

Machine/Statistical Learning

Generative models: Discriminant Analysis, Naïve Bayes

SICOM, M2 Sigma

2022-23

Generative models

Two kinds of approaches based on a model :

1. Discriminative approaches : direct learning of $p(Y|X)$,
e.g. Regression, logistic regression ← next lecture
2. Generative models : learning of the joint distribution $p(X, Y)$

$$p(X, Y) = \underbrace{p(X|Y)}_{\text{likelihood}} \underbrace{\Pr(Y)}_{\text{prior}},$$

e.g. linear/quadratic discriminant analysis, Naïve Bayes ← today's lecture

Discriminant functions

For both model based approaches, Bayes classifier is defined as

$$f^*(x) = \arg \max_{k \in \mathcal{Y}} \Pr(Y = k | X = x)$$

- ▶ equivalent to consider a set of functions $\delta_k(x)$, for $k \in \mathcal{Y}$, derived from a monotone transformation of posterior probability $\Pr(Y = k | X = x)$
- ▶ decision boundary between classes k and l is then defined as the set $\{x \in \mathcal{X} : \delta_k(x) = \delta_l(x)\}$

Definition

$\delta_k(x)$ are called the **discriminant functions** of each class k

- ☞ x is predicted in the k_0 class such that $k_0 = \arg \max_{k \in \mathcal{Y}} \delta_k(x)$

Generative models : Estimation problem

Assumptions

- ▶ classification problem with K classes : $Y \in \mathcal{Y} = \{1, \dots, K\}$,
- ▶ input variables : $X \in \mathbb{R}^P$

Bayes rule :

$$\Pr(Y = k | X = x) = \frac{p(x | Y = k) \Pr(Y = k)}{p(x)} = \frac{p(x | Y = k) \Pr(Y = k)}{\sum_{j=1}^K p(x | Y = j) \Pr(Y = j)}.$$

In practice, the following quantities are unknown :

- ▶ densities of each class $p_k(x) \equiv p(x | Y = k)$
- ▶ weights, or prior probabilities, of each class $\pi_k \equiv \Pr(Y = k)$

Estimation problem

These quantities must be learned on a training set :

learning problem \Leftrightarrow estimation problem in a parametric or not way




Discriminant Analysis

Two kinds of Discriminant Analysis :

- ▶ Linear Discriminant Analysis
- ▶ Quadratic Discriminant Analysis

In both cases, the key assumption is that, within each class, the input variables X_i are assumed to be normally distributed.

Supplementary materials

-  short (12mn) Sidney Univ. online video
https://www.youtube.com/watch?time_continue=719&v=D4C7YbfFQSk&feature=emb_logo
-  Wikipedia page (quite complete and detailed)
https://en.wikipedia.org/wiki/Linear_discriminant_analysis
-  short and simple Scikit-learn documentation (with examples)
https://scikit-learn.org/stable/modules/lda_qda.html

Quadratic Discriminant Analysis (QDA)

Supervised classification assumptions

- ▶ $X \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \{1, \dots, K\}$,
- ▶ sized n training set $(X_1, Y_1), \dots, (X_n, Y_n)$

QDA Assumptions

The input variables X , given a class $Y = k$, are distributed according to a parametric and Gaussian distribution :

$$X|Y = k \sim \mathcal{N}(\mu_k, \Sigma_k) \Leftrightarrow p_k(x) = \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)}$$

The Gaussian parameters are, for each class $k = 1, \dots, K$

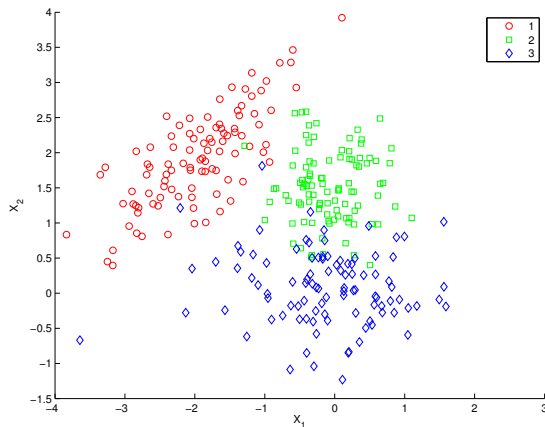
- ▶ mean vectors $\mu_k \in \mathbb{R}^p$,
- ▶ covariance matrices $\Sigma_k \in \mathbb{R}^{p \times p}$,
- ▶ set of parameters $\theta_k \equiv \{\mu_k, \Sigma_k\}$, plus the weights π_k , for $k = 1, \dots, K$.

Example

Mixture of $K = 3$ Gaussians

► $Y \in \{1, 2, 3\}$

► $X \in \mathbb{R}^2$

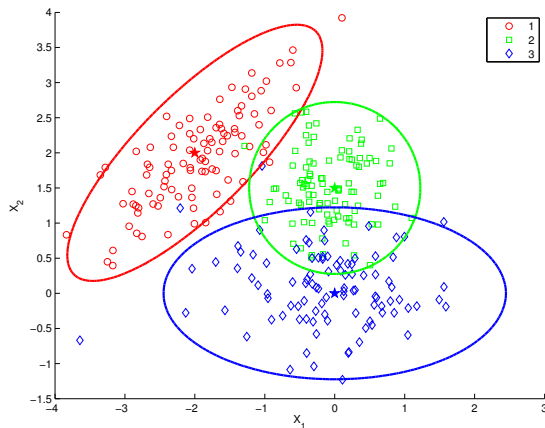


Example

Mixture of $K = 3$ Gaussians

► $Y \in \{1, 2, 3\}$

► $X \in \mathbb{R}^2$



True mean μ_k and covariance Σ_k parameters, for $k = 1, 2, 3$

QDA parameter estimation

Log-likelihood

For the training set,

$$\begin{aligned}\ell(\theta_1, \dots, \theta_K, \pi_1, \dots, \pi_{K-1}) &= \log p((x_1, y_1), \dots, (x_n, y_n)), \\ &= \sum_{i=1}^n \log p((x_i, y_i)), \quad \leftarrow \text{i.i.d. training set,} \\ &= \sum_{i=1}^n \log [p(x_i | y_i) \Pr(y_i)], \\ &= \sum_{i=1}^n \log [\pi_{y_i} p_{y_i}(x_i; \theta_{y_i})].\end{aligned}$$

Rk : $\pi_K = 1 - \sum_{j=1}^{K-1} \pi_j$ is not a parameter

QDA parameter estimation (Cont'd)

Notations

- ▶ $n_k = \#\{y_i = k\}$ is the number of training samples in class k ,
- ▶ $\sum_{y_i=k}$ is the sum over all the indices i of the training samples in class k

(Unbiased) Maximum likelihood estimators (MLE)

- ▶ $\hat{\pi}_k = \frac{n_k}{n}$, \leftarrow sample proportion
- ▶ $\hat{\mu}_k = \frac{\sum_{y_i=k} x_i}{n_k}$, \leftarrow sample mean
- ▶ $\hat{\Sigma}_k = \frac{1}{n_k - 1} \sum_{y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$, \leftarrow sample covariance

Rk : $\frac{1}{n_k - 1}$ is a bias correction factor for the covariance MLE (otherwise $\frac{1}{n_k}$)

QDA decision rule

The classification rule becomes

$$\begin{aligned} f(x) &= \arg \max_{k \in \mathcal{Y}} \Pr(Y = k | X = x, \hat{\theta}, \hat{\pi}), \\ &= \arg \max_{k \in \mathcal{Y}} \underbrace{\log \Pr(Y = k | X = x, \hat{\theta}, \hat{\pi})}_{\delta_k(x)}, \end{aligned}$$

where

$$\delta_k(x) = -\frac{1}{2} \log |\hat{\Sigma}_k| - \frac{1}{2} (x - \hat{\mu}_k)^T \hat{\Sigma}_k^{-1} (x - \hat{\mu}_k) + \log \hat{\pi}_k - C_{st},$$

is the **discriminant function**

Remarks

1. different rule than the Bayes classifier as θ replaced by $\hat{\theta}$ (and π replaced by $\hat{\pi}$)
2. when $n \gg p$, $\hat{\theta} \rightarrow \theta$ (and $\hat{\pi} \rightarrow \pi$) : convergence to the optimal classifier if the Gaussian model is correct...

QDA decision boundary

The boundary between two classes k and l is described by the equation

$$\delta_k(x) = \delta_l(x) \Leftrightarrow C_{k,l} + L_{k,l}^T x + x^T Q_{k,l} x = 0, \quad \leftarrow \text{quadratic equation}$$

where

$$\blacktriangleright C_{k,l} = -\frac{1}{2} \log \frac{|\hat{\Sigma}_k|}{|\hat{\Sigma}_l|} + \log \frac{\hat{\pi}_k}{\hat{\pi}_l} - \frac{1}{2} \hat{\mu}_k^T \hat{\Sigma}_k^{-1} \hat{\mu}_k + \frac{1}{2} \hat{\mu}_l^T \hat{\Sigma}_l^{-1} \hat{\mu}_l, \quad \leftarrow \text{scalar}$$

$$\blacktriangleright L_{k,l} = \hat{\Sigma}_k^{-1} \hat{\mu}_k - \hat{\Sigma}_l^{-1} \hat{\mu}_l, \quad \leftarrow \text{vector in } \mathbb{R}^p$$

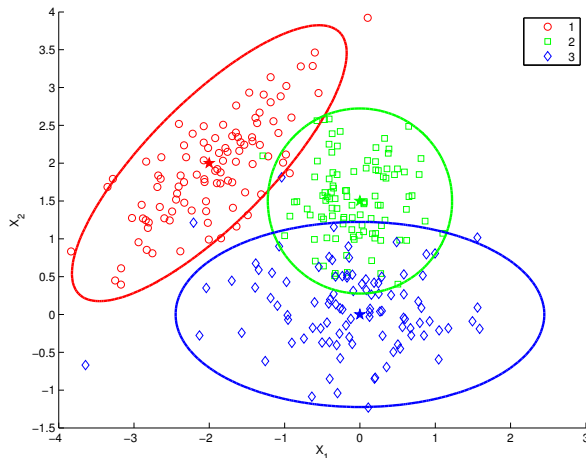
$$\blacktriangleright Q_{k,l} = \frac{1}{2} \left(-\hat{\Sigma}_k^{-1} + \hat{\Sigma}_l^{-1} \right), \quad \leftarrow \text{matrix in } \mathbb{R}^{p \times p}$$

👉 Quadratic discriminant analysis

QDA example

Mixture of $K = 3$ Gaussians

- Estimation of the parameters $\hat{\mu}_k$, $\hat{\Sigma}_k$ and $\hat{\pi}_k$, for $k = 1, 2, 3$

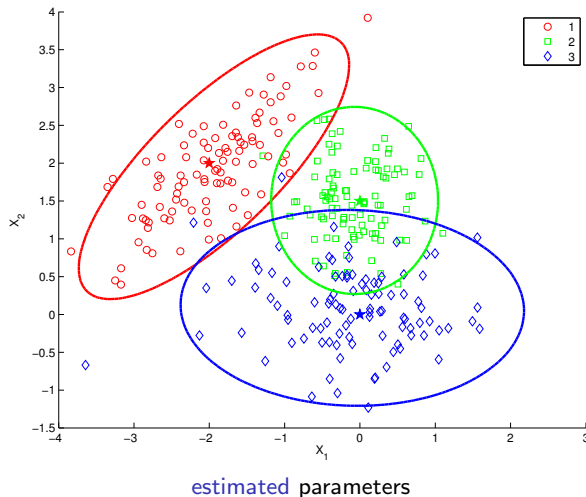


true parameters

QDA example

Mixture of $K = 3$ Gaussians

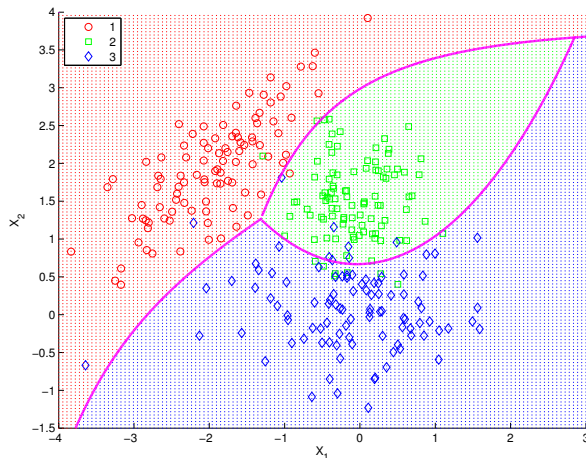
- Estimation of the parameters $\hat{\mu}_k$, $\hat{\Sigma}_k$ and $\hat{\pi}_k$, for $k = 1, 2, 3$



QDA example (Cont'd)

Mixture of $K = 3$ Gaussians

- Classification rule : $\arg \max_{k=1,2,3} \delta_k(x)$
- Quadratic boundaries $\{x; \delta_k(x) = \delta_l(x)\}$



LDA principle

LDA Assumptions

Additional simplifying assumption w.r.t. QDA : all the class covariance matrices are identical ("homoscedasticity"), i.e. $\Sigma_k = \Sigma$, for $k = 1, \dots, K$

(Unbiased) Maximum likelihood estimators (MLE)

- ▶ $\hat{\pi}_k$ and $\hat{\mu}_k$ are unchanged,
- ▶ $\hat{\Sigma} = \frac{1}{n-K} \sum_{k=1}^K \sum_{y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$, \leftarrow pooled covariance

Rk : $\frac{1}{n-K}$ is a bias correction factor for the covariance MLE (otherwise $\frac{1}{n}$)

LDA discriminant function

$$\delta_k(x) = -\frac{1}{2} \log |\hat{\Sigma}| - \frac{1}{2} (x - \hat{\mu}_k)^T \hat{\Sigma}^{-1} (x - \hat{\mu}_k) + \log \hat{\pi}_k + \text{Cst},$$

LDA decision boundary

The boundary between two classes k and l reduces to the equation

$$\delta_k(x) = \delta_l(x) \Leftrightarrow C_{k,l} + L_{k,l}^T x = 0, \quad \leftarrow \text{linear equation}$$

where

$$\blacktriangleright C_{k,l} = \log \frac{\hat{\pi}_k}{\hat{\pi}_l} - \frac{1}{2} \hat{\mu}_k^T \hat{\Sigma}^{-1} \hat{\mu}_k + \frac{1}{2} \hat{\mu}_l^T \hat{\Sigma}^{-1} \hat{\mu}_l, \quad \leftarrow \text{scalar}$$

$$\blacktriangleright L_{k,l} = \hat{\Sigma}^{-1} (\hat{\mu}_k - \hat{\mu}_l), \quad \leftarrow \text{vector in } \mathbb{R}^p$$

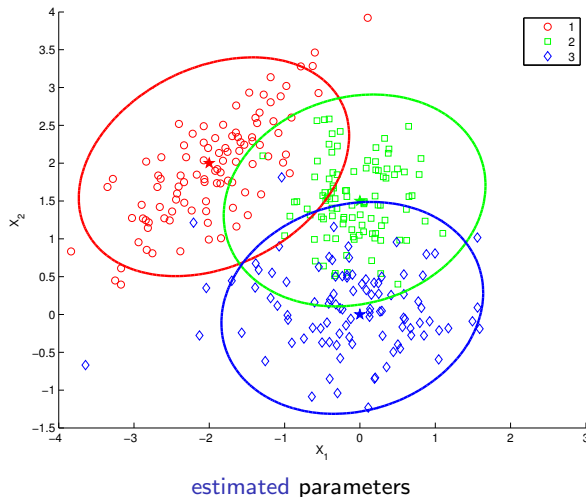
$$\blacktriangleright Q_{k,l} = 0,$$

👉 Linear discriminant analysis

Linear Discriminant Analysis (LDA)

Mixture of $K = 3$ Gaussians

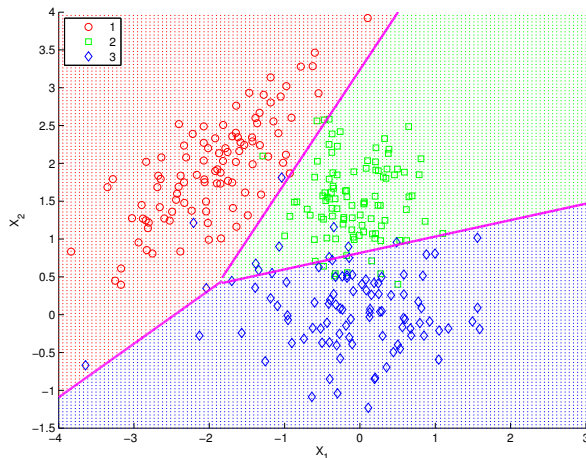
- Estimation of the parameters $\hat{\mu}_k$, $\hat{\pi}_k$, for $k = 1, 2, 3$, and $\hat{\Sigma}$



Linear Discriminant Analysis (LDA)

Mixture of $K = 3$ Gaussians

- Classification rule : $\arg \max_{k=1,2,3} \delta_k(x)$
- linear boundaries $\{x; \delta_k(x) = \delta_l(x)\}$



Complexity of discriminant analysis methods

Effective number of parameters

- ▶ LDA : $(K - 1) \times (p + 1) = O(Kp)$
- ▶ QDA : $(K - 1) \times \left(\frac{p(p+3)}{2} + 1 \right) = O(Kp^2)$

Remarks

- ▶ in high dimension, i.e. $p \approx n$ or $p > n$, LDA is more stable than QDA which is more prone to overfitting,
- ▶ both methods appear however to be robust on a large number of real-world datasets
- ▶ LDA can be viewed in some cases as a least squares regression method
- ▶ LDA performs a dimension reduction to a subspace of dimension $\leq K - 1$ generated by the vectors $z_k = \hat{\Sigma}^{-1}(\hat{\mu}_k - \hat{\mu}_K) \leftarrow$ dimension reduction from p to $K - 1$!

Naïve Bayes (NB)



NB classifiers

Family of "probabilistic classifiers" based on applying Bayes' theorem on a generative model, with strong (naïve) independence assumptions between the features.

Can be coupled with

- ▶ parametric models (Gaussian, Bernoulli, Multinomial,...) with maximum likelihood estimation
- ▶ or non-parametric models with kernel density estimation

Supplementary materials

-  Wikipedia page (quite detailed) https://en.wikipedia.org/wiki/Naive_Bayes_classifier
-  short and simple Scikit-learn documentation
https://scikit-learn.org/stable/modules/naive_bayes.html

Naïve Bayes (NB)

General assumptions

- ▶ $X = (X_1, \dots, X_p) \in \mathbb{R}^p$, $Y \in \mathcal{Y} = \{1, \dots, K\}$,

NB Assumption

Simplifying assumption : given Y , the components X_1, \dots, X_p are assumed to be **independent** :

$$p_k(x) = \prod_{j=1}^p p_{k,j}(x_j).$$

Remarks

- ▶ independence reduces one estimation problem in p dimensions to p much simpler 1D estimation problems ← prevent from curse of dimensionality
- ▶ independence assumption is **naïve**, i.e. not realistic in practice... but yields efficient/stable/robust approaches especially in high dimension !

Naïve Bayes for parametric estimation

Gaussian model

- ▶ NB + QDA : $X|Y = k \sim \mathcal{N}(\mu_k, \Sigma_k)$, where the Σ_k are **diagonal**, for $k = 1, \dots, K$
- ▶ NB + LDA : $X|Y = k \sim \mathcal{N}(\mu_k, \Sigma)$, where Σ is **diagonal**,

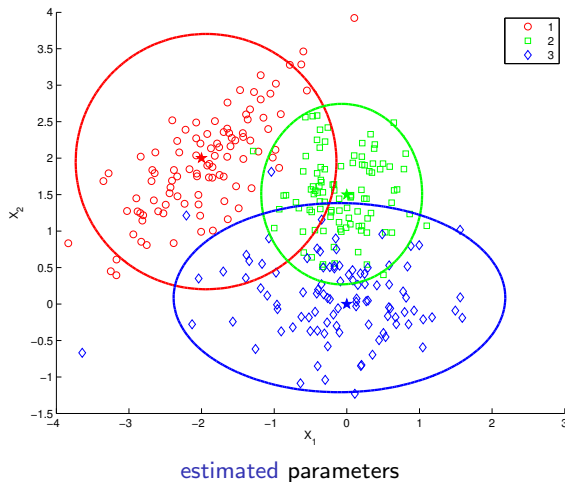
Other classical parametric models

- ▶ Bernoulli NB for binary events models (e.g., word occurrence vectors in text processing)
- ▶ Multinomial NB for multiple events models (e.g., word count vectors in text processing)
- ▶ Mixed models (e.g. Gaussian and Multinomial) for mixed quantitative/qualitative features
- ▶ ...

NB + QDA example

Mixture of $K = 3$ Gaussians

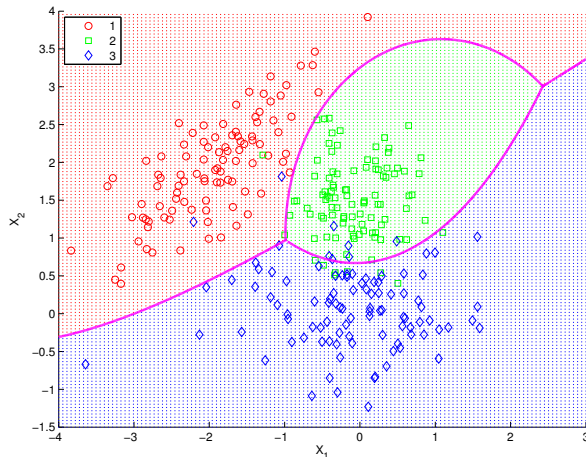
- Gaussian model : $X|Y = k \sim \mathcal{N}(\mu_k, \Sigma_k)$ with $\Sigma_k = \begin{pmatrix} \sigma_{1k}^2 & 0 \\ 0 & \sigma_{2k}^2 \end{pmatrix}$



Naïve Bayes (NB)

Mixture of $K = 3$ Gaussians

- Classification rule : $\arg \max_{k=1,2,3} \delta_k(x)$
- quadratic boundaries $\{x; \delta_k(x) = \delta_l(x)\}$



Naïve Bayes for non-parametric estimation

Non-parametric estimation of $p_{k,j}(x_j) = p(x_j | Y = k)$, where x_j is the j th component of x

Empirical approach

$$\hat{p}_{k,j}(x_j) = \frac{\#\{x_{j,i} \in V(x_j) \mid y_i = k\}}{n_k \lambda}$$

where $V_\lambda(x_j)$ is a neighborhood of x_j with volume λ (and $n_k = \#\{y_i = k\}$)

Parzen kernel approach

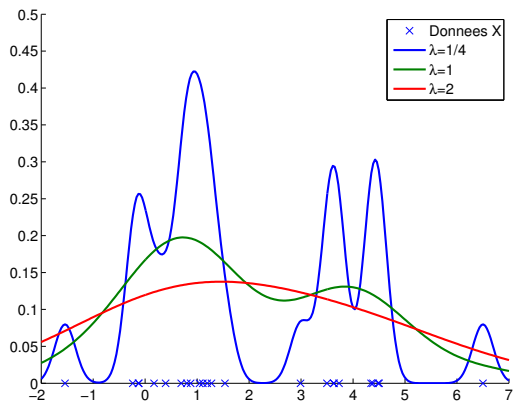
$$\hat{p}_{k,j}(x_j) = \frac{1}{n_k \lambda} \sum_{i \text{ st } y_i = k} K_\lambda(x_j, x_{j,i})$$

where K_λ is a given kernel, e.g. :

- ▶ 0-1 kernel : $K_\lambda(x, x_i) = 1$ if $x_i \in V_\lambda(x)$, 0 otherwise \leftarrow empirical approach,
- ▶ 1D Gaussian kernel : $K_\lambda(x, x_0) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2\lambda^2}(x-x_0)^2}$,
 $\Rightarrow \hat{p}_{k,j}(x_j) = \frac{1}{n_k \lambda \sqrt{2\pi}} \sum_{i, y_i = k} e^{-\frac{1}{2\lambda^2}(x_j - x_{j,i})^2}$

Kernel density estimation

1D estimation : $X \in \mathbb{R}$



Complexity parameter λ (kernel bandwidth)

- ▶ large λ w.r.t. to the dispersion of $X \rightarrow$ under-fitting
- ▶ small λ w.r.t. to the dispersion of $X \rightarrow$ over-fitting

Conclusions

Generative models

- ▶ learning/estimation of $p(X, Y) = p(X|Y) \Pr(Y)$,
- ▶ derivation of $\Pr(Y|X)$ from Bayes rule,

Different assumptions on the class densities $p_k(x) = p(X = x|Y = k)$

- ▶ QDA/LDA : Gaussian parametric model
 - ▶ performs well on many real-word datasets
 - ▶ LDA is especially useful when n is small
- ▶ NB : independence of the feature X components given Y
 - ▶ useful when p is very large (high dimension)

Perspectives

Discriminative approaches : direct learning of $\Pr(Y|X)$