Reading Group: An Introduction to PAC-Bayes

Andrew Y. K. Foong, David R. Burt, Javier Antorán

22nd April, 2021

UNIVERSITY OF CAMBRIDGE

Introduction to PAC Generalisation Bounds

Motivation

Explaining generalisation in deep learning.

Can prove that with high probability, (stochastic) neural networks with millions of weights will generalise.

Motivation

Explaining generalisation in deep learning.

Can prove that with high probability, (stochastic) neural networks with millions of weights will generalise.

Developing novel learning algorithms.

E.g. recent work has suggested modifying the ELBO on PAC-Bayesian grounds, to deal with misspecification.

Motivation

Explaining generalisation in deep learning.

Can prove that with high probability, (stochastic) neural networks with millions of weights will generalise.

Developing novel learning algorithms.

E.g. recent work has suggested modifying the ELBO on PAC-Bayesian grounds, to deal with misspecification.

Relating Bayesian and frequentist learning.

PAC-Bayes can provide a frequentist justification for Bayesian inference, without assuming the model is correct.

Motivation

Explaining generalisation in deep learning.

Can prove that with high probability, (stochastic) neural networks with millions of weights will generalise.

Developing novel learning algorithms.

E.g. recent work has suggested modifying the ELBO on PAC-Bayesian grounds, to deal with misspecification.

Relating Bayesian and frequentist learning.

PAC-Bayes can provide a frequentist justification for Bayesian inference, without assuming the model is correct.

Please ask questions!

PAC Setup

PAC (Probably Approximately Correct) framework is **frequentist**.

PAC Setup

PAC (Probably Approximately Correct) framework is **frequentist**.

Here we consider **supervised learning**.

PAC Setup

PAC (Probably Approximately Correct) framework is **frequentist**.

Here we consider **supervised learning**.

Consider classification with input space X and output space Y.

PAC Setup

PAC (Probably Approximately Correct) framework is **frequentist**.

Here we consider **supervised learning**.

Consider classification with input space X and output space Y.

E.g. $x \in X$ is an image, and $Y = \{-1, +1\}$ is the set of labels.

PAC Setup

PAC (Probably Approximately Correct) framework is **frequentist**.

Here we consider **supervised learning**.

Consider classification with input space X and output space Y.

E.g. $x \in X$ is an image, and $Y = \{-1, +1\}$ is the set of labels.

Let D be some unknown data-generating distribution over $X \times Y$.

PAC Setup

PAC (Probably Approximately Correct) framework is **frequentist**.

Here we consider **supervised learning**.

Consider classification with input space X and output space Y.

E.g. $x \in X$ is an image, and $Y = \{-1, +1\}$ is the set of labels.

Let D be some unknown data-generating distribution over $X \times Y$.

Define a **hypothesis space** as a set H of functions from $X \to Y$.

PAC Setup

- PAC (Probably Approximately Correct) framework is **frequentist**.
- Here we consider **supervised learning**.
- Consider classification with input space X and output space Y.
- E.g. $x \in X$ is an image, and $Y = \{-1, +1\}$ is the set of labels.
- Let D be some unknown data-generating distribution over $X \times Y$.
- Define a **hypothesis space** as a set H of functions from $X \to Y$.

In the PAC setting, we view the learning algorithm as choosing a **hypothesis/predictor** $h \in H$.

The True Risk

How shall we choose the hypothesis $h \in \mathcal{H}$?

- Define a **loss function** $\ell : \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \to \mathbb{R}$.

The True Risk

How shall we choose the hypothesis $h \in H$?

- Define a **loss function** $\ell : X \times Y \times H \to \mathbb{R}$.
- Define the **generalisation risk/true risk** as:

$$R(h) = \mathbb{E}_{(x,y) \sim D}[\ell((x,y),h)].$$

The True Risk

How shall we choose the hypothesis $h \in \mathcal{H}$?

- Define a **loss function** $\ell : \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \to \mathbb{R}$.
- Define the **generalisation risk/true risk** as:

$$R(h) = \mathbb{E}_{(x,y) \sim D}[\ell((x,y),h)].$$

- We commonly consider the **0-1 loss**:

$$\ell((x,y),h) := \mathbb{1}\{h(x) \neq y\},$$

where $\mathbb{1}\{\}$ is the indicator function. Then $R(h)$ is just the error probability.

The True Risk

How shall we choose the hypothesis $h \in H$?

- Define a **loss function** $\ell : X \times Y \times H \to \mathbb{R}$.
- Define the **generalisation risk/true risk** as:

$$R(h) = \mathbb{E}_{(x,y) \sim D}[\ell((x,y),h)].$$

- We commonly consider the **0-1 loss**:

$$\ell((x,y),h) := \mathbb{1}\{h(x) \neq y\},$$

where $\mathbb{1}\{\}$ is the indicator function. Then $R(h)$ is just the error probability.

- We want to choose h such that $R(h)$ is minimised.

The True Risk

How shall we choose the hypothesis $h \in H$?

- Define a **loss function** $\ell : X \times Y \times H \to \mathbb{R}$.
- Define the **generalisation risk/true risk** as:

$$R(h) = \mathbb{E}_{(x,y) \sim D}[\ell((x,y),h)].$$

- We commonly consider the **0-1 loss**:

$$\ell((x,y),h) := \mathbb{1}\{h(x) \neq y\},$$

where $\mathbb{1}\{\}$ is the indicator function. Then $R(h)$ is just the error probability.

- We want to choose h such that $R(h)$ is minimised.
- However, we don't know D, so we cannot compute $R(h)$.

The Empirical Risk

We estimate the true risk by sampling a **dataset** $S = f(x_n; y_n)g_{n=1}^{N} \stackrel{\text{i.i.d.}}{} D$.

The Empirical Risk

We estimate the true risk by sampling a **dataset** $S = f(x_n; y_n)g_{n=1}^{N} \stackrel{\text{i.i.d.}}{} D$.

We use S to choose h_S. (Sometimes suppress S).

The Empirical Risk

We estimate the true risk by sampling a **dataset** $S = \{(x_n, y_n)\}_{n=1}^N \overset{\text{i.i.d.}}{\sim} D$.

We use S to choose h_S. (Sometimes suppress S).

We then compute the **empirical risk** r_S:

$$r_S(h_S) = \frac{1}{N} \sum_{(x,y) \in S} \ell((x, y), h_S).$$

The Empirical Risk

We estimate the true risk by sampling a **dataset** $S = \{(x_n, y_n)\}_{n=1}^{N} \overset{\text{i.i.d.}}{\sim} D$.

We use S to choose h_S. (Sometimes suppress S).

We then compute the **empirical risk** r_S:

$$r_S(h_S) = \frac{1}{N} \sum_{(x,y) \in S} \ell((x, y), h_S).$$

PAC bounds upper bound true risk $R(h_S)$ in terms of the empirical risk.

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

The dataset S is known, but in the PAC setting it is a random variable sampled from D.

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

The dataset S is known, but in the PAC setting it is a random variable sampled from D.

D is unknown. A Bayesian might place a prior over the parameters of D, and update with Bayes' rule.

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

The dataset S is known, but in the PAC setting it is a random variable sampled from D.

D is unknown. A Bayesian might place a prior over the parameters of D, and update with Bayes' rule.

This is illegal in the PAC setting. D is unknown, but its "form/parameters", whatever they are, are deterministic.

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

The dataset S is known, but in the PAC setting it is a random variable sampled from D.

D is unknown. A Bayesian might place a prior over the parameters of D, and update with Bayes' rule.

This is illegal in the PAC setting. D is unknown, but its "form/parameters", whatever they are, are deterministic.

Hence all randomness comes from sampling S:

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

The dataset S is known, but in the PAC setting it is a random variable sampled from D.

D is unknown. A Bayesian might place a prior over the parameters of D, and update with Bayes' rule.

This is illegal in the PAC setting. D is unknown, but its "form/parameters", whatever they are, are deterministic.

Hence all randomness comes from sampling S:

$S; h_S; r_S(h_S); R(h_S)$ are all random variables through S.

A Warning for Bayesians

In Bayesian thinking, *unknown* is synonymous with *random*, and *known* is synonymous with *deterministic*.

Not so in PAC (and PAC-Bayes)!

The dataset S is known, but in the PAC setting it is a random variable sampled from D.

D is unknown. A Bayesian might place a prior over the parameters of D, and update with Bayes' rule.

This is illegal in the PAC setting. D is unknown, but its "form/parameters", whatever they are, are deterministic.

Hence all randomness comes from sampling S:

$S; h_S; r_S(h_S); R(h_S)$ are all random variables through S.

D is not random. $R(h)$ is non-random if h is non-random (in particular, independent of S).

Worst Case Analysis

But if D is not known, and can't be modelled probabilistically, how can we say anything?

Worst Case Analysis

But if D is not known, and can't be modelled probabilistically, how can we say anything?

Seek theorems that hold with high probability for *any* D.

Worst Case Analysis

But if D is not known, and can't be modelled probabilistically, how can we say anything?

Seek theorems that hold with high probability for *any* D.

PAC bounds constitute a **worst-case analysis**.

Worst Case Analysis

But if D is not known, and can't be modelled probabilistically, how can we say anything?

Seek theorems that hold with high probability for *any* D.

PAC bounds constitute a **worst-case analysis**.

However, very weak assumptions! Typically just i.i.d. assumptions.

Worst Case Analysis

But if D is not known, and can't be modelled probabilistically, how can we say anything?

Seek theorems that hold with high probability for *any* D.

PAC bounds constitute a **worst-case analysis**.

However, very weak assumptions! Typically just i.i.d. assumptions.

No need to worry about priors or model mismatch!

The Simplest PAC Bound — Validation

Consider the case where h does *not* depend on the sample S.

The Simplest PAC Bound — Validation

Consider the case where h does *not* depend on the sample S.

Arises naturally when h is learned using some training data (thought of as non-random), but we want to bound its error using a fresh **validation set** S.

The Simplest PAC Bound — Validation

Consider the case where h does *not* depend on the sample S.

Arises naturally when h is learned using some training data (thought of as non-random), but we want to bound its error using a fresh **validation set** S.

Want to bound the difference between true and empirical risk.

$$R(h) - r_S(h) = \mathrm{E}_{(x,y)\sim D}[\ell((x,y),h)] - \frac{1}{N}\sum_{(x,y)\in S}\ell((x,y),h)$$

The Simplest PAC Bound — Validation

Consider the case where h does *not* depend on the sample S.

Arises naturally when h is learned using some training data (thought of as non-random), but we want to bound its error using a fresh **validation set** S.

Want to bound the difference between true and empirical risk.

$$R(h) - r_S(h) = \mathrm{E}_{(x,y)\sim D}[\ell((x,y),h)] - \frac{1}{N}\sum_{(x,y)\in S} \ell((x,y),h)$$

Since h doesn't depend on S, the RHS is an average of i.i.d. random variables.

The Simplest PAC Bound — Validation

Consider the case where h does *not* depend on the sample S.

Arises naturally when h is learned using some training data (thought of as non-random), but we want to bound its error using a fresh **validation set** S.

Want to bound the difference between true and empirical risk.

$$R(h) - r_S(h) = \mathrm{E}_{(x,y)\sim D}[\ell((x,y),h)] - \frac{1}{N}\sum_{(x,y)\in S} \ell((x,y),h)$$

Since h doesn't depend on S, the RHS is an average of i.i.d. random variables.

The LHS is deterministic, and is just the mean of the RHS.

The Simplest PAC Bound — Validation

Consider the case where h does *not* depend on the sample S.

Arises naturally when h is learned using some training data (thought of as non-random), but we want to bound its error using a fresh **validation set** S.

Want to bound the difference between true and empirical risk.

$$R(h) - r_S(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell((x,y),h)] - \frac{1}{N}\sum_{(x,y)\in S} \ell((x,y),h)$$

Since h doesn't depend on S, the RHS is an average of i.i.d. random variables.

The LHS is deterministic, and is just the mean of the RHS.

Concentration inequalities bound deviations of this average.

Hoeffding's Inequality

Theorem 1 (Hoeffding).

Let $Z_1; : : : ; Z_N$ be i.i.d. random variables bounded in $[0; 1]$. Then for all > 0,

$$\mathrm{P}\left[\frac{1}{N}\sum_{n=1}^{N} Z_n \quad \mathrm{E}[Z_n] \;>\; \right] \qquad 2\exp(\quad 2N \;^2):$$

Hoeffding's Inequality

Theorem 1 (Hoeffding).

Let $Z_1, \ldots, Z_N$ be i.i.d. random variables bounded in $[0, 1]$. Then for all $\epsilon > 0$,

$$\mathbb{P}\left[\left| \frac{1}{N} \sum_{n=1}^{N} Z_n - \mathbb{E}[Z_n] \right| > \epsilon \right] \leq 2 \exp(-2N\epsilon^2).$$

Probability of a deviation greater than ϵ decreases as ϵ and N increase.

By writing $\delta = 2\exp(-2N\epsilon^2)$, we get, with probability at least $1 - \delta$,

$$\left| \frac{1}{N} \sum_{n=1}^{N} Z_n - \mathbb{E}[Z_n] \right| \leq \sqrt{\frac{1}{2N} \log \frac{2}{\delta}}.$$

A PAC Validation Bound

If we let $Z_n = \ell((x, y), h)$, Hoeffding's inequality immediately yields, with probability at least $1 - \delta$,

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N} \log \frac{2}{\delta}}.$$

A PAC Validation Bound

If we let $Z_n = \ell((x, y), h)$, Hoeffding's inequality immediately yields, with probability at least $1 - \delta$,

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N} \log \frac{2}{\delta}}.$$

Think of $\sqrt{\frac{1}{2N} \log \frac{2}{\delta}}$ as a gap term.

A PAC Validation Bound

If we let $Z_n = \ell((x, y), h)$, Hoeffding's inequality immediately yields, with probability at least $1 - \delta$,

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N} \log \frac{2}{\delta}}.$$

- Think of $\sqrt{\frac{1}{2N} \log \frac{2}{\delta}}$ as a gap term.
- It shrinks with more data, or with higher failure probability δ.

A PAC Validation Bound

If we let $Z_n = \ell((x, y), h)$, Hoeffding's inequality immediately yields, with probability at least 1 ,

$$R(h) \quad r_S(h) + \sqrt{\frac{1}{2N} \log \frac{2}{}}.$$

- Think of $\sqrt{\frac{1}{2N} \log \frac{2}{}}$ as a gap term.
- It shrinks with more data, or with higher failure probability .
- This is our first PAC generalisation bound!

A PAC Validation Bound

If we let $Z_n = \ell((x, y), h)$, Hoeffding's inequality immediately yields, with probability at least $1 - \delta$,

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N} \log \frac{2}{\delta}}.$$

- Think of $\sqrt{\frac{1}{2N} \log \frac{2}{\delta}}$ as a gap term.
- It shrinks with more data, or with higher failure probability δ.
- This is our first PAC generalisation bound!
- "Probably" $\rightarrow$ with probability $1 - \delta$ over dataset S,
- "Approximately" $\rightarrow$ with a gap term $\sqrt{\frac{1}{2N} \log \frac{2}{\delta}}$.

A PAC Validation Bound

If we let $Z_n = \ell((x, y), h)$, Hoeffding's inequality immediately yields, with probability at least $1 - \delta$,

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N} \log \frac{2}{\delta}}.$$

- Think of $\sqrt{\frac{1}{2N} \log \frac{2}{\delta}}$ as a gap term.
- It shrinks with more data, or with higher failure probability δ.
- This is our first PAC generalisation bound!
- "Probably" $\rightarrow$ with probability $1 - \delta$ over dataset S,
- "Approximately" $\rightarrow$ with a gap term $\sqrt{\frac{1}{2N} \log \frac{2}{\delta}}$.

But what if we want to choose h dependent on S?

A Warning About Interchanging Order of Words

The previous result holds *for all h, with high probability*.

A Warning About Interchanging Order of Words

The previous result holds *for all h*, *with high probability*.

This is very different from saying *with high probability*, *for all h*!

A Warning About Interchanging Order of Words

The previous result holds *for all h*, *with high probability*.

This is very different from saying *with high probability*, *for all h*!

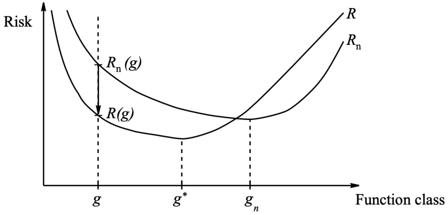

Figure 1: From Bousquet et al. [2003]

A Warning About Interchanging Order of Words

The previous result holds *for all h*, *with high probability*.

This is very different from saying *with high probability*, *for all h*!

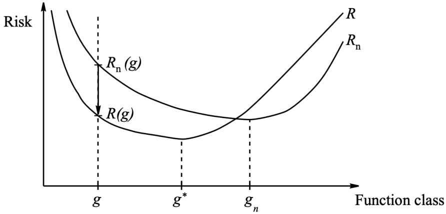

Figure 1: From Bousquet et al. [2003]

Our result says if you fix hypothesis g, then sample S, empirical risk ($R_n(g)$) will be close to true risk ($R(g)$).

A Warning About Interchanging Order of Words

The previous result holds *for all h*, *with high probability*.
This is very different from saying *with high probability*, *for all h*!

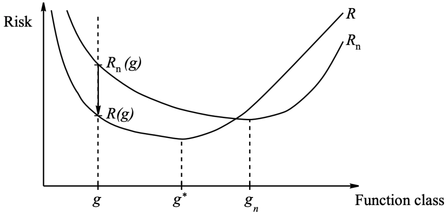

Figure 1: From Bousquet et al. [2003]

Our result says if you fix hypothesis g, then sample S, empirical risk ($R_n(g)$) will be close to true risk ($R(g)$).
Switching order implies that with high probability, the curves R_n and R are close for all g simultaneously!

A Warning About Interchanging Order of Words

The previous result holds *for all h, with high probability*.

This is very different from saying *with high probability, for all h*!

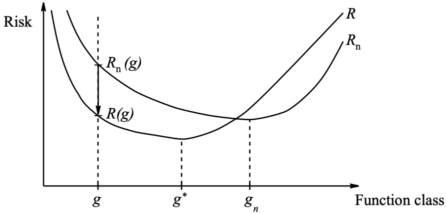

Figure 1: From Bousquet et al. [2003]

Our result says if you fix hypothesis g, then sample S, empirical risk ($R_n(g)$) will be close to true risk ($R(g)$).

Switching order implies that with high probability, the curves R_n and R are close for all g simultaneously!

Latter statement more useful: allows us to *choose the hypothesis depending on S*.

The Union Bound

We seek a statement of the form "with high probability, for all $h \ldots$"

The Union Bound

We seek a statement of the form "with high probability, for all $h \dots$"

Can use the **union bound** for a finite hypothesis space H.

The Union Bound

We seek a statement of the form "with high probability, for all $h \ldots$"

Can use the **union bound** for a finite hypothesis space H.

$P(A_1 \cup \cdots \cup A_N) \leq \sum_{n=1}^{N} P(A_n)$.

The Union Bound

We seek a statement of the form "with high probability, for all $h \ldots$"

- Can use the **union bound** for a finite hypothesis space H.
- $P(A_1 \cup \cdots \cup A_N) \le \sum_{n=1}^{N} P(A_n)$.
- We upper bound the probability of the bound failing for *any* $h \in H$:

The Union Bound

We seek a statement of the form "with high probability, for all $h \ldots$"

- Can use the **union bound** for a finite hypothesis space H.
- $\mathbb{P}(A_1 \cup \cdots \cup A_N) \leq \sum_{n=1}^{N} \mathbb{P}(A_n)$.
- We upper bound the probability of the bound failing for *any* $h \in H$:

$$
\begin{aligned}
\mathbb{P}\left(\exists h \in H : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right)
&= \mathbb{P}\left[\bigcup_{h \in H} \left\{S : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right\}\right] \\
&\leq \sum_{h \in H} \mathbb{P}\left(R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right) \\
&\leq |H|\delta.
\end{aligned}
$$

PAC Bound for Finite Hypothesis Spaces

$$P\left(\exists h \in H : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right) \le \delta |H| .$$

PAC Bound for Finite Hypothesis Spaces

$$\mathbb{P}\left(\exists h \in \mathcal{H} : R(h) > \hat{r}_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right) \le |\mathcal{H}|\delta .$$

If we set $\delta' := |\mathcal{H}|\delta$, we have that with probability at least $1 - \delta'$, *for all $h \in \mathcal{H}$ simultaneously*,

$$\begin{aligned} R(h) &\le \hat{r}_S(h) + \sqrt{\frac{1}{2N}\log\frac{2|\mathcal{H}|}{\delta'}} \\ &= \hat{r}_S(h) + \sqrt{\frac{1}{2N}\left(\log|\mathcal{H}| + \log\frac{2}{\delta'}\right)} \end{aligned}$$

PAC Bound for Finite Hypothesis Spaces

$$P\left(\exists h \in H : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right) \leq |H|\delta .$$

If we set $\delta' := |H|\delta$, we have that with probability at least $1 - \delta'$, *for all $h \in H$ simultaneously*,

$$\begin{aligned} R(h) &\leq r_S(h) + \sqrt{\frac{1}{2N}\log\frac{2|H|}{\delta'}} \\ &= r_S(h) + \sqrt{\frac{1}{2N}\left(\log|H| + \log\frac{2}{\delta'}\right)} \end{aligned}$$

Bound holds even if we pick $h \in H$ dependent on S.

PAC Bound for Finite Hypothesis Spaces

$$\mathbb{P}\left(\exists h \in \mathcal{H} : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right) \leq |\mathcal{H}|\delta .$$

If we set $\delta' := |\mathcal{H}|\delta$, we have that with probability at least $1 - \delta'$, *for all $h \in \mathcal{H}$ simultaneously*,

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N}\log\frac{2|\mathcal{H}|}{\delta'}}$$
$$= r_S(h) + \sqrt{\frac{1}{2N}\left(\log|\mathcal{H}| + \log\frac{2}{\delta'}\right)}$$

- Bound holds even if we pick $h \in \mathcal{H}$ dependent on S.
- Identical to validation bound except for $\log|\mathcal{H}|$, which is a crude measure of "complexity".

PAC Bound for Finite Hypothesis Spaces

$$P\left(\exists h \in \mathcal{H} : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right) \leq |\mathcal{H}|\delta.$$

If we set $\delta' := |\mathcal{H}|\delta$, we have that with probability at least $1-\delta'$, *for all $h \in \mathcal{H}$ simultaneously*,

$$\begin{aligned} R(h) &\leq r_S(h) + \sqrt{\frac{1}{2N}\log\frac{2|\mathcal{H}|}{\delta'}} \\ &= r_S(h) + \sqrt{\frac{1}{2N}\left(\log|\mathcal{H}| + \log\frac{2}{\delta'}\right)} \end{aligned}$$

- Bound holds even if we pick $h \in \mathcal{H}$ dependent on S.
- Identical to validation bound except for $\log|\mathcal{H}|$, which is a crude measure of “complexity”.
- Can we do something more interesting than union bound?

PAC Bound for Finite Hypothesis Spaces

$$\mathbb{P}\left[\exists h \in \mathcal{H} : R(h) > r_S(h) + \sqrt{(2N)^{-1}\log(2/\delta)}\right] \leq \delta|\mathcal{H}| .$$

If we set $\delta' := |\mathcal{H}|\delta$, we have that with probability at least $1-\delta'$, *for all $h \in \mathcal{H}$ simultaneously*,

$$\begin{aligned} R(h) &\leq r_S(h) + \sqrt{\frac{1}{2N}\log\frac{2|\mathcal{H}|}{\delta'}} \\ &= r_S(h) + \sqrt{\frac{1}{2N}\left(\log|\mathcal{H}| + \log\frac{2}{\delta'}\right)} \end{aligned}$$

- Bound holds even if we pick $h \in \mathcal{H}$ dependent on S.
- Identical to validation bound except for $\log|\mathcal{H}|$, which is a crude measure of “complexity”.
- Can we do something more interesting than union bound?
- Yes — if we use *randomised* hypotheses, we can use PAC-Bayes!

PAC-Bayes

PAC-Bayes setup

Define a *prior* P on hypotheses *that doesn't depend on the data,* S.

PAC-Bayes setup

Define a *prior* P on hypotheses *that doesn't depend on the data,* S.

Define a *posterior* $Q(S) = Q$ on hypotheses that *can depend on data,* S.

PAC-Bayes setup

Define a *prior* P on hypotheses *that doesn't depend on the data,* S.

Define a *posterior* $Q(S) = Q$ on hypotheses that *can depend on data,* S.

Define $R_Q = \mathrm{E}_{h \sim Q}[R(h)]$ and $r_{S;Q} = \mathrm{E}_{h \sim Q}[r_S(h)|S]$.

PAC-Bayes setup

Define a *prior* P on hypotheses *that doesn't depend on the data, S*.

Define a *posterior* $Q(S) = Q$ on hypotheses that *can depend on data, S*.

Define $R_Q = \mathrm{E}_{h \sim Q}[R(h)]$ and $r_{S,Q} = \mathrm{E}_{h \sim Q}[r_S(h)|S]$.

PAC-Bayes gives bounds of the form $R_Q \overset{1-\delta}{\leq} r_{S,Q} + f(Q, P, N, \delta)$, for all Q where f depends on how different Q and P are, and usually goes to 0 as $N \to \infty$.

PAC-Bayes setup

Define a *prior* P on hypotheses *that doesn't depend on the data,* S.

Define a *posterior* $Q(S) = Q$ on hypotheses that *can depend on data,* S.

Define $R_Q = \mathrm{E}_{h \sim Q}[R(h)]$ and $r_{S,Q} = \mathrm{E}_{h \sim Q}[r_S(h)|S]$.

PAC-Bayes gives bounds of the form $R_Q \overset{1-\delta}{\leq} r_{S,Q} + f(Q, P, N, \delta)$, for all Q where f depends on how different Q and P are, and usually goes to 0 as $N \to \infty$.

Warning: The assumptions in PAC-Bayes are different than in Bayes.

PAC-Bayes setup

Define a *prior* P on hypotheses *that doesn't depend on the data,* S.

Define a *posterior* $Q(S) = Q$ on hypotheses that *can depend on data,* S.

Define $R_Q = \mathrm{E}_{h \sim Q}[R(h)]$ and $r_{S,Q} = \mathrm{E}_{h \sim Q}[r_S(h)|S]$.

PAC-Bayes gives bounds of the form $R_Q \overset{1-\delta}{\leq} r_{S,Q} + f(Q, P, N, \delta)$, for all Q where f depends on how different Q and P are, and usually goes to 0 as $N \to \infty$.

Warning: The assumptions in PAC-Bayes are different than in Bayes.

- Assumes data is i.i.d. from (unknown) distribution D,

PAC-Bayes setup

- Define a *prior* P on hypotheses *that doesn't depend on the data,* S.
- Define a *posterior* $Q(S) = Q$ on hypotheses that *can depend on data,* S.
- Define $R_Q = \mathsf{E}_{h \sim Q}[R(h)]$ and $r_{S,Q} = \mathsf{E}_{h \sim Q}[r_S(h)|S]$.
- PAC-Bayes gives bounds of the form $R_Q \overset{1-\delta}{\leq} r_{S,Q} + f(Q,P,N,\delta)$, for all Q where f depends on how different Q and P are, and usually goes to 0 as $N \to \infty$.
- **Warning:** The assumptions in PAC-Bayes are different than in Bayes.
  - Assumes data is i.i.d. from (unknown) distribution D,
  - No assumption that P is related to a generating process/prior beliefs about the data.

Mcallester's PAC-Bayes bound

Theorem 2 (Mcallester's Theorem, McAllester, 1999, Maurer Variant).

For any $\delta \in \{0;1\}$, $D; H$ and P a probability measure supported on H, for $N \geq 8$,

$$R_Q \overset{1-\delta}{\leq} r_{S;Q} + \sqrt{\frac{D_{KL}\left[Q||P\right] + \log \sqrt{N} + \log \frac{2}{\delta}}{2N}} \tag{1}$$

for all Q probability measures supported on H.

Mcallester's PAC-Bayes bound

Theorem 2 (Mcallester's Theorem, McAllester, 1999, Maurer Variant).

For any $\ell \in \{0,1\}$, D, H and P a probability measure supported on H, for $N \geq 8$,

$$R_Q^{1} \leq r_{S;Q} + \sqrt{\frac{D_{KL}[Q\|P] + \log \sqrt{N} + \log \frac{2}{\delta}}{2N}} \quad (1)$$

for all Q probability measures supported on H.

This holds *for all Q* simultaneously. The RHS can be minimized with respect to Q to find the posterior!

Mcallester's PAC-Bayes bound

Theorem 2 (Mcallester's Theorem, McAllester, 1999, Maurer Variant).

For any $\ell \in \{0, 1\}$, D, H and P a probability measure supported on H, for $N \geq 8$,

$$R_Q \overset{1-\delta}{\leq} r_{S,Q} + \sqrt{\frac{D_{KL}[Q||P] + \log\sqrt{N} + \log\frac{2}{\delta}}{2N}} \tag{1}$$

for all Q probability measures supported on H.

- This holds *for all Q* simultaneously. The RHS can be minimized with respect to Q to find the posterior!
- If ℓ is the log-likelihood, minimizing this looks a lot like variational inference.

Mcallester's PAC-Bayes bound

Theorem 2 (Mcallester's Theorem, McAllester, 1999, Maurer Variant).

For any $\ell \in \{0,1\}$, D, H and P a probability measure supported on H, for $N \geq 8$,

$$R_Q \overset{1-\delta}{\leq} r_{S,Q} + \sqrt{\frac{D_{\mathrm{KL}}[Q||P] + \log\sqrt{N} + \log\frac{2}{\delta}}{2N}} \tag{1}$$

for all Q probability measures supported on H.

- This holds *for all Q* simultaneously. The RHS can be minimized with respect to Q to find the posterior!
- If ℓ is the log-likelihood, minimizing this looks a lot like variational inference.
- Note if $|H| < \infty$, P is uniform and Q is a point mass, then $D_{\mathrm{KL}}[Q||P] = \log|H|$, in which case this looks a lot like the union bound.

Change of Measure

$Q = Q(S)$ depends on S in a perhaps complicated way.

Change of Measure

$Q = Q(S)$ depends on S in a perhaps complicated way.

A key step in the proof of PAC-Bayes bounds is converting an expectation under Q to an expectation under P.

Change of Measure

$Q = Q(S)$ depends on S in a perhaps complicated way.

A key step in the proof of PAC-Bayes bounds is converting an expectation under Q to an expectation under P.

Suppose P and Q have densities (with respect to   ) p and q.

Change of Measure

- $Q = Q(S)$ depends on S in a perhaps complicated way.
- A key step in the proof of PAC-Bayes bounds is converting an expectation under Q to an expectation under P.
- Suppose P and Q have densities (with respect to) p and q.

Lemma 3 (Change of Measure).

$$\int (h)q(h)d \quad \textcolor{red}{D_{\mathrm{KL}}[Q\|P]} = \log \int e^{\ (h)}p(h)d \quad D_{\mathrm{KL}}\left[Q\|\hat{P}\right] \tag{2}$$

$$\log \int e^{\ (h)}p(h)d \ ; \tag{3}$$

where $\hat{P}$ is the measure with density $\hat{p}(h) = \frac{e^{\ (h)}p(h)}{\int e^{\ (h')}p(h')d}$.

Change of Measure

- $Q = Q(S)$ depends on S in a perhaps complicated way.
- A key step in the proof of PAC-Bayes bounds is converting an expectation under Q to an expectation under P.
- Suppose P and Q have densities (with respect to) p and q.

Lemma 3 (Change of Measure).

$$\int (h)q(h)d \qquad D_{\mathrm{KL}}[Q||P] = \log \int e^{(h)}p(h)d \qquad D_{\mathrm{KL}}\left[Q||\hat{P}\right] \tag{2}$$

$$\log \int e^{(h)}p(h)d \;; \tag{3}$$

where $\hat{P}$ is the measure with density $\hat{p}(h) = \frac{e^{(h)}p(h)}{\int e^{(h')}p(h')d}$.

- $\;!\;$ log-likelihood, $P \;!\;$ Bayesian prior, $Q \;!\;$ variational posterior:

$$\mathrm{E}_Q[\text{log likelihood}] \quad D_{\mathrm{KL}}[Q||P] \quad \text{log marginal likelihood}.$$

Change of Measure

$Q = Q(S)$ depends on S in a perhaps complicated way.

A key step in the proof of PAC-Bayes bounds is converting an expectation under Q to an expectation under P.

Suppose P and Q have densities (with respect to) p and q.

Lemma 3 (Change of Measure).

$$\int (h)q(h)d \quad D_{\mathrm{KL}}[Q||P] = \log \int e^{\ (h)} p(h)d \quad D_{\mathrm{KL}}\left[Q||\hat{P}\right] \tag{2}$$

$$\log \int e^{\ (h)} p(h)d \ ; \tag{3}$$

where $\hat{P}$ *is the measure with density* $\hat{p}(h) = \frac{e^{\ (h)} p(h)}{\int e^{\ (h')} p(h') d}$.

$!$ log-likelihood, P $!$ Bayesian prior, Q $!$ variational posterior:

$$\mathrm{E}_Q[\text{log likelihood}] \quad D_{\mathrm{KL}}[Q||P] \quad \text{log marginal likelihood}.$$

This is just the variational ‘ELBO’!

Proof of Change of Measure

We expand out the term $D_{\mathrm{KL}}\left[Q||\hat{P}\right]$:

$$\int \log \frac{q(h)}{\hat{p}(h)} q(h) d \quad = \quad \int \log \frac{q(h)}{p(h)} \frac{p(h)}{\hat{p}(h)} q(h) d \tag{4}$$

Proof of Change of Measure

We expand out the term $D_{KL}\left[Q||\hat{P}\right]$:

$$\int \log \frac{q(h)}{\hat{p}(h)} q(h) d = \int \log \frac{q(h)}{p(h)} \frac{p(h)}{\hat{p}(h)} q(h) d \tag{4}$$

$$= \underbrace{\int \log \frac{q(h)}{p(h)} q(h) d}_{D_{KL}[Q||P]} - \int \log \frac{\hat{p}(h)}{p(h)} q(h) d \; . \tag{5}$$

We expand out the term $D_{\mathrm{KL}}\left[Q||\hat{P}\right]$:

$$\int \log \frac{q(h)}{\hat{p}(h)} q(h) d \;=\; \int \log \frac{q(h)}{p(h)} \frac{p(h)}{\hat{p}(h)} q(h) d \qquad (4)$$

$$= \underbrace{\int \log \frac{q(h)}{p(h)} q(h) d}_{D_{\mathrm{KL}}[Q||P]} \quad \log \frac{\hat{p}(h)}{p(h)} q(h) d \;. \qquad (5)$$

The second term on the RHS is,

$$\int \log \frac{\hat{p}(h)}{p(h)} q(h) d \;=\; \int (h) q(h) d \quad \log \int e \;(h) p(h) d \;.$$

Some Useful Inequalities

Theorem 4 (Markov's inequality).

Let X be a non-negative random variable. Then for any $a > 0$,
$\mathrm{P}(X > a\mathrm{E}[X]) < 1{=}a$.

Some Useful Inequalities

Theorem 4 (Markov's inequality).

Let X be a non-negative random variable. Then for any $a > 0$,
$\mathrm{P}(X > a\mathrm{E}[X]) < 1 = a$.

Theorem 5 (Jensen's inequality).

Let f be a convex function, and suppose $\mathrm{E}[X]; \mathrm{E}[f(X)]$ are nite. Then

$$f(\mathrm{E}[X]) \quad \mathrm{E}[f(X)]:$$

Proof of Mcallester's Bound

Define $\Delta(R_Q, r_{S,Q}) = |R_Q - r_{S,Q}|^2$, our goal will be to upper bound Δ with high probability.

[0] We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q, r_{S,Q}) = |R_Q - r_{S,Q}|^2$, our goal will be to upper bound Δ with high probability.

As $|R_Q - r_{S,Q}|^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q, r_{S,Q}) \leq \mathrm{E}_{h \sim Q}[|R(h) - r_S(h)|^2].$$

[0]We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q, r_{S,Q}) = |R_Q - r_{S,Q}|^2$, our goal will be to upper bound Δ with high probability.
As $|R_Q - r_{S,Q}|^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q, r_{S,Q}) \leq \mathrm{E}_{h \sim Q}[|R(h) - r_S(h)|^2].$$

We would like to switch from an expectation that depends on Q (and hence S) to one that does not.

[0] We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q; r_{S;Q}) = jR_Q \quad r_{S;Q}j^2$, our goal will be to upper bound Δ with high probability.

As $jR_Q \quad r_{S;Q}j^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q; r_{S;Q}) \quad \mathrm{E}_{h \quad Q}[jR(h) \quad r_S(h)j^2]:$$

We would like to switch from an expectation that depends on Q (and hence S) to one that does not.

Recall, $\int \quad (h)q(h)d \qquad D_{\mathrm{KL}}[QjjP] \quad \log \int e^{\ (h)}p(h)d \quad .$

[0] We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q; r_{S;Q}) = jR_Q \quad r_{S;Q}j^2$, our goal will be to upper bound Δ with high probability.

As $jR_Q \quad r_{S;Q}j^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q; r_{S;Q}) \quad \mathrm{E}_{h \quad Q}[jR(h) \quad r_S(h)j^2]:$$

We would like to switch from an expectation that depends on Q (and hence S) to one that does not.

Recall, $\int \quad (h)q(h)d \quad$ $D_{\mathrm{KL}}[QjjP]$ $\quad \log \int e^{\quad (h)}p(h)d \quad$.

Applying change of measure to $\quad (h) = 2NjR(h) \quad r_S(h)j^2$,

[0]We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q, r_{S,Q}) = |R_Q - r_{S,Q}|^2$, our goal will be to upper bound Δ with high probability.
As $|R_Q - r_{S,Q}|^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q, r_{S,Q}) \leq \mathrm{E}_{h \sim Q}[|R(h) - r_S(h)|^2].$$

We would like to switch from an expectation that depends on Q (and hence S) to one that does not.
Recall, $\int \phi(h) q(h) d\theta \leq \color{red}{D_{\mathrm{KL}}[Q||P]} + \log \int e^{\phi(h)} p(h) d\theta$.
Applying change of measure to $\phi(h) = 2N|R(h) - r_S(h)|^2$,

$$\mathop{\mathrm{E}}_{h \sim Q}[|R(h) - r_S(h)|^2] \leq \frac{1}{2N}\left(D_{\mathrm{KL}}[Q||P] + \log \mathop{\mathrm{E}}_{h \sim P}[e^{2N|R(h) - r_S(h)|^2}]\right).$$

[0] We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q; r_{S;Q}) = jR_Q \quad r_{S;Q}j^2$, our goal will be to upper bound Δ with high probability.
As $jR_Q \quad r_{S;Q}j^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q; r_{S;Q}) \quad \mathrm{E}_{h \quad Q}[jR(h) \quad r_S(h)j^2]:$$

We would like to switch from an expectation that depends on Q (and hence S) to one that does not.
Recall, $\int \quad (h)q(h)d \qquad \color{red}{D_{\mathrm{KL}}[QjjP]} \quad \log \int e^{\quad (h)}p(h)d \quad .$
Applying change of measure to $\quad (h) = 2NjR(h) \quad r_S(h)j^2$,

$$\mathop{\mathrm{E}}_{h \quad Q}[jR(h) \quad r_S(h)j^2] \quad \frac{1}{2N} \quad D_{\mathrm{KL}}[QjjP] + \log \mathop{\mathrm{E}}_{h \quad P}[e^{2NjR(h) \quad r_S(h)j^2}] \quad :$$

Applying Markov's inequality to $\mathrm{E}_{h \quad P}[e^{2NjR(h) \quad r_S(h)j^2}]$,

[0] We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound

Define $\Delta(R_Q; r_{S;Q}) = jR_Q \quad r_{S;Q}j^2$, our goal will be to upper bound Δ with high probability.

As $jR_Q \quad r_{S;Q}j^2$ is convex, by Jensen's inequality,

$$\Delta(R_Q; r_{S;Q}) \quad \mathrm{E}_{h \quad Q}[jR(h) \quad r_S(h)j^2]:$$

We would like to switch from an expectation that depends on Q (and hence S) to one that does not.

Recall, $\int \quad (h)q(h)d \qquad D_{\mathrm{KL}}[QjjP] \quad \log \int e^{\quad (h)}p(h)d \quad .$

Applying change of measure to $\quad (h) = 2NjR(h) \quad r_S(h)j^2$,

$$\mathop{\mathrm{E}}_{h \quad Q}[jR(h) \quad r_S(h)j^2] \quad \frac{1}{2N} \quad D_{\mathrm{KL}}[QjjP] + \log \mathop{\mathrm{E}}_{h \quad P}[e^{2NjR(h) \quad r_S(h)j^2}] \quad :$$

Applying Markov's inequality to $\mathrm{E}_{h \quad P}[e^{2NjR(h) \quad r_S(h)j^2}]$,

$$\log \mathop{\mathrm{E}}_{h \quad P}[e^{2NjR(h) \quad r_S(h)j^2}]^{\quad 1} \quad \log \frac{1}{\quad}\mathrm{E} \quad \mathop{\mathrm{E}}_{h \quad P}[e^{2NjR(h) \quad r_S(h)j^2}] \quad :$$

[0] We follow the proof in Bégin et al. [2016].

Proof of Mcallester's Bound (continued)

It remains to upper bound $\mathbb{E}_S\left[\mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}]\right]$.

Proof of Mcallester's Bound (continued)

It remains to upper bound $\mathrm{E}_S \left[\mathrm{E}_{h \sim P}[e^{2N|R(h) - r_S(h)|^2}] \right]$.

Interchanging order of integration,

$$\mathop{\mathrm{E}}_{S} \mathop{\mathrm{E}}_{h \sim P}[e^{2N|R(h) - r_S(h)|^2}] = \mathop{\mathrm{E}}_{h \sim P} \mathop{\mathrm{E}}_{S}[e^{2N|R(h) - r_S(h)|^2}].$$

Proof of Mcallester's Bound (continued)

It remains to upper bound $\mathbb{E}_S\left[\mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}]\right]$.

Interchanging order of integration,

$$\mathbb{E}_S \mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}] = \mathbb{E}_{h\sim P}\mathbb{E}_S[e^{2N|R(h)-r_S(h)|^2}].$$

$Nr_S(h)|h$ is a binomial random variable, with N "coin tosses" each with probability $R(h)$ of heads.

Proof of Mcallester's Bound (continued)

It remains to upper bound $\mathbb{E}_S\left[\mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}]\right]$.

Interchanging order of integration,

$$\mathbb{E}_S \mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}] = \mathbb{E}_{h\sim P} \mathbb{E}_S[e^{2N|R(h)-r_S(h)|^2}].$$

$Nr_S(h)|h$ is a binomial random variable, with N "coin tosses" each with probability $R(h)$ of heads.

Hence,

$$\begin{aligned}
&\mathbb{E}_{h\sim P} \mathbb{E}[e^{2N|R(h)-r_S(h)|^2}] \\
&= \mathbb{E}_{h\sim P}\left[\sum_{k=0}^{N} \underbrace{\binom{N}{k} R(h)^k (1-R(h))^{N-k}}_{\text{Binomial PMF}} e^{2N|R(h)-\frac{k}{N}|^2}\right] \qquad (6)
\end{aligned}$$

Proof of Mcallester's Bound (continued)

It remains to upper bound $\mathbb{E}_S\left[\mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}]\right]$.

Interchanging order of integration,

$$\mathbb{E}_S \mathbb{E}_{h\sim P}[e^{2N|R(h)-r_S(h)|^2}] = \mathbb{E}_{h\sim P}\mathbb{E}_S[e^{2N|R(h)-r_S(h)|^2}].$$

$Nr_S(h)|h$ is a binomial random variable, with N "coin tosses" each with probability $R(h)$ of heads.

Hence,

$$\mathbb{E}_{h\sim P}\mathbb{E}[e^{2N|R(h)-r_S(h)|^2}]$$

$$= \mathbb{E}_{h\sim P}\left[\sum_{k=0}^{N} \underbrace{\binom{N}{k} R(h)^k(1-R(h))^{N-k}}_{\text{Binomial PMF}} e^{2N|R(h)-\frac{k}{N}|^2}\right] \tag{6}$$

$$\leq \sup_{m\in[0,1]} \sum_{k=0}^{N} \binom{N}{k} m^k(1-m)^{N-k} e^{2N|m-\frac{k}{N}|^2}. \tag{7}$$

Finishing the Proof

It can be shown [Maurer, 2004] that,

$$\sup_{m \in [0,1]} \sum_{k=0}^{N} \binom{N}{k} m^k (1-m)^{N-k} e^{N|m-\frac{k}{N}|^2} \leq 2\sqrt{N}.$$

Finishing the Proof

It can be shown [Maurer, 2004] that,

$$\sup_{m \in [0,1]} \sum_{k=0}^{N} \binom{N}{k} m^k (1-m)^{N-k} e^{N|m-\frac{k}{N}|^2} \leq 2\sqrt{N}.$$

Putting this altogether,

$$|R_Q - r_{S,Q}|^2 \, {}^{1} \leq \frac{D_{\mathrm{KL}}\left[Q||P\right] + \log \frac{2\sqrt{N}}{}}{2N}, \tag{8}$$

Finishing the Proof

It can be shown [Maurer, 2004] that,

$$\sup_{m \in [0,1]} \sum_{k=0}^{N} \binom{N}{k} m^k (1-m)^{N-k} e^{N|m-\frac{k}{N}|^2} \le 2\sqrt{N}.$$

Putting this altogether,

$$|R_Q - r_{S,Q}|^2 \overset{1-\delta}{\le} \frac{D_{\mathrm{KL}}\left[Q||P\right] + \log\frac{2\sqrt{N}}{\delta}}{2N}; \tag{8}$$

After rearranging gives,

$$R_Q \overset{1-\delta}{\le} r_{S,Q} + \sqrt{\frac{D_{\mathrm{KL}}\left[Q||P\right] + \log\frac{2\sqrt{N}}{\delta}}{2N}}. \tag{9}$$

Other PAC-Bayes bounds

There are a variety of other PAC-Bayes bounds with similar proofs.
(e.g. Catoni [2003], Seeger [2002], Bégin et al. [2016]).

Other PAC-Bayes bounds

There are a variety of other PAC-Bayes bounds with similar proofs. (e.g. Catoni [2003], Seeger [2002], Bégin et al. [2016]).

There are also various generalizations e.g. that allow the prior to depend in certain ways on the data S (e.g. Ambroladze et al. [2007]) or that allow for non-i.i.d. data (e.g. Ralaivola et al. [2009]).

Applications to Neural Networks

Does understanding deep learning require rethinking generalization?

Zhang et al. [2017] show that NNs trained with SGD find solutions h that:

Does understanding deep learning require rethinking generalization?

Zhang et al. [2017] show that NNs trained with SGD find solutions h that:

1. Are able to obtain 0 training error $r_S(h)$ while still generalizing (low $R(h)$).

Does understanding deep learning require rethinking generalization?

Zhang et al. [2017] show that NNs trained with SGD find solutions h that:

1. Are able to obtain 0 training error $r_S(h)$ while still generalizing (low $R(h)$).
2. Are able to achieve 0 training error $r_S(h)$ when the training labels are randomised. (Of course here $R(h)$ is large.)

Does understanding deep learning require rethinking generalization?

Zhang et al. [2017] show that NNs trained with SGD find solutions h that:

1. Are able to obtain 0 training error $r_S(h)$ while still generalizing (low $R(h)$).
2. Are able to achieve 0 training error $r_S(h)$ when the training labels are randomised. (Of course here $R(h)$ is large.)

NNs can overfit but in practise don't: **why?**

Applying traditional bounds to neural networks

Consider a NN with 2 hidden layers, 100 hidden units and no biases: 10200 parameters.

Applying traditional bounds to neural networks

Consider a NN with 2 hidden layers, 100 hidden units and no biases: 10200 parameters. **Naive hypothesis class:** every possible setting of the float-32 weights $jHj = 2^{32 \quad 60}$.

Applying traditional bounds to neural networks

Consider a NN with 2 hidden layers, 100 hidden units and no biases: 10200 parameters. **Naive hypothesis class:** every possible setting of the float-32 weights $|H| = 2^{32 \quad 60}$.

Finite hypothesis space bound:

$$R(h) \quad r_S(h) + \sqrt{\frac{1}{2N} \quad \log |H| + \log \frac{2}{0}}$$

With $\quad = 0.2$, we would need $N > 113{,}122$ for rhs to be < 1.
For ResNet50, $N > 255{,}078{,}163$.

Applying traditional bounds to neural networks

Consider a NN with 2 hidden layers, 100 hidden units and no biases: 10200 parameters. **Naive hypothesis class:** every possible setting of the float-32 weights $|H| = 2^{32 \quad 60}$.

Finite hypothesis space bound:

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N}\left(\log |H| + \log \frac{2}{\delta}\right)}$$

With $\delta = 0.2$, we would need $N > 113,122$ for rhs to be < 1.
For ResNet50, $N > 255,078,163$.

Analogously, for PAC-Bayes choosing $P(h) = \frac{1}{|H|}\ \forall h$ is problematic.

Applying traditional bounds to neural networks

Consider a NN with 2 hidden layers, 100 hidden units and no biases: 10200 parameters. **Naive hypothesis class:** every possible setting of the float-32 weights $|H| = 2^{32\ 60}$.

Finite hypothesis space bound:

$$R(h) \leq r_S(h) + \sqrt{\frac{1}{2N}\left(\log |H| + \log \frac{2}{\delta}\right)}$$

With $\delta = 0.2$, we would need $N > 113,122$ for rhs to be < 1.
For ResNet50, $N > 255,078,163$.

Analogously, for PAC-Bayes choosing $P(h) = \frac{1}{|H|}\ \forall h$ is problematic.

Bounds are vacuous: for empirically well-performing models on standard datasets, the generalisation error is bounded by a value greater than 1.

Can we do better?

Observations:

Can we do better?

Observations:

Neural networks are relatively insensitive to noise in the weights: we can quantise the weights with negligible loss in precision [Krishnamoorthi, 2018].

Can we do better?

Observations:

Neural networks are relatively insensitive to noise in the weights: we can quantise the weights with negligible loss in precision [Krishnamoorthi, 2018].

We can prune (set to 0) a large proportion of NN weights with negligible loss in precision [Blalock et al., 2020].

Can we do better?

Observations:

Neural networks are relatively insensitive to noise in the weights: we can quantise the weights with negligible loss in precision [Krishnamoorthi, 2018].

We can prune (set to 0) a large proportion of NN weights with negligible loss in precision [Blalock et al., 2020].

Hypothesis: the complexity of functions found by fitting NN models is much lower than the number of network parameters would suggest.

Nonvacous bounds for deep (stochastic) NN

Fix > 0 and P on H. Collect dataset s D. **Idea:** Optimise Q with

Nonvacous bounds for deep (stochastic) NN

Fix > 0 and P on H. Collect dataset s D. **Idea:** Optimise Q with

$$r_{S;Q} + \sqrt{\frac{D_{KL}[Q||P] + \log \frac{2\sqrt{N}}{}}{2N}}.$$

Nonvacous bounds for deep (stochastic) NN

Fix > 0 and P on H. Collect dataset S D. **Idea:** Optimise Q with

$$r_{S,Q} + \sqrt{\frac{D_{\mathrm{KL}}\left[Q || P\right] + \log 2\sqrt{N}}{2N}}.$$

Computational considerations:

Nonvacous bounds for deep (stochastic) NN

Fix > 0 and P on H. Collect dataset $S \quad D$. **Idea:** Optimise Q with

$$r_{S,Q} + \sqrt{\frac{D_{\mathrm{KL}}\left[Q \| P\right] + \log \frac{2\sqrt{N}}{}}{2N}}.$$

Computational considerations:

$r_{S,Q} = \mathrm{E}[\frac{1}{N} \sum_{(x,y) \in S} \{h(x) \neq y\}]$ is not differentiable! Use convex surrogate upper bound:

$$r_{S,Q} \quad \bar{r_{S,Q}} = \mathrm{E}_{h \quad Q}[\frac{1}{N \sqrt{2}} \sum_{(x,y) \in S} \log(1 + \exp(\quad h(x) y))]$$

Nonvacous bounds for deep (stochastic) NN

Fix > 0 and P on H. Collect dataset S D. **Idea:** Optimise Q with

$$r_{S;Q} + \sqrt{\frac{D_{KL}[Q||P] + \log \frac{2\sqrt{N}}{\ }}{2N}}.$$

Computational considerations:

$r_{S;Q} = \mathbb{E}[\frac{1}{N}\sum_{(x;y)\in S} \{h(x) \neq y\}]$ is not differentiable! Use convex surrogate upper bound:

$$r_{S;Q} \quad \bar{r}_{S;Q} = \mathbb{E}_{h \ Q}[\frac{1}{N\sqrt{2}} \sum_{(x;y)\in S} \log(1 + \exp(\ h(x)y))]$$

Choose Q to be a multivariate diagonal Gaussian over network weights w: $N(w; \ ; \ I)$

Nonvacous bounds for deep (stochastic) NN

Fix $\;> 0$ and P on H. Collect dataset $s \quad D$. **Idea:** Optimise Q with

$$r_{S;Q} + \sqrt{\frac{D_{KL}\left[QjjP\right] + \log \frac{2\sqrt{N}}{}}{2N}} :$$

Computational considerations:

- $r_{S;Q} = \mathrm{E}[\frac{1}{N} \sum_{(x;y)2S} fh(x) \neq yg]$ is not differentiable! Use convex surrogate upper bound:

$$r_{S;Q} \quad \bar{r_{S;Q}} = \mathrm{E}_{h \quad Q}[\frac{1}{N\sqrt{2}} \sum_{(x;y)2S} \log(1 + \exp(\quad h(x)y))]$$

- Choose Q to be a multivariate diagonal Gaussian over network weights w: $N(w; \;\; ; \;\; I)$
- Choose $P = N(w; 0; \;\; I)$. $\quad$ is chosen from a predefined set using a union bound.

Nonvacous bounds for deep (stochastic) neural networks (continued)

Algorithm:

Nonvacous bounds for deep (stochastic) neural networks (continued)

Algorithm:

1. Fit regular NN using SGD until convergence

Nonvacous bounds for deep (stochastic) neural networks (continued)

Algorithm:

1. Fit regular NN using SGD until convergence
2. Initialize at the local optima of the loss w . Initialize at $jw\ j$.

Nonvacous bounds for deep (stochastic) neural networks (continued)

Algorithm:

1. Fit regular NN using SGD until convergence
2. Initialize at the local optima of the loss w. Initialize at $|w|$.
3. Optimize bound until convergence

$$\bar{r}_{S,Q} + \sqrt{\frac{D_{\mathrm{KL}}\left[Q||P\right] + \log \frac{2\sqrt{N}}{\;}}{2N}}.$$

Nonvacous bounds for deep (stochastic) neural networks (continued)

Algorithm:

1. Fit regular NN using SGD until convergence
2. Initialize at the local optima of the loss w . Initialize at $|w|$.
3. Optimize bound until convergence

$$\bar{r}_{S,Q} + \sqrt{\frac{D_{\mathrm{KL}}[Q||P] + \log \frac{2\sqrt{N}}{\ }}{2N}}.$$

4. Estimate bound using original $r_{S,Q}$ loss and samples from Q.

Intuition:

Local optima in flat regions have a smaller description length

Nonvacous bounds for deep (stochastic) neural networks (continued)

Algorithm:

1. Fit regular NN using SGD until convergence
2. Initialize at the local optima of the loss w . Initialize at $jw\ j$.
3. Optimize bound until convergence

$$\bar{r}_{S;Q} + \sqrt{\frac{D_{\mathrm{KL}}[QjjP] + \log \frac{2\sqrt{N}}{\ }}{2N}}:$$

4. Estimate bound using original $r_{S;Q}$ loss and samples from Q.

Intuition:

- Local optima in flat regions have a smaller description length
- This approach is very similar to Bayes by Backprop: we are approximately optimising a lower bound on the marginal likelihood.

Nonvacous bounds for deep (stochastic) neural networks (some results)

Experiment	T-600	T-1200	T-300^2	T-600^2	T-1200^2	T-600^3	R-600
Train error	0.001	0.002	0.000	0.000	0.000	0.000	0.007
Test error	0.018	0.018	0.015	0.016	0.015	0.013	0.508
PAC-Bayes bound	0.161	0.179	0.170	0.186	0.223	0.201	1.352
KL divergence	5144	5977	5791	6534	8558	7861	201131
# parameters	471k	943k	326k	832k	2384k	1193k	472k
VC dimension	26m	56m	26m	66m	187m	121m	26m

Table 1: Results for experiments on binary class variant of MNIST. SGD is either trained on (T) true labels or (R) random labels. The network architecture is expressed as N^L, indicating L hidden layers with N nodes each. Errors are classification error. The reported VC dimension is the best known upper bound (in millions) for ReLU networks. The SNN error rates are tight upper bounds (see text for details). The PAC-Bayes bounds upper bound the test error with probability 0.965.

Nonvacous bounds for deep (stochastic) neural networks (some results)

Experiment	T-600	T-1200	T-300^2	T-600^2	T-1200^2	T-600^3	R-600
Train error	0.001	0.002	0.000	0.000	0.000	0.000	0.007
Test error	0.018	0.018	0.015	0.016	0.015	0.013	0.508
PAC-Bayes bound	0.161	0.179	0.170	0.186	0.223	0.201	1.352
KL divergence	5144	5977	5791	6534	8558	7861	201131
# parameters	471k	943k	326k	832k	2384k	1193k	472k
VC dimension	26m	56m	26m	66m	187m	121m	26m

Table 1: Results for experiments on binary class variant of MNIST. SGD is either trained on (T) true labels or (R) random labels. The network architecture is expressed as N^L, indicating L hidden layers with N nodes each. Errors are classification error. The reported VC dimension is the best known upper bound (in millions) for ReLU networks. The SNN error rates are tight upper bounds (see text for details). The PAC-Bayes bounds upper bound the test error with probability 0.965.

Takeaways:

Nonvacous bounds for deep (stochastic) neural networks (some results)

Experiment	T-600	T-1200	T-300^2	T-600^2	T-1200^2	T-600^3	R-600
Train error	0.001	0.002	0.000	0.000	0.000	0.000	0.007
Test error	0.018	0.018	0.015	0.016	0.015	0.013	0.508
PAC-Bayes bound	0.161	0.179	0.170	0.186	0.223	0.201	1.352
KL divergence	5144	5977	5791	6534	8558	7861	201131
# parameters	471k	943k	326k	832k	2384k	1193k	472k
VC dimension	26m	56m	26m	66m	187m	121m	26m

Table 1: Results for experiments on binary class variant of MNIST. SGD is either trained on (T) true labels or (R) random labels. The network architecture is expressed as N^L, indicating L hidden layers with N nodes each. Errors are classification error. The reported VC dimension is the best known upper bound (in millions) for ReLU networks. The SNN error rates are tight upper bounds (see text for details). The PAC-Bayes bounds upper bound the test error with probability 0.965.

Takeaways:

Bounds are less than 1 when models perform well

Nonvacous bounds for deep (stochastic) neural networks (some results)

Experiment	T-600	T-1200	T-300^2	T-600^2	T-1200^2	T-600^3	R-600
Train error	0.001	0.002	0.000	0.000	0.000	0.000	0.007
Test error	0.018	0.018	0.015	0.016	0.015	0.013	0.508
PAC-Bayes bound	0.161	0.179	0.170	0.186	0.223	0.201	1.352
KL divergence	5144	5977	5791	6534	8558	7861	201131
# parameters	471k	943k	326k	832k	2384k	1193k	472k
VC dimension	26m	56m	26m	66m	187m	121m	26m

Table 1: Results for experiments on binary class variant of MNIST. SGD is either trained on (T) true labels or (R) random labels. The network architecture is expressed as N^L, indicating L hidden layers with N nodes each. Errors are classification error. The reported VC dimension is the best known upper bound (in millions) for ReLU networks. The SNN error rates are tight upper bounds (see text for details). The PAC-Bayes bounds upper bound the test error with probability 0.965.

Takeaways:

Bounds are less than 1 when models perform well

Bounds can warn us when our model will not generalise

Scaling to Imagenet using NN compression

Observation:

Scaling to Imagenet using NN compression

Observation:

- The KL divergence $D_{\mathrm{KL}}[Q || P]$ can be seen as the expected number of bits needed to encode a message sampled from Q using a coding scheme optimal for P.

Scaling to Imagenet using NN compression

Observation:

- The KL divergence $D_{KL}[Q||P]$ can be seen as the expected number of bits needed to encode a message sampled from Q using a coding scheme optimal for P.

Zhou et al. [2019] leverage this interpretation to derive bounds for large networks after **pruning and quantization**.

Scaling to Imagenet using NN compression

Observation:

- The KL divergence $D_{\mathrm{KL}}[Q \| P]$ can be seen as the expected number of bits needed to encode a message sampled from Q using a coding scheme optimal for P.

Zhou et al. [2019] leverage this interpretation to derive bounds for large networks after **pruning and quantization**.

On Imagenet, they obtain a bound of 96.5% while the validation error is 35%. (Non-vacuous!)

Relating PAC-Bayes and Bayesian inference

Relating PAC-Bayes and Bayesian inference

The expected minus log likelihood associated with some posterior Q:

Relating PAC-Bayes and Bayesian inference

The expected minus log likelihood associated with some posterior Q:

$$CE_Q = \mathrm{E}_{(x,y)\sim D}[-\log \mathrm{E}_{h\sim Q}[p(y|x;h)]]$$

Relating PAC-Bayes and Bayesian inference

The expected minus log likelihood associated with some posterior Q:

$$CE_Q = \mathrm{E}_{(x,y) \sim D}[-\log \mathrm{E}_{h \sim Q}[p(y|x;h)]]$$

Losses considered before where not log-loss functions. Lets define:

Relating PAC-Bayes and Bayesian inference

The expected minus log likelihood associated with some posterior Q:

$$CE_Q = \mathrm{E}_{(x,y) \sim D}[-\log \mathrm{E}_{h \sim Q}[p(y|x,h)]]$$

Losses considered before where not log-loss functions. Lets define:

$$R_Q = \mathrm{E}_{h \sim Q}[\mathrm{E}_{(x,y) \sim D}[-\log p(y|x,h)]]$$
$$r_{S,Q} = \mathrm{E}_{h \sim Q}[\frac{1}{N} \sum_{(x,y) \in S} -\log p(y|x,h)]$$

Relating PAC-Bayes and Bayesian inference

The expected minus log likelihood associated with some posterior Q:

$$CE_Q = \mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]$$

Losses considered before where not log-loss functions. Lets define:

$$R_Q = \mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x;h)]]$$

$$r_{S,Q} = \mathbb{E}_{h\sim Q}[\frac{1}{N}\sum_{(x,y)\in S} -\log p(y|x;h)]$$

Using Jensen's we can see:

$$\underbrace{\mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]}_{CE_Q} \leq \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x;h)]]}_{R_Q}$$

Relating PAC-Bayes and Bayesian inference (Cont.)

A PAC-Bayes bound using our new loss functions:

Relating PAC-Bayes and Bayesian inference (Cont.)

A PAC-Bayes bound using our new loss functions:

$$CE_Q \quad R_Q \quad r_{S;Q} + \frac{D_{\mathrm{KL}}\left[Q\|P\right] \quad \log \quad + \quad {}_{P;D}(c;N)}{cN}$$

Relating PAC-Bayes and Bayesian inference (Cont.)

A PAC-Bayes bound using our new loss functions:

$$CE_Q \quad R_Q \quad r_{S;Q} + \frac{D_{\mathrm{KL}}[QjjP] \quad \log \quad + \quad {}_{P;D}(c;N)}{cN}$$

Here and ${}_{P;D}(c;N) = \log \mathrm{E}_{h \quad P;\,(x;y) \quad D}[\exp(cN(R_h \quad r_{h;S}))]$, const. wrt. Q! If $c = 1$, this reduces to the ELBO.

Relating PAC-Bayes and Bayesian inference (Cont.)

A PAC-Bayes bound using our new loss functions:

$$CE_Q \quad R_Q \quad r_{S;Q} + \frac{D_{\mathrm{KL}}[Q||P] \quad \log \quad + \quad {}_{P;D}(c;N)}{cN}$$

Here and ${}_{P;D}(c;N) = \log \mathrm{E}_{h \quad P; (x;y) \quad D}[\exp(cN(R_h \quad r_{h;S}))]$, const. wrt. Q! If $c = 1$, this reduces to the ELBO.

This bound is miminised when Q matches the Bayesian posterior.

Relating PAC-Bayes and Bayesian inference (Cont.)

A PAC-Bayes bound using our new loss functions:

$$CE_Q \quad R_Q \quad r_{S;Q} + \frac{D_{\mathrm{KL}}[Q||P] \quad \log \quad + \quad {}_{P;D}(c;N)}{cN}$$

Here and ${}_{P;D}(c;N) = \log \mathbb{E}_{h \quad P;\,(x;y) \quad D}[\exp(cN(R_h \quad r_{h;S}))]$, const. wrt. Q! If $c = 1$, this reduces to the ELBO.

This bound is miminised when Q matches the Bayesian posterior.

Minimising the above bound seems like it could be a good idea. Does the optima of R_Q also minimise CE_Q?

Model Misspecification

We will say that our model is correctly specified if the true data generating progress is contained within our hypothesis space H:

Model Misspecification

We will say that our model is correctly specified if the true data generating progress is contained within our hypothesis space H:

$$9 \ h \ 2 \ H \text{ s.t. } p(yjx;h) = D(yjx)$$

Model Misspecification

We will say that our model is correctly specified if the true data generating progress is contained within our hypothesis space H:

$$9 \; h \; 2 \; H \text{ s.t. } p(yjx;h) = D(yjx)$$

Otherwise we are learning under model misspecification.

The Bayesian posterior is suboptimal under misspecification

Recall:

$$\underbrace{\mathbb{E}_{(x;y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]}_{CE_Q} \qquad \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x;y)\sim D}[-\log p(y|x;h)]]}_{R_Q}$$

The Bayesian posterior is suboptimal under misspecification

Recall:

$$\underbrace{\mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]}_{CE_Q} \le \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x;h)]]}_{R_Q}$$

The distribution that minimises R_Q is $Q = \delta(h - h^{\mathrm{ML}})$: a point mass at the Maximum Likelihood solution.

The Bayesian posterior is suboptimal under mis-specification

Recall:

$$\underbrace{\mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]}_{CE_Q} \leq \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x;h)]]}_{R_Q}$$

The distribution that minimises R_Q is $Q = \delta(h - h^{\text{ML}})$: a point mass at the Maximum Likelihood solution.

This will only be a minimiser of CE_Q if:

$$\mathbb{E}_{(x,y)\sim D}[-\log[p(y|x;h^{\text{ML}})] \leq \mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]$$

for all Q.

The Bayesian posterior is suboptimal under mis-specification

Recall:

$$\underbrace{\mathbb{E}_{(x;y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]}_{CE_Q} \leq \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x;y)\sim D}[-\log p(y|x;h)]]}_{R_Q}$$

The distribution that minimises R_Q is $Q = \delta(h - h^{\mathrm{ML}})$: a point mass at the Maximum Likelihood solution.

This will only be a minimiser of CE_Q if:

$$\mathbb{E}_{(x;y)\sim D}[-\log[p(y|x;h^{\mathrm{ML}}) \leq \mathbb{E}_{(x;y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x;h)]]$$

for all Q. In other words: the single hypothesis h^{ML} is better than any model combination.

The Bayesian posterior is suboptimal under misspecification (Continued)

$(h \quad h^{\mathrm{ML}})$ a minimiser of CE_Q if:

$$\mathrm{E}_{(x;y) \quad D}[\quad \log[p(yjx;h^{\mathrm{ML}}) \quad \mathrm{E}_{(x;y) \quad D}[\quad \log \mathrm{E}_{h \quad Q}[p(yjx;h)]$$

for all Q.In other words: the single hypothesis h^{ML} is better than any model combination.

The Bayesian posterior is suboptimal under misspecification (Continued)

$(h \quad h^{\mathrm{ML}})$ a minimiser of CE_Q if:

$$\mathrm{E}_{(x;y) \quad D}[\quad \log[p(yjx;h^{\mathrm{ML}}) \quad \mathrm{E}_{(x;y) \quad D}[\quad \log \mathrm{E}_{h \quad Q}[p(yjx;h)]$$

for all Q.In other words: the single hypothesis h^{ML} is better than any model combination.

Masegosa [2019] shows this only happens under perfect model specification.

The Bayesian posterior is suboptimal under misspecification (Continued)

$(h \quad h^{\text{ML}})$ a minimiser of CE_Q if:

$$\mathrm{E}_{(x;y) \quad D}[\quad \log[p(y|x;h^{\text{ML}}) \quad \mathrm{E}_{(x;y) \quad D}[\quad \log \mathrm{E}_{h \quad Q}[p(y|x;h)]$$

for all Q.In other words: the single hypothesis h^{ML} is better than any model combination.

Masegosa [2019] shows this only happens under perfect model specification. Here, the distribution induced by our model matches the data generating distribution:

The Bayesian posterior is suboptimal under mis-specification (Continued)

$(h \quad h^{\mathrm{ML}})$ a minimiser of CE_Q if:

$$\mathrm{E}_{(x;y) \quad D}[\quad \log[p(y|x;h^{\mathrm{ML}}) \quad \mathrm{E}_{(x;y) \quad D}[\quad \log \mathrm{E}_{h \quad Q}[p(y|x;h)]$$

for all Q.In other words: the single hypothesis h^{ML} is better than any model combination.

Masegosa [2019] shows this only happens under perfect model specification. Here, the distribution induced by our model matches the data generating distribution:

$$D_{\mathrm{KL}}\left[D_{y|x}||p(y|x;h^{\mathrm{ML}})\right] = 0$$

The Bayesian posterior is suboptimal under misspecification (Continued)

$(h \quad h^{\mathrm{ML}})$ a minimiser of CE_Q if:

$$\mathrm{E}_{(x;y) \quad D}[\quad \log[p(y|x;h^{\mathrm{ML}}) \quad \mathrm{E}_{(x;y) \quad D}[\quad \log \mathrm{E}_{h \quad Q}[p(y|x;h)]$$

for all Q.In other words: the single hypothesis h^{ML} is better than any model combination.

Masegosa [2019] shows this only happens under perfect model specification. Here, the distribution induced by our model matches the data generating distribution:

$$D_{\mathrm{KL}} \Big[D_{y|x} || p(y|x;h^{\mathrm{ML}}) \Big] = 0$$

$$H(D_{y|x}) = CE_{h^{\mathrm{ML}}}$$

Second order PAC-Bayes bounds

Recall:

$$\underbrace{\mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x,h)]]}_{CE_Q} \qquad \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x,h)]]}_{R_Q}$$

Second order PAC-Bayes bounds

Recall:

$$\underbrace{\mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x,h)]]}_{CE_Q} \le \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x,h)]]}_{R_Q}$$

We can sharpen our previous bound using a second order Jensen bound:

Second order PAC-Bayes bounds

Recall:

$$\underbrace{\mathbb{E}_{(x,y)\sim D}[-\log \mathbb{E}_{h\sim Q}[p(y|x,h)]]}_{CE_Q} \leq \underbrace{\mathbb{E}_{h\sim Q}[\mathbb{E}_{(x,y)\sim D}[-\log p(y|x,h)]]}_{R_Q}$$

We can sharpen our previous bound using a second order Jensen bound:

$$CE_Q \leq R_Q - V_Q \leq R_Q$$

where V_Q is a variance encouraging term.

Second order PAC-Bayes bounds

Recall:

$$\underbrace{\mathbb{E}_{(x,y)\ D}[\ \log \mathbb{E}_{h\ Q}[p(y|x,h)]]}_{CE_Q} \quad \underbrace{\mathbb{E}_{h\ Q}[\mathbb{E}_{(x,y)\ D}[\ \log p(y|x,h)]]}_{R_Q}$$

We can sharpen our previous bound using a second order Jensen bound:

$$CE_Q \quad R_Q \quad V_Q \quad R_Q$$

where V_Q is a variance encouraging term.

$$V_Q = \mathbb{E}_{(x,y)\ D}[\frac{1}{(x,y)}\mathbb{E}_{h\ Q}[\ p(y|x,h) \quad \mathbb{E}_{h'\ Q}p(y|x,h')\ ^2]]$$

Second order PAC-Bayes bounds

Recall:

$$\underbrace{E_{(x,y)\quad D}[\quad \log E_{h\quad Q}[p(y|x,h)]}_{CE_Q} \quad \underbrace{E_{h\quad Q}[E_{(x,y)\quad D}[\quad \log p(y|x,h)]]}_{R_Q}$$

We can sharpen our previous bound using a second order Jensen bound:

$$CE_Q \quad R_Q \quad V_Q \quad R_Q$$

where V_Q is a variance encouraging term.

$$V_Q = E_{(x,y)\quad D}[\frac{1}{\quad(x,y)}E_{h\quad Q}[\;p(y|x,h) \quad E_{h'\quad Q}p(y|x,h')\;^2]]$$

V_Q takes positive values for posteriors different than a delta. It reduces to 0 otherwise (perfect model specification).

Second order PAC-Bayes: Illustration

We can add this new term to our PAC-Bayes bound:

Second order PAC-Bayes: Illustration

We can add this new term to our PAC-Bayes bound:

$$CE_Q \quad R_Q \quad V_Q \quad r_{S;Q} \quad V_Q + \frac{D_{\mathrm{KL}}\left[QjjP\right] \quad \log \quad + \quad {}_{P;D}(c;N)}{cN}$$

Second order PAC-Bayes: Illustration

We can add this new term to our PAC-Bayes bound:

$$CE_Q \quad R_Q \quad V_Q \quad r_{S;Q} \quad V_Q + \frac{D_{\mathrm{KL}}[Q\|P] \quad \log \quad + \quad {}_{P;D}(c;N)}{cN}$$

Second order PAC-Bayes: Misspecified noise model

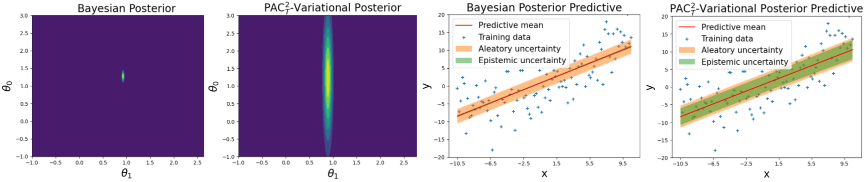

Figure 1: The exact Bayesian posterior and our new proposed (PAC^2_T-Variational) posterior, and their respective posterior predictive distributions, for a linear regression model with a misspecified constant noise term (the data noise is higher than the linear model's noise). The Bayesian posterior concentrates around the best single linear model, while our method estimates a posterior which introduces high variance in the intercept parameter θ_0 to induce a posterior predictive distribution with higher noise that better fits the data distribution (see Appendix C.2 for details).

The new variance term is able to increase disagreement among hypothesis, increasing predictive variance.

References I

A. Ambroladze, E. Parrado-hernández, and J. Shawe-taylor. Tighter pac-bayes bounds. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2007. URL `https://proceedings.neurips.cc/paper/2006/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

L. Bégin, P. Germain, F. Laviolette, and J.-F. Roy. Pac-bayesian bounds based on the rényi divergence. In *Arti cial Intelligence and Statistics*, pages 435–444. PMLR, 2016.

D. Blalock, J. J. G. Ortiz, J. Frankle, and J. Guttag. What is the state of neural network pruning?, 2020.

O. Bousquet, S. Boucheron, and G. Lugosi. Introduction to statistical learning theory. In *Summer School on Machine Learning*, pages 169–207. Springer, 2003.

O. Catoni. A pac-bayesian approach to adaptive classification. 2003.

References II

R. Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper, 2018.

A. R. Masegosa. Learning under Model Misspecification: Applications to Variational and Ensemble methods. *arXiv e-prints*, art. arXiv:1912.08335, Dec. 2019.

A. Maurer. A note on the pac bayesian theorem. *arXiv preprint cs/0411099*, 2004.

D. A. McAllester. Pac-bayesian model averaging. In *Proceedings of the twelfth annual conference on Computational learning theory*, pages 164–170, 1999.

References III

L. Ralaivola, M. Szafranski, and G. Stempfel. Chromatic pac-bayes bounds for non-iid data. In D. A. V. Dyk and M. Welling, editors, *Proceedings of the Twelfth International Conference on Arti cial Intelligence and Statistics, AISTATS 2009, Clearwater Beach, Florida, USA, April 16-18, 2009*, volume 5 of *JMLR Proceedings*, pages 416–423. JMLR.org, 2009. URL `http://proceedings.mlr.press/v5/ralaivola09a.html`.

M. Seeger. Pac-bayesian generalisation error bounds for gaussian process classification. *Journal of machine learning research*, 3(Oct):233–269, 2002.

C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization, 2017.

References IV

W. Zhou, V. Veitch, M. Austern, R. P. Adams, and P. 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.