UVA CS 4501: Machine Learning

Lecture 20: Generative Gaussian Bayes Classifiers

Dr. Yanjun Qi

University of Virginia
Department of Computer Science

Where are we ? ➔

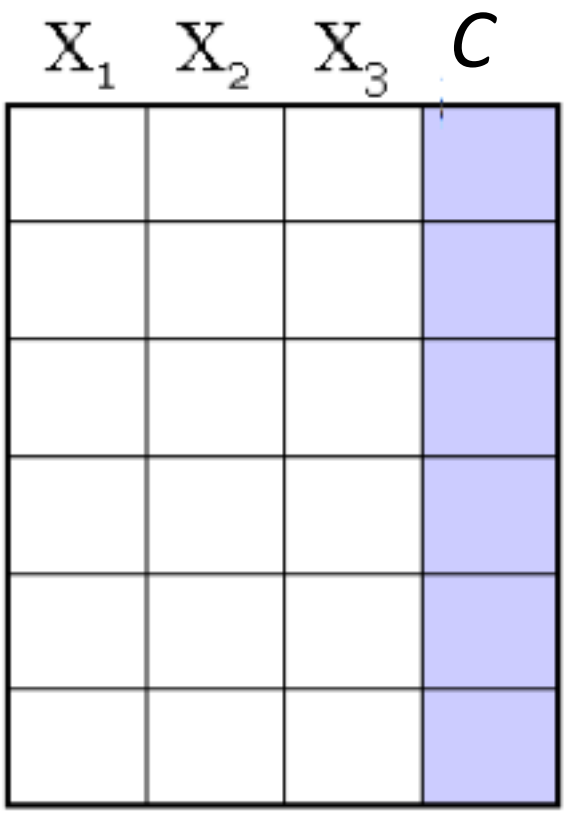
Three major sections for classification

- We can divide the large variety of classification approaches into roughly three major types

1. Discriminative
   - directly estimate a decision rule/boundary
   - e.g., support vector machine, decision tree

2. Generative:
   - build a generative statistical model
   - e.g., naïve bayes classifier, Bayesian networks

3. Instance based classifiers
   - Use observation directly (no models)
   - e.g. K nearest neighbors



A Dataset for classification

$$f : X \longrightarrow C$$

Output as Discrete Class Label
$C_1, C_2, \ldots, C_L$

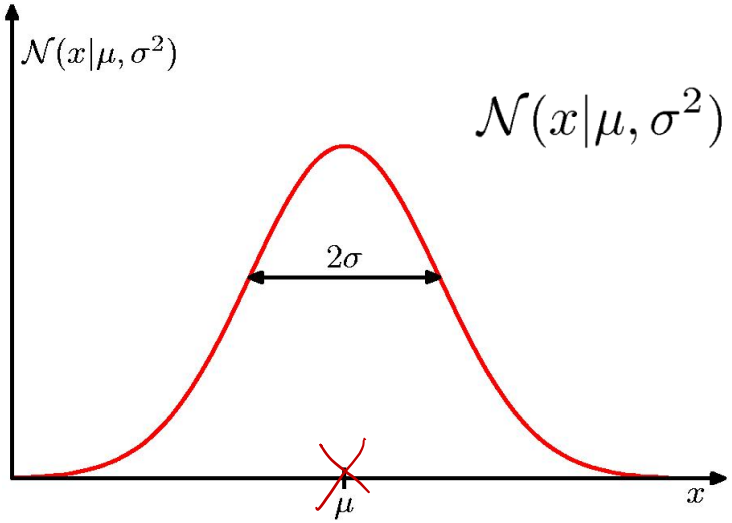
$$P(C \mid \mathbf{X})$$

- **Data**/*points/instances/examples/samples/records*: [rows]
- **Features**/*attributes/dimensions/independent variables/covariates/predictors/regressors*: [columns, except the last]
- **Target**/*outcome/response/label/dependent variable*: special column to be predicted [last column]

Today : Generative Bayes Classifiers

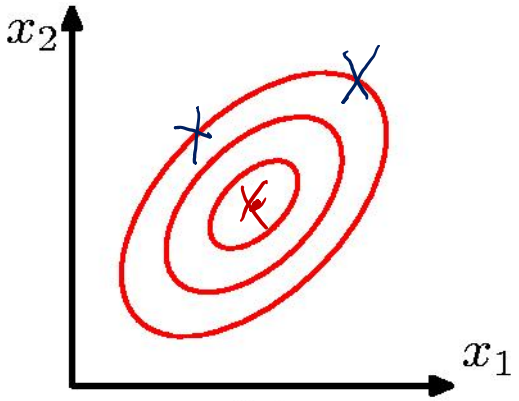
- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC ➔ LDA, QDA

Gaussian Distribution



$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

$$X \sim N(\mu, \sigma^2)$$



$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Mean

Covariance Matrix

Courtesy: http://research.microsoft.com/~cmbishop/PRML/index.htm

Gaussian Naïve Bayes Classifier

$$\underset{C}{\operatorname{argmax}}\, P(C \mid X) = \underset{C}{\operatorname{argmax}}\, P(X, C) = \underset{C}{\operatorname{argmax}}\, P(X \mid C)P(C)$$

Naïve Bayes Classifier

$$\begin{aligned} P(X \mid C) &= P(X_1, X_2, \cdots, X_p \mid C) \\ &= P(X_1 \mid X_2, \cdots, X_p, C)P(X_2, \cdots, X_p \mid C) \\ &= P(X_1 \mid C)P(X_2, \cdots, X_p \mid C) \\ &= P(X_1 \mid C)P(X_2 \mid C)\cdots P(X_p \mid C) \end{aligned}$$

Gaussian Naïve Bayes Classifier

$$\underset{C}{\operatorname{argmax}}\, P(C \mid X) = \underset{C}{\operatorname{argmax}}\, P(X, C) = \underset{C}{\operatorname{argmax}}\, P(X \mid C)P(C)$$

Naïve Bayes Classifier

$$\begin{aligned} P(X \mid C) &= P(X_1, X_2, \cdots, X_p \mid C) \\ &= P(X_1 \mid X_2, \cdots, X_p, C)P(X_2, \cdots, X_p \mid C) \\ &= P(X_1 \mid C)P(X_2, \cdots, X_p \mid C) \\ &= P(X_1 \mid C)P(X_2 \mid C)\cdots P(X_p \mid C) \end{aligned}$$

$$\hat{P}(X_j \mid C = c_i) = \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

μ_{ji} : mean (avearage) of attribute values X_j of examples for which $C = c_i$

σ_{ji} : standard deviation of attribute values X_j of examples for which $C = c_i$

Gaussian Naïve Bayes Classifier

- Continuous-valued Input Attributes
  - Conditional probability modeled with the normal distribution

$$\hat{P}(X_j \mid C = c_i) = \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

μ_{ji} : mean (avearage) of attribute values X_j of examples for which $C = c_i$

σ_{ji} : standard deviation of attribute values X_j of examples for which $C = c_i$

  - Learning Phase: for $\mathbf{X} = (X_1, \cdots, X_p)$, $C = c_1, \cdots, c_L$
    Output: $p \times L$ normal distributions and $P(C = c_i)$ $i = 1, \cdots, L$

$N(\mu_{ji}, \sigma_{ji}^2)$ $\{ \mu_{ji}, \sigma_{ji}$ → $j \in \{1, 2, \cdots, p\}$, $i \in \{1, 2, \cdots, L\}$

MLE { sample mean, sample variance

Gaussian Naïve Bayes Classifier

- Continuous-valued Input Attributes
  - Conditional probability modeled with the normal distribution

$$\hat{P}(X_j \mid C = c_i) = \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

μ_{ji} : mean (avearage) of attribute values X_j of examples for which $C = c_i$

σ_{ji} : standard deviation of attribute values X_j of examples for which $C = c_i$

  - Learning Phase: for $\mathbf{X} = (X_1, \cdots, X_p)$, $C = c_1, \cdots, c_L$
    Output: $p \times L$ normal distributions and $P(C = c_i)$ $i = 1, \cdots, L$

  - Test Phase: for $\mathbf{X}' = (X_1', \cdots, X_p')$
    - Calculate conditional probabilities with all the normal distributions
    - Apply the MAP rule to make a decision

$$\underset{i}{\operatorname{argmax}}\ P(C=i)\,P(X_1 \mid i) \cdots P(X_p \mid i)$$

Today : Generative Bayes Classifiers

- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC ➔ LDA, QDA

Naïve Gaussian means ?

Naïve

$$P(X_1, X_2, \cdots, X_p \mid C = c_j) = P(X_1 \mid C)P(X_2 \mid C)\cdots P(X_p \mid C)$$

$$= \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

Total #para ⇒ $L \times (p + pf$

$\Sigma | C_i = \begin{bmatrix} \sigma_{1i} & & \\ & \sigma_{2i} & \\ & & \sigma_{pi} \end{bmatrix}$

Diagonal Matrix

$$\Sigma\_c_k = \Lambda\_c_k$$

Each class' covariance matrix is diagonal

Not Naïve Gaussian means ?

Not Naïve

$$P(X_1, X_2, \cdots, X_p \mid C) =$$

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Total # para $\Rightarrow L \times \{p + p \times p\}$; μ/c ; Σ/c

Naïve

$$P(X_1, X_2, \cdots, X_p \mid C = c_j) = P(X_1 \mid C) P(X_2 \mid C) \cdots P(X_p \mid C)$$

$$= \prod_i \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

Total # para $\Rightarrow L \times (p + p)$

$\Sigma | c_i = \begin{bmatrix} \sigma_{1i} & & & \\ & \sigma_{2i} & & \\ & & \ddots & \\ & & & \sigma_{pi} \end{bmatrix}$

Diagonal Matrix $\Sigma\_c_k = \Lambda\_c_k$

Each class' covariance matrix is diagonal

Today : Generative Bayes Classifiers

- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC
  - LDA: Linear Discriminant Analysis
  - QDA: Quadratic Discriminant Analysis

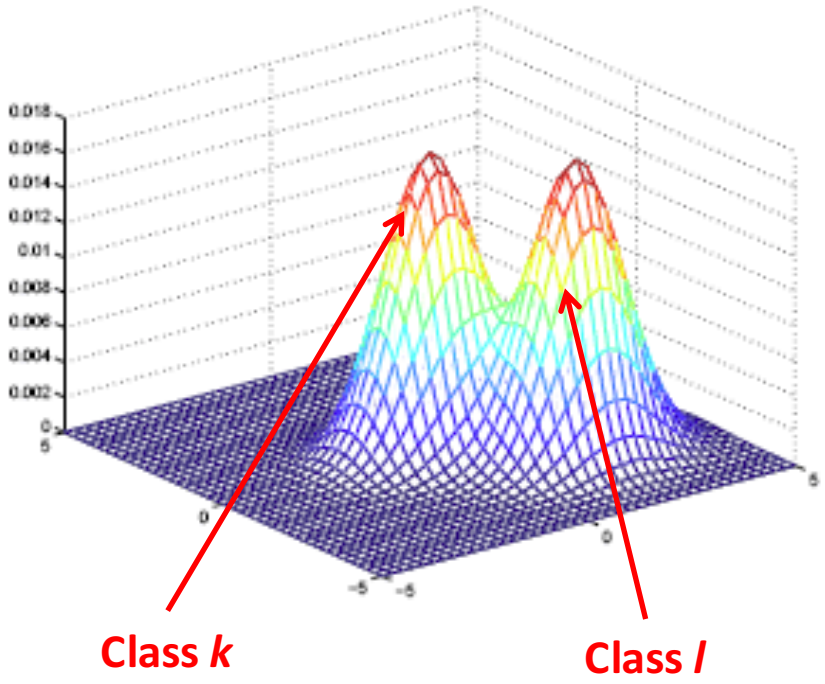
(1) covariance matrix are the same across classes

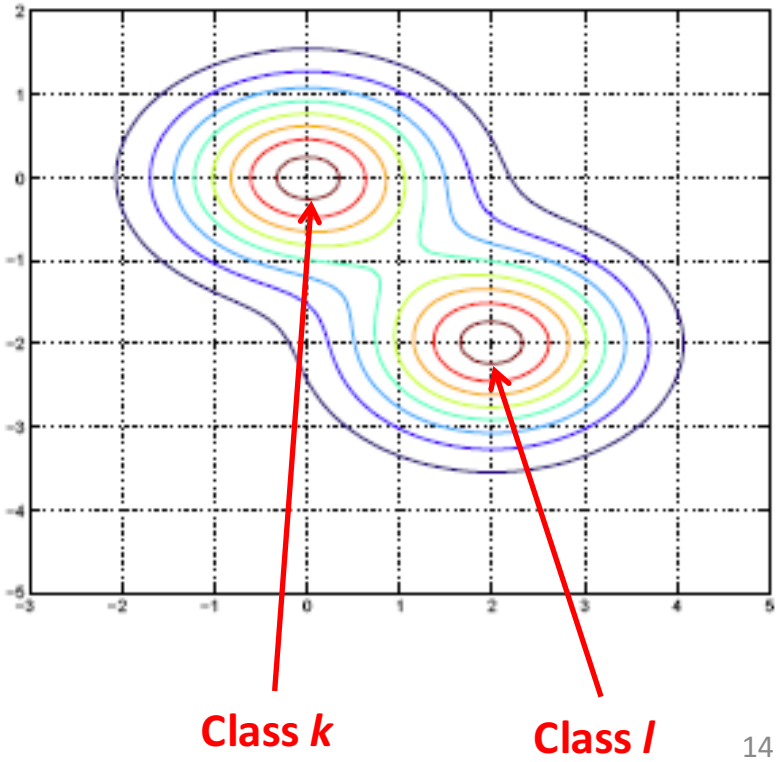
➔ LDA (Linear Discriminant Analysis)

Linear Discriminant Analysis : $\Sigma_k = \Sigma, \ \forall k$ (p×p)

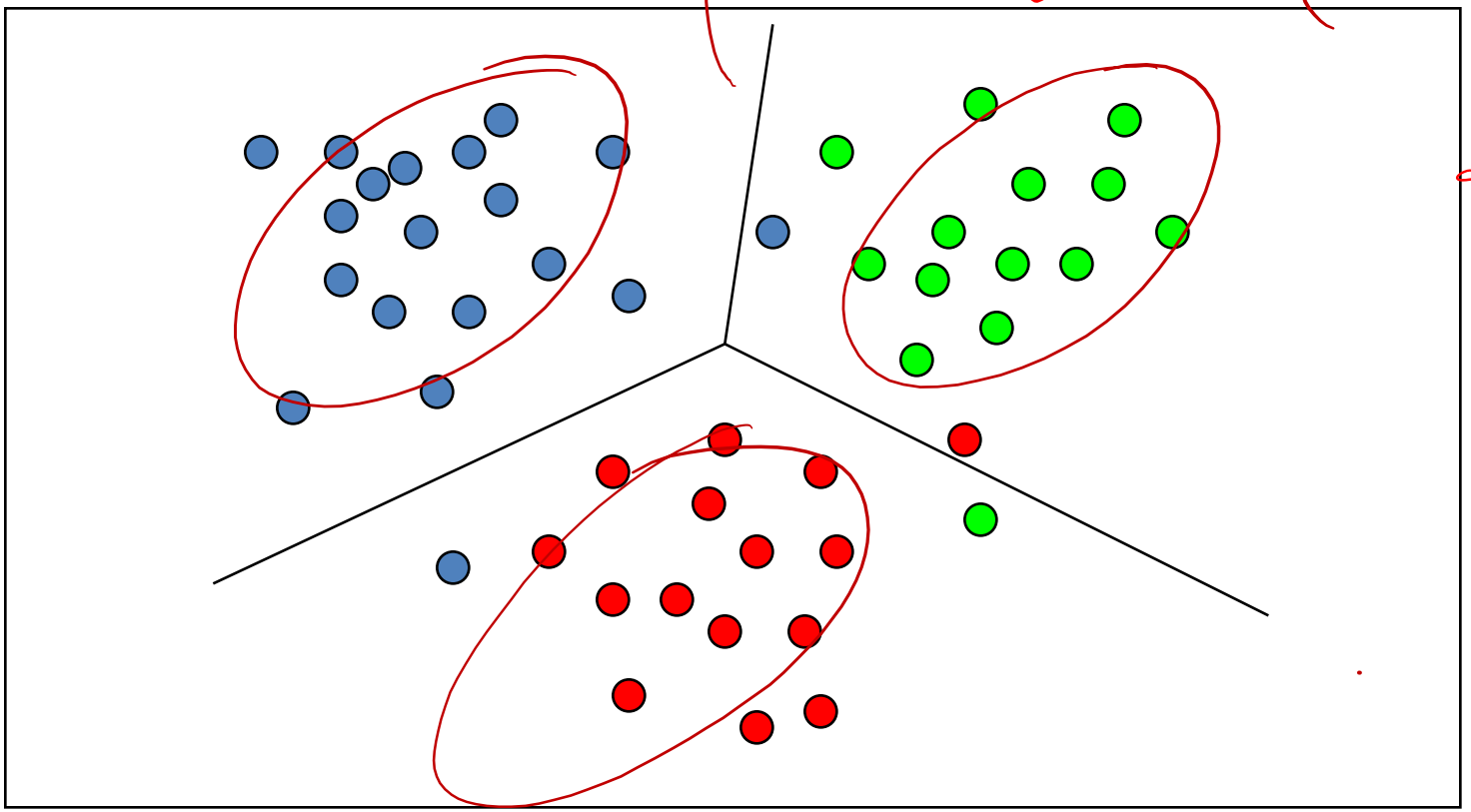
Each class' covariance matrix is the same

The Gaussian Distribution are shifted versions of each other





Visualization (three classes)



$$\underset{k}{\operatorname{argmax}} P(C_k \mid X) = \underset{k}{\operatorname{argmax}} P(X, C_k) = \underset{k}{\operatorname{argmax}} P(X \mid C_k) P(C_k)$$

$$= \underset{k}{\operatorname{argmax}} \log\{P(X \mid C_k) P(C_k)\}$$

Decision Boundary means those points

Satisfying: $p(C_i \mid X) = p(C_j \mid X)$

$$\frac{p(C_i \mid X)}{p(C_j \mid X)} = 1$$

$$\Rightarrow \log \frac{p(C_i \mid X)}{p(C_j \mid X)} = 0$$

$$\underset{k}{\operatorname{argmax}} P(C_k \mid X) = \underset{k}{\operatorname{argmax}} P(X, C_k) = \underset{k}{\operatorname{argmax}} P(X \mid C_k) P(C_k)$$

$$= \underset{k}{\operatorname{argmax}} \log\{P(X \mid C_k) P(C_k)\}$$

$$= \underset{k}{\operatorname{argmax}} \log P(X \mid C_k) + \log P(C_k) \Rightarrow \pi_k$$

Decision Boundary points

$$\log \frac{P(C_k \mid X)}{P(C_\ell \mid X)} = 0 = \log \frac{P(X \mid C_k)}{P(X \mid C_\ell)} + \log \frac{\pi_k}{\pi_\ell}$$

$$= \log P(X \mid C_k) - \log P(X \mid C_\ell) + \log \frac{\pi_k}{\pi_\ell}$$

$$\log \frac{P(C_k \mid X)}{P(C_l \mid X)} = \log \frac{P(X \mid C_k)}{P(X \mid C_l)} + \log \frac{P(C_k)}{P(C_l)}$$

$$= \log \frac{\pi_k}{\pi_\ell} - \frac{1}{2}(\mu_k + \mu_\ell)^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_\ell) \qquad (4.9)$$
$$+ x^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_\ell),$$

$$\log \frac{P(C_k \mid X)}{P(C_l \mid X)} = \log \frac{P(X \mid C_k)}{P(X \mid C_l)} + \log \frac{P(C_k)}{P(C_l)}$$

$$= \log \frac{\pi_k}{\pi_\ell} - \frac{1}{2}(\mu_k + \mu_\ell)^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_\ell) \qquad (4.9)$$
$$+ x^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_\ell),$$

The above is derived from the following :

$$-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k - \frac{1}{2}x^T\Sigma^{-1}x$$

$$\log \frac{P(C_k \mid X)}{P(C_l \mid X)} = \log \frac{P(X \mid C_k)}{P(X \mid C_l)} + \log \frac{P(C_k)}{P(C_l)}$$

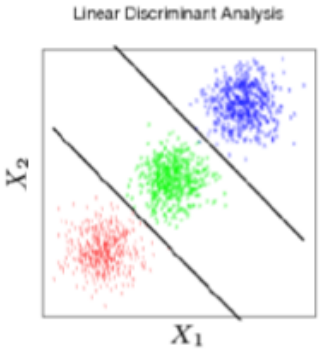
$$= \log \frac{\pi_k}{\pi_\ell} - \frac{1}{2}(\mu_k + \mu_\ell)^T \Sigma^{-1}(\mu_k - \mu_\ell) \quad (4.9)$$

$$+ x^T \Sigma^{-1}(\mu_k - \mu_\ell) = 0$$

b

a

$\Rightarrow$ $x^T a + b = 0 \Rightarrow$ a linear line decision boundary



LDA Classification Rule (also called as Linear discriminant function:)

$$\underset{k}{\operatorname{argmax}} P(C_k \mid X) = \underset{k}{\operatorname{argmax}} P(X, C_k) = \underset{k}{\operatorname{argmax}} P(X \mid C_k) P(C_k)$$

$$= \arg\max_k \left[-\log((2\pi)^{p/2}|\Sigma|^{1/2}) - \frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) + \log(\pi_k) \right]$$

$$= \arg\max_k \left[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) + \log(\pi_k) \right]$$

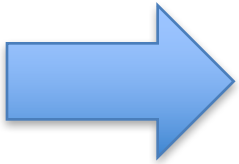
Linear Discriminant Function for LDA

- Note

$$-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k - \frac{1}{2}x^T\Sigma^{-1}x$$

Today : Generative Bayes Classifiers

- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC
    - LDA: Linear Discriminant Analysis
    - QDA: Quadratic Discriminant Analysis



(2) If covariance matrix are not the same e.g. ➔ QDA (Quadratic Discriminant Analysis)

- Estimate the covariance matrix Σ_k separately for each class k, $k = 1, 2, ..., K$.
- *Quadratic discriminant function*:

$$\delta_k(x) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log\pi_k \ .$$

- Classification rule:

$$\hat{G}(x) = \arg\max_k \delta_k(x) \ .$$

- Decision boundaries are quadratic equations in x.
- QDA fits the data better than LDA, but has more parameters to estimate.

(2) If covariance matrix are not the same e.g. ➔ QDA (Quadratic Discriminant Analysis)

- Estimate the covariance matrix Σ_k separately for each class k, $k = 1, 2, ..., K$.
- *Quadratic discriminant function*:

$$\delta_k(x) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log\pi_k \ .$$

$\{\Sigma_1, \Sigma_2, ..\Sigma_K, \mu_1, \mu_2, .. \mu_K\}$

- Classification rule:

$$\hat{G}(x) = \arg\max_k \delta_k(x) \ .$$

$\delta_k(x) - \delta_\ell(x) = 0$

Total # para

$K(p+p^2)$

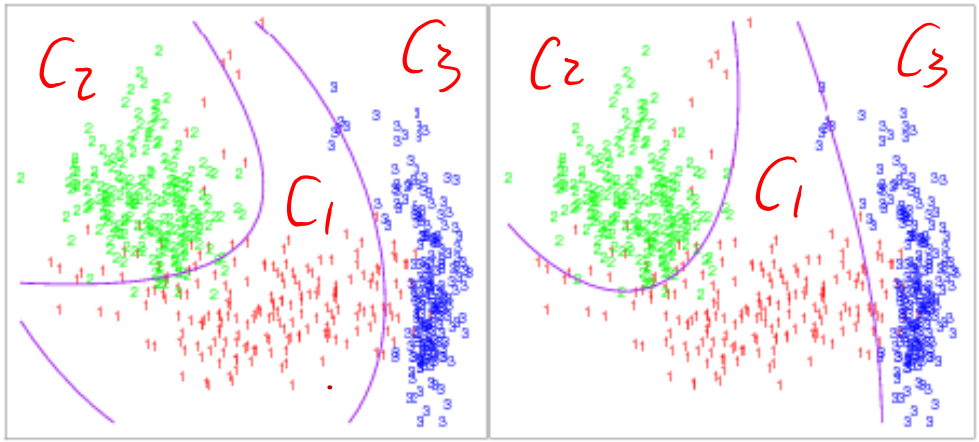
$\{\mu_k, \Sigma_k\}$

- Decision boundaries are quadratic equations in x.
- QDA fits the data better than LDA, but has more parameters to estimate.

QDA vs. LDA on Expanded Basis

- Expand input space to include X_1X_2, X_1^2, and X_2^2.
- Input is five dimensional: $X = (X_1, X_2, X_1X_2, X_1^2, X_2^2)$.

LDA with $Q(X)$



QDA

LDA with quadratic basis Versus QDA

Figure 4.6: *Two methods for fitting quadratic boundaries. The left plot shows the quadratic decision boundaries for the data in Figure 4.1 (obtained using LDA in the five-dimensional space $x_1, x_2, x_{12}, x_1^2, x_2^2$). The right plot shows the quadratic decision boundaries found by QDA. The differences are small, as is usually the case.*

Both with Quadratic decision Boundary

(3) Regularized Discriminant Analysis

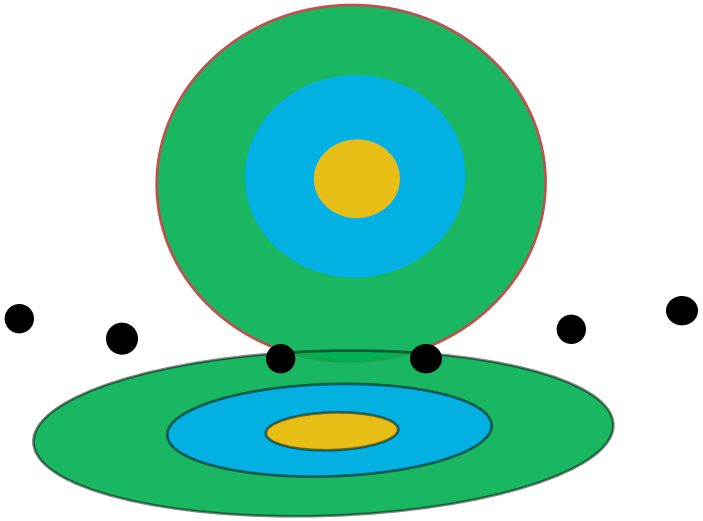
- A compromise between LDA and QDA.
- Shrink the separate covariances of QDA toward a common covariance as in LDA.
- Regularized covariance matrices:

$$\hat{\Sigma}_k(\alpha) = \alpha\hat{\Sigma}_k + (1-\alpha)\hat{\Sigma} \, .$$

- The quadratic discriminant function $\delta_k(x)$ is defined using the shrunken covariance matrices $\hat{\Sigma}_k(\alpha)$.
- The parameter α controls the complexity of the model.

An example: Gaussian Bayes Classifier

Naive BC



$\Sigma_1 \neq \Sigma_2$

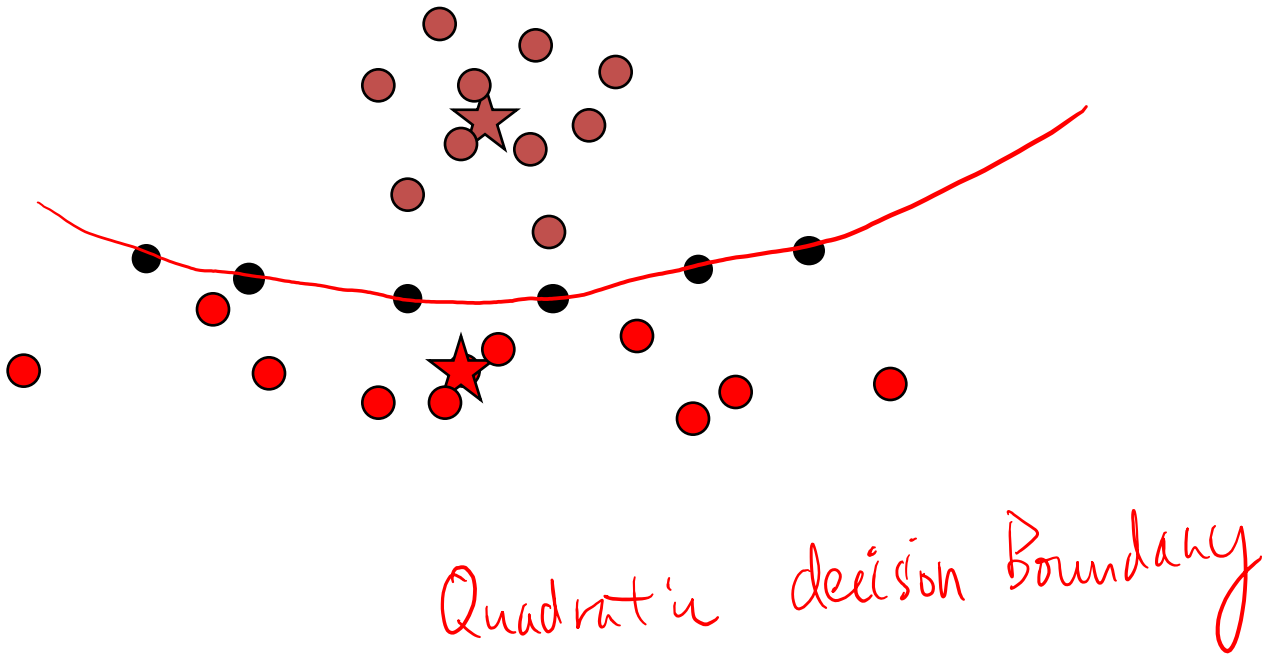
diagonal

$\Sigma_1 = \Lambda_1$

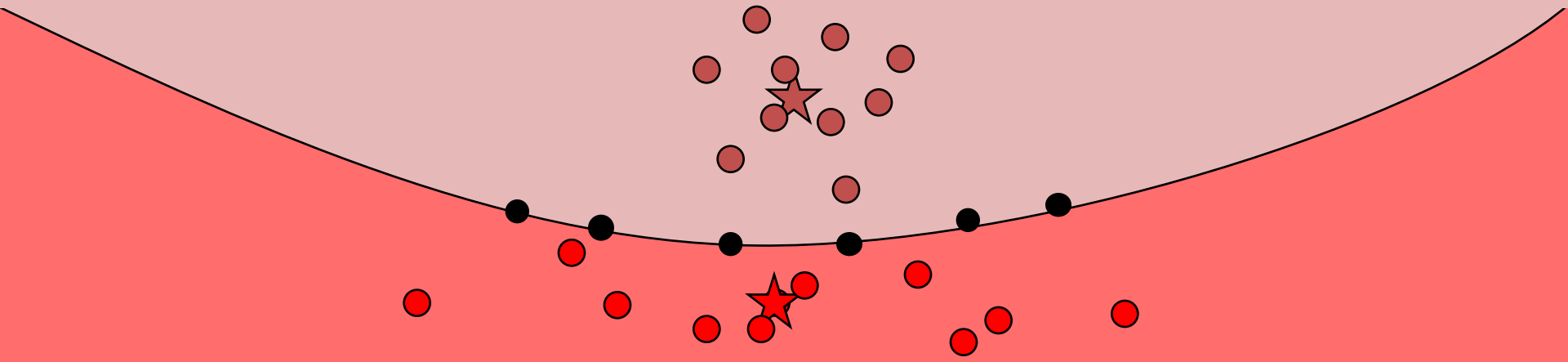
$\Sigma_2 = \Lambda_2$

diagonal

Gaussian Bayes Classifier

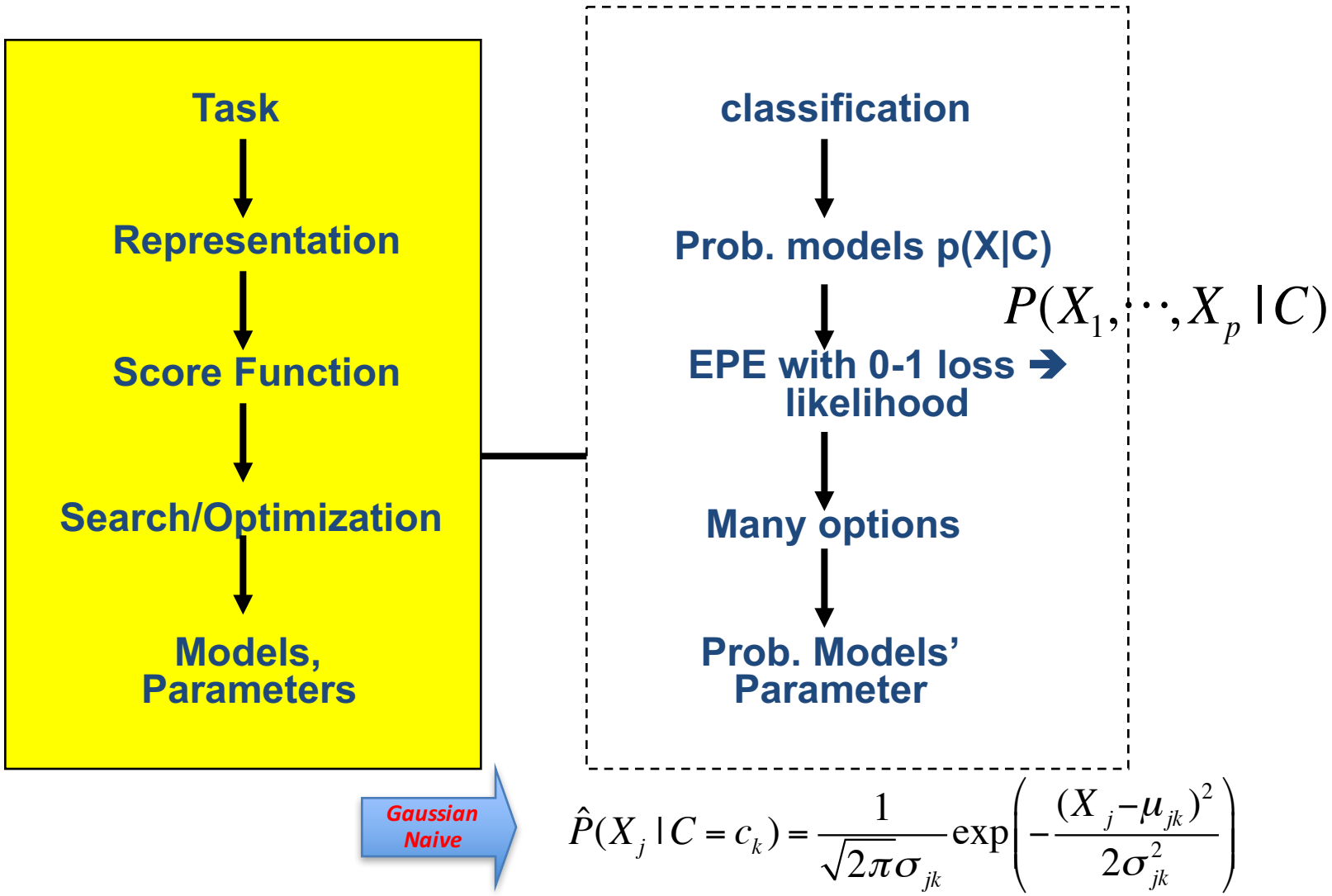


Gaussian Bayes Classifier



Naïve Gaussian Bayes Classifier is not a linear classifier!

Generative Gaussian Bayes Classifier



$$\underset{k}{\operatorname{argmax}}\, P(C\_k \mid X) = \underset{k}{\operatorname{argmax}}\, P(X, C) = \underset{k}{\operatorname{argmax}}\, P(X \mid C) P(C)$$

References

- Prof. Andrew Moore's review tutorial
- Prof. Ke Chen NB slides
- Prof. Carlos Guestrin recitation slides