PAC Learning

CPT_S 434/534 Neural network design and application

Core questions to answer

- What can be learned by machine learning models?
- What conditions are required to successfully learn?

Underlying concept function

$$concept\ c\colon \mathcal{X} \to \mathcal{Y}$$

Underlying concept function

An underlying concept function (mapping)

$$concept\ c : \mathcal{X} \rightarrow \mathcal{Y}$$

Underlying concept function

An underlying concept function (mapping)

$$concept\ c : \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

Underlying concept function

An underlying concept function (mapping)

$$concept\ c : \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Underlying concept function

An underlying concept function (mapping)

$$concept\ c: \mathcal{X} \to \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Features

Size
bedrooms
Yard
……

A model

Price?

Image from https://www.rockpapershotgun.com/minecraft-house-ideas

Underlying concept function

An underlying concept function (mapping)

$$concept\ c: \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Features

Size
Color
Fur
……

A model

Husky?

Underlying concept function

An underlying concept function (mapping)

$$concept\ c : \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Features

Color
Shape
Yellow spot
……

A model

Sweet?

Image from https://www.self.com/story/heres-how-to-pick-a-perfect-melon-every-time

Underlying concept function

An underlying concept function (mapping)

$$concept\ c\colon \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Features

Color
Shape
Yellow spot
……

A model

Sweet?

Q: how to measure the model's performance?

Image from https://www.self.com/story/heres-how-to-pick-a-perfect-melon-every-time

Underlying concept function

An underlying concept function (mapping)

$$concept\ c: \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Q: What if our model has completely different behavior with the underlying concept function?

Q: how to measure the model's performance?

Features

Color
Shape
Yellow spot
……

A model

Sweet?

Image from https://www.self.com/story/heres-how-to-pick-a-perfect-melon-every-time

Underlying concept function

An underlying concept function (mapping)

$$concept\ c: \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Q: What if our model has identical behavior with the underlying concept function?

Features

Color
Shape
Yellow spot
……

A model

Sweet?

Q: how to measure the model's performance?

Image from https://www.self.com/story/heres-how-to-pick-a-perfect-melon-every-time

Risk

A hypothesis class

Definition 2.1 (Generalization error) *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and an underlying distribution $\mathcal{D}$, the* generalization error *or* risk *of h is defined by*

$$R(h) = \boxed{\underset{x \sim \mathcal{D}}{\mathbb{P}}[h(x) \neq c(x)]} = \underset{x \sim \mathcal{D}}{\mathbb{E}}\left[1_{h(x) \neq c(x)}\right], \tag{2.1}$$

where 1_ω is the indicator function of the event ω.[2]

Risk

A hypothesis class

Definition 2.1 (Generalization error) *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and an underlying distribution $\mathcal{D}$, the* generalization error *or* risk *of h is defined by*

$$R(h) = \boxed{\mathbb{P}_{x \sim \mathcal{D}}[h(x) \neq c(x)]} = \mathbb{E}_{x \sim \mathcal{D}}\left[1_{h(x) \neq c(x)}\right], \tag{2.1}$$

where 1_ω is the indicator

Risk: in population level
→
scan all samples in the world
(not feasible in general)

Risk

Q: What is it?
A hypothesis class

Definition 2.1 (Generalization error) *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and an underlying distribution $\mathcal{D}$, the* generalization error *or* risk *of h is defined by*

$$R(h) = \boxed{\underset{x \sim \mathcal{D}}{\mathbb{P}}[h(x) \neq c(x)]} = \underset{x \sim \mathcal{D}}{\mathbb{E}}\left[1_{h(x) \neq c(x)}\right], \tag{2.1}$$

where 1_ω is the indicator

Risk: in population level
→
scan all samples in the world
(not feasible in general)

Review: Build a model

- What is a model

A hypothesis class

A linear function

or

An ellipse (nonlinear function)

Another hypothesis class

Q: what are their parameters?

$$y = ax + b$$

Try to separate two classes

Q: how to separate them?

$$\frac{x - {x_0}^2}{a} + \frac{y - {y_0}^2}{b} = 1$$

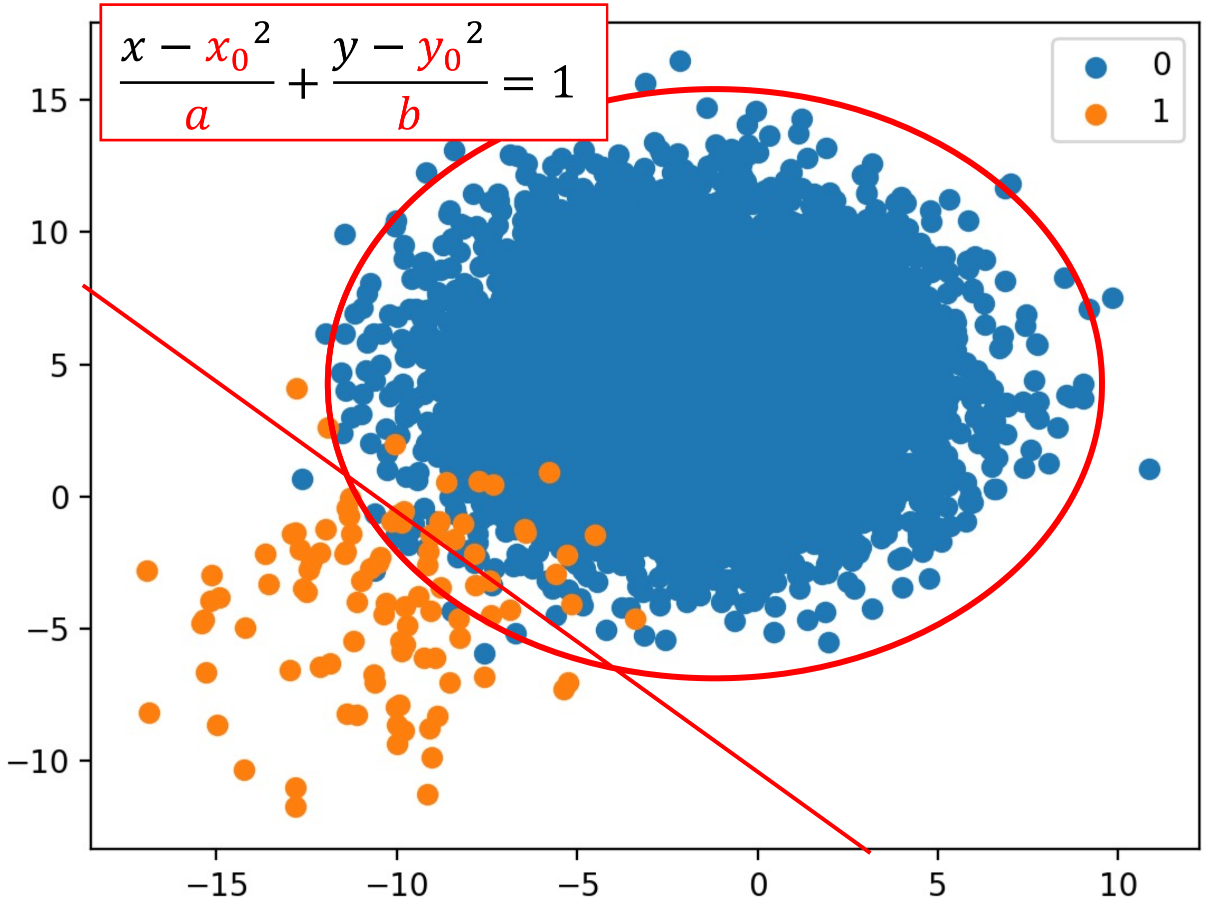


Image retrieved from https://machinelearningmastery.com/how-to-develop-an-intuition-skewed-class-distributions/

Risk

Q: What is it?
A hypothesis class

Definition 2.1 (Generalization error) *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and an underlying distribution $\mathcal{D}$, the* generalization error *or* risk *of h is defined by*

$$R(h) = \boxed{\underset{x \sim \mathcal{D}}{\mathbb{P}}[h(x) \neq c(x)]} = \boxed{\underset{x \sim \mathcal{D}}{\mathbb{E}}\left[1_{h(x) \neq c(x)}\right],} \tag{2.1}$$

where 1_ω is the indicator

Risk: in population level
→
scan all samples in the world
(not feasible in general)

Expected mistakes that h
makes over data distribution D

PAC learning

Definition 2.3 (PAC-learning) *A concept class* $\mathcal{C}$ *is said to be* PAC-learnable *if there exists an algorithm* $\mathcal{A}$ *and a polynomial function* $poly(\cdot,\cdot,\cdot,\cdot)$ *such that for any* $\boxed{\epsilon > 0 \text{ and } \delta > 0}$, *for all distributions* $\mathcal{D}$ *on* $\mathcal{X}$ *and for any target concept* $c \in \mathcal{C}$, *the following holds for any sample size* $m \geq \boxed{poly(1/\epsilon, 1/\delta, n, size(c))}$:

$$\underset{S\sim\mathcal{D}^m}{\mathbb{P}}[R(h_S) \leq \epsilon] \geq 1 - \delta. \tag{2.4}$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n, size(c))$, *then* $\mathcal{C}$ *is said to be* efficiently PAC-learnable. *When such an algorithm* $\mathcal{A}$ *exists, it is called a* PAC-learning algorithm *for* $\mathcal{C}$.

PAC learning

Definition 2.3 (PAC-learning) *A concept class* $\mathcal{C}$ *is said to be* PAC-learnable *if there exists an algorithm* $\mathcal{A}$ *and a polynomial function* $poly(\cdot,\cdot,\cdot,\cdot)$ *such that for any* $\boxed{\epsilon > 0 \text{ and } \delta > 0}$, *for all distributions* $\mathcal{D}$ *on* $\mathcal{X}$ *and for any target concept* $c \in \mathcal{C}$, *the following holds for any sample size* $m \geq \boxed{poly(1/\epsilon, 1/\delta, n, size(c))}$:

$$\underset{S \sim \mathcal{D}^m}{\mathbb{P}}\left[\boxed{R(h_S) \leq \epsilon}\right] \geq \boxed{1 - \delta.} \tag{2.4}$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n, size(c))$, *then* $\mathcal{C}$ *is said to be* efficiently PAC-learnable. *When such an algorithm* $\mathcal{A}$ *exists, it is called a* PAC-learning algorithm *for* $\mathcal{C}$.

PAC learning

Definition 2.3 (PAC-learning) *A concept class* $\mathcal{C}$ *is said to be* PAC-learnable *if there exists an algorithm* $\mathcal{A}$ *and a polynomial function* $poly(\cdot,\cdot,\cdot,\cdot)$ *such that for any* $\epsilon > 0$ *and* $\delta > 0$*, for all distributions* $\mathcal{D}$ *on* $\mathcal{X}$ *and for any target concept* $c \in \mathcal{C}$*, the following holds for any sample size* $m \geq \boxed{poly(1/\epsilon, 1/\delta, n, size(c))}$:

$$\mathop{\mathbb{P}}_{S\sim\mathcal{D}^m}\left[\boxed{R(h_S) \leq \epsilon}\right] \geq \boxed{1-\delta.} \quad \text{Probably} \tag{2.4}$$

Approximately Correct

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n, size(c))$*, then* $\mathcal{C}$ *is said to be* efficiently PAC-learnable. *When such an algorithm* $\mathcal{A}$ *exists, it is called a* PAC-learning algorithm *for* $\mathcal{C}$.

PAC learning

Polynomial: $\boxed{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0 \,,}$

Definition 2.3 (PAC-learning) *A concept class* $\mathcal{C}$ *is said to be* PAC-learnable *if there exists an algorithm* $\mathcal{A}$ *and a polynomial function* $poly(\cdot,\cdot,\cdot,\cdot)$ *such that for any* $\epsilon > 0$ *and* $\delta > 0$*, for all distributions* $\mathcal{D}$ *on* $\mathcal{X}$ *and for any target concept* $c \in \mathcal{C}$*, the following holds for any sample size* $m \geq \boxed{poly(1/\epsilon, 1/\delta, n, size(c))}$*:*

$$\underset{S\sim\mathcal{D}^m}{\mathbb{P}}[R(h_S) \leq \epsilon] \geq 1 - \delta. \tag{2.4}$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n, size(c))$*, then* $\mathcal{C}$ *is said to be* efficiently PAC-learnable*. When such an algorithm* $\mathcal{A}$ *exists, it is called a* PAC-learning algorithm *for* $\mathcal{C}$*.*

PAC learning

Polynomial: $\boxed{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0 \,,}$

Definition 2.3 (PAC-learning) *A concept class* $\mathcal{C}$ *is said to be* PAC-learnable *if there exists an algorithm* $\mathcal{A}$ *and a polynomial function* $poly(\cdot,\cdot,\cdot,\cdot)$ *such that for any* $\epsilon > 0$ *and* $\delta > 0$*, for all distributions* $\mathcal{D}$ *on* $\mathcal{X}$ *and for any target concept* $c \in \mathcal{C}$*, the following holds for any sample size* $m \geq \boxed{poly(1/\epsilon, 1/\delta, n, size(c))}$:

$$\mathbb{P}_{S \sim \mathcal{D}^m}[R(h_S) \leq \epsilon] \geq 1 - \delta. \tag{2.4}$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n, size(c))$*, then* $\mathcal{C}$ *is said to be* efficiently PAC-learnable. *When such an algorithm* $\mathcal{A}$ *exists, it is called a* PAC-learning algorithm *for* $\mathcal{C}$.

$m \rightarrow S \rightarrow h_S$

Underlying concept function

An underlying concept function (mapping)

$$concept\ c: \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

A new Q: with almost the same features, can we guarantee the prediction (sweetness)?

Features

Color
Shape
Yellow spot
……

A model

Sweet?

Q: how to measure the model's performance?

Image from https://www.self.com/story/heres-how-to-pick-a-perfect-melon-every-time

Underlying concept function

An underlying concept function (mapping)

$$concept\ c : \mathcal{X} \rightarrow \mathcal{Y}$$

Some feature space

$\mathcal{Y} = \{0, 1\}$

or $\mathcal{Y} = \mathbb{R}$

Nearly the same color/shape/…
→
Different sweetness?

A new Q: with almost the same features, can we guarantee the prediction (sweetness)?

Features

Color
Shape
Yellow spot
……

A model

Sweet?

Q: how to measure the model's performance?

Image from https://www.self.com/story/heres-how-to-pick-a-perfect-melon-every-time

Agnostic PAC learning

Definition 2.14 (Agnostic PAC-learning) *Let* $\boxed{\mathcal{H}}$ *be a hypothesis set.* $\mathcal{A}$ *is an* agnostic PAC-learning *algorithm if there exists a polynomial function* $poly(\cdot,\cdot,\cdot,\cdot)$ *such that for any* $\boxed{\epsilon > 0 \text{ and } \delta > 0,}$ *for all* $\boxed{\text{distributions } \mathcal{D} \text{ over } \mathcal{X} \times \mathcal{Y},}$ *the following holds for any sample size* $\boxed{m} \geq poly(1/\epsilon, 1/\delta, n, size(c))$*:*

$$\mathop{\mathbb{P}}_{S \sim \mathcal{D}^m}\boxed{\left[R(h_S) - \min_{h \in \mathcal{H}} R(h) \leq \epsilon\right]} \geq \boxed{1 - \delta.} \qquad (2.21)$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n)$*, then it is said to be an* efficient agnostic PAC-learning algorithm.

Agnostic PAC learning

$$R(h) = \mathop{\mathbb{P}}_{x \sim \mathcal{D}}[h(x) \neq c(x)] = \mathop{\mathbb{E}}_{x \sim \mathcal{D}}\left[1_{h(x) \neq c(x)}\right],$$

c(x) is deterministic

$$R(h) = \mathop{\mathbb{P}}_{(x,y) \sim \mathcal{D}}[h(x) \neq y] = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}}[1_{h(x) \neq y}].$$

stochastic: joint distribution D

Definition 2.14 (Agnostic PAC-learning) *Let* $\boxed{\mathcal{H}}$ *be a hypothesis set.* $\mathcal{A}$ *is an* agnostic PAC-learning *algorithm if there exists a polynomial function* $poly(\cdot, \cdot, \cdot, \cdot)$ *such that for any* $\boxed{\epsilon > 0 \text{ and } \delta > 0}$, *for all* $\boxed{\text{distributions } \mathcal{D} \text{ over } \mathcal{X} \times \mathcal{Y}}$, *the following holds for any sample size* $\boxed{m} \geq poly(1/\epsilon, 1/\delta, n, size(c))$:

$$\mathop{\mathbb{P}}_{S \sim \mathcal{D}^m}\boxed{\left[R(h_S) - \min_{h \in \mathcal{H}} R(h) \leq \epsilon\right]} \geq \boxed{1 - \delta.} \tag{2.21}$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n)$, *then it is said to be an* efficient agnostic PAC-learning algorithm.

Agnostic PAC learning

$$R(h) = \mathop{\mathbb{P}}_{x \sim \mathcal{D}}[h(x) \neq \boxed{c(x)}] = \mathop{\mathbb{E}}_{x \sim \mathcal{D}}\left[1_{h(x) \neq c(x)}\right],$$

c(x) is deterministic

$$R(h) = \mathop{\mathbb{P}}_{(x,y) \sim \mathcal{D}}[h(x) \neq \boxed{y}] = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}}[1_{h(x) \neq y}].$$

stochastic: joint distribution D

Definition 2.14 (Agnostic PAC-learning) *Let* $\boxed{\mathcal{H}}$ *be a hypothesis set.* $\mathcal{A}$ *is an* agnostic PAC-learning *algorithm if there exists a polynomial function* $poly(\cdot, \cdot, \cdot, \cdot)$ *such that for any* $\boxed{\epsilon > 0 \text{ and } \delta > 0}$, *for all* $\boxed{\text{distributions } \mathcal{D} \text{ over } \mathcal{X} \times \mathcal{Y}}$, *the following holds for any sample size* $\boxed{m} \geq poly(1/\epsilon, 1/\delta, n, size(c))$:

$$\mathop{\mathbb{P}}_{S \sim \mathcal{D}^m}\boxed{\left[R(h_S) - \min_{h \in \mathcal{H}} R(h) \leq \epsilon\right]} \geq \boxed{1 - \delta}. \tag{2.21}$$

If $\mathcal{A}$ *further runs in* $poly(1/\epsilon, 1/\delta, n)$, *then it is said to be an* efficient agnostic PAC-learning algorithm.

Bayes error

Definition 2.15 (Bayes error) *Given a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, the* Bayes error R^* *is defined as the infimum of the errors achieved by measurable functions $h\colon \mathcal{X} \to \mathcal{Y}$:*

$$R^\star = \boxed{\inf_{\substack{h \\ h \text{ measurable}}}} R(h). \tag{2.22}$$

A hypothesis h with $R(h) = R^*$ is called a Bayes hypothesis *or* Bayes classifier.

All possible hypotheses
(may not be included in H)

Bayes error

Definition 2.15 (Bayes error) *Given a distribution* $\mathcal{D}$ *over* $\mathcal{X} \times \mathcal{Y}$*, the* Bayes error R^* *is defined as the infimum of the errors achieved by measurable functions* $h \colon \mathcal{X} \to \mathcal{Y}$*:*

The best risk we may reach

$$R^\star = \boxed{\inf_{\substack{h \\ h \text{ measurable}}}} R(h). \tag{2.22}$$

A hypothesis h *with* $R(h) = R^*$ *is called a* Bayes hypothesis *or* Bayes classifier.

All possible hypotheses
(may not be included in H)

Estimation and approximation

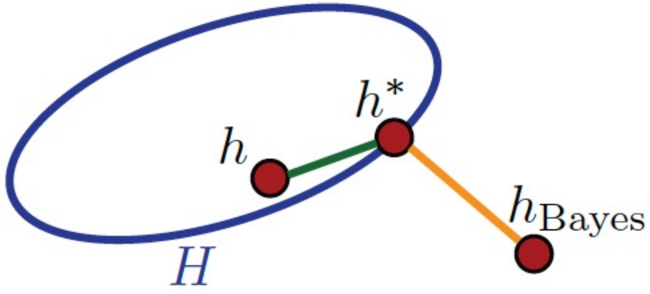
$$R(h) - R^*$$

Estimation and approximation

$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$

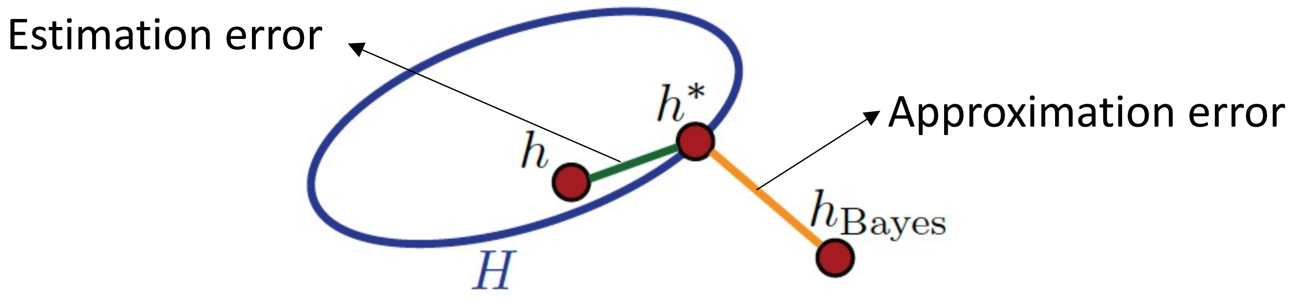
Estimation and approximation

$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$



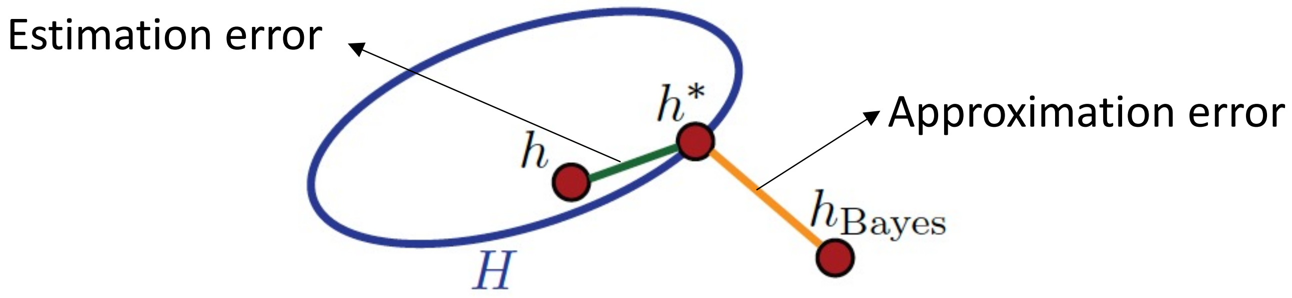
Estimation and approximation

$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$



Estimation and approximation

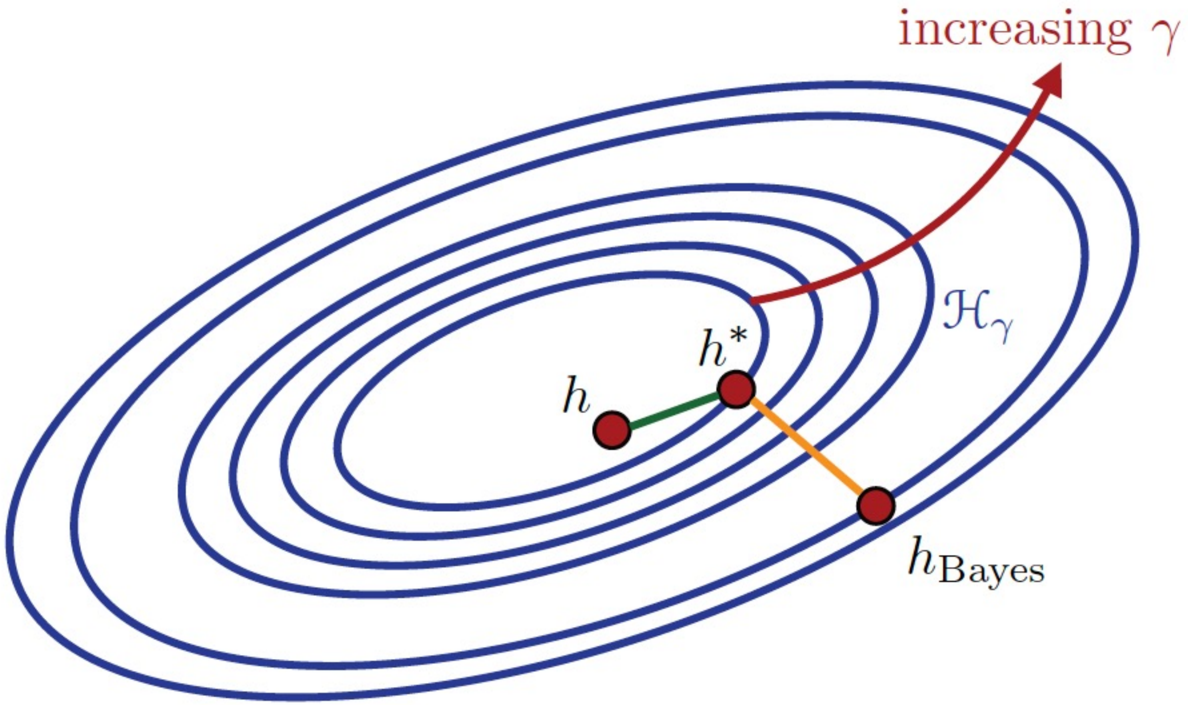
$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$



Q: can we enlarge H?

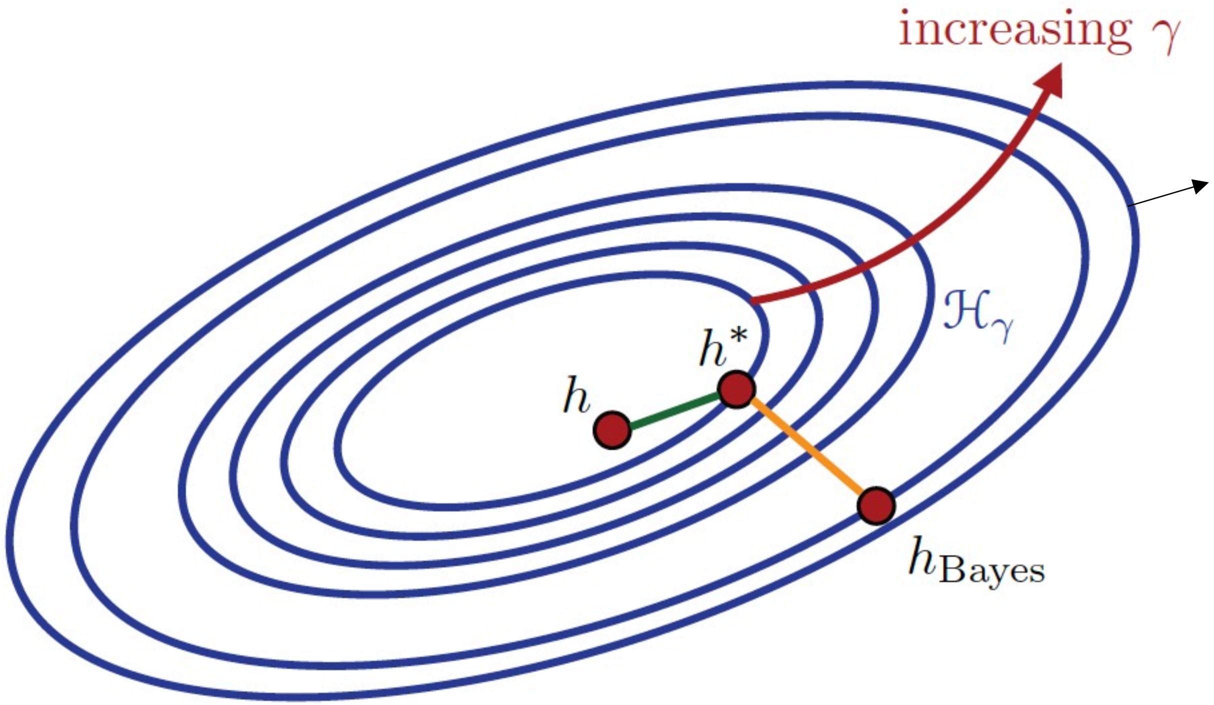
Estimation and approximation

$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$



Estimation and approximation

$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$



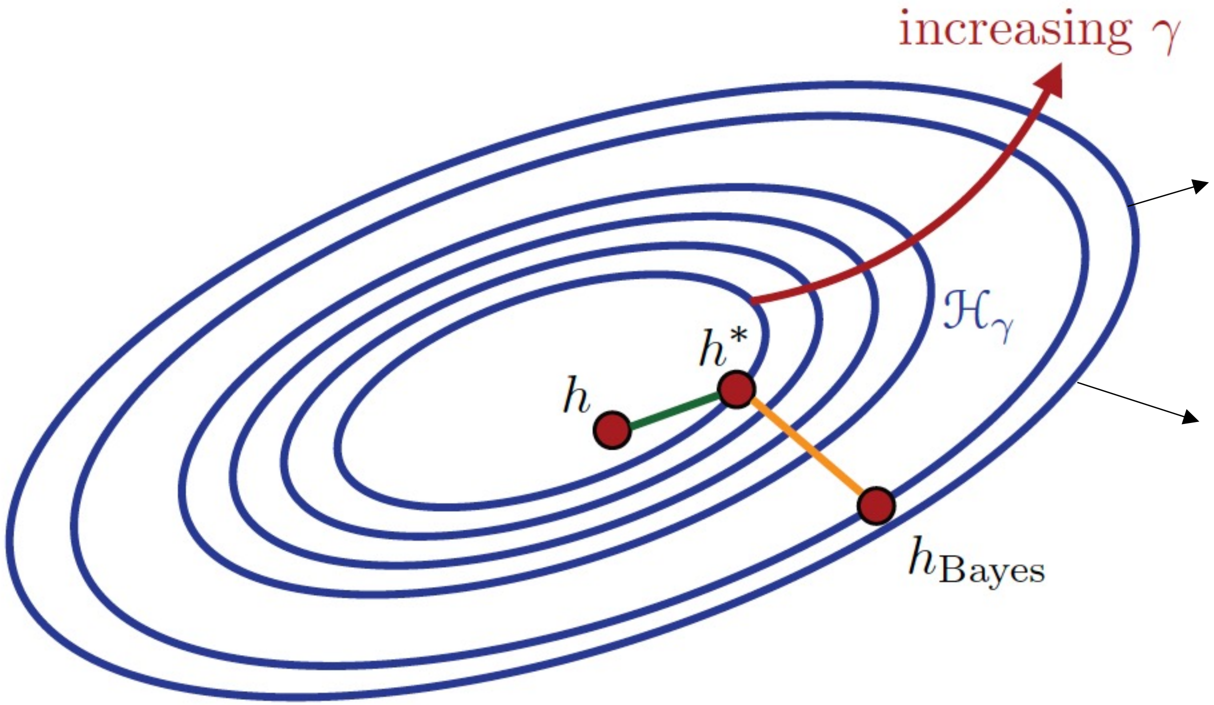
- When H is very rich:

  $h^* = h_{Bayes}$

  (Approximation error = 0)

Estimation and approximation

$$R(h) - R^* = \underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.$$



· When H is very rich:

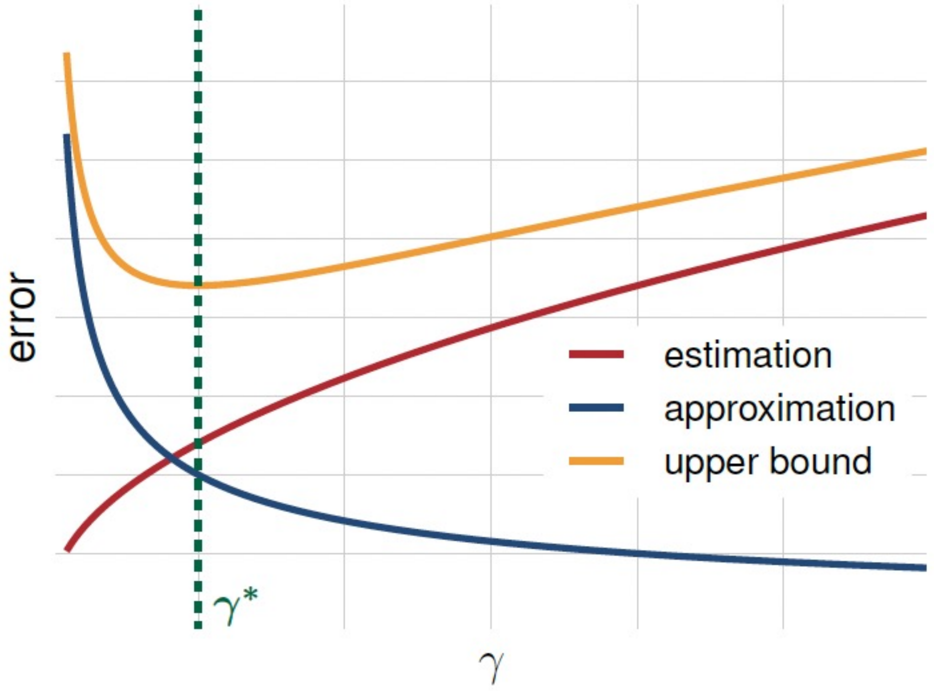
$$h^* = h_{Bayes}$$

(Approximation error = 0)

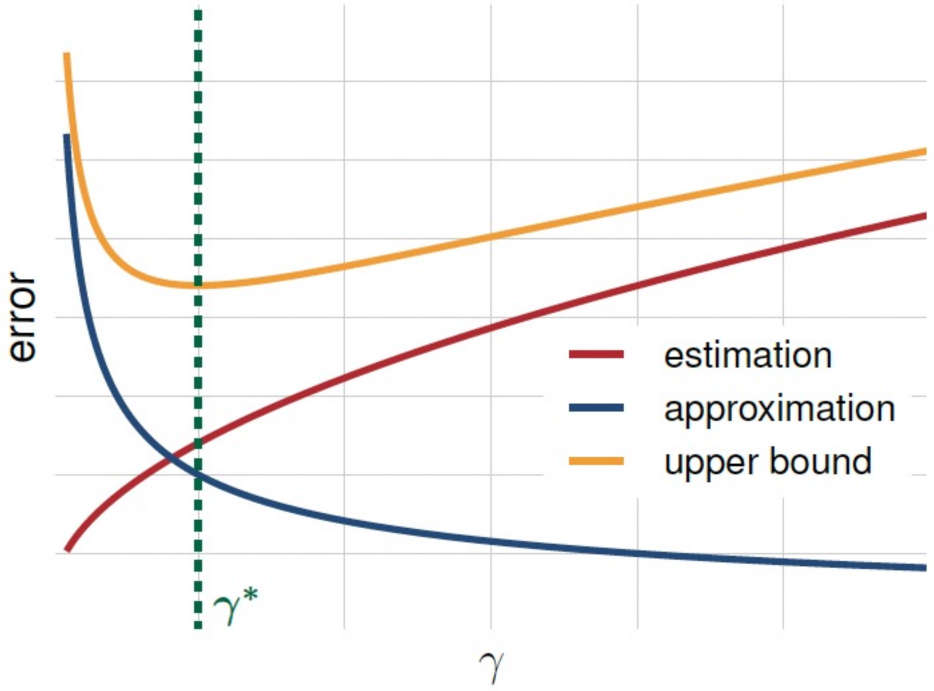
· H is so rich that it is very hard to make:

$$R(h) \to \inf_{h \in \mathcal{H}} R(h)$$

Trade-off: estimation and approximation



Trade-off: estimation and approximation

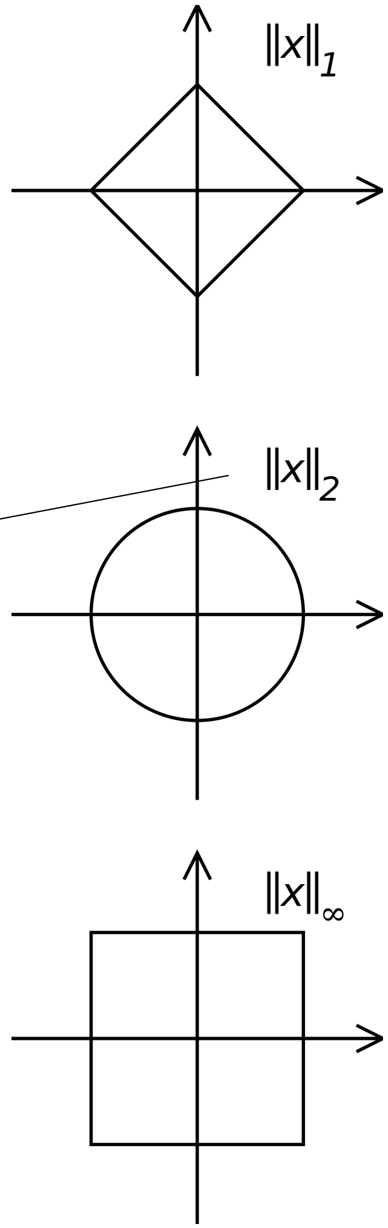


Q: how to control the richness of H?

Constrained problem

$$(CP): \quad \min_x f(x), \ \text{s.t.} \ \boxed{h(x) \le b},$$

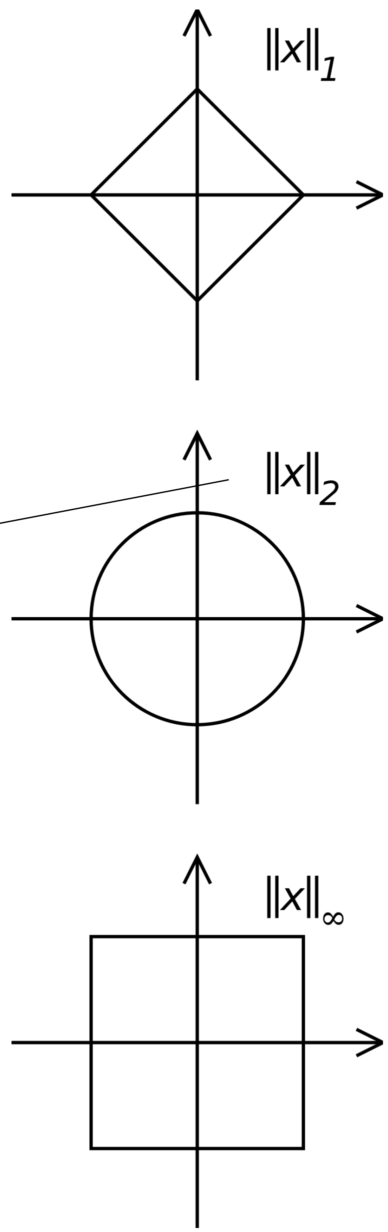
$$\boxed{\begin{aligned} h(x) &= \|x\|_2^2 \\ &= {x_1}^2 + {x_2}^2 + \cdots + {x_n}^2 \end{aligned}}$$



Regularized problem

$$(RP): \quad \min_x f(x) + \boxed{\lambda h(x),}$$

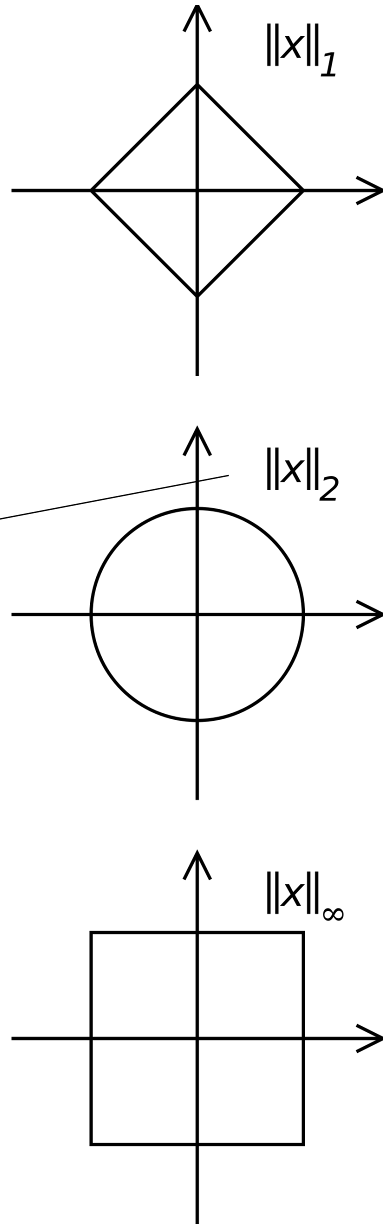
$$\boxed{\begin{aligned} h(x) &= \|x\|_2^2 \\ &= {x_1}^2 + {x_2}^2 + \cdots + {x_n}^2 \end{aligned}}$$



Regularized problem

$$(RP): \ \min_x f(x) + \boxed{\lambda h(x),}$$

$$\boxed{\begin{aligned} h(x) &= \|x\|_2^2 \\ &= {x_1}^2 + {x_2}^2 + \cdots + {x_n}^2 \end{aligned}}$$



Equivalence between (CP) and (RP):
$\lambda \leftarrow\rightarrow b$
We can find a b given λ such that:
Corresponding optimal solutions of (CP) and (RP) are identical

Empirical risk

Definition 2.2 (Empirical error) *Given a hypothesis* $h \in \mathcal{H}$, *a target concept* $c \in \mathcal{C}$, *and a sample* $\boxed{S = (x_1, \ldots, x_m)}$ *the* empirical error *or* empirical risk *of* h *is defined by*

Training set

$$\widehat{R}_S(h) = \boxed{\frac{1}{m} \sum_{i=1}^{m} 1_{h(x_i) \neq c(x_i)}.} \tag{2.2}$$

Interpret: average mistakes a hypothesis h makes on a sample

Empirical risk

Definition 2.2 (Empirical error) *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and a sample* $\boxed{S = (x_1, \ldots, x_m)}$ *the* empirical error *or* empirical risk *of h is defined by*

Training set

$$\widehat{R}_S(h) = \boxed{\frac{1}{m} \sum_{i=1}^{m} 1_{h(x_i) \neq c(x_i)}.} \tag{2.2}$$

Interpret: average mistakes a hypothesis h makes on a sample

stochastic version

$$\boxed{\widehat{R}_S(f) := \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{f(x) \neq y}.}$$

$$\boxed{R(h) = \mathop{\mathbb{P}}_{(x,y)\sim\mathcal{D}}[h(x) \neq y] = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[1_{h(x)\neq y}].}$$

risk (in population): not accessible

Empirical risk minimization

$$h_S^{\mathrm{ERM}} = \operatorname*{argmin}_{h \in \mathcal{H}} \boxed{\widehat{R}_S(h)}. \longrightarrow \text{Empirical risk}$$

Empirical risk minimization

$$\boxed{h_S^{\mathrm{ERM}}} = \operatorname*{argmin}_{h \in \mathcal{H}} \boxed{\widehat{R}_S(h)}.$$

Empirical risk

Plug in

$$R(h) - R^* = \boxed{\underbrace{\left(R(h) - \inf_{h \in \mathcal{H}} R(h)\right)}_{\text{estimation}}} + \boxed{\underbrace{\left(\inf_{h \in \mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}}.$$

Controlled by constraints/regularization

Q: How to control?

Empirical risk minimization

$$\boxed{h_S^{\mathrm{ERM}}} = \underset{h\in\mathcal{H}}{\operatorname{argmin}} \boxed{\widehat{R}_S(h)}.$$ → Empirical risk

Plug in

$$R(h) - R^* = \boxed{\underbrace{\left(R(h) - \inf_{h\in\mathcal{H}} R(h)\right)}_{\text{estimation}}} + \boxed{\underbrace{\left(\inf_{h\in\mathcal{H}} R(h) - R^*\right)}_{\text{approximation}}.}$$

→ Controlled by constraints/regularization

Q: How to control?

Proposition 4.1 *For any sample S, the following inequality holds for the hypothesis returned by ERM:*

$$\mathbb{P}\left[\boxed{R(h_S^{\mathrm{ERM}}) - \inf_{h\in\mathcal{H}} R(h)} > \epsilon\right] \le \mathbb{P}\left[\sup_{h\in\mathcal{H}} \boxed{|R(h) - \widehat{R}_S(h)|} > \frac{\epsilon}{2}\right]. \qquad (4.3)$$

Corollary 3.19 (VC-dimension generalization bounds) *Let $\mathcal{H}$ be a family of functions taking values in $\{-1,+1\}$ with VC-dimension d. Then, for any $\delta > 0$, with probability at least $1-\delta$, the following holds for all $h \in \mathcal{H}$:*

$$\boxed{R(h) \le \widehat{R}_S(h)} + \sqrt{\frac{2d\log\frac{em}{d}}{m}} + \sqrt{\frac{\log\frac{1}{\delta}}{2m}}. = \boxed{O(\sqrt{1/m})} \qquad (3.29)$$