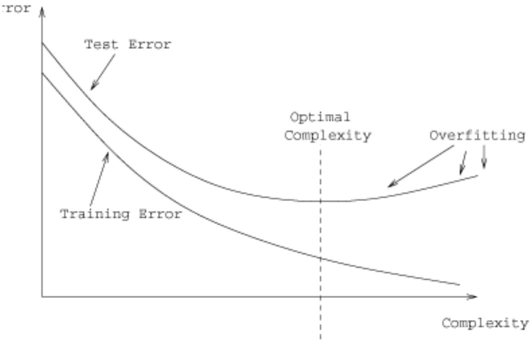
Lecture
–
Model Assessment
Model Selection

Agenda

- Generalization ability
- Bias, variance and model complexity
- The "data-rich situation": Train-Validation-Test
- The training error: a too optimistic estimate
- Structural risk minimization (VC theory)
- Cross-validation: a popular method for prediction error estimation
- Bootstrap techniques

Looking for the right amount of complexity



Errors, training errors, generalization errors

- Learning is based on a training sample

$$\mathcal{D}_n = \{(X_1, Y_1),\ \ldots,\ (X_n, Y_n)\}$$

- The classifier $\hat{C}_n \in \mathcal{G}$ selected through an "ERM like" method is **random**, depending on $\mathcal{D}_n$, as well as its **error**:

$$L(\hat{C}_n) = \mathbb{E}\left[\mathbb{I}\{Y \neq \hat{C}_n(X)\} \mid \mathcal{D}_n\right]$$

  Expectation is taken over a pair (X, Y) independent from training data $\mathcal{D}_n$

- The **generalization error**: take next expectation over $\mathcal{D}_n$

$$Err = \mathbb{E}\left[L(\hat{C}_n)\right]$$

Methods for performance assessment, for model selection

- Training error is not a good estimate!

$$\hat{L}_n(\hat{C}_n) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{Y_i \neq C(X_i)\}$$

  It vanishes as soon as the class $\mathcal{G}$ is complex enough
  $\Rightarrow$ Overfitting and poor generalization

- The objective is twofold
  - Model selection: choose the best model among a collection of models
  - Model assessment: for a given model, estimate its generalization error

When data are not expensive

- Divide the data into three parts:

  Training - Validation - Test

- Typical choice: 50% - 25% - 25%

- $K \geq 1$ model candidates: $\mathcal{G}_1, \ldots, \mathcal{G}_K$

  - For each $k \in \{1, \ldots, K\}$, apply ERM to training data $\Rightarrow \hat{C}^{(k)}$
  - Use validation data to find the "best" $\hat{k} \in \{1, \ldots, K\}$
  - Estimate the error using the test data (independent from $\hat{k}$)

- How to proceed in a data-poor situation?

  Complexity regularization (structural risk minimiation), resampling methods, *etc.*

Model selection by penalization

- Consider a sequence of model classes $\mathcal{G}_1, \mathcal{G}_2, \ldots$
  As $k \nearrow +\infty$, $\mathcal{G}_k$ gets richer
- Let $\hat{C}^{(k)}$ be the empirical risk minimizer over $\mathcal{G}_k$
- Our goal: select $\hat{k}$ so that $\mathbb{E}[L(\hat{C}^{(\hat{k})})] - L^*$ is close to

$$\min_k \mathbb{E}[L(\hat{C}^{(k)})] - L^* =$$

$$\min_k \left\{ \left(\mathbb{E}[L(\hat{C}^{(k)})] - \inf_{C \in G_k} L(C) \right) + \left(\inf_{C \in G_k} L(C) - L^* \right) \right\}$$

- Idea: add a complexity penalty to the training error to compensate the overfitting effect

$$\hat{L}_n(\hat{C}^{(k)}) + pen(n, k)$$

- The penalty may depend on the data or not
- The penalty is related to a distribution-free upper bound for

Complexity regularization

- Suppose that an estimate $R_{n,k}$ of $L(\hat{C}_k)$ is available, s.t. for all $\epsilon > 0$

$$\mathbb{P}\left\{L(\hat{C}_k) - R_{n,k} > \epsilon\right\} \leq ce^{-2m\epsilon^2}$$

  for fixed constants c, m

- The ideal optimization would be

$$L(\hat{C}_k) - \hat{L}_n(\hat{C}_k)$$

  that can be estimated by

$$R_{n,k} - \hat{L}_n(\hat{C}_k)$$

- This yields $pen(n,k) = R_{n,k} - \hat{L}_n(\hat{C}_k) + \sqrt{\log(k)/m}$

Complexity regularization

- Select the prediction rule

$$C_n^* = \arg\min_k \tilde{L}_n(\hat{g}_k)$$

based on the complexity penalized training error

$$\tilde{L}_n(\hat{g}_k) = \hat{L}_n(\hat{g}_k) + pen(n,k) = R_{n,k} + \sqrt{\log(k)/m}$$

- Penalization by the VC dimension

$$R_{n,k} = \hat{L}_n(\hat{g}_k) + 2\sqrt{\frac{V_{\mathcal{G}_k}\log(n+1) + \log 2}{n}}$$

Cross-Validation

- Goal: estimate the generalization error
- Let $K \geq 1$ (typical choices are 5 or 10), "K-fold cross-validation" (K=n "leave-one-out" estimation)
- Split the data into K parts (of same size)
- For all $k \in \{1, \ldots, K\}$,
  - learn $\hat{C}^{(-k)}$ based on all data except the k-th part
  - calculate the error of $\hat{C}^{(-k)}$ over the k-th part
- Average the K quantities

"Pulling yourself up by your own bootstrap" (Baron de Münchausen)

- Bootstrap (the plug-in principle): estimate the distribution of

  $$\mathbb{E}^*[\mathbb{I}\{\hat{C}(X) \neq Y\}]$$

  where $\mathbb{E}^*[.]$ is the expectation w.r.t. the empirical df of the $(X_i, Y_i)'s$
- Heuristics: replace the unknown df by an estimate
- Monte-Carlo approximation
- Higher-order validity