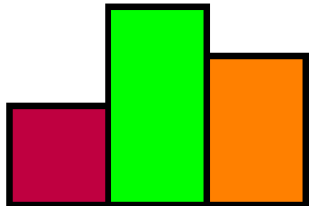


# Introduction to Machine Learning

## Evaluation: Generalization Error



### Learning goals

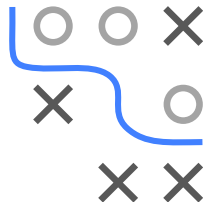
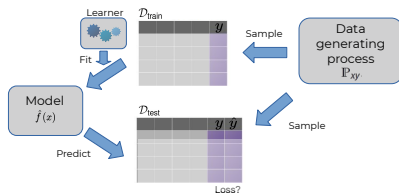
- Understand the goal of performance estimation
- Know the formal definition of generalization error as a statistical estimator of future performance
- Understand the difference between GE for a model and GE for a learner.
- Understand the difference between outer and inner loss



# GE FOR A FIXED MODEL

- GE for a fixed model:  $\text{GE}(\hat{f}, L) := \mathbb{E} \left[ L(y, \hat{f}(\mathbf{x})) \right]$   
Expectation over a single, random test point  $(\mathbf{x}, y) \sim \mathbb{P}_{xy}$ .
- Estimator, **if a dedicated test set is available** (size  $m$ )

$$\widehat{\text{GE}}(\hat{f}, L) := \frac{1}{m} \sum_{(\mathbf{x}, y) \in \mathcal{D}_{\text{test}}} \left[ L(y, \hat{f}(\mathbf{x})) \right]$$



NB: Very often, no dedicated test-set is available, and what we describe here is not same as hold-out splitting (see later).



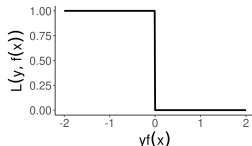
# INNER VS OUTER LOSS

- Sometimes, we would like to evaluate our learner with a different loss  $L$  or metric  $\rho$ .
- Nomenclature: ERM and **inner loss**; evaluation and **outer loss**.
- Different losses, if computationally advantageous to deviate from outer loss of application; e.g., optimization faster with inner L2 or maybe no implementation for outer loss exists.



**Example:** Linear binary classifier / Logistic regression.

- Outside: We often want to eval with "nr of misclassified examples", so 0-1 loss.
- Problem: 0-1 neither differentiable nor continuous. Hence: Inner loss = binomial. (0-1 actually NP hard).
- For evaluation, differentiability is not required.



# SET-BASED PERFORMANCE METRICS

- Metric  $\rho$  measures quality of predictions as scalar on one test set.

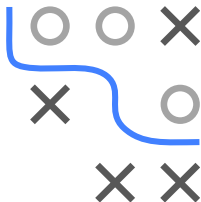
$$\rho : \bigcup_{m \in \mathbb{N}} (\mathcal{Y}^m \times \mathbb{R}^{m \times g}) \rightarrow \mathbb{R}, \quad (\mathbf{y}, \mathbf{F}) \mapsto \rho(\mathbf{y}, \mathbf{F}).$$

- Needed as some metrics are not observation-based losses but defined on sets, e.g. AUC or metrics in survival analysis.
- For test data of size  $m$ ,  $\mathbf{F}$  is prediction matrix

$$\mathbf{F} = \begin{bmatrix} \hat{f}(\mathbf{x}^{(1)}) \\ \dots \\ \hat{f}(\mathbf{x}^{(m)}) \end{bmatrix} \in \mathbb{R}^{m \times g}$$

- Point-wise loss  $L$  can easily be extended to a  $\rho_L$ :

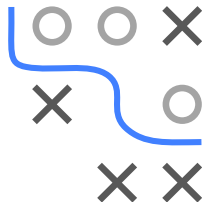
$$\rho_L(\mathbf{y}, \mathbf{F}) = \frac{1}{m} \sum_{i=1}^m L(y^{(i)}, \mathbf{F}^{(i)}) \quad \left( = \frac{1}{m} \sum_{i=1}^m L(y^{(i)}, \hat{f}(\mathbf{x}^{(i)})) \right).$$



# MODEL GE VS. LEARNER GE

To clear up a major point of confusion (or totally confuse you):

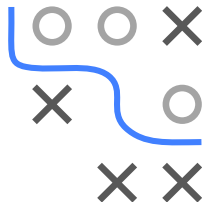
- In ML we frequently face a weird situation.
- We are usually given a single data set, and at the end of our model fitting (and evaluation and selection) process, we will likely fit one model on exactly that complete data set.
- We only trust in unseen-test-error estimation – but have no data left for that final model.
- So in the construction of any practical estimator we cannot really use that final model!
- Hence, we will now evaluate the next best thing: The inducer, and the quality of a model produced when fitted on (nearly) the same number of points!



# GENERALIZATION ERROR FOR INDUCER

$$\text{GE}(\mathcal{I}, \lambda, n_{\text{train}}, \rho) := \lim_{n_{\text{test}} \rightarrow \infty} \mathbb{E} [\rho(\mathbf{y}, \mathbf{F}_{\mathcal{D}_{\text{test}}, \mathcal{I}(\mathcal{D}_{\text{train}}, \lambda)})]$$

- Quality of models when fitted with  $\mathcal{I}_\lambda$  on  $n_{\text{train}}$  points from  $\mathbb{P}_{xy}$ .
- Expectation **both** over  $\mathcal{D}_{\text{train}}$  and  $\mathcal{D}_{\text{test}}$ , sampled independently.
- This is estimated by all following **resampling** procedures.
- NB: All of the models produced during that phase of evaluation are only intermediate results.





## GENERALIZATION ERROR FOR INDUCER

$$\text{GE}(\mathcal{I}, \lambda, n_{\text{train}}, \rho) := \lim_{n_{\text{test}} \rightarrow \infty} \mathbb{E} [\rho(\mathbf{y}, \mathbf{F}_{\mathcal{D}_{\text{test}}, (\mathcal{I}(\mathcal{D}_{\text{train}}, \lambda))})]$$

- We can already see a potential source of pessimistic bias in our estimator: While we would like to estimate a GE with  $n_{\text{train}} = |\mathcal{D}|$ , the size of the complete data set, in practice we can only do this for strictly smaller values, so that test data is left to work with.
- For pointwise losses  $\rho_L$ :

$$\text{GE}(\mathcal{I}, \boldsymbol{\lambda}, n_{\text{train}}, \rho_L) := \mathbb{E} [L(y, \mathcal{I}(\mathcal{D}_{\text{train}}, \boldsymbol{\lambda})(\mathbf{x}))]$$

Expectation **both** over  $\mathcal{D}_{\text{train}}$  and  $(\mathbf{x}, y)$  independently from  $\mathbb{P}_{xy}$ .

- Retcon for GE of model: GE of learner, conditional on  $\mathcal{D}_{\text{train}}$

$$\text{GE}(\hat{f}, L) := \text{GE}(\mathcal{I}, \boldsymbol{\lambda}, n_{\text{train}}, \rho_L | \mathcal{D}_{\text{train}})$$

$$\text{if } \hat{f} = \mathcal{I}(\mathcal{D}_{\text{train}}, \lambda) \text{ and } n_{\text{train}} = |\mathcal{D}_{\text{train}}|.$$
