A Primer on PAC-Bayesian Learning

Benjamin Guedj   John Shawe-Taylor

ICML 2019
June 10, 2019

What to expect

What to expect

We will...

What to expect

We will...

- Provide an overview of what PAC-Bayes is

What to expect

We will...

- Provide an overview of what PAC-Bayes is
- Illustrate its flexibility and relevance to tackle modern machine learning tasks, and rethink generalization

What to expect

We will...

- Provide an overview of what PAC-Bayes is
- Illustrate its flexibility and relevance to tackle modern machine learning tasks, and rethink generalization
- Cover main existing results and key ideas, and briefly sketch some proofs

What to expect

We will...

- Provide an overview of what PAC-Bayes is
- Illustrate its flexibility and relevance to tackle modern machine learning tasks, and rethink generalization
- Cover main existing results and key ideas, and briefly sketch some proofs

We won't...

What to expect

We will...

- Provide an overview of what PAC-Bayes is
- Illustrate its flexibility and relevance to tackle modern machine learning tasks, and rethink generalization
- Cover main existing results and key ideas, and briefly sketch some proofs

We won't...

- Cover all of Statistical Learning Theory: see the NeurIPS 2018 tutorial "Statistical Learning Theory: A Hitchhiker's guide" (Shawe-Taylor and Rivasplata)

What to expect

We will...

- Provide an overview of what PAC-Bayes is
- Illustrate its flexibility and relevance to tackle modern machine learning tasks, and rethink generalization
- Cover main existing results and key ideas, and briefly sketch some proofs

We won't...

- Cover all of Statistical Learning Theory: see the NeurIPS 2018 tutorial "Statistical Learning Theory: A Hitchhiker's guide" (Shawe-Taylor and Rivasplata)
- Provide an encyclopaedic coverage of the PAC-Bayes literature (apologies!)

In a nutshell

In a nutshell

PAC-Bayes is a generic framework to efficiently rethink generalization for numerous machine learning algorithms. It leverages the flexibility of Bayesian learning and allows to derive new learning algorithms.

The plan

1. Elements of Statistical Learning

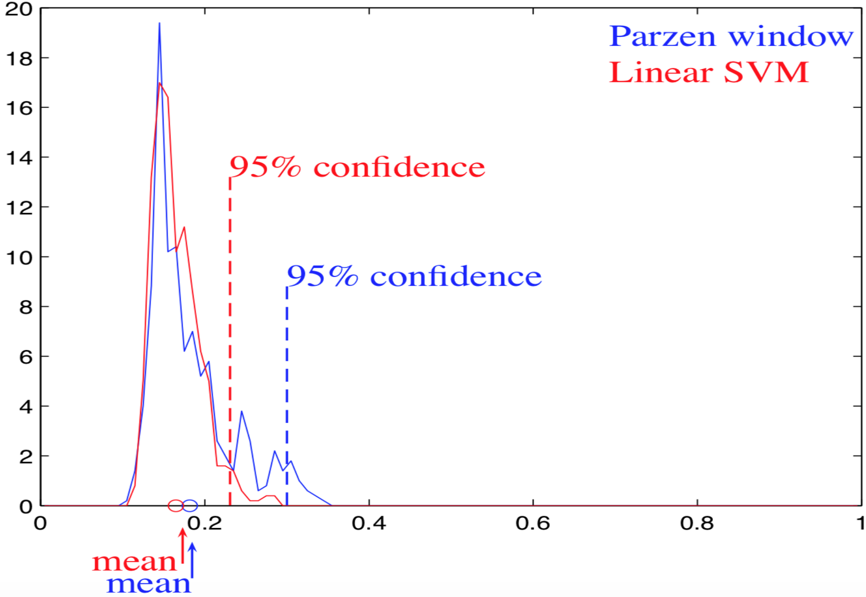
2. The PAC-Bayesian Theory

3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

The plan

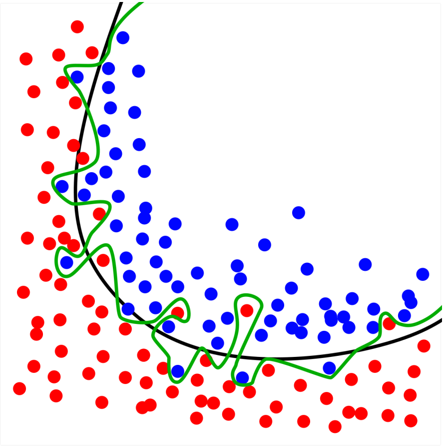
1. Elements of Statistical Learning
2. The PAC-Bayesian Theory
3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

Error distribution



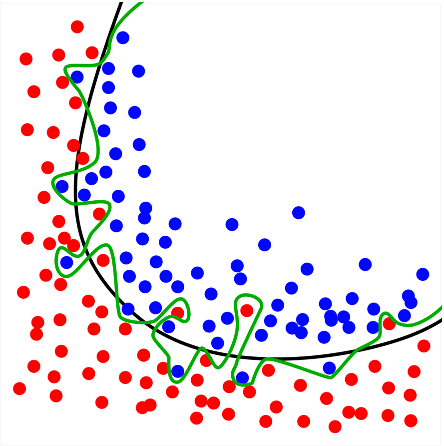
Learning is to be able to generalize

Learning is to be able to generalize



[Figure from Wikipedia]

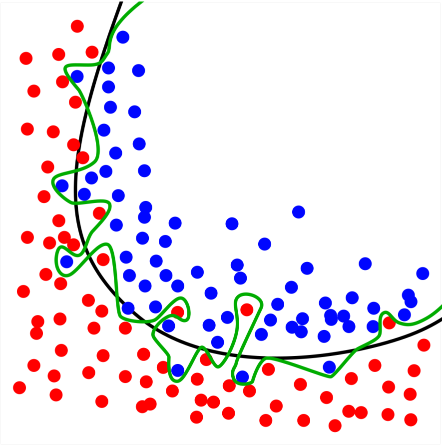
Learning is to be able to generalize



[Figure from Wikipedia]

From examples, what can a system learn about the underlying phenomenon?

Learning is to be able to generalize

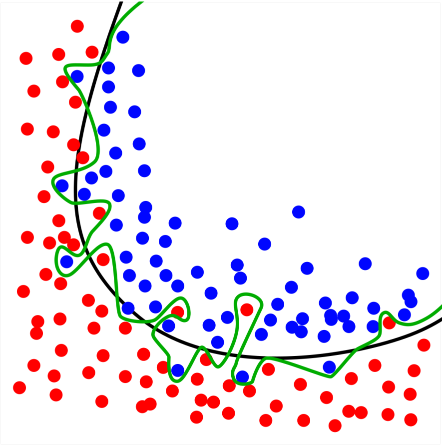


[Figure from Wikipedia]

From examples, what can a system learn about the underlying phenomenon?

Memorizing the already seen data is usually bad $\longrightarrow$ overfitting

Learning is to be able to generalize



[Figure from Wikipedia]

From examples, what can a system learn about the underlying phenomenon?

Memorizing the already seen data is usually bad $\longrightarrow$ overfitting

Generalization is the ability to 'perform' well on unseen data.

Statistical Learning Theory is about high confidence

Statistical Learning Theory is about high confidence

For a fixed algorithm, function class and sample size, generating random samples $\longrightarrow$ distribution of test errors

Statistical Learning Theory is about high confidence

For a fixed algorithm, function class and sample size, generating random samples $\longrightarrow$ distribution of test errors

- Focusing on the mean of the error distribution?
  - ▷ can be misleading: learner only has one sample

Statistical Learning Theory is about high confidence

For a fixed algorithm, function class and sample size, generating random samples $\longrightarrow$ distribution of test errors

- Focusing on the mean of the error distribution?
  - ▷ can be misleading: learner only has one sample
- Statistical Learning Theory: tail of the distribution
  - ▷ finding bounds which hold with high probability over random samples of size m

Statistical Learning Theory is about high confidence

For a fixed algorithm, function class and sample size, generating random samples $\longrightarrow$ distribution of test errors

- Focusing on the mean of the error distribution?
  - ▷ can be misleading: learner only has one sample
- Statistical Learning Theory: tail of the distribution
  - ▷ finding bounds which hold with high probability over random samples of size m
- Compare to a statistical test – at 99% confidence level
  - ▷ chances of the conclusion not being true are less than 1%

Statistical Learning Theory is about high confidence

For a fixed algorithm, function class and sample size, generating random samples $\longrightarrow$ distribution of test errors

- Focusing on the mean of the error distribution?
  - ▷ can be misleading: learner only has one sample
- Statistical Learning Theory: tail of the distribution
  - ▷ finding bounds which hold with high probability over random samples of size m
- Compare to a statistical test – at 99% confidence level
  - ▷ chances of the conclusion not being true are less than 1%
- PAC: probably approximately correct (Valiant, 1984)
  Use a 'confidence parameter' δ: $\mathbb{P}^m[\text{large error}] \leqslant \delta$
  δ is the probability of being misled by the training set

Statistical Learning Theory is about high confidence

For a fixed algorithm, function class and sample size, generating random samples $\longrightarrow$ distribution of test errors

- Focusing on the mean of the error distribution?
  - ▷ can be misleading: learner only has one sample
- Statistical Learning Theory: tail of the distribution
  - ▷ finding bounds which hold with high probability
    over random samples of size m
- Compare to a statistical test – at 99% confidence level
  - ▷ chances of the conclusion not being true are less than 1%
- PAC: probably approximately correct (Valiant, 1984)
  Use a 'confidence parameter' δ: $\mathbb{P}^m[\text{large error}] \leqslant \delta$
  δ is the probability of being misled by the training set
- Hence high confidence: $\mathbb{P}^m[\text{approximately correct}] \geqslant 1 - \delta$

Mathematical formalization

Mathematical formalization

Learning algorithm $A : \mathcal{Z}^m \to \mathcal{H}$

- $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$
  $\mathcal{X}$ = set of inputs
  $\mathcal{Y}$ = set of outputs (e.g. labels)

- $\mathcal{H}$ = hypothesis class
  = set of predictors
  (e.g. classifiers)
  functions $\mathcal{X} \to \mathcal{Y}$

Mathematical formalization

Learning algorithm $A : \mathcal{Z}^m \to \mathcal{H}$

- $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$
  $\mathcal{X}$ = set of inputs
  $\mathcal{Y}$ = set of outputs (e.g. labels)
- $\mathcal{H}$ = hypothesis class
  = set of predictors
  (e.g. classifiers)
  functions $\mathcal{X} \to \mathcal{Y}$

Training set (aka sample): $S_m = ((X_1, Y_1), \dots, (X_m, Y_m))$
a finite sequence of input-output examples.

Mathematical formalization

Learning algorithm $A : \mathcal{Z}^m \to \mathcal{H}$

- $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$

  $\mathcal{X}$ = set of inputs

  $\mathcal{Y}$ = set of outputs (e.g. labels)

- $\mathcal{H}$ = hypothesis class

  = set of predictors

  (e.g. classifiers)

  functions $\mathcal{X} \to \mathcal{Y}$

Training set (aka sample): $S_m = ((X_1, Y_1), \ldots, (X_m, Y_m))$
a finite sequence of input-output examples.

Classical assumptions:

- A data-generating distribution $\mathbb{P}$ over $\mathcal{Z}$.
- Learner doesn't know $\mathbb{P}$, only sees the training set.
- The training set examples are *i.i.d.* from $\mathbb{P}$: $S_m \sim \mathbb{P}^m$

Mathematical formalization

Learning algorithm $A : \mathcal{Z}^m \to \mathcal{H}$

- $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$
  $\mathcal{X}$ = set of inputs
  $\mathcal{Y}$ = set of outputs (e.g. labels)
- $\mathcal{H}$ = hypothesis class
  = set of predictors
  (e.g. classifiers)
  functions $\mathcal{X} \to \mathcal{Y}$

Training set (aka sample): $S_m = ((X_1, Y_1), \ldots, (X_m, Y_m))$
a finite sequence of input-output examples.

Classical assumptions:

- A data-generating distribution $\mathbb{P}$ over $\mathcal{Z}$.
- Learner doesn't know $\mathbb{P}$, only sees the training set.
- The training set examples are *i.i.d.* from $\mathbb{P}$: $S_m \sim \mathbb{P}^m$

▷ these can be relaxed (mostly beyond the scope of this tutorial)

What to achieve from the sample?

What to achieve from the sample?

Use the available sample to:

1. learn a predictor
2. certify the predictor's performance

What to achieve from the sample?

Use the available sample to:

1. learn a predictor
2. certify the predictor's performance

Learning a predictor:

- algorithm driven by some learning principle
- informed by prior knowledge resulting in inductive bias

What to achieve from the sample?

Use the available sample to:

1. learn a predictor
2. certify the predictor's performance

Learning a predictor:

- algorithm driven by some learning principle
- informed by prior knowledge resulting in inductive bias

Certifying performance:

- what happens beyond the training set
- generalization bounds

What to achieve from the sample?

Use the available sample to:

1. learn a predictor
2. certify the predictor's performance

Learning a predictor:

- algorithm driven by some learning principle
- informed by prior knowledge resulting in inductive bias

Certifying performance:

- what happens beyond the training set
- generalization bounds

Actually these two goals interact with each other!

Risk (aka error) measures

Risk (aka error) measures

A loss function $\ell(h(X), Y)$ is used to measure the discrepancy between a predicted output $h(X)$ and the true output Y.

Risk (aka error) measures

A loss function $\ell(h(X), Y)$ is used to measure the discrepancy between a predicted output $h(X)$ and the true output Y.

Empirical risk: (in-sample) $R_{\mathrm{in}}(h) = \frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)$

Risk (aka error) measures

A loss function $\ell(h(X), Y)$ is used to measure the discrepancy between a predicted output $h(X)$ and the true output Y.

Empirical risk: (in-sample) $R_{\text{in}}(h) = \frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)$

Theoretical risk: (out-of-sample) $R_{\text{out}}(h) = \mathbb{E}\left[\ell(h(X), Y)\right]$

Risk (aka error) measures

A loss function $\ell(h(X), Y)$ is used to measure the discrepancy between a predicted output $h(X)$ and the true output Y.

Empirical risk: $R_{\text{in}}(h) = \frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)$
(in-sample)

Theoretical risk: $R_{\text{out}}(h) = \mathbb{E}\left[\ell(h(X), Y)\right]$
(out-of-sample)

Examples:

- $\ell(h(X), Y) = \mathbf{1}[h(X) \neq Y]$: 0-1 loss (classification)
- $\ell(h(X), Y) = (Y - h(X))^2$: square loss (regression)
- $\ell(h(X), Y) = (1 - Yh(X))_+$: hinge loss
- $\ell(h(X), 1) = -\log(h(X))$: log loss (density estimation)

Generalization

Generalization

If predictor h does well on the in-sample (X, Y) pairs...

...will it still do well on out-of-sample pairs?

Generalization

If predictor h does well on the in-sample (X, Y) pairs...

...will it still do well on out-of-sample pairs?

Generalization gap: $\Delta(h) = R_{\text{out}}(h) - R_{\text{in}}(h)$

Generalization

If predictor h does well on the in-sample (X, Y) pairs...

...will it still do well on out-of-sample pairs?

Generalization gap: $\Delta(h) = R_{\text{out}}(h) - R_{\text{in}}(h)$

Upper bounds: w.h.p. $\Delta(h) \leqslant \epsilon(m, \delta)$

Generalization

If predictor h does well on the in-sample (X, Y) pairs...

...will it still do well on out-of-sample pairs?

Generalization gap: $\Delta(h) = R_{\mathrm{out}}(h) - R_{\mathrm{in}}(h)$

Upper bounds: w.h.p. $\Delta(h) \leqslant \epsilon(m, \delta)$

- $R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \epsilon(m, \delta)$

Generalization

If predictor h does well on the in-sample (X, Y) pairs...

...will it still do well on out-of-sample pairs?

Generalization gap: $\Delta(h) = R_{\mathrm{out}}(h) - R_{\mathrm{in}}(h)$

Upper bounds: w.h.p. $\Delta(h) \leqslant \epsilon(m, \delta)$

- $R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \epsilon(m, \delta)$

Lower bounds: w.h.p. $\Delta(h) \geqslant \tilde{\epsilon}(m, \delta)$

Generalization

If predictor h does well on the in-sample (X, Y) pairs...

...will it still do well on out-of-sample pairs?

Generalization gap: $\Delta(h) = R_{\mathrm{out}}(h) - R_{\mathrm{in}}(h)$

Upper bounds: w.h.p. $\Delta(h) \leqslant \epsilon(m, \delta)$

- $R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \epsilon(m, \delta)$

Lower bounds: w.h.p. $\Delta(h) \geqslant \tilde{\epsilon}(m, \delta)$

Flavours:

- distribution-free
- algorithm-free
- distribution-dependent
- algorithm-dependent

Why you should care about generalization bounds

Why you should care about generalization bounds

Generalization bounds are a safety check: give a theoretical guarantee on the performance of a learning algorithm on any unseen data.

Why you should care about generalization bounds

Generalization bounds are a safety check: give a theoretical guarantee on the performance of a learning algorithm on any unseen data.

Generalization bounds:

Why you should care about generalization bounds

Generalization bounds are a safety check: give a theoretical guarantee on the performance of a learning algorithm on any unseen data.

Generalization bounds:

- may be computed with the training sample only, do not depend on any test sample

Why you should care about generalization bounds

Generalization bounds are a safety check: give a theoretical guarantee on the performance of a learning algorithm on any unseen data.

Generalization bounds:

- may be computed with the training sample only, do not depend on any test sample
- provide a computable control on the error on any unseen data with prespecified confidence

Why you should care about generalization bounds

Generalization bounds are a safety check: give a theoretical guarantee on the performance of a learning algorithm on any unseen data.

Generalization bounds:

- may be computed with the training sample only, do not depend on any test sample
- provide a computable control on the error on any unseen data with prespecified confidence
- explain why specific learning algorithms actually work

Why you should care about generalization bounds

Generalization bounds are a safety check: give a theoretical guarantee on the performance of a learning algorithm on any unseen data.

Generalization bounds:

- may be computed with the training sample only, do not depend on any test sample
- provide a computable control on the error on any unseen data with prespecified confidence
- explain why specific learning algorithms actually work
- and even lead to designing new algorithm which scale to more complex settings

Building block: one single hypothesis

Building block: one single hypothesis

For one fixed (non data-dependent) h:

$$\mathbb{E}[R_{\mathrm{in}}(h)] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)\right] = R_{\mathrm{out}}(h)$$

Building block: one single hypothesis

For one fixed (non data-dependent) h:

$$\mathbb{E}[R_{\mathrm{in}}(h)] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)\right] = R_{\mathrm{out}}(h)$$

- $\mathbb{P}^m[\Delta(h) > \epsilon] = \mathbb{P}^m\left[\mathbb{E}[R_{\mathrm{in}}(h)] - R_{\mathrm{in}}(h) > \epsilon\right]$ deviation ineq.

Building block: one single hypothesis

For one fixed (non data-dependent) h:

$$\mathbb{E}[R_{\mathrm{in}}(h)] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)\right] = R_{\mathrm{out}}(h)$$

- $\mathbb{P}^m[\Delta(h) > \epsilon] = \mathbb{P}^m\left[\mathbb{E}[R_{\mathrm{in}}(h)] - R_{\mathrm{in}}(h) > \epsilon\right]$ deviation ineq.
- $\ell(h(X_i), Y_i)$ are independent r.v.'s

Building block: one single hypothesis

For one fixed (non data-dependent) h:

$$\mathbb{E}[R_{\text{in}}(h)] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m}\ell(h(X_i), Y_i)\right] = R_{\text{out}}(h)$$

- $\mathbb{P}^m[\Delta(h) > \epsilon] = \mathbb{P}^m\left[\mathbb{E}[R_{\text{in}}(h)] - R_{\text{in}}(h) > \epsilon\right]$ deviation ineq.
- $\ell(h(X_i), Y_i)$ are independent r.v.'s
- If $0 \leqslant \ell(h(X), Y) \leqslant 1$, using Hoeffding's inequality:

$$\mathbb{P}^m\left[\Delta(h) > \epsilon\right] \leqslant \exp\left\{-2m\epsilon^2\right\} = \delta$$

Building block: one single hypothesis

For one fixed (non data-dependent) h:

$$\mathbb{E}[R_{\mathrm{in}}(h)] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)\right] = R_{\mathrm{out}}(h)$$

- $\mathbb{P}^m[\Delta(h) > \epsilon] = \mathbb{P}^m\left[\mathbb{E}[R_{\mathrm{in}}(h)] - R_{\mathrm{in}}(h) > \epsilon\right]$ deviation ineq.
- $\ell(h(X_i), Y_i)$ are independent r.v.'s
- If $0 \leqslant \ell(h(X), Y) \leqslant 1$, using Hoeffding's inequality:

$$\mathbb{P}^m\left[\Delta(h) > \epsilon\right] \leqslant \exp\left\{-2m\epsilon^2\right\} = \delta$$

- Given $\delta \in (0, 1)$, equate RHS to δ, solve equation for ϵ, get

$$\mathbb{P}^m\left[\Delta(h) > \sqrt{(1/2m)\log(1/\delta)}\right] \leqslant \delta$$

Building block: one single hypothesis

For one fixed (non data-dependent) h:

$$\mathbb{E}[R_{\mathrm{in}}(h)] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} \ell(h(X_i), Y_i)\right] = R_{\mathrm{out}}(h)$$

- $\mathbb{P}^m[\Delta(h) > \epsilon] = \mathbb{P}^m\left[\mathbb{E}[R_{\mathrm{in}}(h)] - R_{\mathrm{in}}(h) > \epsilon\right]$ deviation ineq.
- $\ell(h(X_i), Y_i)$ are independent r.v.'s
- If $0 \leqslant \ell(h(X), Y) \leqslant 1$, using Hoeffding's inequality:

$$\mathbb{P}^m\left[\Delta(h) > \epsilon\right] \leqslant \exp\left\{-2m\epsilon^2\right\} = \delta$$

- Given $\delta \in (0, 1)$, equate RHS to δ, solve equation for ϵ, get

$$\mathbb{P}^m\left[\Delta(h) > \sqrt{(1/2m)\log(1/\delta)}\right] \leqslant \delta$$

- with probability $\geqslant 1 - \delta$, $\quad R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \sqrt{\frac{1}{2m}\log\left(\frac{1}{\delta}\right)}$

Finite function class

Finite function class

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$

Finite function class

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$

Function class $\mathcal{H}$ with $|\mathcal{H}| < \infty$

Finite function class

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$ Function class $\mathcal{H}$ with $|\mathcal{H}| < \infty$

Aim for a uniform bound: $\mathbb{P}^m\left[\forall f \in \mathcal{H},\ \Delta(f) \leqslant \epsilon\right] \geqslant 1 - \delta$

Finite function class

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$

Function class $\mathcal{H}$ with $|\mathcal{H}| < \infty$

Aim for a uniform bound: $\mathbb{P}^m\left[\forall f \in \mathcal{H},\ \ \Delta(f) \leqslant \epsilon\right] \geqslant 1 - \delta$

Basic tool: $\mathbb{P}^m(E_1 \text{ or } E_2 \text{ or } \cdots) \leqslant \mathbb{P}^m(E_1) + \mathbb{P}^m(E_2) + \cdots$

known as the union bound (aka countable sub-additivity)

Finite function class

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$

Function class $\mathcal{H}$ with $|\mathcal{H}| < \infty$

Aim for a uniform bound: $\mathbb{P}^m\left[\forall f \in \mathcal{H},\ \Delta(f) \leqslant \epsilon\right] \geqslant 1 - \delta$

Basic tool: $\mathbb{P}^m(E_1 \text{ or } E_2 \text{ or } \cdots) \leqslant \mathbb{P}^m(E_1) + \mathbb{P}^m(E_2) + \cdots$

known as the union bound (aka countable sub-additivity)

$$\begin{aligned} \mathbb{P}^m\Big[\exists f \in \mathcal{H},\ \Delta(f) > \epsilon\Big] &\leqslant \textstyle\sum_{f\in\mathcal{H}} \mathbb{P}^m\Big[\Delta(f) > \epsilon\Big] \\ &\leqslant |\mathcal{H}| \exp\left\{-2m\epsilon^2\right\} = \delta \end{aligned}$$

Finite function class

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$ Function class $\mathcal{H}$ with $|\mathcal{H}| < \infty$

Aim for a uniform bound: $\mathbb{P}^m\left[\forall f \in \mathcal{H},\ \Delta(f) \leqslant \epsilon\right] \geqslant 1 - \delta$

Basic tool: $\mathbb{P}^m(E_1 \text{ or } E_2 \text{ or } \cdots) \leqslant \mathbb{P}^m(E_1) + \mathbb{P}^m(E_2) + \cdots$

known as the union bound (aka countable sub-additivity)

$$\begin{aligned}\mathbb{P}^m\left[\exists f \in \mathcal{H},\ \Delta(f) > \epsilon\right] &\leqslant \textstyle\sum_{f \in \mathcal{H}} \mathbb{P}^m\left[\Delta(f) > \epsilon\right] \\ &\leqslant |\mathcal{H}| \exp\left\{-2m\epsilon^2\right\} = \delta\end{aligned}$$

w.p. $\geqslant 1 - \delta$, $\quad \forall h \in \mathcal{H},\ R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \sqrt{\frac{1}{2m} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$

Finite function class

Algorithm $A : \mathcal{Z}^m \to \mathcal{H}$

Function class $\mathcal{H}$ with $|\mathcal{H}| < \infty$

Aim for a uniform bound: $\mathbb{P}^m\Big[\forall f \in \mathcal{H},\ \Delta(f) \leqslant \epsilon\Big] \geqslant 1 - \delta$

Basic tool: $\mathbb{P}^m(E_1 \text{ or } E_2 \text{ or } \cdots) \leqslant \mathbb{P}^m(E_1) + \mathbb{P}^m(E_2) + \cdots$

known as the union bound (aka countable sub-additivity)

$$\mathbb{P}^m\Big[\exists f \in \mathcal{H},\ \Delta(f) > \epsilon\Big] \leqslant \sum_{f \in \mathcal{H}} \mathbb{P}^m\Big[\Delta(f) > \epsilon\Big]$$
$$\leqslant |\mathcal{H}| \exp\left\{-2m\epsilon^2\right\} = \delta$$

w.p. $\geqslant 1 - \delta, \quad \forall h \in \mathcal{H}, \quad R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \sqrt{\frac{1}{2m} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$

This is a worst-case approach, as it considers uniformly all hypotheses.

Towards non-uniform learnability

Towards non-uniform learnability

A route to improve this is to consider data-dependent hypotheses h_i, associated with prior distribution $P = (p_i)_i$ (structural risk minimization):

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H}, \quad R_{\mathrm{out}}(h_i) \leqslant R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m} \log\left(\frac{1}{p_i \delta}\right)}$$

Towards non-uniform learnability

A route to improve this is to consider data-dependent hypotheses h_i, associated with prior distribution $P = (p_i)_i$ (structural risk minimization):

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H}, \quad R_{\text{out}}(h_i) \leqslant R_{\text{in}}(h_i) + \sqrt{\frac{1}{2m} \log\left(\frac{1}{p_i \delta}\right)}$$

Note that we can also write

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H},$$
$$R_{\text{out}}(h_i) \leqslant R_{\text{in}}(h_i) + \sqrt{\frac{1}{2m}\left(\mathrm{KL}(\mathrm{Dirac}(h_i) \| P) + \log\left(\frac{1}{\delta}\right)\right)}$$

Towards non-uniform learnability

A route to improve this is to consider data-dependent hypotheses h_i, associated with prior distribution $P = (p_i)_i$ (structural risk minimization):

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H}, \quad R_{\mathrm{out}}(h_i) \leqslant R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m}\log\left(\frac{1}{p_i\delta}\right)}$$

Note that we can also write

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H},$$
$$R_{\mathrm{out}}(h_i) \leqslant R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m}\left(\mathrm{KL}(\mathrm{Dirac}(h_i)\|P) + \log\left(\frac{1}{\delta}\right)\right)}$$

- First attempt to introduce hypothesis-dependence
  (i.e. complexity depends on the chosen function)

Towards non-uniform learnability

A route to improve this is to consider data-dependent hypotheses h_i, associated with prior distribution $P = (p_i)_i$ (structural risk minimization):

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H}, \quad R_{\mathrm{out}}(h_i) \leqslant R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m} \log\left(\frac{1}{p_i \delta}\right)}$$

Note that we can also write

$$\text{w.p.} \geqslant 1-\delta, \quad \forall h_i \in \mathcal{H},$$
$$R_{\mathrm{out}}(h_i) \leqslant R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m}\left(\mathrm{KL}(\mathrm{Dirac}(h_i) \| P) + \log\left(\frac{1}{\delta}\right)\right)}$$

- First attempt to introduce hypothesis-dependence (i.e. complexity depends on the chosen function)
- This leads to a bound-minimizing algorithm:

$$\text{return} \quad \operatorname*{arg\,min}_{h_i \in \mathcal{H}} \left\{ R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m} \log\left(\frac{1}{p_i \delta}\right)} \right\}$$

Uncountably infinite function class?

Uncountably infinite function class?

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$

Uncountably infinite function class?

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$

Function class $\mathcal{H}$ with $|\mathcal{H}| \geqslant |\mathbb{N}|$

Uncountably infinite function class?

Algorithm $A : \mathcal{Z}^m \to \mathcal{H}$ Function class $\mathcal{H}$ with $|\mathcal{H}| \geqslant |\mathbb{N}|$

- Vapnik & Chervonenkis dimension: for $\mathcal{H}$ with $d = VC(\mathcal{H})$ finite, for any m, for any $\delta \in (0, 1)$,

  $$\text{w.p.} \geqslant 1 - \delta, \quad \forall h \in \mathcal{H}, \ \Delta(h) \leqslant \sqrt{\frac{8d}{m} \log\left(\frac{2em}{d}\right) + \frac{8}{m} \log\left(\frac{4}{\delta}\right)}$$

  The bound holds for all functions in the class (uniform over $\mathcal{H}$) and for all distributions (uniform over $\mathbb{P}$)

Uncountably infinite function class?

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$ Function class $\mathcal{H}$ with $|\mathcal{H}| \geqslant |\mathbb{N}|$

- Vapnik & Chervonenkis dimension: for $\mathcal{H}$ with $d = VC(\mathcal{H})$ finite, for any m, for any $\delta \in (0, 1)$,

  $$\text{w.p.} \geqslant 1 - \delta, \quad \forall h \in \mathcal{H}, \quad \Delta(h) \leqslant \sqrt{\frac{8d}{m} \log\left(\frac{2em}{d}\right) + \frac{8}{m} \log\left(\frac{4}{\delta}\right)}$$

  The bound holds for all functions in the class (uniform over $\mathcal{H}$) and for all distributions (uniform over $\mathbb{P}$)

- Rademacher complexity (measures how well a function can align with randomly perturbed labels – can be used to take advantage of margin assumptions)

Uncountably infinite function class?

Algorithm $A : \mathcal{Z}^m \to \mathcal{H}$ Function class $\mathcal{H}$ with $|\mathcal{H}| \geqslant |\mathbb{N}|$

- Vapnik & Chervonenkis dimension: for $\mathcal{H}$ with $d = VC(\mathcal{H})$ finite, for any m, for any $\delta \in (0, 1)$,

  $$\text{w.p.} \geqslant 1 - \delta, \quad \forall h \in \mathcal{H}, \ \Delta(h) \leqslant \sqrt{\frac{8d}{m} \log\left(\frac{2em}{d}\right) + \frac{8}{m} \log\left(\frac{4}{\delta}\right)}$$

  The bound holds for all functions in the class (uniform over $\mathcal{H}$) and for all distributions (uniform over $\mathbb{P}$)

- Rademacher complexity (measures how well a function can align with randomly perturbed labels – can be used to take advantage of margin assumptions)

  These approaches are suited to analyse the performance of individual functions, and take some account of correlations

Uncountably infinite function class?

Algorithm $\mathsf{A} : \mathcal{Z}^m \to \mathcal{H}$ Function class $\mathcal{H}$ with $|\mathcal{H}| \geqslant |\mathbb{N}|$

- Vapnik & Chervonenkis dimension: for $\mathcal{H}$ with $d = VC(\mathcal{H})$ finite, for any m, for any $\delta \in (0, 1)$,

  $$\text{w.p.} \geqslant 1 - \delta, \quad \forall h \in \mathcal{H}, \quad \Delta(h) \leqslant \sqrt{\frac{8d}{m} \log\left(\frac{2em}{d}\right) + \frac{8}{m} \log\left(\frac{4}{\delta}\right)}$$

  The bound holds for all functions in the class (uniform over $\mathcal{H}$) and for all distributions (uniform over $\mathbb{P}$)

- Rademacher complexity (measures how well a function can align with randomly perturbed labels – can be used to take advantage of margin assumptions)

  These approaches are suited to analyse the performance of individual functions, and take some account of correlations
  $\longrightarrow$ Extension: PAC-Bayes allows to consider *distributions* over hypotheses.

The plan

1. Elements of Statistical Learning

2. The PAC-Bayesian Theory

3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

The PAC-Bayes framework

The PAC-Bayes framework

- Before data, fix a distribution $P \in M_1(\mathcal{H})$ $\triangleright$ 'prior'

The PAC-Bayes framework

- Before data, fix a distribution $P \in M_1(\mathcal{H})$ $\triangleright$ 'prior'
- Based on data, learn a distribution $Q \in M_1(\mathcal{H})$ $\triangleright$ 'posterior'

The PAC-Bayes framework

- Before data, fix a distribution $P \in M_1(\mathcal{H})$ $\triangleright$ 'prior'
- Based on data, learn a distribution $Q \in M_1(\mathcal{H})$ $\triangleright$ 'posterior'
- Predictions:
  - draw $h \sim Q$ and predict with the chosen h.
  - each prediction with a fresh random draw.

The PAC-Bayes framework

- Before data, fix a distribution $P \in M_1(\mathcal{H})$ $\triangleright$ 'prior'
- Based on data, learn a distribution $Q \in M_1(\mathcal{H})$ $\triangleright$ 'posterior'
- Predictions:
  - draw $h \sim Q$ and predict with the chosen h.
  - each prediction with a fresh random draw.

The risk measures $R_{\mathrm{in}}(h)$ and $R_{\mathrm{out}}(h)$ are extended by averaging:

$$R_{\mathrm{in}}(Q) \equiv \int_{\mathcal{H}} R_{\mathrm{in}}(h)\, dQ(h) \qquad R_{\mathrm{out}}(Q) \equiv \int_{\mathcal{H}} R_{\mathrm{out}}(h)\, dQ(h)$$

$\mathrm{KL}(Q \| P) = \underset{h \sim Q}{\mathbf{E}} \ln \frac{Q(h)}{P(h)}$ is the Kullback-Leibler divergence.

The PAC-Bayes framework

- Before data, fix a distribution $P \in M_1(\mathcal{H})$ $\triangleright$ 'prior'
- Based on data, learn a distribution $Q \in M_1(\mathcal{H})$ $\triangleright$ 'posterior'
- Predictions:
  - draw $h \sim Q$ and predict with the chosen h.
  - each prediction with a fresh random draw.

The risk measures $R_{\mathrm{in}}(h)$ and $R_{\mathrm{out}}(h)$ are extended by averaging:

$$R_{\mathrm{in}}(Q) \equiv \int_{\mathcal{H}} R_{\mathrm{in}}(h)\, dQ(h) \qquad R_{\mathrm{out}}(Q) \equiv \int_{\mathcal{H}} R_{\mathrm{out}}(h)\, dQ(h)$$

$\mathrm{KL}(Q\|P) = \underset{h\sim Q}{\mathbf{E}} \ln \frac{Q(h)}{P(h)}$ is the Kullback-Leibler divergence.

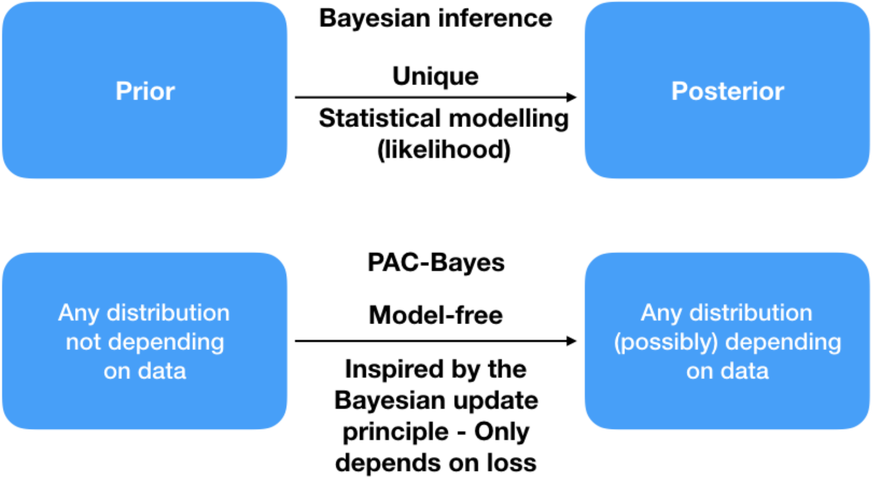
Recall the bound for data-dependent hypotheses h_i associated with prior weights p_i:

w.p. $\geqslant 1-\delta, \quad \forall h_i \in \mathcal{H},$

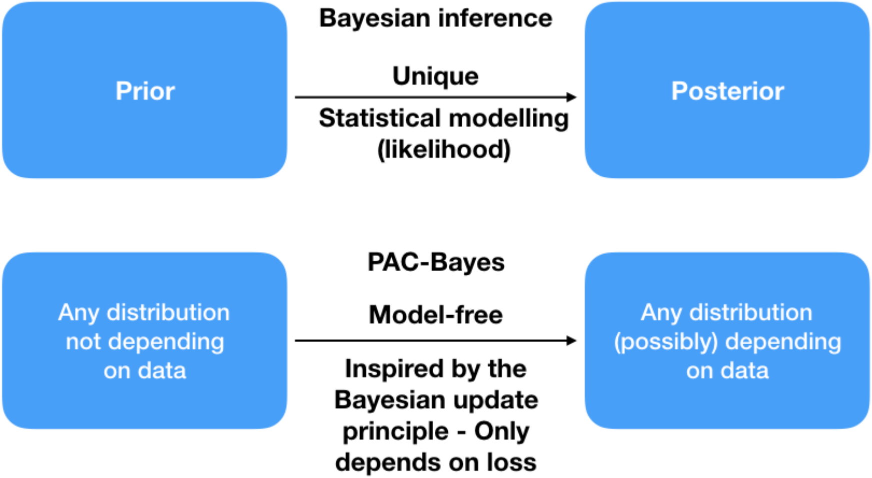
$$R_{\mathrm{out}}(h_i) \leqslant R_{\mathrm{in}}(h_i) + \sqrt{\frac{1}{2m}\left(\mathrm{KL}(\mathrm{Dirac}(h_i)\|P) + \log\left(\frac{1}{\delta}\right)\right)}$$

PAC-Bayes aka Generalized Bayes

PAC-Bayes aka Generalized Bayes

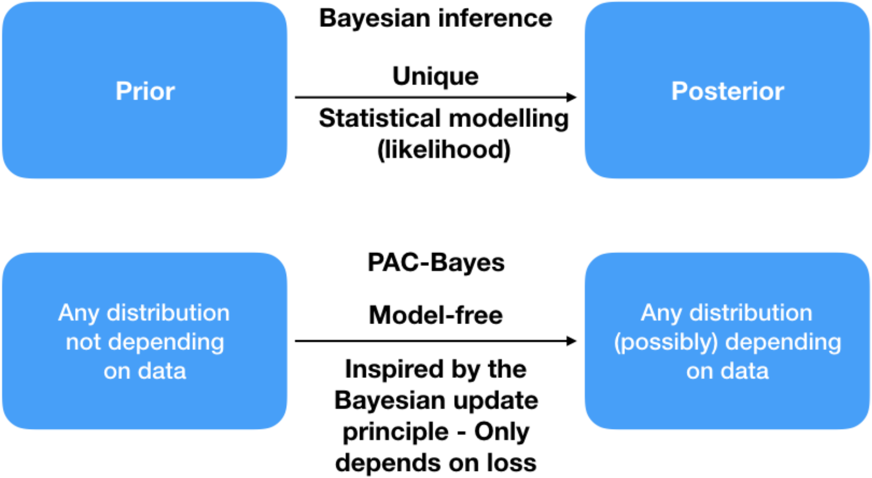


PAC-Bayes aka Generalized Bayes



"Prior": exploration mechanism of $\mathcal{H}$

PAC-Bayes aka Generalized Bayes



"Prior": exploration mechanism of $\mathcal{H}$

"Posterior" is the twisted prior after confronting with data

PAC-Bayes bounds vs. Bayesian learning

PAC-Bayes bounds vs. Bayesian learning

- Prior

PAC-Bayes bounds vs. Bayesian learning

- Prior
  - PAC-Bayes bounds: bounds hold even if prior incorrect
  - Bayesian: inference must assume prior is correct

PAC-Bayes bounds vs. Bayesian learning

- Prior
  - PAC-Bayes bounds: bounds hold even if prior incorrect
  - Bayesian: inference must assume prior is correct
- Posterior

PAC-Bayes bounds vs. Bayesian learning

- Prior
  - PAC-Bayes bounds: bounds hold even if prior incorrect
  - Bayesian: inference must assume prior is correct
- Posterior
  - PAC-Bayes bounds: bound holds for all posteriors
  - Bayesian: posterior computed by Bayesian inference, depends on statistical modeling

PAC-Bayes bounds vs. Bayesian learning

- Prior
  - PAC-Bayes bounds: bounds hold even if prior incorrect
  - Bayesian: inference must assume prior is correct
- Posterior
  - PAC-Bayes bounds: bound holds for all posteriors
  - Bayesian: posterior computed by Bayesian inference, depends on statistical modeling
- Data distribution

PAC-Bayes bounds vs. Bayesian learning

- Prior
  - PAC-Bayes bounds: bounds hold even if prior incorrect
  - Bayesian: inference must assume prior is correct
- Posterior
  - PAC-Bayes bounds: bound holds for all posteriors
  - Bayesian: posterior computed by Bayesian inference, depends on statistical modeling
- Data distribution
  - PAC-Bayes bounds: can be used to define prior, hence no need to be known explicitly
  - Bayesian: input effectively excluded from the analysis, randomness lies in the noise model generating the output

A history of PAC-Bayes

A history of PAC-Bayes

Pre-history: PAC analysis of Bayesian estimators

Shawe-Taylor and Williamson (1997); Shawe-Taylor et al. (1998)

A history of PAC-Bayes

Pre-history: PAC analysis of Bayesian estimators

Shawe-Taylor and Williamson (1997); Shawe-Taylor et al. (1998)

Birth: PAC-Bayesian bound

McAllester (1998, 1999)

McAllester Bound

For any prior P, any $\delta \in (0,1]$, we have

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\;\; R_{\mathrm{out}}(Q) \leqslant\; R_{\mathrm{in}}(Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln\frac{2\sqrt{m}}{\delta}}{2m}} \right) \;\geqslant\; 1 - \delta\,,$$

A history of PAC-Bayes

Introduction of the kl form

Langford and Seeger (2001); Seeger (2002, 2003); Langford (2005)

A history of PAC-Bayes

Introduction of the kl form

Langford and Seeger (2001); Seeger (2002, 2003); Langford (2005)

Langford and Seeger Bound

For any prior P, any $\delta \in (0, 1]$, we have

$$\mathbb{P}^m \left(\begin{array}{l} \forall\, Q \text{ on } \mathcal{H}: \\ \mathrm{kl}(R_{\mathrm{in}}(Q) \| R_{\mathrm{out}}(Q)) \leqslant \frac{1}{m} \left[\mathrm{KL}(Q \| P) + \ln \frac{2\sqrt{m}}{\delta} \right] \end{array} \right) \geqslant 1 - \delta \,,$$

where $\quad \mathrm{kl}(q \| p) \stackrel{\text{def}}{=} q \ln \frac{q}{p} + (1-q) \ln \frac{1-q}{1-p} \geqslant 2(q-p)^2.$

A General PAC-Bayesian Theorem

A General PAC-Bayesian Theorem

Δ-function: "distance" between $R_{\text{in}}(Q)$ and $R_{\text{out}}(Q)$

Convex function $\Delta : [0,1] \times [0,1] \to \mathbb{R}$.

A General PAC-Bayesian Theorem

Δ-function: "distance" between $R_{\mathrm{in}}(Q)$ and $R_{\mathrm{out}}(Q)$

Convex function $\Delta : [0,1] \times [0,1] \to \mathbb{R}$.

General theorem

Bégin et al. (2014, 2016); Germain (2015)

For any prior P on $\mathcal{H}$, for any $\delta \in (0,1]$, and for any Δ-function, we have, with probability at least $1-\delta$ over the choice of $S_m \sim \mathbb{P}^m$,

$$\forall\, Q \text{ on } \mathcal{H} : \quad \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \;\leqslant\; \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right],$$

where

$$\mathcal{I}_\Delta(m) \;=\; \sup_{r\in[0,1]} \left[\; \sum_{k=0}^{m} \underbrace{\binom{m}{k} r^k (1-r)^{m-k}}_{\mathbf{Bin}\left(k;m,r\right)} e^{m\Delta\left(\frac{k}{m},\, r\right)} \right].$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\text{in}}(Q), R_{\text{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof ideas.

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1{-}\delta\,.$$

Proof ideas.

Change of Measure Inequality (Csiszár, 1975; Donsker and Varadhan, 1975)

For any P and Q on $\mathcal{H}$, and for any measurable function $\phi : \mathcal{H} \to \mathbb{R}$, we have

$$\mathop{\mathbf{E}}_{h\sim Q} \phi(h) \ \leqslant\ \mathrm{KL}(Q\|P) + \ln\left(\mathop{\mathbf{E}}_{h\sim P} e^{\phi(h)}\right).$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\rm in}(Q), R_{\rm out}(Q)\Big) \ \leq\ \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \ \geqslant\ 1{-}\delta\,.$$

Proof ideas.

Change of Measure Inequality (Csiszár, 1975; Donsker and Varadhan, 1975)

For any P and Q on $\mathcal{H}$, and for any measurable function $\phi : \mathcal{H} \to \mathbb{R}$, we have

$$\mathop{\mathbf{E}}_{h\sim Q} \phi(h) \ \leqslant\ \mathrm{KL}(Q\|P) + \ln\left(\mathop{\mathbf{E}}_{h\sim P} e^{\phi(h)}\right).$$

Markov's inequality

$$\mathbb{P}\,(X \geqslant a) \leq \frac{\mathbf{E}\,X}{a} \qquad \Longleftrightarrow \qquad \mathbb{P}\left(X \leqslant \frac{\mathbf{E}\,X}{\delta}\right) \geq 1{-}\delta\,.$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\rm in}(Q), R_{\rm out}(Q)\Big) \le \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof ideas.

Change of Measure Inequality (Csiszár, 1975; Donsker and Varadhan, 1975)

For any P and Q on $\mathcal{H}$, and for any measurable function $\phi : \mathcal{H} \to \mathbb{R}$, we have

$$\mathop{\mathbf{E}}_{h\sim Q} \phi(h) \leqslant \mathrm{KL}(Q\|P) + \ln\left(\mathop{\mathbf{E}}_{h\sim P} e^{\phi(h)}\right).$$

Markov's inequality

$$\mathbb{P}\,(X \geqslant a) \le \frac{\mathbf{E}X}{a} \quad\Longleftrightarrow\quad \mathbb{P}\left(X \leqslant \frac{\mathbf{E}X}{\delta}\right) \ge 1-\delta\,.$$

Probability of observing k misclassifications among m examples

Given a voter h, consider a **binomial variable** of m trials with **success** $R_{\rm out}(h)$:

$$\mathbb{P}^m\left(R_{\rm in}(h) = \tfrac{k}{m}\right) = \binom{m}{k}\Big(R_{\rm out}(h)\Big)^k\Big(1 - R_{\rm out}(h)\Big)^{m-k} = \mathbf{Bin}\Big(k; m, R_{\rm out}(h)\Big)$$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\text{in}}(Q), R_{\text{out}}(Q)\Big) \le \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$m \cdot \Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\text{in}}(h), \mathop{\mathbf{E}}_{h\sim Q} R_{\text{out}}(h)\Big)$$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \le \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$m \cdot \Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\mathrm{in}}(h),\ \mathop{\mathbf{E}}_{h\sim Q} R_{\mathrm{out}}(h)\Big)$$

Jensen's Inequality

$$\leqslant \quad \mathop{\mathbf{E}}_{h\sim Q} m \cdot \Delta\Big(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h)\Big)$$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\text{in}}(Q), R_{\text{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$m \cdot \Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\text{in}}(h),\ \mathop{\mathbf{E}}_{h\sim Q} R_{\text{out}}(h)\Big)$$

Jensen's Inequality

$$\leqslant \quad \mathop{\mathbf{E}}_{h\sim Q} m \cdot \Delta\Big(R_{\text{in}}(h), R_{\text{out}}(h)\Big)$$

Change of measure

$$\leqslant \quad \mathrm{KL}(Q\|P) + \ln \mathop{\mathbf{E}}_{h\sim P} e^{m\Delta\left(R_{\text{in}}(h), R_{\text{out}}(h)\right)}$$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1{-}\delta\,.$$

Proof

$$m\cdot\Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\mathrm{in}}(h), \mathop{\mathbf{E}}_{h\sim Q} R_{\mathrm{out}}(h)\Big)$$

Jensen's Inequality $\leqslant \mathop{\mathbf{E}}_{h\sim Q} m\cdot\Delta\Big(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h)\Big)$

Change of measure $\leqslant \mathrm{KL}(Q\|P) + \ln \mathop{\mathbf{E}}_{h\sim P} e^{m\Delta\left(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h)\right)}$

Markov's Inequality $\leq_{1-\delta} \mathrm{KL}(Q\|P) + \ln\frac{1}{\delta} \mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m} \mathop{\mathbf{E}}_{h\sim P} e^{m\cdot\Delta(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h))}$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\text{in}}(Q), R_{\text{out}}(Q)\Big) \le \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$m\cdot\Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\text{in}}(h),\ \mathop{\mathbf{E}}_{h\sim Q} R_{\text{out}}(h)\Big)$$

Jensen's Inequality

$$\leqslant \quad \mathop{\mathbf{E}}_{h\sim Q} m\cdot\Delta\Big(R_{\text{in}}(h), R_{\text{out}}(h)\Big)$$

Change of measure

$$\leqslant \quad \mathrm{KL}(Q\|P) + \ln \mathop{\mathbf{E}}_{h\sim P} e^{m\Delta\big(R_{\text{in}}(h), R_{\text{out}}(h)\big)}$$

Markov's Inequality

$$\le_{1-\delta} \quad \mathrm{KL}(Q\|P) + \ln\frac{1}{\delta}\mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m}\ \mathop{\mathbf{E}}_{h\sim P} e^{m\cdot\Delta(R_{\text{in}}(h), R_{\text{out}}(h))}$$

Expectation swap

$$= \quad \mathrm{KL}(Q\|P) + \ln\frac{1}{\delta}\mathop{\mathbf{E}}_{h\sim P}\ \mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m} e^{m\cdot\Delta(R_{\text{in}}(h), R_{\text{out}}(h))}$$

$$\mathbb{P}^m\left(\forall Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\text{in}}(Q), R_{\text{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$
\begin{array}{lll}
 & & m\cdot\Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\text{in}}(h), \mathop{\mathbf{E}}_{h\sim Q} R_{\text{out}}(h)\Big) \\
\textbf{Jensen's Inequality} & \leqslant & \mathop{\mathbf{E}}_{h\sim Q} m\cdot\Delta\Big(R_{\text{in}}(h), R_{\text{out}}(h)\Big) \\
\textbf{Change of measure} & \leqslant & \mathrm{KL}(Q\|P) + \ln \mathop{\mathbf{E}}_{h\sim P} e^{m\Delta\big(R_{\text{in}}(h), R_{\text{out}}(h)\big)} \\
\textbf{Markov's Inequality} & \leq_{1-\delta} & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m} \mathop{\mathbf{E}}_{h\sim P} e^{m\cdot\Delta(R_{\text{in}}(h), R_{\text{out}}(h))} \\
\textbf{Expectation swap} & = & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \mathop{\mathbf{E}}_{h\sim P} \mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m} e^{m\cdot\Delta(R_{\text{in}}(h), R_{\text{out}}(h))} \\
\textbf{Binomial law} & = & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \mathop{\mathbf{E}}_{h\sim P} \sum_{k=0}^{m} \mathbf{Bin}\Big(k; m, R_{\text{out}}(h)\Big)\, e^{m\cdot\Delta(\frac{k}{m}, R_{\text{out}}(h))}
\end{array}
$$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\text{in}}(Q), R_{\text{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$
\begin{array}{lcl}
 & & m\cdot\Delta\Big(\underset{h\sim Q}{\mathbf{E}} R_{\text{in}}(h),\, \underset{h\sim Q}{\mathbf{E}} R_{\text{out}}(h)\Big) \\
\textbf{Jensen's Inequality} & \leqslant & \underset{h\sim Q}{\mathbf{E}}\, m\cdot\Delta\Big(R_{\text{in}}(h), R_{\text{out}}(h)\Big) \\
\textbf{Change of measure} & \leqslant & \mathrm{KL}(Q\|P) + \ln \underset{h\sim P}{\mathbf{E}}\, e^{m\Delta\left(R_{\text{in}}(h), R_{\text{out}}(h)\right)} \\
\textbf{Markov's Inequality} & \leq_{1-\delta} & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \underset{S'_m\sim\mathbb{P}^m}{\mathbf{E}}\ \underset{h\sim P}{\mathbf{E}}\, e^{m\cdot\Delta(R_{\text{in}}(h), R_{\text{out}}(h))} \\
\textbf{Expectation swap} & = & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \underset{h\sim P}{\mathbf{E}}\ \underset{S'_m\sim\mathbb{P}^m}{\mathbf{E}}\, e^{m\cdot\Delta(R_{\text{in}}(h), R_{\text{out}}(h))} \\
\textbf{Binomial law} & = & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \underset{h\sim P}{\mathbf{E}} \sum_{k=0}^{m} \mathbf{Bin}\Big(k;\, m,\, R_{\text{out}}(h)\Big)\, e^{m\cdot\Delta(\frac{k}{m}, R_{\text{out}}(h))} \\
\textbf{Supremum over risk} & \leqslant & \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \sup_{r\in[0,1]} \left[\sum_{k=0}^{m} \mathbf{Bin}\Big(k;\, m,\, r\Big)\, e^{m\Delta(\frac{k}{m}, r)}\right]
\end{array}
$$

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Proof

$$m\cdot\Delta\Big(\mathop{\mathbf{E}}_{h\sim Q} R_{\mathrm{in}}(h),\ \mathop{\mathbf{E}}_{h\sim Q} R_{\mathrm{out}}(h)\Big)$$

Jensen's Inequality

$$\leqslant \quad \mathop{\mathbf{E}}_{h\sim Q} m\cdot\Delta\Big(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h)\Big)$$

Change of measure

$$\leqslant \quad \mathrm{KL}(Q\|P) + \ln \mathop{\mathbf{E}}_{h\sim P} e^{m\Delta\left(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h)\right)}$$

Markov's Inequality

$$\leq_{1-\delta} \quad \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m} \mathop{\mathbf{E}}_{h\sim P} e^{m\cdot\Delta(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h))}$$

Expectation swap

$$= \quad \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \mathop{\mathbf{E}}_{h\sim P} \mathop{\mathbf{E}}_{S'_m\sim\mathbb{P}^m} e^{m\cdot\Delta(R_{\mathrm{in}}(h), R_{\mathrm{out}}(h))}$$

Binomial law

$$= \quad \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \mathop{\mathbf{E}}_{h\sim P} \sum_{k=0}^{m} \mathbf{Bin}\Big(k; m, R_{\mathrm{out}}(h)\Big)\, e^{m\cdot\Delta(\frac{k}{m}, R_{\mathrm{out}}(h))}$$

Supremum over risk

$$\leqslant \quad \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta} \sup_{r\in[0,1]} \left[\sum_{k=0}^{m} \mathbf{Bin}\Big(k; m, r\Big)\, e^{m\Delta(\frac{k}{m}, r)}\right]$$

$$= \quad \mathrm{KL}(Q\|P) + \ln \frac{1}{\delta}\, \mathcal{I}_\Delta(m)\,. \qquad \square$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \ \leq\ \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \ \geqslant\ 1-\delta\,.$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Corollary

[...] with probability at least $1-\delta$ *over the choice of* $S_m \sim \mathbb{P}^m$*, for all Q on* $\mathcal{H}$:

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Corollary

[...] with probability at least $1-\delta$ *over the choice of* $S_m \sim \mathbb{P}^m$*, for all* Q *on* $\mathcal{H}$ *:*

(a) $\mathrm{kl}\Big(R_{\mathrm{in}}(Q)), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{2\sqrt{m}}{\delta}\right]$ *Langford and Seeger (2001)*

$$\mathrm{kl}(q,p) \quad \stackrel{\text{def}}{=} \quad q\ln\frac{q}{p} + (1-q)\ln\frac{1-q}{1-p}$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{J}_\Delta(m)}{\delta}\right]\right) \geqslant 1{-}\delta\,.$$

Corollary

[...] with probability at least $1{-}\delta$ *over the choice of* $S_m \sim \mathbb{P}^m$*, for all* Q *on* $\mathcal{H}$ *:*

(a) $\mathrm{kl}\Big(R_{\mathrm{in}}(Q)), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{2\sqrt{m}}{\delta}\right]$ *Langford and Seeger (2001)*

(b) $R_{\mathrm{out}}(Q) \leq R_{\mathrm{in}}(Q) + \sqrt{\frac{1}{2m}\left[\mathrm{KL}(Q\|P) + \ln\frac{2\sqrt{m}}{\delta}\right]}\,,$ *McAllester (1999, 2003a)*

$$\mathrm{kl}(q,p) \overset{\text{def}}{=} q\ln\frac{q}{p} + (1-q)\ln\frac{1-q}{1-p} \geqslant 2(q-p)^2\,,$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H} :\ \Delta\Big(R_{\rm in}(Q), R_{\rm out}(Q)\Big) \le \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Corollary

[...] with probability at least $1-\delta$ *over the choice of* $S_m \sim \mathbb{P}^m$*, for all* Q *on* $\mathcal{H}$ *:*

(a) $\mathrm{kl}\Big(R_{\rm in}(Q)), R_{\rm out}(Q)\Big) \le \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{2\sqrt{m}}{\delta}\right]$ *Langford and Seeger (2001)*

(b) $R_{\rm out}(Q) \le R_{\rm in}(Q) + \sqrt{\frac{1}{2m}\left[\mathrm{KL}(Q\|P) + \ln \frac{2\sqrt{m}}{\delta}\right]}\,,$ *McAllester (1999, 2003a)*

(c) $R_{\rm out}(Q) \le \frac{1}{1-e^{-c}}\left(c \cdot R_{\rm in}(Q) + \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln \frac{1}{\delta}\right]\right),$ *Catoni (2007)*

$$\mathrm{kl}(q,p) \stackrel{\text{def}}{=} q\ln\frac{q}{p} + (1-q)\ln\frac{1-q}{1-p} \geqslant 2(q-p)^2\,,$$

$$\Delta_c(q,p) \stackrel{\text{def}}{=} -\ln[1-(1-e^{-c})\cdot p] - c\cdot q\,,$$

General theorem

$$\mathbb{P}^m\left(\forall\, Q \text{ on } \mathcal{H}:\ \Delta\Big(R_{\mathrm{in}}(Q), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{\mathcal{I}_\Delta(m)}{\delta}\right]\right) \geqslant 1-\delta\,.$$

Corollary

[...] with probability at least $1-\delta$ *over the choice of* $S_m \sim \mathbb{P}^m$*, for all* Q *on* $\mathcal{H}$ *:*

(a) $\mathrm{kl}\Big(R_{\mathrm{in}}(Q)), R_{\mathrm{out}}(Q)\Big) \leq \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{2\sqrt{m}}{\delta}\right]$ *Langford and Seeger (2001)*

(b) $R_{\mathrm{out}}(Q) \leq R_{\mathrm{in}}(Q) + \sqrt{\frac{1}{2m}\left[\mathrm{KL}(Q\|P) + \ln\frac{2\sqrt{m}}{\delta}\right]}\,,$ *McAllester (1999, 2003a)*

(c) $R_{\mathrm{out}}(Q) \leq \frac{1}{1-e^{-c}}\left(c\cdot R_{\mathrm{in}}(Q) + \frac{1}{m}\left[\mathrm{KL}(Q\|P) + \ln\frac{1}{\delta}\right]\right),$ *Catoni (2007)*

(d) $R_{\mathrm{out}}(Q) \leq R_{\mathrm{in}}(Q) + \frac{1}{\lambda}\left[\mathrm{KL}(Q\|P) + \ln\frac{1}{\delta} + f(\lambda, m)\right].$ *Alquier et al. (2016)*

$$\mathrm{kl}(q,p) \overset{\text{def}}{=} q\ln\frac{q}{p} + (1-q)\ln\frac{1-q}{1-p} \geqslant 2(q-p)^2\,,$$

$$\Delta_c(q,p) \overset{\text{def}}{=} -\ln[1-(1-e^{-c})\cdot p] - c\cdot q\,,$$

$$\Delta_\lambda(q,p) \overset{\text{def}}{=} \frac{\lambda}{m}(p-q)\,.$$

Recap

What we've seen so far

Recap

What we've seen so far

- Statistical learning theory is about high confidence control of generalization

Recap

What we've seen so far

- Statistical learning theory is about high confidence control of generalization
- PAC-Bayes is a generic, powerful tool to derive generalization bounds

Recap

What we've seen so far

- Statistical learning theory is about high confidence control of generalization
- PAC-Bayes is a generic, powerful tool to derive generalization bounds

What is coming next

Recap

What we've seen so far

- Statistical learning theory is about high confidence control of generalization
- PAC-Bayes is a generic, powerful tool to derive generalization bounds

What is coming next

- PAC-Bayes application to large classes of algorithms

Recap

What we've seen so far

- Statistical learning theory is about high confidence control of generalization
- PAC-Bayes is a generic, powerful tool to derive generalization bounds

What is coming next

- PAC-Bayes application to large classes of algorithms
- PAC-Bayesian-inspired algorithms

Recap

What we've seen so far

- Statistical learning theory is about high confidence control of generalization
- PAC-Bayes is a generic, powerful tool to derive generalization bounds

What is coming next

- PAC-Bayes application to large classes of algorithms
- PAC-Bayesian-inspired algorithms
- Case studies

A flexible framework

A flexible framework

Since 1997, PAC-Bayes has been successfully used in many machine learning settings.

A flexible framework

Since 1997, PAC-Bayes has been successfully used in many machine learning settings.

Statistical learning theory *Shawe-Taylor and Williamson (1997); McAllester (1998, 1999, 2003a,b); Seeger (2002, 2003); Maurer (2004); Catoni (2004, 2007); Audibert and Bousquet (2007); Thiemann et al. (2017)*

A flexible framework

Since 1997, PAC-Bayes has been successfully used in many machine learning settings.

Statistical learning theory *Shawe-Taylor and Williamson (1997); McAllester (1998, 1999, 2003a,b); Seeger (2002, 2003); Maurer (2004); Catoni (2004, 2007); Audibert and Bousquet (2007); Thiemann et al. (2017)*

SVMs & linear classifiers *Langford and Shawe-Taylor (2002); McAllester (2003a); Germain et al. (2009a)*

A flexible framework

Since 1997, PAC-Bayes has been successfully used in many machine learning settings.

Statistical learning theory *Shawe-Taylor and Williamson (1997); McAllester (1998, 1999, 2003a,b); Seeger (2002, 2003); Maurer (2004); Catoni (2004, 2007); Audibert and Bousquet (2007); Thiemann et al. (2017)*

SVMs & linear classifiers *Langford and Shawe-Taylor (2002); McAllester (2003a); Germain et al. (2009a)*

Supervised learning algorithms reinterpreted as bound minimizers *Ambroladze et al. (2007); Shawe-Taylor and Hardoon (2009); Germain et al. (2009b)*

A flexible framework

Since 1997, PAC-Bayes has been successfully used in many machine learning settings.

Statistical learning theory *Shawe-Taylor and Williamson (1997); McAllester (1998, 1999, 2003a,b); Seeger (2002, 2003); Maurer (2004); Catoni (2004, 2007); Audibert and Bousquet (2007); Thiemann et al. (2017)*

SVMs & linear classifiers *Langford and Shawe-Taylor (2002); McAllester (2003a); Germain et al. (2009a)*

Supervised learning algorithms reinterpreted as bound minimizers *Ambroladze et al. (2007); Shawe-Taylor and Hardoon (2009); Germain et al. (2009b)*

High-dimensional regression *Alquier and Lounici (2011); Alquier and Biau (2013); Guedj and Alquier (2013); Li et al. (2013); Guedj and Robbiano (2018)*

A flexible framework

Since 1997, PAC-Bayes has been successfully used in many machine learning settings.

Statistical learning theory *Shawe-Taylor and Williamson (1997); McAllester (1998, 1999, 2003a,b); Seeger (2002, 2003); Maurer (2004); Catoni (2004, 2007); Audibert and Bousquet (2007); Thiemann et al. (2017)*

SVMs & linear classifiers *Langford and Shawe-Taylor (2002); McAllester (2003a); Germain et al. (2009a)*

Supervised learning algorithms reinterpreted as bound minimizers *Ambroladze et al. (2007); Shawe-Taylor and Hardoon (2009); Germain et al. (2009b)*

High-dimensional regression *Alquier and Lounici (2011); Alquier and Biau (2013); Guedj and Alquier (2013); Li et al. (2013); Guedj and Robbiano (2018)*

Classification *Langford and Shawe-Taylor (2002); Catoni (2004, 2007); Lacasse et al. (2007); Parrado-Hernández et al. (2012)*

A flexible framework

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

Non-iid or heavy-tailed data *Lever et al. (2010); Seldin et al. (2011, 2012); Alquier and Guedj (2018)*

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

Non-iid or heavy-tailed data *Lever et al. (2010); Seldin et al. (2011, 2012); Alquier and Guedj (2018)*

Density estimation *Seldin and Tishby (2010); Higgs and Shawe-Taylor (2010)*

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

Non-iid or heavy-tailed data *Lever et al. (2010); Seldin et al. (2011, 2012); Alquier and Guedj (2018)*

Density estimation *Seldin and Tishby (2010); Higgs and Shawe-Taylor (2010)*

Reinforcement learning *Fard and Pineau (2010); Fard et al. (2011); Seldin et al. (2011, 2012); Ghavamzadeh et al. (2015)*

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

Non-iid or heavy-tailed data *Lever et al. (2010); Seldin et al. (2011, 2012); Alquier and Guedj (2018)*

Density estimation *Seldin and Tishby (2010); Higgs and Shawe-Taylor (2010)*

Reinforcement learning *Fard and Pineau (2010); Fard et al. (2011); Seldin et al. (2011, 2012); Ghavamzadeh et al. (2015)*

Sequential learning *Gerchinovitz (2011); Li et al. (2018)*

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

Non-iid or heavy-tailed data *Lever et al. (2010); Seldin et al. (2011, 2012); Alquier and Guedj (2018)*

Density estimation *Seldin and Tishby (2010); Higgs and Shawe-Taylor (2010)*

Reinforcement learning *Fard and Pineau (2010); Fard et al. (2011); Seldin et al. (2011, 2012); Ghavamzadeh et al. (2015)*

Sequential learning *Gerchinovitz (2011); Li et al. (2018)*

Algorithmic stability, differential privacy *London et al. (2014); London (2017); Dziugaite and Roy (2018a,b); Rivasplata et al. (2018)*

A flexible framework

Transductive learning, domain adaptation *Derbeko et al. (2004); Bégin et al. (2014); Germain et al. (2016)*

Non-iid or heavy-tailed data *Lever et al. (2010); Seldin et al. (2011, 2012); Alquier and Guedj (2018)*

Density estimation *Seldin and Tishby (2010); Higgs and Shawe-Taylor (2010)*

Reinforcement learning *Fard and Pineau (2010); Fard et al. (2011); Seldin et al. (2011, 2012); Ghavamzadeh et al. (2015)*

Sequential learning *Gerchinovitz (2011); Li et al. (2018)*

Algorithmic stability, differential privacy *London et al. (2014); London (2017); Dziugaite and Roy (2018a,b); Rivasplata et al. (2018)*

Deep neural networks *Dziugaite and Roy (2017); Neyshabur et al. (2017)*

PAC-Bayes-inspired learning algorithms

PAC-Bayes-inspired learning algorithms

In all the previous bounds, with an arbitrarily high probability and for any posterior distribution Q,

$$\text{Error on unseen data} \leqslant \text{Error on sample} + \text{complexity term}$$

$$R_{\text{out}}(Q) \leqslant R_{\text{in}}(Q) + F(Q, \cdot)$$

PAC-Bayes-inspired learning algorithms

In all the previous bounds, with an arbitrarily high probability and for any posterior distribution Q,

$$\text{Error on unseen data} \leqslant \text{Error on sample} + \text{complexity term}$$
$$R_{\text{out}}(Q) \leqslant R_{\text{in}}(Q) + F(Q, \cdot)$$

This defines a principled strategy to obtain new learning algorithms:

$$h \sim Q^\star$$
$$Q^\star \in \underset{Q \ll P}{\arg\inf} \left\{ R_{\text{in}}(Q) + F(Q, \cdot) \right\}$$

PAC-Bayes-inspired learning algorithms

In all the previous bounds, with an arbitrarily high probability and for any posterior distribution Q,

$$\text{Error on unseen data} \leqslant \text{Error on sample} + \text{complexity term}$$
$$R_{\mathrm{out}}(Q) \leqslant R_{\mathrm{in}}(Q) + F(Q, \cdot)$$

This defines a principled strategy to obtain new learning algorithms:

$$h \sim Q^\star$$
$$Q^\star \in \operatorname*{arg\,inf}_{Q \ll P} \left\{ R_{\mathrm{in}}(Q) + F(Q, \cdot) \right\}$$

(optimization problem which can be solved or approximated by [stochastic] gradient descent-flavored methods, Monte Carlo Markov Chain, Variational Bayes...)

PAC-Bayes interpretation of celebrated algorithms

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\text{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\mathrm{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

is the celebrated Gibbs posterior

$$Q_\lambda(h) \propto \exp\left(-\lambda R_{\mathrm{in}}(h)\right) P(h), \quad \forall h \in \mathcal{H}.$$

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\text{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

is the celebrated Gibbs posterior

$$Q_\lambda(h) \propto \exp\left(-\lambda R_{\text{in}}(h)\right) P(h), \quad \forall h \in \mathcal{H}.$$

Extreme cases: $\lambda \to 0$

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\text{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

is the celebrated Gibbs posterior

$$Q_\lambda(h) \propto \exp\left(-\lambda R_{\text{in}}(h)\right) P(h), \quad \forall h \in \mathcal{H}.$$

Extreme cases: $\lambda \to 0$ (flat posterior)

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\mathrm{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

is the celebrated Gibbs posterior

$$Q_\lambda(h) \propto \exp\left(-\lambda R_{\mathrm{in}}(h)\right) P(h), \quad \forall h \in \mathcal{H}.$$

Extreme cases: $\lambda \to 0$ (flat posterior) and $\lambda \to \infty$

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\mathrm{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

is the celebrated Gibbs posterior

$$Q_\lambda(h) \propto \exp\left(-\lambda R_{\mathrm{in}}(h)\right) P(h), \quad \forall h \in \mathcal{H}.$$

Extreme cases: $\lambda \to 0$ (flat posterior) and $\lambda \to \infty$ (Dirac mass on ERMs).

PAC-Bayes interpretation of celebrated algorithms

SVM with a sigmoid loss and KL-regularized Adaboost have been reinterpreted as minimizers of PAC-Bayesian bounds.

Ambroladze et al. (2007), *Shawe-Taylor and Hardoon (2009)*, *Germain et al. (2009b)*

For any $\lambda > 0$, the minimizer of

$$\left\{ R_{\mathrm{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}$$

is the celebrated Gibbs posterior

$$Q_\lambda(h) \propto \exp\left(-\lambda R_{\mathrm{in}}(h)\right) P(h), \quad \forall h \in \mathcal{H}.$$

Extreme cases: $\lambda \to 0$ (flat posterior) and $\lambda \to \infty$ (Dirac mass on ERMs). Note: continuous version of the exponentially weighted aggregate (EWA).

Variational definition of KL-divergence (Csiszár, 1975; Donsker and Varadhan, 1975; Catoni, 2004).

Variational definition of KL-divergence (Csiszár, 1975; Donsker and Varadhan, 1975; Catoni, 2004).

Let $(A, \mathcal{A})$ be a measurable space.

(i) For any probability P on $(A, \mathcal{A})$ and any measurable function $\phi : A \to \mathbb{R}$ such that $\int (\exp \circ \phi) \mathrm{d}P < \infty$,

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}.$$

Variational definition of KL-divergence (Csiszár, 1975; Donsker and Varadhan, 1975; Catoni, 2004).

Let $(A, \mathcal{A})$ be a measurable space.

(i) For any probability P on $(A, \mathcal{A})$ and any measurable function $\phi : A \to \mathbb{R}$ such that $\int (\exp \circ \phi) \mathrm{d}P < \infty$,

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}.$$

(ii) If ϕ is upper-bounded on the support of P, the supremum is reached for the Gibbs distribution G given by

$$\frac{\mathrm{d}G}{\mathrm{d}P}(a) = \frac{\exp \circ \phi(a)}{\int (\exp \circ \phi) \mathrm{d}P}, \quad a \in A.$$

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \qquad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$-\mathrm{KL}(Q, G) = -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \frac{\mathrm{d}P}{\mathrm{d}G} \right) \mathrm{d}Q$$

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$\begin{aligned} -\mathrm{KL}(Q, G) &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \frac{\mathrm{d}P}{\mathrm{d}G} \right) \mathrm{d}Q \\ &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \right) \mathrm{d}Q + \int \log \left(\frac{\mathrm{d}G}{\mathrm{d}P} \right) \mathrm{d}Q \end{aligned}$$

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$\begin{aligned} -\mathrm{KL}(Q, G) &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \frac{\mathrm{d}P}{\mathrm{d}G} \right) \mathrm{d}Q \\ &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \right) \mathrm{d}Q + \int \log \left(\frac{\mathrm{d}G}{\mathrm{d}P} \right) \mathrm{d}Q \\ &= -\mathrm{KL}(Q, P) + \int \phi \mathrm{d}\rho - \log \int (\exp \circ \phi) \, \mathrm{d}P. \end{aligned}$$

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$\begin{aligned} -\mathrm{KL}(Q, G) &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \frac{\mathrm{d}P}{\mathrm{d}G} \right) \mathrm{d}Q \\ &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \right) \mathrm{d}Q + \int \log \left(\frac{\mathrm{d}G}{\mathrm{d}P} \right) \mathrm{d}Q \\ &= -\mathrm{KL}(Q, P) + \int \phi \mathrm{d}\rho - \log \int (\exp \circ \phi) \, \mathrm{d}P. \end{aligned}$$

$\mathrm{KL}(\cdot, \cdot)$ is non-negative, $Q \mapsto -\mathrm{KL}(Q, G)$ reaches its max. in $Q = G$:

$$\log \int (\exp\circ\phi)\mathrm{d}P = \sup_{Q\ll P}\left\{\int \phi \mathrm{d}Q - \mathrm{KL}(Q,P)\right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp\circ\phi}{\int(\exp\circ\phi)\mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$\begin{aligned}
-\mathrm{KL}(Q,G) &= -\int \log\left(\frac{\mathrm{d}Q}{\mathrm{d}P}\frac{\mathrm{d}P}{\mathrm{d}G}\right)\mathrm{d}Q \\
&= -\int \log\left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right)\mathrm{d}Q + \int \log\left(\frac{\mathrm{d}G}{\mathrm{d}P}\right)\mathrm{d}Q \\
&= -\mathrm{KL}(Q,P) + \int \phi \mathrm{d}\rho - \log\int(\exp\circ\phi)\,\mathrm{d}P.
\end{aligned}$$

$\mathrm{KL}(\cdot,\cdot)$ is non-negative, $Q \mapsto -\mathrm{KL}(Q,G)$ reaches its max. in $Q = G$:

$$0 = \sup_{Q\ll P}\left\{\int \phi \mathrm{d}Q - \mathrm{KL}(Q,P)\right\} - \log\int(\exp\circ\phi)\,\mathrm{d}P.$$

$$\log \int (\exp \circ \phi) \mathrm{d}P = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\}, \quad \frac{\mathrm{d}G}{\mathrm{d}P} = \frac{\exp \circ \phi}{\int (\exp \circ \phi) \mathrm{d}P}.$$

Proof: let $Q \ll P$.

$$\begin{aligned} -\mathrm{KL}(Q, G) &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \frac{\mathrm{d}P}{\mathrm{d}G} \right) \mathrm{d}Q \\ &= -\int \log \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \right) \mathrm{d}Q + \int \log \left(\frac{\mathrm{d}G}{\mathrm{d}P} \right) \mathrm{d}Q \\ &= -\mathrm{KL}(Q, P) + \int \phi \mathrm{d}\rho - \log \int (\exp \circ \phi) \, \mathrm{d}P. \end{aligned}$$

$\mathrm{KL}(\cdot, \cdot)$ is non-negative, $Q \mapsto -\mathrm{KL}(Q, G)$ reaches its max. in $Q = G$:

$$0 = \sup_{Q \ll P} \left\{ \int \phi \mathrm{d}Q - \mathrm{KL}(Q, P) \right\} - \log \int (\exp \circ \phi) \, \mathrm{d}P.$$

Take $\phi = -\lambda R_{\mathrm{in}}$,

$$Q_\lambda \propto \exp\left(-\lambda R_{\mathrm{in}}\right) P = \arg\inf_{Q \ll P} \left\{ R_{\mathrm{in}}(Q) + \frac{\mathrm{KL}(Q, P)}{\lambda} \right\}.$$

PAC-Bayes for non-iid or heavy-tailed data

We drop the iid and bounded loss assumptions.

PAC-Bayes for non-iid or heavy-tailed data

We drop the iid and bounded loss assumptions. For any integer p,

$$\mathcal{M}_p := \int \mathbb{E}\left(|R_{\text{in}}(h) - R_{\text{out}}(h)|^p\right) \mathrm{d}P(h).$$

PAC-Bayes for non-iid or heavy-tailed data

We drop the iid and bounded loss assumptions. For any integer p,

$$\mathcal{M}_p := \int \mathbb{E}\left(|R_{\mathrm{in}}(h) - R_{\mathrm{out}}(h)|^p\right) \mathrm{d}P(h).$$

Csiszár f-divergence: let f be a convex function with $f(1) = 0$,

$$D_f(Q, P) = \int f\left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right) \mathrm{d}P$$

when $Q \ll P$ and $D_f(Q, P) = +\infty$ otherwise.

PAC-Bayes for non-iid or heavy-tailed data

We drop the iid and bounded loss assumptions. For any integer p,

$$\mathcal{M}_p := \int \mathbb{E}\left(|R_{\text{in}}(h) - R_{\text{out}}(h)|^p\right) \mathrm{d}P(h).$$

Csiszár f-divergence: let f be a convex function with $f(1) = 0$,

$$D_f(Q, P) = \int f\left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right) \mathrm{d}P$$

when $Q \ll P$ and $D_f(Q, P) = +\infty$ otherwise.

The KL is given by the special case $\mathrm{KL}(Q\|P) = D_{x\log(x)}(Q, P)$.

PAC-Bayes with f-divergences *Alquier and Guedj (2018)*

PAC-Bayes with f-divergences *Alquier and Guedj (2018)*

Let $\phi_p \colon x \mapsto x^p$. Fix $p > 1$, $q = \frac{p}{p-1}$ and $\delta \in (0, 1)$. With probability at least $1 - \delta$ we have for any distribution Q

$$|R_{\text{out}}(Q) - R_{\text{in}}(Q)| \leqslant \left(\frac{\mathcal{M}_q}{\delta}\right)^{\frac{1}{q}} \left(D_{\phi_p - 1}(Q, P) + 1\right)^{\frac{1}{p}} .$$

PAC-Bayes with f-divergences *Alquier and Guedj (2018)*

Let $\phi_p \colon x \mapsto x^p$. Fix $p > 1$, $q = \frac{p}{p-1}$ and $\delta \in (0, 1)$. With probability at least $1 - \delta$ we have for any distribution Q

$$|R_{\text{out}}(Q) - R_{\text{in}}(Q)| \leqslant \left(\frac{\mathcal{M}_q}{\delta}\right)^{\frac{1}{q}} \left(D_{\phi_p - 1}(Q, P) + 1\right)^{\frac{1}{p}}.$$

The bound decouples

- the moment $\mathcal{M}_q$ (which depends on the distribution of the data)
- and the divergence $D_{\phi_p - 1}(Q, P)$ (measure of complexity).

PAC-Bayes with f-divergences *Alquier and Guedj (2018)*

Let $\phi_p \colon x \mapsto x^p$. Fix $p > 1$, $q = \frac{p}{p-1}$ and $\delta \in (0, 1)$. With probability at least $1 - \delta$ we have for any distribution Q

$$|R_{\mathrm{out}}(Q) - R_{\mathrm{in}}(Q)| \leqslant \left(\frac{\mathcal{M}_q}{\delta}\right)^{\frac{1}{q}} \left(D_{\phi_p - 1}(Q, P) + 1\right)^{\frac{1}{p}}.$$

The bound decouples

- the moment $\mathcal{M}_q$ (which depends on the distribution of the data)
- and the divergence $D_{\phi_p - 1}(Q, P)$ (measure of complexity).

Corolloray: with probability at least $1 - \delta$, for any Q,

$$R_{\mathrm{out}}(Q) \leqslant R_{\mathrm{in}}(Q) + \left(\frac{\mathcal{M}_q}{\delta}\right)^{\frac{1}{q}} \left(D_{\phi_p - 1}(Q, P) + 1\right)^{\frac{1}{p}}.$$

Again, strong incitement to define the posterior as the minimizer of the right-hand side!

Proof

Proof

Let $\Delta(h) := |R_{\mathrm{in}}(h) - R_{\mathrm{out}}(h)|$.

Proof

Let $\Delta(h) := |R_{\text{in}}(h) - R_{\text{out}}(h)|$.

$$\left| \int R_{\text{out}} \mathrm{d}Q - \int R_{\text{in}} \mathrm{d}Q \right|$$

Proof

Let $\Delta(h) := |R_{\text{in}}(h) - R_{\text{out}}(h)|$.

$$\left| \int R_{\text{out}} \mathrm{d}Q - \int R_{\text{in}} \mathrm{d}Q \right|$$

Jensen

$$\leqslant \int \Delta \mathrm{d}Q$$

Proof

Let $\Delta(h) := |R_{\mathrm{in}}(h) - R_{\mathrm{out}}(h)|$.

$$\begin{aligned}
& && \left|\int R_{\mathrm{out}} \mathrm{d}Q - \int R_{\mathrm{in}} \mathrm{d}Q\right| \\
& \text{Jensen} && \leqslant \int \Delta \mathrm{d}Q \\
& \text{Change of measure} && = \int \Delta \frac{\mathrm{d}Q}{\mathrm{d}P} \mathrm{d}P
\end{aligned}$$

Proof

Let $\Delta(h) := |R_{\mathrm{in}}(h) - R_{\mathrm{out}}(h)|$.

$$\left|\int R_{\mathrm{out}} \mathrm{d}Q - \int R_{\mathrm{in}} \mathrm{d}Q\right|$$

Jensen $$\leqslant \int \Delta \mathrm{d}Q$$

Change of measure $$= \int \Delta \frac{\mathrm{d}Q}{\mathrm{d}P} \mathrm{d}P$$

Holder $$\leqslant \left(\int \Delta^q \mathrm{d}P\right)^{\frac{1}{q}} \left(\int \left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right)^p \mathrm{d}P\right)^{\frac{1}{p}}$$

Proof

Let $\Delta(h) := |R_{\text{in}}(h) - R_{\text{out}}(h)|$.

$$\left| \int R_{\text{out}} \mathrm{d}Q - \int R_{\text{in}} \mathrm{d}Q \right|$$

Jensen $$\leqslant \int \Delta \mathrm{d}Q$$

Change of measure $$= \int \Delta \frac{\mathrm{d}Q}{\mathrm{d}P} \mathrm{d}P$$

Holder $$\leqslant \left(\int \Delta^q \mathrm{d}P \right)^{\frac{1}{q}} \left(\int \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \right)^p \mathrm{d}P \right)^{\frac{1}{p}}$$

Markov $$\underset{1-\delta}{\leqslant} \left(\frac{\mathbb{E} \int \Delta^q \mathrm{d}P}{\delta} \right)^{\frac{1}{q}} \left(\int \left(\frac{\mathrm{d}Q}{\mathrm{d}P} \right)^p \mathrm{d}P \right)^{\frac{1}{p}}$$

Proof

Let $\Delta(h) := |R_{\mathrm{in}}(h) - R_{\mathrm{out}}(h)|$.

$$\left|\int R_{\mathrm{out}}\,\mathrm{d}Q - \int R_{\mathrm{in}}\,\mathrm{d}Q\right|$$

Jensen $$\leqslant \int \Delta \,\mathrm{d}Q$$

Change of measure $$= \int \Delta \frac{\mathrm{d}Q}{\mathrm{d}P}\,\mathrm{d}P$$

Holder $$\leqslant \left(\int \Delta^q \mathrm{d}P\right)^{\frac{1}{q}} \left(\int \left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right)^p \mathrm{d}P\right)^{\frac{1}{p}}$$

Markov $$\underset{1-\delta}{\leqslant} \left(\frac{\mathbb{E}\int \Delta^q \mathrm{d}P}{\delta}\right)^{\frac{1}{q}} \left(\int \left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right)^p \mathrm{d}P\right)^{\frac{1}{p}}$$

$$= \left(\frac{\mathcal{M}_q}{\delta}\right)^{\frac{1}{q}} \left(D_{\phi_p - 1}(Q, P) + 1\right)^{\frac{1}{p}}.$$

Oracle bounds

Oracle bounds

Catoni (2004, 2007) further derived PAC-Bayesian bound for the Gibbs posterior

$$Q_\lambda \propto \exp\left(-\lambda R_{\mathrm{in}}\right) P.$$

Oracle bounds

Catoni (2004, 2007) further derived PAC-Bayesian bound for the Gibbs posterior

$$Q_\lambda \propto \exp\left(-\lambda R_{\text{in}}\right) P.$$

Assume that the loss is upper-bounded by B, for any $\lambda > 0$, with probability greater that $1 - \delta$

$$R_{\text{out}}(Q_\lambda) \leqslant \inf_{Q \ll P} \left\{ R_{\text{out}}(Q) + \frac{\lambda B}{m} + \frac{2}{\lambda}\left(\mathrm{KL}(Q, P) + \log \frac{2}{\delta}\right)\right\}$$

(can be optimized with respect to λ)

Oracle bounds

Catoni (2004, 2007) further derived PAC-Bayesian bound for the Gibbs posterior

$$Q_\lambda \propto \exp\left(-\lambda R_{\mathrm{in}}\right) P.$$

Assume that the loss is upper-bounded by B, for any $\lambda > 0$, with probability greater that $1 - \delta$

$$R_{\mathrm{out}}(Q_\lambda) \leqslant \inf_{Q \ll P} \left\{ R_{\mathrm{out}}(Q) + \frac{\lambda B}{m} + \frac{2}{\lambda}\left(\mathrm{KL}(Q, P) + \log\frac{2}{\delta}\right)\right\}$$

(can be optimized with respect to λ)

Pros: Q_λ now enjoys stronger guarantees as its performance is comparable to the (forever unknown) oracle.
Cons: the right-hand side is no longer computable.

The plan

1. Elements of Statistical Learning

2. The PAC-Bayesian Theory

3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

The plan

1. Elements of Statistical Learning

2. The PAC-Bayesian Theory

3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

Data- or distribution-dependent priors

Data- or distribution-dependent priors

PAC-Bayesian bounds express a tradeoff between empirical accuracy and a measure of complexity

$$R_{\mathrm{out}}(Q) \leqslant R_{\mathrm{in}}(Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln \frac{\xi(m)}{\delta}}{2m}}.$$

Data- or distribution-dependent priors

PAC-Bayesian bounds express a tradeoff between empirical accuracy and a measure of complexity

$$R_{\mathrm{out}}(Q) \leqslant R_{\mathrm{in}}(Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln \frac{\xi(m)}{\delta}}{2m}}.$$

How can this complexity be controlled?

Data- or distribution-dependent priors

PAC-Bayesian bounds express a tradeoff between empirical accuracy and a measure of complexity

$$R_{\mathrm{out}}(Q) \leqslant R_{\mathrm{in}}(Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln \frac{\xi(m)}{\delta}}{2m}}.$$

How can this complexity be controlled?

- An important component in the PAC-Bayes analysis is the choice of the **prior distribution**
- The results hold whatever the choice of prior, provided that it is chosen *before* seeing the data sample
- Are there ways we can choose a 'better' prior?
- Will explore:

Data- or distribution-dependent priors

PAC-Bayesian bounds express a tradeoff between empirical accuracy and a measure of complexity

$$R_{\mathrm{out}}(Q) \leqslant R_{\mathrm{in}}(Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln \frac{\xi(m)}{\delta}}{2m}}.$$

How can this complexity be controlled?

- An important component in the PAC-Bayes analysis is the choice of the **prior distribution**
- The results hold whatever the choice of prior, provided that it is chosen *before* seeing the data sample
- Are there ways we can choose a 'better' prior?
- Will explore:
  - using part of the data to *learn the prior* for SVMs, but also more interestingly and more generally

Data- or distribution-dependent priors

PAC-Bayesian bounds express a tradeoff between empirical accuracy and a measure of complexity

$$R_{\text{out}}(Q) \leqslant R_{\text{in}}(Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln \frac{\xi(m)}{\delta}}{2m}}.$$

How can this complexity be controlled?

- An important component in the PAC-Bayes analysis is the choice of the **prior distribution**
- The results hold whatever the choice of prior, provided that it is chosen *before* seeing the data sample
- Are there ways we can choose a 'better' prior?
- Will explore:
  - using part of the data to *learn the prior* for SVMs, but also more interestingly and more generally
  - defining the prior in terms of the *data generating distribution* *(aka localised PAC-Bayes)*.

SVM Application

- Prior and posterior distributions are spherical Gaussians:

SVM Application

- Prior and posterior distributions are spherical Gaussians:
    - Prior centered at the origin

SVM Application

- Prior and posterior distributions are spherical Gaussians:
  - Prior centered at the origin
  - Posterior centered at a scaling μ of the unit SVM weight vector

SVM Application

- Prior and posterior distributions are spherical Gaussians:
  - Prior centered at the origin
  - Posterior centered at a scaling μ of the unit SVM weight vector
- Implies KL term is $\mu^2/2$

SVM Application

- Prior and posterior distributions are spherical Gaussians:
  - Prior centered at the origin
  - Posterior centered at a scaling μ of the unit SVM weight vector
- Implies KL term is $\mu^2/2$
- We can compute the stochastic error of the posterior distribution exactly and it behaves like a *soft margin*; scaling μ trades between margin loss and KL

SVM Application

- Prior and posterior distributions are spherical Gaussians:
  - Prior centered at the origin
  - Posterior centered at a scaling μ of the unit SVM weight vector
- Implies KL term is $\mu^2/2$
- We can compute the stochastic error of the posterior distribution exactly and it behaves like a *soft margin*; scaling μ trades between margin loss and KL
- Bound holds for all μ, so choose to optimise the bound

SVM Application

- Prior and posterior distributions are spherical Gaussians:
  - Prior centered at the origin
  - Posterior centered at a scaling μ of the unit SVM weight vector
- Implies KL term is $\mu^2/2$
- We can compute the stochastic error of the posterior distribution exactly and it behaves like a *soft margin*; scaling μ trades between margin loss and KL
- Bound holds for all μ, so choose to optimise the bound
- Generalization of deterministic classifier can be bounded by *twice* stochastic error

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data
- Introduce the learnt prior **in the bound**

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data
- Introduce the learnt prior **in the bound**
- Compute stochastic error with **remaining data**: **PrPAC**

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data
- Introduce the learnt prior **in the bound**
- Compute stochastic error with **remaining data**: **PrPAC**
- We can go a step further:

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data
- Introduce the learnt prior **in the bound**
- Compute stochastic error with **remaining data**: **PrPAC**
- We can go a step further:
  - Consider scaling the prior in the chosen direction: **τ-PrPAC**

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data
- Introduce the learnt prior **in the bound**
- Compute stochastic error with **remaining data**: **PrPAC**
- We can go a step further:
  - Consider scaling the prior in the chosen direction: **τ-PrPAC**
  - adapt the SVM algorithm to optimise the new bound: **η-Prior SVM**

Learning the prior for SVMs

- Bound depends on the **distance between prior and posterior**
- Better prior (closer to posterior) would lead to **tighter bound**
- **Learn** the prior P with part of the data
- Introduce the learnt prior **in the bound**
- Compute stochastic error with **remaining data**: **PrPAC**
- We can go a step further:
  - Consider scaling the prior in the chosen direction: **τ-PrPAC**
  - adapt the SVM algorithm to optimise the new bound: **η-Prior SVM**
- We present some results to show the bounds and their use in model selection (regularisation and band-width of kernel).

Results

		Classifier					
		SVM				ηPrior SVM	
Problem		2FCV	10FCV	PAC	PrPAC	PrPAC	τ-PrPAC
digits	Bound	–	–	0.175	0.107	0.050	**0.047**
	TE	**0.007**	**0.007**	**0.007**	0.014	0.010	0.009
waveform	Bound	–	–	0.203	0.185	0.178	**0.176**
	TE	0.090	0.086	**0.084**	0.088	0.087	0.086
pima	Bound	–	–	0.424	0.420	0.428	**0.416**
	TE	0.244	0.245	**0.229**	**0.229**	0.233	0.233
ringnorm	Bound	–	–	0.203	0.110	0.053	**0.050**
	TE	**0.016**	**0.016**	0.018	0.018	**0.016**	**0.016**
spam	Bound	–	–	0.254	0.198	0.186	**0.178**
	TE	0.066	**0.063**	0.067	0.077	0.070	0.072

Results

- Bounds are remarkably tight: for final column average factor between bound and TE is under 3.

Results

- Bounds are remarkably tight: for final column average factor between bound and TE is under 3.

- Model selection from the bounds is as good as 10FCV: in fact all but one of the PAC-Bayes model selections give better averages for TE.

Results

- Bounds are remarkably tight: for final column average factor between bound and TE is under 3.

- Model selection from the bounds is as good as 10FCV: in fact all but one of the PAC-Bayes model selections give better averages for TE.

- The better bounds do not appear to give better model selection - best model selection is from the simplest bound.
  Ambroladze et al. (2007), *Germain et al. (2009a)*

Distribution-defined priors

Distribution-defined priors

- Consider P and Q are Gibbs-Boltzmann distributions

$$P_\gamma(h) := \frac{1}{Z'} e^{-\gamma R_{\mathrm{out}}(h)} \qquad Q_\gamma(h) := \frac{1}{Z} e^{-\gamma R_{\mathrm{in}}(h)}$$

Distribution-defined priors

- Consider P and Q are Gibbs-Boltzmann distributions

$$P_\gamma(h) := \frac{1}{Z'} e^{-\gamma R_{\text{out}}(h)} \qquad Q_\gamma(h) := \frac{1}{Z} e^{-\gamma R_{\text{in}}(h)}$$

- These distributions are hard to work with since we cannot apply the bound to a single weight vector, but the bounds can be very tight:

$$\mathrm{kl}(R_{\text{in}}(Q_\gamma)\|R_{\text{out}}(Q_\gamma)) \leqslant \frac{1}{m}\left(\frac{\gamma}{\sqrt{m}}\sqrt{\ln\frac{8\sqrt{m}}{\delta}} + \frac{\gamma^2}{4m} + \ln\frac{4\sqrt{m}}{\delta}\right)$$

with the only uncertainty the dependence on γ.

Catoni (2003), *Catoni (2007)*, *Lever et al. (2010)*

Observations

- We cannot compute the prior distribution P or even sample from it:

Observations

- We cannot compute the prior distribution P or even sample from it:
  - Note that this would not be possible to consider in normal Bayesian inference;

Observations

- We cannot compute the prior distribution P or even sample from it:
  - Note that this would not be possible to consider in normal Bayesian inference;
  - Trick here is that the error measures only depend on the posterior Q, while the bound depends on KL between posterior and prior: an estimate of this KL is made without knowing the prior explicitly

Observations

- We cannot compute the prior distribution P or even sample from it:
  - Note that this would not be possible to consider in normal Bayesian inference;
  - Trick here is that the error measures only depend on the posterior Q, while the bound depends on KL between posterior and prior: an estimate of this KL is made without knowing the prior explicitly

- The Gibbs distributions are hard to sample from so not easy to work with this bound.

Other distribution defined priors

Other distribution defined priors

- An alternative distribution defined prior for an SVM is to place symmetrical Gaussian at the weight vector: $\mathbf{w}_p = \mathbb{E}_{(\mathbf{x},y)\sim D}(y\,\boldsymbol{\phi}(\mathbf{x}))$ to give distributions that are easier to work with, but results not impressive...

Other distribution defined priors

- An alternative distribution defined prior for an SVM is to place symmetrical Gaussian at the weight vector: $\mathbf{w}_p = \mathbb{E}_{(\mathbf{x},y)\sim D}(y\,\boldsymbol{\phi}(\mathbf{x}))$ to give distributions that are easier to work with, but results not impressive...

- What if we were to take the expected weight vector returned from a random training set of size m: then the KL between posterior and prior is related to the concentration of weight vectors from different training sets

Other distribution defined priors

- An alternative distribution defined prior for an SVM is to place symmetrical Gaussian at the weight vector: $\mathbf{w}_p = \mathbb{E}_{(\mathbf{x},y)\sim D}(y\,\boldsymbol{\phi}(\mathbf{x}))$ to give distributions that are easier to work with, but results not impressive...

- What if we were to take the expected weight vector returned from a random training set of size m: then the KL between posterior and prior is related to the concentration of weight vectors from different training sets

- This is connected to stability...

The plan

1. Elements of Statistical Learning

2. The PAC-Bayesian Theory

3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

Stability

Stability

Uniform hypothesis sensitivity β at sample size m:

$$\| \mathsf{A}(z_{1:m}) - \mathsf{A}(z'_{1:m}) \| \leqslant \beta \sum_{i=1}^{m} \mathbf{1}[z_i \neq z'_i]$$

$(z_1, \dots, z_m)$

- $\mathsf{A}(z_{1:m}) \in \mathcal{H}$ normed space
- $w_m = \mathsf{A}(z_{1:m})$ 'weight vector'

$(z'_1, \dots, z'_m)$

- Lipschitz
- smoothness

Uniform loss sensitivity β at sample size m:

$$|\ell(\mathsf{A}(z_{1:m}), z) - \ell(\mathsf{A}(z'_{1:m}), z)| \leqslant \beta \sum_{i=1}^{m} \mathbf{1}[z_i \neq z'_i]$$

Stability

Uniform hypothesis sensitivity β at sample size m:

$$\| \mathsf{A}(z_{1:m}) - \mathsf{A}(z'_{1:m}) \| \leqslant \beta \sum_{i=1}^{m} \mathbf{1}[z_i \neq z'_i]$$

$(z_1, \ldots, z_m)$

- $\mathsf{A}(z_{1:m}) \in \mathcal{H}$ normed space
- $w_m = \mathsf{A}(z_{1:m})$ 'weight vector'

$(z'_1, \ldots, z'_m)$

- Lipschitz
- smoothness

Uniform loss sensitivity β at sample size m:

$$|\ell(\mathsf{A}(z_{1:m}), z) - \ell(\mathsf{A}(z'_{1:m}), z)| \leqslant \beta \sum_{i=1}^{m} \mathbf{1}[z_i \neq z'_i]$$

- worst-case
- data-insensitive
- distribution-insensitive
- Open: data-dependent?

Generalization from Stability

If A has sensitivity β at sample size m, then for any $\delta \in (0, 1)$,

w.p. $\geqslant 1 - \delta$, $\quad R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \epsilon(\beta, m, \delta)$

Bousquet and Elisseeff (2002)

Generalization from Stability

If A has sensitivity β at sample size m, then for any $\delta \in (0, 1)$,

w.p. $\geqslant 1 - \delta, \qquad R_{\text{out}}(h) \leqslant R_{\text{in}}(h) + \epsilon(\beta, m, \delta)$

Bousquet and Elisseeff (2002)

- the intuition is that if individual examples do not affect the loss of an algorithm then it will be concentrated

Generalization from Stability

If A has sensitivity β at sample size m, then for any $\delta \in (0, 1)$,

w.p. $\geqslant 1 - \delta$, $\quad R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \epsilon(\beta, m, \delta)$

Bousquet and Elisseeff (2002)

- the intuition is that if individual examples do not affect the loss of an algorithm then it will be concentrated
- can be applied to kernel methods where β is related to the regularisation constant, but bounds are quite weak

Generalization from Stability

If A has sensitivity β at sample size m, then for any $\delta \in (0, 1)$,

w.p. $\geqslant 1 - \delta$, $\quad R_{\mathrm{out}}(h) \leqslant R_{\mathrm{in}}(h) + \epsilon(\beta, m, \delta)$

Bousquet and Elisseeff (2002)

- the intuition is that if individual examples do not affect the loss of an algorithm then it will be concentrated
- can be applied to kernel methods where β is related to the regularisation constant, but bounds are quite weak
- question: algorithm output is highly concentrated
  $\Longrightarrow$ stronger results?

Stability + PAC-Bayes

If A has uniform hypothesis stability β at sample size m, then for any $\delta \in (0,1)$, w.p. $\geqslant 1-2\delta$,

$$\mathrm{kl}\big(R_{\mathrm{in}}(Q) \| R_{\mathrm{out}}(Q)\big) \leqslant \frac{\frac{m\beta^2}{2\sigma^2}\left(1+\sqrt{\frac{1}{2}\log\left(\frac{1}{\delta}\right)}\right)^2 + \log\left(\frac{m+1}{\delta}\right)}{m}$$

Gaussian randomization

- $P = \mathcal{N}(\mathbb{E}[W_m], \sigma^2 I)$
- $Q = \mathcal{N}(W_m, \sigma^2 I)$
- $\mathrm{KL}(Q\|P) = \frac{1}{2\sigma^2}\|W_m - \mathbb{E}[W_n]\|^2$

Main proof components:

- w.p. $\geqslant 1-\delta$, $\quad \mathrm{kl}\big(R_{\mathrm{in}}(Q) \| R_{\mathrm{out}}(Q)\big) \leqslant \frac{\mathrm{KL}(Q\|Q_0) + \log\left(\frac{m+1}{\delta}\right)}{m}$
- w.p. $\geqslant 1-\delta$, $\quad \|W_m - \mathbb{E}[W_m]\| \leqslant \sqrt{m}\,\beta\left(1+\sqrt{\frac{1}{2}\log\left(\frac{1}{\delta}\right)}\right)$

Dziugaite and Roy (2018a), *Rivasplata et al. (2018)*

The plan

1. Elements of Statistical Learning

2. The PAC-Bayesian Theory

3. State-of-the-art PAC-Bayes results: a case study
   - Localized PAC-Bayes: data- or distribution-dependent priors
   - Stability and PAC-Bayes
   - PAC-Bayes analysis of deep neural networks

Is deep learning breaking the statistical paradigm we know?

Is deep learning breaking the statistical paradigm we know?

Neural networks architectures trained on massive datasets achieve zero training error which does not bode well for their performance...

Is deep learning breaking the statistical paradigm we know?

Neural networks architectures trained on massive datasets achieve zero training error which does not bode well for their performance...

... yet they also achieve remarkably low errors on test sets!

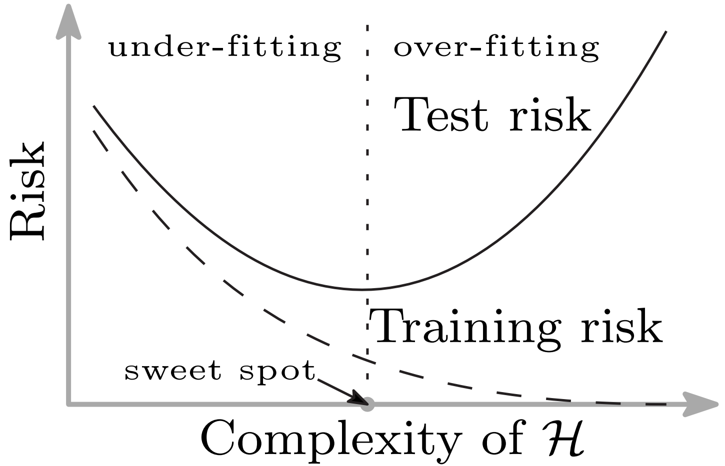
Is deep learning breaking the statistical paradigm we know?

Neural networks architectures trained on massive datasets achieve zero training error which does not bode well for their performance...

... yet they also achieve remarkably low errors on test sets!

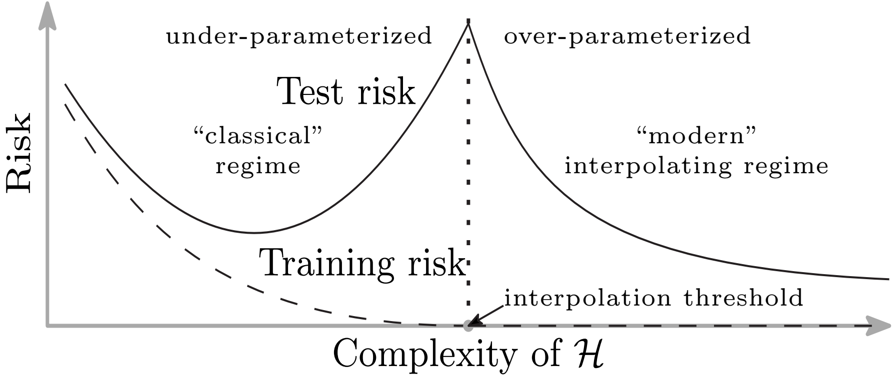
PAC-Bayes is a solid candidate to better understand how deep nets generalize.

The celebrated bias-variance tradeoff



Belkin et al. (2018)

Towards a better understanding of deep nets



Belkin et al. (2018)

Performance of deep nets

Performance of deep nets

- Deep learning has thrown down a challenge to Statistical Learning Theory: outstanding performance with overly complex hypothesis classes (most bounds turn vacuous)

Performance of deep nets

- Deep learning has thrown down a challenge to Statistical Learning Theory: outstanding performance with overly complex hypothesis classes (most bounds turn vacuous)
- For SVMs we can think of the margin as capturing an accuracy with which we need to estimate the weights

Performance of deep nets

- Deep learning has thrown down a challenge to Statistical Learning Theory: outstanding performance with overly complex hypothesis classes (most bounds turn vacuous)

- For SVMs we can think of the margin as capturing an accuracy with which we need to estimate the weights

- If we have a deep network solution with a wide basin of good performance we can take a similar approach using PAC-Bayes with a broad posterior around the solution

Performance of deep nets

Performance of deep nets

- *Dziugaite and Roy (2017)*, *Neyshabur et al. (2017)* have derived some of the tightest deep learning bounds in this way

Performance of deep nets

- *Dziugaite and Roy (2017)*, *Neyshabur et al. (2017)* have derived some of the tightest deep learning bounds in this way
  - by training to expand the basin of attraction

Performance of deep nets

- *Dziugaite and Roy (2017)*, *Neyshabur et al. (2017)* have derived some of the tightest deep learning bounds in this way
  - by training to expand the basin of attraction
  - hence not measuring good generalisation of normal training

Performance of deep nets

- *Dziugaite and Roy (2017)*, *Neyshabur et al. (2017)* have derived some of the tightest deep learning bounds in this way
  - by training to expand the basin of attraction
  - hence not measuring good generalisation of normal training
  - *Dziugaite and Roy (2017)* have also tried to apply the *Lever et al. (2013)* bound but observed cannot measure generalisation correctly for deep networks as has no way of distinguishing between successful fitting of true and random labels

Performance of deep nets

- *Dziugaite and Roy (2017)*, *Neyshabur et al. (2017)* have derived some of the tightest deep learning bounds in this way
  - by training to expand the basin of attraction
  - hence not measuring good generalisation of normal training
  - *Dziugaite and Roy (2017)* have also tried to apply the *Lever et al. (2013)* bound but observed cannot measure generalisation correctly for deep networks as has no way of distinguishing between successful fitting of true and random labels
- There have also been suggestions that stability of SGD is important in obtaining good generalization *(see Dziugaite and Roy (2018b))*

Performance of deep nets

- *Dziugaite and Roy (2017)*, *Neyshabur et al. (2017)* have derived some of the tightest deep learning bounds in this way
  - by training to expand the basin of attraction
  - hence not measuring good generalisation of normal training
  - *Dziugaite and Roy (2017)* have also tried to apply the *Lever et al. (2013)* bound but observed cannot measure generalisation correctly for deep networks as has no way of distinguishing between successful fitting of true and random labels
- There have also been suggestions that stability of SGD is important in obtaining good generalization *(see Dziugaite and Roy (2018b))*
- We presented stability approach combining with PAC-Bayes: this results in a new learning principle linked to recent analysis of information stored in weights

Information contained in training set

Information contained in training set

- *Achille and Soatto (2018)* studied the amount of information stored in the weights of deep networks

Information contained in training set

- *Achille and Soatto (2018)* studied the amount of information stored in the weights of deep networks
- Overfitting is related to information being stored in the weights that encodes the particular training set, as opposed to the data generating distribution

Information contained in training set

- *Achille and Soatto (2018)* studied the amount of information stored in the weights of deep networks
- Overfitting is related to information being stored in the weights that encodes the particular training set, as opposed to the data generating distribution
- This corresponds to reducing the concentration of the distribution of weight vectors output by the algorithm

Information contained in training set

- *Achille and Soatto (2018)* studied the amount of information stored in the weights of deep networks
- Overfitting is related to information being stored in the weights that encodes the particular training set, as opposed to the data generating distribution
- This corresponds to reducing the concentration of the distribution of weight vectors output by the algorithm
- They argue that the Information Bottleneck criterion introduced by *Tishby et al. (1999)* can control this information: hence could potentially lead to a tighter PAC-Bayes bound

Information contained in training set

- *Achille and Soatto (2018)* studied the amount of information stored in the weights of deep networks
- Overfitting is related to information being stored in the weights that encodes the particular training set, as opposed to the data generating distribution
- This corresponds to reducing the concentration of the distribution of weight vectors output by the algorithm
- They argue that the Information Bottleneck criterion introduced by *Tishby et al. (1999)* can control this information: hence could potentially lead to a tighter PAC-Bayes bound
- Potential for algorithms that optimize the bound

Conclusion

Conclusion

- PAC-Bayes arises from two fields:

Conclusion

- PAC-Bayes arises from two fields:
  - Statistical learning theory

Conclusion

- PAC-Bayes arises from two fields:
  - Statistical learning theory
  - Bayesian learning

Conclusion

- PAC-Bayes arises from two fields:
  - Statistical learning theory
  - Bayesian learning
- As such, it generalizes both in interesting and promising directions.

Conclusion

- PAC-Bayes arises from two fields:
  - Statistical learning theory
  - Bayesian learning
- As such, it generalizes both in interesting and promising directions.
- We believe PAC-Bayes can be an inspiration towards

Conclusion

- PAC-Bayes arises from two fields:
  - Statistical learning theory
  - Bayesian learning
- As such, it generalizes both in interesting and promising directions.
- We believe PAC-Bayes can be an inspiration towards
  - new theoretical analyses

Conclusion

- PAC-Bayes arises from two fields:
    - Statistical learning theory
    - Bayesian learning
- As such, it generalizes both in interesting and promising directions.
- We believe PAC-Bayes can be an inspiration towards
    - new theoretical analyses
    - but also drive novel algorithms design, especially in settings where theory has proven difficult.

Acknowledgments

We warmly thank our many co-authors on PAC-Bayes, with a special mention to Omar Rivasplata, François Laviolette and Pascal Germain who helped shape this tutorial.

We both acknowledge the generous support of

- UK Defence Science and Technology Laboratory (Dstl)
- Engineering and Physical Research Council (EPSRC)

Benjamin also acknowledges support from the French funding Agency (ANR) and Inria.

Acknowledgments

We warmly thank our many co-authors on PAC-Bayes, with a special mention to Omar Rivasplata, François Laviolette and Pascal Germain who helped shape this tutorial.

We both acknowledge the generous support of

- UK Defence Science and Technology Laboratory (Dstl)
- Engineering and Physical Research Council (EPSRC)

Benjamin also acknowledges support from the French funding Agency (ANR) and Inria.

Thank you!

Slides available on
`https://bguedj.github.io/icml2019/index.html`

References I

A. Achille and S. Soatto. Emergence of invariance and disentanglement in deep representations. *Journal of Machine Learning Research*, 19(50):1–34, 2018. URL `http://jmlr.org/papers/v19/17-646.html`.

P. Alquier and G. Biau. Sparse single-index model. *Journal of Machine Learning Research*, 14:243–280, 2013.

P. Alquier and B. Guedj. Simpler PAC-Bayesian bounds for hostile data. *Machine Learning*, 107(5):887–902, 2018.

P. Alquier and K. Lounici. PAC-Bayesian theorems for sparse regression estimation with exponential weights. *Electronic Journal of Statistics*, 5:127–145, 2011.

P. Alquier, J. Ridgway, and N. Chopin. On the properties of variational approximations of Gibbs posteriors. *The Journal of Machine Learning Research*, 17(1):8374–8414, 2016.

A. Ambroladze, E. Parrado-Hernández, and J. Shawe-taylor. Tighter PAC-Bayes bounds. In *Advances in Neural Information Processing Systems, NIPS*, pages 9–16, 2007.

J.-Y. Audibert and O. Bousquet. Combining PAC-Bayesian and generic chaining bounds. *Journal of Machine Learning Research*, 2007.

L. Bégin, P. Germain, F. Laviolette, and J.-F. Roy. PAC-Bayesian theory for transductive learning. In *AISTATS*, 2014.

L. Bégin, P. Germain, F. Laviolette, and J.-F. Roy. PAC-Bayesian bounds based on the Rényi divergence. In *AISTATS*, 2016.

M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine learning and the bias-variance trade-off. *arXiv preprint arXiv:1812.11118*, 2018.

O. Bousquet and A. Elisseeff. Stability and generalization. *Journal of machine learning research*, 2(Mar):499–526, 2002.

O. Catoni. A PAC-Bayesian approach to adaptive classification, 2003.

O. Catoni. *Statistical Learning Theory and Stochastic Optimization*. École d'Été de Probabilités de Saint-Flour 2001. Springer, 2004.

O. Catoni. *PAC-Bayesian Supervised Classification: The Thermodynamics of Statistical Learning*, volume 56 of *Lecture notes – Monograph Series*. Institute of Mathematical Statistics, 2007.

I. Csiszár. I-divergence geometry of probability distributions and minimization problems. *Annals of Probability*, 3:146–158, 1975.

P. Derbeko, R. El-Yaniv, and R. Meir. Explicit learning curves for transduction and application to clustering and compression algorithms. *J. Artif. Intell. Res. (JAIR)*, 22, 2004.

References II

M. D. Donsker and S. S. Varadhan. Asymptotic evaluation of certain Markov process expectations for large time. *Communications on Pure and Applied Mathematics*, 28, 1975.

G. K. Dziugaite and D. M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *Proceedings of Uncertainty in Artificial Intelligence (UAI)*, 2017.

G. K. Dziugaite and D. M. Roy. Data-dependent PAC-Bayes priors via differential privacy. In *NeurIPS*, 2018a.

G. K. Dziugaite and D. M. Roy. Entropy-SGD optimizes the prior of a PAC-Bayes bound: Generalization properties of Entropy-SGD and data-dependent priors. In *International Conference on Machine Learning*, pages 1376–1385, 2018b.

M. M. Fard and J. Pineau. PAC-Bayesian model selection for reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2010.

M. M. Fard, J. Pineau, and C. Szepesvári. PAC-Bayesian Policy Evaluation for Reinforcement Learning. In *UAI, Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 195–202, 2011.

S. Gerchinovitz. *Prédiction de suites individuelles et cadre statistique classique : étude de quelques liens autour de la régression parcimonieuse et des techniques d'agrégation*. PhD thesis, Université Paris-Sud, 2011.

P. Germain. *Généralisations de la théorie PAC-bayésienne pour l'apprentissage inductif, l'apprentissage transductif et l'adaptation de domaine.* PhD thesis, Université Laval, 2015.

P. Germain, A. Lacasse, F. Laviolette, and M. Marchand. PAC-Bayesian learning of linear classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML*, 2009a.

P. Germain, A. Lacasse, M. Marchand, S. Shanian, and F. Laviolette. From PAC-Bayes bounds to KL regularization. In *Advances in Neural Information Processing Systems*, pages 603–610, 2009b.

P. Germain, A. Habrard, F. Laviolette, and E. Morvant. A new PAC-Bayesian perspective on domain adaptation. In *Proceedings of International Conference on Machine Learning*, volume 48, 2016.

M. Ghavamzadeh, S. Mannor, J. Pineau, and A. Tamar. Bayesian reinforcement learning: A survey. *Foundations and Trends in Machine Learning*, 8(5-6):359–483, 2015.

B. Guedj and P. Alquier. PAC-Bayesian estimation and prediction in sparse additive models. *Electron. J. Statist.*, 7:264–291, 2013.

References III

B. Guedj and S. Robbiano. PAC-Bayesian high dimensional bipartite ranking. *Journal of Statistical Planning and Inference*, 196:70 – 86, 2018. ISSN 0378-3758.

M. Higgs and J. Shawe-Taylor. A PAC-Bayes bound for tailored density estimation. In *Proceedings of the International Conference on Algorithmic Learning Theory (ALT)*, 2010.

A. Lacasse, F. Laviolette, M. Marchand, P. Germain, and N. Usunier. PAC-Bayes bounds for the risk of the majority vote and the variance of the Gibbs classifier. In *Advances in Neural information processing systems*, pages 769–776, 2007.

J. Langford. Tutorial on practical prediction theory for classification. *Journal of Machine Learning Research*, 2005.

J. Langford and M. Seeger. Bounds for averaging classifiers. Technical report, Carnegie Mellon, Departement of Computer Science, 2001.

J. Langford and J. Shawe-Taylor. PAC-Bayes & margins. In *Advances in Neural Information Processing Systems (NIPS)*, 2002.

G. Lever, F. Laviolette, and J. Shawe-Taylor. Distribution-dependent PAC-Bayes priors. In *International Conference on Algorithmic Learning Theory*, pages 119–133. Springer, 2010.

G. Lever, F. Laviolette, and J. Shawe-Taylor. Tighter PAC-Bayes bounds through distribution-dependent priors. *Theoretical Computer Science*, 473:4–28, 2013.

C. Li, W. Jiang, and M. Tanner. General oracle inequalities for Gibbs posterior with application to ranking. In *Conference on Learning Theory*, pages 512–521, 2013.

L. Li, B. Guedj, and S. Loustau. A quasi-Bayesian perspective to online clustering. *Electron. J. Statist.*, 12(2):3071–3113, 2018.

B. London. A PAC-Bayesian analysis of randomized learning with application to stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 2931–2940, 2017.

B. London, B. Huang, B. Taskar, and L. Getoor. PAC-Bayesian collective stability. In *Artificial Intelligence and Statistics*, pages 585–594, 2014.

A. Maurer. A note on the PAC-Bayesian Theorem. *arXiv preprint cs/0411099*, 2004.

D. McAllester. Some PAC-Bayesian theorems. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 1998.

D. McAllester. Some PAC-Bayesian theorems. *Machine Learning*, 37, 1999.

References IV

D. McAllester. PAC-Bayesian stochastic model selection. *Machine Learning*, 51(1), 2003a.

D. McAllester. Simplified PAC-Bayesian margin bounds. In *COLT*, 2003b.

B. Neyshabur, S. Bhojanapalli, D. A. McAllester, and N. Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, pages 5947–5956, 2017.

E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor, and S. Sun. PAC-Bayes bounds with data dependent priors. *Journal of Machine Learning Research*, 13:3507–3531, 2012.

O. Rivasplata, E. Parrado-Hernandez, J. Shawe-Taylor, S. Sun, and C. Szepesvari. PAC-Bayes bounds for stable algorithms with instance-dependent priors. In *Advances in Neural Information Processing Systems*, pages 9214–9224, 2018.

M. Seeger. PAC-Bayesian generalization bounds for gaussian processes. *Journal of Machine Learning Research*, 3:233–269, 2002.

M. Seeger. *Bayesian Gaussian Process Models: PAC-Bayesian Generalisation Error Bounds and Sparse Approximations*. PhD thesis, University of Edinburgh, 2003.

Y. Seldin and N. Tishby. PAC-Bayesian analysis of co-clustering and beyond. *Journal of Machine Learning Research*, 11:3595–3646, 2010.

Y. Seldin, P. Auer, F. Laviolette, J. Shawe-Taylor, and R. Ortner. PAC-Bayesian analysis of contextual bandits. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

Y. Seldin, F. Laviolette, N. Cesa-Bianchi, J. Shawe-Taylor, and P. Auer. PAC-Bayesian inequalities for martingales. *IEEE Transactions on Information Theory*, 58(12):7086–7093, 2012.

J. Shawe-Taylor and D. Hardoon. Pac-bayes analysis of maximum entropy classification. In *Proceedings on the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009.

J. Shawe-Taylor and R. C. Williamson. A PAC analysis of a Bayes estimator. In *Proceedings of the 10th annual conference on Computational Learning Theory*, pages 2–9. ACM, 1997. doi: 10.1145/267460.267466.

J. Shawe-Taylor, P. L. Bartlett, R. C. Williamson, and M. Anthony. Structural risk minimization over data-dependent hierarchies. *IEEE Transactions on Information Theory*, 44(5), 1998.

N. Thiemann, C. Igel, O. Wintenberger, and Y. Seldin. A Strongly Quasiconvex PAC-Bayesian Bound. In *International Conference on Algorithmic Learning Theory, ALT*, pages 466–492, 2017.

N. Tishby, F. Pereira, and W. Bialek. The information bottleneck method. In *Allerton Conference on Communication, Control and Computation*, 1999.

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