Perceptrons and the Fundmental Theorem of Statistical Learning

Charlotte Aten

University of Denver

2022 October 24

Introduction

- Rosenblatt introduced the perceptron algorithm for binary classification in 1958.
- For those familiar with neural nets, this is basically a single neuron whose activation/transfer/threshold function is just the identity.

Perceptron algorithm

- We are given a binary classification task for points in $\mathbb{R}^d$.
- Our hypothesis class $\mathcal{H}$ is the collection of all functions $h\colon \mathbb{R}^d \to \{-1, 1\}$ of the form

$$h(x) = \operatorname{sgn}(w \cdot x + b)$$

  for some $w \in \mathbb{R}^d$ and some $b \in \mathbb{R}$.
- We have that $\operatorname{VCdim}(\mathcal{H}) = d + 1$ in this case.

Perceptron algorithm

- The perceptron algorithm takes a training set

$$\{(x_1, y_1), \ldots, (x_m, y_m)\}$$

  as input.
- We choose an initial vector $w \in \mathbb{R}^d$, say $w^{(1)} = (0, \ldots, 0)$ and an initial constant $b \in \mathbb{R}$, say $b^{(1)} = 0$.
- At each iteration of the algorithm, we check whether there is some i for which

$$y_i(w^{(t)} \cdot x_i + b^{(t)}) \leq 0.$$

Perceptron algorithm

- For some such i we output

$$w^{(t+1)} = w^{(t)} + y_i x_i$$

and

$$b^{(t+1)} = b^{(t)} + y_i.$$

- It is possible to prove that the algorithm terminates after a certain number of steps, but we won't discuss that there.

Applying the fundamental theorem

Theorem (The Fundamental Theorem of Statistical Learning (Quantitative Version))

Let $\mathcal{H}$ be a hypothesis class of functions from a domain X to $\{0, 1\}$ and let the loss function be the 01 loss. Assume that $VCdim(H) = d < \infty$. Then, there are absolute constants C_1, C_2 such that:

1. *$\mathcal{H}$ has the uniform convergence property with sample complexity*

$$C_1 \frac{d + \log(1/\delta)}{\epsilon^2} \leq m_{\mathcal{H}}^{UC}(\epsilon, \delta) \leq C_2 \frac{d + \log(1/\delta)}{\epsilon^2}$$

2. *$\mathcal{H}$ is agnostic PAC learnable with sample complexity*

$$C_1 \frac{d + \log(1/\delta)}{\epsilon^2} \leq m_{\mathcal{H}}(\epsilon, \delta) \leq C_2 \frac{d + \log(1/\delta)}{\epsilon^2}$$

3. *$\mathcal{H}$ is PAC learnable with sample complexity*

$$C_1 \frac{d + \log(1/\delta)}{\epsilon} \leq m_{\mathcal{H}}(\epsilon, \delta) \leq C_2 \frac{d \log(1/\epsilon) + \log(1/\delta)}{\epsilon}$$

Applying the fundamental theorem

- In the agnostic case, an explicit upper bound for the sample complexity is

$$m_{\mathcal{H}}(\epsilon, \delta) \leq 4\frac{32d}{\epsilon^2} \log\left(\frac{64d}{\epsilon^2}\right) + \frac{8}{\epsilon^2}\left(8d \log\left(\frac{e}{d}\right) + 2\log\left(\frac{4}{\delta}\right)\right)$$

and an explicit lower bound for the sample complexity is

$$m_{\mathcal{H}}(\epsilon, \delta) \geq \frac{8d}{\epsilon^2}$$

assuming that $\delta < \frac{1}{8}$.

Applying the fundamental theorem

- Taking $\epsilon = 0.1$ and $\delta = \frac{1}{8}$ and replacing d with $d+1$ we obtain

$$m_{\mathcal{H}}(\epsilon, \delta) \leq 4\frac{32(d+1)}{0.01}\log\left(\frac{64(d+1)}{0.01}\right) + \frac{8}{0.01}\left(8(d+1)\log\left(\frac{e}{(d+1)}\right) + 2\log(2)\right)$$

and an explicit lower bound for the sample complexity is

$$m_{\mathcal{H}}(\epsilon, \delta) \geq \frac{8(d+1)}{0.01}$$

assuming that $\delta < \frac{1}{8}$.

References

- Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. 32 Avenue of the Americas, New York, NY 10013-2473, USA: Cambridge University Press, 2014. ISBN: 978-1-107-05713-5