PATTERN RECOGNITION AND MACHINE LEARNING

CHAPTER 4: LINEAR MODELS FOR CLASSIFICATION

Learning Objectives

1、What are linear classification models?

2、What are the three linear classification approaches?

3、What is the Fisher’s discriminant method?

4、What is the Perceptron method?

5、What is the Gaussian mixture model method?

6、What is the logistic regression method?

7、How to compare the discriminative and generative methods?

8、What is the Bayesian Information Criterion?

Outlines

- Three Approaches to Linear Classification Models
- Approach I: Discriminant Functions
- Least Square Classification
- Fisher Discriminant Function
- Perceptrons
- Approach II: Probabilistic Generative Models
- Approach III: Probabilistic Discriminative Models
- Bayesian Information Criterion

Probabilistic Generative Models

- **Use a separate generative model of the input vectors for each class, and see which model makes a test input vector most probable.**
- **The posterior probability of class 1 is given by**:

$$p(C_1 \mid \mathbf{x}) = \frac{p(C_1)\,p(\mathbf{x} \mid C_1)}{p(C_1)\,p(\mathbf{x} \mid C_1) + p(C_0)\,p(\mathbf{x} \mid C_0)} = \frac{1}{1+e^{-z}} = \sigma(z)$$

$$where \quad z = \ln \frac{p(C_1)\,p(\mathbf{x} \mid C_1)}{p(C_0)\,p(\mathbf{x} \mid C_0)} = \boxed{\ln \frac{p(C_1 \mid \mathbf{x})}{1 - p(C_1 \mid \mathbf{x})}}$$

z is called the logit and is given by the log odds

A Simple Example

- Assume that the input vectors for each class are from a Gaussian distribution, and all classes have the same covariance matrix.

$$p(\mathbf{x} \mid C_k) = \underset{\substack{\text{normalizing} \\ \text{constant}}}{a} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \underset{\text{inverse covariance matrix}}{\Sigma^{-1}}(\mathbf{x}-\boldsymbol{\mu}_k)\right\}$$

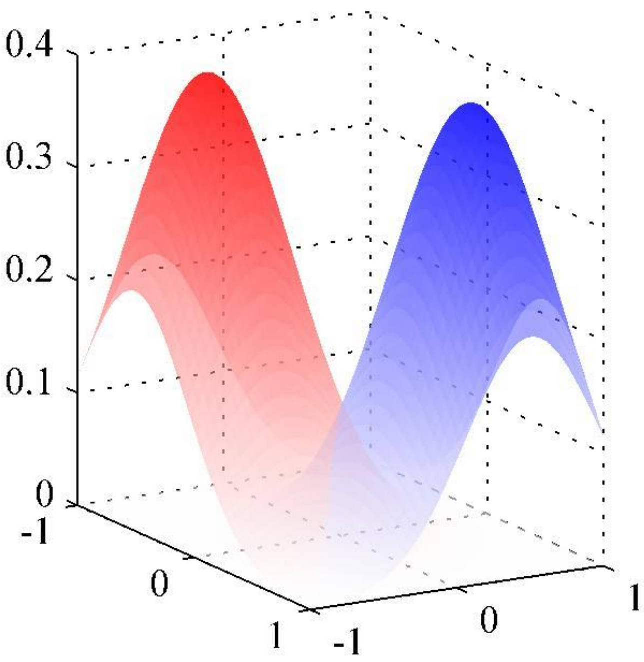
- For two classes, C1 and C0, the posterior is a logistic:

$$p(C_1 \mid \mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$$

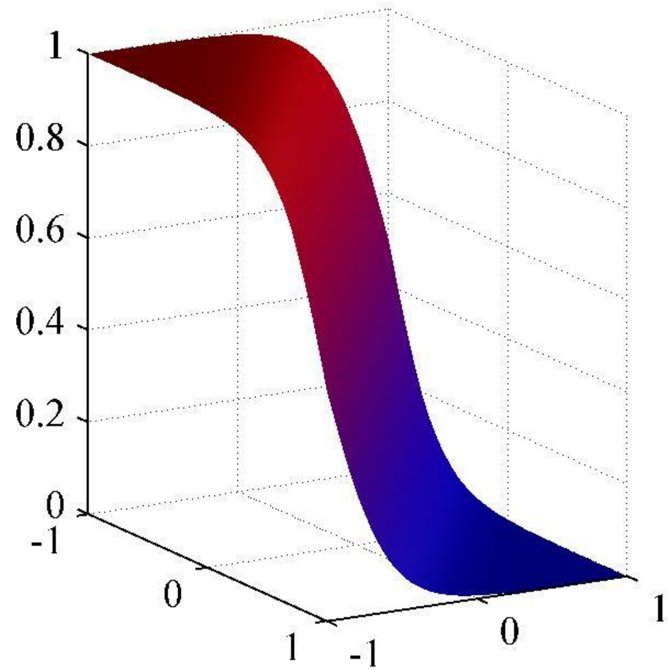
$$\mathbf{w} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$$

$$w_0 = -\frac{1}{2}\boldsymbol{\mu}_1^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_0^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0 + \ln\frac{p(C_1)}{p(C_0)}$$

Likelihood and Posterior



$p(x|C_0)$ $\quad$ $p(x|C_1)$



$p(C_0|x)$

K-Case Classification

$$\begin{aligned} p(\mathcal{C}_k|\mathbf{x}) &= \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{\sum_j p(\mathbf{x}|\mathcal{C}_j)p(\mathcal{C}_j)} \\ &= \frac{\exp(a_k)}{\sum_j \exp(a_j)} \end{aligned}$$

$$a_k = \ln p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)$$

$$a_k(\mathbf{x}) = \mathbf{w}_k^{\mathrm{T}}\mathbf{x} + w_{k0}$$

$$\begin{aligned} \mathbf{w}_k &= \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k \\ w_{k0} &= -\frac{1}{2}\boldsymbol{\mu}_k^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k + \ln p(\mathcal{C}_k) \end{aligned}$$

Inverse Covariance Matrix

- If the Gaussian is spherical we don't need to worry about the covariance matrix.
- So we could start by transforming the data space to make the Gaussian spherical
  - ✓ This is called "whitening" the data.
  - ✓ It pre-multiplies by the matrix square root of the inverse covariance matrix.
- In the transformed space, the weight vector is just the difference between the transformed means.

$$\mathbf{w} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$$

$$\textit{gives the same value}$$

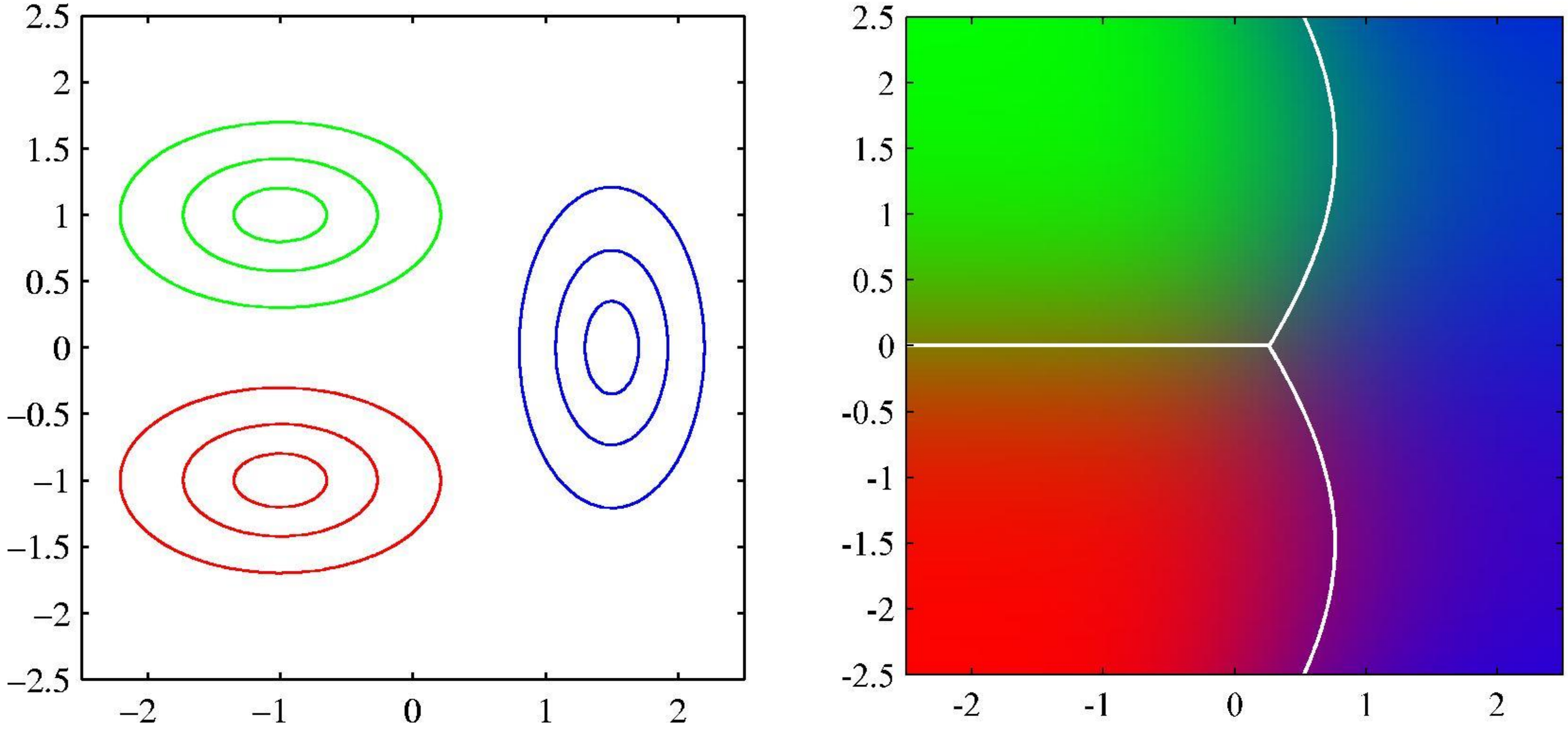
$$\textit{for} \ \ \mathbf{w}^T \mathbf{x} \ \ \textit{as}:$$

$$\mathbf{w}_{aff} = \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\mu}_1 - \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\mu}_0$$

$$\textit{and} \ \ \mathbf{x}_{aff} = \boldsymbol{\Sigma}^{-\frac{1}{2}} \mathbf{x}$$

$$\textit{gives} \ \ \textit{for} \ \ \mathbf{w}_{aff}^T \mathbf{x}_{aff}$$

Different Covariance Matrices



The decision surface is planar when the covariance matrices are the same; the decision surface is quadratic when they are not.

Generative: ML Gaussian Mixtures

$$p(x, C_1) = p(C_1)p(x|C_1) = \pi N(x|\mu_1, \Sigma)$$

$$p(x, C_2) = p(C_2)p(x|C_2) = (1-\pi)N(x|\mu_2, \Sigma)$$

Likelihood $$p(\boldsymbol{t}, \boldsymbol{X}|\pi, \mu_1, \mu_2, \Sigma) = \prod_{n=1}^{N} [\pi N(x_n|\mu_1, \Sigma)]^{t_n} [(1-\pi)N(x_n|\mu_2, \Sigma)]^{1-t_n}$$

$$\Rightarrow \quad \pi_{ML} = \frac{1}{N}\sum_{n=1}^{N} t_n = \frac{N_1}{N} = \frac{N_1}{N_1 + N_2} \qquad \mu_{1\mathrm{ML}} = \frac{1}{N_1}\sum_{n=1}^{N} t_n x_n \quad \mu_{2\mathrm{ML}} = \frac{1}{N_2}\sum_{n=1}^{N} (1-t_n) x_n$$

$$\Sigma = \pi\Sigma_1 + (1-\pi)\Sigma_2 \qquad \Sigma_{i\mathrm{ML}} = \frac{1}{N_i}\sum_{x_n \in C_i} (x_n - \mu_i)(x_n - \mu_i)^T \qquad \text{i=1,2}$$

Generative: MAP Gaussian Mixtures

$$\pi_0 = \frac{N_{10}}{N_{10} + N_{20}} \qquad x \in \mathcal{C}_i \sim \mathcal{N}(x|\mu_{i0}, \Sigma_{i0})$$

$$\pi_{MAP} = \frac{N_1 + N_{10}}{N + N_0} = \frac{N_1 + N_{10}}{N_1 + N_2 + N_{10} + N_{20}}$$

$$\begin{cases} {\sum_{iMAP}}^{-1} = {\sum_{iML}}^{-1} + {\sum_{i0}}^{-1} \\ {\sum_{iMAP}}^{-1} \mu_{iMAP} = {\sum_{iML}}^{-1} \mu_{iML} + {\sum_{i0}}^{-1} \mu_{i0} \end{cases}$$

$$\Sigma = \pi \Sigma_1 + (1 - \pi) \Sigma_2$$

Outlines

- Three Approaches to Linear Classification Models
- Approach I: Discriminant Functions
- Least Square Classification
- Fisher Discriminant Function
- Perceptrons
- Approach II: Probabilistic Generative Models
- Approach III: Probabilistic Discriminative Models
- Bayesian Information Criterion

Probabilistic Discriminative Models

- *Discriminative training*: we can maximize the likelihood function defined through the conditional distribution $p(\mathcal{C}_k|\mathbf{x})$

- *Advantages of discriminative approaches*: fewer parameters to be determined

Logistic Regression

- When there are only two classes we can model the conditional probability of the positive class as

$$p(C_1 \mid \mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x} + w_0) \qquad where \quad \sigma(z) = \frac{1}{1+\exp(-z)}$$

- If we use the right error function, something nice happens: The gradient of the logistic and the gradient of the error function cancel each other:

$$E(\mathbf{w}) = -\ln p(\mathbf{t} \mid \mathbf{w}), \qquad \nabla E(\mathbf{w}) = \sum_{n=1}^{N} (y_n - t_n)\, \mathbf{x}_n$$

The Logistic Function

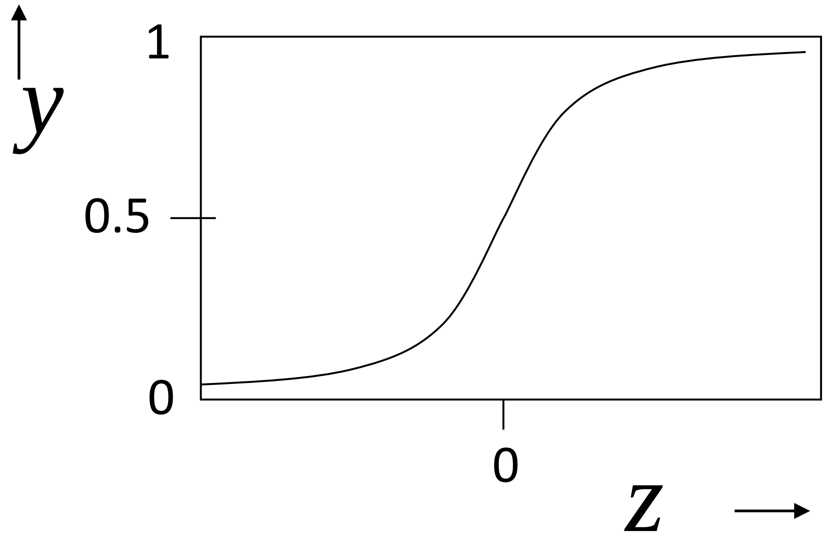
- The output is a smooth function of the inputs and the weights.

$$z = \mathbf{w}^T \mathbf{x} + w_0$$

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\frac{\partial z}{\partial w_i} = x_i \qquad \frac{\partial z}{\partial x_i} = w_i$$

$$\frac{dy}{dz} = y\,(1 - y)$$



It is odd to express it in terms of y.

The Natural Error Function

- To fit a logistic model using maximum likelihood, we need to minimize the negative log probability of the correct answer summed over the training set.

$$E = -\sum_{n=1}^{N} \ln p(t_n \mid y_n) \longleftarrow \boxed{\text{cross-entropy}}$$

$$= -\sum_{n=1}^{N} t_n \ln y_n + (1-t_n) \ln (1-y_n)$$

if t =1

if t =0

$$\frac{\partial E_n}{\partial y_n} = -\frac{t_n}{y_n} + \frac{1-t_n}{1-y_n}$$

$$= \frac{y_n - t_n}{y_n(1-y_n)}$$

error derivative on training case n

The Chain Rule for Error Derivatives

$$z_n = \mathbf{w}^T \mathbf{x}_n + w_0, \qquad \frac{\partial z_n}{\partial \mathbf{w}} = \mathbf{x}_n$$

$$\frac{\partial E_n}{\partial y_n} = \frac{y_n - t_n}{y_n(1 - y_n)}, \qquad \frac{dy_n}{dz_n} = y_n\,(1 - y_n)$$

$$\frac{\partial E_n}{\partial \mathbf{w}} = \frac{\partial E_n}{\partial y_n}\frac{dy_n}{dz_n}\frac{\partial z_n}{\partial \mathbf{w}} = (y_n - t_n)\,\mathbf{x}_n$$

$$\boldsymbol{w}^{new} = \boldsymbol{w}^{old} - (y_n - t_n)\mathbf{x}_n$$

If the step size is taken as 1

Softmax for Two Classes

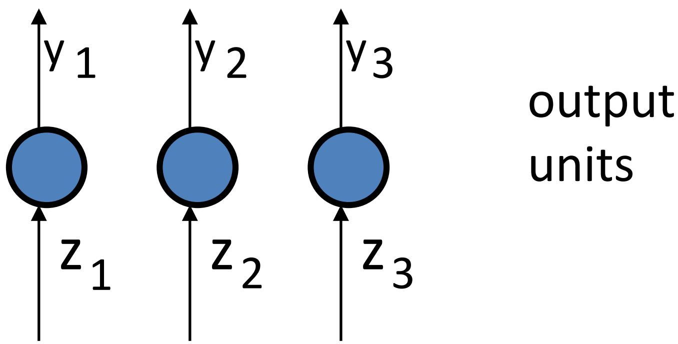
$$y_1 = \frac{e^{z_1}}{e^{z_1} + e^{z_0}} = \frac{1}{1 + e^{-(z_1 - z_0)}}$$

- So the logistic is just a special case that avoids using redundant parameters:
  - ✓ Adding the same constant to both z_1 and z_0 has no effect.
  - ✓ The over-parameterization of the softmax is because the probabilities must add to 1.

Softmax for Multiple Classes

The output units use a non-local non-linearity:

$$y_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$$



$$\frac{\partial y_i}{\partial z_i} = y_i \,(1 - y_i)$$

The natural cost function is the negative log prob of the right answer

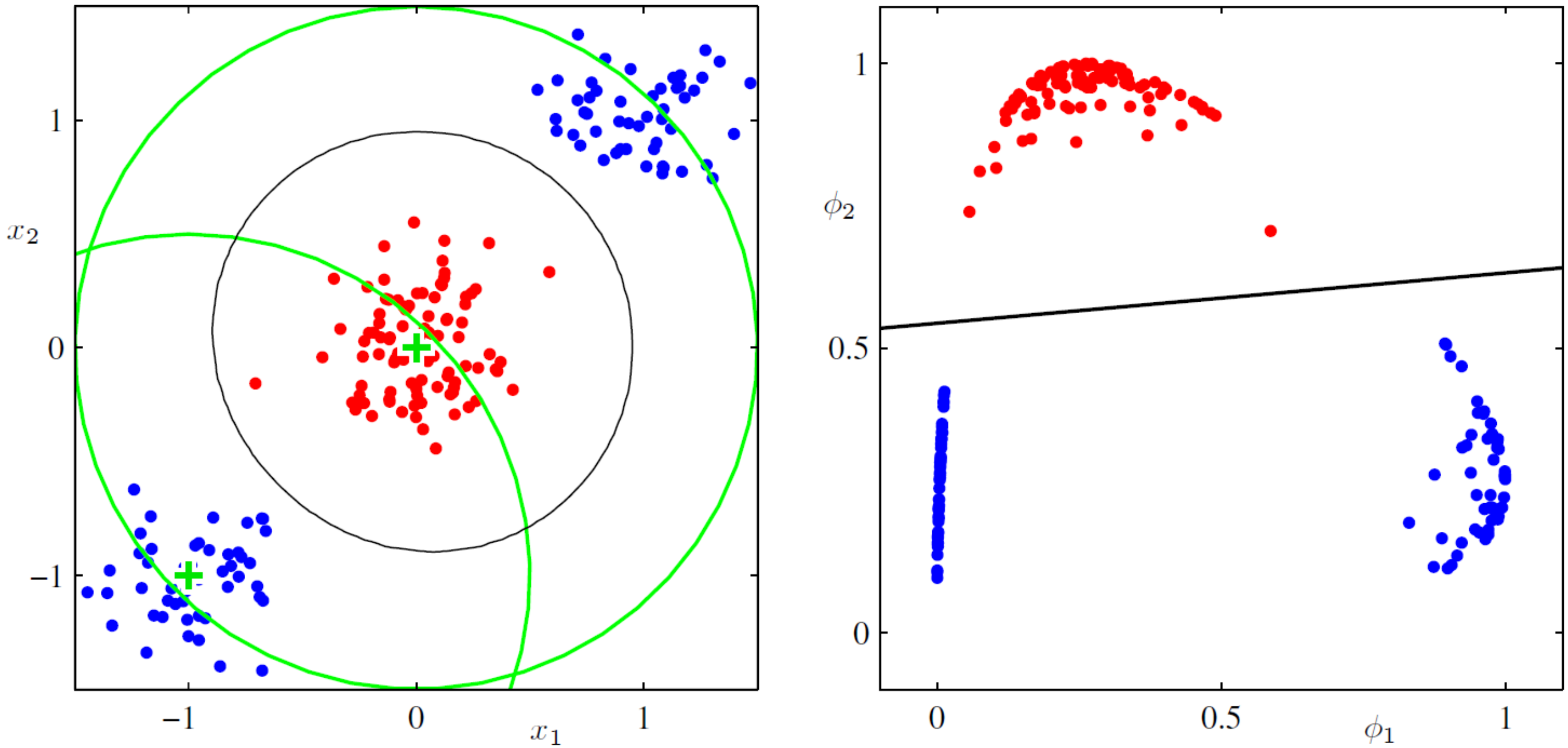
target value

$$E = -\sum_j t_j \ln y_j$$

The steepness of E exactly balances the flatness of the softmax.

$$\frac{\partial E}{\partial z_i} = \sum_j \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial z_i} = y_i - t_i$$

Fixed Basis Functions



Using Gaussian basis functions to achieve “linearly separable” cases

Discriminative: ML Logistic Regression

$$p(C_0|\phi) = y(\phi) = \sigma(w^T\phi) \qquad p(C_1|\phi) = 1 - p(C_0|\phi)$$

where $\dfrac{d\sigma(a)}{da} = \sigma(1-\sigma)$

$$p(t|w) = \prod_{n=1}^{N} {y_n}^{t_n}(1-y_n)^{1-t_n}$$

$$E(w) = -\ln p(t|w) = -\sum_{n=1}^{N}[t_n \ln y_n + (1-t_n)\ln(1-y_n)]$$ Likelihood

$$\nabla E(w) = \sum_{n=1}^{N}(y_n - t_n)\phi_n \qquad H = \nabla\nabla E(w) = \sum_{n=1}^{N} y_n(1-y_n)\phi_n{\phi_n}^T$$

$$w_{ML} \longleftarrow w^{new} = w^{old} - H^{-1}\nabla E(w)$$

$\longleftarrow$ step size taken as H^{-1}： Gauss-Newton Method

Discriminative: ML Logistic Regression

$$\nabla E(w) = \sum_{n=1}^{N} (y_n - t_n)\phi_n = \boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{y} - \boldsymbol{t})$$

$$H = \nabla\nabla E(w) = \sum_{n=1}^{N} y_n(1 - y_n)\phi_n{\phi_n}^T = \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{\Phi}$$

$$w^{new} = w^{old} - H^{-1}\nabla E(w) = \left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{v}$$

where $\boldsymbol{v} = \boldsymbol{\Phi} w^{old} - \boldsymbol{R}^{-1}(\boldsymbol{y} - \boldsymbol{t})$

Discriminative: MAP Logistic Regression

$$p(w) = N(w|m_0, S_0) \qquad p(w|t) \propto p(w)p(t|w)$$

$$E(w) = -\ln p(w|t) = \frac{1}{2}(w - m_0)^T {S_0}^{-1} (w - m_0) - \sum_{n=1}^{N} [t_n \ln y_n + (1 - t_n) \ln(1 - y_n)]$$

$$\nabla E(w) = {S_0}^{-1}(w - m_0) + \sum_{n=1}^{N} (y_n - t_n)\phi_n$$

$$H = \nabla\nabla E(w) = {S_0}^{-1} + \sum_{n=1}^{N} y_n(1 - y_n)\phi_n {\phi_n}^T$$

$$w_{\mathrm{MAP}} \longleftarrow w^{new} = w^{old} - H^{-1}\nabla E(w) \qquad q(w) = N(w|w_{MAP}, H^{-1})$$

$\longleftarrow$ $\boxed{\text{step size taken as } H^{-1}\text{： Gauss-Newton Method}}$

Discriminative: MAP Predictive Distribution

$$\mathbb{E}[z] = \mathbb{E}[w^T\phi] = {w_{MAP}}^T\phi$$

$$\mathrm{var}[z] = \mathrm{var}[w^T\phi] = \phi^T H^{-1}\phi$$

$$y = \sigma(z)$$

$$p(\mathrm{C}_1|\phi^{new}, \boldsymbol{t}) = \mathbb{E}[y] = \int \sigma(z)p(z)dz \simeq \sigma(\kappa(\sigma_z^2)\mu_z)$$

$$\mu_z = {w_{MAP}}^T\phi^{new}$$

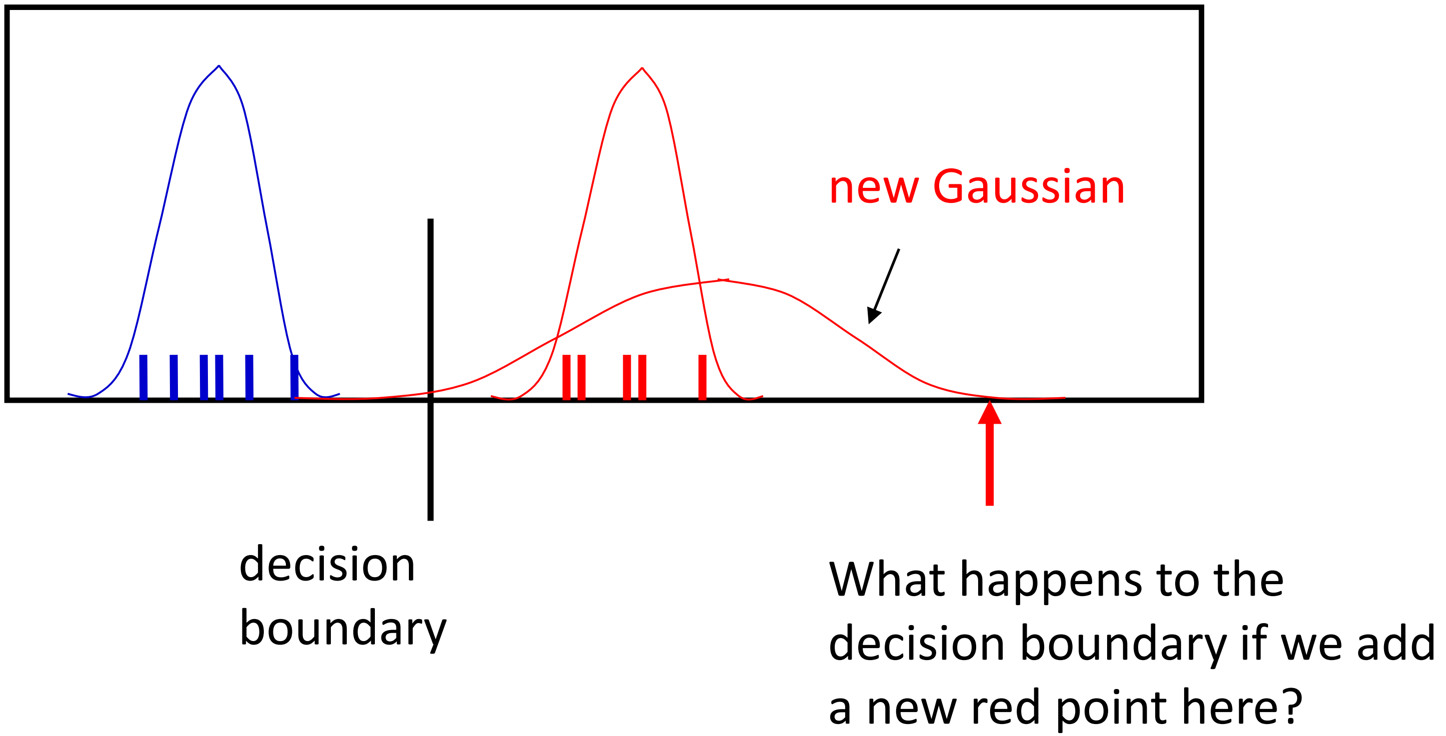
$$\sigma_z^2 = {\phi^{new}}^T H^{-1}\phi^{new}$$

Comparison of Two Approaches

- **Generative approach:** train each model separately to fit the input vectors of that class
  - ✓ Different models can be trained on different cores
  - ✓ It is easy to add a new class without retraining all the other classes
- There are significant advantages when the linear models are harder to train
- **Gaussian Mixture Model**

- **Discriminative approach:** train both models to maximize the probability of getting the labels right
  - ✓ Emphasize the boundary among different classes
  - ✓ Fewer parameters to be determined
- There are significant advantages when the linear models are easy to train
- **Logistic Regression Model**

Comparison of Two Approaches



For generative fitting, the red mean moves rightwards but the decision boundary moves leftwards!

Outlines

- Three Approaches to Linear Classification
- Approach I: Discriminant Functions
- Least Square Classification
- Fisher's Discriminants
- Perceptrons
- Approach II: Probabilistic Generative Models
- Approach III: Probabilistic Discriminative Models
- Bayesian Information Criterion

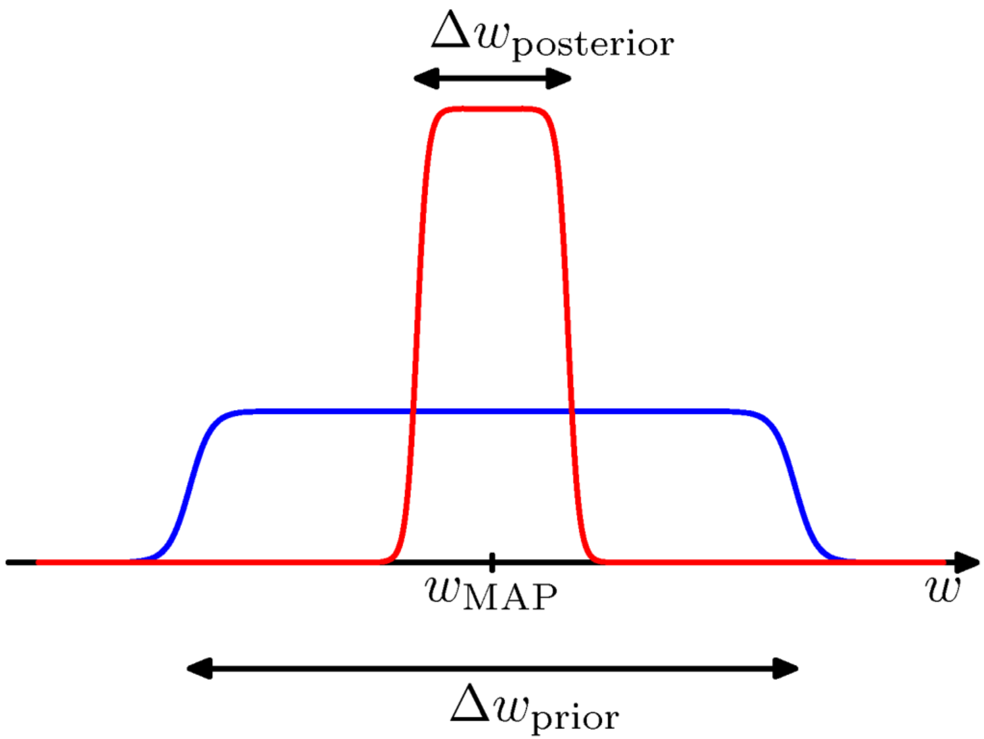
Bayesian Model Comparison (1)

For a given model with a single parameter, w, consider the approximation

$$p(\mathcal{D}) = \int p(\mathcal{D}|w)p(w)\,\mathrm{d}w$$

$$\simeq \quad p(\mathcal{D}|w_{\mathrm{MAP}})\frac{\Delta w_{\mathrm{posterior}}}{\Delta w_{\mathrm{prior}}}$$

where the posterior is assumed to be sharply peaked.



Bayesian Model Comparison (2)

Taking logarithms, we obtain

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|w_{\mathrm{MAP}}) + \underbrace{\ln\left(\frac{\Delta w_{\mathrm{posterior}}}{\Delta w_{\mathrm{prior}}}\right)}_{\text{Negative}}.$$

With $\mathtt{M}$ parameters, all assumed to have the same ratio $\Delta w_{\mathrm{posterior}}/\Delta w_{\mathrm{prior}}$, we get

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\mathbf{w}_{\mathrm{MAP}}) + \underbrace{M\ln\left(\frac{\Delta w_{\mathrm{posterior}}}{\Delta w_{\mathrm{prior}}}\right)}_{\text{Negative and linear in } \mathtt{M}.}.$$

Bayesian Information Criterion

Akaike Information Criterion (AIC)

$$\ln p(\mathcal{D}|\mathbf{w}_{\mathrm{ML}}) - M$$

Bayesian Information Criterion (BIC)

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\boldsymbol{\theta}_{\mathrm{MAP}}) - \frac{1}{2} M \ln N$$

M： model order； N： data number

Laplace Approximation

$$\ln f(\mathbf{z}) \simeq \ln f(\mathbf{z}_0) - \frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^{\mathrm{T}}\mathbf{A}(\mathbf{z}-\mathbf{z}_0)$$

where $$\mathbf{A} = -\nabla\nabla \ln f(\mathbf{z})\big|_{\mathbf{z}=\mathbf{z}_0}$$

$$\begin{aligned}
Z &= \int f(\mathbf{z})\,\mathrm{d}\mathbf{z} \\
&\simeq f(\mathbf{z}_0)\int \exp\left\{-\frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^{\mathrm{T}}\mathbf{A}(\mathbf{z}-\mathbf{z}_0)\right\}\mathrm{d}\mathbf{z} \\
&= f(\mathbf{z}_0)\frac{(2\pi)^{M/2}}{|\mathbf{A}|^{1/2}}
\end{aligned}$$

Model Evaluation

Let $Z = p(\mathcal{D})$ $f(\boldsymbol{\theta}) = p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})$

$$p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}$$

Then, the evidence is given by

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\boldsymbol{\theta}_{\mathrm{MAP}}) + \underbrace{\ln p(\boldsymbol{\theta}_{\mathrm{MAP}}) + \frac{M}{2}\ln(2\pi) - \frac{1}{2}\ln|\mathbf{A}|}_{\text{Occam factor}}$$

penalizes model complexity

where

$$\mathbf{A} = -\nabla\nabla \ln p(\mathcal{D}|\boldsymbol{\theta}_{\mathrm{MAP}})p(\boldsymbol{\theta}_{\mathrm{MAP}}) = -\nabla\nabla \ln p(\boldsymbol{\theta}_{\mathrm{MAP}}|\mathcal{D})$$

Summary

- Three Approaches to Linear Classification
- Approach I: Discriminant Functions
- Least Square Classification
- Fisher's Discriminants
- Perceptrons
- Approach II: Probabilistic Generative Models
- Approach III: Probabilistic Discriminative Models
- Bayesian Information Criterion