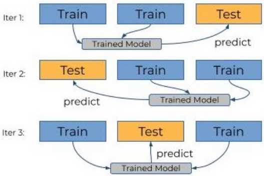
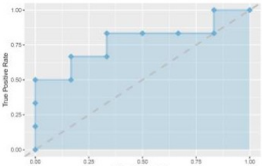
Introduction to Machine Learning

Evaluation: Introduction and Remarks





Learning goals

- Understand the goal of performance estimation
- Know the definition of generalization error
- Understand the difference between outer and inner loss

PERFORMANCE ESTIMATION

- After training our model, we are naturally interested in its **performance**.
- We recall what supervised learning is about:

$$\mathcal{I} : \mathbb{D} \times \boldsymbol{\Lambda} \to \mathcal{H}, \quad (\mathcal{D}, \boldsymbol{\lambda}) \mapsto \hat{f}_{\mathcal{D}, \boldsymbol{\lambda}}$$

- $\mathcal{I}$ minimizes the empirical risk resulting from $L(y, f)$.
- This so-called **inner loss**, however, is only a statistical proxy to what we are really interested in: the **true expected loss** for new, unlabeled data.
- After all, we chose our model precisely so it would be loss-minimal on the data we trained it on, but we cannot hope for it to perform equally well on general data from $\mathbb{P}_{xy}$.

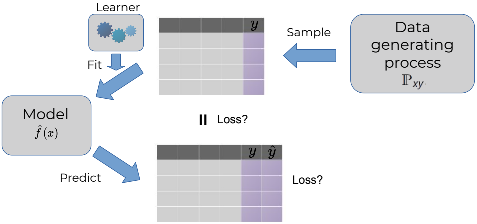
$\rightarrow$ Evaluation based on the inner loss would be **optimistically biased**.

GENERALIZATION ERROR

- The true expected loss for a model $\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}}$, learned on $\mathcal{D}_n \sim \mathbb{P}_{xy}$, is measured w.r.t. to previously unseen data $(\mathbf{x}, y) \sim \mathbb{P}_{xy}$.
- We refer to this as **generalization error** or **outer loss**:

$$\mathrm{GE}(\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}}) := \mathbb{E}_{(\mathbf{x}, y) \sim \mathbb{P}_{xy}} \left[L \left(y, \hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}}(\mathbf{x}) \right) \right]$$

- The goal of **performance evaluation** is to measure $\mathrm{GE}(\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}})$.
$\rightarrow$ As $\mathbb{P}_{xy}$ is unknown to us, we can only estimate it.



INNER VS OUTER LOSS

- Supervised learning thus implies the following dichotomy:
  - **Learning**: train $\hat{f}_{\mathcal{D}_n,\boldsymbol{\lambda}}$ minimizing *inner* loss
  - **Evaluation**: evaluate $\hat{f}_{\mathcal{D}_n,\boldsymbol{\lambda}}$ estimating *outer* loss
- Beyond evaluating a single learner, the outer loss lends itself to comparing different types of learners, or learners with varying hyperparameter configurations $\boldsymbol{\lambda}$.

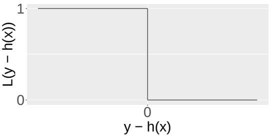
- Ideally, we have **inner loss = outer loss**.
- This is not always possible – sometimes we use losses that are hard to optimize or do not even specify one directly, as in:
  - Logistic regression: minimization of binomial loss
  - k-NN: no explicit loss minimization
- On the other hand, there are some special metrics for evaluation, such as those derived from ROC curves.

INNER VS OUTER LOSS

Example: logistic regression

- An intuitive choice for the loss in logistic regression would be to evaluate the share of incorrect predictions.
- This leads to the **mislassification error rate (MCE)**, computing the mean over pointwise **0-1 loss**.
- 0-1 loss simply assigns a loss of 1 for incorrect predictions and 0 otherwise:

$$L(y, h(\mathbf{x})) = [y - h(\mathbf{x})]$$



- Problem: 0-1 loss is not differentiable (not continuous even).
  $\rightarrow$ This is why we use binomial loss as **inner** loss instead.
- For evaluation, differentiability is not required.
  $\rightarrow$ We may use 0-1 as **outer** loss and evaluate our learner on its MCE.

TRAINING AND TEST DATA

- For reliable estimates of $\mathrm{GE}(\hat{f}_{\mathcal{D}_n,\boldsymbol{\lambda}})$ we need **test data** that are independent of the data we trained our model on.
- Such test sets are not always available, but we will learn about techniques of **resampling** that allow us to carve out test sets from the data at hand.

- Note that this paradigm is somewhat different from traditional statistical model diagnosis where models are judged by their goodness-of-fit rather than their generalization ability.