T-61.3050 Machine Learning: Basic Principles

Multivariate Methods

Kai Puolamäki

Laboratory of Computer and Information Science (CIS)
Department of Computer Science and Engineering
Helsinki University of Technology (TKK)

Autumn 2007

Outline

- Cross-validation: most robust if there is enough data.
- Related:
  - Bayesian model selection: use prior and Bayes' formula.
  - Regularization: add penalty term for complex models (can be obtained, for example, from prior).
  - Minimum description length (MDL): can be viewed as MAP estimate. [Basic idea good to know, details not required in this course.]
- Structural risk minimization (SRM): used, for example, in support vector machines (SVM). [Not required to know in this course.]
- The latter do not strictly require a validation set.
- There is no single best way for small amounts of data (your prior assumptions matter).

Outline

Cross-validation

- Separate data into training and validation sets.
- Learn using training set.
- Use error on validation set to select a model.
- You need a test set also if you want an unbiased estimate of error on new data.
- Question: what is a sufficient size for the validation set?

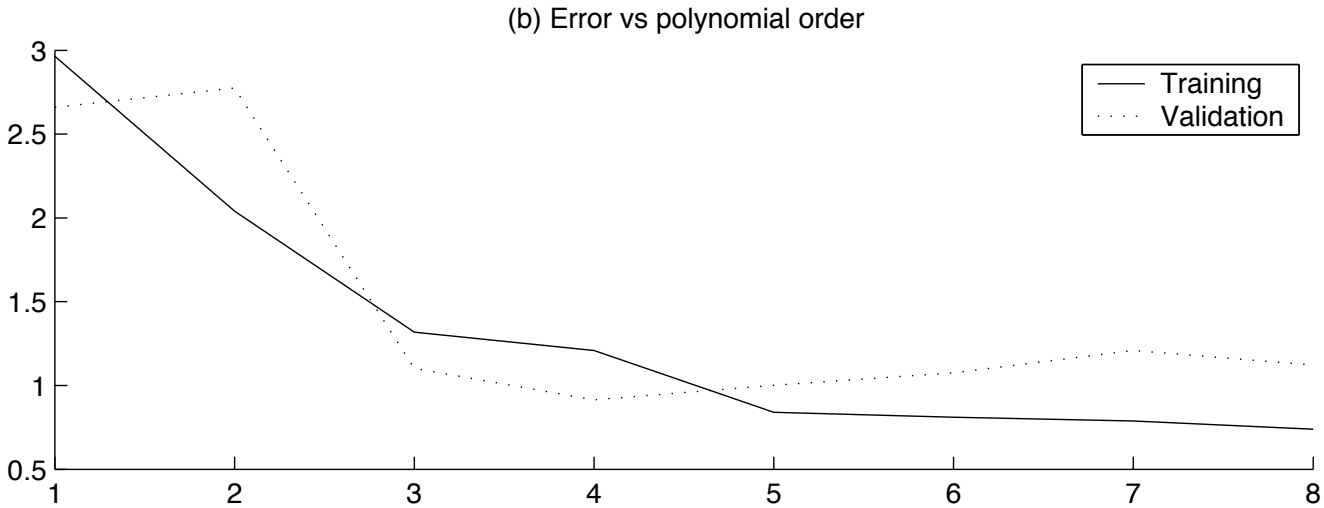


Figure 4.7 of Alpaydin (2004).

Cross-validation

- Assumption: training data $\mathcal{X} = \{(r^t, x^t)\}_{t=1}^N$ has been sampled iid from some (usually unknown) distribution F, $(r^t, x^t) \sim F$.
- In cross-validation, training data is split in random in training set of size $N - n$ and validation set of size n. Effectively then also the validation set is sampled iid from F.
- Classifier $h(x)$ is trained using the training set.
- Generalization error $\mathcal{E}$: probability of misclassification for a new data point $(r, x) \sim F$, $\mathcal{E} = E_F[I(r \neq h(x))]$.
- Fraction of misclassified items in the validation set, E_{VALID}, can be used as an estimate of the generalization error $\mathcal{E}$.
- E_{VALID} is an unbiased estimator of $\mathcal{E}$.
- The variance of the estimator E_{VALID} is $\mathrm{Var}(E_{VALID}) = \sqrt{\mathcal{E}(1-\mathcal{E})/n} \leq 1/(2\sqrt{n})$.

Cross-validation

- Classifier $h(x)$ is trained using the training set.
- Fraction of misclassified items in the validation set, E_{VALID}, can be used as an estimate of the generalization error $\mathcal{E}$.
- If we select model that has the smallest E_{VALID} it is no longer unbiased estimate of the generalization error.
- To get an unbiased estimate of the generalization error we must split the data into three parts (training, validation and test sets).

Outline

Bayesian Model Selection

- Define prior probability over models, $p(\text{model})$.

$$p(\text{model} \mid \text{data}) = \frac{p(\text{data} \mid \text{model})p(\text{model})}{p(\text{data})}$$

- Equivalent to regularization, when prior favors simpler models.
- MAP: choose model which maximizes

$$\mathcal{L} = \log p(\text{data} \mid \text{model}) + \log p(\text{model})$$

- (Notice: we again take logs of probabilities for computational convenience; log of posterior has the same maximum as the original posterior. Evidence $p(\text{data})$ is constant with respect to the model, we can therefore drop it.)

Regularization

- Augment the cost by a term which penalizes more complex models: $E(\theta \mid \mathcal{X}) \rightarrow E'(\theta \mid \mathcal{X}) = E(\theta \mid \mathcal{X}) + \lambda \times \text{complexity}$.
- Example 1, Bayesian linear regression: define a Gaussian prior for the model parameters $\theta = (w_0, w_1)$: $p(w_0) \sim N(0, 1/\lambda)$, $p(w_1) \sim N(0, 1/\lambda)$. The old ML function reads (if the error has an unit variance)

$$\mathcal{L}_{ML}(\theta \mid \mathcal{X}) = -\frac{1}{2} \sum_{t=1}^{N} \left[r^t - w_0 - w_1 x^t \right]^2 + \ldots$$

The MAP estimate gives an additional term

$$\mathcal{L}_{MAP}(\theta \mid \mathcal{X}) = \mathcal{L}_{ML}(\theta \mid \mathcal{X}) - \frac{1}{2} \lambda \left(w_0^2 + w_1^2 \right).$$

This is an example of regularization (the prior favours models with small w_0, w_1).

Regularization

- Example 2, Akaike Information Criterion (AIC): Penalize for more parameters and choose model that maximizes:

$$\mathcal{L}(\theta \mid \mathcal{X}) = \mathcal{L}_{ML}(\theta \mid \mathcal{X}) - M,$$

  where M is the number of adjustable parameters in the model.
- Example 3, Bayesian Information Criterion (BIC): Penalize for more parameters and choose model which maximizes:

$$\mathcal{L}(\theta \mid \mathcal{X}) = \mathcal{L}_{ML}(\theta \mid \mathcal{X}) - \frac{1}{2} M \log N,$$

  where M is the number of adjustable parameters in the model and N is the size of the sample $\mathcal{X}$.
- AIC and BIC have some theoretical justification, however, they are very approximate. They are useful because of their simplicity. They tend to favour (too) simple models.
- Weird intro: `http://www.cs.cmu.edu/~zhuxj/courseproject/aicbic/`

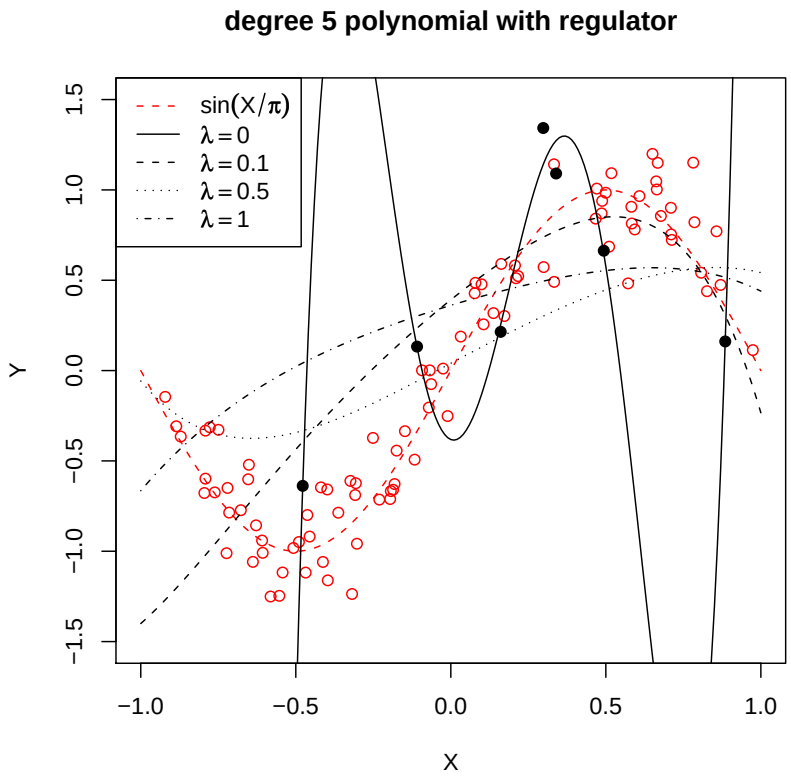
Regression Using Regularization

- Do Bayesian regression with $\sigma^2 = 1$ with the similar data as in the 2nd lecture, use MAP solution with Gaussian prior over parameters.

$$-\mathcal{L}_{MAP} =$$

$$\frac{1}{2}\sum_{t=1}^{7}\left[y^t - g(x^t \mid \overline{w})\right]^2 + \frac{1}{2}\lambda \overline{w}^T \overline{w}.$$

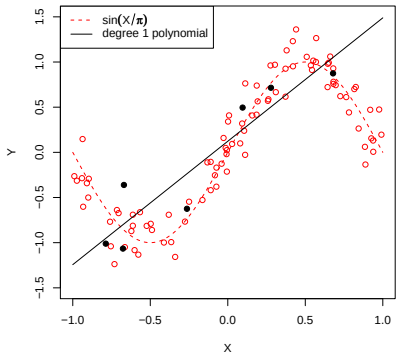
$$g(x \mid \overline{w}) = \sum_{i=0}^{5} w_i x^i.$$



Regression Using Regularization

Do Bayesian regression with $\sigma^2 = 1$ with the same data as in the 2nd lecture, use ML solutions and AIC and BIC regularization:

k	E_{TRAIN}	E_{TEST}	$-\mathcal{L}_{AIC}$	$-\mathcal{L}_{BIC}$
0	0.580	0.541	3.03	3.00
1	0.077	0.294	**2.26**	**2.21**
2	0.076	0.275	3.26	3.18
3	0.057	**0.057**	4.19	4.09
4	0.046	0.562	5.16	5.02
5	0.035	4.637	6.12	5.96
6	0	10^6	7.00	6.81



$$N = 7\ ,\ M = k + 1\ ,\ -\mathcal{L}_{AIC} = \frac{N}{2} E_{TRAIN} + M\ ,$$

$$-\mathcal{L}_{BIC} = \frac{N}{2} E_{TRAIN} + \frac{1}{2} M \log N,$$

$$g(x \mid w_0, \ldots, w_k) = \sum_{i=0}^{k} w_i x^i,$$

$$E_{TRAIN} = -\frac{2}{N}\mathcal{L}_{ML} = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - g(x^t \mid w)\right]^2.$$

Minimum Description Length (MDL)

- Minimum Description Length (MDL): a good model is such that it can be used to give the data the shortest description.
- Kolmogorov complexity: shortest description of the data.
- Idea:
  - Model can be described using $L(M)$ bits.
  - Data can be described using $L(D \mid M)$ bits, when the model is known.
  - Total description length $L = L(M) + L(D \mid M)$ (approx. Kolmogorov complexity).
  - Occam's razor: prefer the shortest description/hypothesis, choose model with smallest L.
- The data could in principle be compressed to L bits.
- (In model selection we do not usually need explicit compression, just the description lengths.)

Minimum Description Length (MDL)

- MAP estimate finds a model that minimizes

$$-\mathcal{L} = -\log_2 p(\text{data} \mid \text{model}) - \log_2 p(\text{model})$$

- $-\log_2 p(\text{model})$: number of bits it takes to describe the model.
- $-\log_2 p(\text{data} \mid \text{model})$: number of bits it takes to describe the data, if the model is known.
- $-\mathcal{L}$: the description length of the data.
- MAP estimate can be seen as finding a shortest description of the data (that is, the best compression of the data).

Minimum Description Length (MDL)

Coding lengths

- Information theory: the optimal (shortest expected coding length) code for an event with probability p is $-\log_2 p$ bits.
- Example (Huffman coding; in model selection we do not usually need to construct the coding):
  - Let the probabilities of four letters be $P(A) = \frac{1}{2}$, $P(B) = \frac{1}{4}$, $P(C) = \frac{1}{8}$, $P(D) = \frac{1}{8}$.
  - Optimal coding: $A \rightarrow 0$, $B \rightarrow 10$, $C \rightarrow 110$, $D \rightarrow 111$.
  - For example, $ADAB$ would be coded as 0111010 (7 bits).
  - Expected coding length $L = \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{8} \times 3 = 1.75$ bits per number. "Compression ratio" $1.75/2 = 0.875$ as compared to the naive coding of each letter with 2 bits (e.g., $A = 00$, $B = 01$, $C = 10$, $D = 11$).

Minimum Description Length (MDL)

Coding lengths

- An integer in $\{0, \ldots, n\}$ can be expressed using $\log_2 (n+1)$ bits.
- Example: To express an integer in $\{0, \ldots, 15\}$ using binary numbers you need $\log_2 16 = 4$ bits.
- Usually we do not need to find explicit coding in model selection, knowing the coding length is enough.

Minimum Description Length (MDL)

Example: modeling binary sequence

- Data: an ordered sequence D of N binary numbers.
- Model 1: Code the sequence as such.
  - Coding length of the model $L(M_1) = 0$ bits.
  - Coding length of the data $L(D \mid M_1) = N$ bits.
  - Total coding length $L_1 = L(M_1) + L(D \mid M_1) = N$ bits.
- Model 2: Use the frequency of ones for better coding.
  - The model is the number of ones n_1 which is a integer in $[0, N]$. It can be expressed using $L(M_2) = \log_2(N+1)$ bits.
  - There are $\begin{pmatrix} N \\ n_1 \end{pmatrix}$ possible binary sequences of length N having n_1 ones. A sequence can be expressed using $L(D \mid M_2) = \log_2 \begin{pmatrix} N \\ n_1 \end{pmatrix}$ bits when n_1 is known.
  - Total coding length

$$L_2 = L(M_2) + L(D \mid M_2) = \log_2(N+1) + \log_2 \begin{pmatrix} N \\ n_1 \end{pmatrix} \text{ bits.}$$

Minimum Description Length (MDL)

Example: modeling binary sequence

- Example 1: $D = 0111010010$, $N = 10$.
  - $L_1 = 10$ bits. (Choose 1.)
  - $L_2 = \log_2(10+1) + \log_2 \binom{10}{5} = 3.4 + 8.0 = 11.4$ bits.
- Example 2: $D = 0001000010$, $N = 10$.
  - $L_1 = 10$ bits.
  - $L_2 = \log_2(10+1) + \log_2 \binom{10}{2} = 3.4 + 5.5 = 8.9$ bits.

  (Choose 2.)
- Example 3: $D = 0000000000$, $N = 10$.
  - $L_1 = 10$ bits.
  - $L_2 = \log_2(10+1) + \log_2 \binom{10}{0} = 3.4 + 0 = 3.4$ bits.

  (Choose 2.)

Structural Risk Minimization (SRM)

- According to the PAC theory, with probability $1 - \delta$,

$$E_{TEST} \leq E_{TRAIN} + \sqrt{\frac{\mathcal{VC}(H)\left(\log \frac{2N}{\mathcal{VC}(H)} + 1\right) - \log \frac{\delta}{4}}{N}},$$

  where N is the size of the training data, $\mathcal{VC}(H)$ is the VC-dimension of the hypothesis class and E_{TEST} is the expected error on new data and E_{TRAIN} is the error on the training set, respectively.
- SRM: Choose hypothesis class (for example, the degree of a polynomial) such that the bound on E_{TEST} is minimized.
- Often used to train the Support Vector Machines (SVM).
- (Vapnik (1995) contains more discussion of the SRM inductive principle; it won't be discussed in this course in more detail.)

Remainder of the lecture on the blackboard.
For slides see Alpaydin's site:
`http://www.cmpe.boun.edu.tr/~ethem/i2ml/slides/v1-1/i2ml-chap5-v1-1.pdf`