Data Science Theory and Practices

Generalization and Regularization

Joseph Chuang-Chieh Lin
Dept. CSIE, Tamkang University

7th June 2021

Outline

- Formalization of learning
- Overfitting and uniform convergence
- Occam's razor
- Regularization

Formalization

- Instance space $\mathcal{X} = R^d$
  - Data described by *d* features.
    - Email messages with presence or absence of various types of words.
    - Patient records with the results of various medical tests.
- Learning task:
  - Given $S \subset \mathcal{X}$: training examples
    - Could be labeled.
      - Email messages: each is labeled ‘spam’ nor ‘not spam’.
      - Patient records: each is labeled whether or not they respond well to the treatment.

Formalization (contd.)

- **Goal:** An algorithm using the training examples to produce a classification rule (or regression) that will perform well over **new** data.
  - The key feature of machine learning: **generalization**.
- **How?**

Formalization (contd.)

- **Goal:** An algorithm using the training examples to produce a classification rule (or regression) that will perform well over **new** data.
  - The key feature of machine learning: **generalization**.
- **How?**
  - In general, find a “simple” rule with good performance on the training data.
    - e.g., find highly **indicative words** or **weighting of words** such that weighted sum can be used to classify spam and non-spam emails.
  - Argue that the training data is **representative** of the future data.

Formalization (contd.)

- Assume that

$$\mathcal{X} \sim D, \text{ for some probability distribution } D$$

- Training set S:

  drawn uniformly at random from D.

- c^*: target concept

  denote the subset of $\mathcal{X}$ corresponding to the positive class

  - E.g, all spam emails, all patients who respond well to the treatment.

  – Hence, each training data point is labeled according to whether it belongs to c^* or not.

Formalization (contd.)

- Our goal: produce

$$h \subseteq \mathcal{X}: \text{ hypothesis.}$$

- True error of *h*:

$$\mathrm{err}_D(h) = \Pr(h \Delta c^*)$$

Δ : symmetric difference;

$\Pr$: according to D

  - The probability that *h incorrectly* classifies a data point drawn randomly from *D*.

- Training error of *h*:

$$\mathrm{err}_S(h) = |S \cap (h \Delta c^*)| / |S|$$

the fraction of points in S where h and c^* disagree

Formalization (contd.)

- What if we obtain an h for which $\mathrm{err}_S(h)$ is pretty low while $\mathrm{err}_D(h)$ is high?

Formalization (contd.)

- What if we obtain an h for which $\mathrm{err}_S(h)$ is pretty low while $\mathrm{err}_D(h)$ is high?
  - For example, memorizing all the training examples and predicting positive for each example if and only if it already appeared positively in the training set.

Formalization (contd.)

- What if we obtain an h for which $\text{err}_S(h)$ is pretty low while $\text{err}_D(h)$ is high?
  - For example, memorizing all the training examples and predicting positive for each example if and only if it already appeared positively in the training set.
  - We call it **overfitting**.

Formalization (contd.)

- What if we obtain an h for which $\mathrm{err}_S(h)$ is pretty low while $\mathrm{err}_D(h)$ is high?
  - For example, memorizing all the training examples and predicting positive for each example if and only if it already appeared positively in the training set.
  - We call it **overfitting**.

- Algorithms will typically optimize over the data (training data).

Formalization (contd.)

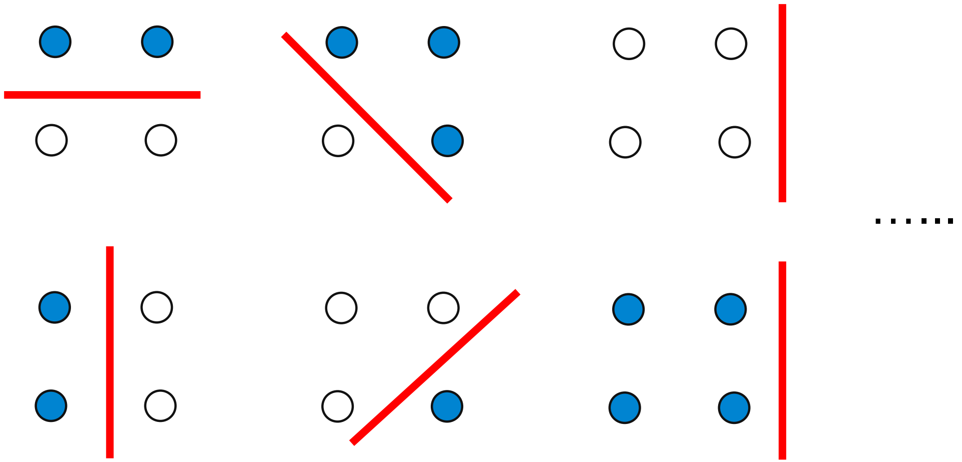
- **Hypothesis class** $\mathcal{H}$ over $\mathcal{X}$: a collection of subsets of $\mathcal{X}$. (assume it finite)
  - Example: the class of linear separators over $\mathcal{X} = R^d$.

$$\{\{\mathbf{x} \in R^d \mid \mathbf{w} \cdot \mathbf{x} \geq w_0\} : \mathbf{w} \in R^d, w_0 \in R\}$$

Formalization (contd.)

- **Hypothesis class** $\mathcal{H}$ over $\mathcal{X}$: a collection of subsets of $\mathcal{X}$ (assume it finite)
  - Example: the class of linear separators over $\mathcal{X} = R^d$.

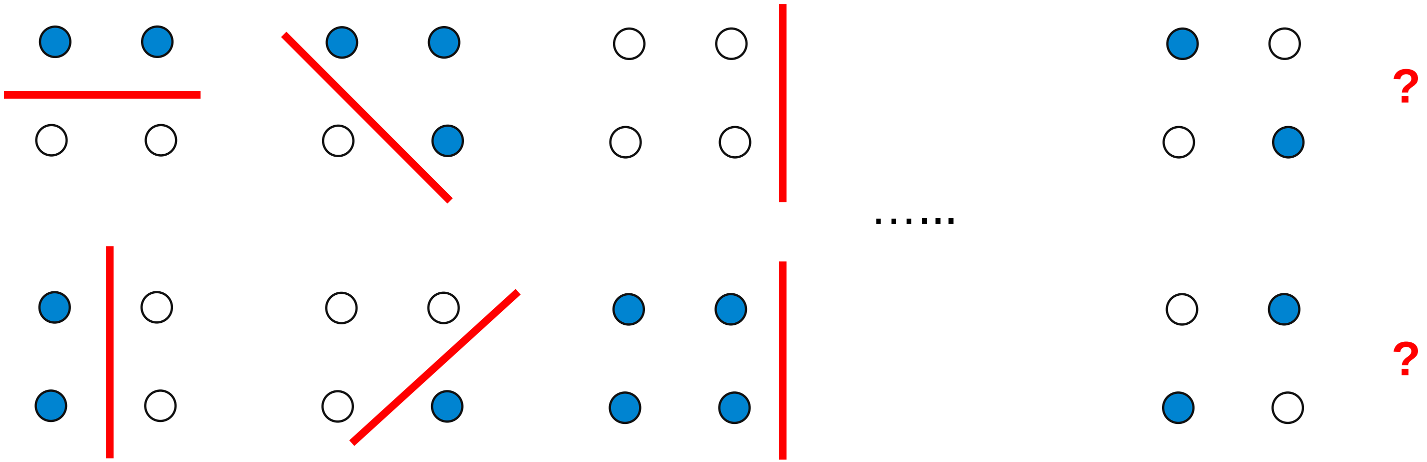
$$\{\{\mathbf{x} \in R^d \mid \mathbf{w} \cdot \mathbf{x} \geq w_0\} : \mathbf{w} \in R^d, w_0 \in R\}$$



Formalization (contd.)

- **Hypothesis class** $\mathcal{H}$ over $\mathcal{X}$: a collection of subsets of $\mathcal{X}$. (assume it finite)
  - Example: the class of linear separators over $\mathcal{X} = R^d$.

$$\{\{\mathbf{x} \in R^d \mid \mathbf{w} \cdot \mathbf{x} \geq w_0\} : \mathbf{w} \in R^d, w_0 \in R\}$$



Formalization (contd.)

- To facilitate our discussion, let

$$h(x) = \begin{cases} 1 & x \in h \\ -1 & x \notin h \end{cases}$$

$$\text{true error: } \mathrm{err}_D(h) = \Pr_{x \sim D}[h(x) \neq c^*(x)]$$

$$\text{training error: } \mathrm{err}_S(h) = \Pr_{x \sim S}[h(x) \neq c^*(x)]$$

Overfitting and uniform convergence

- **Theorem [PAC**-learning guarantee (**P**robably **A**pproximately **C**orrect)].

  Let $\mathscr{H}$ be a hypothesis class and let ε, $\delta > 0$.

  If a training set S of size

$$n \geq \frac{1}{\epsilon}(\ln|\mathcal{H}| + \ln(1/\delta)),$$

  is drawn from distribution D, then with probability $\geq 1-\delta$, every $h \in \mathscr{H}$ with true error $\mathrm{err}_D(h) \geq \varepsilon$ has training error $\mathrm{err}_S(h) > 0$.

Overfitting and uniform convergence

- **Theorem [PAC-**learning guarantee (**P**robably **A**pproximately **C**orrect)].

  Let $\mathscr{H}$ be a hypothesis class and let ε, $\delta > 0$.

  If a training set S of size

$$n \geq \frac{1}{\epsilon}(\ln|\mathcal{H}| + \ln(1/\delta)),$$

  is drawn from distribution D, then with probability $\geq 1-\delta$, every $h \in \mathscr{H}$

  with training error $\mathbf{err}_S(h) = 0$ has true error $\mathbf{err}_D(h) < \varepsilon$.

Let $h_1, h_2, \ldots$ be the hypotheses in $\mathcal{H}$ with $\mathrm{err}_D(h_i) \geq \epsilon$ for each i.

- The hypotheses that we don't want to output.

Consider drawing the sample S of size n.

A_i: the event that $\mathrm{err}_S(h_i) = 0$.

Let $h_1, h_2, \ldots$ be the hypotheses in $\mathcal{H}$ with $\mathrm{err}_D(h_i) \geq \epsilon$ for each i.

– The hypotheses that we don't want to output.

Consider drawing the sample S of size n.

A_i: the event that $\mathrm{err}_S(h_i) = 0$.

Since $\mathrm{err}_D(h_i) \geq \epsilon$, we have

$$\Pr[A_i] \leq (1 - \epsilon)^n.$$

Thus,

$$\Pr\left[\bigcup_i A_i\right] \leq |\mathcal{H}|(1 - \epsilon)^n \leq |\mathcal{H}|e^{-\epsilon n}.$$

Let $h_1, h_2, \ldots$ be the hypotheses in $\mathcal{H}$ with $\mathrm{err}_D(h_i) \geq \epsilon$ for each i.

- The hypotheses that we don't want to output.

Consider drawing the sample S of size n.

A_i: the event that $\mathrm{err}_S(h_i) = 0$.

Since $\mathrm{err}_D(h_i) \geq \epsilon$, we have

$$\Pr[A_i] \leq (1-\epsilon)^n.$$

Using $n \geq \frac{1}{\epsilon}(\ln|\mathcal{H}| + \ln(1/\delta))$

Thus,

$$\Pr\left[\bigcup_i A_i\right] \leq |\mathcal{H}|(1-\epsilon)^n \leq |\mathcal{H}|e^{-\epsilon n}.$$

Remark on PAC Theorem

- What if the best h_i in $\mathcal{H}$ has **> 0% error** (say 5%) on S?
  - Can we still be confident that its true error is low? (say 10%?)

Theorem (Hoeffding bounds; reformulate)

Let $x_1, x_2, \ldots, x_n$ be independent $\{0, 1\}$ random variables with $\Pr[x_i = 1] = p$. Let $s = \sum_i x_i$. Then, for any $0 < \alpha \leq 1$,

$$\Pr\left[\frac{s}{n} > p + \alpha\right] \leq e^{-2n\alpha^2},$$
$$\Pr\left[\frac{s}{n} < p - \alpha\right] \leq e^{-2n\alpha^2},$$

Theorem (Uniform convergence)

Let $\mathcal{H}$ be a hypothesis class and let $\epsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{2\epsilon^2}(\ln|\mathcal{H}| + \ln(2/\delta)),$$

is drawn from distribution D, then with probability $\geq 1 - \delta$, every $h \in \mathcal{H}$ satisfies $|\mathrm{err}_S(h) - \mathrm{err}_D(h)| \leq \epsilon$.

Fix some $h \in \mathcal{H}$, let x_j be the indicator random variable for $\{h$ makes a mistake on the jth sample in $S\}$.

- $\Pr[x_j = 1]$ $=$ the true error of h.
- The fraction of the x_j's equal to 1: the training error $\mathrm{err}_S(h)$ of h.

Fix some $h \in \mathcal{H}$, let x_j be the indicator random variable for $\{h$ makes a mistake on the jth sample in $S\}$.

- $\Pr[x_j = 1] =$ the true error of h.
- The fraction of the x_j's equal to 1: the training error $\mathrm{err}_S(h)$ of h.

Let A_h: $|\mathrm{err}_D(h) - \mathrm{err}_S(h)| > \epsilon$.

By Hoeffding bounds,

- $\Pr[A_h] \leq 2e^{-2n\epsilon^2}$.

Fix some $h \in \mathcal{H}$, let x_j be the indicator random variable for $\{h$ makes a mistake on the jth sample in $S\}$.

- $\Pr[x_j = 1] =$ the true error of h.
- The fraction of the x_j's equal to 1: the training error $\mathrm{err}_S(h)$ of h.

Let A_h: $|\mathrm{err}_D(h) - \mathrm{err}_S(h)| > \epsilon$.

By Hoeffding bounds,

- $\Pr[A_h] \leq 2e^{-2n\epsilon^2}$.
- Union bound: $\Pr[\exists h \in \mathcal{H}$ such that $|\mathrm{err}_D(h) - \mathrm{err}_S(h)| > \epsilon] \leq 2|\mathcal{H}|e^{-2n\epsilon^2}$.

Fix some $h \in \mathcal{H}$, let x_j be the indicator random variable for $\{h$ makes a mistake on the jth sample in $S\}$.

- $\Pr[x_j = 1] =$ the true error of h.
- The fraction of the x_j's equal to 1: the training error $\mathrm{err}_S(h)$ of h.

Let A_h: $|\mathrm{err}_D(h) - \mathrm{err}_S(h)| > \epsilon$.

By Hoeffding bounds,

$$\boxed{n \geq \frac{1}{2\epsilon^2}(\ln|\mathcal{H}| + \ln(2/\delta)),}$$

- $\Pr[A_h] \leq 2e^{-2n\epsilon^2}$.
- Union bound: $\Pr[\exists h \in \mathcal{H}$ such that $|\mathrm{err}_D(h) - \mathrm{err}_S(h)| > \epsilon] \leq 2|\mathcal{H}|e^{-2n\epsilon^2}$.

Description language

- Suppose that the target concept can be represented by a *disjunction* (OR) over *d* features.

Description language

- Suppose that the target concept can be represented by a *disjunction* (OR) over *d* features.

features for non-spam								features for spam								
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	
0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	spam
0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	1	spam
1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	non-spam
0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	1	spam

$$|\mathcal{H}| = 2^d; \;\; \ln(|\mathcal{H}|) = d$$

Description language

- Suppose that the target concept can be represented by a *disjunction* (OR) over *d* features.

features for non-spam								features for spam								
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	
0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	spam
0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	1	spam
1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	non-spam
0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	1	spam

$|\mathcal{H}| = 2^d; \;\; \ln(|\mathcal{H}|) = d$

Description language

- Suppose that the target concept can be represented by a *disjunction* (OR) over *d* features.

features for non-spam

features for spam

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	
0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	spam
0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	1	spam
1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	non-spam
0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	1	spam

$|\mathcal{H}| = 2^d; \;\; \ln(|\mathcal{H}|) = d$

Occam's razor

- William of Occam (1287–1347)
- a.k.a. **Law of Parsimony**.
- In general, one should prefer simpler explanations over more complicated ones.
- Use < b bits for the description language.
  - $1 + 2 + 4 + \cdots + 2^{b-1} < 2^b$.

Mathematical statement of Occam's razor

- **Theorem (Occam's razor)**

  Fix any description language.

  Consider a training sample S drawn from distribution D with $|S| = \frac{1}{\epsilon}\left(b \ln 2 + \ln(1/\delta)\right)$.

  For any rule h

  - $\mathrm{err}_S(h) = 0$
  - h can be described using $< b$ bits,

  $\Pr[\mathrm{err}_D(h) \leq \epsilon] \geq 1 - \delta$.

Mathematical statement of Occam's razor

- **Theorem (Occam's razor)**

  Fix any description language.

  Consider a training sample S drawn from distribution D.

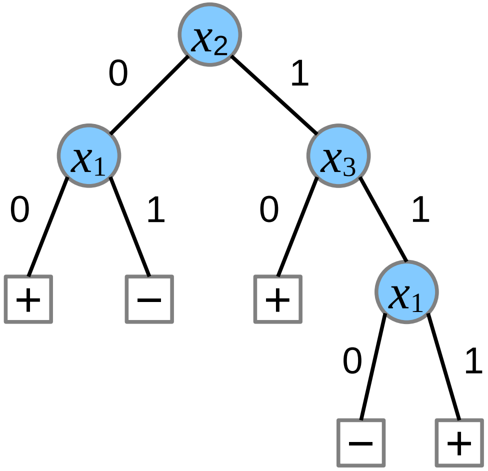
  For any rule h

  – $\mathrm{err}_S(h) = 0$
  – h can be described using $< b$ bits,

  $$\Pr\left[\mathrm{err}_D(h) \le \frac{b\ln(2)+\ln(1/\delta)}{|S|}\right] \ge 1-\delta.$$

Application: Learning decision trees

- Find the smallest decision tree to fit the training sample *S*: **NP**-hard.
- Suppose we run a heuristic *h* on *S* and it outputs a tree with *k* nodes.



$$\bar{x}_1\bar{x}_2 \vee x_1x_2x_3 \vee x_2\bar{x}_3$$

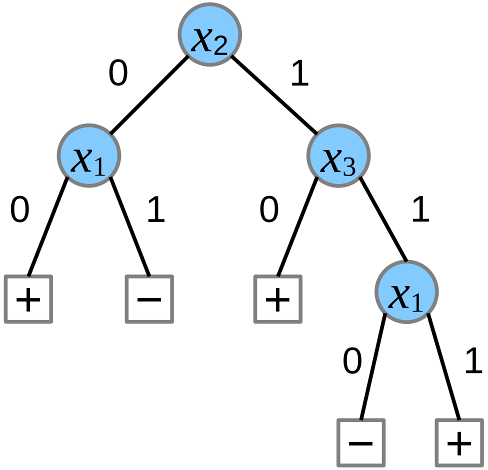
Application: Learning decision trees

- Find the smallest decision tree to fit the training sample S: **NP**-hard.
- Suppose we run a heuristic h on S and it outputs a tree with k nodes.
- Such a tree can be described using $O(k \log d)$ bits.

$\log_2(d)$ bits for the index of the feature in the root.

$O(1)$ bits: indicate if it's a leaf and what label it should have.

$O(k_L \log d)$ bits for left subtree $+ \, O(k_R \log d)$ bits for right subtree.



$\bar{x}_1\bar{x}_2 \vee x_1x_2x_3 \vee x_2\bar{x}_3$

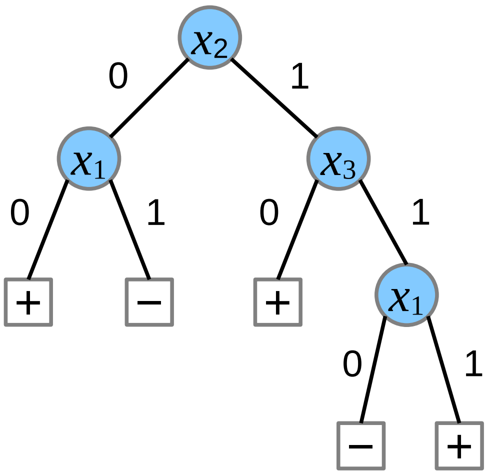
Application: Learning decision trees

- Find the smallest decision tree to fit the training sample S: **NP**-hard.
- Suppose we run a heuristic h on S and it outputs a tree with k nodes.
- Such a tree can be described using $O(k \log d)$ bits.

  By the theorem of Occam's razor, we can be confident that $\mathrm{err}_D(h)$ is low if we can produce a consistent tree with

$$< \epsilon|S|/\log(d) \text{ nodes}$$

$\log_2(d)$ bits for the index of the feature in the root



$\bar{x}_1\bar{x}_2 \vee x_1 x_2 x_3 \vee x_2 \bar{x}_3$

Simple rules? Complex rules?

- Suppose that there is NO simple rule that is perfectly consistent with the training data.

Simple rules? Complex rules?

- Suppose that there is NO simple rule that is perfectly consistent with the training data.
- But,
  - We have a VERY simple rule with training error 20%.
  - We have also a complex rule with training error 10%.
  - …

Simple rules? Complex rules?

- Suppose that there is NO simple rule that is perfectly consistent with the training data.
- But,
  - We have a VERY simple rule with training error 20%.
  - We have also a complex rule with training error 10%.
  - …
- Optimize some combination of **training error** and **simplicity**?

Simple rules? Complex rules?

- Suppose that there is NO simple rule that is perfectly consistent with the training data.
- But,
  - We have a VERY simple rule with training error 20%.
  - We have also a complex rule with training error 10%.
  - …
- Optimize some combination of **training error** and **simplicity**?
  - The notion of **regularization** (**complexity penalization**).

Regularization (penalizing complexity)

- Corollary.

Fix any description language, and consider a training set S drawn from distribution D. With probability $\geq 1-\delta$, every hypothesis h satisfies

$$\mathrm{err}_D(h) \leq \mathrm{err}_S(h) + \sqrt{\frac{\mathrm{size}(h)\ln 4 + \ln(2/\delta)}{2|S|}}$$

where $\mathrm{size}(h)$ denotes the number of bits needed to describe h in the given language.

The idea

Consider fixing some description language.

Let $\mathcal{H}_i$ be the hypotheses that can be described in i bits ($|\mathcal{H}_i| \leq 2^i$).

Let $\delta_i = \delta/2^i$. $\quad \delta_1 + \delta_2 + \cdots = \delta$

The idea

Theorem (Occam's razor)

Fix any description language.

Consider a training sample S drawn from distribution D with $|S| = \frac{1}{\epsilon}(b \ln 2 + \ln(1/\delta))$.

For any rule h

- $\mathrm{err}_S(h) = 0$
- h can be described using $< b$ bits,

$\Pr\left[\mathrm{err}_D(h) \le \frac{b\ln(2)+\ln(1/\delta)}{|S|}\right] \ge 1-\delta.$

Consider fixing some description language.

Let $\mathcal{H}_i$ be the hypotheses that can be described in i bits ($|\mathcal{H}_i| \le 2^i$).

Let $\delta_i = \delta/2^i$. $\quad \delta_1 + \delta_2 + \cdots = \delta$

With probability $\ge 1 - \delta_i$, all $h \in \mathcal{H}_i$ satisfy

$$\mathrm{err}_D(h) \le \mathrm{err}_S(h) + \sqrt{\frac{\ln(|\mathcal{H}_i|) + \ln(2/\delta_i)}{2|S|}}$$

.

The idea

Theorem (Occam's razor)

Fix any description language.

Consider a training sample S drawn from distribution D with $|S| = \frac{1}{\epsilon}(b \ln 2 + \ln(1/\delta))$.

For any rule h

- $\mathrm{err}_S(h) = 0$
- h can be described using $< b$ bits,

$\Pr\left[\mathrm{err}_D(h) \le \frac{b\ln(2)+\ln(1/\delta)}{|S|}\right] \ge 1-\delta$.

Consider fixing some description language.

Let $\mathcal{H}_i$ be the hypotheses that can be described in i bits $(|\mathcal{H}_i| \le 2^i)$.

Let $\delta_i = \delta/2^i$. $\quad \delta_1 + \delta_2 + \cdots = \delta$

With probability $\ge 1-\delta$, all hypothesis h satisfy

$$\mathrm{err}_D(h) \le \mathrm{err}_S(h) + \sqrt{\frac{\mathrm{size}(h) + \ln(2/\delta)}{2|S|}}$$

.

The idea

Theorem (Uniform convergence)

Let $\mathcal{H}$ be a hypothesis class and let $\epsilon, \delta > 0$. If a training set S of size

$$n \geq \frac{1}{2\epsilon^2}(\ln|\mathcal{H}| + \ln(2/\delta)),$$

is drawn from distribution D, then with probability $\geq 1-\delta$, every $h \in \mathcal{H}$ satisfies $|\mathrm{err}_S(h) - \mathrm{err}_D(h)| \leq \epsilon$.

Don't want the model to be too complex

With probability $\geq 1-\delta$, all hypothesis h satisfy

$$\mathrm{err}_D(h) \leq \boxed{\mathrm{err}_S(h) + \sqrt{\frac{\mathrm{size}(h) + \ln(2/\delta)}{2|S|}}}$$

.

Try to minimize the training error

Minimize the right-hand size hopefully