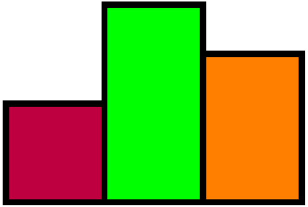
Introduction to Machine Learning

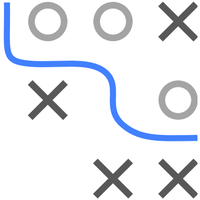
Evaluation: Generalization Error



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

PERFORMANCE ESTIMATION



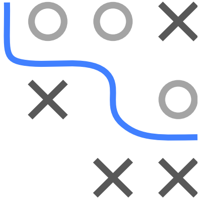
- For a trained model, we want to know its future **performance**.
- Training works by ERM on $\mathcal{D}_{\text{train}}$ (inducer, loss, risk minimization):

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

$$\min_{\theta \in \Theta} \sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)$$

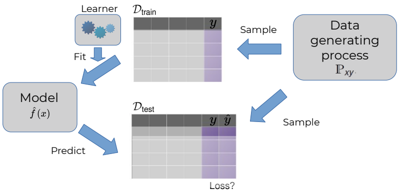
- Due to effects like overfitting, we cannot simply use this **training error** to gauge our model, this is likely optimistically biased. (more on this later!)
- We want: the true expected loss, a.k.a. **generalization error**.
- To reliably estimate it, we need independent, unseen **test data**.
- This simply simulates the application of the model in reality.

GE FOR A FIXED MODEL



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

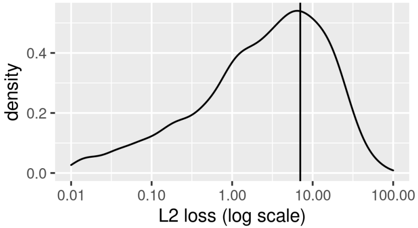
$$\widehat{\mathrm{GE}}(\hat{f}, L) := \frac{1}{m} \sum_{(\mathbf{x},y)\in\mathcal{D}_{\text{test}}} \left[L\left(y, \hat{f}(\mathbf{x})\right)\right]$$



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

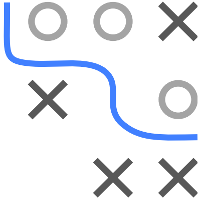
EXAMPLE: TEST LOSS AS RANDOM VARIABLE

- For a fixed model and dedicated i.i.d. test set, we can easily approximate the complete test loss distribution $L(y, \hat{f}(\mathbf{x}))$.
- LM on $\mathrm{mlbench::friedman1}$ test problem
- With $n_{\mathrm{train}} = 500$ we create a fixed model
- We feed 5000 fresh test points to model
- And plot the pointwise $L2$ loss.



- The result is a unimodal distribution with long tails.
- Mean and one standard deviation to either side are highlighted in grey.

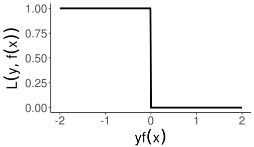
INNER VS OUTER LOSS



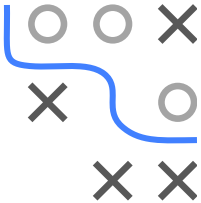
- Sometimes, we would like to evaluate our learner with a different loss L or metric ρ.
- Nomenclature: ERM and **inner loss**; evaluation and **outer loss**.
- Different losses, if computationally advantageous to deviate from outer loss of application; e.g., optimization faster with inner L2 or maybe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

Example: Linear binary classifier / Logistic regression.

- Outside: We often want to eval with "nr of misclassifed 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 ρ measures quality of predictions as scalar on one test set.

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

- 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, $\boldsymbol{F}$ is prediction matrix

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

- Point-wise loss L can easily be extended to a ρ_L:

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

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

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

- Quality of models when fitted with $\mathcal{I}_{\boldsymbol{\lambda}}$ on n_{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

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

- We can already see a potential source of pessimistic bias in our estimator: While we would like to estimate a GE with $n_{\mathrm{train}} = |\mathcal{D}|$, 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 ρ_L:

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

  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 $\mathcal{D}_{\text{train}}$

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

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