

Machine Learning

Classification 1 and 2

Hiroshi Shimodaira and Hao Tang

2026 *Ver. 1.0.2*

Topics - you should be able to explain

- Data and data preprocessing
- Features and labels
- Statistical classification
- Bayes decision rule for classification
- Generative classifier vs discriminative classifier
- Curse of dimensionality
- Naive Bayes model
- Multivariate Gaussian distributions
- Gaussian discriminant analysis (GDA)
- Covariance matrices
- Decision regions and decision boundaries
- Minimum error rate classification, MAP decision rule
- Discriminant functions of GDA
- Linear discriminant analysis (LDA)

Topics - you should be able to explain (cont.)

- Linear classifiers
- Hyperplanes, decision boundaries, and decision regions
- Training of classifiers
- Loss and cost functions
- Logistic regression
- Extension of binary classification to multiclass classification
- Sigmoid and softmax functions

Data in machine learning

Types of data

- Numerical (quantitative): discrete / continuous
- Categorical (qualitative): nominal / ordinal
- Sequential / non-sequential

Examples

- Image data, video data, speech data
- Text data

Data in machine learning

Types of data

- Numerical (quantitative): discrete / continuous
- Categorical (qualitative): nominal / ordinal
- Sequential / non-sequential

Examples

- Image data, video data, speech data
- Text data

Data need to be collected and stored in a machine-readable form.

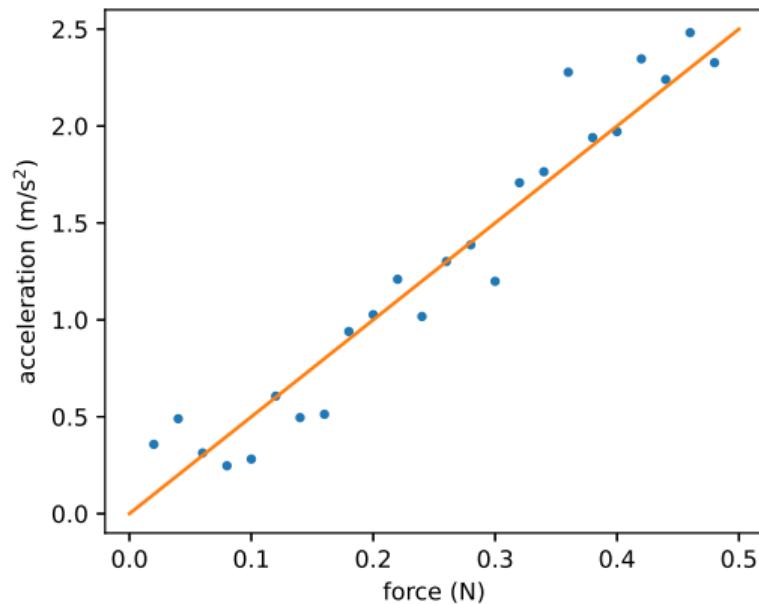


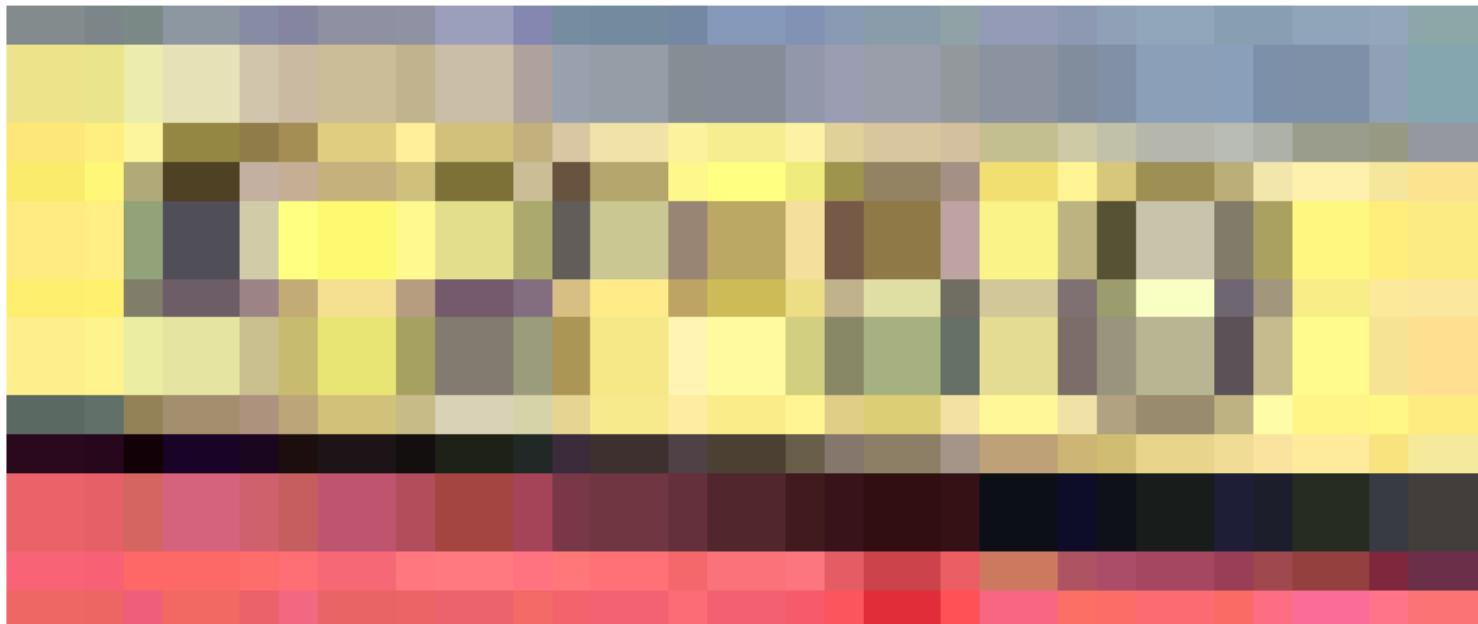
Image data



Image data



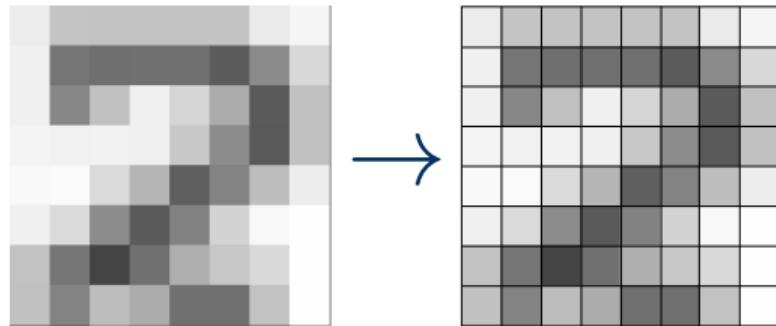
Image data



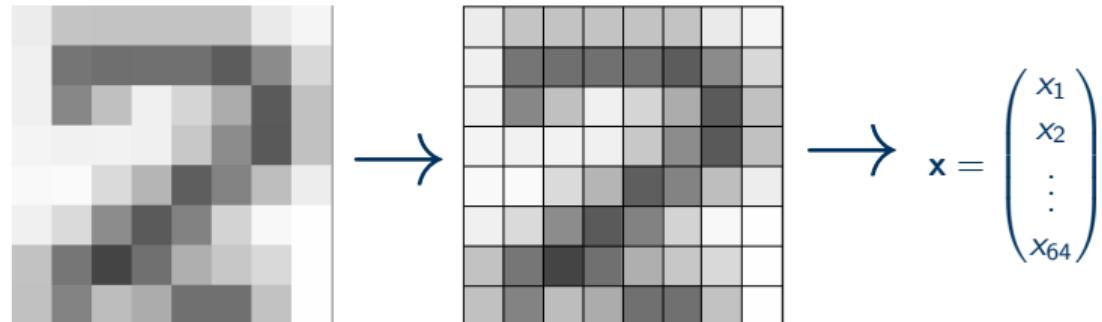
Pixel image to a feature vector



Pixel image to a feature vector



Pixel image to a feature vector



Turn each cell (pixel) into a number

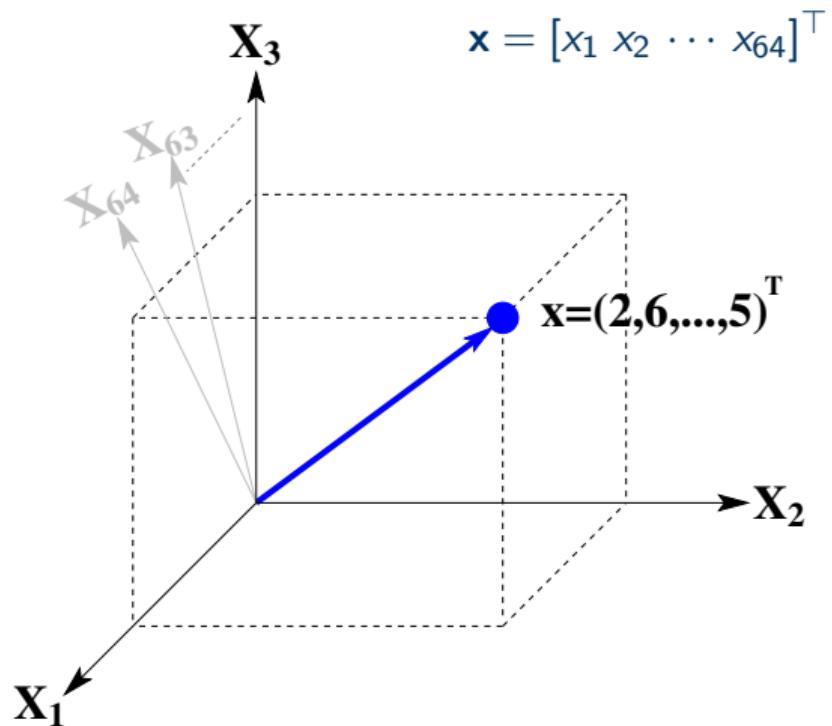
Unravel into a column vector, a *feature vector*

⇒ represented digit as a point in $64D$

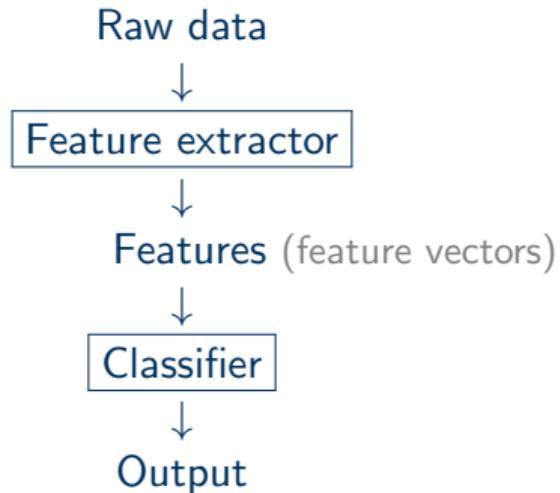
$$\mathbf{x} = (x_1 \ x_2 \ \cdots \ x_{64})^\top, \quad x_i \in [0, 127] \text{ or } x_i \in [0, 1]$$

<http://alex.seewald.at/digits/>

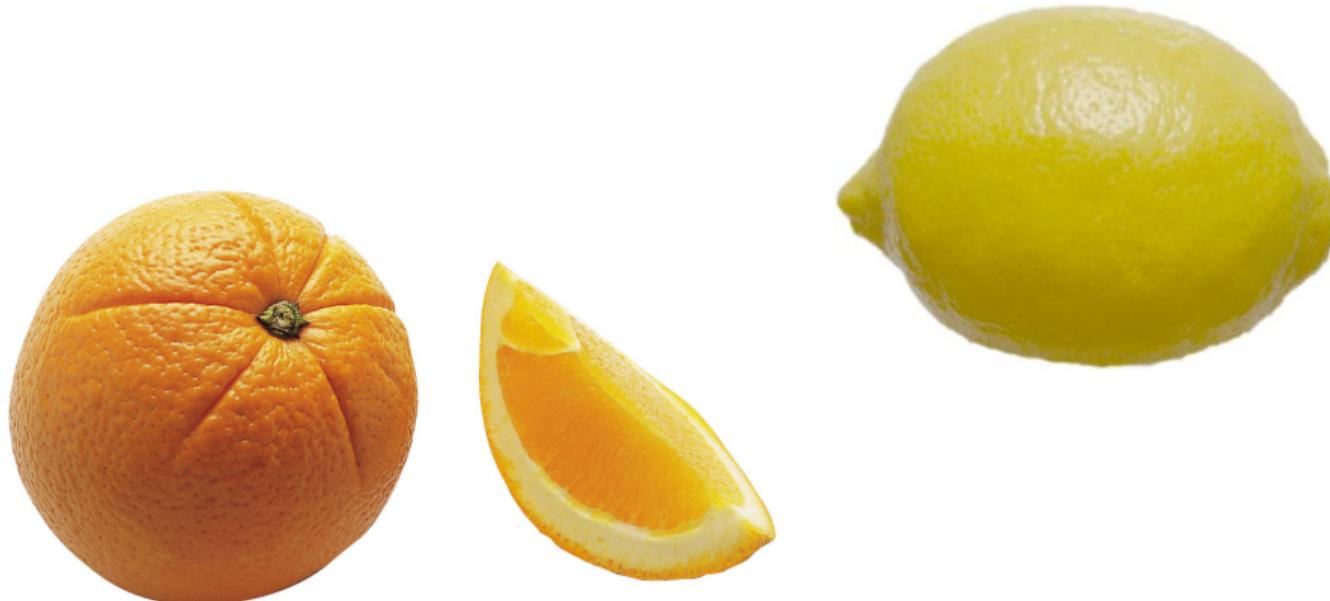
Image data as a point in a vector space



Feature/attribute extraction and classification

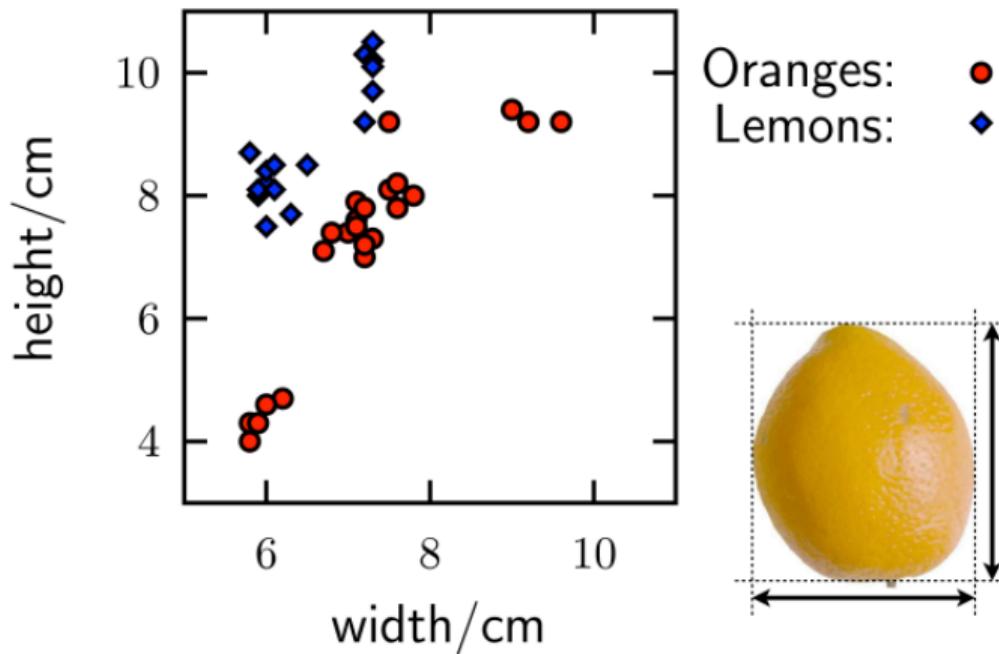


Classification of oranges and lemons



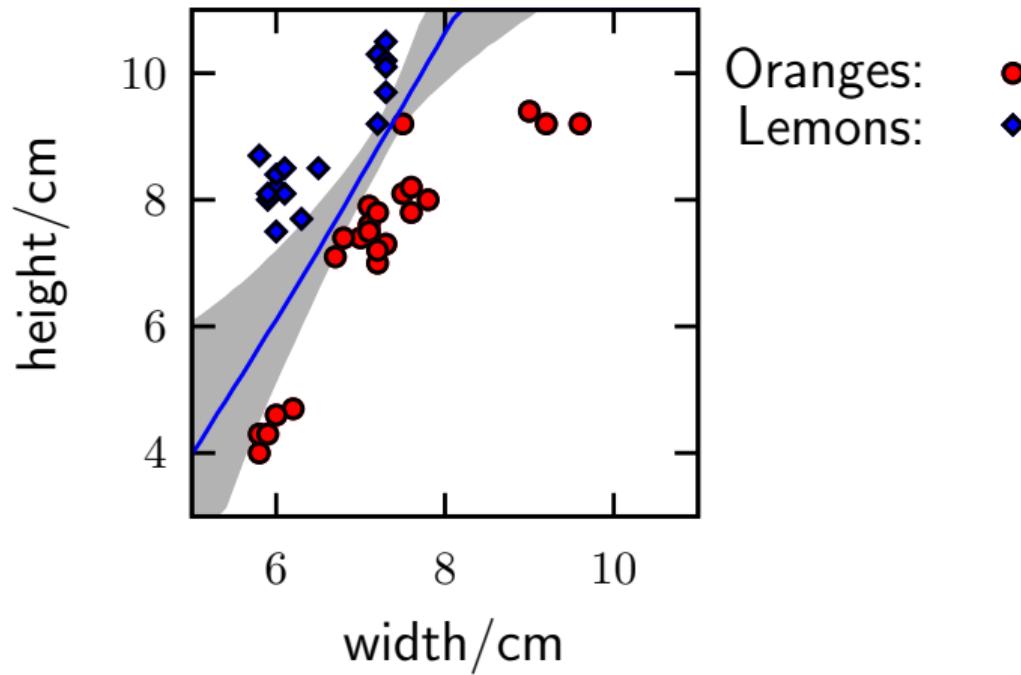
A two-dimensional space

Represent each sample as a point (w, h) in a 2D space



credit: Iain Murray

Classification



Statistical classification

- Classes: $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ Labels: $\mathcal{Y} = \{1, 2, \dots, K\}$

Statistical classification

- Classes: $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ Labels: $\mathcal{Y} = \{1, 2, \dots, K\}$
- Observation (feature vector): $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_d]^T \in \mathbb{R}^d$

Statistical classification

- Classes: $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ Labels: $\mathcal{Y} = \{1, 2, \dots, K\}$
- Observation (feature vector): $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_d]^\top \in \mathbb{R}^d$
- Bayes decision rule (for classification):

$$p(C_k | \mathbf{x}) > p(C_{k'} | \mathbf{x}) \quad \forall k' \neq k \quad \quad p(y=k | \mathbf{x}) > p(y=k' | \mathbf{x}) \quad \forall k' \neq k$$

$$\hat{y}(\mathbf{x}) = \arg \max_k p(C_k | \mathbf{x}) \quad (1)$$

Statistical classification

- Classes: $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ Labels: $\mathcal{Y} = \{1, 2, \dots, K\}$
- Observation (feature vector): $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_d]^\top \in \mathbb{R}^d$
- Bayes decision rule (for classification):

$$p(C_k | \mathbf{x}) > p(C_{k'} | \mathbf{x}) \quad \forall k' \neq k \quad p(y=k | \mathbf{x}) > p(y=k' | \mathbf{x}) \quad \forall k' \neq k$$

$$\hat{y}(\mathbf{x}) = \arg \max_k p(C_k | \mathbf{x}) \quad (1)$$

where

$$\overbrace{p(C_k | \mathbf{x})}^{\text{posterior}} = \overbrace{\frac{p(\mathbf{x} | C_k) p(C_k)}{p(\mathbf{x})}}^{\text{likelihood} \quad \text{prior}} = \frac{p(\mathbf{x} | C_k) p(C_k)}{\sum_{k'=1}^K p(\mathbf{x} | C_{k'}) p(C_{k'})} \quad (2)$$

$$p(y=k | \mathbf{x}) = \frac{p(\mathbf{x} | y=k) p(y=k)}{p(\mathbf{x})} = \frac{p(\mathbf{x} | y=k) p(y=k)}{\sum_{k'=1}^K p(\mathbf{x} | y=k') p(y=k')}$$

Statistical classification (*cont.*)

$$\begin{aligned}\hat{y}(\mathbf{x}) &= \arg \max_k p(C_k | \mathbf{x}) \\ &= \arg \max_k p(\mathbf{x} | C_k) p(C_k)\end{aligned}$$

$$p(C_k | \mathbf{x}) = \frac{p(\mathbf{x} | C_k) p(C_k)}{p(\mathbf{x})} = \frac{p(\mathbf{x} | C_k) p(C_k)}{\sum_{k'=1}^K p(\mathbf{x} | C_{k'}) p(C_{k'})}$$

Statistical classification (*cont.*)

$$\begin{aligned}\hat{y}(\mathbf{x}) &= \arg \max_k p(C_k | \mathbf{x}) \\ &= \arg \max_k p(\mathbf{x} | C_k) p(C_k)\end{aligned}$$

$$p(C_k | \mathbf{x}) = \frac{p(\mathbf{x} | C_k) p(C_k)}{p(\mathbf{x})} = \frac{p(\mathbf{x} | C_k) p(C_k)}{\sum_{k'=1}^K p(\mathbf{x} | C_{k'}) p(C_{k'})}$$

- *Generative classifier / approach* : models each term on RHS.

$$p(\mathbf{x} | C_k; \theta), p(C_k; \theta)$$

Statistical classification (*cont.*)

$$\begin{aligned}\hat{y}(\mathbf{x}) &= \arg \max_k p(C_k | \mathbf{x}) \\ &= \arg \max_k p(\mathbf{x} | C_k) p(C_k)\end{aligned}$$

$$p(C_k | \mathbf{x}) = \frac{p(\mathbf{x} | C_k) p(C_k)}{p(\mathbf{x})} = \frac{p(\mathbf{x} | C_k) p(C_k)}{\sum_{k'=1}^K p(\mathbf{x} | C_{k'}) p(C_{k'})}$$

- *Generative classifier / approach* : models each term on RHS.

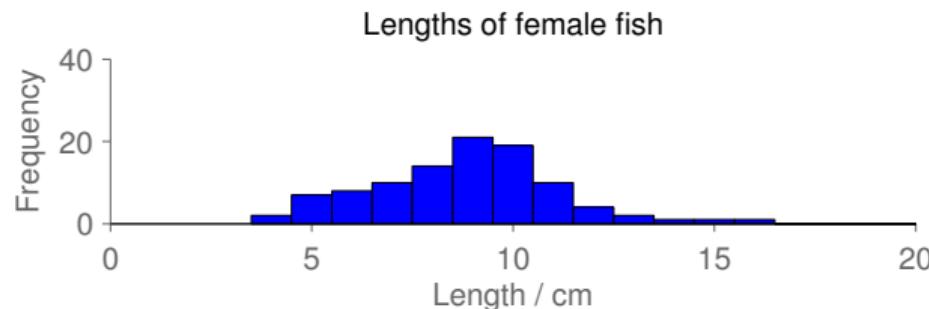
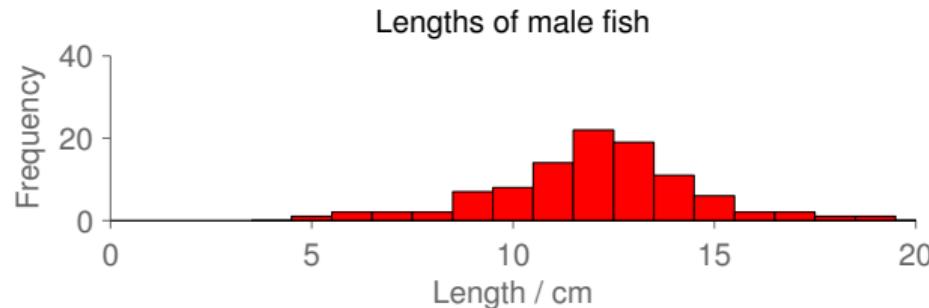
$$p(\mathbf{x} | C_k; \theta), p(C_k; \theta)$$

- *Discriminative classifier / approach* : models LHS directly

$$p(C_k | \mathbf{x}; \theta)$$

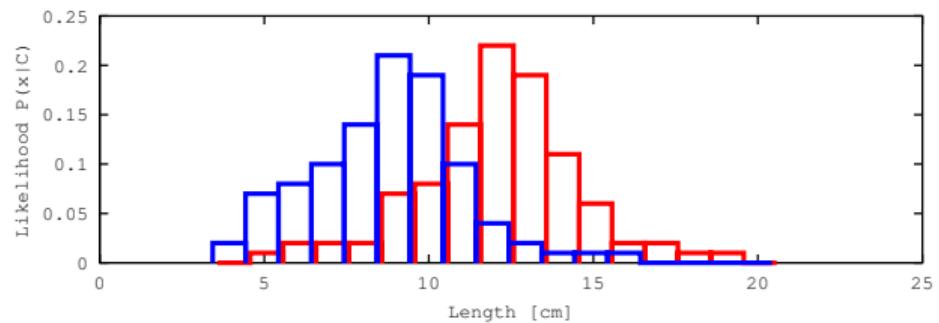
Example: determining the sex of fish

Histograms of fish lengths ($N_F = N_M = 100$)



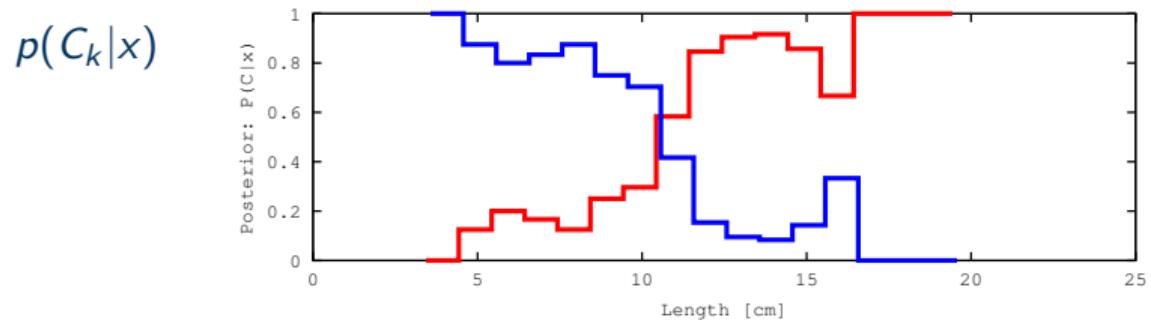
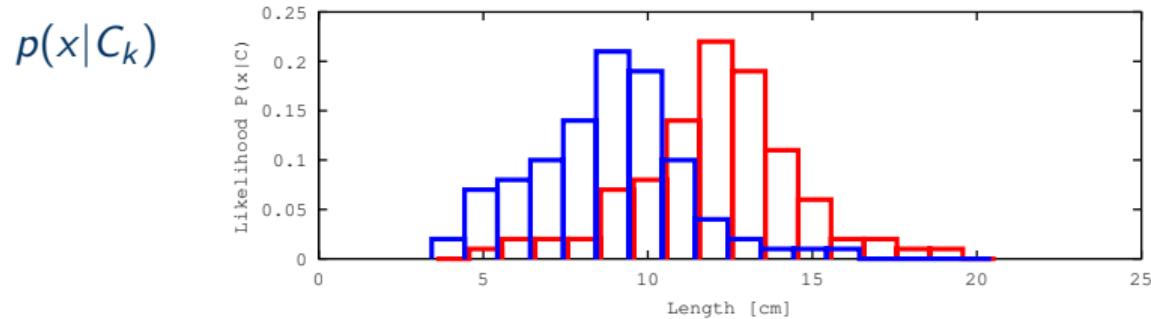
Example: determining the sex of fish (cont.)

$$p(x|C_k)$$



Example: determining the sex of fish (cont.)

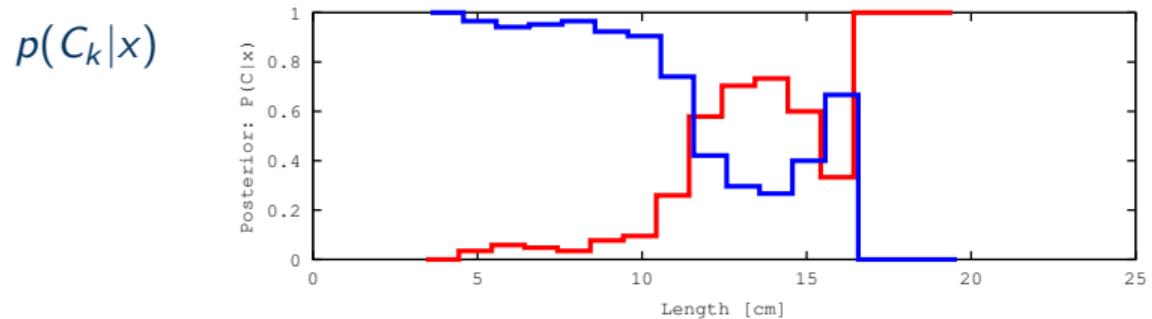
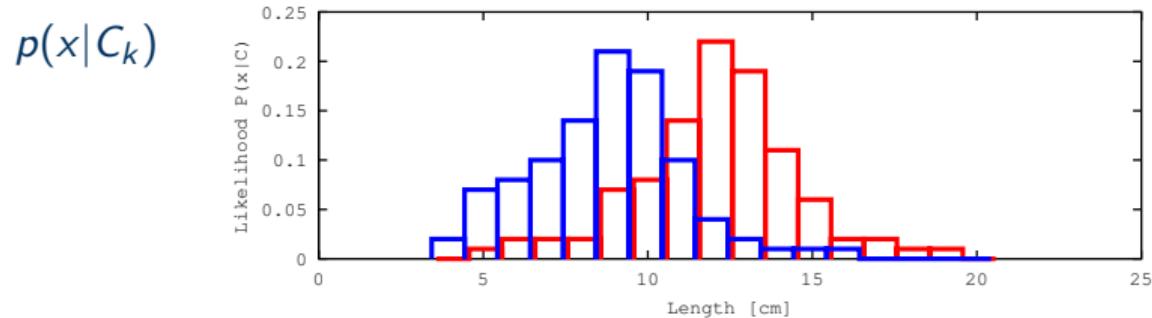
$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$



$$p(M) : p(F) = 1 : 1$$

Example: determining the sex of fish (cont.)

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$



$$p(M) : p(F) = 1 : 4$$

More features to improve classification accuracy!?

$$p(x | C_k) \approx \frac{n_{C_k}(x_1, \dots, x_d)}{N_{C_k}}$$

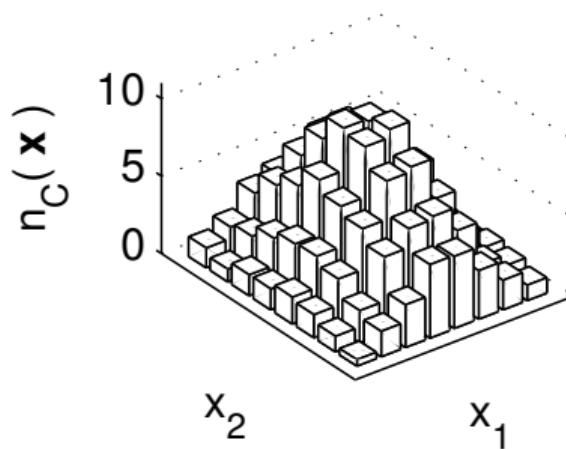
1D histogram: $n_{C_k}(x_1)$

More features to improve classification accuracy!?

$$p(\mathbf{x} | C_k) \approx \frac{n_{C_k}(x_1, \dots, x_d)}{N_{C_k}}$$

1D histogram: $n_{C_k}(x_1)$

2D histogram: $n_{C_k}(x_1, x_2)$



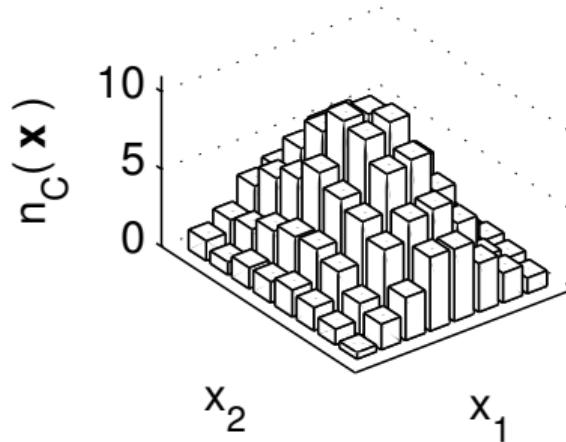
More features to improve classification accuracy!?

$$p(x | C_k) \approx \frac{n_{C_k}(x_1, \dots, x_d)}{N_{C_k}}$$

1D histogram: $n_{C_k}(x_1)$

2D histogram: $n_{C_k}(x_1, x_2)$

3D cube of numbers: $n_{C_k}(x_1, x_2, x_3)$



More features to improve classification accuracy!?

$$p(x | C_k) \approx \frac{n_{C_k}(x_1, \dots, x_d)}{N_{C_k}}$$

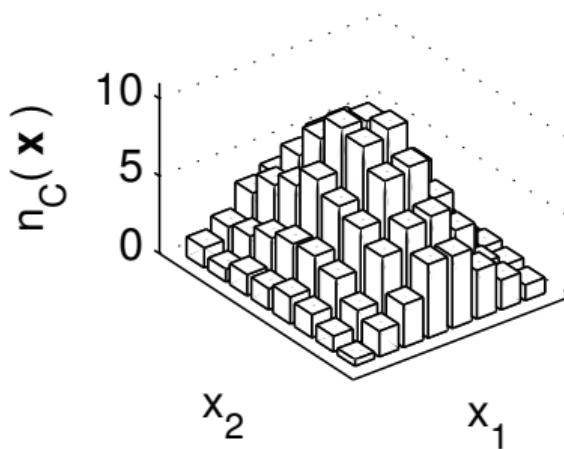
1D histogram: $n_{C_k}(x_1)$

2D histogram: $n_{C_k}(x_1, x_2)$

3D cube of numbers: $n_{C_k}(x_1, x_2, x_3)$

⋮

100 binary variables, 2^{100} settings (the universe is $\approx 2^{98}$ picoseconds old)



More features to improve classification accuracy!?

$$p(x | C_k) \approx \frac{n_{C_k}(x_1, \dots, x_d)}{N_{C_k}}$$

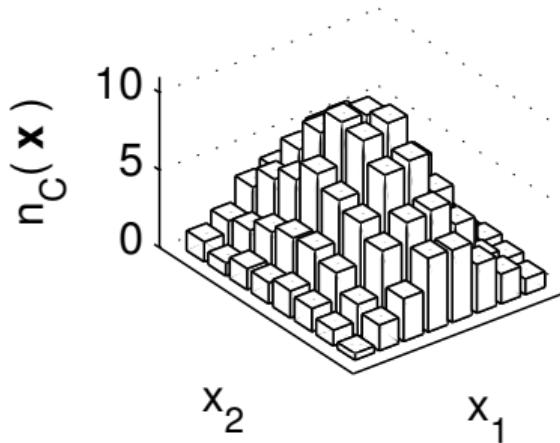
1D histogram: $n_{C_k}(x_1)$

2D histogram: $n_{C_k}(x_1, x_2)$

3D cube of numbers: $n_{C_k}(x_1, x_2, x_3)$

⋮

100 binary variables, 2^{100} settings (the universe is $\approx 2^{98}$ picoseconds old)



In high dimensions almost all $n_{C_k}(x_1, \dots, x_d)$ are zero

More features to improve classification accuracy!?

$$p(x | C_k) \approx \frac{n_{C_k}(x_1, \dots, x_d)}{N_{C_k}}$$

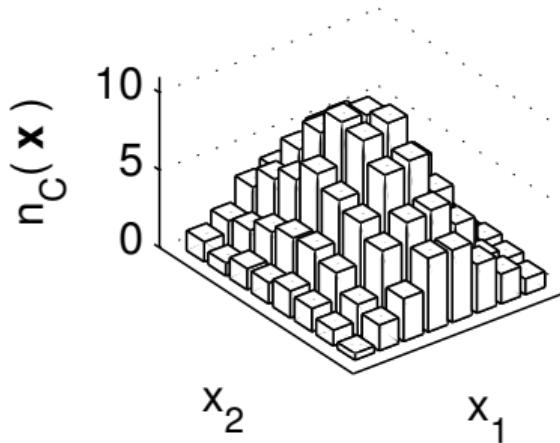
1D histogram: $n_{C_k}(x_1)$

2D histogram: $n_{C_k}(x_1, x_2)$

3D cube of numbers: $n_{C_k}(x_1, x_2, x_3)$

⋮

100 binary variables, 2^{100} settings (the universe is $\approx 2^{98}$ picoseconds old)



In high dimensions almost all $n_{C_k}(x_1, \dots, x_d)$ are zero

⇒ Bellman's “curse of dimensionality”

Avoiding the Curse of Dimensionality

Apply the chain rule?

Avoiding the Curse of Dimensionality

Apply the chain rule?

$$\begin{aligned} p(\mathbf{x} | C_k) &= p(x_1, x_2, \dots, x_d | C_k) \\ &= p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, x_1, C_k) p(x_4 | x_3, x_2, x_1, C_k) \cdots \\ &\quad \cdots p(x_{d-1} | x_{d-2}, \dots, x_1, C_k) p(x_d | x_{d-1}, \dots, x_1, C_k) \end{aligned}$$

Avoiding the Curse of Dimensionality

Apply the chain rule?

$$\begin{aligned} p(\mathbf{x} | C_k) &= p(x_1, x_2, \dots, x_d | C_k) \\ &= p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, x_1, C_k) p(x_4 | x_3, x_2, x_1, C_k) \cdots \\ &\quad \cdots p(x_{d-1} | x_{d-2}, \dots, x_1, C_k) p(x_d | x_{d-1}, \dots, x_1, C_k) \end{aligned}$$

Solution: assume structure in $p(\mathbf{x} | C_k)$

Avoiding the Curse of Dimensionality

Apply the chain rule?

$$\begin{aligned} p(\mathbf{x} | C_k) &= p(x_1, x_2, \dots, x_d | C_k) \\ &= p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, x_1, C_k) p(x_4 | x_3, x_2, x_1, C_k) \cdots \\ &\quad \cdots p(x_{d-1} | x_{d-2}, \dots, x_1, C_k) p(x_d | x_{d-1}, \dots, x_1, C_k) \end{aligned}$$

Solution: assume structure in $p(\mathbf{x} | C_k)$

For example,

- Assume x_{i+1} depends on x_i only

$$p(\mathbf{x} | C_k) \approx p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, C_k) \cdots p(x_d | x_{d-1}, C_k)$$

Avoiding the Curse of Dimensionality

Apply the chain rule?

$$\begin{aligned} p(\mathbf{x} | C_k) &= p(x_1, x_2, \dots, x_d | C_k) \\ &= p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, x_1, C_k) p(x_4 | x_3, x_2, x_1, C_k) \cdots \\ &\quad \cdots p(x_{d-1} | x_{d-2}, \dots, x_1, C_k) p(x_d | x_{d-1}, \dots, x_1, C_k) \end{aligned}$$

Solution: assume structure in $p(\mathbf{x} | C_k)$

For example,

- Assume x_{i+1} depends on x_i only

$$p(\mathbf{x} | C_k) \approx p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, C_k) \cdots p(x_d | x_{d-1}, C_k)$$

- Assume $\mathbf{x} \in \mathbb{R}^d$ distributes in a low dimensional vector space

Avoiding the Curse of Dimensionality

Apply the chain rule?

$$\begin{aligned} p(\mathbf{x} | C_k) &= p(x_1, x_2, \dots, x_d | C_k) \\ &= p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, x_1, C_k) p(x_4 | x_3, x_2, x_1, C_k) \cdots \\ &\quad \cdots p(x_{d-1} | x_{d-2}, \dots, x_1, C_k) p(x_d | x_{d-1}, \dots, x_1, C_k) \end{aligned}$$

Solution: assume structure in $p(\mathbf{x} | C_k)$

For example,

- Assume x_{i+1} depends on x_i only

$$p(\mathbf{x} | C_k) \approx p(x_1 | C_k) p(x_2 | x_1, C_k) p(x_3 | x_2, C_k) \cdots p(x_d | x_{d-1}, C_k)$$

- Assume $\mathbf{x} \in \mathbb{R}^d$ distributes in a low dimensional vector space
 - Dimensionality reduction by PCA (Principal Component Analysis) / KL-transform

Avoiding the Curse of Dimensionality (cont.)

- Apply smoothing windows (e.g., Parzen windows)
- Assume x follows a probability distribution (e.g., Normal dist.)
- Assume x_1, \dots, x_d are conditionally independent given class

Avoiding the Curse of Dimensionality (cont.)

- Apply smoothing windows (e.g., Parzen windows)
- Assume x follows a probability distribution (e.g., Normal dist.)
- Assume x_1, \dots, x_d are **conditionally independent** given class
⇒ **Naive Bayes** rule/model/assumption or *idiot Bayes rule*

$$\begin{aligned} p(x_1, x_2, \dots, x_d | C_k) &= p(x_1 | C_k) p(x_2 | C_k) \cdots p(x_d | C_k) \\ &= \prod_{d'=1}^d p(x_{d'} | C_k) \end{aligned}$$

Avoiding the Curse of Dimensionality (cont.)

- Apply smoothing windows (e.g., Parzen windows)
- Assume x follows a probability distribution (e.g., Normal dist.)
- Assume x_1, \dots, x_d are **conditionally independent** given class
⇒ **Naive Bayes** rule/model/assumption or *idiot Bayes rule*

$$\begin{aligned} p(x_1, x_2, \dots, x_d | C_k) &= p(x_1 | C_k) p(x_2 | C_k) \cdots p(x_d | C_k) \\ &= \prod_{d'=1}^d p(x_{d'} | C_k) \end{aligned}$$

- *Is it reasonable?*

Avoiding the Curse of Dimensionality (cont.)

- Apply smoothing windows (e.g., Parzen windows)
- Assume x follows a probability distribution (e.g., Normal dist.)
- Assume x_1, \dots, x_d are **conditionally independent** given class
⇒ **Naive Bayes** rule/model/assumption or *idiot Bayes rule*

$$\begin{aligned} p(x_1, x_2, \dots, x_d | C_k) &= p(x_1 | C_k) p(x_2 | C_k) \cdots p(x_d | C_k) \\ &= \prod_{d'=1}^d p(x_{d'} | C_k) \end{aligned}$$

- *Is it reasonable?*
Often not, of course!
Although it can still be *useful*.

Gaussian discriminant analysis

Consider a generative classifier where the class conditional densities are given as multivariate Gaussians:

$$p(\mathbf{x} | C_k; \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (3)$$

$$= \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right) \quad (4)$$

where $\boldsymbol{\mu}_k$ is the mean vector and $\boldsymbol{\Sigma}_k$ is the covariance matrix for class C_k .
The posterior:

$$p(C_k | \mathbf{x}) \propto p(C_k) \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

This classifier is called *Gaussian discriminant analysis* or *GDA*.

Multivariate Gaussian distribution (recap)

- The d -dimensional vector $\mathbf{x} = [x_1 \cdots x_d]^\top$ is multivariate Gaussian if it has a probability density function of the following form:

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

The pdf is parameterised by the mean vector $\boldsymbol{\mu} = [\mu_1 \cdots \mu_d]^\top$ and the covariance matrix $\boldsymbol{\Sigma} = (\sigma_{ij})$.

Multivariate Gaussian distribution (recap)

- The d -dimensional vector $\mathbf{x} = [x_1 \cdots x_d]^\top$ is multivariate Gaussian if it has a probability density function of the following form:

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

The pdf is parameterised by the mean vector $\boldsymbol{\mu} = [\mu_1 \cdots \mu_d]^\top$ and the covariance matrix $\boldsymbol{\Sigma} = (\sigma_{ij})$.

- The 1-dimensional Gaussian is a special case of this pdf

Multivariate Gaussian distribution (recap)

- The d -dimensional vector $\mathbf{x} = [x_1 \cdots x_d]^\top$ is multivariate Gaussian if it has a probability density function of the following form:

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

The pdf is parameterised by the mean vector $\boldsymbol{\mu} = [\mu_1 \cdots \mu_d]^\top$ and the covariance matrix $\boldsymbol{\Sigma} = (\sigma_{ij})$.

- The 1-dimensional Gaussian is a special case of this pdf
- The argument to the exponential $\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})$ is referred to as a *quadratic form*.

The parameters of a Gaussian distribution (recap)

- The mean vector μ is the expectation of \mathbf{x} :

$$\mu = E[\mathbf{x}]$$

The parameters of a Gaussian distribution (recap)

- The mean vector μ is the expectation of \mathbf{x} :

$$\mu = E[\mathbf{x}]$$

- The covariance matrix Σ is the expectation of the deviation of \mathbf{x} from the mean:

$$\Sigma = E[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^\top]$$

The parameters of a Gaussian distribution (recap)

- The mean vector μ is the expectation of \mathbf{x} :

$$\mu = E[\mathbf{x}]$$

- The covariance matrix Σ is the expectation of the deviation of \mathbf{x} from the mean:

$$\Sigma = E[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^\top]$$

- Σ is a $d \times d$ symmetric matrix: $\Sigma^\top = \Sigma$

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)] = E[(x_j - \mu_j)(x_i - \mu_i)] = \sigma_{ji} .$$

The parameters of a Gaussian distribution (recap)

- The mean vector μ is the expectation of \mathbf{x} :

$$\mu = E[\mathbf{x}]$$

- The covariance matrix Σ is the expectation of the deviation of \mathbf{x} from the mean:

$$\Sigma = E[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^\top]$$

- Σ is a $d \times d$ symmetric matrix: $\Sigma^\top = \Sigma$

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)] = E[(x_j - \mu_j)(x_i - \mu_i)] = \sigma_{ji}.$$

- The sign of the covariance σ_{ij} helps to determine the relationship between two components:

The parameters of a Gaussian distribution (recap)

- The mean vector μ is the expectation of \mathbf{x} :

$$\mu = E[\mathbf{x}]$$

- The covariance matrix Σ is the expectation of the deviation of \mathbf{x} from the mean:

$$\Sigma = E[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^\top]$$

- Σ is a $d \times d$ symmetric matrix: $\Sigma^\top = \Sigma$

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)] = E[(x_j - \mu_j)(x_i - \mu_i)] = \sigma_{ji}.$$

- The sign of the covariance σ_{ij} helps to determine the relationship between two components:
 - If x_j is large when x_i is large, then $(x_j - \mu_j)(x_i - \mu_i)$ will tend to be positive;

The parameters of a Gaussian distribution (recap)

- The mean vector μ is the expectation of \mathbf{x} :

$$\mu = E[\mathbf{x}]$$

- The covariance matrix Σ is the expectation of the deviation of \mathbf{x} from the mean:

$$\Sigma = E[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^\top]$$

- Σ is a $d \times d$ symmetric matrix: $\Sigma^\top = \Sigma$

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)] = E[(x_j - \mu_j)(x_i - \mu_i)] = \sigma_{ji}.$$

- The sign of the covariance σ_{ij} helps to determine the relationship between two components:
 - If x_j is large when x_i is large, then $(x_j - \mu_j)(x_i - \mu_i)$ will tend to be positive;
 - If x_j is small when x_i is large, then $(x_j - \mu_j)(x_i - \mu_i)$ will tend to be negative.

Covariance matrix (recap)

$$\Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \cdots & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \cdots & \cdots & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & & & \vdots \\ \vdots & \vdots & & \sigma_{ii} & & \vdots \\ \vdots & \vdots & & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \cdots & \cdots & \sigma_{dd} \end{pmatrix}$$

Covariance matrix (recap)

$$\Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \cdots & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \cdots & \cdots & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & & & \vdots \\ \vdots & \vdots & & \sigma_{ii} & & \vdots \\ \vdots & \vdots & & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \cdots & \cdots & \sigma_{dd} \end{pmatrix}$$

- $\sigma_i^2 = \sigma_{ii}$

Covariance matrix (recap)

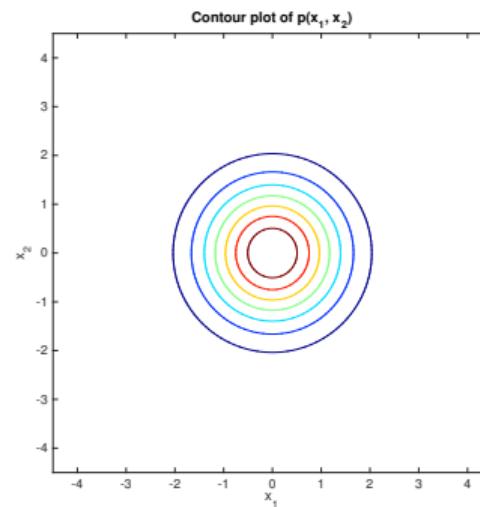
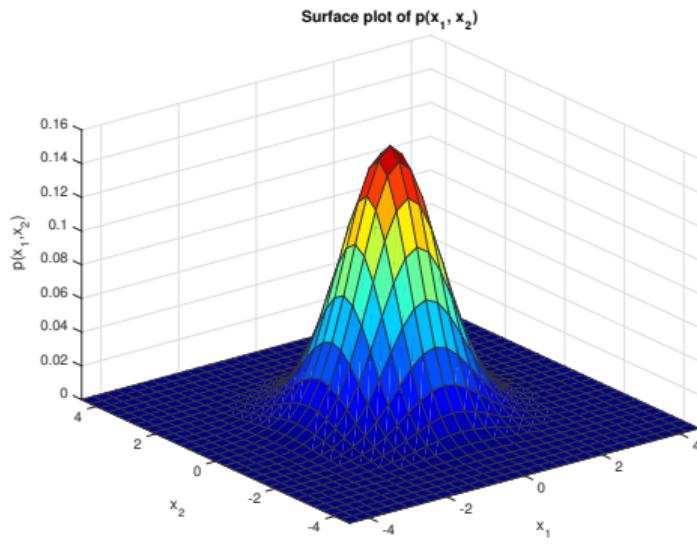
$$\Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \cdots & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \cdots & \cdots & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & & & \vdots \\ \vdots & \vdots & & \sigma_{ii} & & \vdots \\ \vdots & \vdots & & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \cdots & \cdots & \sigma_{dd} \end{pmatrix}$$

- $\sigma_i^2 = \sigma_{ii}$
- $|\Sigma| = \det(\Sigma)$: determinant

e.g., for $d = 2$,

$$|\Sigma| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = a \times d - b \times c$$

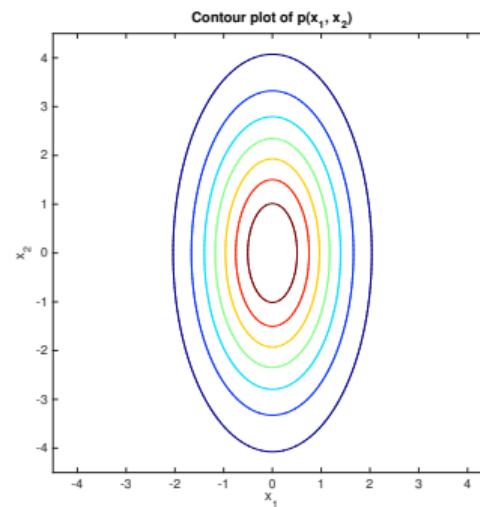
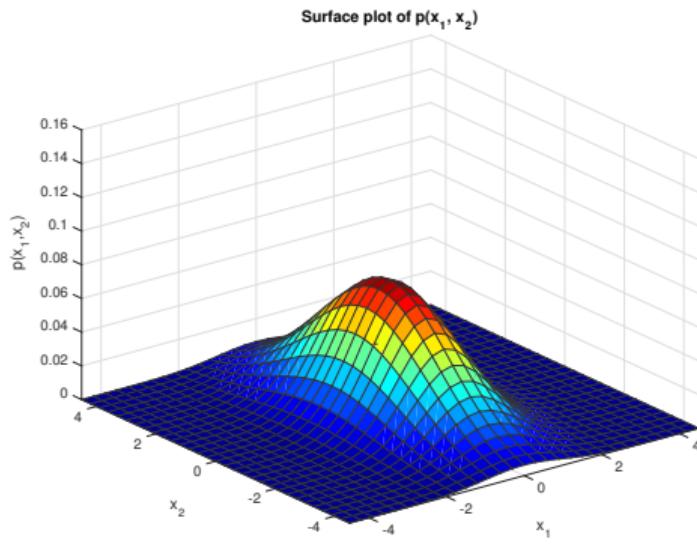
Spherical Gaussian



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \rho_{12} = 0$$

NB: Correlation coefficient $\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}$ $(-1 \leq \rho_{ij} \leq 1)$

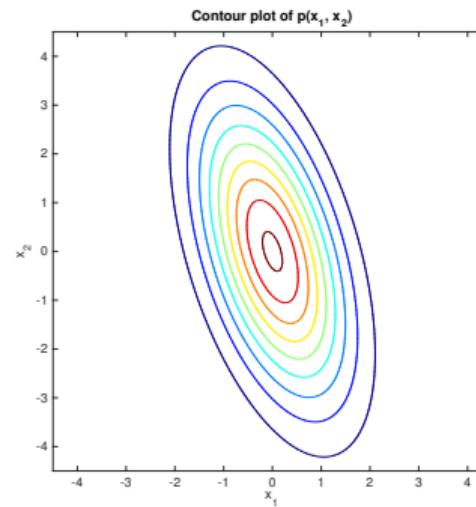
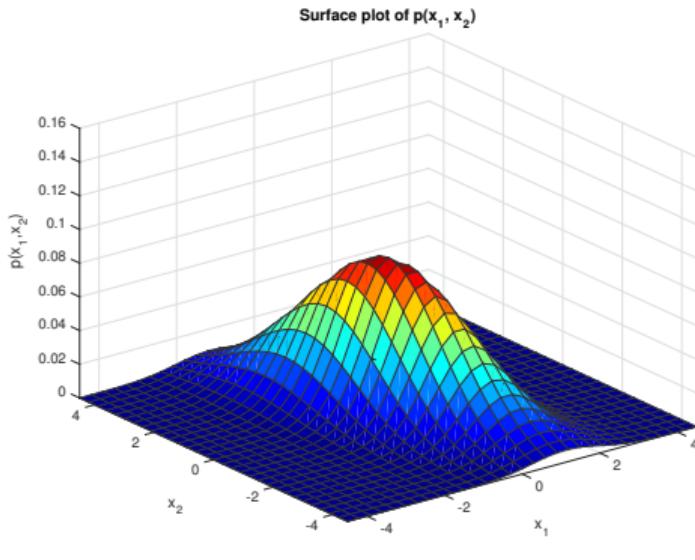
2-D Gaussian with a diagonal covariance matrix



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix} \quad \rho_{12} = 0$$

NB: Correlation coefficient $\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}$ $(-1 \leq \rho_{ij} \leq 1)$

2-D Gaussian with a full covariance matrix



$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 1 & -1 \\ -1 & 4 \end{pmatrix} \quad \rho_{12} = -0.5$$

NB: Correlation coefficient $\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}$ $(-1 \leq \rho_{ij} \leq 1)$

Training in GDA

- Training data set $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$
- Parameter estimation of each Gaussian distribution $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ for $k = 1, \dots, K$ based on the maximum likelihood estimation (MLE)
 - Mean vector:

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{\{y_n \mid y_n=k\}} \mathbf{x}_n \quad \text{where } N_k = |\{y \mid y=k\}|$$

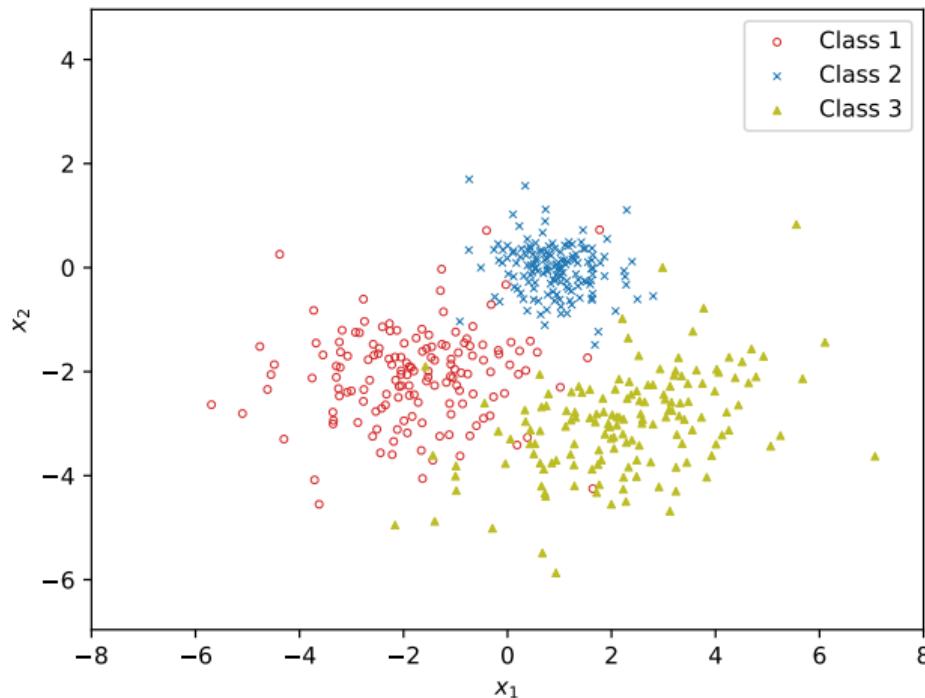
- Covariance matrix:

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{\{y_n \mid y_n=k\}} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top$$

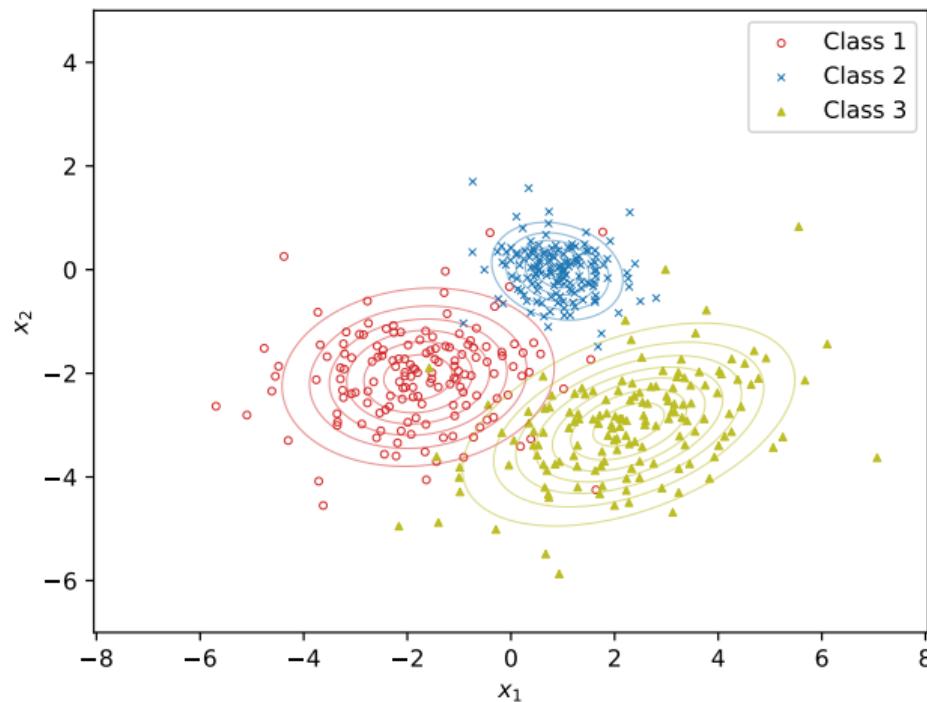
$$(\boldsymbol{\Sigma}_k)_{ij} = \frac{1}{N_k} \sum_{\{y_n \mid y_n=k\}} (x_{ni} - \mu_{ki})(x_{nj} - \mu_{kj})$$

- Estimation of the class prior probability $p(C_k)$ based on the data or knowledge/assumption

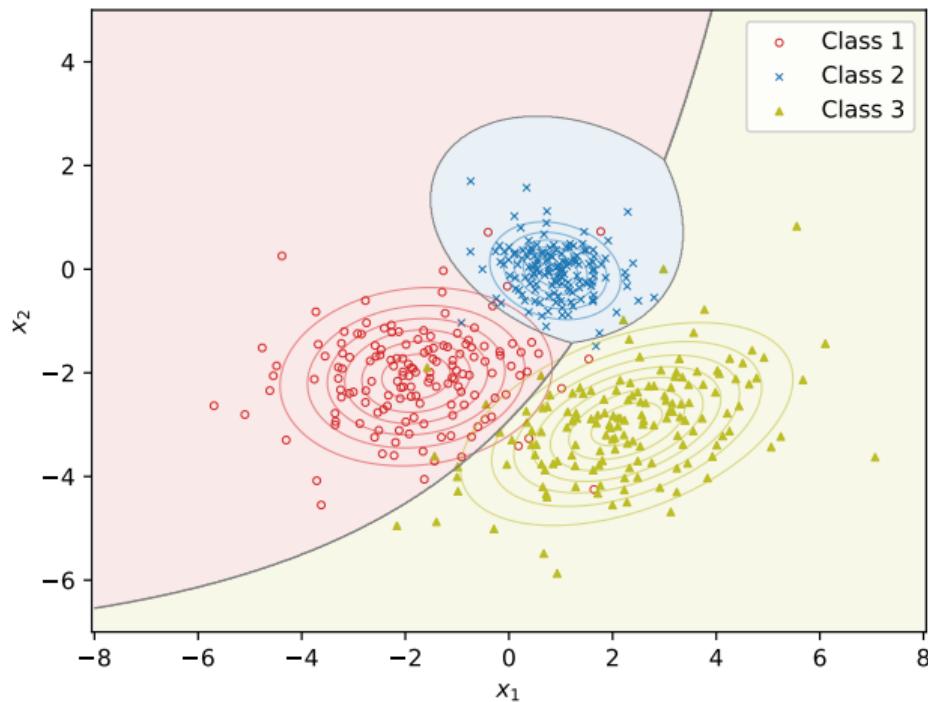
Example of training data



Gaussians estimated from the data



Decision boundaries and regions with the GDA



Decision regions

- Recall Bayes' Rule:

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

Decision regions

- Recall Bayes' Rule:

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

- Given an unseen point x , we assign to the class for which $p(C_k|x)$ is largest.
($k^* = \arg \max_k p(C_k|x)$)

Decision regions

- Recall Bayes' Rule:

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

- Given an unseen point x , we assign to the class for which $p(C_k|x)$ is largest.
($k^* = \arg \max_k p(C_k|x)$)
- Thus x -space (the input space) may be regarded as being divided into decision regions \mathcal{R}_k such that a point falling in \mathcal{R}_k is assigned to class C_k .

Decision regions

- Recall Bayes' Rule:

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

- Given an unseen point x , we assign to the class for which $p(C_k|x)$ is largest.
($k^* = \arg \max_k p(C_k|x)$)
- Thus x -space (the input space) may be regarded as being divided into decision regions \mathcal{R}_k such that a point falling in \mathcal{R}_k is assigned to class C_k .
- *Decision region* \mathcal{R}_k needs not be contiguous, but may consist of several disjoint regions each associated with class C_k .

Decision regions

- Recall Bayes' Rule:

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

- Given an unseen point x , we assign to the class for which $p(C_k|x)$ is largest.
($k^* = \arg \max_k p(C_k|x)$)
- Thus x -space (the input space) may be regarded as being divided into decision regions \mathcal{R}_k such that a point falling in \mathcal{R}_k is assigned to class C_k .
- *Decision region* \mathcal{R}_k needs not be contiguous, but may consist of several disjoint regions each associated with class C_k .
- The boundaries between these regions are called *decision boundaries*.

Placement of decision boundaries

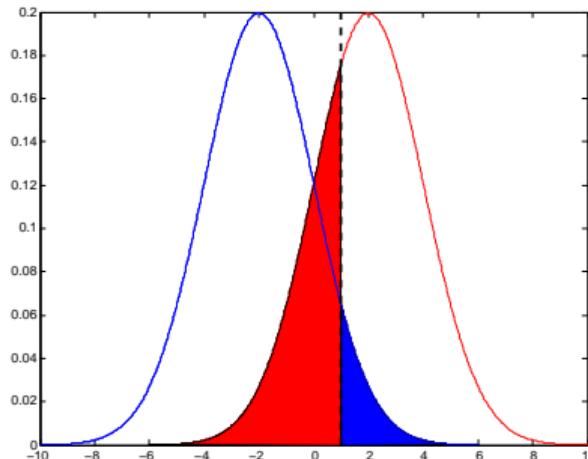
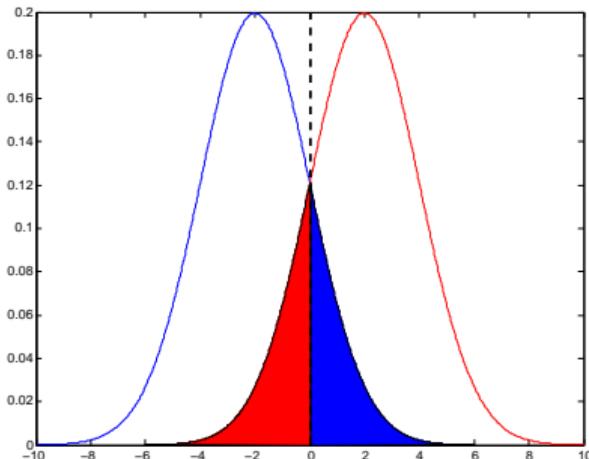
- Consider a 1-dimensional feature space (x) and two classes C_1 and C_2 .

Placement of decision boundaries

- Consider a 1-dimensional feature space (x) and two classes C_1 and C_2 .
- How to place the decision boundary to minimise the probability of misclassification (based on $p(x, C_k)$)?

Placement of decision boundaries

- Consider a 1-dimensional feature space (x) and two classes C_1 and C_2 .
- How to place the decision boundary to minimise the probability of misclassification (based on $p(x, C_k)$)?



Decision regions and misclassification

Confusion matrix

In\Out	C_1	C_2
C_1	N_{11}	N_{12}
C_2	N_{21}	N_{22}

Normalised confusion matrix

In\Out	C_1	C_2
C_1	P_{11}	P_{12}
C_2	P_{21}	P_{22}

\Rightarrow

$$\begin{aligned} P_{11} + P_{12} &= 1 \\ P_{21} + P_{22} &= 1 \end{aligned}$$

Decision regions and misclassification

Confusion matrix

In\Out	C_1	C_2
C_1	N_{11}	N_{12}
C_2	N_{21}	N_{22}

Normalised confusion matrix

In\Out	C_1	C_2
C_1	P_{11}	P_{12}
C_2	P_{21}	P_{22}

\Rightarrow

$$P_{11} + P_{12} = 1$$

$$P_{21} + P_{22} = 1$$

$$P_{11} = p(x \in \mathcal{R}_1 | C_1) = \frac{N_{11}}{N_1}, \quad P_{12} = p(x \in \mathcal{R}_2 | C_1) = \frac{N_{12}}{N_1}$$

$$P_{21} = p(x \in \mathcal{R}_1 | C_2) = \frac{N_{21}}{N_2}, \quad P_{22} = p(x \in \mathcal{R}_2 | C_2) = \frac{N_{22}}{N_2}$$

$$N_1 = N_{11} + N_{12}, \quad N_2 = N_{21} + N_{22}, \quad p(C_1) = \frac{N_1}{N_1 + N_2}, \quad p(C_2) = \frac{N_2}{N_1 + N_2}$$

Decision regions and misclassification

Confusion matrix

In\Out	C_1	C_2
C_1	N_{11}	N_{12}
C_2	N_{21}	N_{22}

Normalised confusion matrix

In\Out	C_1	C_2
C_1	P_{11}	P_{12}
C_2	P_{21}	P_{22}

\Rightarrow

$$P_{11} + P_{12} = 1$$

$$P_{21} + P_{22} = 1$$

$$P_{11} = p(x \in \mathcal{R}_1 | C_1) = \frac{N_{11}}{N_1}, \quad P_{12} = p(x \in \mathcal{R}_2 | C_1) = \frac{N_{12}}{N_1}$$

$$P_{21} = p(x \in \mathcal{R}_1 | C_2) = \frac{N_{21}}{N_2}, \quad P_{22} = p(x \in \mathcal{R}_2 | C_2) = \frac{N_{22}}{N_2}$$

$$p(\text{correct}) = \frac{N_{11} + N_{22}}{N_1 + N_2} = P_{11} p(C_1) + P_{22} p(C_2)$$
$$N_1 = N_{11} + N_{12}, \quad N_2 = N_{21} + N_{22}, \quad p(C_1) = \frac{N_1}{N_1 + N_2}, \quad p(C_2) = \frac{N_2}{N_1 + N_2}$$

Decision regions and misclassification

Confusion matrix

In\Out	C_1	C_2
C_1	N_{11}	N_{12}
C_2	N_{21}	N_{22}

Normalised confusion matrix

In\Out	C_1	C_2
C_1	P_{11}	P_{12}
C_2	P_{21}	P_{22}

\Rightarrow

$$P_{11} + P_{12} = 1$$

$$P_{21} + P_{22} = 1$$

$$P_{11} = p(x \in \mathcal{R}_1 | C_1) = \frac{N_{11}}{N_1}, \quad P_{12} = p(x \in \mathcal{R}_2 | C_1) = \frac{N_{12}}{N_1}$$

$$P_{21} = p(x \in \mathcal{R}_1 | C_2) = \frac{N_{21}}{N_2}, \quad P_{22} = p(x \in \mathcal{R}_2 | C_2) = \frac{N_{22}}{N_2}$$

$$p(\text{correct}) = \frac{N_{11} + N_{22}}{N_1 + N_2} = P_{11} p(C_1) + P_{22} p(C_2)$$
$$N_1 = N_{11} + N_{12}, \quad N_2 = N_{21} + N_{22}, \quad p(C_1) = \frac{N_1}{N_1 + N_2}, \quad p(C_2) = \frac{N_2}{N_1 + N_2}$$

$$p(\text{error}) = \frac{N_{12} + N_{21}}{N_1 + N_2} = P_{12} p(C_1) + P_{21} p(C_2)$$

Decision regions and misclassification

Confusion matrix

In\Out	C_1	C_2
C_1	N_{11}	N_{12}
C_2	N_{21}	N_{22}

Normalised confusion matrix

In\Out	C_1	C_2
C_1	P_{11}	P_{12}
C_2	P_{21}	P_{22}

\Rightarrow

$$P_{11} + P_{12} = 1$$

$$P_{21} + P_{22} = 1$$

$$P_{11} = p(x \in \mathcal{R}_1 | C_1) = \frac{N_{11}}{N_1}, \quad P_{12} = p(x \in \mathcal{R}_2 | C_1) = \frac{N_{12}}{N_1}$$

$$P_{21} = p(x \in \mathcal{R}_1 | C_2) = \frac{N_{21}}{N_2}, \quad P_{22} = p(x \in \mathcal{R}_2 | C_2) = \frac{N_{22}}{N_2}$$

$$p(\text{correct}) = \frac{N_{11} + N_{22}}{N_1 + N_2} = P_{11} p(C_1) + P_{22} p(C_2)$$
$$N_1 = N_{11} + N_{12}, \quad N_2 = N_{21} + N_{22}, \quad p(C_1) = \frac{N_1}{N_1 + N_2}, \quad p(C_2) = \frac{N_2}{N_1 + N_2}$$

$$p(\text{error}) = \frac{N_{12} + N_{21}}{N_1 + N_2} = P_{12} p(C_1) + P_{21} p(C_2)$$

$$= \int_{\mathcal{R}_2} p(x | C_1) p(C_1) dx + \int_{\mathcal{R}_1} p(x | C_2) p(C_2) dx$$

Decision regions and misclassification

Confusion matrix

In\Out	C_1	C_2
C_1	N_{11}	N_{12}
C_2	N_{21}	N_{22}

\Rightarrow

Normalised confusion matrix

In\Out	C_1	C_2
C_1	P_{11}	P_{12}
C_2	P_{21}	P_{22}

$$P_{11} + P_{12} = 1$$

$$P_{21} + P_{22} = 1$$

$$P_{11} = p(x \in \mathcal{R}_1 | C_1) = \frac{N_{11}}{N_1}, \quad P_{12} = p(x \in \mathcal{R}_2 | C_1) = \frac{N_{12}}{N_1}$$

$$P_{21} = p(x \in \mathcal{R}_1 | C_2) = \frac{N_{21}}{N_2}, \quad P_{22} = p(x \in \mathcal{R}_2 | C_2) = \frac{N_{22}}{N_2}$$

$$p(\text{correct}) = \frac{N_{11} + N_{22}}{N_1 + N_2} = P_{11} p(C_1) + P_{22} p(C_2)$$

$$N_1 = N_{11} + N_{12}, \quad N_2 = N_{21} + N_{22}, \quad p(C_1) = \frac{N_1}{N_1 + N_2}, \quad p(C_2) = \frac{N_2}{N_1 + N_2}$$

$$p(\text{error}) = \frac{N_{12} + N_{21}}{N_1 + N_2} = P_{12} p(C_1) + P_{21} p(C_2)$$

$$= \int_{\mathcal{R}_2} p(x | C_1) p(C_1) dx + \int_{\mathcal{R}_1} p(x | C_2) p(C_2) dx$$

$$= \int_{\mathcal{R}_2} p(C_1 | x) p(x) dx + \int_{\mathcal{R}_1} p(C_2 | x) p(x) dx$$

Minimising probability of misclassification

$$p(\text{error} | \mathcal{R}_1, \mathcal{R}_2) = \int_{\mathcal{R}_2} p(C_1 | x) p(x) dx + \int_{\mathcal{R}_1} p(C_2 | x) p(x) dx \quad (5)$$

Minimising probability of misclassification

$$p(\text{error} | \mathcal{R}_1, \mathcal{R}_2) = \int_{\mathcal{R}_2} p(C_1 | x) p(x) dx + \int_{\mathcal{R}_1} p(C_2 | x) p(x) dx \quad (5)$$

- If $\hat{x} = x_0 \in \mathcal{R}_2$ such that $p(C_1 | x_0) > p(C_2 | x_0)$,

Minimising probability of misclassification

$$p(\text{error} | \mathcal{R}_1, \mathcal{R}_2) = \int_{\mathcal{R}_2} p(C_1 | x) p(x) dx + \int_{\mathcal{R}_1} p(C_2 | x) p(x) dx \quad (5)$$

- If $\hat{x} = x_0 \in \mathcal{R}_2$ such that $p(C_1 | x_0) > p(C_2 | x_0)$,
letting $\mathcal{R}_2^* = \mathcal{R}_2 - \{x_0\}$ and $\mathcal{R}_1^* = \mathcal{R}_1 + \{x_0\}$ gives

Minimising probability of misclassification

$$p(\text{error} | \mathcal{R}_1, \mathcal{R}_2) = \int_{\mathcal{R}_2} p(C_1 | x) p(x) dx + \int_{\mathcal{R}_1} p(C_2 | x) p(x) dx \quad (5)$$

- If $\hat{x} = x_0 \in \mathcal{R}_2$ such that $p(C_1 | x_0) > p(C_2 | x_0)$,
letting $\mathcal{R}_2^* = \mathcal{R}_2 - \{x_0\}$ and $\mathcal{R}_1^* = \mathcal{R}_1 + \{x_0\}$ gives

$$p(\text{error} | \mathcal{R}_1^*, \mathcal{R}_2^*) < p(\text{error} | \mathcal{R}_1, \mathcal{R}_2)$$

Minimising probability of misclassification

$$p(\text{error}|\mathcal{R}_1, \mathcal{R}_2) = \int_{\mathcal{R}_2} p(C_1|x) p(x) dx + \int_{\mathcal{R}_1} p(C_2|x) p(x) dx \quad (5)$$

- If $\hat{x} = x_0 \in \mathcal{R}_2$ such that $p(C_1|x_0) > p(C_2|x_0)$,
letting $\mathcal{R}_2^* = \mathcal{R}_2 - \{x_0\}$ and $\mathcal{R}_1^* = \mathcal{R}_1 + \{x_0\}$ gives

$$p(\text{error}|\mathcal{R}_1^*, \mathcal{R}_2^*) < p(\text{error}|\mathcal{R}_1, \mathcal{R}_2)$$

- $p(\text{error})$ is minimised by assigning each point to the class with the maximum posterior probability – Bayes decision rule / MAP decision rule / minimum error rate classification.

Minimising probability of misclassification

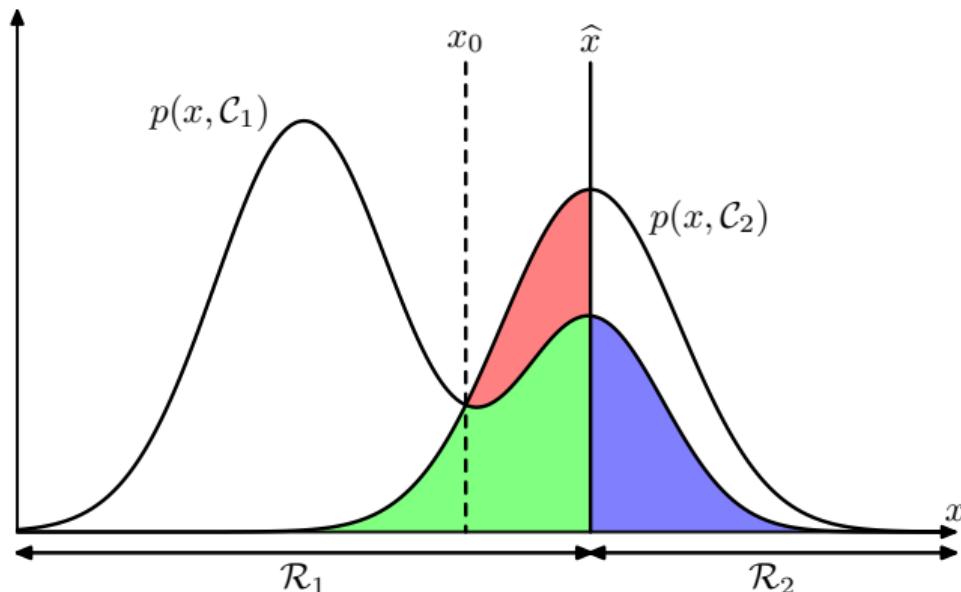
$$p(\text{error} | \mathcal{R}_1, \mathcal{R}_2) = \int_{\mathcal{R}_2} p(C_1 | x) p(x) dx + \int_{\mathcal{R}_1} p(C_2 | x) p(x) dx \quad (5)$$

- If $\hat{x} = x_0 \in \mathcal{R}_2$ such that $p(C_1 | x_0) > p(C_2 | x_0)$,
letting $\mathcal{R}_2^* = \mathcal{R}_2 - \{x_0\}$ and $\mathcal{R}_1^* = \mathcal{R}_1 + \{x_0\}$ gives

$$p(\text{error} | \mathcal{R}_1^*, \mathcal{R}_2^*) < p(\text{error} | \mathcal{R}_1, \mathcal{R}_2)$$

- $p(\text{error})$ is minimised by assigning each point to the class with the maximum posterior probability – Bayes decision rule / MAP decision rule / minimum error rate classification.
- This justification for the maximum posterior probability may be extended to d -dimensional feature vectors and K classes

Minimising probability of misclassification (cont.)



After Fig. 1.24, C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

\hat{x} denotes the current decision boundary, which causes error shown in red, green, and blue regions. The error is minimised by locating the boundary at x_o .

Should we always use the Bayes decision rule?

See “Predictions and Decision Boundaries”, LWLS 3.2.

Discriminant function of GDA

Recall GDA

$$p(C_k | \mathbf{x}, \boldsymbol{\theta}) \propto p(C_k) \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where

$$\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right) \quad (6)$$

Discriminant function of GDA

Recall GDA

$$p(C_k | \mathbf{x}, \boldsymbol{\theta}) \propto p(C_k) \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where

$$\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right) \quad (6)$$

- The *discriminant function* of GDA: (taking log and ignoring constant terms yields)

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2} \log |\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \quad (7)$$

⋯ quadratic function of \mathbf{x} .

Discriminant function of GDA

Recall GDA

$$p(C_k | \mathbf{x}, \boldsymbol{\theta}) \propto p(C_k) \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where

$$\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right) \quad (6)$$

- The *discriminant function* of GDA: (taking log and ignoring constant terms yields)

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2} \log |\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \quad (7)$$

⋯ quadratic function of \mathbf{x} .

- Classification (estimating the class label):

$$\hat{y}(\mathbf{x}) = \arg \max_k g_k(\mathbf{x}) \quad (8)$$

Discriminant function of GDA

Recall GDA

$$p(C_k | \mathbf{x}, \boldsymbol{\theta}) \propto p(C_k) \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where

$$\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right) \quad (6)$$

- The *discriminant function* of GDA: (taking log and ignoring constant terms yields)

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2} \log |\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \quad (7)$$

⋯ quadratic function of \mathbf{x} .

- Classification (estimating the class label):

$$\hat{y}(\mathbf{x}) = \arg \max_k g_k(\mathbf{x}) \quad (8)$$

- So, the decision boundaries are *piecewise quadratic* functions of \mathbf{x} . (Check!)

Special case of GDA – equal covariance

Assume all class covariances Σ_k share the same covariance, $\Sigma_k = \Sigma$.
The discriminant function is reduced to

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) \quad (9)$$

Special case of GDA – equal covariance

Assume all class covariances Σ_k share the same covariance, $\Sigma_k = \Sigma$.

The discriminant function is reduced to

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) \quad (9)$$

$$= \log p(C_k) + \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} \quad (10)$$

Special case of GDA – equal covariance

Assume all class covariances Σ_k share the same covariance, $\Sigma_k = \Sigma$.

The discriminant function is reduced to

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) \quad (9)$$

$$= \log p(C_k) + \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} \quad (10)$$

$$= \mathbf{w}_k^\top \mathbf{x} + w_{k0} + \text{const} \quad (11)$$

where $\mathbf{w}_k^\top = \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1}$ and $w_{k0} = -\frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \log p(C_k)$

Special case of GDA – equal covariance

Assume all class covariances Σ_k share the same covariance, $\Sigma_k = \Sigma$.

The discriminant function is reduced to

$$g_k(\mathbf{x}) = \log p(C_k) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) \quad (9)$$

$$= \log p(C_k) + \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} \quad (10)$$

$$= \mathbf{w}_k^\top \mathbf{x} + w_{k0} + \text{const} \quad (11)$$

$$\text{where } \mathbf{w}_k^\top = \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \text{ and } w_{k0} = -\frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \log p(C_k)$$

This is called a *linear discriminant function* as it is a linear function of \mathbf{x} .
The method is called *Linear Discriminant Analysis (LDA)*.

Special case of GDA – equal covariance (cont.)

Including the constant terms to w_{k0} , we have:

$$g_k(\mathbf{x}) = \mathbf{w}_k^\top \mathbf{x} + w_{k0} \quad (12)$$

Since $g_k(\mathbf{x}) = \log p(C_k) p(\mathbf{x} | C_k, \theta)$,

$$p(C_k | \mathbf{x}, \theta) = \frac{g_k(\mathbf{x})}{\sum_{k'=1}^K g_{k'}(\mathbf{x})} \quad (13)$$

$$= \frac{e^{\mathbf{w}_k^\top \mathbf{x} + w_{k0}}}{\sum_{k'=1}^K e^{\mathbf{w}_{k'}^\top \mathbf{x} + w_{k'0}}} \quad (14)$$

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \quad \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \quad \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

$$g_k(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\Sigma_k| + \ln p(C_k) \quad (15)$$

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

$$g_k(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\Sigma_k| + \ln p(C_k) \quad (15)$$

$$= -\frac{1}{2\sigma^2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top (\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln \sigma^{2d} + \ln p(C_k) \quad (16)$$

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

$$g_k(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\Sigma_k| + \ln p(C_k) \quad (15)$$

$$= -\frac{1}{2\sigma^2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln \sigma^{2d} + \ln p(C_k) \quad (16)$$

$$g_k(\mathbf{x}) = -\frac{1}{2\sigma^2} \|\mathbf{x} - \boldsymbol{\mu}_k\|^2 + \ln p(C_k) \quad (17)$$

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

$$g_k(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\Sigma_k| + \ln p(C_k) \quad (15)$$

$$= -\frac{1}{2\sigma^2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top (\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln \sigma^{2d} + \ln p(C_k) \quad (16)$$

$$g_k(\mathbf{x}) = -\frac{1}{2\sigma^2} \|\mathbf{x} - \boldsymbol{\mu}_k\|^2 + \ln p(C_k) \quad (17)$$

- If equal prior probabilities are assumed,

$$g_k(\mathbf{x}) = -\|\mathbf{x} - \boldsymbol{\mu}_k\|^2 \quad (18)$$

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

$$g_k(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\Sigma_k| + \ln p(C_k) \quad (15)$$

$$= -\frac{1}{2\sigma^2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top (\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln \sigma^{2d} + \ln p(C_k) \quad (16)$$

$$g_k(\mathbf{x}) = -\frac{1}{2\sigma^2} \|\mathbf{x} - \boldsymbol{\mu}_k\|^2 + \ln p(C_k) \quad (17)$$

- If equal prior probabilities are assumed,

$$g_k(\mathbf{x}) = -\|\mathbf{x} - \boldsymbol{\mu}_k\|^2 \quad (18)$$

The decision rule: “assign a test data to the class whose mean is closest”.

Another special case of GDA

- Spherical Gaussians: $\Sigma = \sigma^2 \mathbf{I} \Rightarrow |\Sigma| = \sigma^{2d}, \Sigma^{-1} = \frac{1}{\sigma^2} \mathbf{I}$
- Discriminant function:

$$g_k(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\Sigma_k| + \ln p(C_k) \quad (15)$$

$$= -\frac{1}{2\sigma^2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top(\mathbf{x} - \boldsymbol{\mu}_k) - \frac{1}{2} \ln \sigma^{2d} + \ln p(C_k) \quad (16)$$

$$g_k(\mathbf{x}) = -\frac{1}{2\sigma^2} \|\mathbf{x} - \boldsymbol{\mu}_k\|^2 + \ln p(C_k) \quad (17)$$

- If equal prior probabilities are assumed,

$$g_k(\mathbf{x}) = -\|\mathbf{x} - \boldsymbol{\mu}_k\|^2 \quad (18)$$

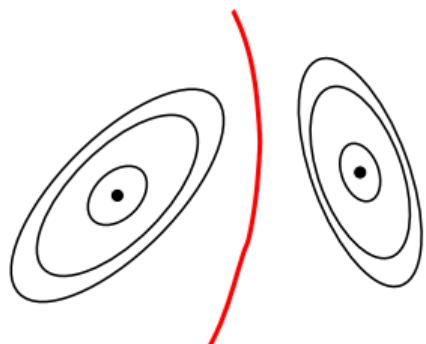
The decision rule: “assign a test data to the class whose mean is closest”.

The class means ($\boldsymbol{\mu}_k$) may be regarded as class **templates** or **prototypes**.

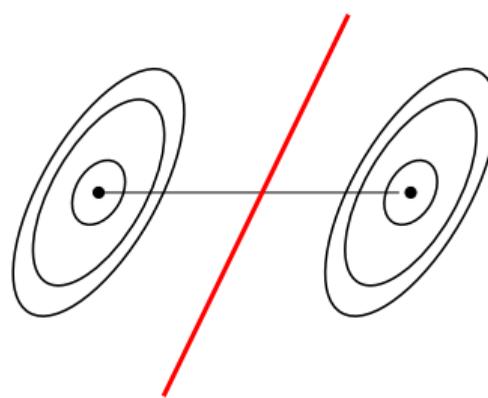
Decision boundaries of GDA

Considering a binary classification between C_1 and C_2 , the decision boundaries are defined as:

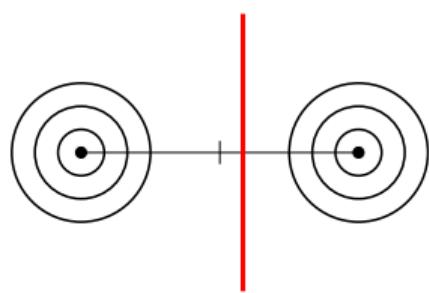
$$\left\{ \mathbf{x} \in \mathbb{R}^d \mid g_1(\mathbf{x}) - g_2(\mathbf{x}) = 0 \right\} \quad (19)$$



(a)



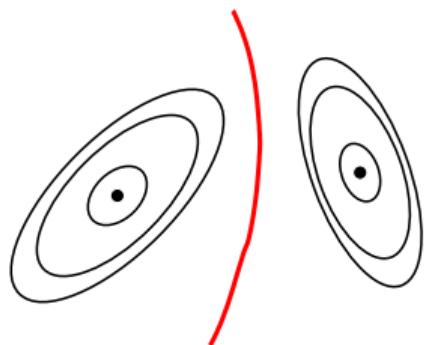
(b)



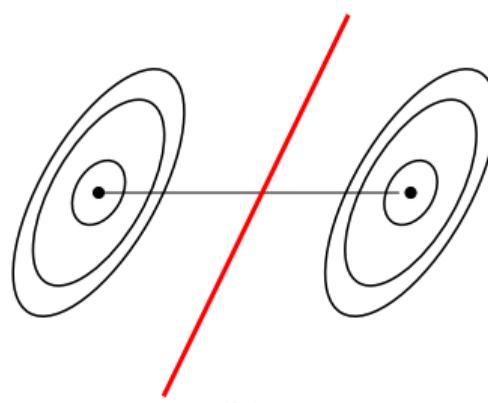
(c)

Which type of GDA should you use in practice?

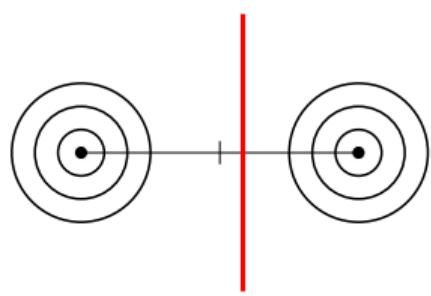
- GDA with general covariance matrices
- GDA with an equal (or tied or shared) covariance matrix, i.e., LDA
- GDA with an spherical (or isotropic) covariance matrix



(a)



(b)



(c)

Quizzes

- Show:

$$p(C_k | \mathbf{x}) = \frac{p(\mathbf{x} | C_k) p(C_k)}{\sum_{k'=1}^K p(\mathbf{x} | C_{k'}) p(C_{k'})}$$

- Write Python code that generates 2D and 3D visualisations of a two-dimensional Gaussian distribution with a specified mean vector and covariance matrix.
 - Run the code using various sets of parameters.
 - You will find that the code does not work with some covariance matrices. Describe the conditions for valid covariance matrices.

Quizzes (cont.)

- Show that the natural logarithm of a multivariate Gaussian distribution

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

is given as

$$-\frac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x} + \boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \frac{1}{2}\log |\boldsymbol{\Sigma}| - \frac{d}{2}\log 2\pi$$

Quizzes (cont.)

- Show that the decision boundary of GDA for binary classification is quadratic in \mathbf{x} , where $\mathbf{x} \in \mathbb{R}^d$.
- In GDA with an equal covariance, discuss how to estimate the shared covariance matrix from a given training set $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$.
- When deriving the discriminant function in Gaussian discriminant analysis (GDA), we used the natural logarithm. Discuss the validity of using the logarithm.