I2ML :: CHEAT SHEET

The **I2ML**: Introduction to Machine Learning course offers an introductory and applied overview of "supervised" Machine Learning. It is organized as a digital lecture.

Performance Evaluation

Performance Evaluation refers to estimating performance on new data.

Different levels of randomness:

- The sample can be too small, then our estimator will be of high variance or if the sample could not be from the distribution of interest, then our estimator will be biased.
- Many learning algorithms are stochastic. Example: Random forest, Stochastic gradient descent

Metrics: inner vs. outer loss

Inner loss is used in learning and outer loss is used in evaluation. Optimally inner loss should always match outer loss. But this is not always possible because some losses are hard to optimize.

Simple Metrics

Metrics for Label based prediction methods: Accuracy, MCE, Costs, Confusion matrix, F1 measure, ROC curve, Precision, Recall etc.

Metrics for Probabilistic prediction methods: Brier Score, Log-Loss etc.

	Metrics	Mathematical expression
Regression	Mean Squared Error (Squared Error Loss)	$MSE = \frac{1}{n}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 \in [0; \infty]$
	Mean Absolute Error (Mean Error Loss))	$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert y^{(i)} - \hat{y}^{(i)}\rvert \in [0; \infty]$
	R^2	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^{n}(y^{(i)} - \bar{y})^2}$
Labels	Misclassification error rate	$MCE = \frac{1}{n}\sum_{i=1}^{n}[y^{(i)} \neq \hat{y}^{(i)}] \in [0; 1]$
	Accuracy	$ACC = \frac{1}{n}\sum_{i=1}^{n}[y^{(i)} = \hat{y}^{(i)}] \in [0; 1]$
	Costs	$Costs = \frac{1}{n}\sum_{i=1}^{n} C[y^{(i)}, \hat{y}^{(i)}]$
Probabilities	Brier Score	$BS = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{\pi}(\mathbf{x}^{(i)}) - y^{(i)}\right)^2$
	Bernoulli loss	$LL = \frac{1}{n}\sum_{i=1}^{n}\left(-y^{(i)}\log(\hat{\pi}(\mathbf{x}^{(i)})) - (1 - y^{(i)})\log(1 - \hat{\pi}(\mathbf{x}^{(i)}))\right)$

Train and Test Error

Training Error: estimated by the averaging error over the same data set we fitted on

Problems of training error:

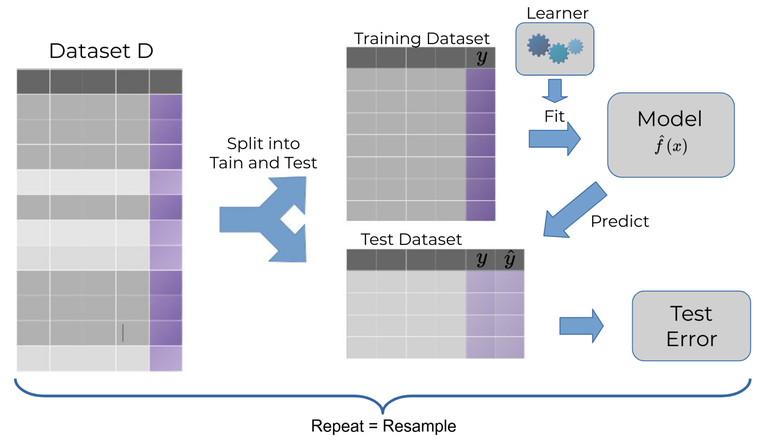
- Unreliable and overly optimistic estimator of future Performance Evaluation
- There are interpolators - interpolating splines, interpolating Gaussian processes - they are not necessarily good as they will also interpolate the noise
- Goodness-of-fit measures like R^2, likelihood, AIC, BIC etc are based on the training error

Test Error: the test is a good way to estimate future Performance Evaluation, given that the test data is i.i.d. compared to the data we will see when we apply the model.

Problems of test error:

- The estimator will suffer from high variance and be less reliable if the test set is too small
- Sometimes the test set is large, but one of the two classes is small

Holdout Splitting: It is a tool for estimating future Performance Evaluation. All of the models produced during that phase of evaluation are intermediate results.



Generalization error: is a measure of how accurately an algorithm is able to predict outcome values for previously unseen data

Overfitting: happens when our algorithm starts modelling patterns in the data that are not actually true in the real world, e.g., noise in the training data

Avoiding Overfitting: Use less complex models, get more and better data, early stopping, regularization etc.

Resampling

The aim is to assess the Performance Evaluation of learning algorithm. Uses the data more efficiently and repeatedly splits in train and test, then average results.

Cross-validation: Split the data into k roughly equally-sized partitions. Use each part once as test set and join the $k - 1$ others for training, obtain k test errors and average.

Bootstrapping: Randomly draw B training sets of size n with replacement from the original training set $\mathcal{D}_{\text{train}}$

Subsampling: Repeated hold-out with averaging, a.k.a. monte-carlo CV. Similar to bootstrap, but draws without replacement