Hypothesis Evaluation

- Two definitions of error

. The true error of hypothesis h with respect to target function f and distribution D is the probability that h will misclassify an instance drawn at random according to D :

$$error_D(h) \equiv \Pr_{x \in D}[f(x) \neq h(x)]$$

. The sample error of h with respect to target function f and data sample S is proportion of examples h misclassifies

$$error_S(h) \equiv \frac{1}{n} \sum_{x \in S} I(f(x) \neq h(x))$$

where $I(f(x) \neq h(x))$ is 1 if $f(x) \neq h(x)$, and 0 otherwise.

- Problems of estimating error

. $error_S(h)$ is an estimator of $error_D(h)$.

. How well does $error_S(h)$ estimate $error_D(h)$?

. bias of $error_S(h)$ as an estimator of $error_D(h)$:

$$b_{error_D}(error_s) = E[error_s] - error_D$$

if $b_{error_D}(error_s) = 0$ for all $error_D$, we say $error_s$ is an unbiased estimator of $error_D$.

. The mean square error of $error_s$ is given as follows:

$$\begin{aligned} E[(error_s - error_D)^2] &= E[(error_s - E[error_s] + E[error_s] - error_D)^2] \\ &= E[(error_s - E[error_s])^2] + E[(E[error_s] - error_D)^2] + \\ &\quad 2E[(E[error_s] - error_D](error_s - E[error_s])] \\ &= E[(error_s - E[error_s])^2] + (E[error_s] - error_D)^2 \\ &= Var(error_s) + b^2_{error_D}(error_s) \end{aligned}$$

That is, the mean square error of $error_s$ is equivalent to the variance of $error_s$ plus the square of bias of $error_s$.

. Let $X_i \in \{0,1\}$ be a random variable which has the mean $error_D$, that is, $E[X_i] = error_D$. Here, we assume that X_is are

independent and identically distributed.
Then, $error_s$ can be described by

$$error_S = \frac{1}{N}\sum_{i=1}^{N} X_i$$

where N represents the total number of trials.

In this case,

$$E[error_S] = E[\frac{1}{N}\sum_{i=1}^{N} X_i] = \frac{1}{N}\sum_{i=1}^{N} E[X_i] = error_D.$$

That is, $error_S$ is an unbiased estimator of $error_D$.

. example:

Hypothesis h misclassifies 50 of the 100 samples in S.

In this case,

$$error_S(h) = \frac{50}{100} = 0.50.$$

Then, what is $error_D(h)$?

. Given observed $error_S(h)$ what can we conclude about $error_D(h)$?

- Binomial probability distribution

. Let X be a binomial random variable with parameters (n, p). Then, X represents the number of successes in n trials and p represents the probability of success.

. example: tossing a coin.

Probability $\Pr(r)$ of r heads in n coin flips can be described by

$$\Pr(r) = \binom{n}{r} p^r (1-p)^{n-r} = \frac{n!}{r!(n-r)!} p^r (1-p)^{n-r}$$

where $p = \Pr(head)$.

In this case, the mean value of X is

$$E[X] = \sum_{i=0}^{n} i \Pr(i) = np \quad \text{and}$$

the variance of X is

$$Var(X) = E[(X - E[X])^2] = np(1-p).$$

. $error_S(h)$ follows a binomial distribution, that is,

$$error_S(h) = \frac{X}{n},$$

$$E[error_S] = E[\frac{X}{n}] = \frac{1}{n} E[X] = p = error_D, \text{ and}$$

$$Var(error_S) = Var(\frac{X}{n}) = \frac{1}{n^2} Var(X) = \frac{p(1-p)}{n} = \frac{error_D(1-error_D)}{n}.$$

- Normal distribution approximates Binomial

. Let X_i be a random variable which has the value of 0 or 1 and $\Pr[X_i = 1] = p$.

Then, the random variable X having binomial distribution with parameters (n, p) can be described by

$$X = \sum_{i=1}^{n} X_i.$$

Here, the mean of X_i is

$$E[X_i] = 1 \cdot p + 0 \cdot (1-p) = p \quad \text{and}$$

the variance of X_i is

$$Var(X_i) = E[X_i^2] - E^2[X_i] = p - p^2 = p(1-p).$$

. Central Limit Theorem:

Consider a set of independent, identically distributed (i. i. d.) random variables $X_1, X_2, \cdots, X_n$ all governed by an arbitrary probability distribution with mean μ and finite variance σ^2. Let us define a new random vector

$$X = \sum_{i=1}^{n} X_i.$$

Then, as n goes to infinity, the distribution governing X approaches a normal (or Gaussian) distribution, with mean $n\mu$ and variance $n\sigma^2$. That is,

$$X \sim N(n\mu, n\sigma^2).$$

cf. In the case of Bernoulli trial, $X \dot{\sim} N(n\mu, n\sigma^2)$ when $n \geqq 30$. That is, X has an approximately Normal distribution with mean $n\mu$ and variance $n\sigma^2$. Here, the sample error of h can be described by

$$error_S(h) = \frac{X}{n} \dot{\sim} N(\mu, \frac{\sigma^2}{n})$$

where

$$\mu = error_D(h) \quad \text{and}$$

$$\frac{\sigma^2}{n} = \frac{error_D(1 - error_D)}{n} \approx \frac{error_S(1 - error_S)}{n}.$$

- Normal distribution

. The probability density function is given by

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}.$$

. The mean value of X: $E[X] = \mu$.

. The variance of X: $Var(X) = \sigma^2$

. The standard deviation of X: $\sigma_X = \sigma$.

- Calculating confidence intervals

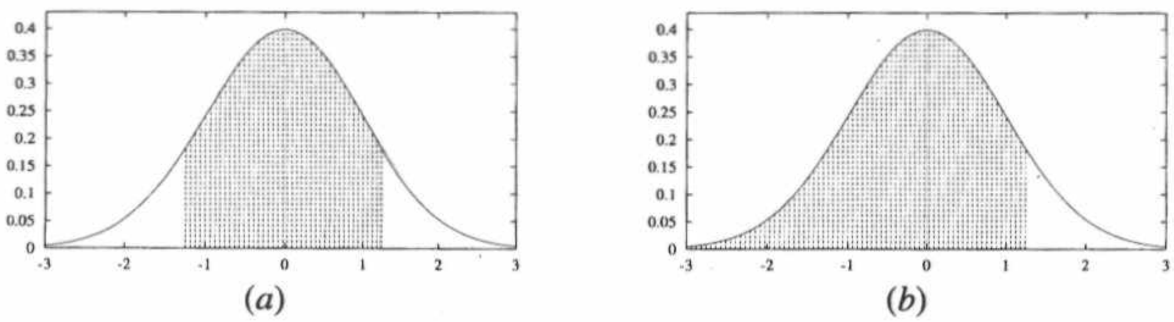


FIGURE 5.1
A Normal distribution with mean 0, standard deviation 1. (*a*) With 80% confidence, the value of the random variable will lie in the two-sided interval [−1.28, 1.28]. Note $z_{.80} = 1.28$. With 10% confidence it will lie to the right of this interval, and with 10% confidence it will lie to the left. (*b*) With 90% confidence, it will lie in the one-sided interval [−∞, 1.28].

. a $100(1-\alpha)\%$ two-sided confidence interval for μ: $\hat{\mu}\pm z_{\alpha/2}\sigma$.

Values of $z_{\alpha/2}$ for two-sided confidence intervals:

$\%$	50%	68%	80%	90%	95%	98%	99%
$z_{\alpha/2}$	0.67	1.00	1.28	1.64	1.96	2.33	2.58

eg. a 95% two-sided confidence interval for μ: $\hat{\mu}\pm 1.96\sigma$.

Let $\hat{\mu}$ is an estimator of μ and

$$\hat{\mu}=\frac{1}{n}\sum_{i=1}^{n}X_i$$

where X_is are i. i. d. random variables having mean $\mu=p$ and variance $\sigma^2=p(1-p)$. Then,

$$\hat{\mu}\dot{\sim}N(\mu,\frac{\sigma^2}{n}).$$

Let us make a unit (or standard) normal distribution of $\hat{\mu}$:

$$\frac{\hat{\mu}-\mu}{\sigma/\sqrt{n}}\dot{\sim}N(0,1).$$

This implies that

$$-1.96 < \frac{\hat{\mu} - \mu}{\sigma/\sqrt{n}} < 1.96$$ with the probability of 0.95.

Due to the symmetry of normal distribution,

$$-1.96 < \frac{\mu - \hat{\mu}}{\sigma/\sqrt{n}} < 1.96.$$

Therefore, we get

$$\hat{\mu} - 1.96\frac{\sigma}{\sqrt{n}} < \mu < \hat{\mu} + 1.96\frac{\sigma}{\sqrt{n}}$$

where $\sigma = \sqrt{p(1-p)}$.

-> True mean μ lies in $\hat{\mu} \pm 1.96\frac{\sigma}{\sqrt{n}}$ with the probability of 0.95.

In general, if $\hat{\mu} \sim N(\mu, \sigma^2)$,

the $N\%$ confidence interval of $\hat{\mu}$: $\hat{\mu} \pm z_N\sigma$

-> With $N\%$ probability, μ lies in interval $\hat{\mu} \pm z_N\sigma$.

The sample error is given by

$$error_S(h) = \frac{X}{n} \dot{\sim} N(\mu, \frac{\sigma^2}{n})$$

where

$\mu = error_D(h)$ and

$$\frac{\sigma^2}{n} = \frac{error_D(1 - error_D)}{n} \approx \frac{error_S(1 - error_S)}{n}.$$

With approximately 95% probability, $error_D(h)$ lies in interval

$$error_S(h) \pm 1.96\sqrt{\frac{error_S(h)(1-error_S(h))}{n}}.$$

example.
Hypothesis h misclassifies 50 of the 100 samples in S.

In this case,

$error_S(h) = \frac{50}{100} = 0.50$ and

$Var(error_S(h)) = \frac{0.5 \cdot 0.5}{100}$.

Then, with approximately 95% probability, $error_D(h)$ lies in interval

$$0.50 \pm 1.96\sqrt{\frac{0.50 \cdot 0.50}{100}} = 0.50 \pm 0.098.$$

That is, the 95% confidence interval of $error_S(h)$ is

0.50 ± 0.098.

- Comparing two hypotheses

. Problem: What is the probability that

$$error_D(h_1) > error_D(h_2)?$$

. Let

$$d \equiv error_D(h_1) - error_D(h_2)$$

and an estimator of d

$$\hat{d} \equiv error_{S_1}(h_1) - error_{S_2}(h_2).$$

If $error_{S_i}(h_i)$, $i = 1, 2$ are unbiased estimators,

$$E[\hat{d}] = d.$$

. Variance of $\hat{d}$:

$$Var(\hat{d}) = Var(error_{S_1}(h_1)) + Var(error_{S_2}(h_2))$$

assuming $error_{S_1}(h_1)$ and $error_{S_2}(h_2)$ are independent each other.

From the previous results,

$$Var(error_{S_1}(h_1)) \approx \frac{error_{S_1}(h_1)(1 - error_{S_1}(h_1))}{n_1} \text{ and}$$

$$Var(error_{S_2}(h_2)) \approx \frac{error_{S_2}(h_2)(1 - error_{S_2}(h_2))}{n_2}.$$

Therefore,

$$Var(\hat{d}) \approx \frac{error_{S_1}(h_1)(1 - error_{S_1}(h_1))}{n_1} + \frac{error_{S_2}(h_2)(1 - error_{S_2}(h_2))}{n_2}.$$

Example:

What is the probability that $d = error_D(h_2) - error_D(h_1) > 0$ when $error_{S_1}(h_1) = 0.2$ and $error_{S_2}(h_2) = 0.3$ using two sample sets of 100 instances?

Let $\hat{d} = error_{S_2}(h_2) - error_{S_1}(h_1)$. Then,

$$\mu_{\hat{d}} = 0.3 - 0.2 = 0.1 \quad \text{and}$$

$$\sigma_{\hat{d}} = \sqrt{Var(\hat{d})} = \sqrt{\frac{0.2 \cdot 0.8}{100} + \frac{0.3 \cdot 0.7}{100}} = 0.0608.$$

Since

$$Z = \frac{\hat{d} - \mu_{\hat{d}}}{\sigma_{\hat{d}}} = \frac{\hat{d} - 0.1}{0.0608} \sim N(0,1) \text{ and } \hat{d} > 0,$$

$$Z > \frac{-0.1}{0.0608} = -1.644 \text{ or } Z < \frac{0.1}{0.0608} = 1.644.$$

From the definition of z quantity; that is, $P(Z > z_\alpha) = \alpha$,

$$z_{0.05} = 1.644.$$

Therefore,

$$P(Z < 1.644) = 1 - P(Z \geqq 1.644) = 1 - 0.05 = 0.95;$$

that is, h_1 is better than h_2 with 95% confidence.

- k-fold cross-validation

. Evaluation of learning algorithms
. Partition the available data into k disjoint subsets.
. k-1 disjoint sets are used to training samples and
the remaining 1 disjoint set is used to test samples.
. Usually, k is set to 10.

k-fold cross-validation method

Step 1. Partition the available data D_0 into k disjoint subsets
$T_1, T_2, \cdots, T_k$ of equal size, where this size is at least 30.

Step 2. For i from 1 to k, do
use T_i for the test set, and the remaining data for
training set S_i:
(1) $S_i \leftarrow \{D_0 - T_i\}$
(2) $h_i \leftarrow L(S_i)$
(3) Evaluate $error_{T_i}(h_i)$.

Step 3. Evaluate the error mean $\hat{\mu}$ and standard deviation s:

$$\hat{\mu} = \frac{1}{k}\sum_{i=1}^{k} error_{T_i}(h_i)$$

$$s = \sqrt{\frac{1}{k-1}\sum_{i=1}^{k}(error_{T_i}(h_i) - \hat{\mu})^2}$$

What is the relationship between $\hat{\mu}$ and μ?

- t-distribution

. If Z and χ_n^2 are independent random variables, with Z having standard normal distribution and χ_n^2 having a chi-square distribution with n degrees of freedom, then the random variable T_n defined by

$$T_n = \frac{Z}{\sqrt{\chi_n^2 / n}}$$

is said to have a t-distribution with n degrees of freedom.

. The t-density is symmetric about zero.
If n becomes larger, it becomes more and more like a standard normal density since

$$E[\chi_n^2/n] = E[\sum_{i=1}^{n} Z_i^2/n] \approx E[Z_i^2] = 1.$$

. The mean and variance of T_n:

$$E[T_n] = 0, \quad n > 1$$

$$Var(T_n) = \frac{n}{n-2}, \quad n > 2$$

Thus the variance of T_n decreases to 1 as n increases to ∞.

- t-Test

. From the result of k-fold cross-validation method,

$$\frac{\hat{\mu} - \mu}{s/\sqrt{k}} \sim T_{k-1}.$$

. This implies that with the probability of $1-\alpha$,

$$\hat{\mu} - t_{\alpha/2,k-1}\frac{s}{\sqrt{k}} < \mu < \hat{\mu} + t_{\alpha/2,k-1}\frac{s}{\sqrt{k}}$$

where $t_{\alpha/2,k-1}$ represents a constant such that

$$\Pr[T_{k-1} \geq t_{\alpha/2,k-1}] = \alpha/2.$$

Values of $t_{\alpha/2,n}$:

	$\alpha=0.1$	$\alpha=0.05$	$\alpha=0.02$	$\alpha=0.01$
$n=2$	2.92	4.30	6.96	9.92
$n=5$	2.02	2.57	3.36	4.03
$n=10$	1.81	2.23	2.76	3.17
$n=20$	1.72	2.09	2.53	2.84
$n=30$	1.70	2.04	2.46	2.75
$n=120$	1.66	1.98	2.36	2.62
$n=\infty$	1.64	1.96	2.33	2.58

Note that n=k-1.

Example: k-fold cross-validation method

11 subsets and each subset has 30 instances.

After measuring the performance of learning algorithm using the k-fold cross-validation method, we get

$\hat{\mu}=0.1$ and $s=0.01$.

In this case, k=11. Let $\alpha=0.05$. Then, $t_{0.025,10}=2.23$.

Then, with the probability of 0.95,

$0.1-2.23\cdot 0.01<\mu<0.1+2.23\cdot 0.01$, that is,

$0.0819<\mu<0.1181$.

- Comparing two learning algorithms

. What we would like to estimate is

$$E_{S \subset D}[error_D(L_A(S)) - error_D(L_B(S))]$$

where $L(S)$ is the hypothesis output by the learning algorithm L using training set S.

That is, the expected difference in true error between hypotheses output by learning algorithms L_A and L_B when trained using randomly selected training sets S drawn according to distribution D.

. But given limited data D_0 what is a good estimator?

(1) We could partition D_0 into training set S and test set T_0, and measure

$$error_{T_0}(L_A(S_0)) - error_{T_0}(L_B(S_0)).$$

(2) Even better, repeat this many times and average the results. That is, apply the k-fold cross-validation method.

k-fold cross-validation method

Step 1. Partition the available data D_0 into k disjoint subsets

$T_1, T_2, \cdots, T_k$ of equal size, where this size is at least 30.

Step 2. For i from 1 to k, do

use T_i for the test set, and the remaining data for training set S_i:

(1) $S_i \leftarrow \{D_0 - T_i\}$

(2) $h_A \leftarrow L_A(S_i)$

(3) $h_B \leftarrow L_B(S_i)$

(4) $\delta_i \leftarrow error_{T_i}(h_A) - error_{T_i}(h_B)$

Step 3. Return the average value of δ_i:

$$\bar{\delta} \equiv \frac{1}{k}\sum_{i=1}^{k}\delta_i.$$

. From the t-distribution, the approximate $(1-\alpha)\times 100$% confidence interval for δ is

$$\bar{\delta} \pm t_{\delta/2, k-1}\frac{s_\delta}{\sqrt{k}}$$

where

$$s_\delta = \sqrt{\frac{1}{k-1}\sum_{i=1}^{k}(\delta_i - \bar{\delta})^2}.$$

. k-fold cross-validation method comments

(1) Every example gets used as a test example.
(2) Every test set is independent.
(3) Training sets overlap significantly.
(4) 10 is a standard number of folds, that is, k=10.
(5) No method for comparing learning systems with limited data is perfect. However, some statistical analysis is preferable to ignoring the issue of random variation in testing and training.

Reference: T. Mitchell, "Machine Learning," chapter 5.

- Bootstrap method

. Bootstrap method is a general tool for accessing statistical accuracy.

. Let us consider the sample set

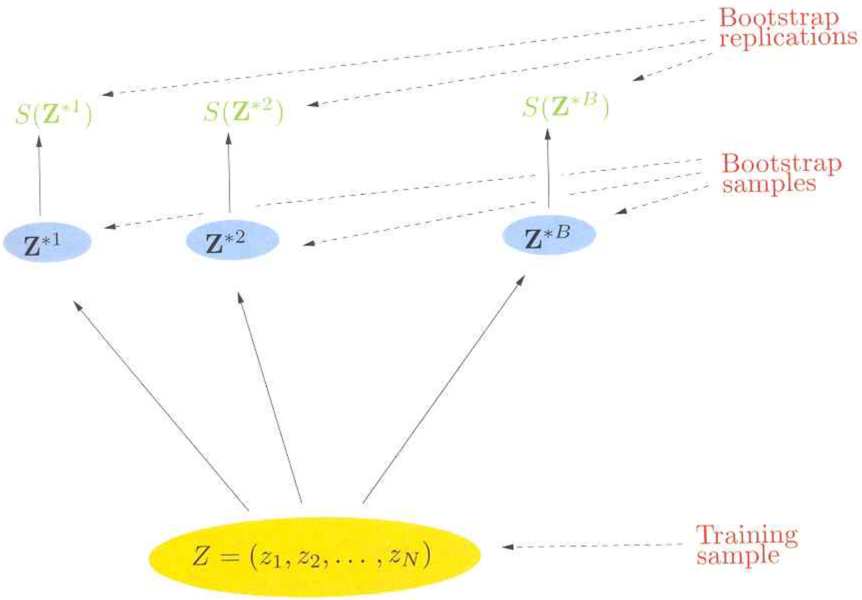
$Z = (z_1, z_2, \cdots, z_N)$ and

. the statistical quantity $S(Z)$ computed from the sample set Z.

eg. sample mean:

$$S(Z) = \frac{1}{N}\sum_{i=1}^{N} Z_i$$

. bootstrap process



Z^{*b}, $b = 1, 2, \cdots, B$ are bootstrap samples in which each sample is drawn randomly with replacement from Z.

. variance estimation

From the bootstrap process, variance can be estimated as

$$\widehat{Var}(S(Z)) = \frac{1}{B-1}\sum_{b=1}^{B}(S(Z^{*b}) - \overline{S}^{*})^2$$

where

$$\overline{S}^{*} = \frac{1}{B}\sum_{b=1}^{B} S(Z^{*b}).$$

We can consider $\widehat{Var}(S(Z))$ as a Monte-Carlo estimation of $Var(S(Z))$ under the sampling from the empirical distribution $\hat{F}$ for the data $Z=(Z_1, Z_2, \cdots, Z_N)$.
For this estimation, the proper value of B is typically between 25 and 200.

Bootstrap theorem shows that

$$\lim_{B\to\infty} \widehat{Var}(S(Z)) = Var(S(Z))$$

under the distribution of $\hat{F}$.

. confidence interval

From the bootstrap process, percentile interval is obtained.
Let $\hat{\theta}$ be an estimation of parameter θ

eg. $\hat{\theta}= S(Z) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} Z_i$

and $\hat{\theta}^*$ be $\hat{\theta}$ for bootstrap samples, that is,

$$\hat{\theta}^* = S(Z^*).$$

Then, $1-2\alpha$ percentile interval is given by

$$\left[\hat{\theta}_{\%lo}, \hat{\theta}_{\%up}\right]= \left[\hat{G}^{-1}(\alpha), \hat{G}^{-1}(1-\alpha)\right]$$

where $\hat{G}$ represents the cumulative distribution function of $\hat{\theta}^*$.

eg. If $\alpha = 0.05$ and $B = 1000$,

$\hat{\theta}_{\%lo}$ and $\hat{\theta}_{\%up}$ represent the 50th and 950th samples from the sorted $\hat{\theta}^*$ in ascending order respectively.

This estimate of confidence interval is good for unbiased estimate of θ.

. bias

The bias of bootstrap estimate is defined by

$$bias_B = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}^{*b}$$

where

$$\hat{\theta}^{*b} = S(Z^{*b}).$$

If $bias_B \ll (\widehat{Var}(S(Z))^{1/2}$, $\hat{\theta}$ is a good estimator. Otherwise, use the bias corrected estimator $\bar{\theta} = \hat{\theta} - bias_B$.

Reference: B. Fron and R. Tibshirani, "An Introduction to the Bootstrap," Chapman and Hall, 1993.

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Metrics for Performance Evaluation

- Focus on the predictive capability of a model
  - Rather than how fast it takes to classify or build models, scalability, etc.
- Confusion Matrix:

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)
b: FN (false negative)
c: FP (false positive)
d: TN (true negative)

Metrics for Performance Evaluation…

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric:

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Limitation of Accuracy

- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10
- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %
  - Accuracy is misleading because model does not detect any class 1 example

Cost Matrix

	PREDICTED CLASS		
	C(i|j)	**Class=Yes**	**Class=No**
ACTUAL CLASS	**Class=Yes**	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

C(i|j): Cost of misclassifying class j example as class i

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M_2	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Cost vs Accuracy

Count	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

Cost	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	p	q
	Class=No	q	p

Accuracy is proportional to cost if

1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

N = a + b + c + d

Accuracy = (a + d)/N

$$\begin{aligned}\text{Cost} &= p\,(a + d) + q\,(b + c)\\ &= p\,(a + d) + q\,(N - a - d)\\ &= q\,N - (q - p)(a + d)\\ &= N\,[q - (q-p) \times \text{Accuracy}]\end{aligned}$$

Cost-Sensitive Measures

$$\text{Precision (p)} = \frac{a}{a+c}$$

$$\text{Recall (r)} = \frac{a}{a+b}$$

$$\text{F-measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

- Precision is biased towards C(Yes|Yes) & C(Yes|No)
- Recall is biased towards C(Yes|Yes) & C(No|Yes)
- F-measure is biased towards all except C(No|No)

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

Model Evaluation

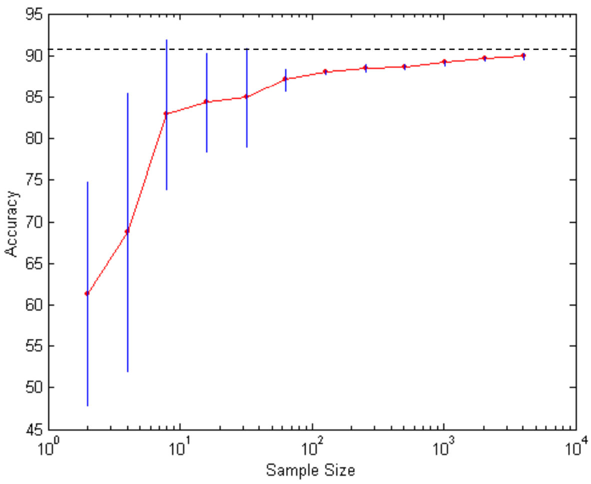
- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Methods for Performance Evaluation

- How to obtain a reliable estimate of performance?

- Performance of a model may depend on other factors besides the learning algorithm:
  - Class distribution
  - Cost of misclassification
  - Size of training and test sets

Learning Curve



- Learning curve shows how accuracy changes with varying sample size
- Requires a sampling schedule for creating learning curve:
  - Arithmetic sampling (Langley, et al)
  - Geometric sampling (Provost et al)

Effect of small sample size:
- Bias in the estimate
- Variance of estimate

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing

작은 dataset일 때 사용함.

- Random subsampling
  - Repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n
- Stratified sampling
  - oversampling vs undersampling
- Bootstrap
  - Sampling with replacement

Model Evaluation

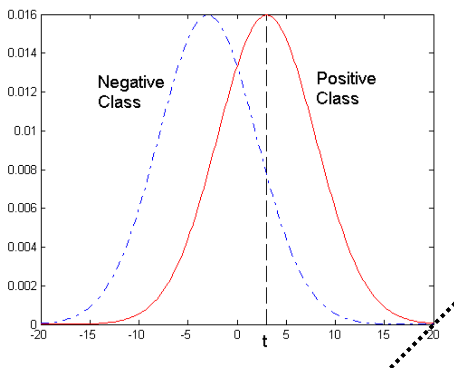
- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

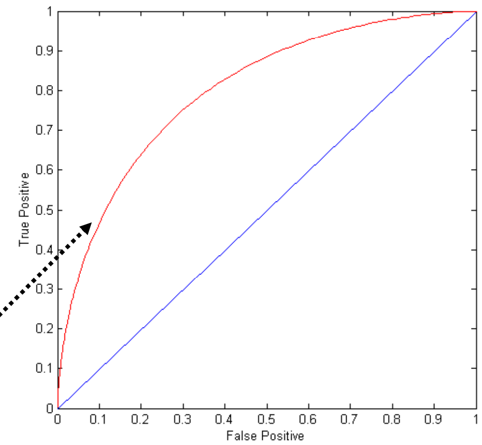
ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots TP (on the y-axis) against FP (on the x-axis)
- Performance of each classifier represented as a point on the ROC curve
  - changing the threshold of algorithm, sample distribution or cost matrix changes the location of the point

ROC Curve

- 1-dimensional data set containing 2 classes (positive and negative)
- any points located at x > t is classified as positive





At threshold t:

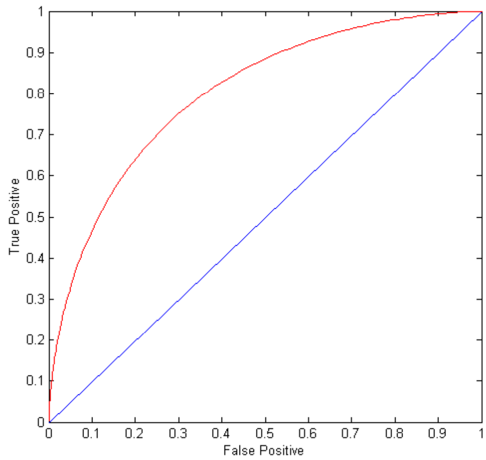
TP=0.5, FN=0.5, FP=0.12, FN=0.88

ROC Curve

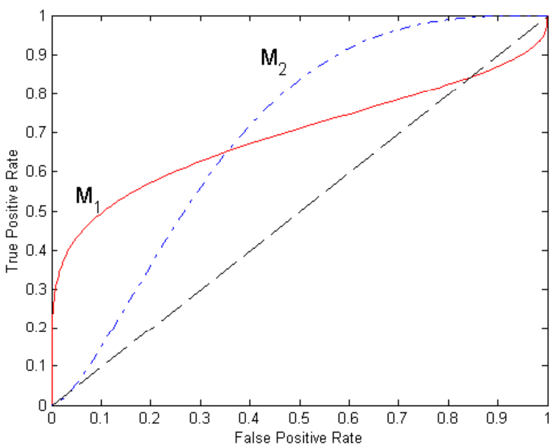
(TP,FP):

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (1,0): ideal

- Diagonal line:
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class



Using ROC for Model Comparison



- No model consistently outperform the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5

How to Construct an ROC curve

Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Use classifier that produces posterior probability for each test instance P(+|A)
- Sort the instances according to P(+|A) in decreasing order
- Apply threshold at each unique value of P(+|A)
- Count the number of TP, FP, TN, FN at each threshold
- TP rate, TPR = TP/(TP+FN)
- FP rate, FPR = FP/(FP + TN)

How to construct an ROC curve

Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC Curve: