Exercises

☺ 1. How to use the **prior** and **likehood** to calculate the **posterior**? What is the formula?

We can use the Bayes' formula to answer the question:

☺ 2. What's the difference in the ideas of the **minimum error Bayesian decision** and **minimum risk Bayesian decision**? What's the condition that makes the minimum error Bayesian decision identical to the minimum risk Bayesian decision?

the minimum error Bayesian decision: to minimize the classification error of the Bayesian decision.

the minimum risk Bayesian decision: to minimize the risk of the Bayesian decision.

$R(\alpha_1 \mid x) = \lambda_{11}P(\omega_1 \mid x) + \lambda_{12}P(\omega_2 \mid x)$

$R(\alpha_2 \mid x) = \lambda_{21}P(\omega_1 \mid x) + \lambda_{22}P(\omega_2 \mid x)$

if $R(\alpha_1 \mid x) < R(\alpha_2 \mid x)$ "decide ω_1" is taken

$(\lambda_{21}-\lambda_{11})P(\omega_1 \mid x) > (\lambda_{12}-\lambda_{22})P(\omega_2 \mid x)$ "decide ω_1" is taken

Condition: factor $(\lambda_{21}-\lambda_{11})$, $(\lambda_{12}-\lambda_{22})$ *both are positive, and* $(\lambda_{12}-\lambda_{22}) > (\lambda_{21}-\lambda_{11})$

☺ 3. A person takes a lab test of nuclear radiation and the result is positive. The test returns a correct positive result in 99% of the cases in which the nuclear radiation is actually present, and a correct negative result in 95% of the cases in which the nuclear radiation is not present. Furthermore, 3% of the entire population are radioaetively eontaminated. Is this person eontaminated?

- **Solution**:

Given: P(+|eontaminated)=0.99 P(-|no eontaminated)=0.95

P(eontaminated)=0.03 P(no eontaminated)=0.97

Compute:P(no eontaminated | +) , P(eontaminated | +),

P(eontaminated | +)=P(+|eontaminated)* P(eontaminated)/p(+)=0.99*0.03/p(+)=0.99*0.03/p(+)= 0.0297 /p(+)

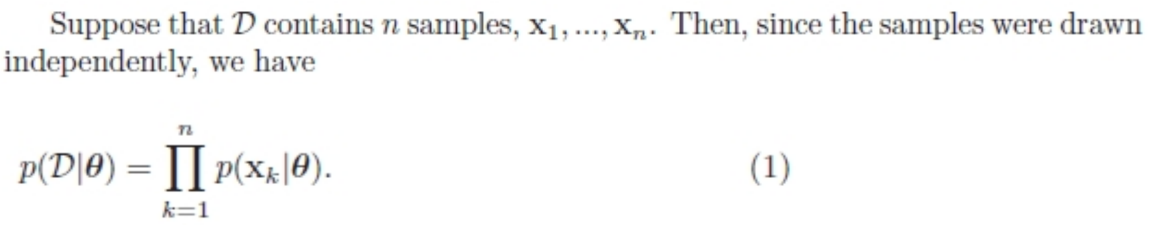
P(no eontaminated |+)=P(+|no eontaminated)*P(no eontaminated)/P(+)

=(1-0.95)*(1-0.03)/ P(+)=0.0485 / P(+)

Since P(no eontaminated | +) > P(eontaminated | +), we decide that the patient does not have cancer (Bayse decision rule)

☺ 4. Please present the basic ideas of the **maximum likehood estimation** method and **Bayesian estimation** method. When do these two methods have similar results ?

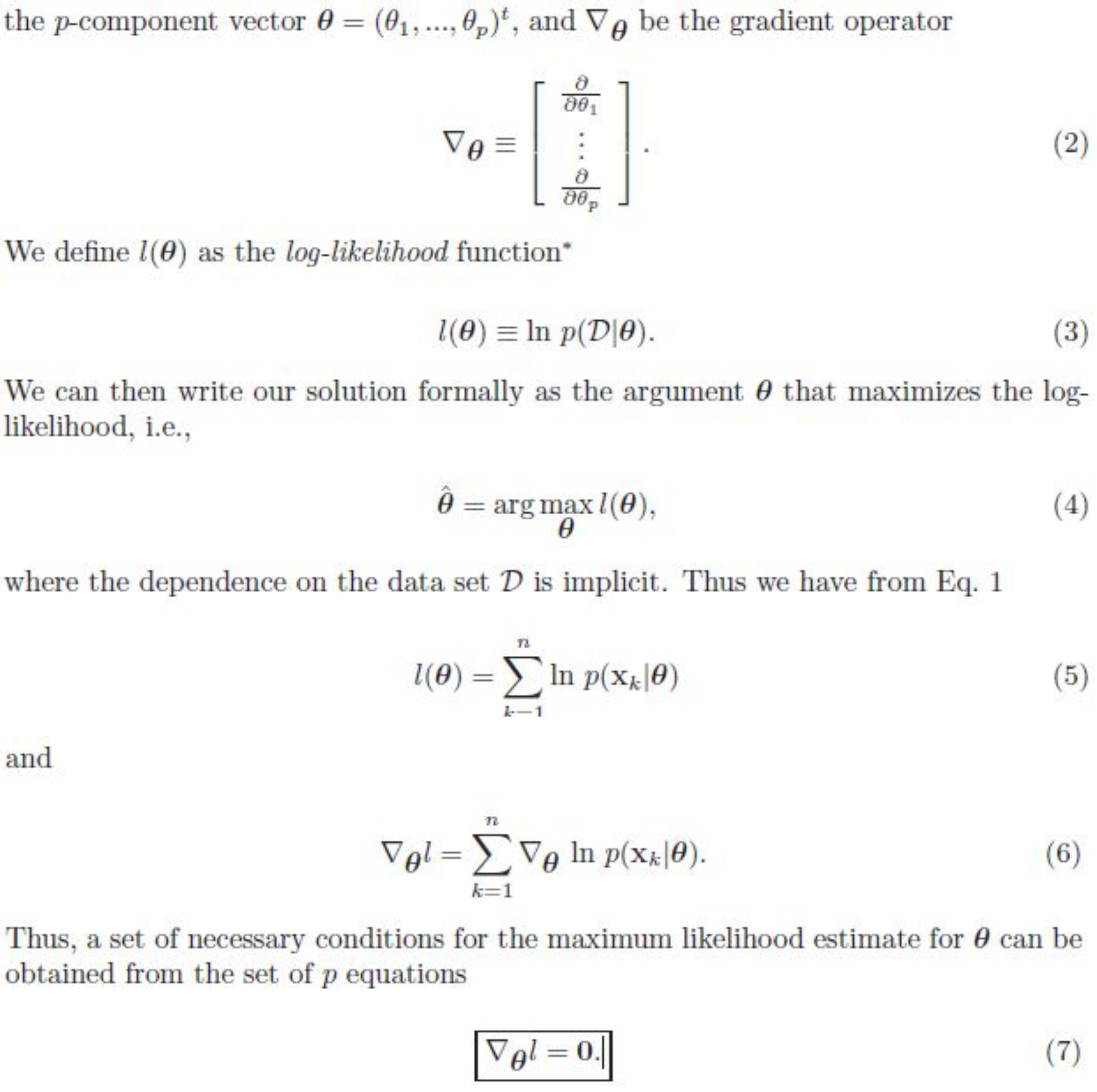
(1)General principle



Suppose that $\mathcal{D}$ contains n samples, $\mathbf{x}_1, ..., \mathbf{x}_n$. Then, since the samples were drawn independently, we have

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{k=1}^{n} p(\mathbf{x}_k|\boldsymbol{\theta}). \qquad (1)$$

For analytical purposes, it is usually easier to work with the logarithm of the likelihood than with the likelihood itself. If the number of parameters to be set is p, then we let θdenote



the p-component vector $\boldsymbol{\theta} = (\theta_1, ..., \theta_p)^t$, and $\nabla_{\boldsymbol{\theta}}$ be the gradient operator

$$\nabla_{\boldsymbol{\theta}} \equiv \begin{bmatrix} \frac{\partial}{\partial \theta_1} \\ \vdots \\ \frac{\partial}{\partial \theta_p} \end{bmatrix}. \qquad (2)$$

We define $l(\boldsymbol{\theta})$ as the *log-likelihood* function*

$$l(\boldsymbol{\theta}) \equiv \ln p(\mathcal{D}|\boldsymbol{\theta}). \qquad (3)$$

We can then write our solution formally as the argument $\boldsymbol{\theta}$ that maximizes the log-likelihood, i.e.,

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} l(\boldsymbol{\theta}), \qquad (4)$$

where the dependence on the data set $\mathcal{D}$ is implicit. Thus we have from Eq. 1

$$l(\boldsymbol{\theta}) = \sum_{k=1}^{n} \ln p(\mathbf{x}_k|\boldsymbol{\theta}) \qquad (5)$$

and

$$\nabla_{\boldsymbol{\theta}} l = \sum_{k=1}^{n} \nabla_{\boldsymbol{\theta}} \ln p(\mathbf{x}_k|\boldsymbol{\theta}). \qquad (6)$$

Thus, a set of necessary conditions for the maximum likelihood estimate for $\boldsymbol{\theta}$ can be obtained from the set of p equations

$$\boxed{\nabla_{\boldsymbol{\theta}} l = \mathbf{0}.} \qquad (7)$$

If all solutions are found, we are guaranteed that one represents the true maximum, though we might have to check each solution individually (or calculate second derivatives) to identify which is the global optimum.

(2)Goal: compute P(ωi | x, D)

Given the sample D, Bayes formula can be written

$$P(\omega_i \mid x, \mathrm{D}) = \frac{p(x \mid \omega_i, \mathrm{D}).P(\omega_i \mid \mathrm{D})}{\sum_{j=1}^{c} p(x \mid \omega_j, \mathrm{D}).P(\omega_j \mid \mathrm{D})}$$

To demonstrate the preceding equation, use:

$D = D_1 \cup D_2 ... \cup D_c \quad x \in D_i \rightarrow x \text{ is } \omega_i$

D_i has no influlence on $p(x | \omega_j , D_j)$ if $i <> j$

$P(\omega_i) = P(\omega_i | \mathrm{D})$ (Training sample provides this!)

Thus :

$$P(\omega_i | x, \mathrm{D}) = \frac{P(x | \omega_i , \mathrm{D}_i).P(\omega_i)}{\sum_{j=1}^{c} P(x | \omega_j , \mathrm{D}_j).P(\omega_j)}$$

Parameter Distribution

P(x) is unknown, we assume it has a known parametric form P(x|θ) , and value of parameter θ is unknown

Know prior density P(θ)

Training data convert P(θ) to a posterior density P(θ|D)

Our path:

$$\begin{aligned} & p(x | \omega_i , D) = p(x) \cong p(x | D) \\ & = \int p(x, \theta | D) d\theta \\ & = \int p(x | \theta) p(\theta | D) d\theta \end{aligned}$$

In virtually every case, maximum likelihood and Bayes solutions are equivalent in the asymptotic limit of infinite training data.

5. Please present the nature of **principal component analysis**.

1) PCA is an unsupervised method.
2) PCA is a good dimension reduction method. Usually, even if the dimension of the sample is greatly reduced, the main information can be still stored.
3) PCA is a de-correlation method. After the PCA transform, the obtained components are statistically uncorrelated.
4) PCA can be used as a compression method. PCA can achieve a high compression ratio.
5) PCA is the optimal representation method, which allows us to obtain the minimum reconstruction error.
6) As the transform axes of PCA are orthogonal, it is also referred to as an orthogonal transform method.

Disadvantage: need to a training phase.

6. Describe the basic idea and possible advantage of **Fisher discriminant analysis**.

Basic idea:

1) Fisher discriminant analysis's task is to find the best such direction w to obtain accurate classification.
2) Fisher discriminant analysis is to maximize the between-class distance and minimize the within-class distance
3) Fisher discriminant analysis exploits the training sample to produce transform axes.

Advantage:

1) The time complex is O(d^2n). It is linear.
2) Fisher discriminant analysis is a linear discriminant method, thus it is easy to generalize.
3) Fisher discriminant analysis is supervised; it can exploit the training sample to produce the transform axes.

7. What is the **K nearest neighbor classifier**? Is it reasonable?

We can center a cell about **x** and let it grow until it captures *kn* samples, where *kn* is some specified function of *n*. These samples are the *kn nearest-neighbors* of **x**. If the density is high near **x**, the cell will be relatively small, which leads to good resolution. If the density is low, it is true that the cell will grow large, but it will stop soon after it enters regions of higher density.

Classify x by assigning it the label most frequently represented among the k nearest samples and use a voting scheme

It is reasonable. The nearest-neighbor rule leads to an error rate greater than the minimum possible value of the Bayes rate; If the number of prototype is large (unlimited), the error rate of the nearest-neighbor classifier is never worse than twice the Bayes rate.

8. Is it possible that a classifier can obtain a higher accuracy for any dataset than any other classifiers?

No. According to the No Free Lunch Theorem, there are no context-independent or problem-independent reasons to favor one learning or classification method over another.

If one algorithm seems to outperform another in a particular situation, it is a consequence of it fit to the particular pattern recognition problem, not the general superiority of the algorithm.

☺ 9. Please describe the **over-fitting problem**.

Over-fitting generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations. A model which has been over-fit will generally have poor predictive performance, as it can exaggerate minor fluctuations in the data.

1) Over-fitting: The classification performs good on training dataset but bad on test datadset.
2) Reason: (i)The training sample is indequate;(ii)There are difference between training samples and test samples;(iii)The classification focus on detail;(iv)Samples exist noise.
3) How to handle: (i)reduce dimensionality(ii)suppose all classes share same covariance matrix(iii) look for a better estimate for covariance matrix.

10. Usually a more complex learning algorithm can obtain a higher accuracy in the training stage. So, **should a more complex learning algorithm be favored** ?

No context-independent or usage-independent reasons to favor one learning or classification method over another to obtain good generalization performance.

When confronting a new pattern recognition problem, we need focus on the aspects — prior information, data distribution, amount of training data and cost or reward functions.

Ugly Duckling Theorem: an analogous theorem, addresses features and patterns. shows that in the absence of assumptions we should not prefer any learning or classification algorithm over another.

11. Under the condition that the number of the training samples approaches to the infinity, **the estimation of the mean obtained using Bayesian estimation method is almost identical to that obtained using the maximum likehood estimation method**. Is this statement correct ?

For the reasonable prior distributions that does not preclude the true solution: Maximun likelihood and Bayesian solution are equivalent in the asympstotic limit of infinit traning data.

☺ 12. Can **the minimum squared error procedure** be used for binary classification ?

Yes, the minimum squared error procedure can be used for binary classification.

$$\begin{pmatrix} y_{10} & y_{11} & \cdots & y_{1d} \\ y_{20} & y_{21} & \cdots & y_{2d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{n0} & y_{n1} & \cdots & y_{nd} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \cdots \\ a_d \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_n \end{pmatrix}$$

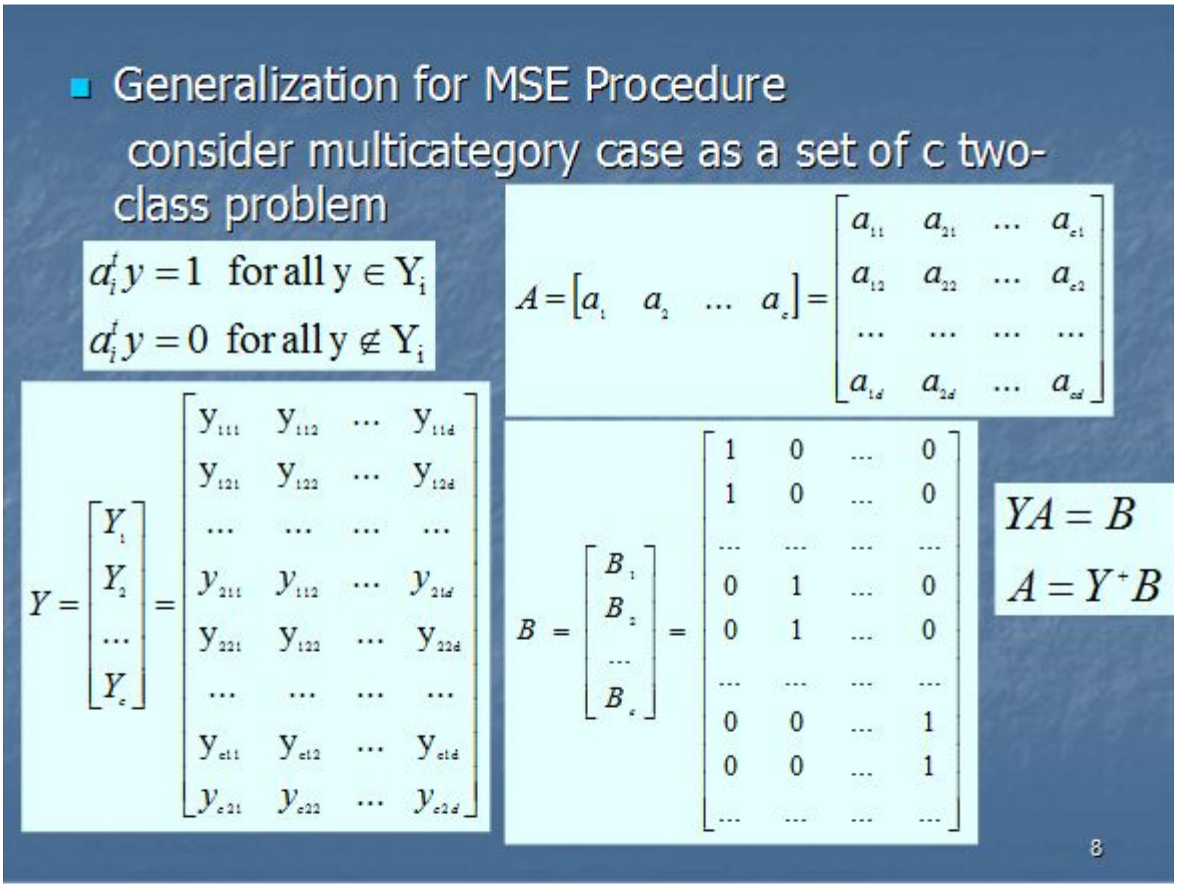
$$Ya = b, \quad Y = \begin{bmatrix} Y_1^T \\ . \\ . \\ . \\ Y_n^T \end{bmatrix}, Y_i^T = \begin{bmatrix} y_{i0} \\ . \\ . \\ . \\ y_{id} \end{bmatrix}.$$

A simple way to set b : if Y_i is from the first class, then b_i is set to 1; if Y_i is from the second class, then b_i is set to -1.

Another simple way to set b : if Y_i is from the first class, then b_i is set to $\frac{n}{n_1}$; if Y_i is from the second class, then b_i is set to -$\frac{n}{n_2}$.

13. Can you **devise a minimum squared error procedure to perform multiclass classification** ?

Yes, a minimum squared error procedure to perform multiclass classification.



- Generalization for MSE Procedure

consider multicategory case as a set of c two-class problem

$$a_i^t y = 1 \quad \text{for all } y \in Y_i$$

$$a_i^t y = 0 \quad \text{for all } y \notin Y_i$$

$$A = [a_1 \quad a_2 \quad \dots \quad a_c] = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{c1} \\ a_{12} & a_{22} & \dots & a_{c2} \\ \dots & \dots & \dots & \dots \\ a_{1d} & a_{2d} & \dots & a_{cd} \end{bmatrix}$$

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \dots \\ Y_c \end{bmatrix} = \begin{bmatrix} y_{111} & y_{112} & \cdots & y_{11d} \\ y_{121} & y_{122} & \cdots & y_{12d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{211} & y_{112} & \cdots & y_{21d} \\ y_{221} & y_{122} & \cdots & y_{22d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{c11} & y_{c12} & \cdots & y_{c1d} \\ y_{c21} & y_{c22} & \cdots & y_{c2d} \end{bmatrix}$$

$$B = \begin{bmatrix} B_1 \\ B_2 \\ \dots \\ B_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

$$YA = B$$

$$A = Y^{+}B$$

8

where Yi={yi11, yi12, ... , yi1d} ai^{t}=[ai1,ai2,.., aid]^{t}, bi=[0 0 ...i...0]
yi21, yi22, ... , yi2d

14. Which kind of applications is the **Markov model** suitable for ? (**了解**) p105

In problems that have an inherent temporality（时间性）— that is, consist of a process that unfolds in time — we may have states at time t that are influenced directly by a state at $t-1$.

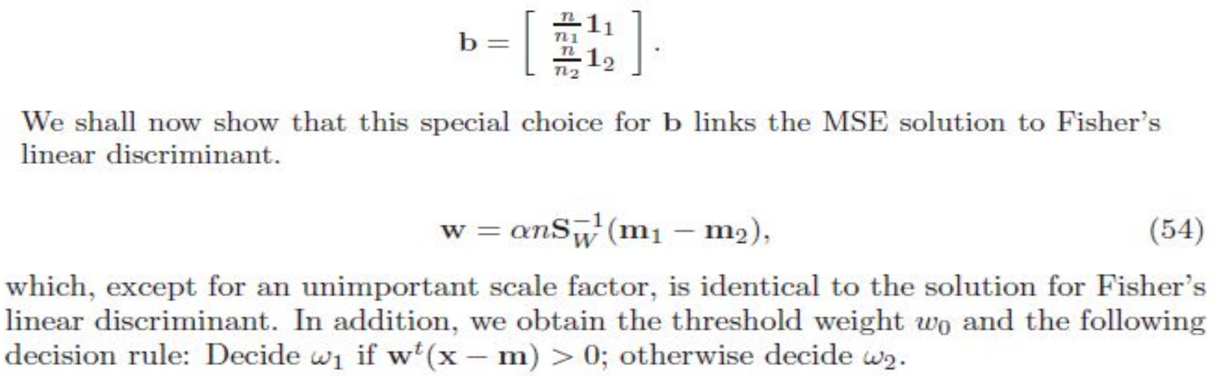
Applications: speech recognition, gesture recognition, parts of speech tagging (词性标注)and DNA sequencing

HMM: evaluation, decoding , learning problems

15. For minimum squared error procedure based on Ya=b (Y is the matrix consisting of all the training samples), if we have proper b and criterion function, then this **minimum squared error**

procedure might be equivalent to Fisher discriminant analysis. Is this presentation correct ?

Yes,



$$\mathbf{b} = \begin{bmatrix} \frac{n}{n_1}\mathbf{1}_1 \\ \frac{n}{n_2}\mathbf{1}_2 \end{bmatrix}.$$

We shall now show that this special choice for $\mathbf{b}$ links the MSE solution to Fisher's linear discriminant.

$$\mathbf{w} = \alpha n \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2), \qquad (54)$$

which, except for an unimportant scale factor, is identical to the solution for Fisher's linear discriminant. In addition, we obtain the threshold weight w_0 and the following decision rule: Decide ω_1 if $\mathbf{w}^t(\mathbf{x} - \mathbf{m}) > 0$; otherwise decide ω_2.

16. Suppose that the number of the training samples approaches to the infinity, then the minimum error Bayesian decision will **perform better than any other classifier** achieving a lower classification error rate. Do you agree on this?

No.

According to the No Free Lunch Theorem, there are no context-independent or problem-independent reasons to favor one learning or classification method over another.

If one algorithm seems to outperform another in a particular situation, it is a consequence of it fit to the particular pattern recognition problem, not the general superiority of the algorithm.

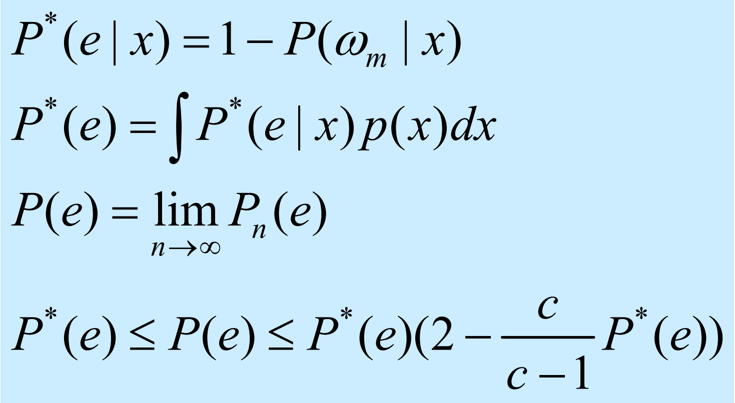
Yes, it is correct. The premise is the number of the training samples approaches to the infinity.

How to prove?

17. What are **the upper and lower bound** of the classification error rate of the **K nearest neighbor classifier** ?

(1)The nearest-neighbor rule leads to an error rate greater than the minimum possible value of the Bayes rate

(2)If the number of prototype is large (unlimited), the error rate of the nearest-neighbor classifier is never worse than twice the Baves rate (it can be demonstrated!)



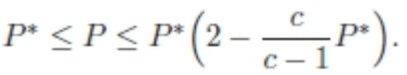
$$P^*(e\,|\,x) = 1 - P(\omega_m\,|\,x)$$

$$P^*(e) = \int P^*(e\,|\,x)p(x)dx$$

$$P(e) = \lim_{n\to\infty} P_n(e)$$

$$P^*(e) \le P(e) \le P^*(e)(2 - \frac{c}{c-1}P^*(e))$$

Here we show that



$$P^* \le P \le P^*\left(2 - \frac{c}{c-1}P^*\right).$$

If different sets of n samples are used to classify x, different vectors x will be obtained for the nearest-neighbor of x. Since the decision rule depends on this nearest-neighbor, we have a conditional probability of error P(e|x, x) that depends on both x and x . By averaging over x , we

$$P(e|\mathbf{x}) = \int P(e|\mathbf{x},\mathbf{x}')p(\mathbf{x}'|\mathbf{x})\, d\mathbf{x}'$$

obtain

As n goes to infinity we expect p(x |x) to approach a delta function centered at x. Assume that at the given x, p(•) is continuous and not equal to zero. the probability that any sample falls within a hypersphere S centered about x is some positive number Ps:

$$P_s = \int_{\mathbf{x}' \in \mathcal{S}} p(\mathbf{x}')\, d\mathbf{x}'.$$

Thus x converges to x in probability, and p(x |x) approaches a delta function, as expected.

18. Can you demonstrate that a **statistics-based classifier** usually cannot lead to a classification accuracy of 100% ?

(1) The training sample is limited, but the real data is infinite and we cannot get all the samples.

(2) We cannot ensure the correct of the training sample. It may has noise.

19. What is **representation-based classification**? Please present the characteristics of Representation-based classification.

(1) The representation-based methods attempts to exploit the training samples to provide local representation for the test sample. Typical examples include sparse representation.

(2) The method first expresses the test sample as a linear combination of a number of training samples and exploits the representation result to perform classification. In other words, it is regarded that the test sample is related with each training sample. The test sample will be classified into the class that has the most “relationship”.

Basic idea   a1~=0; ai=0

Testing sample: y

Training samples: x1, …, xn   Suppose that   y=a1x1+…+ anxn

A=(XTX)-1XTy   or   A=(XTX+uI)-1XTy

Suppose that training samples xp,…,xq are from the s-th class.

The representation error of the s-th class is evaluated by Es=||y-apxp… -aqxq ||

(3) The method considers that testing sample y is from the class that has the smallest representation error.

(4) A good performance can be achieved.

20. A simple **representation-based classification** method is presented as follows:

This method seeks to represent the test sample as a linear combination of all training samples and uses the representation result to classify the test sample:

$$y = b_1\tilde{x}_1 + \ldots + b_M\tilde{x}_M\ , \qquad (1)$$

where $\tilde{x}_i$ ($i = 1,2,\ldots,M$) denote all the training samples and b_i ($i = 1,2,\ldots,M$) are the coefficients.

We rewrite Eq.(1) into $y = \tilde{X}B$, (2)

where $B = [b_1 \ldots b_M]^T$, $\tilde{X} = [\tilde{x}_1 \ldots \tilde{x}_M]$. If $\tilde{X}$ is not singular,

we can solve B using $B = (\tilde{X}^T\tilde{X})^{-1}\tilde{X}^T y$; otherwise, we can solve it using

$B = (\tilde{X}^T\tilde{X} + \gamma I)^{-1}\tilde{X}^T y$, (3) where γ is a positive constant and I is the identity matrix. After we obtain B, we refer to $\tilde{X}B$ as the representation result of our method. We can convert the representation result into a two-dimensional image having the same size of the original sample image.

We exploit the sum of the contribution, to representing the test sample, of the training samples from a class, to classify the test sample. For example, if all the training samples from the r th ($r \in C$) class are $\tilde{x}_s \ldots \tilde{x}_t$, then the sum of the contribution, to representing the test sample, of the r th class will be

$$g_r = a_s\tilde{x}_s + \ldots + a_t\tilde{x}_t . \qquad (4)$$

We calculate the deviation of g_r from y using

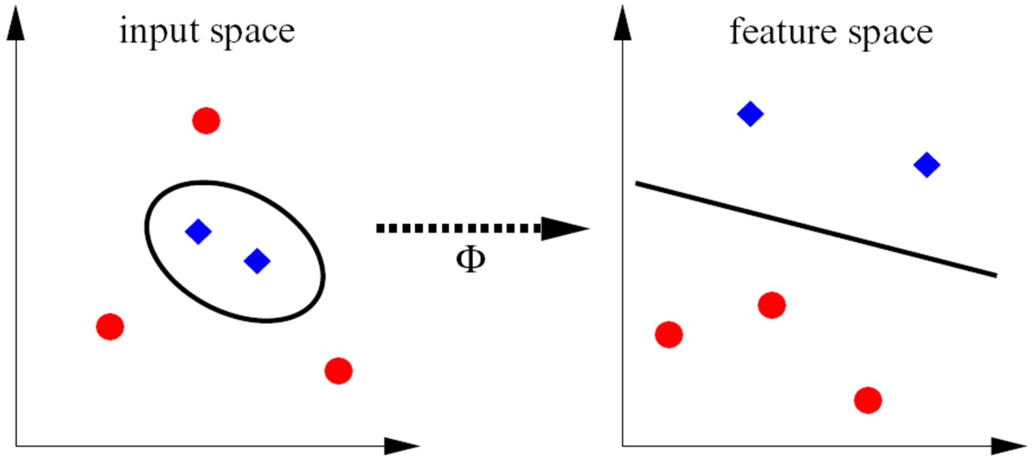
$$D_r = \| y - g_r \|^2, r \in C . \qquad (5)$$

We can also convert g_r into a two-dimensional matrix having the same size of the original sample image. If we do so, we refer to the matrix as the two-dimensional image corresponding to the contribution of the r th class. The smaller the deviation D_r, the greater the contribution to representing the test sample of the r th class. In other words, if $D_q = \min D_r$ ($q, r \in C$), the test sample will be classified into the q th class.

From the above presentation, we know that representation-based classification method is a novel method and totally different from previous classifiers ! It performs very well in image-based classification, such as face recognition and palmprint recognition.

We should understand its nature and advantages.

21. Please describe the difference between **linear and nonlinear discriminant functions**? What potential advantage does nonlinear discriminant function have in comparison with linear discriminant function?



The above figure is just auxiliary for the question !

It was agreed that simplicity is generally best.The use of linear method is recommended wherever possible. It also was agreed that nonlinear methods in some applications can provide better result, particularly with complex and/or other very large data.

A linear discriminant function is defined by a hyperplane's vector w and offset b, the decision boundary is {x| wx+b=0}.

Linear discriminant functions are generally more robust than nonlinear discriminant functions. Because of having only limited flexibility and less prone to overfitting.

A linear algorithm is applied in some appropriate(kernel) feature space.

The linear classification in feature space corresponds to a (powerful) non-linear decision function in input space.

The non-linear methods often involve a number of parameters.

Linear methods seem ideal when limited data and limited knowledge about the data is available. If there are large amount of data, nonlinear methods are suitable to find potentially more complex structure in a data.

22. What is the **naïve Bayes rule** ?

The naïve Bayes rule is a simple probabilistic classifier based on applying Bayes theorem with strong independence assumption that all attributions are independent another.

The procedures are shown as follows:

(1) Given unknown tuple X={x1, x2, …, xn},where xi are attributions.

(2) Class C={w1, w2, …, wn}

(3) Calculate P(w1|X), P(w2|X), …, P(wn|X), $p(\omega_k|\mathbf{x}) \propto \prod_{i=1}^{d} p(x_i|\omega_k).$

(4) If P(wk|X)=Max{ P(w1|X), P(w2|X), …, P(wn|X)} then decide X belong to wk.

23. What is the **difference between supervised and unsupervised learning methods**? Please show two examples of supervised and unsupervised learning methods.

In supervised learning, a teacher provides a category label or cost for each pattern in a training set, and we seek to reduce the sum of the costs for these patterns.

In *unsupervised learning* or *clustering* there is no explicit teacher, and the system forms clusters or "natural groupings" of the input patterns. "Natural" is always defined explicitly or implicitly in the clustering system itself, and given a particular set of patterns or cost function, different clustering algorithms lead to different clusters. Often the user will set the hypothesized number of different clusters ahead of time.

For example, for supervised learning, there are maximum likelihood estimation and Bayesian learning; for unsupervised learning, there are PCA and Kmeans.

24. **In some special real-world** classification applications, the **Bayesian decision theory might perform badly**. What are possible reasons ?

The possible reasons will be that we can not get all the knowledge of the probability structure of the problem or only get some fuzzy or general knowledge about the problem.

25. Suppose that we are **applying a linear discriminant function to a nonlinear separable problem**, what means can we adopt to obtain an optimal solution?

(1) If problem is non-linear separable problem in low-dimension, we can transform it into high-dimension such that it is a linear separable.

Take an example.

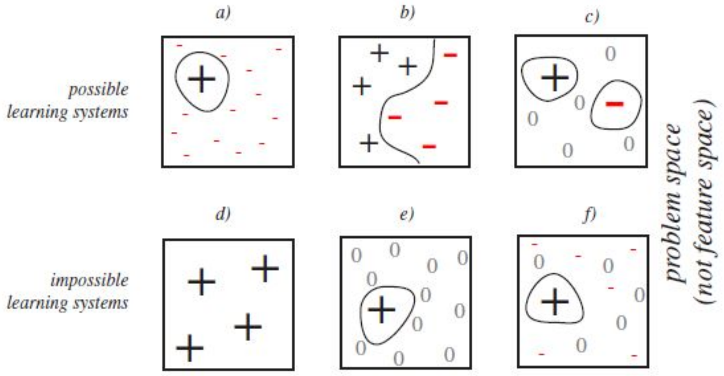
The discriminant function d(x) is quadratic polynomial function, x=(x1, x2)^t , d(x)=w11x1^2+w12x1x2+w22x2^2+w1x1+w2x2+w3.

we can linearize it:

x*=(x1^2, x1x2, x2^2, x1, x2, 1)^t w=(w11, w12, w22, w1, w2, w3)^t.

(2) According to the generalized linear discriminant function, we can transfer any low-dimension nonlinear function into high-dimension linear function. We can adopt kernel function to solve this problem such as KPCA, KMSE, SVM.

26. Please present **possible generalization capability** in the sample space of a method.



① The no free lunch theorem shows the generalization performance on off-training set data (d, e, f) that can be achieved (a, b, c) and also show the performance that can't be achieved.

② Each square represents all possible classification problem consistent with the training data.

③ "+" indicates that the classification algorithm has generalization higher than average, "-" indicates lower than average.

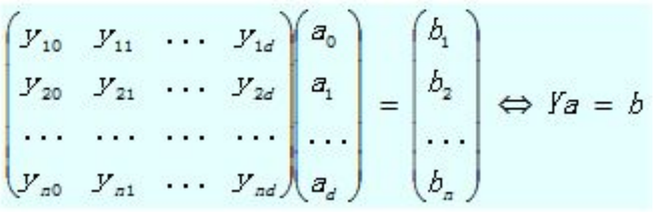
④ a) shows that it's possible for an algorithm to have high accuracy on a small set of problems so long as it has poor performance on all other problem.

b) shows that it's possible to have excellent performance throughout a large range of problem, but this will be balanced by very poor performance on other problems.

⑤ c) shows that it's possible to have higher-than-average performance on a small set of problems, but this will be balanced by poor performance on another small set of problems, and average performance on remaining.

⑥ It is impossible to have good performance throughout the full range of problems. It is also impossible to have higher-than-average performance on some problem while having average performance on every where else.

27. Apply model **Ya=b** to perform classification.



$$\begin{pmatrix} y_{10} & y_{11} & \cdots & y_{1d} \\ y_{20} & y_{21} & \cdots & y_{2d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{n0} & y_{n1} & \cdots & y_{nd} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \cdots \\ a_d \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_n \end{pmatrix} \Leftrightarrow Ya = b$$

If we define the error vector e by

$$\mathbf{e} = \mathbf{Ya} - \mathbf{b}$$

then one approach is to try to minimize the squared length of the error vector. This is equivalent to minimizing the sum-of-squared-error criterion function

$$J_s(\mathbf{a}) = \|\mathbf{Ya} - \mathbf{b}\|^2 = \sum_{i=1}^{n} (\mathbf{a}^t \mathbf{y}_i - b_i)^2.$$

A simple closed-form solution can also be found by forming the gradient

$$\nabla J_s = \sum_{i=1}^{n} 2(\mathbf{a}^t \mathbf{y}_i - b_i)\mathbf{y}_i = 2\mathbf{Y}^t(\mathbf{Ya} - \mathbf{b})$$

and setting it equal to zero. This yields the necessary condition

$$\mathbf{Y}^t\mathbf{Ya} = \mathbf{Y}^t\mathbf{b}$$

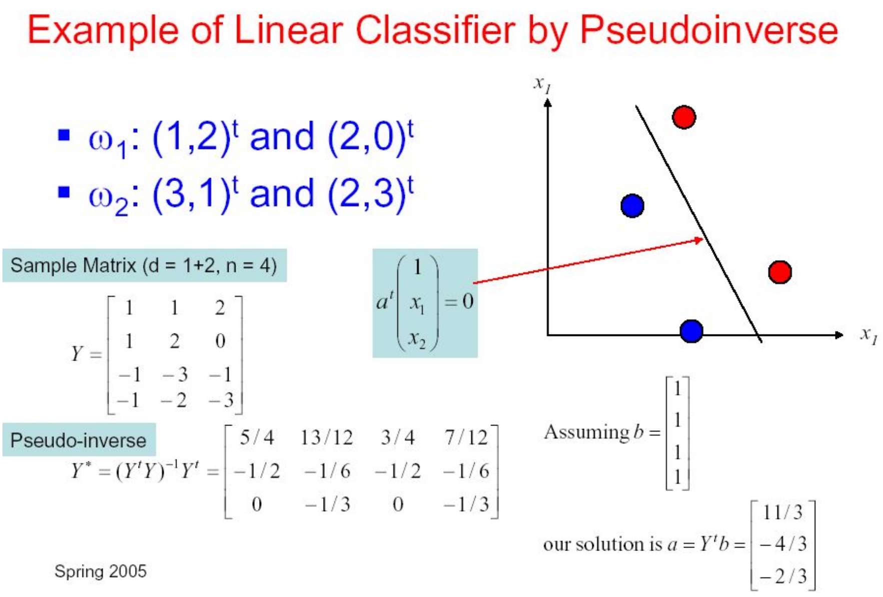
and in this way we have converted the problem of solving Ya = b to that of solving

$$\mathbf{Y}^t\mathbf{Ya} = \mathbf{Y}^t\mathbf{b}.$$

$$\mathbf{a} = (\mathbf{Y}^t\mathbf{Y})^{-1}\mathbf{Y}^t\mathbf{b}$$

$\mathbf{a} = (\mathbf{Y}^t\mathbf{Y})^{-1}\mathbf{Y}^t\mathbf{b}$ is an MSE solution to Ya = b.

28. How to extend the **binary minimum squared error** procedure to the multiclass minimum squared error procedure?



Example of Linear Classifier by Pseudoinverse

- ω_1: $(1,2)^t$ and $(2,0)^t$
- ω_2: $(3,1)^t$ and $(2,3)^t$

Sample Matrix (d = 1+2, n = 4)

$$Y = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 0 \\ -1 & -3 & -1 \\ -1 & -2 & -3 \end{bmatrix}$$

$$a^t \begin{pmatrix} 1 \\ x_1 \\ x_2 \end{pmatrix} = 0$$

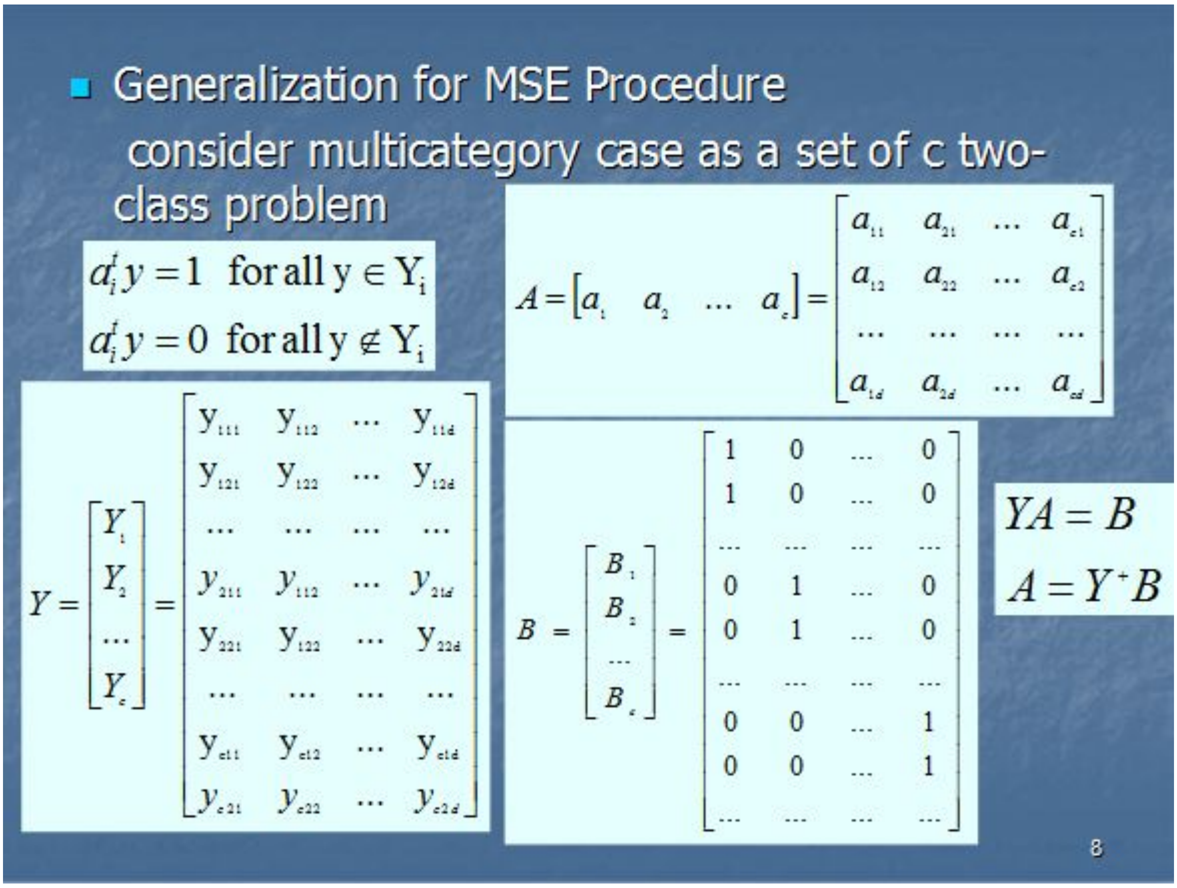
Pseudo-inverse

$$Y^* = (Y^tY)^{-1}Y^t = \begin{bmatrix} 5/4 & 13/12 & 3/4 & 7/12 \\ -1/2 & -1/6 & -1/2 & -1/6 \\ 0 & -1/3 & 0 & -1/3 \end{bmatrix}$$

Assuming $b = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$

our solution is $a = Y^t b = \begin{bmatrix} 11/3 \\ -4/3 \\ -2/3 \end{bmatrix}$

Spring 2005



- Generalization for MSE Procedure

consider multicategory case as a set of c two-class problem

$$a_i^t y = 1 \quad \text{for all } \mathrm{y} \in \mathrm{Y_i}$$
$$a_i^t y = 0 \quad \text{for all } \mathrm{y} \notin \mathrm{Y_i}$$

$$A = \begin{bmatrix} a_1 & a_2 & \dots & a_c \end{bmatrix} = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{c1} \\ a_{12} & a_{22} & \dots & a_{c2} \\ \dots & \dots & \dots & \dots \\ a_{1d} & a_{2d} & \dots & a_{cd} \end{bmatrix}$$

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \dots \\ Y_c \end{bmatrix} = \begin{bmatrix} \mathrm{y}_{111} & \mathrm{y}_{112} & \cdots & \mathrm{y}_{11d} \\ \mathrm{y}_{121} & \mathrm{y}_{122} & \cdots & \mathrm{y}_{12d} \\ \cdots & \cdots & \cdots & \cdots \\ y_{211} & y_{112} & \cdots & y_{21d} \\ \mathrm{y}_{221} & \mathrm{y}_{122} & \cdots & \mathrm{y}_{22d} \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{y}_{c11} & \mathrm{y}_{c12} & \cdots & \mathrm{y}_{c1d} \\ y_{c21} & y_{c22} & \cdots & y_{c2d} \end{bmatrix}$$

$$B = \begin{bmatrix} B_1 \\ B_2 \\ \dots \\ B_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

$$YA = B$$
$$A = Y^+ B$$