Chapter 5:

Multivariate Methods

Multivariate Data

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

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

Multivariate Parameters

$$\text{Mean} : E[x] = \mu = [\mu_1, \cdots, \mu_d]^T$$

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

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

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

Parameter Estimation

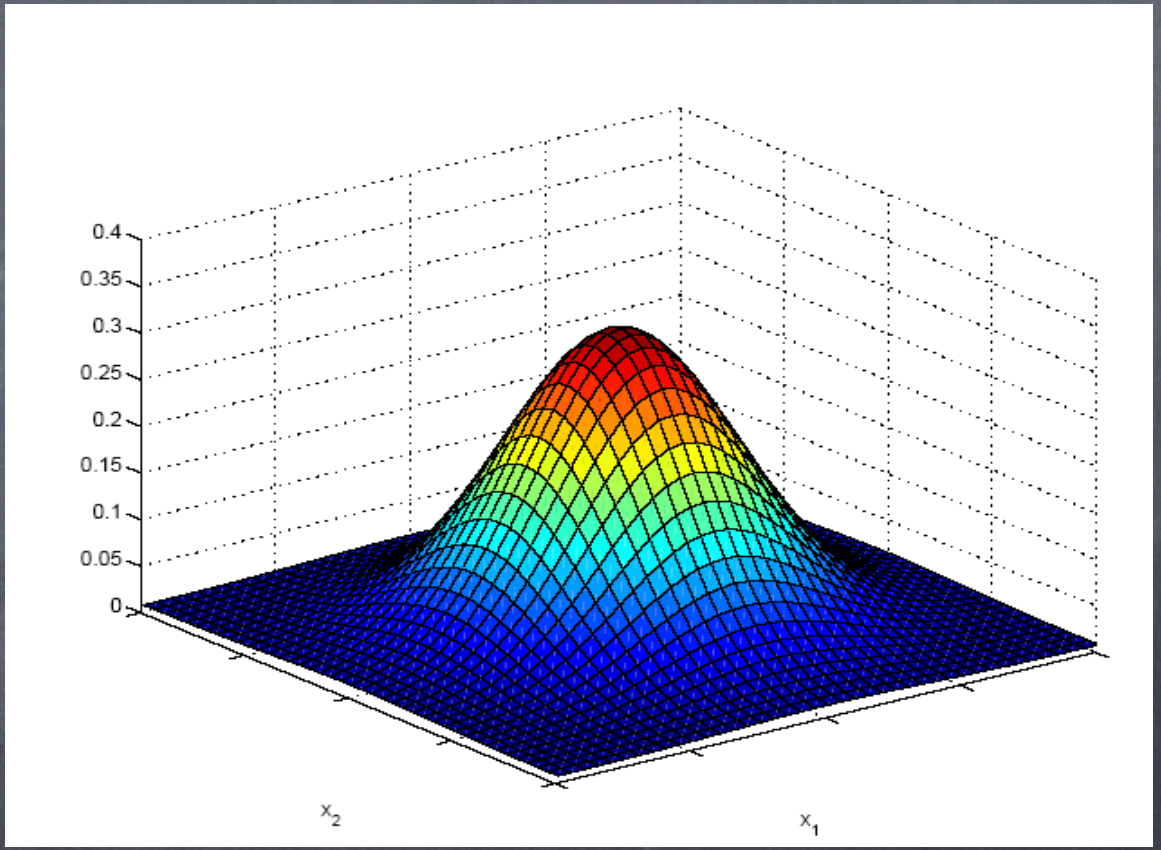
- Sample Mean m $$m_i = \frac{\sum_t x_i^t}{N}, i = 1, \ldots, d$$
- Covariance matrix S $$s_{ij} = \frac{\sum_t (x_i^t - m_i)(x_j^t - m_j)}{N}$$
- Correlation matrix R $$r_{ij} = \frac{s_{ij}}{s_i s_j}$$

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 (There are different type of missing variables.)
- Imputation: Fill in the missing value (How? Model-dependent?)
  - Mean imputation: Use the most likely value (e.g., mean)
  - Imputation by regression: Predict based on other attributes (Possible Project Problem: Can we use kernel regression instead? Let me know, if you want to work on this problem.)

Multivariate Normal Distribution

$$X \sim N_d(\mu, \Sigma)$$



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

Multivariate Normal Distribution

- Mahalanobis distance: $(x - \mu)^T \Sigma^{-1} (x - \mu)$

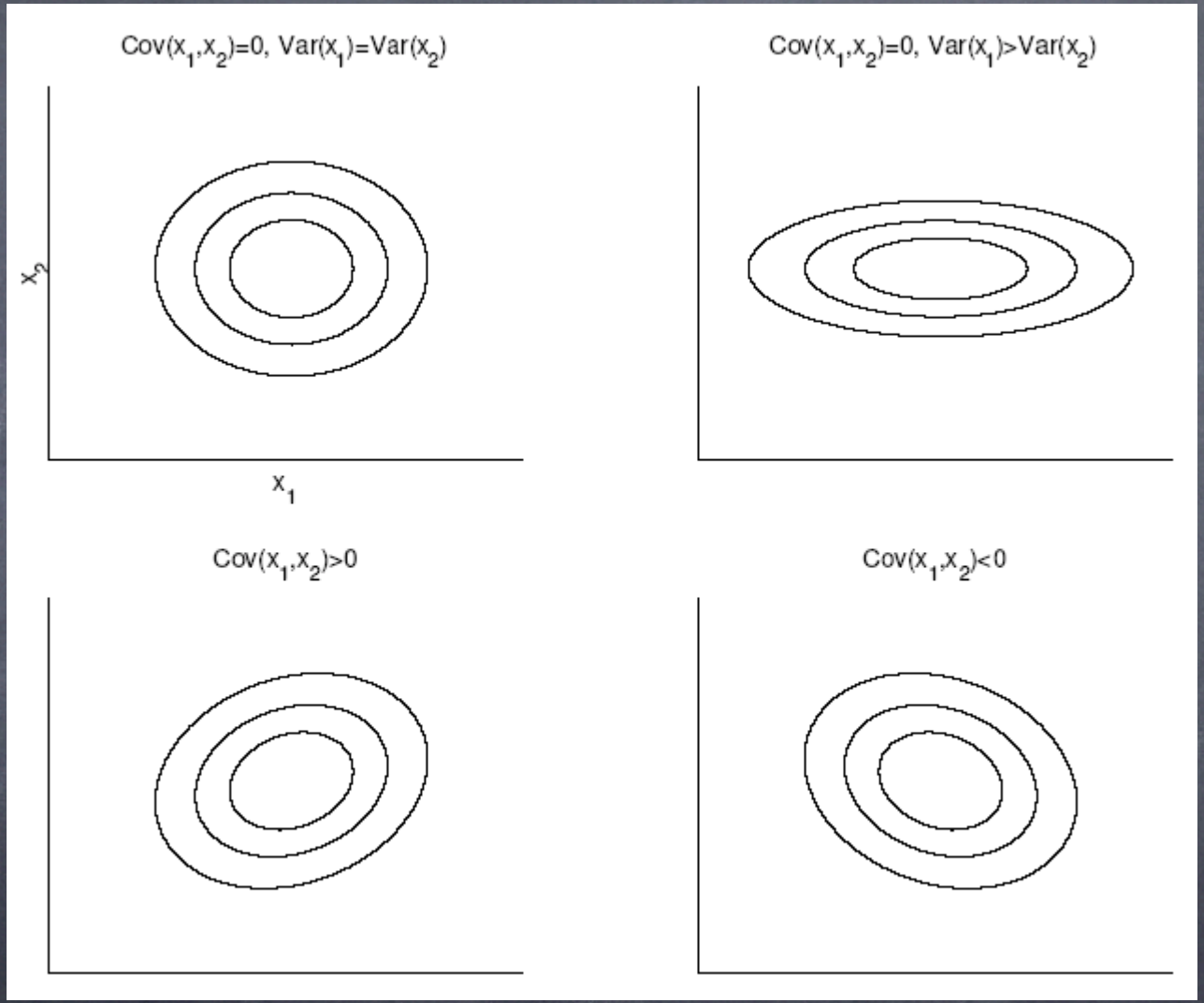
  measures the distance from x to μ in terms of Σ (normalizes for difference in variances and correlations; will see it again in Fisher DA.)

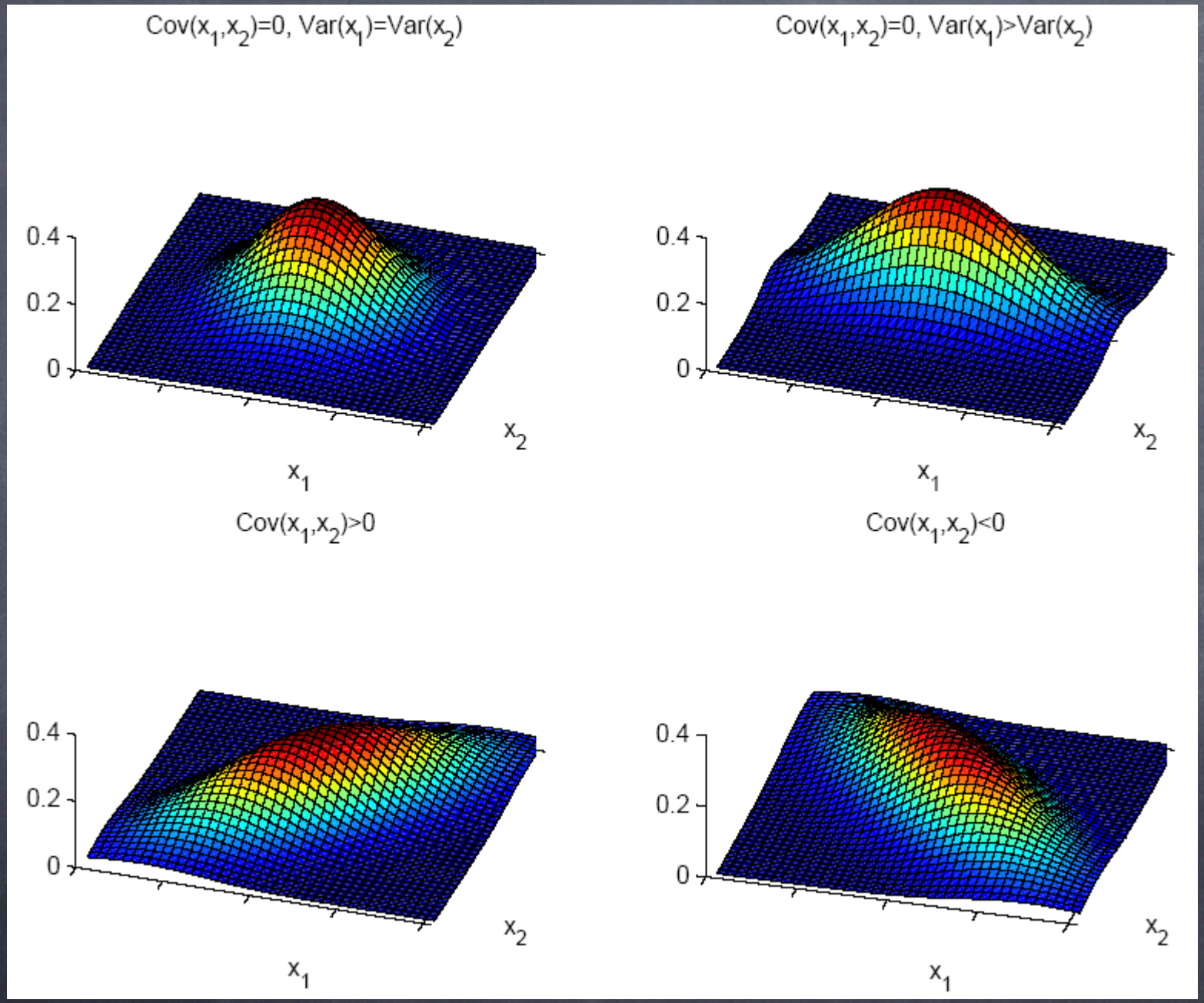
- Bivariate: $d = 2$ $\quad \Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2 \end{bmatrix}$

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

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

Bivariate Normal




Cov(x1,x2)=0, Var(x1)=Var(x2)
0.4
0.2
0
x2
x1
Cov(x1,x2)=0, Var(x1)>Var(x2)
0.4
0.2
0
x2
x1
Cov(x1,x2)>0
0.4
0.2
0
x2
x1
Cov(x1,x2)<0
0.4
0.2
0
x2
x1

Independent Inputs: Naive Bayes

- If x_i are independent, off-diagonals of Σ are 0 (i.e. components are uncorrelated), then Mahalanobis distance reduces to weighted (by $1/\sigma_i$) Euclidean distance:

$$p(x) = \prod_1^d p_i(x_i) = \frac{1}{(2\pi)^{d/2} \prod_i^d \sigma_i} \exp\left[-\frac{1}{2}\sum_1^d \left(\frac{x_i - \mu_i}{\sigma_i}\right)^2\right]$$

- If variances are also equal, reduces to Euclidean distance

Parametric Classification

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

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

- Discriminant functions are

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

(Consider this is a kind of distance from the group mean.)

Estimation of Parameters

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

$$S_i = \frac{\sum_t r_i^t (x^t - m_i)(x^t - m_i)^T}{\sum_t r_i^t}$$

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

Different S_i

- Quadratic discriminant

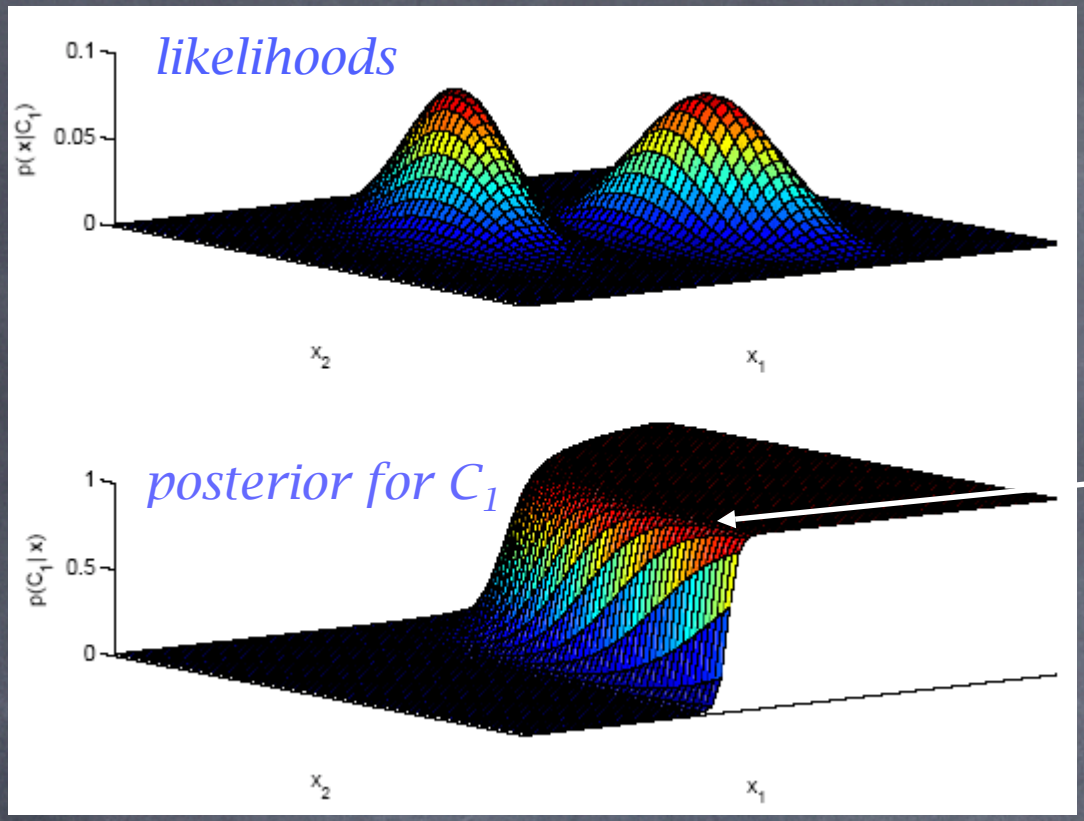
$$g_i(x) = -\frac{1}{2}\log|S_i| - \frac{1}{2}(x^T S_i^{-1} x - 2x^T S_i^{-1} m_i + m_i^T S_i^{-1} m_i) + \log \hat{P}(C_i)$$

$$= x^T W_i x + w_i^T x + w_{i0}$$

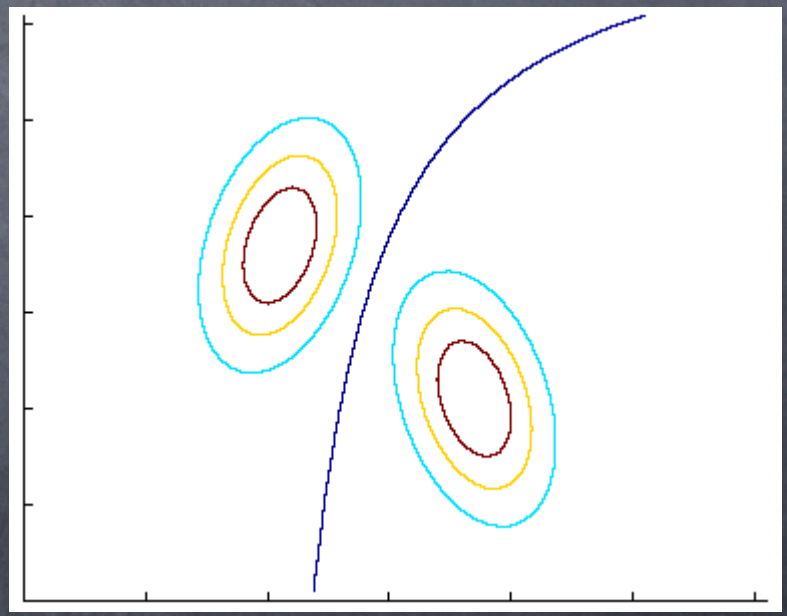
$$\text{where } W_i = \frac{1}{2} S_i^{-1}$$

$$w_i = S_i^{-1} m_i$$

$$w_{i0} = -\frac{1}{2} m_i^T S_i^{-1} m_i - \frac{1}{2}\log|S_i| + \log \hat{P}(C_i)$$



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



Common Covariance Matrix S

- Shared common sample covariance S

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

- Discriminant reduces to

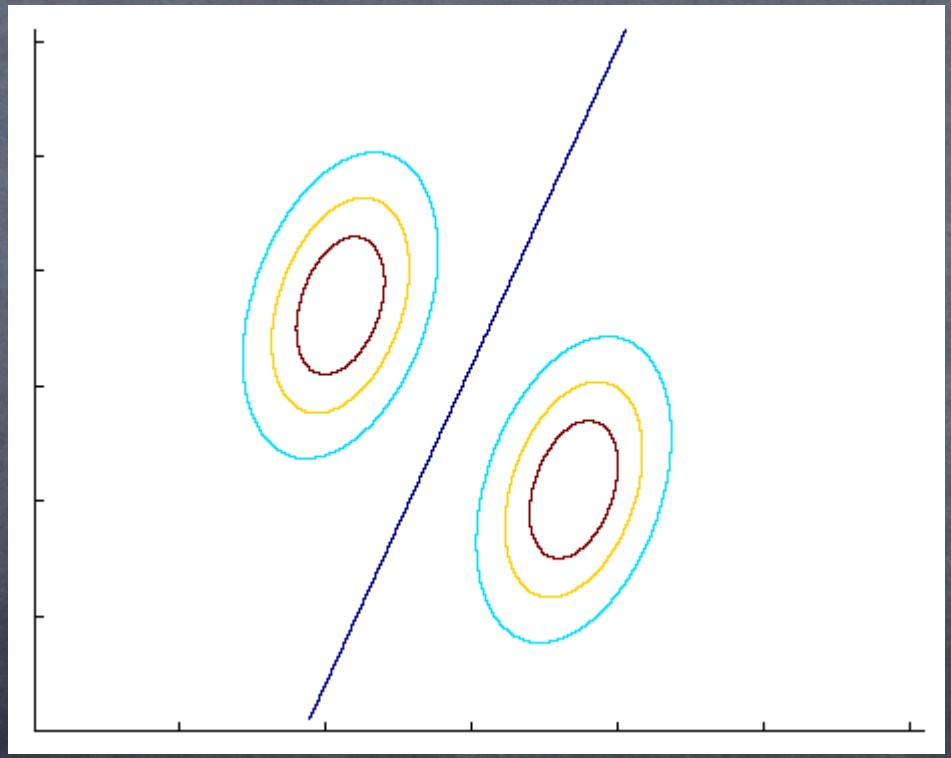
$$g_i(x) = -\frac{1}{2}(x - m_i)^T S_i^{-1}(x - m_i) + \log \hat{P}(C_i)$$

which is a linear discriminant

$$g_i(x) = w_i^T x + w_{i0}$$

$$\text{where } w_i = S_i^{-1} m_i, w_{i0} = -\frac{1}{2} m_i^T S_i^{-1} m_i + \log \hat{P}(C_i)$$

Common Covariance Matrix s



Diagonal s

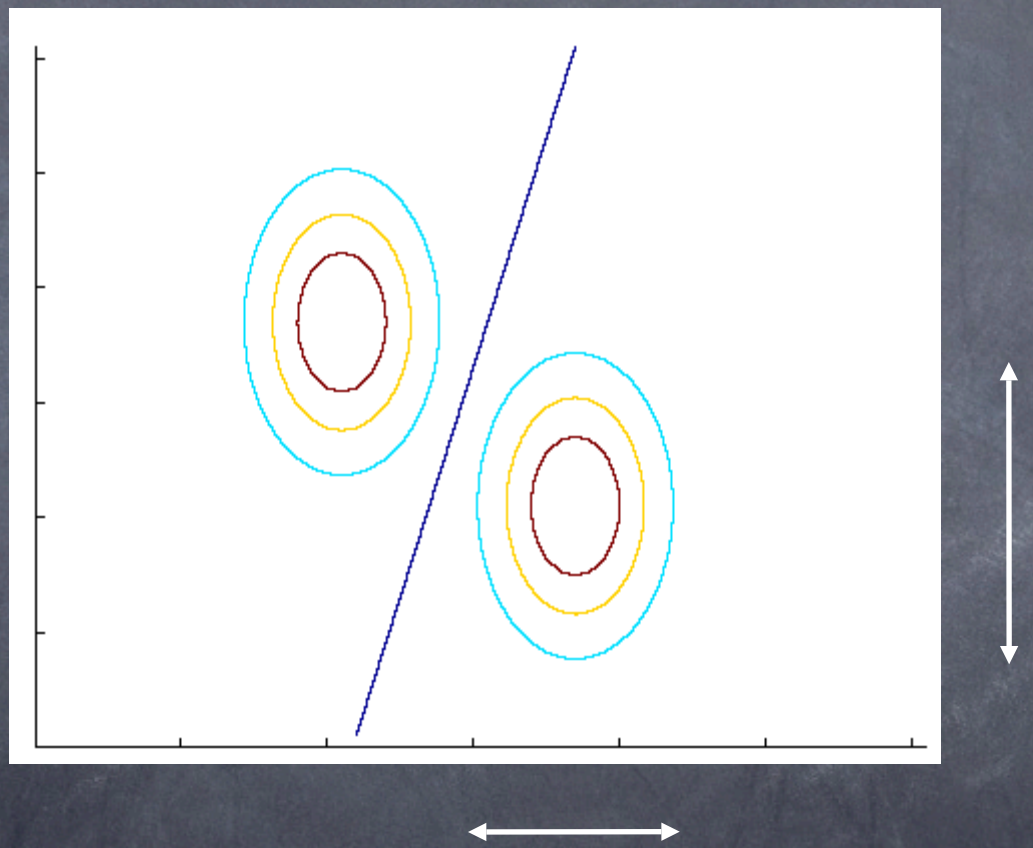
- When x_j j = 1,..d, are independent, Σ is diagonal

  $p(\mathbf{x}|C_i) = \Pi_j\ p(x_j|C_i)$ (Naive Bayes' assumption; i.e. assuming all components are "independent".)

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

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

Diagonal Σ



variances may be different

Diagonal S, equal variances

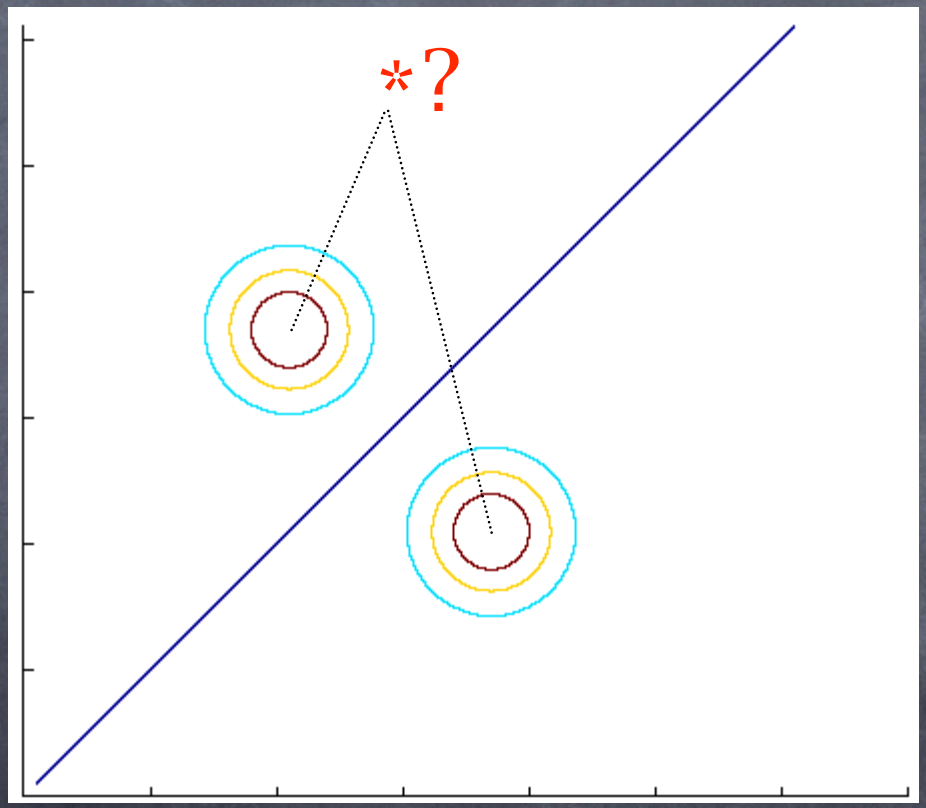
- Nearest mean classifier: Classify based on Euclidean distance to the nearest mean

$$g_i(x) = -\frac{\|x - m_i\|^2}{2s^2} + \log \hat{P}(C_i)$$

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

- 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	$S_i=S=s^2I$	*1*
Shared, Axis-aligned	$S_i=S$, *with* $s_{ij}=0$	*d*
Shared, Hyperellipsoidal	$S_i=S$	$d(d+1)/2$
Different, Hyperellipsoidal	S_i	$K\ d(d+1)/2$

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

Discrete Features

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

  if x_j are independent (Naive Bayes')

$$p(x|C_j) = \prod_{j=1}^{d} p_{ij}^{x_j} (1 - P_{ij})^{1-x_j}$$

  the discriminant is linear

$$g_i(x) = \log p(x|C_i) + \log P(C_i)$$
$$= \sum_j [x_j \log p_{ij} + (1 - x_j) \log(1 - p_{ij})] + \log P(C_i)$$

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

Discrete Features

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

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

if x_j are independent

$$p(x|C_i) = \prod_j^d \prod_k^d P_{ijk}^{z_{jk}}$$

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

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

Multivariate Regression

$$r^t = g(r^t|w_0, w_1, \ldots, w_d) + \epsilon$$

- Multivariate linear model

$$w_0 + w_1 x_1^t + w_2 x_2^t + \cdots + w_d x_d^t$$

$$E(w_o, W-1, \ldots, w_d|X) = \frac{1}{2}\sum_t [r^t - w_0 - w_1 X^T - \cdots - w_d x_d^t]^2$$

- Multivariate polynomial model:

Define new higher-order variables

$Z_1=X_1$, $Z_2=X_2$, $Z_3=X_1^2$, $Z_4=X_2^2$, $Z_5=X_1X_2$

and use the linear model in this new z space

(basis functions, kernel trick, SVM: Chapter 10)