|x\| \leq B$*, we have* $L_{\mathcal{D}}(\mathbf{w}) - L_S(\mathbf{w}) \approx O(MB/\sqrt{m})$.

Actually, if one really think about it, $\prod_{i=1}^{d} \|W_i\|$ is unavoidable... but this is fine since:

- constraints of the form $\prod_{i=1}^{d} \|W_i\| \leq R$ is still a form of *norm constraint*.

$\Rightarrow$ The problem becomes trimming down other *trailing factors*.

Dealing with Exponential Blow-Up II

Focus on trailing factors (ignoring $B\prod_{i=1}^{d}\|W_i\|$ in the following):

1. *Rademacher complexity* used to $\approx \widetilde{O}(2^d/\sqrt{m})$ [NTS15]:
   $\Rightarrow$ when $d \geq \Omega(\ln m)$, the bound becomes vacuous.
2. *Rademacher complexity* is later improved to $\approx \widetilde{O}(\sqrt{d^3/m})$ [BFT17]:
   $\Rightarrow$ when $d \geq \Omega(m^{1/3})$, the bound becomes vacuous.
3. Our *PAC-Bayes bound* $\approx \widetilde{O}(\sqrt{d^3 h/m}R)$ [NBS18]:
   $\Rightarrow$ when $d\sqrt[3]{h} \geq \Omega(m^{1/3})$, the bound becomes trivial.

Intuition (This seems to be the best we can hope. . .)

Norm-based constrains reduces the exponential blow-up of the trailing factor to polynomial.

So it seems like our PAC-Bayes bound is doing its best job. . . Can we do better?

Dealing with Exponential Blow-Up III

The answer is yes. The ground-breaking work [GRS19] proves the following:

Theorem (Size-independent Sample Complexity of Neural Networks [GRS19])

It's possible to get rid of both d and h completely, hence obtains a size-independent generalization error bound for a class of norm-base constrained NNs.

Proof idea.

Under some control over any *Schatten norm* of the parameter matrices (e.g., $\|\cdot\|_F$ and $\|\cdot\|_{\mathrm{tr}}$):

Observe (Key observation)

The prediction function computed by such networks can be approximated by the composition of a shallow network and univariate Lipschitz functions.

Then the Rademacher complexity can be bounded nicely. □

References I

[BFT17] Peter Bartlett, Dylan J. Foster, and Matus Telgarsky. *Spectrally-Normalized Margin Bounds for Neural Networks*. Dec. 5, 2017. DOI: 10.48550/arXiv.1706.08498. arXiv: 1706.08498 [cs, stat]. URL: http://arxiv.org/abs/1706.08498 (visited on 06/27/2024). Pre-published.

[GRS19] Noah Golowich, Alexander Rakhlin, and Ohad Shamir. *Size-Independent Sample Complexity of Neural Networks*. Nov. 17, 2019. DOI: 10.48550/arXiv.1712.06541. arXiv: 1712.06541 [cs, stat]. URL: http://arxiv.org/abs/1712.06541 (visited on 11/15/2023). Pre-published.

[LUZ20] Renjie Liao, Raquel Urtasun, and Richard Zemel. *A PAC-Bayesian Approach to Generalization Bounds for Graph Neural Networks*. Dec. 14, 2020. DOI: 10.48550/arXiv.2012.07690. arXiv: 2012.07690 [cs]. URL: http://arxiv.org/abs/2012.07690 (visited on 06/28/2024). Pre-published.

[MDM21] Jiaqi Ma, Junwei Deng, and Qiaozhu Mei. *Subgroup Generalization and Fairness of Graph Neural Networks*. Nov. 30, 2021. DOI: 10.48550/arXiv.2106.15535. arXiv: 2106.15535 [cs]. URL: http://arxiv.org/abs/2106.15535 (visited on 06/10/2023). Pre-published.

[NBS18] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro. *A PAC-Bayesian Approach to Spectrally-Normalized Margin Bounds for Neural Networks*. Feb. 23, 2018. DOI: 10.48550/arXiv.1707.09564. arXiv: 1707.09564 [cs]. URL: http://arxiv.org/abs/1707.09564 (visited on 09/03/2023). Pre-published.

References II

[NTS15] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. *Norm-Based Capacity Control in Neural Networks*. Apr. 14, 2015. DOI: 10.48550/arXiv.1503.00036. arXiv: 1503.00036 [cs, stat]. URL: http://arxiv.org/abs/1503.00036 (visited on 06/27/2024). Pre-published.

[SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. 1st ed. Cambridge University Press, May 19, 2014. ISBN: 978-1-107-05713-5 978-1-107-29801-9. DOI: 10.1017/CBO9781107298019. URL: https://www.cambridge.org/core/product/identifier/9781107298019/type/book (visited on 09/03/2023).