Machine Learning

Model Selection and Validation

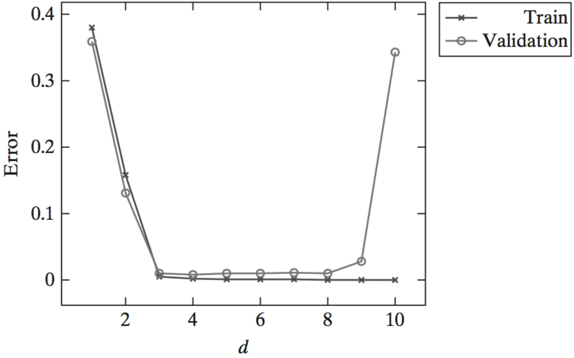
Fabio Vandin

November 5, 2020

Model-Selection Curve

Shows the training error and validation error as a function of the complexity of the model considered

Example



Training error decreases but validation error increases ⇒ *overfitting*

What if we have one or more parameters with values in $\mathbb{R}$?

1. Start with a rough *grid* of values
2. Plot the corresponding model-selection curve
3. Based on the curve, zoom in to the correct *regime*
4. Restart from 1) with a finer grid

Note: the empirical risk on the validation set *is not* an estimate of the true risk, in particular if r is large (i.e., we choose among many models)!

Question: how can we estimate the true risk after model selection?

Train-Validation-Test Split

Assume we have $\mathcal{H} = \cup_{i=1}^{r} \mathcal{H}_i$

Idea: instead of splitting data in 2 parts, divide into 3 parts

1. *training set*: used to learn the best model h_i from each $\mathcal{H}_i$
2. *validation set*: used to pick one hypothesis h from $\{h_1, h_2, \ldots, h_r\}$
3. *test set*: used to estimate the true risk $L_{\mathcal{D}}(h)$

$\Rightarrow$ the estimate from the test set has the guarantees provided by the proposition on estimate of $L_{\mathcal{D}}(h)$ for 1 class

Note:

- the test set *is not involved* in the choice of h
- if after using the test set to estimate $L_{\mathcal{D}}(h)$ we decide to choose another hypothesis (*because we have seen the estimate of $L_{\mathcal{D}}(h)$ from the test set...*)
  $\Rightarrow$ we cannot use the test set again to estimate $L_{\mathcal{D}}(h)$!

k-Fold Cross Validation

When data is not plentiful, we cannot afford to use a *fresh* validation set $\Rightarrow$ cross validation

$\Rightarrow$ k-fold cross validation:

1. partition (training) set into k *folds* of size m/k
2. for each fold:
    - train on union of other folds
    - estimate error (for learned hypothesis) from the fold
3. estimate of the true error = average of the estimated errors above

Lease-one-out cross validation: $k = m$

Often cross validation is used for model selection

- at the end, the final hypothesis is obtained from training on the entire training set

k-Fold Cross Validation for Model Selection

input:

training set $S = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$

set of parameter values Θ

learning algorithm A

integer k

partition S into $S_1, S_2, \ldots, S_k$

foreach $\theta \in \Theta$

for $i = 1 \ldots k$

$h_{i,\theta} = A(S \setminus S_i; \theta)$

$\text{error}(\theta) = \frac{1}{k} \sum_{i=1}^{k} L_{S_i}(h_{i,\theta})$

output

$\theta^\star = \text{argmin}_\theta \left[\text{error}(\theta)\right]$

$h_{\theta^\star} = A(S; \theta^\star)$