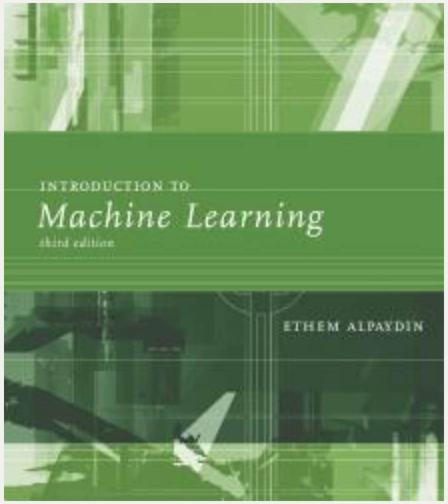


Lecture Slides for

INTRODUCTION TO MACHINE LEARNING

3RD EDITION

ETHEM ALPAYDIN

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml3e

CHAPTER 5:

Multivariate Methods

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

Mean: $E[\mathbf{x}] = \boldsymbol{\mu} = [\mu_1, \ldots, \mu_d]^T$

Covariance: $\sigma_{ij} \equiv \mathrm{Cov}(X_i, X_j)$

Correlation: $Corr(X_i, X_j) \equiv \rho_{ij} = \dfrac{\sigma_{ij}}{\sigma_i \sigma_j}$

The correlation between variables X_i and X_j is a statistic normalized be tween −1 and +1.

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

Parameter Estimation

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

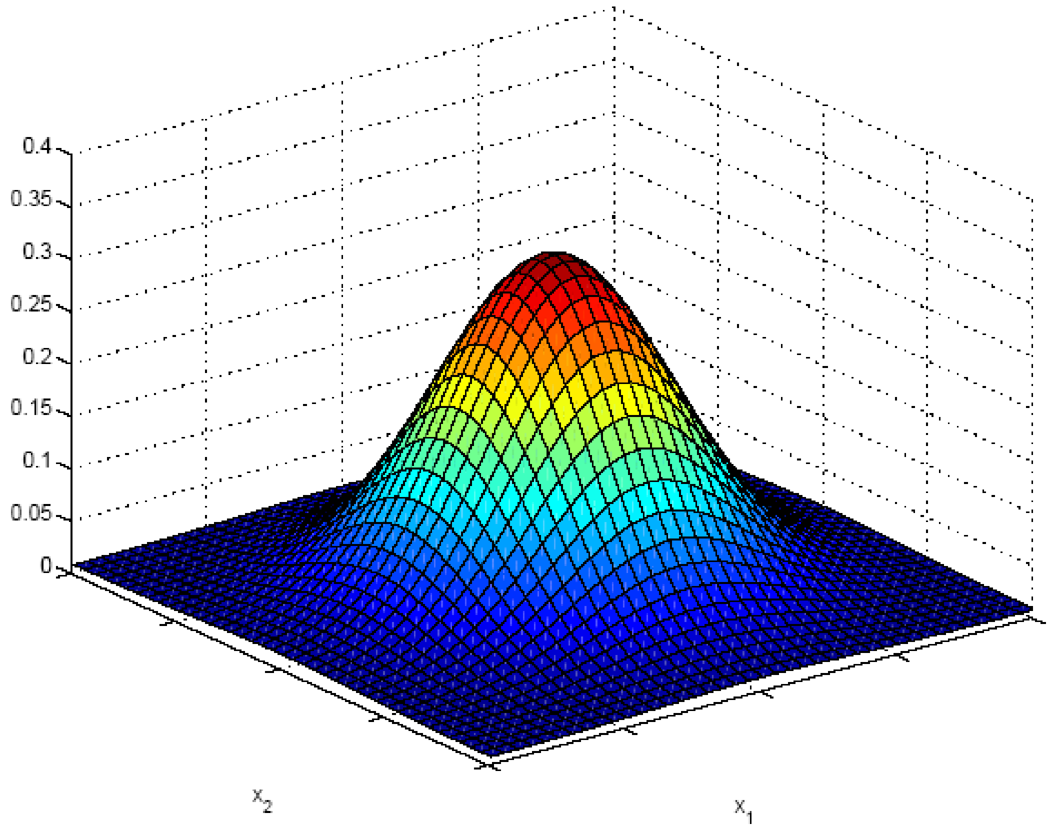
Covariance matrix $\mathbf{S}: s_{ij} = \frac{\sum_{t=1}^{N} \left(x_i^t - m_i\right)\left(x_j^t - m_j\right)}{N}$

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



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

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

Multivariate Normal Distribution

- Mahalanobis distance: $(\boldsymbol{x}-\boldsymbol{\mu})^T \sum^{-\mathbf{1}} (\boldsymbol{x}-\boldsymbol{\mu})$

  measures the distance from $\boldsymbol{x}$ to $\boldsymbol{\mu}$ in terms of $\sum$ (normalizes for difference in variances and correlations)

- Bivariate: $d = 2$

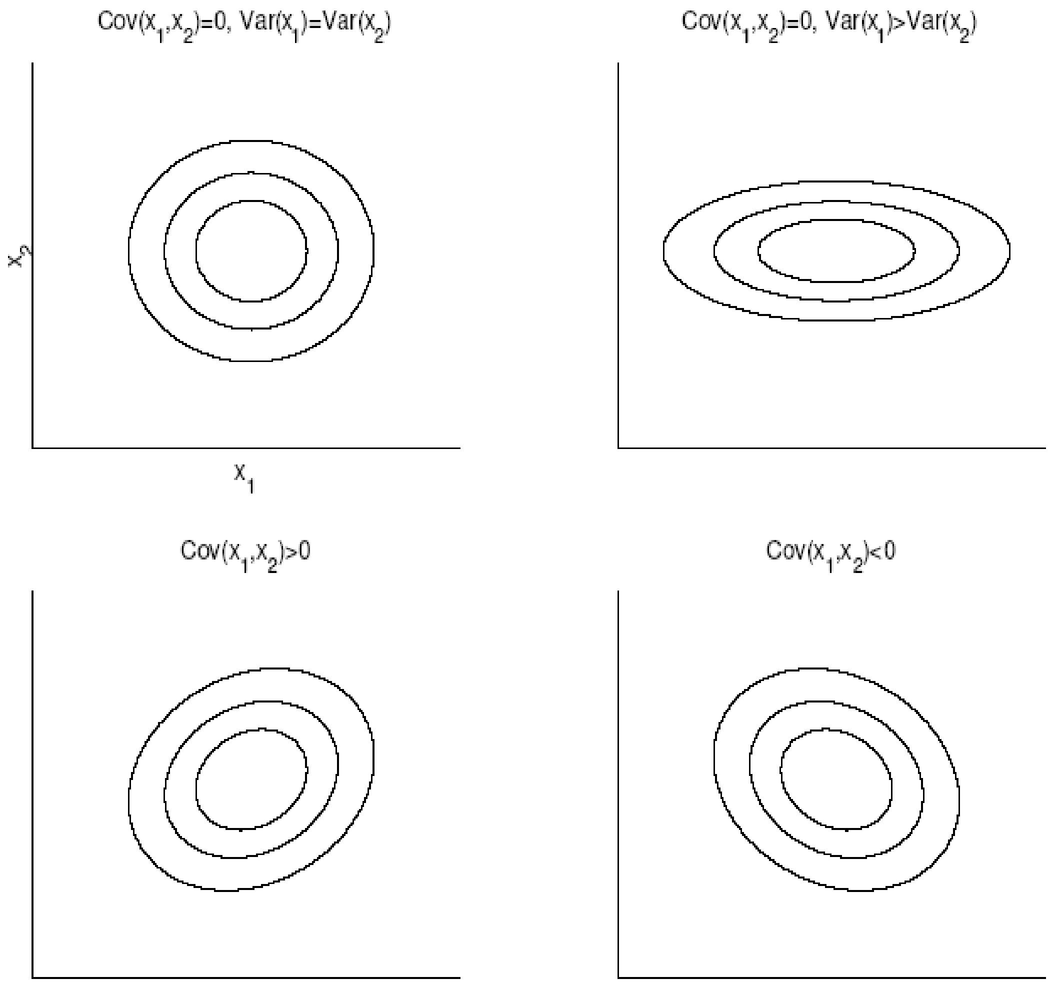
$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

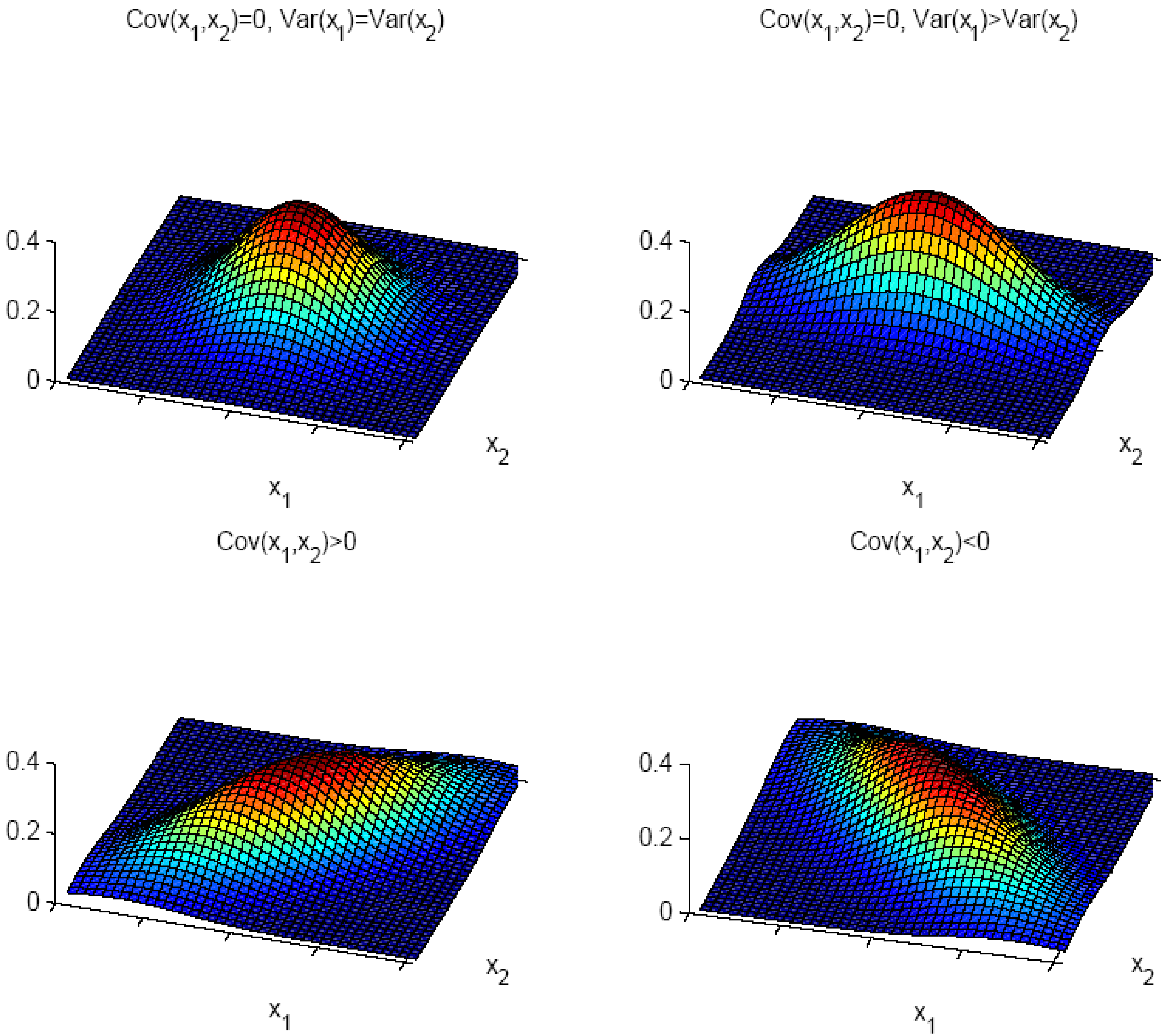
$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[-\frac{1}{2(1-\rho^2)}\left(z_1^2 - 2\rho z_1 z_2 + z_2^2\right)\right]$$

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

z-normalization

Bivariate Normal




Cov(x_1,x_2)=0, Var(x_1)=Var(x_2)
Cov(x_1,x_2)=0, Var(x_1)>Var(x_2)
Cov(x_1,x_2)>0
Cov(x_1,x_2)<0
0.4
0.2
0
x_1
x_2

Independent Inputs: Naive Bayes

- If x_i are independent, off-diagonals 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

Another Property of Normal Dist.

- The projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is: $z = \boldsymbol{w}^T\boldsymbol{x}$
- $z=w_1x_1+w_2x_2+\ldots+w_dx_d$
- $\boldsymbol{x}\sim N_d(\boldsymbol{\mu}, \sum)$ and $\boldsymbol{w} \in R^d$
- $E(\boldsymbol{w}^T\boldsymbol{x})=\boldsymbol{w}^T E(\boldsymbol{x})= \boldsymbol{w}^T \boldsymbol{\mu}$
- $\mathrm{Var}(z) = \mathrm{Var}(\boldsymbol{w}^T\boldsymbol{x}) = E[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})^2]$

$$= E[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})]$$

$$= E[\boldsymbol{w}^T(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{w}]$$ ⟵Note: $\boldsymbol{A}^T\boldsymbol{B}=\boldsymbol{B}^T\boldsymbol{A}$

$$= \mathbf{w}^{\mathrm{T}} E[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T]\boldsymbol{w} = \boldsymbol{w}^T \sum \boldsymbol{w}$$

- In general case, if $\boldsymbol{W}$ is $d\times k$ matrix with rank $k<d$

$$\boldsymbol{z} = \boldsymbol{W}^T\boldsymbol{x}\sim N_k(\boldsymbol{W}^T\boldsymbol{\mu}, \boldsymbol{W}^T \sum \boldsymbol{W})$$

Parametric Classification

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

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

- Discriminant functions

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

Estimation of Parameters

$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N}$$

$$\mathbf{m}_i = \frac{\sum_t r_i^t \mathbf{x}^t}{\sum_t r_i^t}$$

$$\mathbf{S}_i = \frac{\sum_t r_i^t (\mathbf{x}^t - \mathbf{m}_i)(\mathbf{x}^t - \mathbf{m}_i)^T}{\sum_t r_i^t}$$

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

Different $\mathbf{S}_i$

- Quadratic discriminant

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

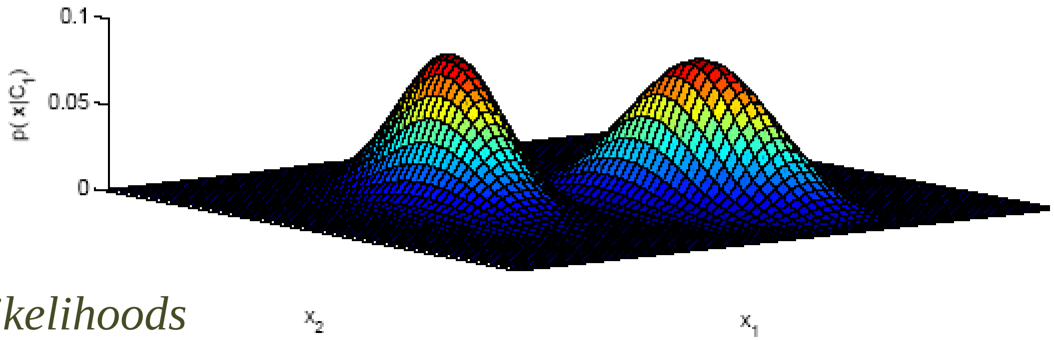
$$= \mathbf{x}^T\mathbf{W}_i\mathbf{x} + \mathbf{w}_i^T\mathbf{x} + w_{i0}$$

where

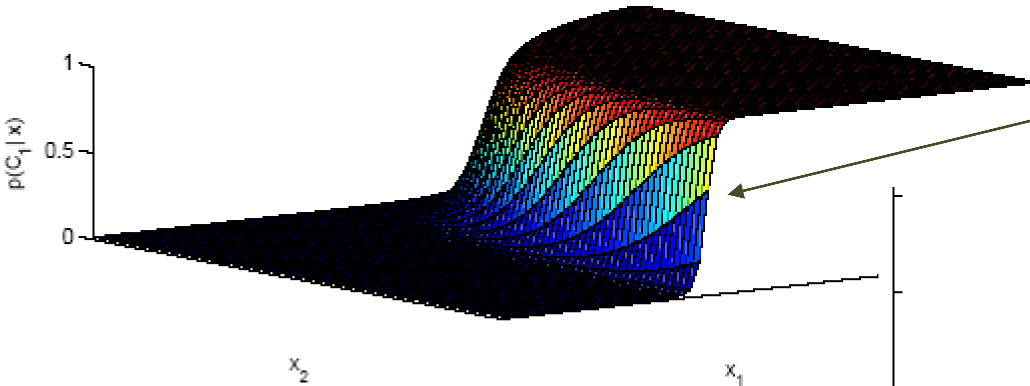
$$\mathbf{W}_i = -\frac{1}{2}\mathbf{S}_i^{-1}$$

$$\mathbf{w}_i = \mathbf{S}_i^{-1}\mathbf{m}_i$$

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

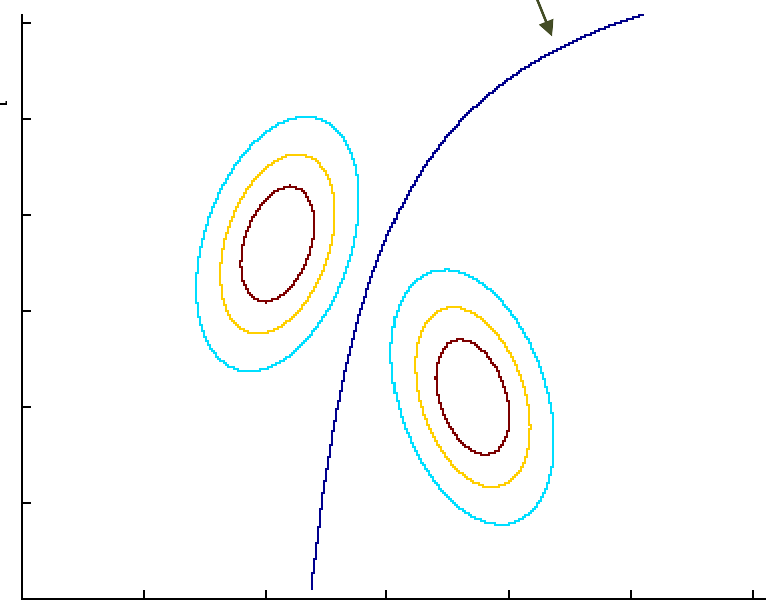


likelihoods



posterior for C_1

discriminant:
$P\,(C_1|\boldsymbol{x}) = 0.5$



Common Covariance Matrix **S**

- Shared common sample covariance **S**

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

- Discriminant reduces to

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

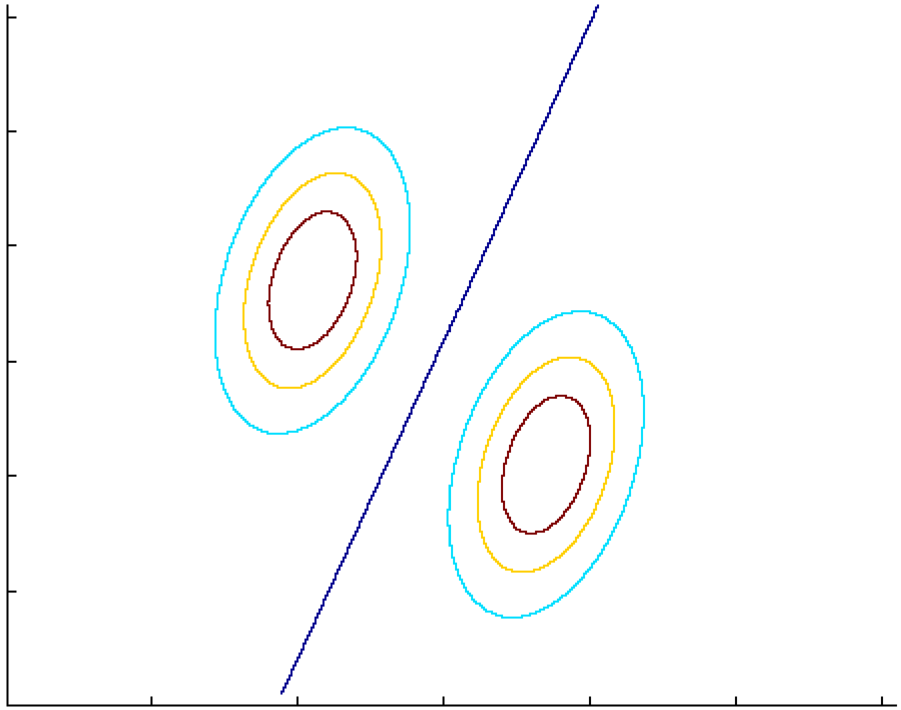
which is a linear discriminant

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

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**



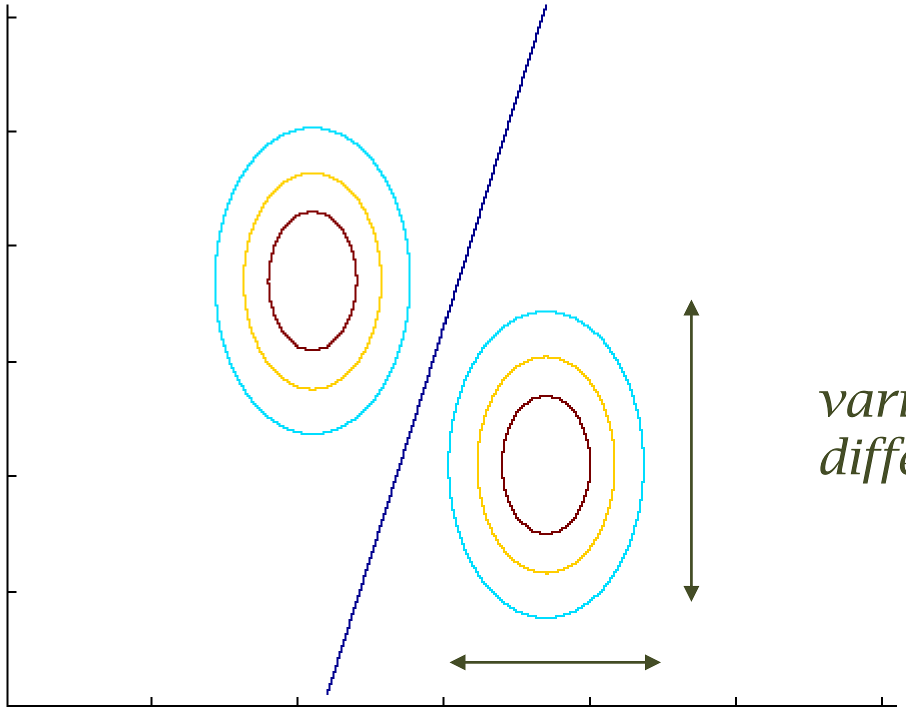
Diagonal **S**

- 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)

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

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

Diagonal S



variances may be different

Diagonal **S**, equal variances

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

$$g_i(\mathbf{x}) = -\frac{\left\|\mathbf{x}-\mathbf{m}_i\right\|^2}{2s^2} + \log \hat{P}(C_i)$$

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

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

$$g_i(\mathbf{x}) = -\left\|\mathbf{x}-\mathbf{m}_i\right\|^2 = -(\mathbf{x}-\mathbf{m}_i)^T(\mathbf{x}-\mathbf{m}_i)$$

$$= -\left(\mathbf{x}^T\mathbf{x} - 2\mathbf{m}_i^T\mathbf{x} + \mathbf{m}_i^T\mathbf{m}_i\right)$$

- Dropping the 1st term,

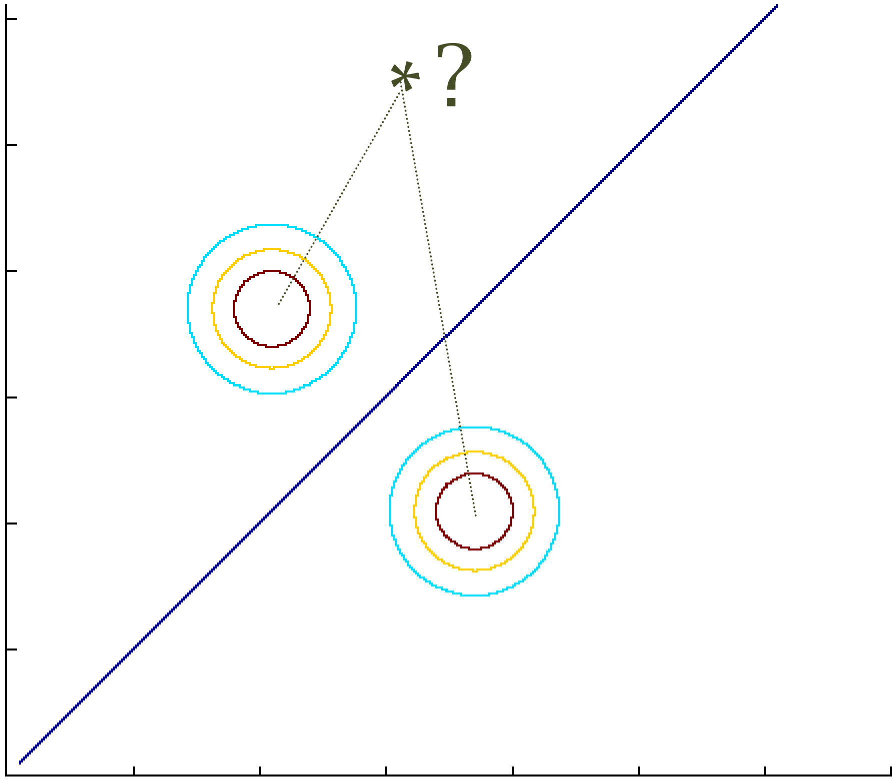
$$g_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

- Where $\mathbf{w}_i = \mathbf{m}_i$ and $w_{i0} = -(1/2)||\mathbf{m}_i||^2$
- If all $\mathbf{m}_i$ have similar norms,

$$g_i(\mathbf{x}) = \mathbf{m}_i^T \mathbf{x}$$

- When the norms of $\mathbf{m}_i$ are comparable, dot product can also be used as the similarity measure instead of the (negative) Euclidean distance.

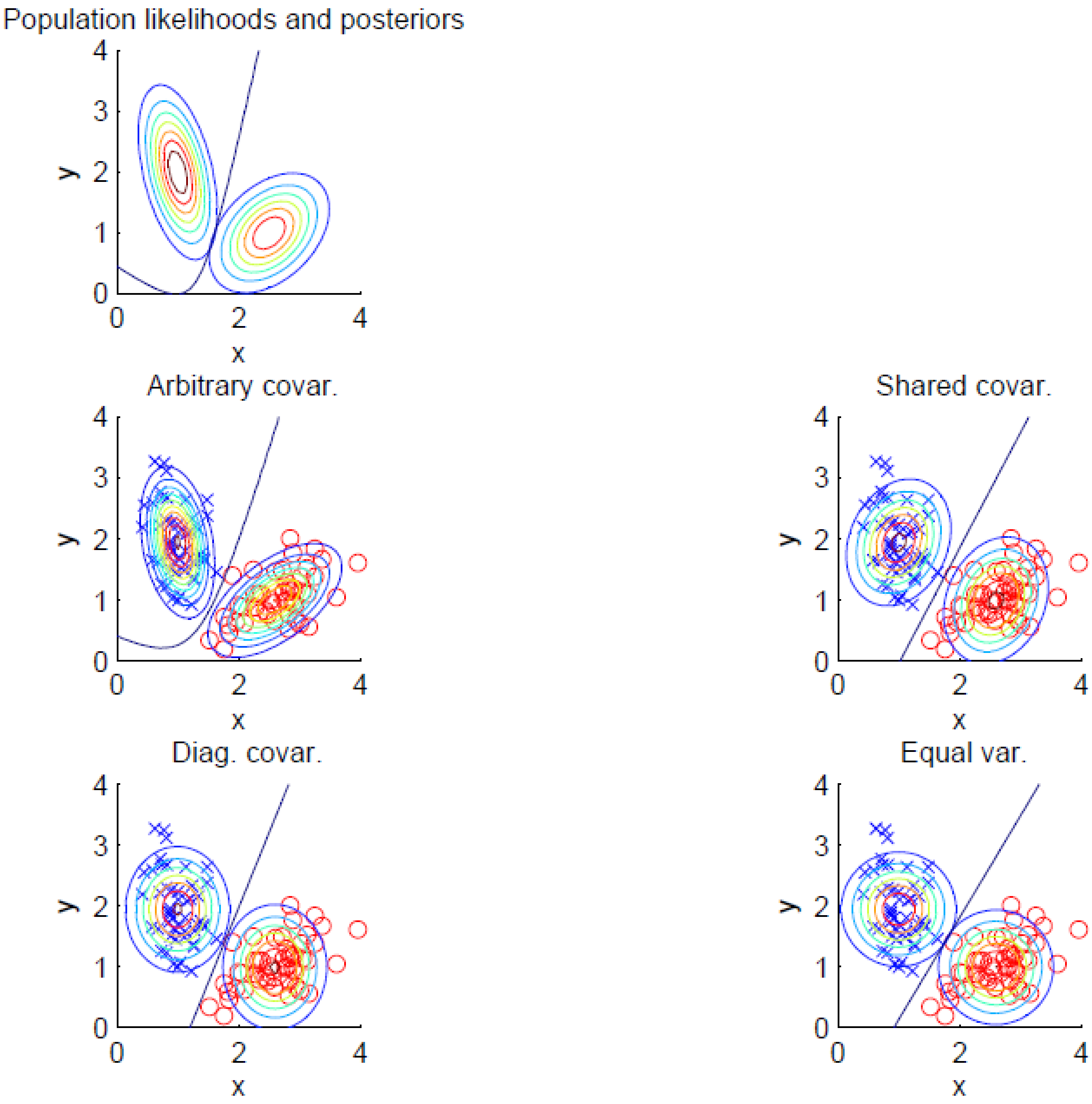
Diagonal **S**, equal variances



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)


Population likelihoods and posteriors
Arbitrary covar.
Shared covar.
Diag. covar.
Equal var.
x
y
0
1
2
3
4

Discrete Features

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

  if x_j are independent (Naive Bayes')

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

  the discriminant is linear

$$g_i\left(\mathbf{x}\right) = \log p\left(\mathbf{x}|C_i\right) + \log P\left(C_i\right)$$

$$= \sum_j \left[x_j \log p_{ij} + \left(1 - x_j\right) \log \left(1 - p_{ij}\right)\right] + \log P\left(C_i\right)$$

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: $x_j \in \{v_1, v_2,\ldots, v_{n_j}\}$

$$z^t_{jk} = \begin{cases} 1 & \text{if } x^t_j = v_k \\ 0 & \text{otherwise} \end{cases} \qquad p_{ijk} \equiv p\left(z_{jk} = 1 | C_i\right) = p\left(x_j = v_k | C_i\right)$$

if x_j are independent

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

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

$$\text{MLE} \rightarrow \quad \hat{p}_{ijk} = \frac{\sum_t z^t_{jk} r^t_i}{\sum_t r^t_i}$$

Multivariate Regression

$$r^t = g\left(x^t | w_0, w_1, \ldots, w_d\right) + \varepsilon = w_0 + w_1 x_1^t + w_2 x_2^t + \cdots + w_d x_d^t + \varepsilon$$

Multivariate linear model

- Minimizing the sum of squared errors:

$$E\left(w_0, w_1, \ldots, w_d | \mathrm{X}\right) = \frac{1}{2} \sum_t \left[r^t - w_0 - w_1 x_1^t - \cdots - w_d x_d^t \right]^2$$

- Taking the derivative with respect to the parameters, w_j ,*j* = *0,…,d*, we get these normal equations:

$$
\begin{aligned}
\sum_t r^t &= N w_0 + w_1 \sum_t x_1^t + w_2 \sum_t x_2^t + \cdots + w_d \sum_t x_d^t \\
\sum_t x_1^t r^t &= w_0 \sum_t x_1^t + w_1 \sum_t (x_1^t)^2 + w_2 \sum_t x_1^t x_2^t + \cdots + w_d \sum_t x_1^t x_d^t \\
\sum_t x_2^t r^t &= w_0 \sum_t x_2^t + w_1 \sum_t x_1^t x_2^t + w_2 \sum_t (x_2^t)^2 + \cdots + w_d \sum_t x_2^t x_d^t \\
&\vdots \\
\sum_t x_d^t r^t &= w_0 \sum_t x_d^t + w_1 \sum_t x_d^t x_1^t + w_2 \sum_t x_d^t x_2^t + \cdots + w_d \sum_t (x_d^t)^2
\end{aligned}
$$

Defining:

$$
\mathbf{X} = \begin{bmatrix} 1 & x_1^1 & x_2^1 & \cdots & x_d^1 \\ 1 & x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & & & & \\ 1 & x_1^N & x_2^N & \cdots & x_d^N \end{bmatrix}, \boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d, \end{bmatrix}, \boldsymbol{r} = \begin{bmatrix} r^1 \\ r^2 \\ \vdots \\ r^N \end{bmatrix}
$$

Then the normal equations can be written as

$$\mathbf{X}^T \mathbf{X}\mathbf{w} = \mathbf{X}^T \mathbf{r} \Rightarrow \mathbf{w} = \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \mathbf{X}^T \mathbf{r}$$

This method is the same as we used for polynomial regression using one input.

Multivariate polynomial model:

Define new higher-order variables

$$z_1 = x_1,\ z_2 = x_2,\ z_3 = x_1^2,\ z_4 = x_2^2,\ z_5 = x_1 x_2$$

and use the linear model in this new ***z*** space

(basis functions, kernel trick: Chapter 13)