Exercises

A selection of exercises from chapter 3 of Understanding Machine Learning: From Theory to Algorithms

Argyris Mouzakis

09/11/2018

Overview

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \rightarrow \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

- True Risk : $L_{\mathcal{D}}(h) = \underset{z \sim \mathcal{D}}{\mathbb{E}}[\ell(h, (x, y))]$

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \rightarrow \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

- True Risk : $L_{\mathcal{D}}(h) = \underset{z \sim \mathcal{D}}{\mathbb{E}}[\ell(h, (x, y))]$
- Empirical Risk (given $S = (z_1, z_2, \ldots, z_m)$) : $L_S(h) = \frac{\sum_{i=1}^{m} \ell(h, z_i)}{m}$

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \rightarrow \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

- True Risk : $L_{\mathcal{D}}(h) = \underset{z\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x,y)]$
- Empirical Risk (given $S = (z_1, z_2, \ldots, z_m)$) : $L_S(h) = \frac{\sum_{i=1}^{m} \ell(h,z_i)}{m}$

- Our use of that will be limited.

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

- True Risk : $L_{\mathcal{D}}(h) = \underset{z \sim \mathcal{D}}{\mathbb{E}}[\ell(h,(x,y))]$
- Empirical Risk (given $S = (z_1, z_2, \ldots, z_m)$) : $L_S(h) = \frac{\sum_{i=1}^{m} \ell(h, z_i)}{m}$

- Our use of that will be limited.

Simpler Setting

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

- True Risk : $L_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{z \sim \mathcal{D}}[\ell(h,(x,y))]$
- Empirical Risk (given $S = (z_1, z_2, \ldots, z_m)$) : $L_S(h) = \frac{\sum_{i=1}^{m} \ell(h,z_i)}{m}$

- Our use of that will be limited.

Simpler Setting

- Restrict Z to $\mathcal{X}$ or $\mathcal{X} \times \{0,1\}$ (is there is a labelling function f or not ?).

Risk Functions Reminder

General Setting

Given a loss function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and a distribution $\mathcal{D}$ over Z we defined :

- True Risk : $L_{\mathcal{D}}(h) = \underset{z \sim \mathcal{D}}{\mathbb{E}}[\ell(h,(x,y))]$
- Empirical Risk (given $S = (z_1, z_2, \dots, z_m)$) : $L_S(h) = \frac{\sum_{i=1}^{m} \ell(h,z_i)}{m}$

- Our use of that will be limited.

Simpler Setting

- Restrict Z to $\mathcal{X}$ or $\mathcal{X} \times \{0,1\}$ (is there is a labelling function f or not ?).
- Define ℓ as the $0-1$ loss.

PAC-learnability

Definition

A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property : For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f : \mathcal{X} \to 0, 1$, if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$ (over the choice of the examples), $L(\mathcal{D}, f)(h) \leq \epsilon$.

PAC-learnability

Definition

A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \rightarrow \mathbb{N}$ and a learning algorithm with the following property : For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f : \mathcal{X} \rightarrow 0, 1$, if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$ (over the choice of the examples), $L(\mathcal{D}, f)(h) \leq \epsilon$.

- Why bother with PAC-learning since we went such a long way to extend it ?

PAC-learnability

Definition

A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \rightarrow \mathbb{N}$ and a learning algorithm with the following property : For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f : \mathcal{X} \rightarrow 0, 1$, if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$ (over the choice of the examples), $L(\mathcal{D}, f)(h) \leq \epsilon$.

- Why bother with PAC-learning since we went such a long way to extend it ?
- Turns out PAC-learnable classes are also APAC-learnable (more on that next week).

Concentric Circles (Exercise 3.3)

Statement

Let $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0, 1\}$, and let $\mathcal{H}$ be the class of concentric circles in the plane, that is, $\mathcal{H} = \{h_r : r \in \mathbb{R}_+\}$, where $h_r(x) = \mathbb{1}_{[||x|| \leq r]}$. Prove that $\mathcal{H}$ is PAC-learnable (assume realizability), and its sample complexity is bounded by

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \frac{\ln\left(\frac{1}{\delta}\right)}{\epsilon}$$

Concentric Circles (Exercise 3.3)

Statement

Let $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0, 1\}$, and let $\mathcal{H}$ be the class of concentric circles in the plane, that is, $\mathcal{H} = \{h_r : r \in \mathbb{R}_+\}$, where $h_r(x) = \mathbb{1}_{[||x|| \leq r]}$. Prove that $\mathcal{H}$ is PAC-learnable (assume realizability), and its sample complexity is bounded by

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \frac{\ln\left(\frac{1}{\delta}\right)}{\epsilon}$$

- Realizability implies there is a circle inside which all points have label 1 while all outside points have label 0.

Concentric Circles (Exercise 3.3)

Statement

Let $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0, 1\}$, and let $\mathcal{H}$ be the class of concentric circles in the plane, that is, $\mathcal{H} = \{h_r : r \in \mathbb{R}_+\}$, where $h_r(x) = \mathbb{1}_{[||x|| \leq r]}$. Prove that $\mathcal{H}$ is PAC-learnable (assume realizability), and its sample complexity is bounded by

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \frac{\ln\left(\frac{1}{\delta}\right)}{\epsilon}$$

- Realizability implies there is a circle inside which all points have label 1 while all outside points have label 0.
- Suppose that circle has radius r^*.

Algorithm for the Concentric Circles problem

- Compute the smallest circle enclosing all positive examples.

Algorithm for the Concentric Circles problem

- Compute the smallest circle enclosing all positive examples.
- ERM rule is implemented (empirical risk is equal to 0).

Algorithm for the Concentric Circles problem

- Compute the smallest circle enclosing all positive examples.
- ERM rule is implemented (empirical risk is equal to 0).
- Why is this algorithm better than others implementing the ERM rule ?

Algorithm for the Concentric Circles problem

- Compute the smallest circle enclosing all positive examples.
- ERM rule is implemented (empirical risk is equal to 0).
- Why is this algorithm better than others implementing the ERM rule ?
- The error is one-sided !

Algorithm for the Concentric Circles problem

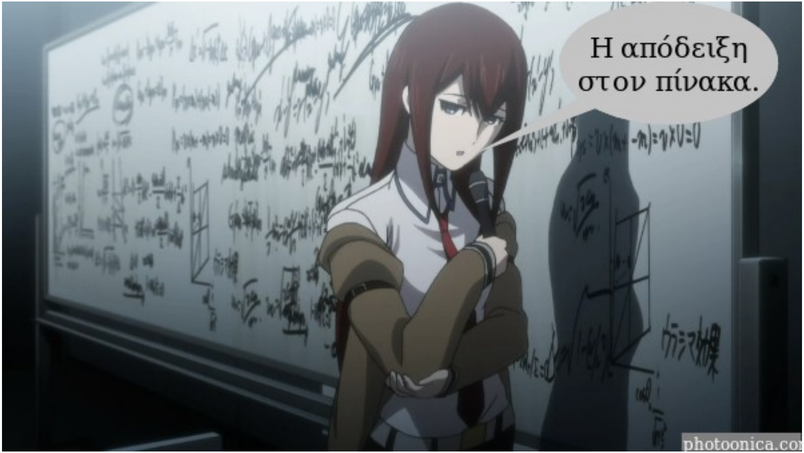
- Compute the smallest circle enclosing all positive examples.
- ERM rule is implemented (empirical risk is equal to 0).
- Why is this algorithm better than others implementing the ERM rule ?
- The error is one-sided !
- Runtime : $\mathcal{O}(m) = \mathcal{O}\left(\frac{\ln\left(\frac{1}{\delta}\right)}{\epsilon}\right)$

Sample Complexity

- Proof for the sample complexity ?

Sample Complexity

- Proof for the sample complexity ?



Independent but not identically distributed (Exercise 3.5)

Statement

Let $\mathcal{X}$ be a domain and let $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_m$ be a sequence of distributions over $\mathcal{X}$. Let $\mathcal{H}$ be a finite class of binary classifiers over $\mathcal{X}$ and let $f \in \mathcal{H}$. Suppose we are getting a sample S of m examples, such that the instances are independent but are not identically distributed ; the ith instance is sampled from $\mathcal{D}_i$ and then y_i is set to be $f(x_i)$. Let $\bar{\mathcal{D}_m}$ denote the average, that is,

$$\bar{\mathcal{D}_m} = \frac{\mathcal{D}_1 + \cdots + \mathcal{D}_m}{m}$$

Fix an accuracy parameter $\epsilon \in (0, 1)$. Show that

$$\mathbb{P}[\exists h \in \mathcal{H} \; s.t. \; L_{(\bar{\mathcal{D}_m}, f)}(h) > \epsilon \wedge L_S(h) = 0] \leq |\mathcal{H}| e^{-\epsilon m}$$

Independent but not identically distributed (Exercise 3.5)

Statement

Let $\mathcal{X}$ be a domain and let $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_m$ be a sequence of distributions over $\mathcal{X}$. Let $\mathcal{H}$ be a finite class of binary classifiers over $\mathcal{X}$ and let $f \in \mathcal{H}$. Suppose we are getting a sample S of m examples, such that the instances are independent but are not identically distributed ; the ith instance is sampled from $\mathcal{D}_i$ and then y_i is set to be $f(x_i)$. Let $\bar{\mathcal{D}_m}$ denote the average, that is,

$$\bar{\mathcal{D}_m} = \frac{\mathcal{D}_1 + \cdots + \mathcal{D}_m}{m}$$

Fix an accuracy parameter $\epsilon \in (0, 1)$. Show that

$$\mathbb{P}[\exists h \in \mathcal{H} \ s.t. \ L_{(\bar{\mathcal{D}_m}, f)}(h) > \epsilon \wedge L_S(h) = 0] \leq |\mathcal{H}| e^{-\epsilon m}$$

- Note that this example does not involve a learning algorithm.

Proof

Overview

Proof

Overview

- $L_{(\bar{\mathcal{D}}_m, f)}(h) = \frac{1}{m} \sum_{i \in [m]} L_{(\mathcal{D}_i, f)}(h)$

Proof

Overview

- $L_{(\bar{\mathcal{D}}_m,f)}(h) = \frac{1}{m}\sum_{i\in[m]} L_{(\mathcal{D}_i,f)}(h)$
- $\{\exists h \in \mathcal{H} : L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\} = \bigcup_{h\in\mathcal{H}}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\}$

Proof

Overview

- $L_{(\bar{\mathcal{D}}_m,f)}(h) = \frac{1}{m}\sum_{i\in[m]} L_{(\mathcal{D}_i,f)}(h)$
- $\{\exists h \in \mathcal{H} : L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\} = \bigcup_{h\in\mathcal{H}}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\}$
- Apply union bound based on the above.

Proof

Overview

- $L_{(\bar{\mathcal{D}}_m,f)}(h) = \frac{1}{m}\sum_{i\in[m]} L_{(\mathcal{D}_i,f)}(h)$
- $\{\exists h \in \mathcal{H} : L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\} = \bigcup_{h\in\mathcal{H}}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\}$
- Apply union bound based on the above.
- In the resulting sum, each element has the form :

$$\mathbb{P}[L_S(h) = 0]\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0]$$

where $\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0] = \mathbb{1}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon\}$.

Proof

Overview

- $L_{(\bar{\mathcal{D}_m},f)}(h) = \frac{1}{m}\sum_{i\in[m]} L_{(\mathcal{D}_i,f)}(h)$
- $\{\exists h \in \mathcal{H} : L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\} = \bigcup_{h\in\mathcal{H}}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\}$
- Apply union bound based on the above.
- In the resulting sum, each element has the form :

$$\mathbb{P}[L_S(h) = 0]\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0]$$

  where $\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0] = \mathbb{1}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon\}$.
- $\mathbb{P}[L_S(h) = 0] = \prod_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right)$

Proof

Overview

- $L_{(\bar{\mathcal{D}}_m,f)}(h) = \frac{1}{m}\sum_{i\in[m]} L_{(\mathcal{D}_i,f)}(h)$
- $\{\exists h \in \mathcal{H} : L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\} = \bigcup_{h\in\mathcal{H}}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\}$
- Apply union bound based on the above.
- In the resulting sum, each element has the form :

$$\mathbb{P}[L_S(h) = 0]\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0]$$

where $\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0] = \mathbb{1}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon\}$.

- $\mathbb{P}[L_S(h) = 0] = \prod_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right)$
- By AM-GM :

$$\prod_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right) \leq \left[\frac{1}{m}\sum_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right)\right]^m = [1 - L_{(\bar{\mathcal{D}}_m,f)}(h)]^m$$

Proof

Overview

- $L_{(\bar{\mathcal{D}}_m,f)}(h) = \frac{1}{m}\sum_{i\in[m]} L_{(\mathcal{D}_i,f)}(h)$
- $\{\exists h \in \mathcal{H} : L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\} = \bigcup_{h\in\mathcal{H}}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon \wedge L_S(h) = 0\}$
- Apply union bound based on the above.
- In the resulting sum, each element has the form :

$$\mathbb{P}[L_S(h) = 0]\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0]$$

  where $\mathbb{P}[L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon | L_S(h) = 0] = \mathbb{1}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon\}$.
- $\mathbb{P}[L_S(h) = 0] = \prod_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right)$
- By AM-GM :

$$\prod_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right) \leq \left[\frac{1}{m}\sum_{i=1}^{m}\left(1 - L_{(\mathcal{D}_i,f)}(h)\right)\right]^m = [1 - L_{(\bar{\mathcal{D}}_m,f)}(h)]^m$$

- We have the upper bound : $\sum_{h\in\mathcal{H}} \mathbb{1}\{L_{(\bar{\mathcal{D}}_m,f)}(h) > \epsilon\}[1 - L_{(\bar{\mathcal{D}}_m,f)}(h)]^m$

The Bayes Optimal Predictor (Exercise 3.7)

Statement

Show that for every probability distribution $\mathcal{D}$, the Bayes optimal predictor $f_{\mathcal{D}}$ is optimal, in the sense that for every classifier g from $\mathcal{X}$ to $\{0, 1\}$, $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

The Bayes Optimal Predictor (Exercise 3.7)

Statement

Show that for every probability distribution $\mathcal{D}$, the Bayes optimal predictor $f_{\mathcal{D}}$ is optimal, in the sense that for every classifier g from $\mathcal{X}$ to $\{0, 1\}$, $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

- Reminder :

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \text{if } \mathbb{P}[y = 1|x] \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

The Bayes Optimal Predictor (Exercise 3.7)

Statement

Show that for every probability distribution $\mathcal{D}$, the Bayes optimal predictor $f_{\mathcal{D}}$ is optimal, in the sense that for every classifier g from $\mathcal{X}$ to $\{0, 1\}$, $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

- Reminder :

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \textit{if } \mathbb{P}[y = 1|x] \geq \frac{1}{2} \\ 0 & \textit{otherwise} \end{cases}$$

- Intuitively, when we have to choose between classifying x in class 0 and class 1, we should choose the one with the higher posterior probability.

The Bayes Optimal Predictor (Exercise 3.7)

Statement

Show that for every probability distribution $\mathcal{D}$, the Bayes optimal predictor $f_{\mathcal{D}}$ is optimal, in the sense that for every classifier g from $\mathcal{X}$ to $\{0,1\}$, $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

- Reminder :

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \textit{if } \mathbb{P}[y=1|x] \geq \frac{1}{2} \\ 0 & \textit{otherwise} \end{cases}$$

- Intuitively, when we have to choose between classifying x in class 0 and class 1, we should choose the one with the higher posterior probability.
- Formal proof is not much harder than that.

The Bayes Optimal Predictor (Exercise 3.7)

Statement

Show that for every probability distribution $\mathcal{D}$, the Bayes optimal predictor $f_{\mathcal{D}}$ is optimal, in the sense that for every classifier g from $\mathcal{X}$ to $\{0, 1\}$, $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

- Reminder :

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \textit{if } \mathbb{P}[y = 1|x] \geq \frac{1}{2} \\ 0 & \textit{otherwise} \end{cases}$$

- Intuitively, when we have to choose between classifying x in class 0 and class 1, we should choose the one with the higher posterior probability.
- Formal proof is not much harder than that.
- Suppose $\mathcal{X}$ is discrete.

Bayes Optimality Proof

Overview

Bayes Optimality Proof

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y]$

Bayes Optimality Proof

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y]$
- We have : $\underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y] = \sum_{x^* \in \mathcal{X}} \mathbb{P}[x = x^*]\mathbb{P}[h(x^*) \neq y | x = x^*]$

Bayes Optimality Proof

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y]$
- We have : $\underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y] = \sum_{x^* \in \mathcal{X}} \mathbb{P}[x = x^*]\mathbb{P}[h(x^*) \neq y | x = x^*]$
- For each term of the sum, we have :

$$\mathbb{P}[h(x^*) \neq y | x = x^*] =$$

$$= \mathbb{P}[y = 0 | x = x^*]\mathbb{P}[h(x^*) \neq 0] + \mathbb{P}[y = 1 | x = x^*]\mathbb{P}[h(x^*) \neq 1]$$

Bayes Optimality Proof

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y]$
- We have : $\underset{(x,y)\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq y] = \sum_{x^* \in \mathcal{X}} \mathbb{P}[x = x^*]\mathbb{P}[h(x^*) \neq y | x = x^*]$
- For each term of the sum, we have :

$$\mathbb{P}[h(x^*) \neq y | x = x^*] =$$

$$= \mathbb{P}[y = 0 | x = x^*]\mathbb{P}[h(x^*) \neq 0] + \mathbb{P}[y = 1 | x = x^*]\mathbb{P}[h(x^*) \neq 1]$$

- Minimizing the above completes the proof.

Probabilistic Classifiers and the Bayes Optimal Predictor (Exercise 3.8a)

Statement

Probabilistic Classifiers and the Bayes Optimal Predictor (Exercise 3.8a)

Statement

- Probabilistic Predictor : $h : \mathcal{X} \to [0, 1]$ (instead of $\{0, 1\}$).

Probabilistic Classifiers and the Bayes Optimal Predictor (Exercise 3.8a)

Statement

- Probabilistic Predictor : $h : \mathcal{X} \rightarrow [0, 1]$ (instead of $\{0, 1\}$).
- Loss function : $\ell(h, (x, y)) = |h(x) - y|$

Probabilistic Classifiers and the Bayes Optimal Predictor (Exercise 3.8a)

Statement

- Probabilistic Predictor : $h : \mathcal{X} \to [0, 1]$ (instead of $\{0, 1\}$).
- Loss function : $\ell(h, (x, y)) = |h(x) - y|$
- The Bayes Optimal Predictor is optimal even in this setting.

Bayes Optimality Proof v2

Overview

Bayes Optimality Proof v2

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x,y))]$

Bayes Optimality Proof v2

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x,y))]$
- We have :

$$\underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x^*,y^*))] = \sum_{x^*\in\mathcal{X}}\sum_{y^*\in\mathcal{Y}} \mathbb{P}[x=x^*, y=y^*]\ell(h,(x^*,y^*)) =$$

$$= \sum_{x^*\in\mathcal{X}} \mathbb{P}[x=x^*] \sum_{y^*\in\mathcal{Y}} \mathbb{P}[y=y^*|x=x^*]\ell(h,(x^*,y^*))$$

Bayes Optimality Proof v2

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x,y))]$
- We have :

$$\underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x^*,y^*))] = \sum_{x^*\in\mathcal{X}}\sum_{y^*\in\mathcal{Y}} \mathbb{P}[x=x^*,y=y^*]\ell(h,(x^*,y^*)) =$$

$$= \sum_{x^*\in\mathcal{X}} \mathbb{P}[x=x^*] \sum_{y^*\in\mathcal{Y}} \mathbb{P}[y=y^*|x=x^*]\ell(h,(x^*,y^*))$$

- Minimize :

$$\sum_{y^*\in\mathcal{Y}} \mathbb{P}[y=y^*|x=x^*]\ell(h,(x,y)) = \mathbb{P}[y=0|x=x^*]|h(x^*)-0|+$$

$$+\mathbb{P}[y=1|x=x^*]|h(x^*)-1| = \mathbb{P}[y=0|x=x^*]h(x^*)+\mathbb{P}[y=1|x=x^*](1-h(x^*))$$

Bayes Optimality Proof v2

Overview

- Minimize : $L_{\mathcal{D}}(h) = \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x,y))]$
- We have :

$$\underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}[\ell(h,(x^*,y^*))] = \sum_{x^*\in\mathcal{X}} \sum_{y^*\in\mathcal{Y}} \mathbb{P}[x = x^*, y = y^*]\ell(h,(x^*,y^*)) =$$

$$= \sum_{x^*\in\mathcal{X}} \mathbb{P}[x = x^*] \sum_{y^*\in\mathcal{Y}} \mathbb{P}[y = y^*|x = x^*]\ell(h,(x^*,y^*))$$

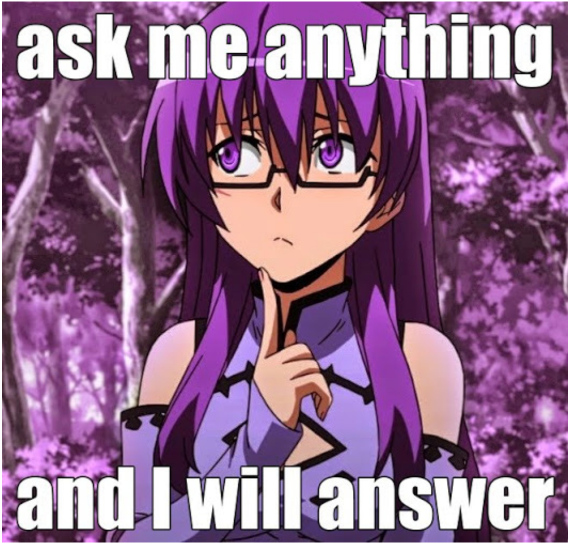
- Minimize :

$$\sum_{y^*\in\mathcal{Y}} \mathbb{P}[y = y^*|x = x^*]\ell(h,(x,y)) = \mathbb{P}[y = 0|x = x^*]|h(x^*) - 0| +$$

$$+\mathbb{P}[y = 1|x = x^*]|h(x^*) - 1| = \mathbb{P}[y = 0|x = x^*]h(x^*) + \mathbb{P}[y = 1|x = x^*](1 - h(x^*))$$

- This leads again to the Bayes Optimal Predictor.

Discussion



The End

Thank You !