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

Generalization Error:

Generalization error (GE) is a measure of how accurately an algorithm is able to predict outcome values for previously unseen data.
For a fixed model: $\mathrm{GE}\left(\hat{f}, L\right) := \mathbb{E}\left[L\left(y, \hat{f}(\mathbf{x})\right)\right]$, i.e. the expected error the model makes for data $(\mathbf{x}, y) \sim \mathbb{P}_{xy}$. If $(\mathbf{x}, y) \in \mathcal{D}_{\text{train}}$, estimator will be biased via overfitting the training data. Thus, we estimate the GE using unseen data $(\mathbf{x}, y) \in \mathcal{D}_{\text{test}}$:

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

$L\left(y, \hat{f}(\mathbf{x})\right)$ indicates how good the target matches our prediction.

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.

Training and Test Error

Training Error:

Training error: average error over the same data set fitted on.
Problems of training error:
- Unreliable and overly optimistic estimator of future performance evaluation.
- There are interpolators, which 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:

Test is a good way to estimate future performance evaluation, given that the test data is i.i.d. compared to the data applied to 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.

Overfitting vs. Underfitting:

Overfitting:
- The model fits the training data too well that it also models patterns in the data that are not actually true, like noise.
- Small training error, at cost of test high error.
- Low bias, high variance.

Avoid overfitting: Use less complex models, get more and better data, early stopping, regularization etc.
Underfitting:
- The model is too simple to learn the underlying structure of the data.
- High training error and high test error.
- High bias, low variance.

Resampling

While $\widehat{\mathrm{GE}}(\hat{f}, L) = \frac{1}{m}\sum_{(\mathbf{x},y)\in\mathcal{D}_{\text{test}}} \left[L\left(y, \hat{f}(\mathbf{x})\right)\right]$ is unbiased, with a small m it has high variance. We have two options to decrease the variance:
- Increase m.
- Compute $\widehat{\mathrm{GE}}(\hat{f}, L)$ for multiple test sets and aggregate them.

With a finite amount of data, increasing m would mean to decrease the size of the training data. Thus, we focus on using multiple (B) test sets:

$$\mathcal{J} = \left((J_{\text{train},1}, J_{\text{test},1}), \ldots, (J_{\text{train},B}, J_{\text{test},B})\right).$$

where we compute $\widehat{\mathrm{GE}}(\hat{f}, L)$ for each set and aggregate the estimates. These B sets are generated through **resampling**.

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. For unbalanced data, **stratification** is used.

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.

Evaluation Curves

Confusion Matrix:

F_1 **score** balances 2 conflicts:
- Maximize positive predictive value.
- Maximize true positive rate.

$$\rho_{F_1} = 2 \cdot \frac{\rho_{PPV} \cdot \rho_{TPR}}{\rho_{PPV} + \rho_{TPR}}.$$

		True Class y +	True Class y −	
Pred. $\hat{y}$	+	TP	FP	$\rho_{PPV} = \frac{\text{TP}}{\text{TP+FP}}$
	−	FN	TN	$\rho_{NPV} = \frac{\text{TN}}{\text{FN+TN}}$
		$\rho_{TPR} = \frac{\text{TP}}{\text{TP+FN}}$	$\rho_{TNR} = \frac{\text{TN}}{\text{FP+TN}}$	$\rho_{ACC} = \frac{\text{TP+TN}}{\text{TOTAL}}$

Receiver Operating Characteristic (ROC):

- Rank test observations on decreasing score.
- Start with $c = 1$ in $(0, 0)$; predict everything as negative.
- Iterate through all possible thresholds c and proceed for each observation x as follows:
  - ▶ If x is positive, move TPR $\frac{1}{n_+}$ up, as we have one TP more.
  - ▶ If x is negative, move FPR $\frac{1}{n_-}$ right, as we have one FP more.

ROC Properties:
- The closer the curve to the top-left corner, the better.
- If ROC curves cross, a different model might be better in different parts of the ROC space.

Area Under the Curve (AUC):

AUC $\in [0, 1]$ is a single metric to evaluate scoring classifiers, independent of the chosen threshold. AUC $= 1$: perfect classifier; AUC $= 0.5$: random, non-discriminant classifier; AUC $= 0$: perfect, with inverted labels.
Mann-Whitney-U test: A **non-parametric hypothesis test** on the difference in location between two samples X_1, X_2 of sizes n_1 and n_2. Test statistic estimates the probability of a random sample from X_1 ranking higher than one from X_2 (R_1 denoting the sum of ranks of the $x_{1,i}$):

$$U = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \mathbb{I}[x_{1,i} > x_{2,j}] = R_1 - \frac{n_1(n_1+1)}{2}$$

AUC is U normalized to the unit square: $\text{AUC} = \frac{U}{n_+ \cdot n_-}$.
Partial AUC (pAUC): Area under the ROC curve in a limited region of interest, e.g. TPR > 0.8 or FPR < 0.2.

Multi-class AUC: AUC($k \mid \ell$) for classes k (pos) and ℓ (neg).

$$\text{AUC}_{MC} = \frac{1}{g(g-1)} \sum_{k\neq\ell} \text{AUC}(k|\ell) \in [0, 1].$$

Precision-Recall (PR) Curves:

Slightly changed ROC plot. Simply plot precision and recall, instead of TPR-FPR.