

# I2ML :: EVALUATION AND TUNING

## Set-Based Performance Metrics

$J \in \{1, \dots, n\}^m$  :  $m$ -dimensional index vector for a dataset  $\mathcal{D} \in \mathbb{D}_n$ , which also induces  $\mathcal{D}_J = \left(\mathcal{D}^{(J^{(1)})}, \dots, \mathcal{D}^{(J^{(m)})}\right) \in \mathbb{D}_m$

$y_J = \left(y^{(J^{(1)})}, \dots, y^{(J^{(m)})}\right) \in \mathcal{Y}^m$  : vector of labels

$F_{J,f} = \left(f(x^{(J^{(1)})}), \dots, f(x^{(J^{(m)})})\right) \in \mathbb{R}^{m \times g}$  : matrix of prediction scores regarding a model  $f$

General **performance measure**:  $\rho : \bigcup_{m \in \mathbb{N}} (\mathcal{Y}^m \times \mathbb{R}^{m \times g}) \rightarrow \mathbb{R}$  maps every  $m$ -dimensional label vector  $y_J$  and its matrix of prediction scores  $F_{J,f}$  to a scalar performance value.

$\rho_L(y, F) = \sum_{i=1}^m L(y^{(i)}, F^{(i)})$  : performance measure induced by an arbitrary point-wise loss  $L$

$\mathcal{P}$  : set of all performance measures  $\rho$

## Generalization Error

The **generalization error** GE is the performance of a model induced by  $\mathcal{I}_\lambda$  from datasets  $\mathcal{D}_{\text{train}} \sim (\mathbb{P}_{xy})^{m_{\text{train}}}$  evaluated with performance measure  $\rho$  over a dataset  $\mathcal{D}_{\text{test}} \sim (\mathbb{P}_{xy})^{m_{\text{test}}}$  when  $m_{\text{test}} \rightarrow \infty$ , i.e.,

$$\text{GE}(\mathcal{I}, \lambda, m_{\text{train}}, \rho) = \lim_{m_{\text{test}} \rightarrow \infty} \mathbb{E} \left[ \rho(y, F_{J_{\text{test}}, f_{\mathcal{D}_{\text{train}}, \lambda}}) \right],$$

where  $f_{\mathcal{D}_{\text{train}}, \lambda} = \mathcal{I}(\mathcal{D}_{\text{train}}, \lambda)$  and the expectation is taken over both datasets  $\mathcal{D}_{\text{train}}$  and  $\mathcal{D}_{\text{test}}$  ( $= \mathcal{D}_{J_{\text{test}}}$ ).

## Data Splitting and Resampling

$S = ((J_{\text{train},1}, J_{\text{test},1}), \dots, (J_{\text{train},K}, J_{\text{test},K})) \in \mathbb{S}_K$  : **resampling strategy** consisting of  $K$  train-test-splits  $(J_{\text{train},i}, J_{\text{test},i})$

**Estimator of the generalization error**  $\text{GE}(\mathcal{I}, \lambda, m_{\text{train}}, \rho)$ :

$$\begin{aligned} \widehat{\text{GE}}_S(\mathcal{I}, \lambda, \rho) &= \text{agr} \left( \rho(y_{J_{\text{test},1}}, F_{J_{\text{test},1}, f_{\mathcal{D}_{\text{train},1}, \lambda}}), \right. \\ &\quad \vdots \\ &\quad \left. \rho(y_{J_{\text{test},K}}, F_{J_{\text{test},K}, f_{\mathcal{D}_{\text{train},K}, \lambda}}) \right), \end{aligned}$$

where the aggregating function  $\text{agr}$  is often the mean and  $m_{\text{train}} \approx m_{\text{train},1} \approx \dots \approx m_{\text{train},K}$  and  $m_{\text{train}} = \text{mode}(m_{\text{train},1}, \dots, m_{\text{train},K})$

### Resampling Strategies

**K-fold cross-validation** :

**Leave-one-out cross validation** : n-fold cross-validation

Repeated **subsampling** / Monte Carlo cross-validation :

**Bootstrap sampling** :

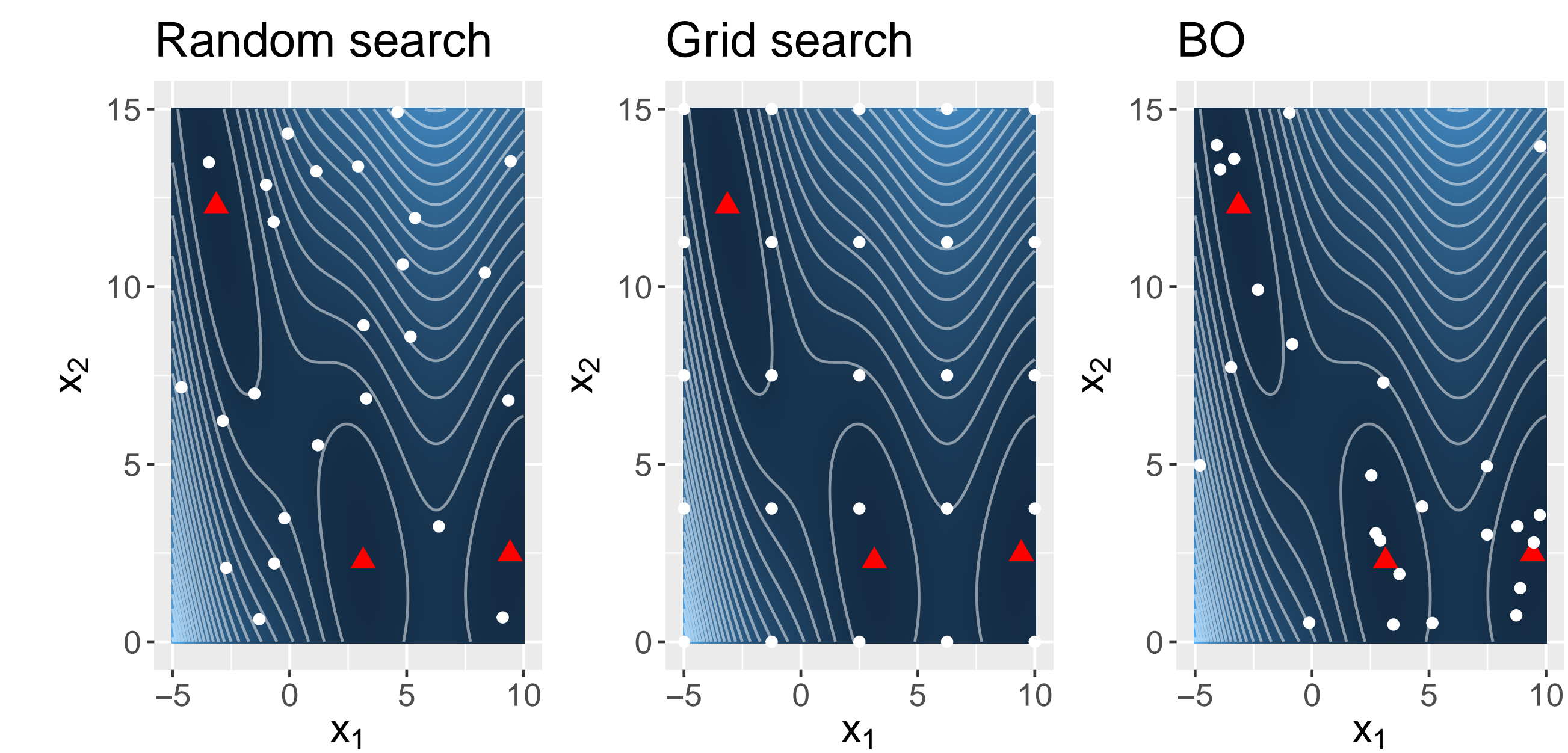
## Tuning

**Tuner**  $\mathcal{T} : \mathbb{D} \times \mathbb{S}_K \times \{\mathcal{I} : \mathcal{I} \text{ is a learner}\} \times \mathcal{P} \rightarrow \mathcal{H}$  takes a dataset  $\mathcal{D} \in \mathbb{D}$ , and produces a model  $f$  learned with inducer  $\mathcal{I}_{\widehat{\lambda}^*}$  where

$$\widehat{\lambda}^* \approx \lambda^* = \arg \min_{\lambda \in \Lambda} \widehat{\text{GE}}_S(\mathcal{I}, \lambda, \rho), \text{ i.e.,}$$

the optimal hyperparameter regarding the performance measure  $\rho \in \mathcal{P}$  and the resampling strategy  $S \in \mathbb{S}_K$  defined on  $\mathcal{D}$ .

### Black-Box Optimization Techniques



**Random search** :

**Grid search** :

**Bayesian optimization** :

**Evolutionary algorithms** :

**Hyperband** :

## Nested Resampling

