Lecture Slides for

INTRODUCTION TO

Machine Learning

ETHEM ALPAYDIN

Edited for CS 536 Fall 2005 – Rutgers University
Ahmed Elgammal

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml

CHAPTER 4:

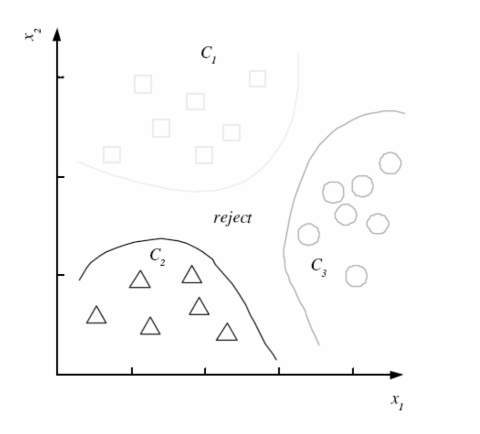
Parametric Methods

CHAPTER 5:

Multivariate Methods

Parametric Classification

- We would like to estimate $p(x|C_i)$ and $p(C_i)$ to be able to estimate the posterior $p(C_i|x)$



Parametric Estimation

- $\mathcal{X} = \{ x^t \}_t$ where $x^t \sim p(x)$
- $\mathcal{X}$ is independent and identically distributed (iid) sample
- Parametric estimation:

  Assume a form for $p(x \mid \theta)$ and estimate θ, its sufficient statistics, using $\mathcal{X}$

  e.g., $\mathcal{N}(\mu, \sigma^2)$ where $\theta = \{\mu, \sigma^2\}$

Maximum Likelihood Estimation

- Likelihood of θ given the sample $\mathcal{X}$

  $l(\theta|\mathcal{X}) = p(\mathcal{X}|\theta) = \prod_t p(x^t|\theta)$

- Log likelihood

  $\mathcal{L}(\theta|\mathcal{X}) = \log l(\theta|\mathcal{X}) = \sum_t \log p(x^t|\theta)$

- Maximum likelihood estimator (MLE)

  $\theta^* = \text{argmax}_\theta \mathcal{L}(\theta|\mathcal{X})$

What is the parameter(s) of the distribution that maximizes the likelihood of the data sample

Examples: Bernoulli/Multinomial

- Bernoulli: Two states, failure/success, x in {0,1}

  $P(x) = p_o^x (1 - p_o)^{(1-x)}$

  $\mathcal{L}(p_o|\mathcal{X}) = \log \prod_t p_o^{x^t} (1 - p_o)^{(1-x^t)}$

  MLE: $p_o = \sum_t x^t / N$

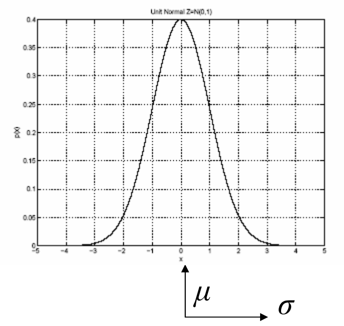
- Multinomial: K>2 states, x_i in {0,1}

  $P(x_1,x_2,...,x_K) = \prod_i p_i^{x_i}$

  $\mathcal{L}(p_1,p_2,...,p_K|\mathcal{X}) = \log \prod_t \prod_i p_i^{x_i^t}$

  MLE: $p_i = \sum_t x_i^t / N$

Gaussian (Normal) Distribution



- $p(x) = \mathcal{N}(\mu, \sigma^2)$

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

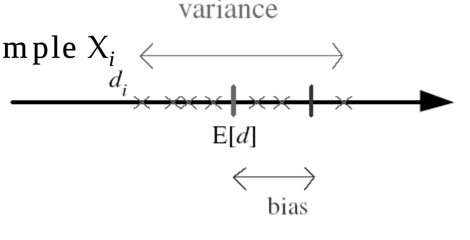
- MLE for μ and σ^2:

$$m = \frac{\sum_t x^t}{N}$$

$$s^2 = \frac{\sum_t (x^t - m)^2}{N}$$

Bias and Variance

How to evaluate the quality of an estimator?
Unknown parameter θ
Estimator $d_i = d(X_i)$ on sample X_i



Bias: $b_\theta(d) = E[d] - \theta$
Variance: $E[(d - E[d])^2]$

Mean square error:
$r(d,\theta) = E[(d-\theta)^2]$
$= (E[d] - \theta)^2 + E[(d - E[d])^2]$
$= \text{Bias}^2 + \text{Variance}$

Bayes' Estimator

- Treat θ as a random var with prior $p(\theta)$
- Bayes' rule: $p(\theta|\mathcal{X}) = p(\mathcal{X}|\theta)\, p(\theta) / p(\mathcal{X})$
- Full: $p(x|\mathcal{X}) = \int p(x|\theta)\, p(\theta|\mathcal{X})\, d\theta$
- Maximum a Posteriori (MAP): $\theta_{MAP} = \operatorname{argmax}_\theta p(\theta|\mathcal{X})$
- Maximum Likelihood (ML): $\theta_{ML} = \operatorname{argmax}_\theta p(\mathcal{X}|\theta)$
- Bayes': $\theta_{Bayes'} = E[\theta|\mathcal{X}] = \int \theta\, p(\theta|\mathcal{X})\, d\theta$

Bayes' Estimator: Example

- $x^t \sim \mathcal{N}(\theta, \sigma_o^2)$ and $\theta \sim \mathcal{N}(\mu, \sigma^2)$
- $\theta_{ML} = m$
- $\theta_{MAP} = \theta_{Bayes'} =$

$$E[\theta|\mathcal{X}] = \frac{N/\sigma_0^2}{N/\sigma_0^2 + 1/\sigma^2} m + \frac{1/\sigma^2}{N/\sigma_0^2 + 1/\sigma^2} \mu$$

Lecture Notes for E Alpaydın 2004 Introduction to Machine Learning © The MIT Press (V1.1)

Parametric Classification

$$g_i(x) = p(x \mid C_i)P(C_i)$$
or equivalently
$$g_i(x) = \log p(x \mid C_i) + \log P(C_i)$$

$$p(x \mid C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left[-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right]$$
$$g_i(x) = -\frac{1}{2}\log 2\pi - \log \sigma_i - \frac{(x-\mu_i)^2}{2\sigma_i^2} + \log P(C_i)$$

- Given the sample $\mathcal{X} = \{x^t, r^t\}_{t=1}^N$

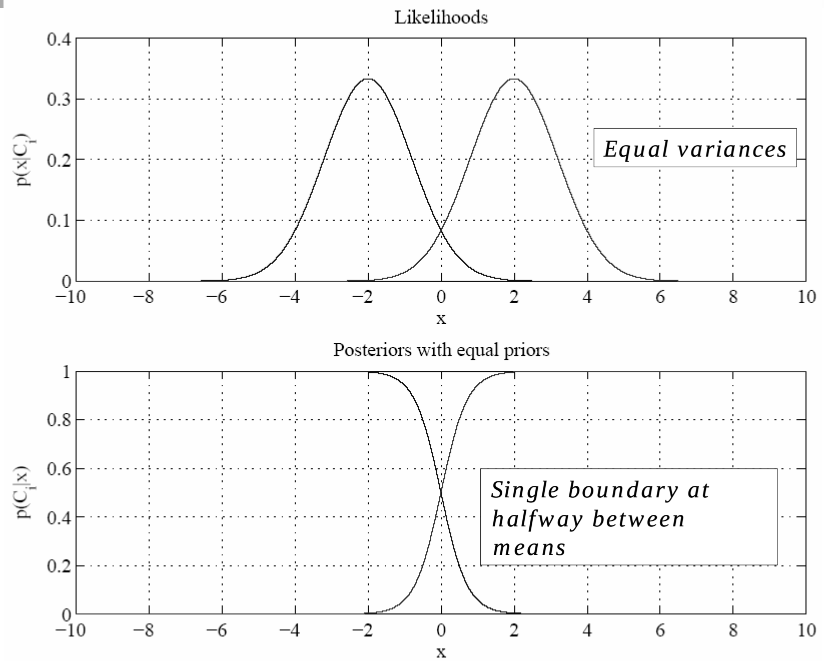
$$x \in \Re \qquad r_i^t = \begin{cases} 1 \text{ if } x^t \in C_i \\ 0 \text{ if } x^t \in C_j, j \neq i \end{cases}$$

- ML estimates are

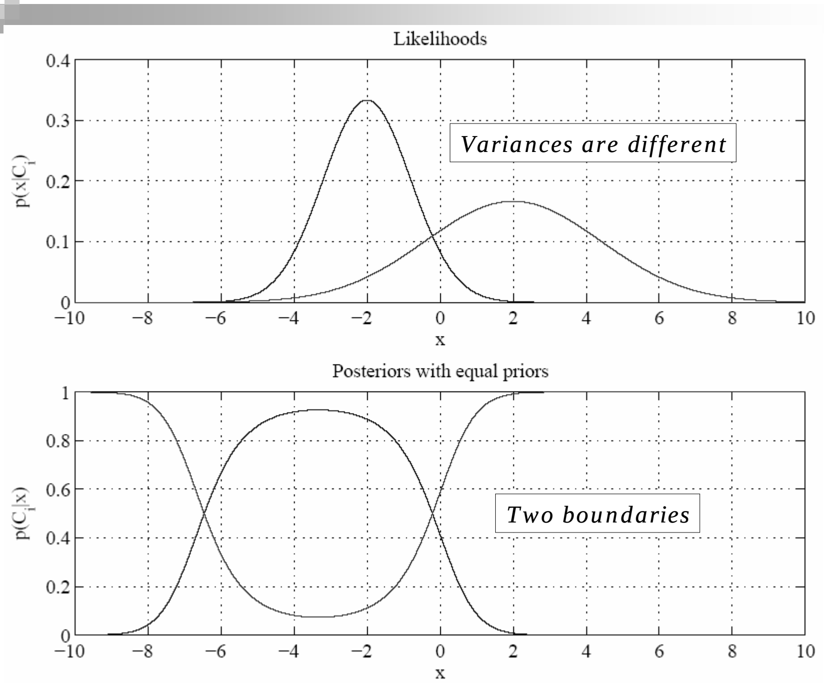
$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N} \quad m_i = \frac{\sum_t x^t r_i^t}{\sum_t r_i^t} \quad s_i^2 = \frac{\sum_t (x^t - m_i)^2 r_i^t}{\sum_t r_i^t}$$

- Discriminant becomes

$$g_i(x) = -\frac{1}{2}\log 2\pi - \log s_i - \frac{(x-m_i)^2}{2s_i^2} + \log \hat{P}(C_i)$$



Likelihoods
p(x|C_i)
x
Equal variances
Posteriors with equal priors
p(C_i|x)
x
Single boundary at halfway between means



Likelihoods
p(x|C_i)
x
Variances are different
Posteriors with equal priors
p(C_i|x)
x
Two boundaries

Multivariate Data

- Multiple measurements (sensors)
- d inputs/features/attributes: d-variate
- N instances/observations/examples

$$\mathbf{X} = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_d^1 \\ X_1^2 & X_2^2 & \cdots & X_d^2 \\ \vdots & & & \\ X_1^N & X_2^N & \cdots & X_d^N \end{bmatrix}$$

Multivariate Parameters

$$\text{Mean}: E[\boldsymbol{x}] = \boldsymbol{\mu} = [\mu_1, \ldots, \mu_d]^T$$
$$\text{Covariance}: \sigma_{ij} \equiv \text{Cov}(X_i, X_j)$$
$$\text{Correlation}: \text{Corr}(X_i, X_j) \equiv \rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$$

$$\Sigma \equiv \text{Cov}(\boldsymbol{X}) = E\left[(\boldsymbol{X} - \mu)(\boldsymbol{X} - \mu)^T\right] = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & & & \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{bmatrix}$$

Parameter Estimation

$$\text{Sample mean } \mathbf{m} : m_i = \frac{\sum_{t=1}^{N} x_i^t}{N}, i = 1,\ldots,d$$

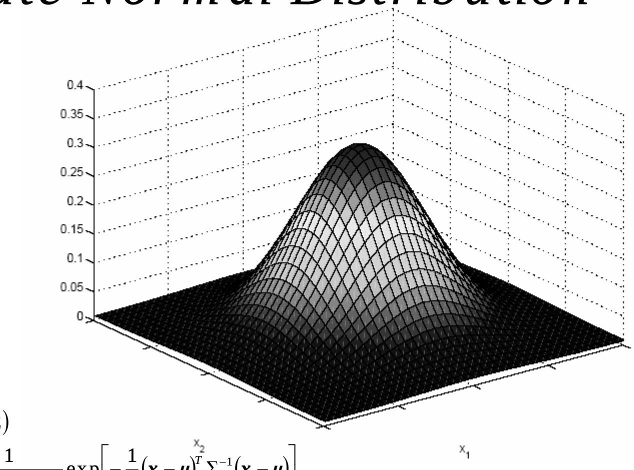
$$\text{Covariance matrix } \mathbf{S} : s_{ij} = \frac{\sum_{t=1}^{N} \left(x_i^t - m_i\right)\left(x_j^t - m_j\right)}{N}$$

$$\text{Correlation matrix } \mathbf{R} : r_{ij} = \frac{s_{ij}}{s_i s_j}$$

Estimation of Missing Values

- What to do if certain instances have missing attributes?
- Ignore those instances: not a good idea if the sample is small
- Use 'missing' as an attribute: may give information
- Imputation: Fill in the missing value
  - Mean imputation: Use the most likely value (e.g., mean)
  - Imputation by regression: Predict based on other attributes

Multivariate Normal Distribution



$$\boldsymbol{x} \sim N_d(\boldsymbol{\mu}, \Sigma)$$

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]$$

Multivariate Normal Distribution

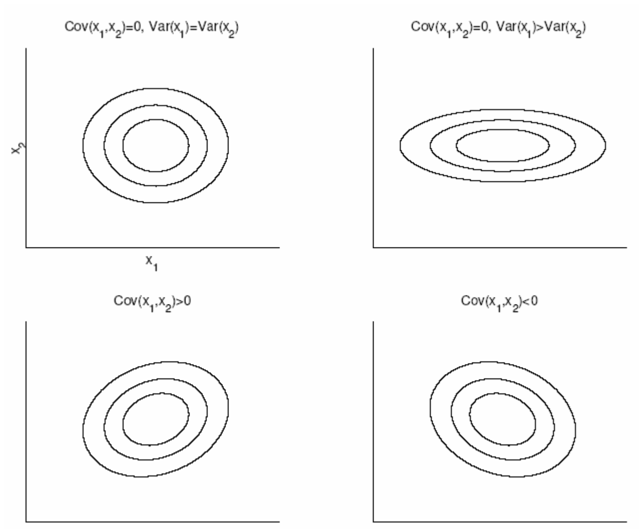
- Mahalanobis distance: $(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})$
  measures the distance from $\boldsymbol{x}$ to $\boldsymbol{\mu}$ in terms of $\boldsymbol{\Sigma}$ (normalizes for difference in variances and correlations)
- Bivariate: $d = 2$

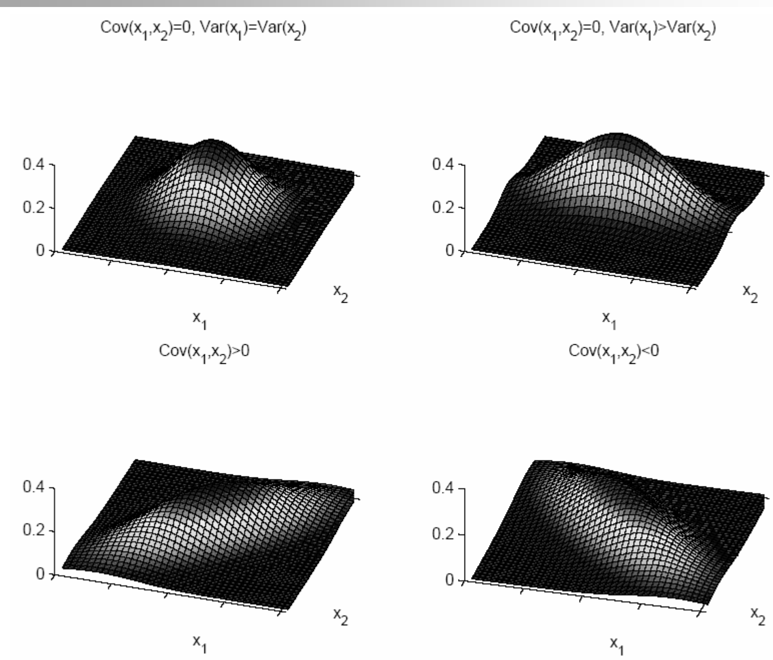
$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[-\frac{1}{2(1-\rho^2)}\left(z_1^2 - 2\rho z_1 z_2 + z_2^2\right)\right]$$

$$z_i = (x_i - \mu_i)/\sigma_i$$

Bivariate Normal





Independent Inputs: Naive Bayes

- If x_i are independent, offdiagonals of $\sum$ are 0, Mahalanobis distance reduces to weighted (by $1/\sigma_i$) Euclidean distance:

$$p(\mathbf{x}) = \prod_{i=1}^{d} p_i(x_i) = \frac{1}{(2\pi)^{d/2}\prod_{i=1}^{d}\sigma_i}\exp\left[-\frac{1}{2}\sum_{i=1}^{d}\left(\frac{x_i-\mu_i}{\sigma_i}\right)^2\right]$$

- If variances are also equal, reduces to Euclidean distance

Parametric Classification

- If $p(\mathbf{x} \mid C_i) \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$

$$p(\mathbf{x} \mid C_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T\Sigma_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right]$$

- Discriminant functions are

$$\begin{aligned} g_i(\mathbf{x}) &= \log p(\mathbf{x} \mid C_i) + \log P(C_i) \\ &= -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_i| - \frac{1}{2}(\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i) + \log P(C_i) \end{aligned}$$

Lecture Notes for E Alpaydın 2004 Introduction to Machine Learning © The MIT Press (V1.1)

Estimation of Parameters

$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N}$$
$$\boldsymbol{m}_i = \frac{\sum_t r_i^t \boldsymbol{x}^t}{\sum_t r_i^t}$$
$$\mathbf{S}_i = \frac{\sum_t r_i^t (\boldsymbol{x}^t - \boldsymbol{m}_i)(\boldsymbol{x}^t - \boldsymbol{m}_i)^T}{\sum_t r_i^t}$$

$$g_i(\boldsymbol{x}) = -\frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}(\boldsymbol{x} - \boldsymbol{m}_i)^T \mathbf{S}_i^{-1}(\boldsymbol{x} - \boldsymbol{m}_i) + \log \hat{P}(C_i)$$

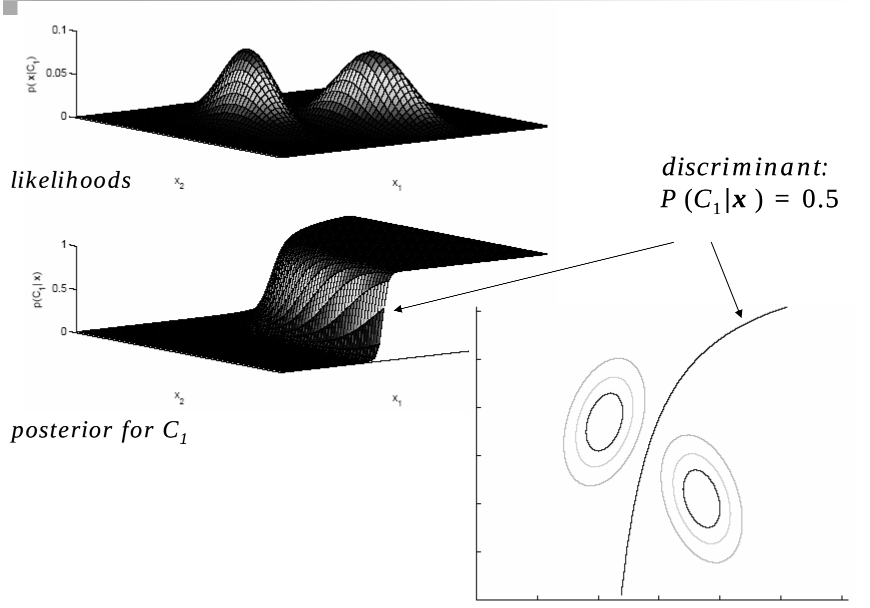
Different $\mathbf{S}_i$

- Quadratic discriminant

$$g_i(\boldsymbol{x}) = -\frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}\left(\boldsymbol{x}^T\mathbf{S}_i^{-1}\boldsymbol{x} - 2\boldsymbol{x}^T\mathbf{S}_i^{-1}\boldsymbol{m}_i + \boldsymbol{m}_i^T\mathbf{S}_i^{-1}\boldsymbol{m}_i\right) + \log\hat{P}(C_i)$$
$$= \boldsymbol{x}^T\mathbf{W}_i\boldsymbol{x} + \boldsymbol{w}_i^T\boldsymbol{x} + w_{i0}$$

where

$$\mathbf{W}_i = -\frac{1}{2}\mathbf{S}_i^{-1}$$
$$\boldsymbol{w}_i = \mathbf{S}_i^{-1}\boldsymbol{m}_i$$
$$w_{i0} = -\frac{1}{2}\boldsymbol{m}_i^T\mathbf{S}_i^{-1}\boldsymbol{m}_i - \frac{1}{2}\log|\mathbf{S}_i| + \log\hat{P}(C_i)$$



Common Covariance Matrix $\mathbf{S}$

- Shared common sample covariance **S**

$$\mathbf{S} = \sum_i \hat{P}(C_i)\mathbf{S}_i$$

- Discriminant reduces to

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T \mathbf{S}^{-1}(\mathbf{x}-\mathbf{m}_i) + \log \hat{P}(C_i)$$

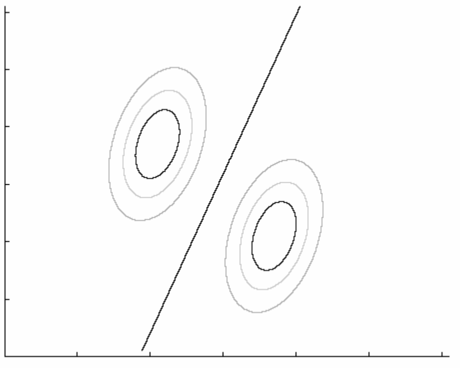
which is a linear discriminant

$$g_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

where

$$\mathbf{w}_i = \mathbf{S}^{-1}\mathbf{m}_i \quad w_{i0} = -\frac{1}{2}\mathbf{m}_i^T \mathbf{S}^{-1}\mathbf{m}_i + \log \hat{P}(C_i)$$

Common Covariance Matrix **S**



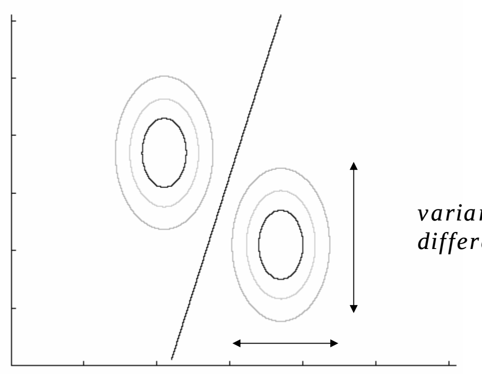
Diagonal **S**

- When x_j $j = 1,..d$, are independent, $\sum$ is diagonal
  $p(\mathbf{x}|C_i) = \prod_j p(x_j|C_i)$ (Naive Bayes' assumption)

$$g_i(\mathbf{x}) = -\frac{1}{2}\left(\frac{x_j^t - m_{ij}}{s_j}\right)^2 + \log \hat{P}(C_i)$$

Classify based on weighted Euclidean distance (in s_j units) to the nearest mean

*Diagonal **S***



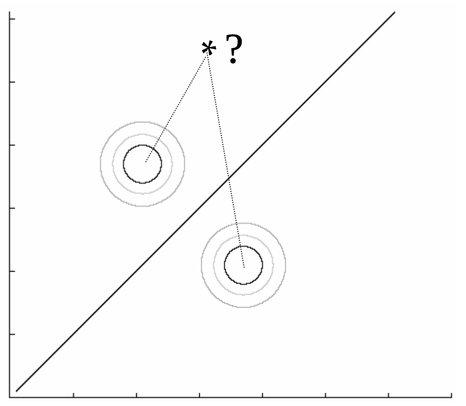
*Diagonal **S**, equal variances*

- Nearest mean classifier: Classify based on Euclidean distance to the nearest mean

$$g_i(\boldsymbol{x}) = -\frac{\|\boldsymbol{x} - \boldsymbol{m}_i\|^2}{2s^2} + \log \hat{P}(C_i)$$
$$= -\frac{1}{2s^2}\sum_{j=1}^{d}\left(x_j^t - m_{ij}\right)^2 + \log \hat{P}(C_i)$$

- Each mean can be considered a prototype or template and this is template matching

*Diagonal **S**, equal variances*



Model Selection

Assumption	*Covariance matrix*	*No of parameters*
Shared, Hyperspheric	$\mathbf{S}_i=\mathbf{S}=s^2\mathbf{I}$	1
Shared, Axis-aligned	$\mathbf{S}_i=\mathbf{S}$, with $s_{ij}=0$	d
Shared, Hyperellipsoidal	$\mathbf{S}_i=\mathbf{S}$	$d(d+1)/2$
Different, Hyperellipsoidal	$\mathbf{S}_i$	$K\ d(d+1)/2$

- As we increase complexity (less restricted **S**), bias decreases and variance increases
- Assume simple models (allow some bias) to control variance (regularization)

Discrete Features

- Binary features: $p_{ij} \equiv p(x_j = 1 \mid C_i)$

  if x_j are independent (Naive Bayes')

  $$p(x \mid C_i) = \prod_{j=1}^{d} p_{ij}^{x_j} (1 - p_{ij})^{(1-x_j)}$$

  the discriminant is linear

  $$\begin{aligned} g_i(\mathbf{x}) &= \log p(\mathbf{x} \mid C_i) + \log P(C_i) \\ &= \sum_j \left[x_j \log p_{ij} + (1 - x_j) \log (1 - p_{ij}) \right] + \log P(C_i) \end{aligned}$$

  Estimated parameters $\hat{p}_{ij} = \frac{\sum_t x_j^t r_i^t}{\sum_t r_i^t}$

Discrete Features

- Multinomial (1-of-n_j) features: $x_j \in \{v_1, v_2, \ldots, v_{n_j}\}$

  $$p_{ijk} \equiv p(z_{jk} = 1 \mid C_i) = p(x_j = v_k \mid C_i)$$

  if x_j are independent

  $$p(\mathbf{x} \mid C_i) = \prod_{j=1}^{d} \prod_{k=1}^{n_j} p_{ijk}^{z_{jk}}$$

  $$g_i(\mathbf{x}) = \sum_j \sum_k z_{jk} \log p_{ijk} + \log P(C_i)$$

  $$\hat{p}_{ijk} = \frac{\sum_t z_{jk}^t r_i^t}{\sum_t r_i^t}$$