UVA CS 4501: Machine Learning

Lecture 17b: Gaussian BC and Generative vs. Discriminative Classifier

Dr. Yanjun Qi

University of Virginia

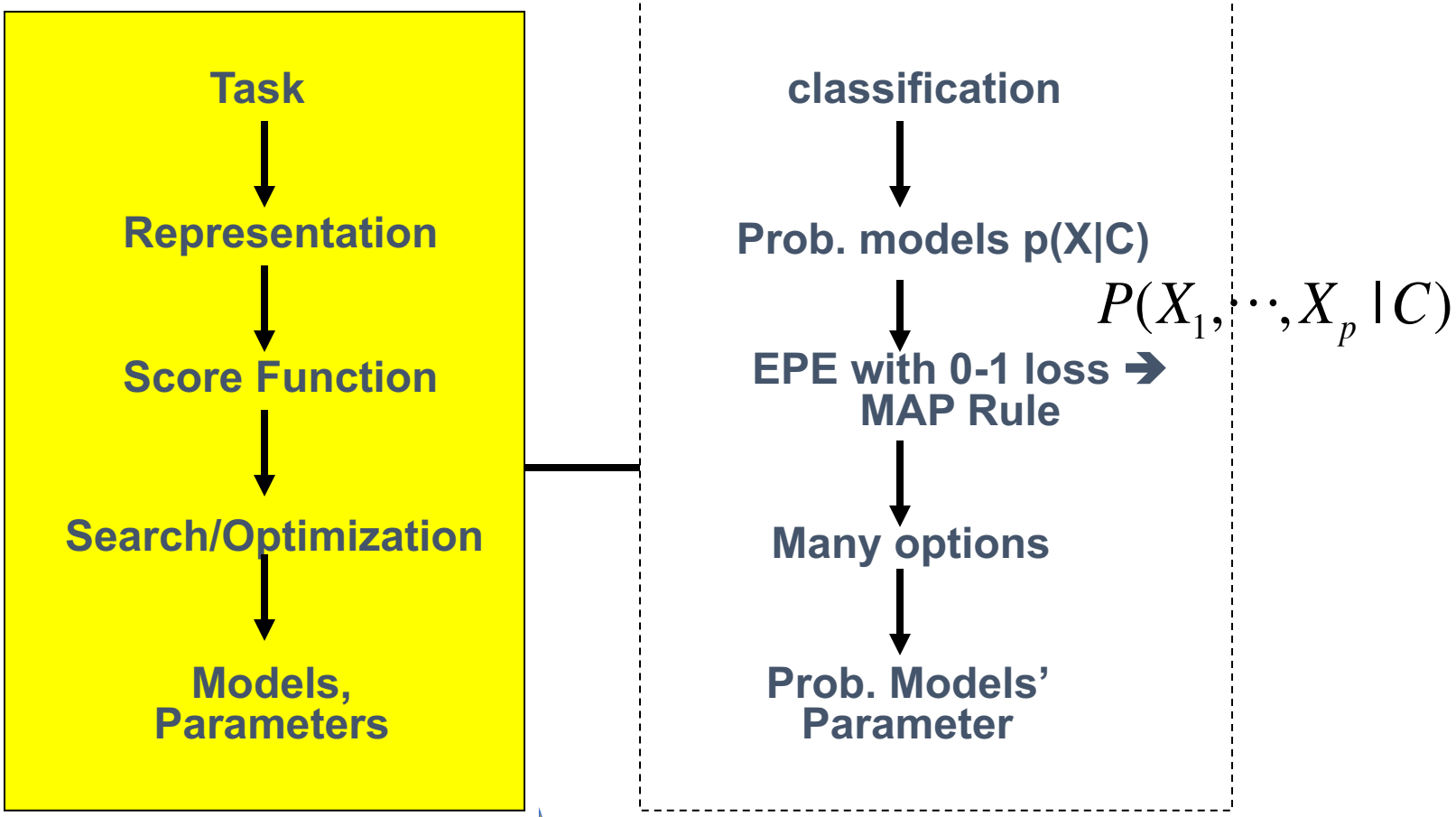
Department of Computer Science

Today: More Generative Bayes Classifiers

- ✓ Generative Bayes Classifier
- ✓ Naïve Bayes Classifier
- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC ➔ LDA, QDA
- ✓ Discriminative vs. Generative

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

Generative Bayes Classifier



Bernoulli Naïve

$$p(W_i = true \mid c_k) = p_{i,k}$$

Gaussian Naive

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

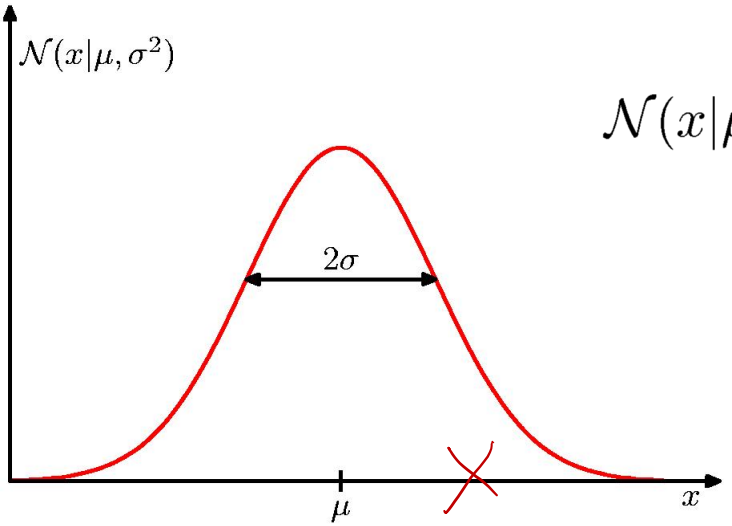
Multinomial

$$P(W_1 = n_1,\ldots,W_v = n_v \mid c_k) = \frac{N!}{n_{1k}!n_{2k}!..n_{vk}!}\theta_{1k}^{n_{1k}}\theta_{2k}^{n_{2k}}..\theta_{vk}^{n_{vk}}$$

Review: Continuous Random Variables

- Probability density function (pdf) instead of probability mass function (pmf)
  - For discrete RV: Probability mass function (pmf): $P(X = x_i)$
- A pdf (prob. Density func.) is any function $f(x)$ that describes the probability density in terms of the input variable x.

Single-Variate Gaussian Distribution

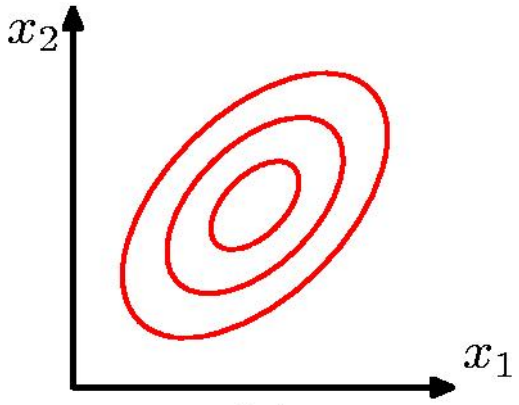


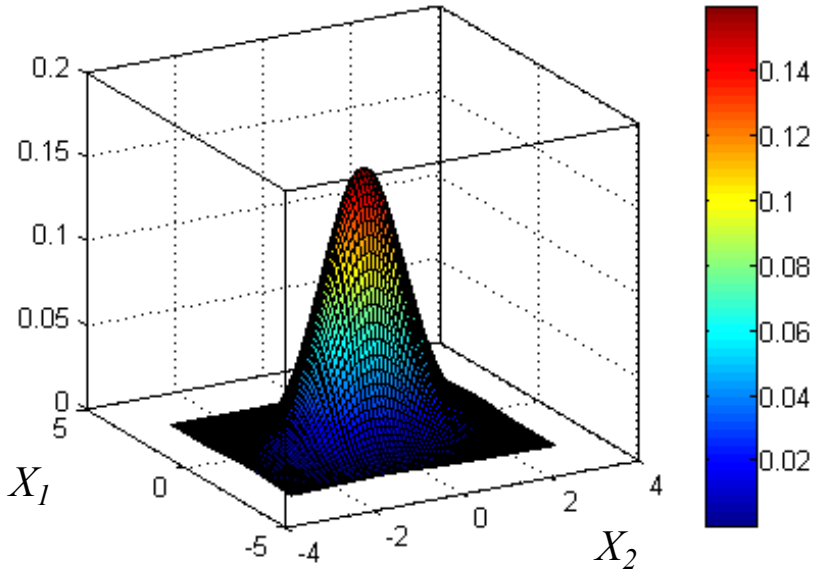
$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

$$X \sim \mathcal{N}\left(\mu, \sigma^2\right)$$

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

Bi-Variate Gaussian Distribution





Bivariate normal PDF:

- Mean of normal PDF is at peak value. Contours of equal PDF form ellipses.

- The covariance matrix captures linear dependencies among the variables

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

Multivariate Normal (Gaussian) PDFs

The only widely used continuous joint PDF is the multivariate normal (or Gaussian):

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

Where |*| represents determinant

Mean

Covariance Matrix

- Mean of normal PDF is at peak value. Contours of equal PDF form ellipses.

- The covariance matrix captures linear dependencies among the variables

Example: the Bivariate Normal distribution

$$f\left(x_1, x_2\right) = \frac{1}{\left(2\pi\right)\left|\Sigma\right|^{1/2}} \mathrm{e}^{-\frac{1}{2}\left(\vec{x}-\vec{\mu}\right)^T \Sigma^{-1}\left(\vec{x}-\vec{\mu}\right)}$$

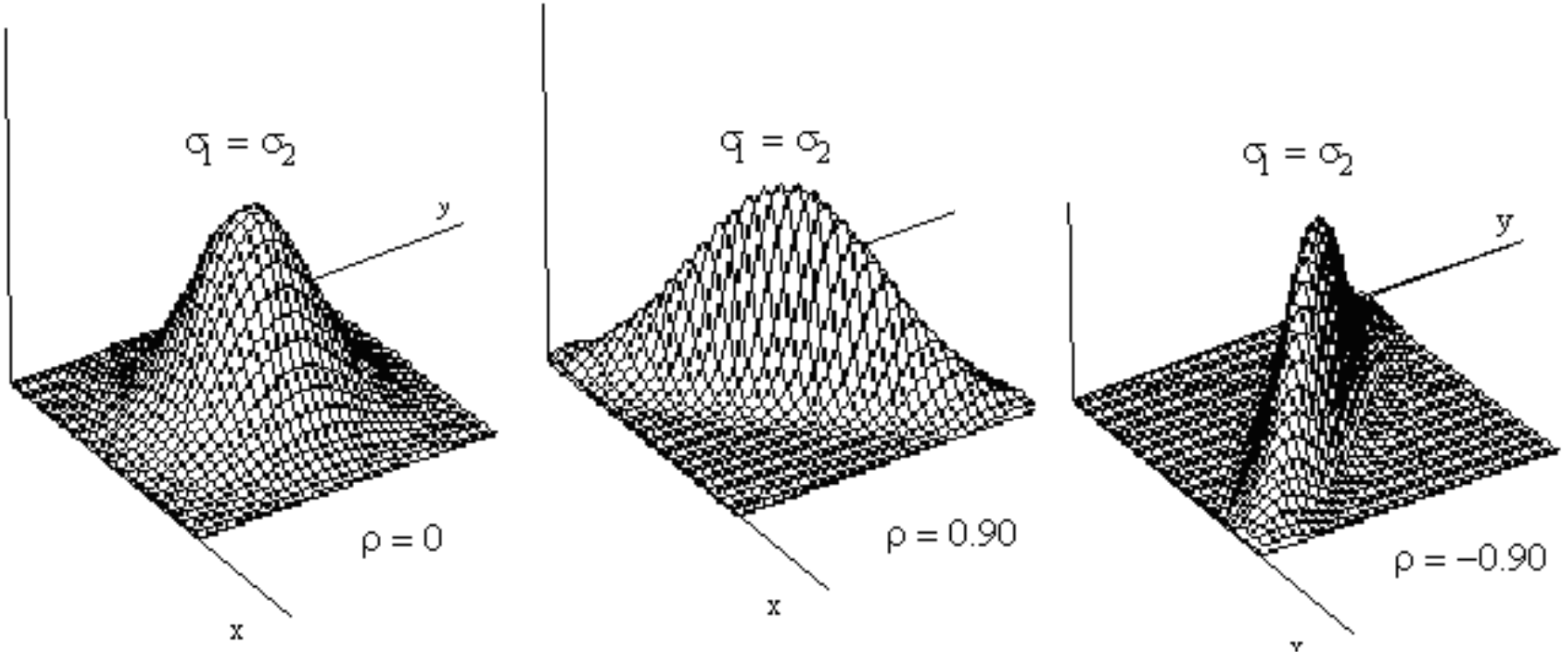
with $\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}$ 2x1 and

$$\underset{2\times 2}{\Sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

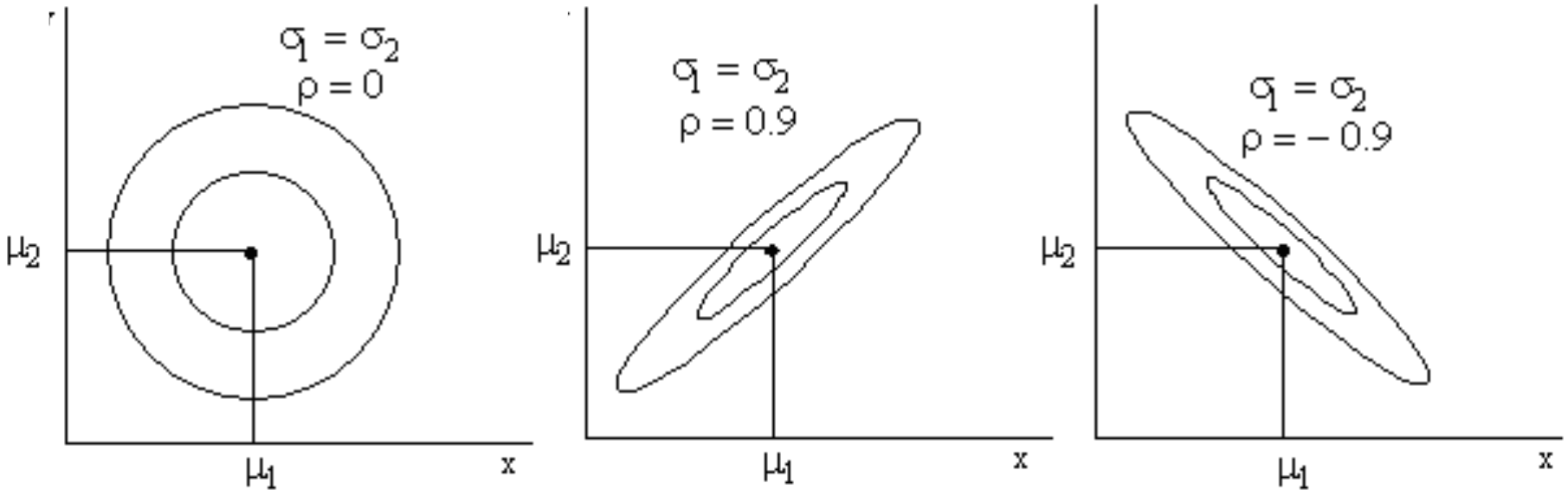
$V(X_1)$ $Cov(X_1, X_2)$ $V(X_2)$ 2 x 2

$$\left|\Sigma\right| = \sigma_{11}\sigma_{22} - \sigma_{12}^2 = \sigma_1^2\sigma_2^2\left(1-\rho^2\right)$$

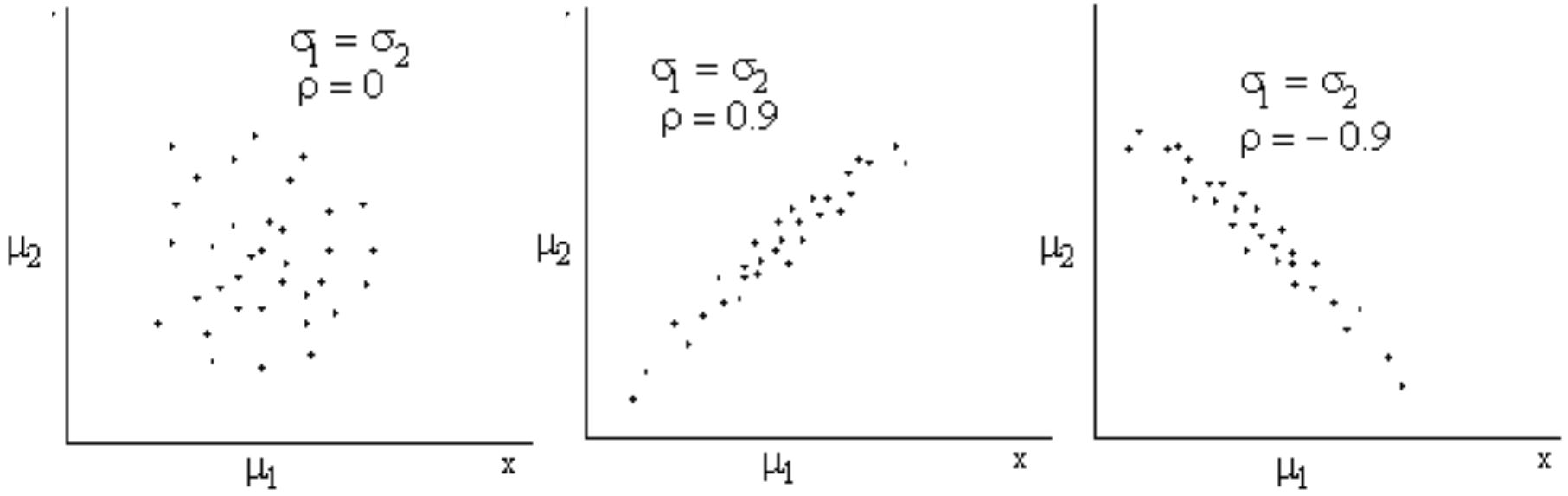
Surface Plots of the bivariate Normal distribution



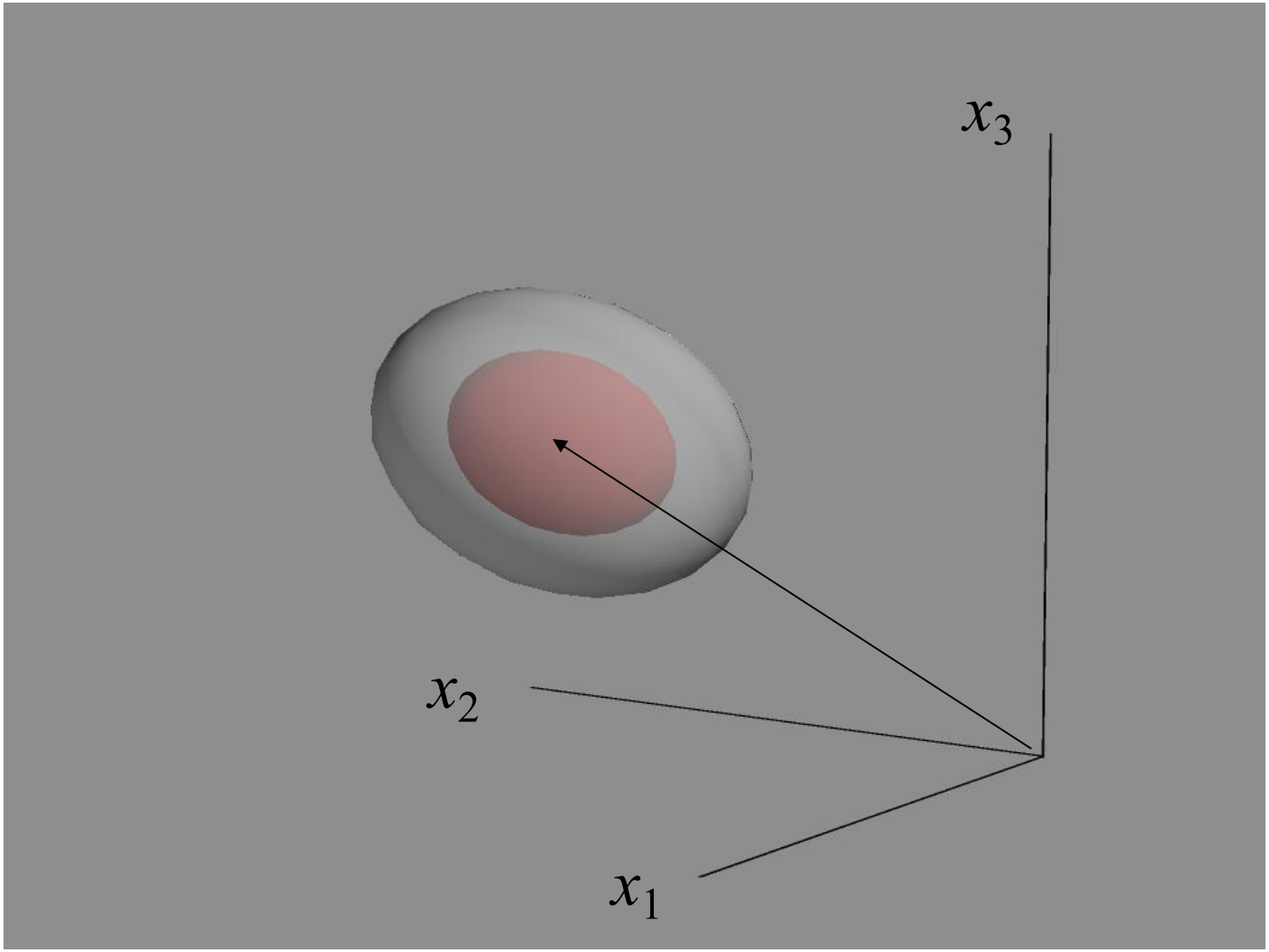
Contour Plots of the bivariate Normal distribution



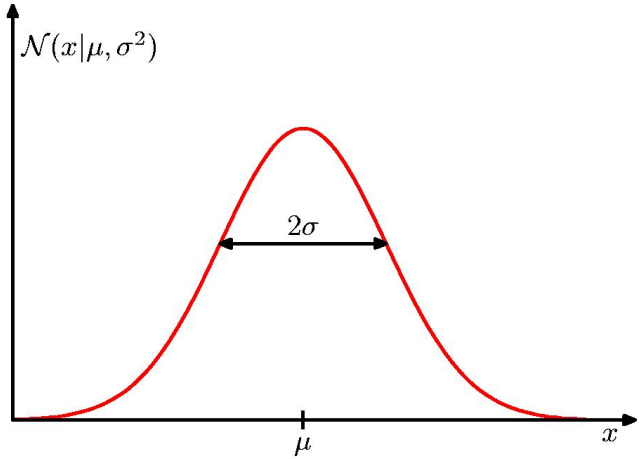
Scatter Plots of data from the bivariate Normal distribution



Trivariate Normal distribution



How to Estimate 1D Gaussian: MLE



• In the 1D Gaussian case, we simply set the mean and the variance to the sample mean and the sample variance:

$$\bar{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\overline{\sigma^2} = \frac{1}{n}\sum_{i=1}^{n} \left(x_i - \overline{\mu}\right)^2$$

How to Estimate p-D Gaussian: MLE

$$< X_1, X_2 \cdots, X_p > \sim N\left(\vec{\mu}, \Sigma\right)$$

$$\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix}_{p \times 1} \qquad \mu_i = \frac{1}{n} \sum_{j=1}^{N} X_j^{(i)}$$

$i \in \{1, 2, \cdots, p\}$: i-th feature

$j \in \{1, 2, \cdots, N\}$: j-th sample

$$\underset{p \times p}{\Sigma} = \begin{bmatrix} Var(X_1) & \cdots & Cov(X_i, X_j) & \cdots \\ & \ddots & & \\ & & & Var(X_p) \end{bmatrix}$$

i, j

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

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

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

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

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

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 $P(C = c_i)$ $i = 1, \cdots, L$

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

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

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

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 $\Rightarrow L \times (p + pf$

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

Diagonal Matrix

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

Each class' covariance matrix is diagonal

Today: More Generative Bayes Classifiers

- ✓ Generative Bayes Classifier
- ✓ Naïve Bayes Classifier
- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC ➔ LDA, QDA
- ✓ Discriminative vs. Generative

Not Naïve Gaussian means ?

Total # para $\Rightarrow L \times \{p + p \times p\}$ (M/C, Σ/C)

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 $\Rightarrow L \times (p + p)$

Naïve

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

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

$\Sigma | C_i = \begin{bmatrix} \sigma_{1i} & & 0 \\ & \sigma_{2i} & \\ 0 & & \sigma_{pi} \end{bmatrix}$

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

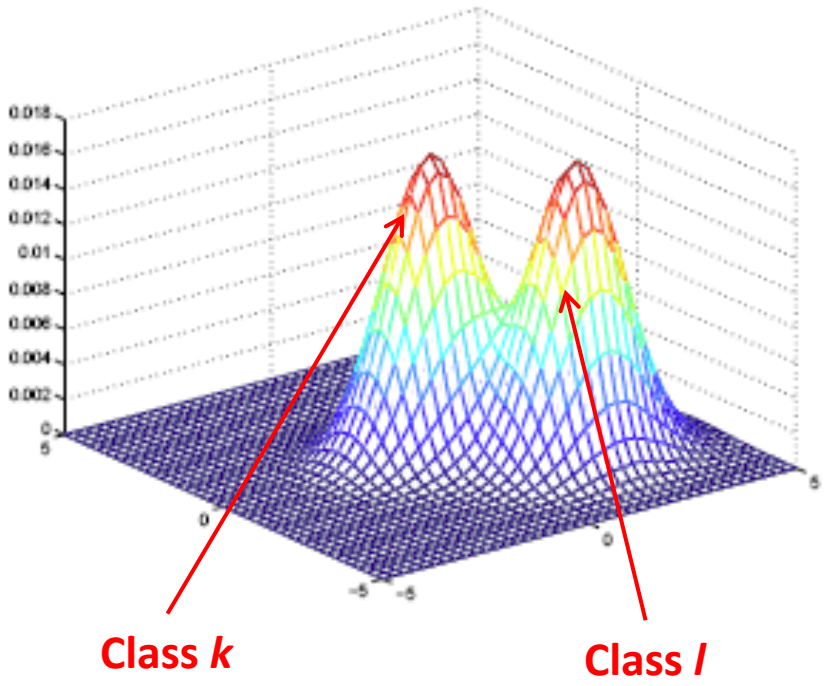
(1) covariance matrix are the same across classes

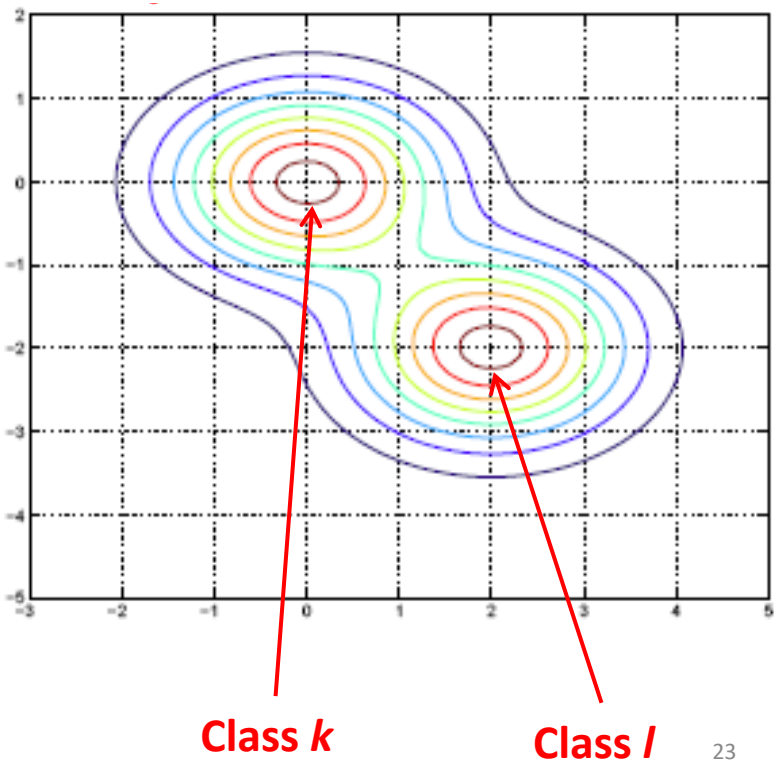
➔ LDA (Linear Discriminant Analysis)

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

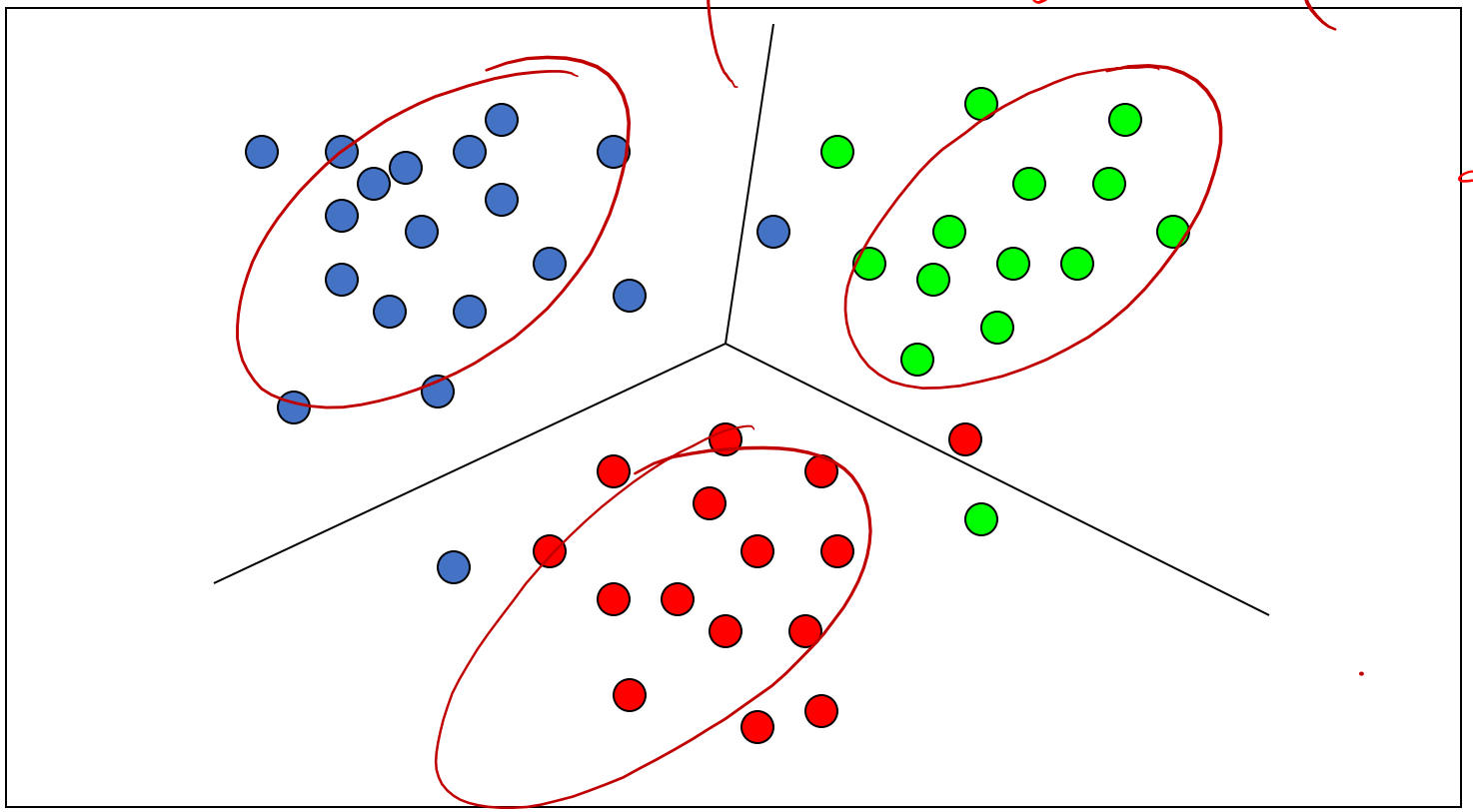
Each class' covariance matrix is the same

The Gaussian Distribution are shifted versions of each other





Visualization (three classes)



LDA

$\{Lxp + p^2/2\}$

$\{\vec{\mu}_1, \vec{\mu}_2, \vec{\mu}_L, \Sigma_{pxp}\}$

$$\arg\max_k P(C_k \mid X) = \arg\max_k P(X, C_k) = \arg\max_k P(X \mid C_k) P(C_k)$$

$$= \arg\max_k \log\{P(X \mid C_k) P(C_k)\}$$

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 :

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

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

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

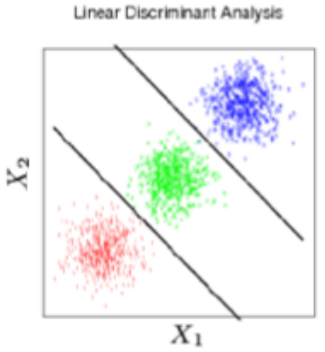
$$+ x^T \Sigma^{-1}(\mu_k - \mu_\ell) = 0$$

b

a

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

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



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

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

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

- Note

Linear Discriminant Function for LDA

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

Today: More Generative Bayes Classifiers

- ✓ Generative Bayes Classifier
- ✓ Naïve Bayes Classifier
- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC ➔ LDA, QDA
- ✓ Discriminative vs. Generative

(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

$K(p+p^2)$

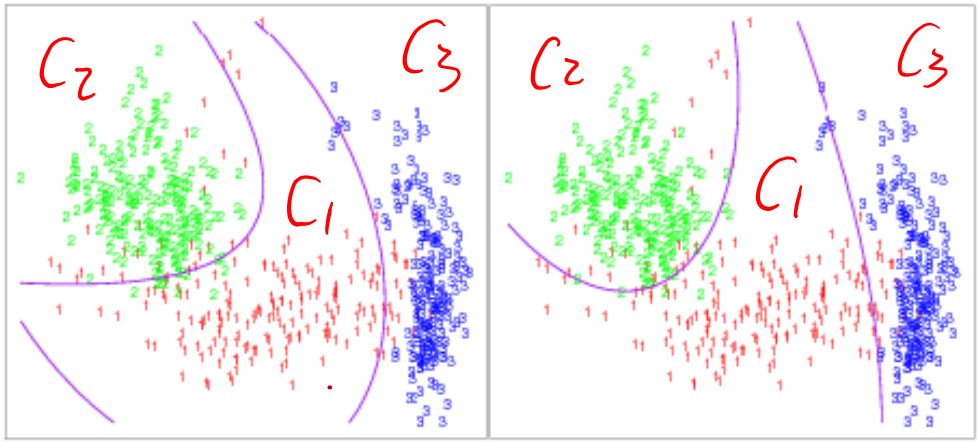
$\{\mu_k, \Sigma_k\}$

- 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

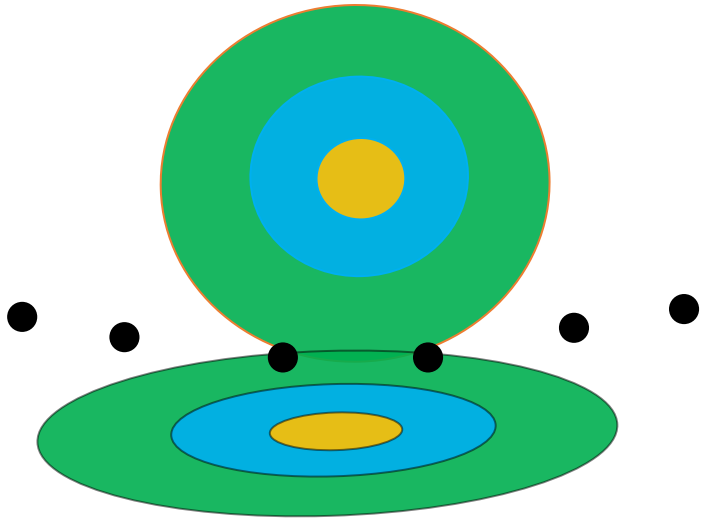
- A compromise between LDA and QDA.
- Shrink the separate covariances of QDA toward a common covariance as in LDA.
- Regularized covariance matrices:

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

- 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

$\Sigma_1 = \Lambda_1$

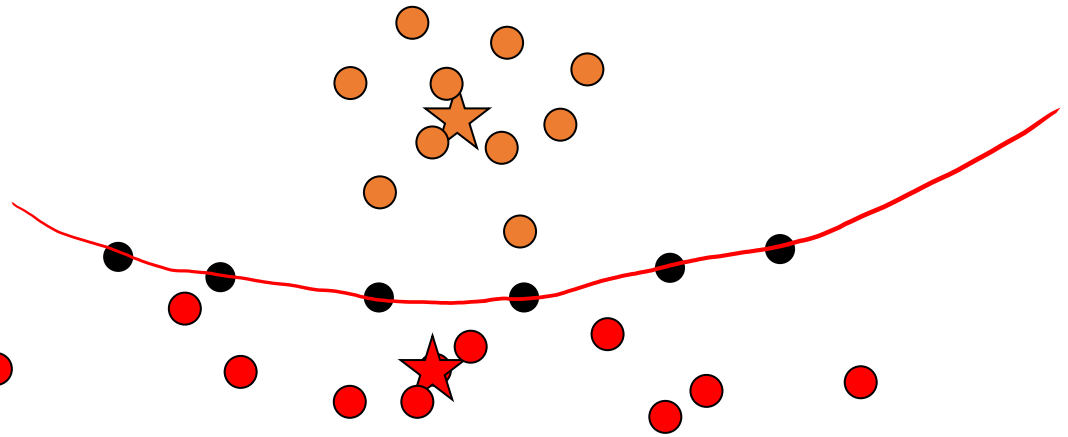
$\Sigma_2 = \Lambda_2$

diagonal

Gaussian Bayes Classifier

Naive

(GNBC)



Size of GNBC model

$O(Kp + Kp)$

mean diag-Σ

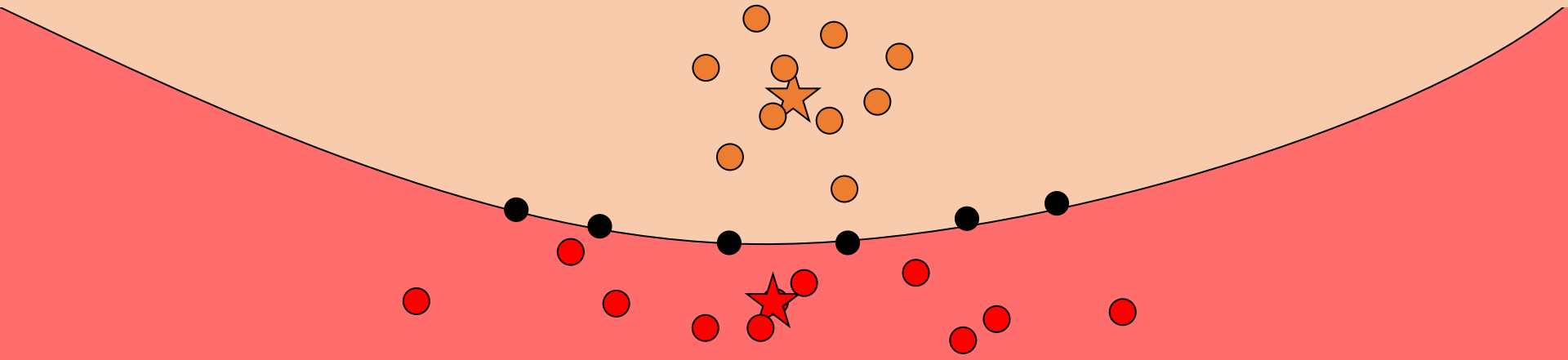
vs. Size of LDA model

$O(Kp + p^2/2)$

Quadratic decision Boundary

Gaussian Bayes Classifier

Blue Team



Red Team

Naïve Gaussian Bayes Classifier is not a linear classifier!

Today: More Generative Bayes Classifiers

- ✓ Generative Bayes Classifier
- ✓ Naïve Bayes Classifier
- ✓ Gaussian Bayes Classifiers
  - Gaussian distribution
  - Naïve Gaussian BC
  - Not-naïve Gaussian BC ➔ LDA, QDA
- ✓ Discriminative vs. Generative

Discriminative vs. Generative

Generative approach

- Model the joint distribution p(X, C) using

  $p(X \mid C = c_k)$ and $p(C = c_k)$

Class prior

Discriminative approach

- Model the conditional distribution p(c| X) directly

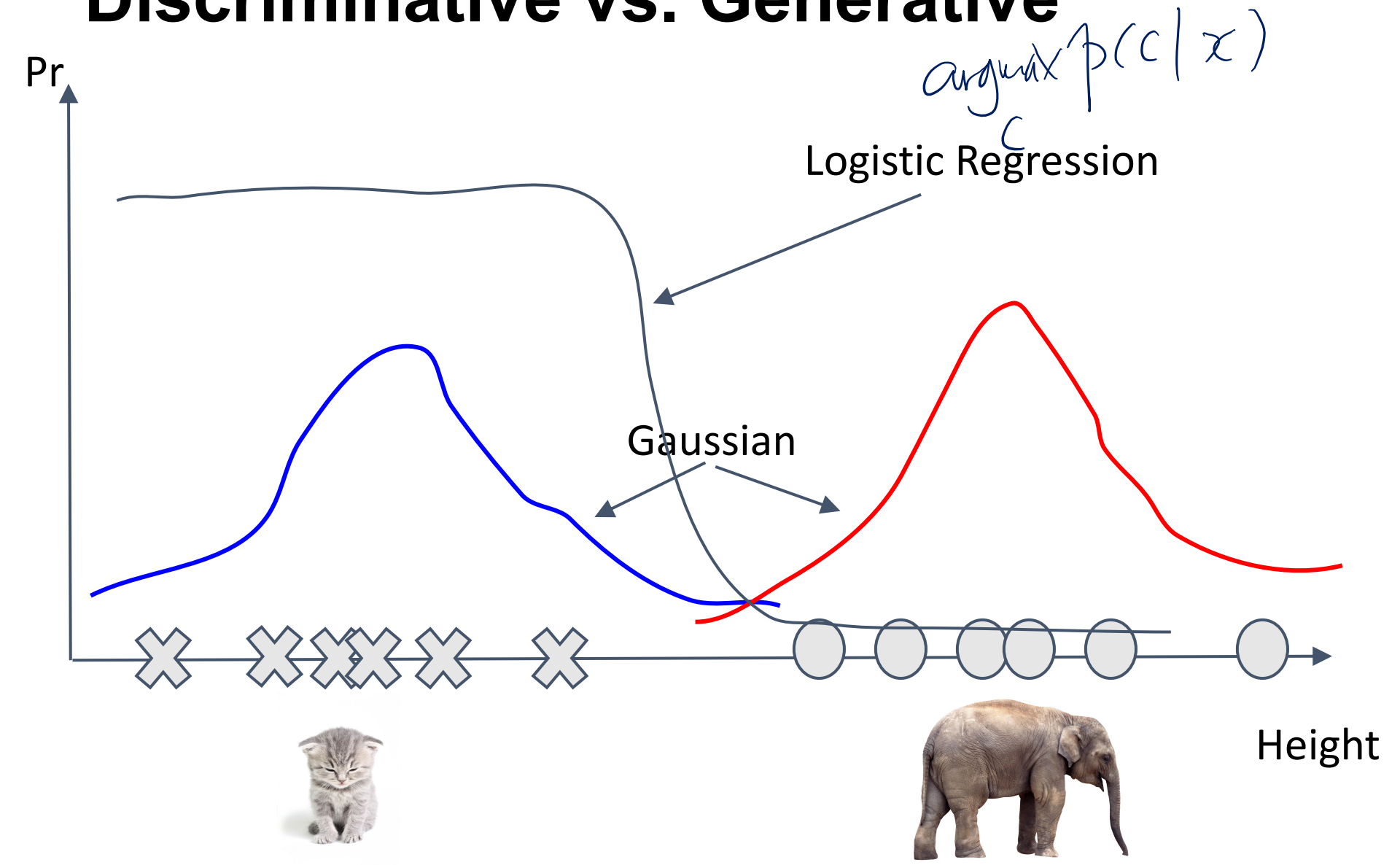
e.g.,

$$P(C=1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 * X)}}$$

Discriminative vs. Generative

- Empirically, generative classifiers approach their asymptotic error faster than discriminative ones
  - Good for small training set
  - Handle missing data well (EM)
- Empirically, discriminative classifiers have lower asymptotic error than generative ones
  - Good for larger training set

Discriminative vs. Generative



LDA vs. Logistic Regression

- **LDA (Generative model)**
  - Assumes Gaussian class-conditional densities and a common covariance
  - Model parameters are estimated by maximizing the full log likelihood, parameters for each class are estimated independently of other classes, $Kp + \frac{p(p+1)}{2} + (K-1)$ parameters
  - Makes use of marginal density information $\Pr(x)$
  - Easier to train, low variance, more efficient if model is correct
  - Higher asymptotic error, but converges faster

- **Logistic Regression (Discriminative model)**
  - Assumes class-conditional densities are members of the (same) exponential family distribution
  - Model parameters are estimated by maximizing the conditional log likelihood, simultaneous consideration of all other classes, $(K-1)(p+1)$parameters
  - Ignores marginal density information $\Pr(x)$
  - Harder to train, robust to uncertainty about the data generation process
  - Lower asymptotic error, but converges more slowly

Discriminative vs. Generative

- Definitions
  - h_{gen} and h_{dis}: generative and discriminative classifiers
  - $h_{gen, inf}$ and $h_{dis, inf}$: same classifiers but trained on the entire population (asymptotic classifiers)
  - n → infinity, $h_{gen} \rightarrow h_{gen, inf}$ and $h_{dis} \rightarrow h_{dis, inf}$

Ng, Jordan,. "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes." *Advances in neural information processing systems* 14 (2002): 841.

Discriminative vs. Generative

Proposition 1:

vs. h_{true}

$$\epsilon\left(h_{dis,\inf}\right) \leq \epsilon\left(h_{gen,\inf}\right)$$

Proposition 2:

= asymptotic error

Proposition 1 states that aymptotically, the error of the discriminative logistic regression is smaller than that of the generative naive Bayes. This is easily shown

- ϵ : generalization error

Logistic Regression vs. NBC

Discriminative classifier (Logistic Regression)

- Smaller asymptotic error
- Slow convergence ~ O(p)

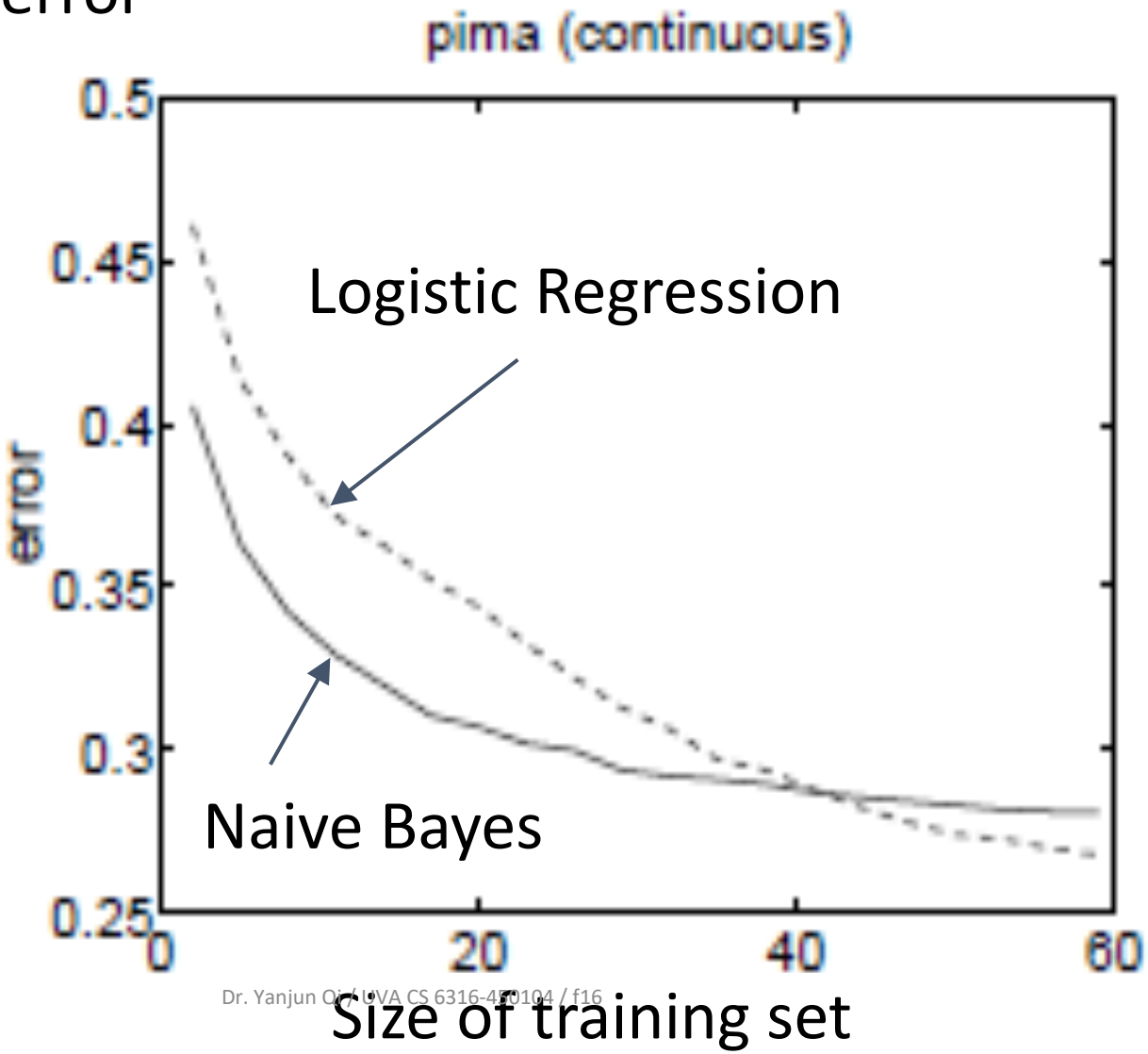
Generative classifier (Naive Bayes)

- Larger asymptotic error
- Can handle missing data (EM)
- Fast convergence ~ O(lg(p))

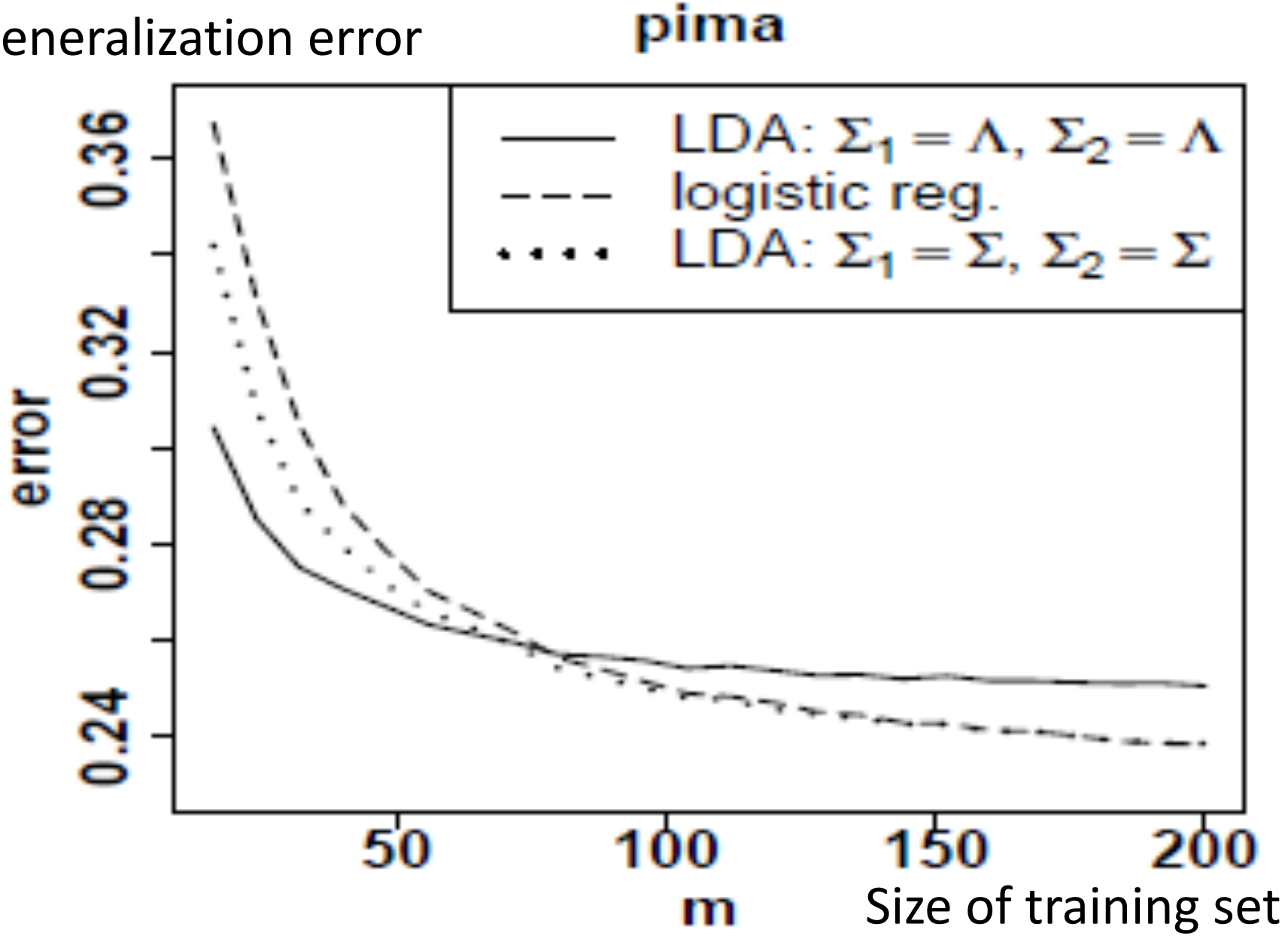
In numerical analysis, the speed at which a convergent sequence approaches its limit is called the rate of convergence.

Ng, Jordan,. "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes." *Advances in neural information processing systems* 14 (2002): 841.

generalization error



generalization error



Size of training set

Xue, Jing-Hao, and D. Michael Titterington. "Comment on “On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes”."*Neural processing letters* 28.3 (2008): 169-187.

Discriminative vs. Generative

- Empirically, generative classifiers approach their asymptotic error faster than discriminative ones
  - Good for small training set
  - Handle missing data well (EM)
- Empirically, discriminative classifiers have lower asymptotic error than generative ones
  - Good for larger training set

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

Generative Bayes Classifier

Task

↓

Representation

↓

Score Function

↓

Search/Optimization

↓

Models, Parameters

classification

↓

Prob. models p(X|C)

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

↓

EPE with 0-1 loss ➔ MAP Rule

↓

Many options

↓

Prob. Models' Parameter

Bernoulli Naïve

$$p(W_i = true \mid c_k) = p_{i,k}$$

Gaussian Naive

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

Multinomial

$$P(W_1 = n_1, \ldots, W_v = n_v \mid c_k) = \frac{N!}{n_{1k}!\, n_{2k}!\, .. n_{vk}!} \theta_{1k}^{n_{1k}} \theta_{2k}^{n_{2k}} .. \theta_{vk}^{n_{vk}}$$

References

- Prof. Tan, Steinbach, Kumar's "Introduction to Data Mining" slide
- Prof. Andrew Moore's slides
- Prof. Eric Xing's slides
- Prof. Ke Chen NB slides
- Hastie, Trevor, et al. *The elements of statistical learning*. Vol. 2. No. 1. New York: Springer, 2009.