Lesson2: Model Selection and Evaluation

Course: Advanced Machine Learning

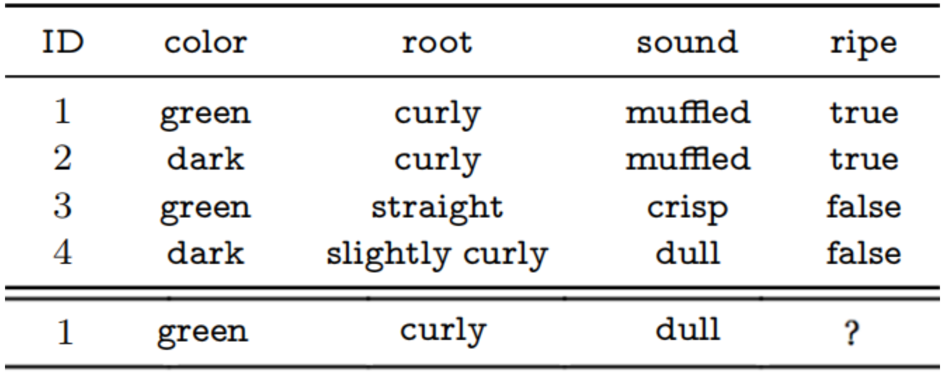
Instructors: Xianming Liu (csxm@hit.edu.cn)， Xiang Deng (dengxiang@hit.edu.cn)

Grading Policy:

The course will be graded based participation (10%), homework (40%), and a course project (50%).

review

- Terminology-Data:features, training data, test data
- Labeled or unlabeled information
  - Supervised learning: classification, regression
  - Unsupervised learning: clustering
  - Semi-supervised learning: a combination of the above two
- generalization ability
- Hypothesis Space
- Inductive Bias



ID	color	root	sound	ripe
1	green	curly	muffled	true
2	dark	curly	muffled	true
3	green	straight	crisp	false
4	dark	slightly curly	dull	false
1	green	curly	dull	?

Outline

- □ Empirical Error and Overfitting
- □ Evaluation Methods
- □ Performance Measure
- □ Comparison Test
- □ Bias and Variance
- □ Further Reading

Empirical Error and Overfitting

□ Error rate & Error：

- Error rate: proportion of incorrectly classified samples $E = a/m$
- Error: the difference between the output predicted by the learner and the ground-truth output
  - Training (empirical) error: on training set
  - Testing error: on testing set
  - Generalization error: the error calculated on the new samples

Since the details of the new samples are unknown during the training phase, we can only try to minimize the empirical error in practice.

Quite often, we obtain learners that perform well on the training set with a small or even zero empirical error, that is, 100% accuracy. However, are they the learners we need? Unfortunately, such learners are not good in most cases.

Empirical Error and Overfitting

□ Overfitting:

When the learner learns the training examples "too well", it is likely that some peculiarities of the training examples are taken as general properties that all potential samples will have, resulting in a reduction in generalization performance.

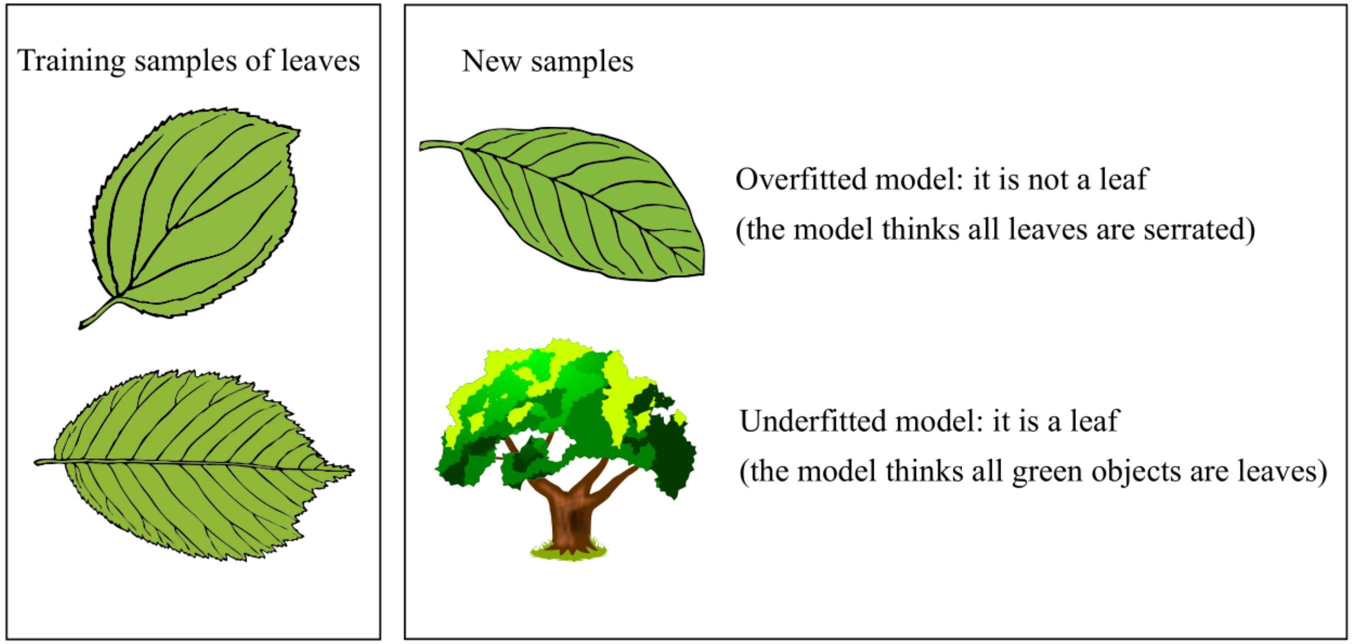
- Regularize the training objective.
- Early stop.

□ Underfitting:

The learner failed to learn the general properties of training examples.

- Do more branching in decision tree learning
- Adding more training epochs in neural network learning

Empirical Error and Overfitting


Training samples of leaves
New samples
Overfitted model: it is not a leaf
(the model thinks all leaves are serrated)
Underfitted model: it is a leaf
(the model thinks all green objects are leaves)

An intuitive analogy of overfitting and underfitting.

Overfitting: some peculiarities of the training examples are taken as general properties that all potential samples will have.

Underfitting: the learner failed to learn the general properties of training examples.

Outline

- □ Empirical Error and Overfitting
- □ Evaluation Methods
- □ Performance Measure
- □ Comparison Test
- □ Bias and Variance
- □ Further Reading

Evaluation Methods

Here, we only consider the generalization error, but in real-world applications, we often consider more factors such as computational cost, memory cost, and interpretability.

We assume that the testing samples are independent and identically sampled from the ground-truth sample distribution, and use the testing error as an approximation to the generalization error, thus the testing set and the training set should be mutually exclusive as much as possible.

Evaluation Methods

Given the only data set of m samples how can we do both training and testing? The answer is to produce both a training set S and a testing set T from the data set D.

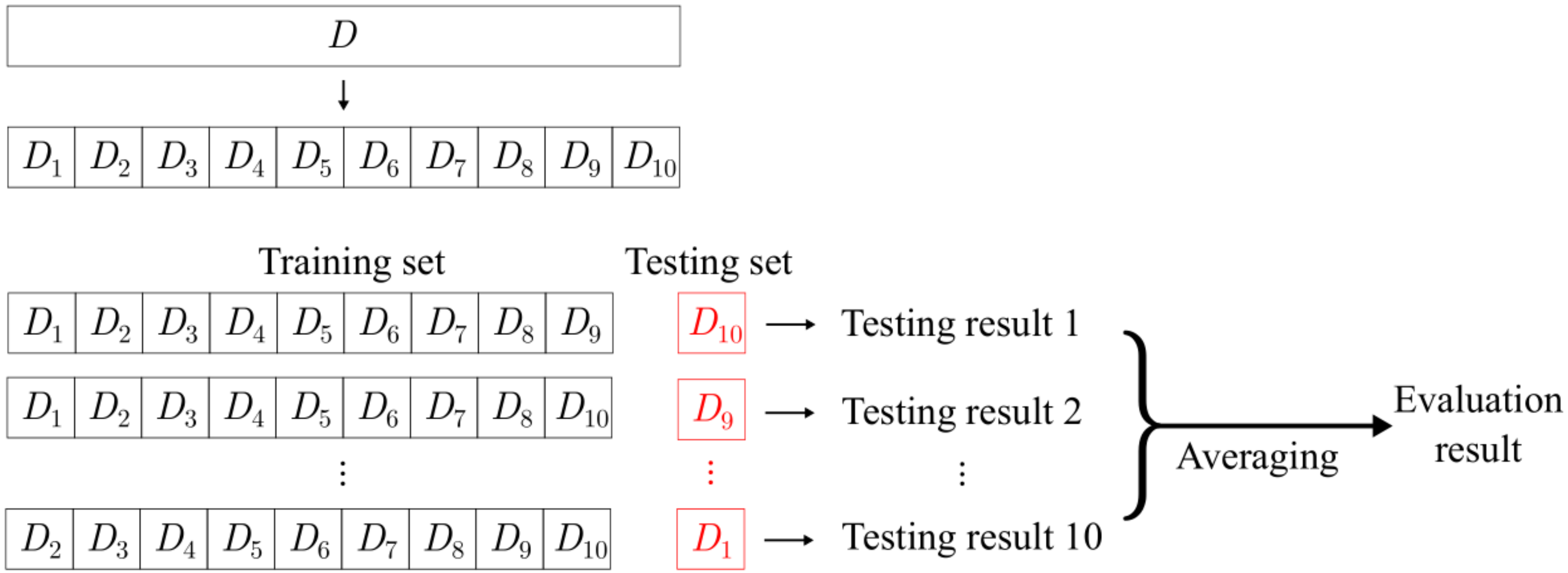
□ Hold-out：

- Splits the data set into two disjoint subsets
- The splitting should maintain the original data distribution to avoid introducing additional bias
- We often perform the hold-out testing multiple times, where each trial splits the data randomly, and we use the average error as the final estimation.
- One routine is to use around 2/3 to 4/5 of the examples for training and the rest for testing

Evaluation Methods

□ Cross-Validation:

Cross-validation splits data set D into k disjoint subsets with similar sizes, In each trial of cross-validation, we use the union of $k-1$ subsets as the training set to train a model and then use the remaining subset as the testing set to evaluate the model, We repeat this process k times and we average over k trials to obtain the evaluation result, The most commonly used value of k is 10.



D

D_1 D_2 D_3 D_4 D_5 D_6 D_7 D_8 D_9 D_{10}

Training set | Testing set

D_1 D_2 D_3 D_4 D_5 D_6 D_7 D_8 D_9 | D_{10} → Testing result 1

D_1 D_2 D_3 D_4 D_5 D_6 D_7 D_8 D_{10} | D_9 → Testing result 2

⋮ ⋮ ⋮

D_2 D_3 D_4 D_5 D_6 D_7 D_8 D_9 D_{10} | D_1 → Testing result 10

10-fold cross-validation

Evaluation Methods

Like hold-out, there are different ways of splitting the data set D into k subsets. To decrease the error introduced by splitting, we often repeat the random splitting p times and average the evaluation results of p times of k-fold cross-validation. For example, a common case is 10-time 10-fold cross-validation.

For a data set D with m samples, a special case of cross-validation is Leave-One-Out (LOO), which lets $k = m$:

- The random splitting does not matter
- The evaluation from LOO is very close to the ideal evaluation
- Computational cost of training m models could be prohibitive for large data sets

Evaluation Methods

□ Bootstrapping:

Given a data set D containing m samples, bootstrapping samples a data set D' by randomly picking one sample from D, copying it to D' , and then placing it back to D so that it still has a chance to be picked next time. Repeating this process m times results in the bootstrap sampling data set D' containing m samples. we can use D' as the training set and $D\backslash \mathrm{D}'$ as the testing set.

- Both the evaluated model and the actual model that we wish to evaluate on D are using m training examples
- Roughly 36.8% of the original samples do not appear in the training data.
- Bootstrapping can create multiple data sets, which can be useful for methods such as ensemble learning
- Bootstrapping is particularly useful when the data set is small, or when there is no effective way of splitting training and testing sets; since the original data distribution has changed by bootstrapping, the estimation is also biased. Therefore, when we have abundant data, hold-out and cross-validation are often used instead.

$$\lim_{m\mapsto\infty}\left(1-\frac{1}{m}\right)^m \mapsto \frac{1}{e} \approx 0.368$$

Outline

- Empirical Error and Overfitting
- Evaluation Methods
- Performance Measure
- Comparison Test
- Bias and Variance
- Further Reading

Performance Measure

Performance measures can quantify the generalization ability, Different performance measures reflect the varied demands of tasks and produce different evaluation results.

In prediction problems, we are given a data set $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$ where y_i is the ground-truth label of the sample x_i. To evaluate the performance of a learner f, we compare its prediction $f(x)$ to the ground-truth label y.

For regression problems, the most commonly used performance measure is the Mean Squared Error (MSE):

$$E(f; D) = \frac{1}{m} \sum_{i=1}^{m} \left(f\left(\boldsymbol{x}_i\right) - y_i\right)^2$$

Performance Measure

Error rate and accuracy are the most commonly used performance measures in classification problems：

- Error rate is the proportion of misclassified samples to all samples
- Accuracy is the proportion of correctly classified samples instead

Error rate

$$E(f;D) = \frac{1}{m}\sum_{i=1}^{m}\mathbb{I}\left(f\left(\boldsymbol{x}_i\right) \neq y_i\right)$$

Accuracy

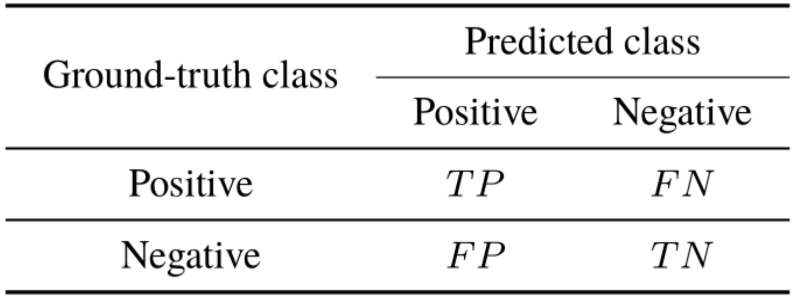
$$\begin{aligned}\operatorname{acc}(f;D) &= \frac{1}{m}\sum_{i=1}^{m}\mathbb{I}\left(f\left(\boldsymbol{x}_i\right) = y_i\right) \\ &= 1 - E(f;D)\ .\end{aligned}$$

Performance Measure

We often want to know “What percentage of the retrieved information is of interest to users?” and “How much of the information the user interested in is retrieved?” in applications like information retrieval and web search. For such questions, precision and recall are better choices.

In binary classification problems, there are four combinations of the ground-truth class and the predicted class, namely true positive, false positive, true negative, and false negative, The four combinations can be displayed in a confusion matrix.

The confusion matrix of binary classification



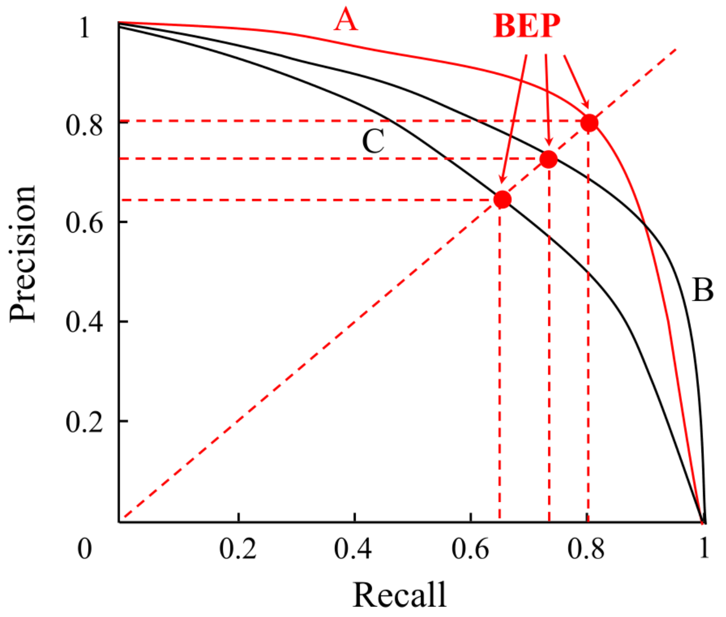
Ground-truth class	Predicted class	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Precision $$P = \frac{TP}{TP + FP}$$

Recall $$R = \frac{TP}{TP + FN}$$

Performance Measure

We can use the learner’s predictions to sort the samples by how likely they are positive. Starting from the top of the ranking list, we can incrementally label the samples as positive to calculate the precision and recall at each increment. Then, plotting the precisions as y-axis and the recalls as x-axis gives the Precision-Recall Curve (P-R curve).



P-R curve and break-even points

Break-Even Point (BEP) is the value when precision and recall are equal, which can be used to compare performance when the P-R curve intersects.

Performance Measure

$$
\begin{aligned}
F1 = \frac{2 \times P \times R}{P + R} &= \frac{2 \times \frac{TP}{TP+FP} \times \frac{TP}{TP+FN}}{\frac{TP}{TP+FP} + \frac{TP}{TP+FN}} \\
&= \frac{2 \times TP \times TP}{TP(TP+FN) + TP(TP+FP)} \\
&= \frac{2 \times TP}{(TP+FN) + (TP+FP)} \\
&= \frac{2 \times TP}{(TP+FN+FP+TN) + TP - TN} \\
&= \frac{2 \times TP}{\text{样例总数} + TP - TN}
\end{aligned}
$$

BEP could be oversimplified, and a more commonly used alternative is F1-measure:

$$F1 = \frac{2 \times P \times R}{P + R} = \frac{2 \times TP}{\text{total number of samples} + TP - TN}$$

harmonic mean of P and R

$$\frac{1}{F1} = \frac{1}{2} \cdot \left(\frac{1}{P} + \frac{1}{R} \right)$$

The general form of F1-measure is F_β:

$$F_\beta = \frac{(1 + \beta^2) \times P \times R}{(\beta^2 \times P) + R}$$

weighted harmonic mean of P and R

$$\frac{1}{F_\beta} = \frac{1}{1 + \beta^2} \cdot \left(\frac{1}{P} + \frac{\beta^2}{R} \right)$$

$\beta = 1$: Standard F1

$\beta > 1$: Recall is more important

$\beta < 1$: Precision is more important

Performance Measure

The ranking quality reflects the learner's "expected generalization ability" for different tasks or the generalization ability for "typical cases". The Receiver Operating Characteristics (ROC) curve follows this idea to measure the generalization ability of learners.

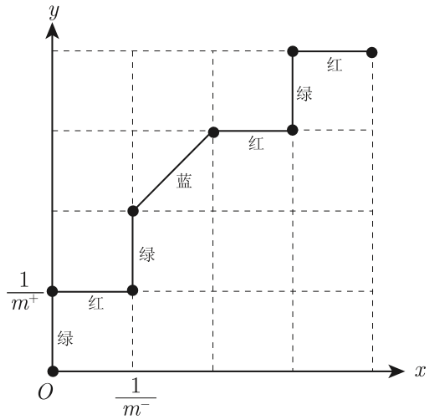
The plotting process is as follows: given m^+ positive samples and m^- negative samples, we first sort all samples by the learner's predictions, and then set the threshold to maximum, that is, predicting all samples as negative. At this moment, both TPR and FPR are 0, so we mark at coordinate $(0, 0)$. Then, we gradually decrease the threshold to the predicted value of each sample along the sorted list, that is, the samples are classified as positive successively. Let (x, y) denote the previous coordinate, we put a mark at $(x, y + \frac{1}{m^+})$ if the current samples is true positive, and we put a mark at $(x + \frac{1}{m^-}, y)$if the current samples is false positive. By connecting all adjacent marked points, we have the ROC curve.

Ground-truth class	Predicted class	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

$$\mathrm{TPR} = \frac{TP}{TP + FN} \,,$$

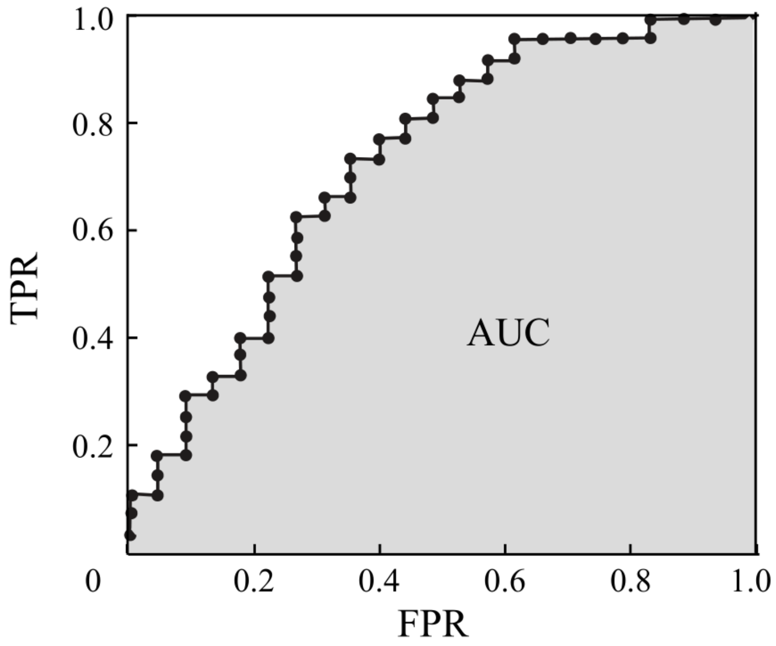
$$\mathrm{FPR} = \frac{FP}{TN + FP} \,.$$

$$(\boldsymbol{s}_1, 0.77, +), (\boldsymbol{s}_2, 0.62, -), (\boldsymbol{s}_3, 0.58, +), (\boldsymbol{s}_4, 0.47, +),$$
$$(\boldsymbol{s}_5, 0.47, -), (\boldsymbol{s}_6, 0.33, -), (\boldsymbol{s}_7, 0.23, +), (\boldsymbol{s}_8, 0.15, -)$$



Performance Measure

Learner A is better than learner B if A's ROC curve entirely encloses B's ROC curve; However, when there exist intersections, no learner is generally better than the other. One way of comparing intersected ROC curves is to calculate the areas under the ROC curves, that is, Area Under ROC Curve (AUC).



ROC curve and AUC with finite samples

Suppose that the ROC curve is obtained by sequentially connecting the points $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, where $x_1 = 0$ and $x_m = 1$. Then, the AUC is estimated as

$$\mathrm{AUC} = \frac{1}{2}\sum_{i=1}^{m-1}(x_{i+1} - x_i)\cdot(y_i + y_{i+1})$$

Cost-Sensitive Error Rate

In some problems, the consequences of making different errors are not the same, thus we need to assign unequal costs to different errors.

For binary classification problems, we can leverage domain knowledge to design a cost matrix. where $cost_{ij}$ represents the cost of misclassifying a sample of class i as class j. The larger the difference between the costs is, the larger the difference between $cost_{01}$ and $cost_{10}$ will be.

With unequal costs, however, we no longer minimize the counts but the total cost, the cost-sensitive error rate is defined as：

$$E(f;D;cost) = \frac{1}{m}\left(\sum_{\boldsymbol{x}_i \in D^+} \mathbb{I}\left(f\left(\boldsymbol{x}_i\right) \neq y_i\right) \times cost_{01} + \sum_{\boldsymbol{x}_i \in D^-} \mathbb{I}\left(f\left(\boldsymbol{x}_i\right) \neq y_i\right) \times cost_{10}\right)$$

Cost Curve

With unequal costs, we find the expected total costs of learners from cost curves rather than ROC curves.

0: + positive class
1: - negative class

The x-axis of cost curves is the probability cost of positive class:

$$P(+)cost = \frac{p \times cost_{01}}{p \times cost_{01} + (1-p) \times cost_{10}}$$

$$cost_{+-} = cost_{-+} \quad P(+)cost = \frac{p}{p + (1-p)} = p$$

Where $p \in [0,1]$ is the probability of a sample being positive.

if cost$_{+-}$=4;

$$P(+)cost = \frac{p \times cost_{+-}}{p \times cost_{+-} + (1-p) \times cost_{-+}}$$
$$= \frac{0.2 \times 4}{0.2 \times 4 + (1-0.2) \times 1} = 0.5$$

$$D = \{\boldsymbol{x}_1^+, \boldsymbol{x}_2^+, \boldsymbol{x}_3^-, \boldsymbol{x}_4^-, \boldsymbol{x}_5^-, \boldsymbol{x}_6^-, \boldsymbol{x}_7^-, \boldsymbol{x}_8^-, \boldsymbol{x}_9^-, \boldsymbol{x}_{10}^-\}$$ p=0.2

$$D' = \{\boldsymbol{x}_1^+, \boldsymbol{x}_1^+, \boldsymbol{x}_1^+, \boldsymbol{x}_1^+, \boldsymbol{x}_2^+, \boldsymbol{x}_2^+, \boldsymbol{x}_2^+, \boldsymbol{x}_2^+, \boldsymbol{x}_3^-, \boldsymbol{x}_4^-, \boldsymbol{x}_5^-, \boldsymbol{x}_6^-, \boldsymbol{x}_7^-, \boldsymbol{x}_8^-, \boldsymbol{x}_9^-, \boldsymbol{x}_{10}^-\}$$

Cost Curve

The y-axis is the normalized cost which takes values from $[0, 1]$:

0: + positive class
1: - negative class

$$cost_{norm} = \frac{\mathrm{FNR} \times p \times cost_{01} + \mathrm{FPR} \times (1-p) \times cost_{10}}{p \times cost_{01} + (1-p) \times cost_{10}}$$

$$\begin{aligned} cost_{se} &= m \times p \times \mathrm{FNR} \times cost_{+-} + m \times (1-p) \times \mathrm{FPR} \times cost_{-+} \\ &+ m \times p \times \mathrm{TPR} \times cost_{++} + m \times (1-p) \times \mathrm{TNR} \times cost_{--} \end{aligned}$$

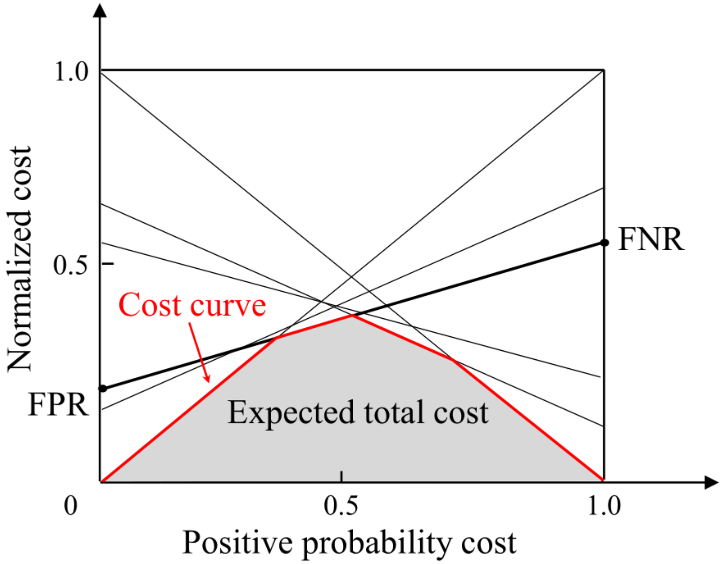
$$cost_{se} = p \times \mathrm{FNR} \times cost_{+-} + (1-p) \times \mathrm{FPR} \times cost_{-+}$$

$$max(cost_{se}) = p \times cost_{+-} + (1-p) \times cost_{-+}$$

$$\frac{cost_{se}}{max(cost_{se})} = \frac{p \times \mathrm{FNR} \times cost_{+-} + (1-p) \times \mathrm{FPR} \times cost_{-+}}{p \times cost_{+-} + (1-p) \times cost_{-+}}$$

Cost Curve

We can draw a cost curve as follows: since every point (FPR, TPR) on the ROC curve corresponds to a line segment on the cost plane, we can calculate the FNR and draw a line segment from (0, FPR) to (1, FNR). Then, the area under the line segment represents the expected total cost for the given p, FPR, and TPR. By converting all points on the ROC curve to line segments on the cost plane, the expected total cost is given by the area under the lower bound of all line segments, as shown in the following figure:



The cost curve and expected total cost

$$P(+)cost = \frac{p \times cost_{01}}{p \times cost_{01} + (1-p) \times cost_{10}}$$

$$cost_{norm} = \frac{\mathrm{FNR} \times p \times cost_{01} + \mathrm{FPR} \times (1-p) \times cost_{10}}{p \times cost_{01} + (1-p) \times cost_{10}}$$

$$cost_{norm} = \mathrm{FNR} \times P(+)cost + \mathrm{FPR} \times (1 - P(+)cost)$$

Outline

- □ Empirical Error and Overfitting
- □ Evaluation Methods
- □ Performance Measure
- □ Comparison Test
- □ Bias and Variance
- □ Further Reading

Performance Comparisons

□ About Performance Comparisons：

- The testing performance may not reflect the actual generalization performance.
- Testing performance depends on the choice of the testing set.
- Many machine learning algorithms have some build-in random behavior.

Hypothesis testing is one of the techniques to compare the performance of learners. Suppose that we observe learner A outperforms learner B on a testing set. Then, hypothesis testing can help us check whether the generalization performance of learner A is better than that of learner B in the statistical sense and how significant it is.

Hypothesis Testing

A generalization error rate of ϵ means that the learner has a probability of ϵ to make an incorrect prediction. A testing error rate of $\hat{\epsilon}$ means that the learner misclassified $\hat{\epsilon} \times m$ samples in a testing set of m samples, We can use binomial test to verify hypotheses such as "$\epsilon \leqslant 0.3$".

For the hypothesis"$\epsilon \leqslant \epsilon_0$", the following equation gives the maximum observable error rate within a probability of $1 - \alpha$:

$$\bar{\epsilon} = \min \epsilon \text{ s.t. } \sum_{i=\epsilon_0 \times m+1}^{m} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} < \alpha$$

If the testing error rate $\hat{\epsilon}$ is less than the critical value ϵ, then, according to the binomial test, the hypothesis "$\epsilon \leqslant \epsilon_0$" cannot be rejected at the significance level of α, that is, the learner's generalization error rate is not greater than ϵ_0 at the confidence level of $1 - \alpha$.

t-test

We often obtain multiple testing error rates from cross-validation or by doing multiple hold-out evaluations. In such cases, we can use t-test.

Let $\hat{\epsilon}_1, \hat{\epsilon}_2, \ldots, \hat{\epsilon}_k$ denote the k testing error rates. For the hypothesis "$\mu = \epsilon_0$" and significance level α, Let $(-\infty, t_{-\alpha/2}]$ and $[t_{\alpha/2}, \infty)$ denote the ranges of the two shaded areas, respectively. If τ_t is within the critical value range $[t_{-\alpha/2}, t_{\alpha/2}]$, then the hypothesis "$\mu = \epsilon_0$" cannot be rejected, that is, the generalization error rate is ϵ_0 at the confidence level of $1 - \alpha$.

Outline

- Empirical Error and Overfitting
- Evaluation Methods
- Performance Measure
- Comparison Test
- Bias and Variance
- Further Reading

Bias and Variance

In addition to estimating the generalization performance of learning algorithms, people often wish to understand "why" learning algorithms have such performance. An essential tool for understanding the generalization performance of algorithms is the bias-variance decomposition, which decomposes the expected generalization error of learning algorithms.

Let x be a testing sample, y_D be the label of x in the data set D, y be the ground-truth label of x, and $f(x;D)$ be the output of x predicted by the model f trained on D. Then, in regression problems, the expected prediction of a learning algorithm is

$$\bar{f}(\boldsymbol{x}) = \mathbb{E}_D[f(\boldsymbol{x};D)]$$

The variance of using different equal-sized training sets is

$$var(\boldsymbol{x}) = \mathbb{E}_D\left[\left(f\left(\boldsymbol{x};D\right) - \bar{f}\left(\boldsymbol{x}\right)\right)^2\right]$$

The noise is

$$\varepsilon^2 = \mathbb{E}_D\left[\left(y_D - y\right)^2\right]$$

Bias and Variance

The difference between the expected output and the ground-truth label is called bias, that is $bias^2(\boldsymbol{x}) = \left(\bar{f}(\boldsymbol{x}) - y\right)^2$

For ease of discussion, we assume the expectation of noise is zero,i.e. $\mathbb{E}_D[y_D - y] = 0$

we can decompose the expected generalization error as follows:

$$\begin{aligned}
E(f;D) &= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - y_D\right)^2\right] \\
&= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x}) + \bar{f}(\boldsymbol{x}) - y_D\right)^2\right] \\
&= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x})\right)^2\right] + \mathbb{E}_D\left[\left(\bar{f}(\boldsymbol{x}) - y_D\right)^2\right] \\
&\quad + \mathbb{E}_D\left[2\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x})\right)\left(\bar{f}(\boldsymbol{x}) - y_D\right)\right] \\
&= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x})\right)^2\right] + \mathbb{E}_D\left[\left(\bar{f}(\boldsymbol{x}) - y_D\right)^2\right]
\end{aligned}$$

Bias and Variance

explanation from line 3-4 to line 5:

$$\mathbb{E}_D\left[2\left(f(\boldsymbol{x};D)-\bar{f}(\boldsymbol{x})\right)\left(\bar{f}(\boldsymbol{x})-y_D\right)\right]=\mathbb{E}_D\left[2\left(f(\boldsymbol{x};D)-\bar{f}(\boldsymbol{x})\right)\cdot\bar{f}(\boldsymbol{x})\right]-\mathbb{E}_D\left[2\left(f(\boldsymbol{x};D)-\bar{f}(\boldsymbol{x})\right)\cdot y_D\right]$$

$$\begin{aligned}\mathbb{E}_D\left[2\left(f(\boldsymbol{x};D)-\bar{f}(\boldsymbol{x})\right)\cdot\bar{f}(\boldsymbol{x})\right]&=\mathbb{E}_D\left[2f(\boldsymbol{x};D)\cdot\bar{f}(\boldsymbol{x})-2\bar{f}(\boldsymbol{x})\cdot\bar{f}(\boldsymbol{x})\right]\\&=2\bar{f}(\boldsymbol{x})\cdot\mathbb{E}_D\left[f(\boldsymbol{x};D)\right]-2\bar{f}(\boldsymbol{x})\cdot\bar{f}(\boldsymbol{x})\\&=2\bar{f}(\boldsymbol{x})\cdot\bar{f}(\boldsymbol{x})-2\bar{f}(\boldsymbol{x})\cdot\bar{f}(\boldsymbol{x})=0\end{aligned}$$

$$\begin{aligned}\mathbb{E}_D\left[2\left(f(\boldsymbol{x};D)-\bar{f}(\boldsymbol{x})\right)\cdot y_D\right]&=2\mathbb{E}_D\left[f(\boldsymbol{x};D)\cdot y_D\right]-2\bar{f}(\boldsymbol{x})\cdot\mathbb{E}_D\left[y_D\right]\\&=2\mathbb{E}_D\left[f(\boldsymbol{x};D)\right]\cdot\mathbb{E}_D\left[y_D\right]-2\bar{f}(\boldsymbol{x})\cdot\mathbb{E}_D\left[y_D\right]\\&=2\bar{f}(\boldsymbol{x})\cdot\mathbb{E}_D\left[y_D\right]-2\bar{f}(\boldsymbol{x})\cdot\mathbb{E}_D\left[y_D\right]\\&=0\end{aligned}$$

$$\bar{f}(\boldsymbol{x})=\mathbb{E}_D[f(\boldsymbol{x};D)]$$

Bias and Variance

$$= \mathbb{E}_D\left[\left(f\left(\boldsymbol{x};D\right) - \bar{f}\left(\boldsymbol{x}\right)\right)^2\right] + \mathbb{E}_D\left[\left(\bar{f}\left(\boldsymbol{x}\right) - y + y - y_D\right)^2\right]$$

$$= \mathbb{E}_D\left[\left(f\left(\boldsymbol{x};D\right) - \bar{f}\left(\boldsymbol{x}\right)\right)^2\right] + \mathbb{E}_D\left[\left(\bar{f}\left(\boldsymbol{x}\right) - y\right)^2\right] + \mathbb{E}_D\left[\left(y - y_D\right)^2\right]$$

$$+ 2\mathbb{E}_D\left[\left(\bar{f}\left(\boldsymbol{x}\right) - y\right)\left(y - y_D\right)\right]$$

Since the expectation of noise is 0, $\mathbb{E}_D\left[\left(y - y_D\right)\right] = 0$

$$E(f;D) = \mathbb{E}_D\left[\left(f\left(\boldsymbol{x};D\right) - \bar{f}\left(\boldsymbol{x}\right)\right)^2\right] + \left(\bar{f}\left(\boldsymbol{x}\right) - y\right)^2 + \mathbb{E}_D\left[\left(y_D - y\right)^2\right]$$

That is

$$E(f;D) = bias^2\left(\boldsymbol{x}\right) + var\left(\boldsymbol{x}\right) + \varepsilon^2$$

which means the generalization error can be decomposed into the sum of bias, variance, and noise.

Bias and Variance

- Bias measures the difference between the learning algorithm's expected prediction and the ground-truth label; that is, expressing the fitting ability of the learning algorithm;
- Variance measures the change of learning performance caused by changes to the equal-sized training set, that is, expressing the impact of data disturbance on the learning outcome;
- Noise represents the lower bound of the expected generalization error that can be achieved by any learning algorithms for the given task, that is, the inherent difficulty of the learning problem.

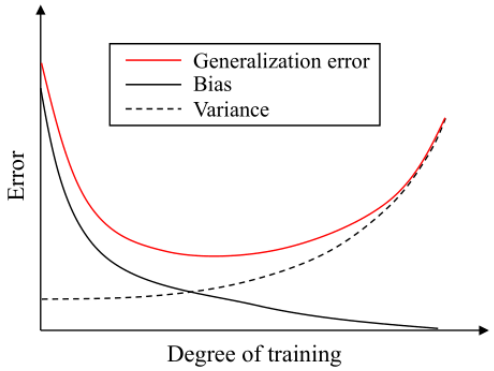
The bias-variance decomposition tells us that the generalization performance is jointly determined by the learning algorithm's ability, data sufficiency, and the inherent difficulty of the learning problem. In order to achieve excellent generalization performance, a small bias is needed by adequately fitting the data, and the variance should also be kept small by minimizing the impact of data disturbance.

Bias and Variance

Generally speaking, bias and variance are conflicted with each other, and this is known as the bias-variance dilemma.

As depicted in this figure, suppose we can control the degree of training:

- If the learner is undertrained, its fitting ability is limited, and hence the data disturbances have a limited impact on the learner, that is, bias dominates the generalization error;
- As the training proceeds, the learner's fitting ability improves, thus variance starts to dominate the generalization error;
- After a large amount of training, the fitting ability of the learner becomes very strong, and hence slight disturbances in the training data will cause significant changes to the learner. At this point, the learner may start to learn the peculiarities of the training data, and hence overfitting occurs.



Relationships between generalization error, bias, and variance.

Outline

- □ Empirical Error and Overfitting
- □ Evaluation Methods
- □ Performance Measure
- □ Comparison Test
- □ Bias and Variance
- □ Further Reading

Further Reading

- Bootstrap sampling has crucial applications in machine learning, and a detailed discussion can be found in [Efron and Tibshirani, 1993].

- ROC curve was introduced to machine learning in the late 1980s[Spackman, 1989], and AUC started to be widely used in the field of machine learning since the middle 1990s [Bradley,1997].[Hand and Till,2001] extended the ROC curve from binary classification problems to multiclass classification problems.[Fawcett,2006] surveyed the use of the ROC curve.

- [Drummond and Holte,2006] invented the cost curve. Cost-sensitive learning [Elkan,2001;Zhou and Liu,2006] is a research topic for learning under unequal cost settings.

Further Reading

□ [Dietterich,1998] pointed out the risk of using the regular k-fold cross-validation method, and proposed the 5 × 2 cross-validation method. [Demsar, 2006] discussed the hypothesis testing methods for comparing multiple algorithms.

□ [Geman et al.,1992] proposed the bias-variance-covariance decomposition for regression problems, which was later shortened as bias-variance decomposition. For classification problems, however, deriving the bias-variance decomposition is difficult since the 0/1 loss function is discontinuous. There exist many empirical methods for estimating bias and variance [Kong and Dietterich,1995;Kohavi and Wolpert, 1996; Breiman, 1996; Friedman,1997; Domingos,2000].

summary

- Empirical Error and Overfitting: overfitting, underfitting
- Evaluation methods: generalization error, Hold-out, Cross-Validation, Bootstrapping
- Performance Measure: Mean Squared Error, Error rate, accuracy, Precision, recall, f1, P-R curve, break-even points, ROC curve and AUC, cost-sensitive error rate
- Performance Comparisons: Hypothesis testing
- Bias and Variance

Thanks!