

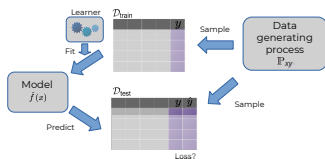
# Introduction to Machine Learning

## Evaluation: In a Nutshell



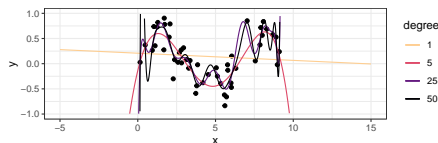
### Learning goals

- Understand what the Generalization Error is
- Get an overview on how we evaluate performance of learners
- Know some evaluation metrics for classification and regression
- Understand why we do resampling

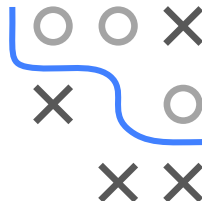


# EVALUATING A MODEL

- We have seen how to train models that are optimal in some sense, relative to other possible models. However, how can we assess how good they actually are, in absolute numbers?
- Idea: Use risk  $\sum_{(\mathbf{x}, y) \in \mathcal{D}_{\text{train}}} \left[ L(y, \hat{f}(\mathbf{x})) \right]$  after training.
- Problem: This value can be very optimistic.
- Example: Overfitting of a polynomial regression.

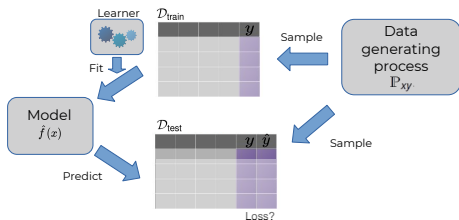
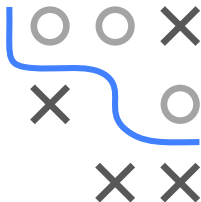


- Degree 50 will result in lowest training loss, however, degree 5 seems to be the "best" model.
- "Best" means that using new data, this model will probably produce the most meaningful predictions.



# GENERALIZATION ERROR

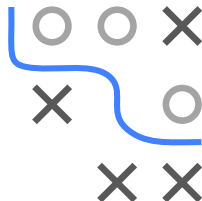
- In other words, the "best" model will generalize well and have a low **Generalization Error**.
- Formally, for a fixed model, the GE can be expressed via:
$$\text{GE}(\hat{f}, L) := \mathbb{E} \left[ L(y, \hat{f}(\mathbf{x})) \right],$$
- i.e., "what is the expected loss for a new observation?"
- Ideally, the GE should be estimated with new, unseen data.
- Usually, we have no access to new **unseen** data, though.
- Thus, we divide our data set manually into  $\mathcal{D}_{\text{train}}$  and  $\mathcal{D}_{\text{test}}$  and use the latter to estimate the GE via some metric  $\rho()$ .



# METRICS

But what is a good metric  $\rho()$ ?

- While we can always use the (inner) loss function that we trained the model on as outer loss to construct a metric  $\rho()$ , this may not always be ideal.
- For both, classification and regression there is a large variety of evaluation metrics, of which we will just cover a fraction.



# METRICS FOR CLASSIFICATION

Commonly used evaluation metrics include:

- Accuracy:

- $\rho_{ACC} = \frac{1}{m} \sum_{i=1}^m [y^{(i)} = \hat{y}^{(i)}] \in [0, 1]$ .
- "Proportion of correctly classified observations."

- Misclassification error (MCE):

- $\rho_{MCE} = \frac{1}{m} \sum_{i=1}^m [y^{(i)} \neq \hat{y}^{(i)}] \in [0, 1]$ .
- "Proportion of incorrectly classified observations."

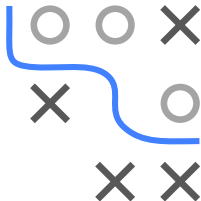
- Brier Score:

- $\rho_{BS} = \frac{1}{m} \sum_{i=1}^m (\hat{\pi}^{(i)} - y^{(i)})^2$
- "Squared error btw. predicted probability and actual label."

- Log-loss:

- $\rho_{LL} = \frac{1}{m} \sum_{i=1}^m (-y^{(i)} \log(\hat{\pi}^{(i)}) - (1 - y^{(i)}) \log(1 - \hat{\pi}^{(i)}))$ .
- "Distance of predicted and actual label distribution."

The probabilistic metrics, Brier Score and Log-Loss, penalize false confidence, i.e. predicting the wrong label with high probability, heavily.



# METRICS FOR CLASSIFICATION / 2

For hard-label classification, the confusion matrix is a useful tool:

		True Class $y$	
		+	-
Pred.	+	True Positive (TP)	False Positive (FP)
$\hat{y}$	-	False Negative (FN)	True Negative (TN)



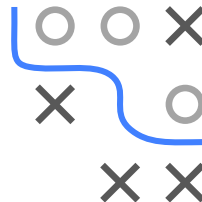
From this matrix a variety of evaluation metrics, including precision and recall, can be computed.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

# METRICS FOR CLASSIFICATION / 3

- Other frequently used metrics like the False Negative Rate can also be derived from the confusion matrix.
- Many of these metrics can go with different names.



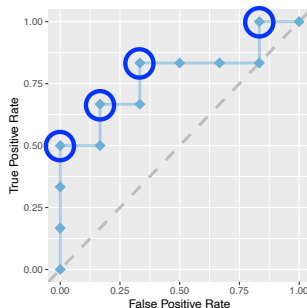
		True condition			
		Total population	Condition positive	Condition negative	
Predicted condition	Predicted condition positive	True positive, Power	False positive, Type I error	Prevalence $= \frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	Accuracy (ACC) = $\frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$
	Predicted condition negative	False negative, Type II error	True negative	Positive predictive value (PPV), Precision = $\frac{\sum \text{True positive}}{\sum \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\sum \text{False positive}}{\sum \text{Predicted condition positive}}$
		True positive rate (TPR), Recall, Sensitivity, probability of detection $= \frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	False omission rate (FOR) = $\frac{\sum \text{False negative}}{\sum \text{Predicted condition negative}}$	Negative predictive value (NPV) $= \frac{\sum \text{True negative}}{\sum \text{Predicted condition negative}}$
		False negative rate (FNR), Miss rate $= \frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) $= \frac{\sum \text{True negative}}{\sum \text{Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Negative likelihood ratio (LR-) $= \frac{\text{FNR}}{\text{TNR}}$
				Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR-}}$	F <sub>1</sub> score = $\frac{1}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}}$

► Clickable version/picture source

► Interactive diagram

# ROC-CURVE

- The ROC-Curve allows to evaluate binary classifiers beyond single metrics. It compares classifiers using their TPR and FPR, for different thresholds.
- We aim to identify good thresholds that dominate others.
- The area under this curve (AUC) can also be used as metric.

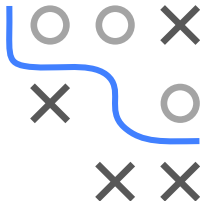




# METRICS FOR REGRESSION

Commonly used evaluation metrics include:

- Sum of Squared Errors (SSE):  $\rho_{SSE} = \sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2$
- Mean Squared Error (MSE):  $\rho_{MSE} = \frac{1}{m} \sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2$
- Root Mean Squared Error (RMSE):  $\rho_{RMSE} = \sqrt{MSE}$
- R-Squared:  $\rho_{R^2} = 1 - \frac{\sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^m (y^{(i)} - \bar{y})^2}$
- Mean Absolute Error (MAE):  $\rho_{MAE} = \frac{1}{m} \sum_{i=1}^m |y^{(i)} - \hat{y}^{(i)}|$



# IMPROVING ESTIMATION OF GE

We can estimate the GE with a test data set via:

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

i.e. we compute the selected metric  $L(y, \hat{f}(\mathbf{x}))$  for each observation in the test set and compute the mean.

This will give an appropriate estimate for the GE. However, with only a few test observations (small  $m$ ), this estimate will be unstable or, in other words, have high variance. We have two options to decrease it:

- Increase  $m$ .
- Compute  $\widehat{\text{GE}}(\hat{f}, L)$  for multiple test sets and aggregate them.





