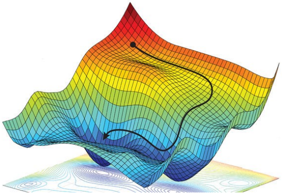
Introduction to Machine Learning

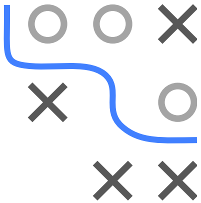
Advanced Risk Minimization
Risk Minimizers



Learning goals

- Bayes optimal model (also: risk minimizer, population minimizer)
- Bayes risk
- Bayes regret, estimation and approximation error
- Optimal constant model
- Consistent learners

EMPIRICAL RISK MINIMIZATION



Very often, in ML, we minimize the empirical risk

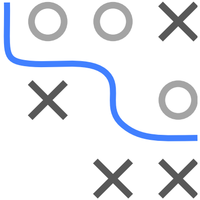
$$\mathcal{R}_{\text{emp}}(f) = \sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right)$$

- each observation $\left(\mathbf{x}^{(i)}, y^{(i)}\right) \in \mathcal{X} \times \mathcal{Y}$, so from feature and target space
- $f_{\mathcal{H}} : \mathcal{X} \to \mathbb{R}^g$, f is a model from hypothesis space $\mathcal{H}$; maps a feature vector to output score; sometimes or often we omit $\mathcal{H}$ in the index
- $L : (\mathcal{Y} \times \mathbb{R}^g) \to \mathbb{R}$ is loss;
  $L(y, f)$ measures distance between label and prediction
- We assume that $(\mathbf{x}, y) \sim \mathbb{P}_{xy}$ and $\left(\mathbf{x}^{(i)}, y^{(i)}\right) \overset{\text{i.i.d.}}{\sim} \mathbb{P}_{xy}$
  $\mathbb{P}_{xy}$ is the distribution of the data generating process (DGP)

Let's define (and minimize) loss in expectation, the theoretical risk:

$$\mathcal{R}(f) := \mathbb{E}_{xy}[L(y, f(\mathbf{x}))] = \int L(y, f(\mathbf{x}))\, d\mathbb{P}_{xy}$$

TWO SHORT EXAMPLES



Regression with linear model:

- Model: $f(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x} + \theta_0$
- Squared loss: $L(y, f) = (y - f)^2$
- Hypothesis space:

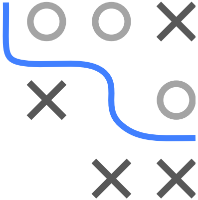
$$\mathcal{H}_{\text{lin}} = \left\{ \mathbf{x} \mapsto \boldsymbol{\theta}^\top \mathbf{x} + \theta_0 : \boldsymbol{\theta} \in \mathbb{R}^d, \theta_0 \in \mathbb{R} \right\}$$

Binary classification with shallow MLP:

- Model: $f(\mathbf{x}) = \pi(\mathbf{x}) = \sigma(\boldsymbol{w}_2^\top \text{ReLU}(\boldsymbol{W}_1 \mathbf{x} + \boldsymbol{b}_1) + b_2)$
- Binary cross-entropy loss:
  $L(y, \boldsymbol{\pi}) = -(y \log(\pi) + (1 - y) \log(1 - \pi))$
- Hypothesis space:

$$\mathcal{H}_{\text{MLP}} = \{ \mathbf{x} \mapsto \sigma(\boldsymbol{w}_2^\top \text{ReLU}(\boldsymbol{W}_1 \mathbf{x} + \boldsymbol{b}_1) + b_2) : \mathbf{W}_1 \in \mathbb{R}^{h \times d}, \mathbf{b}_1 \in \mathbb{R}^h, \mathbf{w}_2 \in \mathbb{R}^h, b_2 \in \mathbb{R} \}$$

OPTIMAL CONSTANTS FOR A LOSS



- Let's assume some RV $z \in \mathcal{Y}$ for a label
- z not RV y, because we want to fiddle with its distribution
- Assume z has distribution Q, so $z \sim Q$
- We can now consider $\arg\min_c \mathbb{E}_{z \sim Q}[L(z, c)]$
  so the score-constant which loss-minimally approximates z

We will consider 3 cases for Q

- $Q = P_y$, simply our labels and their marginal distribution in $\mathbb{P}_{xy}$
- $Q = P_{y|x=x}$, conditional label distribution at point $x = \tilde{x}$
- $Q = P_n$, the empirical product distribution for data $y_1, \dots, y_n$

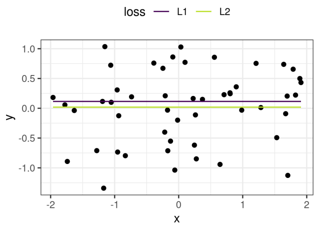
If we can solve $\arg\min_c \mathbb{E}_{z \sim Q}[L(z, c)]$ for any Q, we will get multiple useful results!

OPTIMAL CONSTANT MODEL

- We would like a loss optimal, constant baseline predictor
- A "featureless" ML model, which always predicts the same value
- Can use it as baseline in experiments, if we don't beat this with more complex model, that model is useless
- Will also be useful as component in algorithms and derivations

$$f_c^* = \underset{c \in \mathbb{R}}{\arg\min}\, \mathbb{E}_{xy}\left[L(y, c)\right] = \underset{c \in \mathbb{R}}{\arg\min}\, \mathbb{E}_{y}\left[L(y, c)\right]$$

and $f(\mathbf{x}) = \theta = c$ that optimizes the empirical risk $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ is denoted as as $\hat{f}_c = \arg\min_{c \in \mathbb{R}} \mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$.



OPTIMAL CONSTANT MODEL

- Let's start with the simplest case, L2 loss
- And we want to find and optimal constant model for

$$\arg\min \mathbb{E}[L(z, c)] =$$

$$\arg\min \mathbb{E}[(z - c)^2] =$$

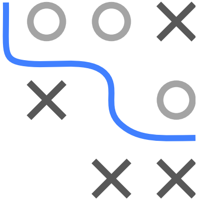
$$\arg\min \mathbb{E}[z^2] - 2cE[z] + c^2 =$$

$$E[z]$$

- Using $Q = P_y$, this means that, given we know the label distribution, the best constant is $c = E[y]$.
- If we only have data $y_1, \dots y_n$

  $\arg\min \mathbb{E}_{z \sim P_n}[(z - c)^2] = \mathbb{E}_{z \sim P_n}[z] = \frac{1}{n} \sum_{i=1}^{n} y^{(i)} = \bar{y}$
- And we want to find and optimal constant model for

RISK MINIMIZER

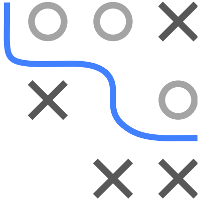


Let us assume we are in an "ideal world":

- The hypothesis space $\mathcal{H} = \mathcal{H}_{all}$ is unrestricted. We can choose any measurable $f : \mathcal{X} \to \mathbb{R}^g$.
- We also assume an ideal optimizer; the risk minimization can always be solved perfectly and efficiently.
- We know $\mathbb{P}_{xy}$.

How should f be chosen?

RISK MINIMIZER / 2



The f with minimal risk across all (measurable) functions is called the **risk minimizer**, **population minimizer** or **Bayes optimal model**.

$$
\begin{aligned}
f^*_{\mathcal{H}_{all}} &= \operatorname*{arg\,min}_{f \in \mathcal{H}_{all}} \mathcal{R}\left(f\right) = \operatorname*{arg\,min}_{f \in \mathcal{H}_{all}} \mathbb{E}_{xy}\left[L\left(y, f(\mathbf{x})\right)\right] \\
&= \operatorname*{arg\,min}_{f \in \mathcal{H}_{all}} \int L\left(y, f(\mathbf{x})\right) \mathrm{d}\mathbb{P}_{xy}.
\end{aligned}
$$

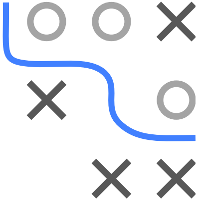
The resulting risk is called **Bayes risk**: $\mathcal{R}^* = \mathcal{R}(f^*_{\mathcal{H}_{all}})$

Note that if we leave out the hypothesis space in the subscript it becomes clear from the context!

Similarly, we define the risk minimizer over some $\mathcal{H} \subset \mathcal{H}_{all}$ as

$$
f^*_{\mathcal{H}} = \operatorname*{arg\,min}_{f \in \mathcal{H}} \mathcal{R}\left(f\right)
$$

OPTIMAL POINT-WISE PREDICTIONS

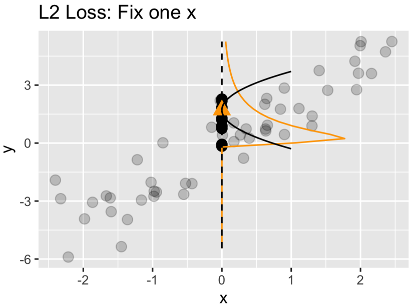


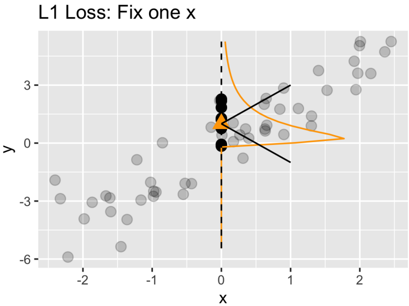
To derive the risk minimizer, observe that by law of total expectation

$$\mathcal{R}(f) = \mathbb{E}_{xy}\left[L\left(y, f(\mathbf{x})\right)\right] = \mathbb{E}_x\left[\mathbb{E}_{y|x}\left[L\left(y, f(\mathbf{x})\right) \mid \mathbf{x}\right]\right].$$

- We can choose $f(\mathbf{x})$ as we want (unrestricted hypothesis space, no assumed functional form)
- Hence, for a fixed value $\mathbf{x} \in \mathcal{X}$ we can select **any** value c we want to predict. So we construct the **point-wise optimizer**

$$f^*(\tilde{\mathbf{x}}) = \text{argmin}_c \mathbb{E}_{y|x}\left[L(y, c) \mid \mathbf{x} = \tilde{\mathbf{x}}\right]$$





THEORETICAL AND EMPIRICAL RISK

The risk minimizer is mainly a theoretical tool:

- In practice we need to restrict the hypothesis space $\mathcal{H}$ such that we can efficiently search over it.
- In practice we (usually) do not know $\mathbb{P}_{xy}$. Instead of $\mathcal{R}(f)$, we are optimizing the empirical risk

$$\hat{f}_{\mathcal{H}} = \underset{f \in \mathcal{H}}{\arg\min}\, \mathcal{R}_{\text{emp}}(f) = \underset{f \in \mathcal{H}}{\arg\min} \sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right)$$

Note that according to the **law of large numbers** (LLN), the empirical risk converges to the true risk (but beware of overfitting!):

$$\bar{\mathcal{R}}_{\text{emp}}(f) = \frac{1}{n} \sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right) \overset{n \to \infty}{\longrightarrow} \mathcal{R}(f).$$

ESTIMATION AND APPROXIMATION ERROR

Goal of learning: Train a model $\hat{f}_{\mathcal{H}}$ for which the true risk $\mathcal{R}\left(\hat{f}_{\mathcal{H}}\right)$ is close to the Bayes risk $\mathcal{R}^*$. In other words, we want the **Bayes regret** or **excess risk**

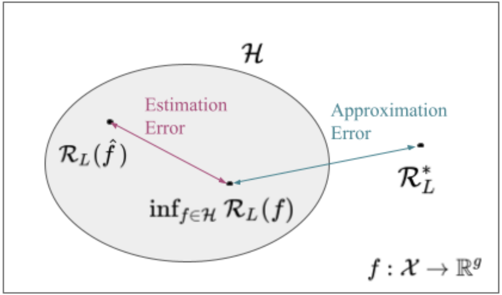
$$\mathcal{R}\left(\hat{f}_{\mathcal{H}}\right) - \mathcal{R}^*$$

to be as low as possible.

The Bayes regret can be decomposed as follows:

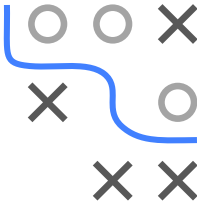
$$\begin{aligned}
\mathcal{R}\left(\hat{f}_{\mathcal{H}}\right) - \mathcal{R}^* &= \underbrace{\left[\mathcal{R}\left(\hat{f}_{\mathcal{H}}\right) - \inf_{f \in \mathcal{H}} \mathcal{R}(f)\right]}_{\text{estimation error}} + \underbrace{\left[\inf_{f \in \mathcal{H}} \mathcal{R}(f) - \mathcal{R}^*\right]}_{\text{approximation error}} \\
&= \left[\mathcal{R}(\hat{f}_{\mathcal{H}}) - \mathcal{R}(f^*_{\mathcal{H}})\right] + \left[\mathcal{R}(f^*_{\mathcal{H}}) - \mathcal{R}(f^*_{\mathcal{H}_{all}})\right]
\end{aligned}$$

ESTIMATION AND APPROXIMATION ERROR / 2



- $\mathcal{R}\left(\hat{f}\right) - \inf_{f \in \mathcal{H}} \mathcal{R}(f)$ is the **estimation error**. We fit $\hat{f}$ via empirical risk minimization and (usually) use approximate optimization, so we usually do not find the optimal $f \in \mathcal{H}$.
- $\inf_{f \in \mathcal{H}} \mathcal{R}(f) - \mathcal{R}^*$ is the **approximation error**. We need to restrict to a hypothesis space $\mathcal{H}$ which might not even contain the Bayes optimal model f^*.

(UNIVERSALLY) CONSISTENT LEARNERS



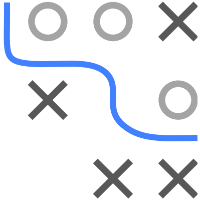
Consistency is an asymptotic property of a learning algorithm, which ensures the algorithm returns **the correct model** when given **unlimited data**.

Let $\mathcal{I} : \mathbb{D} \rightarrow \mathcal{H}$ be a learning algorithm that takes a training set $\mathcal{D}_{\text{train}} \sim \mathbb{P}_{xy}$ of size n_{train} and estimates a model $\hat{f} : \mathcal{X} \rightarrow \mathbb{R}^g$.

The learning method $\mathcal{I}$ is said to be **consistent** w.r.t. a certain distribution $\mathbb{P}_{xy}$ if the risk of the estimated model $\hat{f}$ converges in probability ("$\overset{p}{\longrightarrow}$") to the Bayes risk $\mathcal{R}^*$ when n_{train} goes to ∞:

$$\mathcal{R}\left(\mathcal{I}\left(\mathcal{D}_{\text{train}}\right)\right) \overset{p}{\longrightarrow} \mathcal{R}^* \quad \text{for } n_{\text{train}} \rightarrow \infty.$$

(UNIVERSALLY) CONSISTENT LEARNERS / 2



Consistency is defined w.r.t. a particular distribution $\mathbb{P}_{xy}$. But since we usually do not know $\mathbb{P}_{xy}$, consistency does not offer much help to choose an algorithm for a particular task.

More interesting is the stronger concept of **universal consistency**: An algorithm is universally consistent if it is consistent for **any** distribution.

In Stone's famous consistency theorem from 1977, the universal consistency of a weighted average estimator as KNN was proven. Many other ML models have since then been proven to be universally consistent (SVMs, ANNs, etc.).

Note that universal consistency is obviously a desirable property - however, (universal) consistency does not tell us anything about convergence rates ...