Machine Learning 1

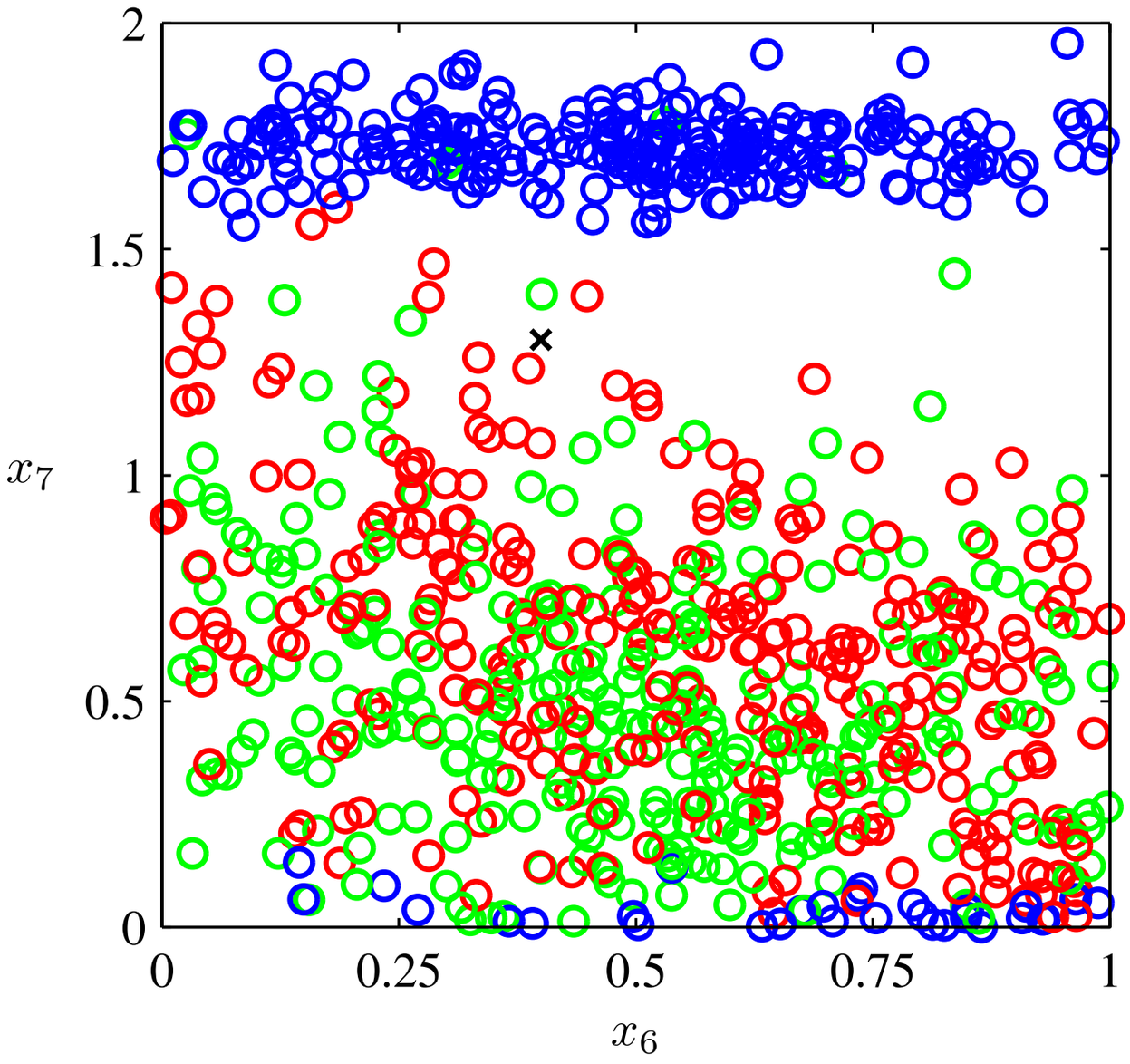
- Lecture 6 -
Linear Classification

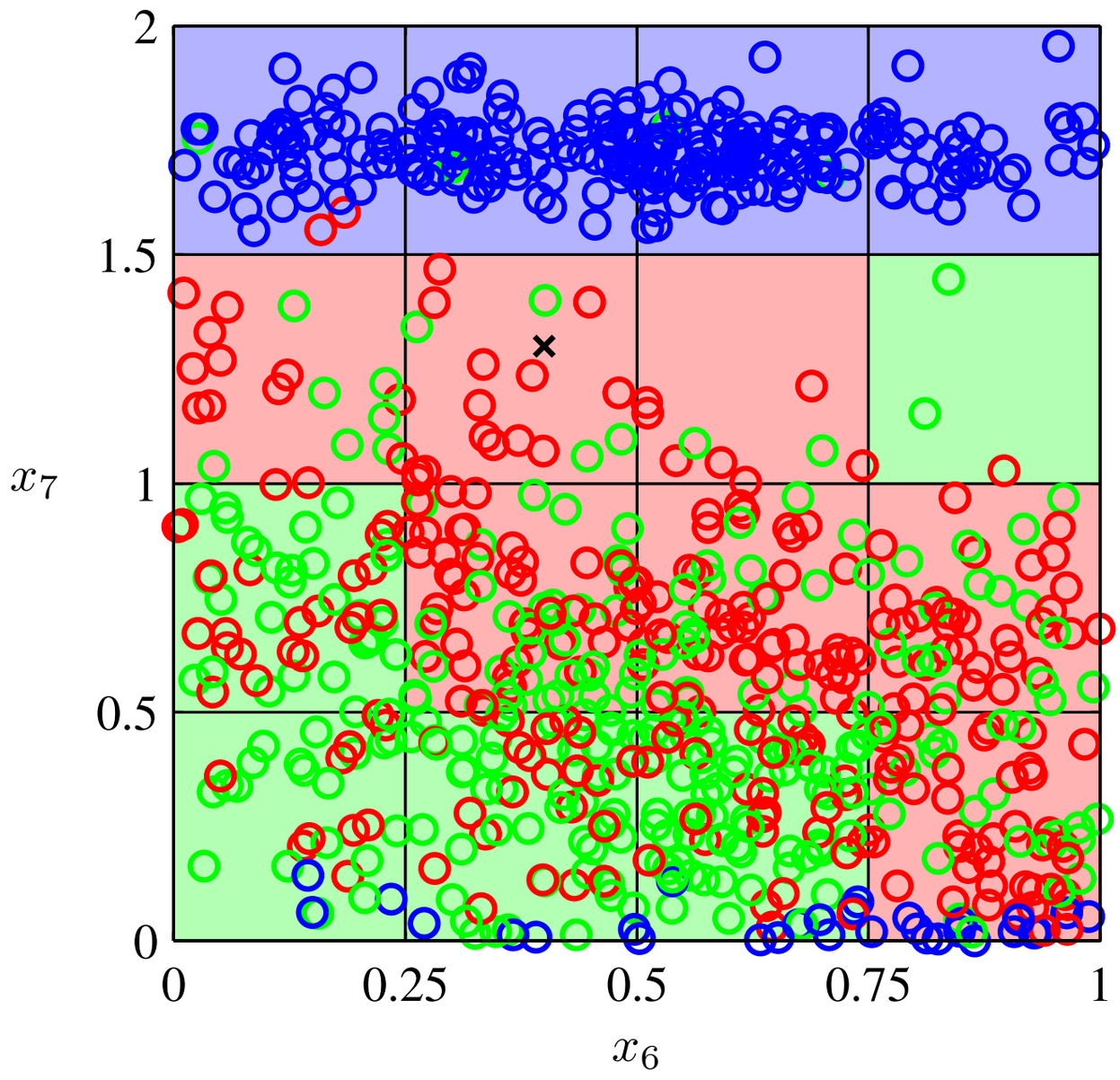
- Patrick Forré -

Slides created by:
Rianne van den Berg

Image credit: Kirillm | Getty Images

Classification through Decision Regions





Figures: 3 class problem with decision boundaries. (Bishop 1.19 & 1.20)

Linear Classification

- Linear Classification: consider only *linear* decision boundaries
- For D - dimensional input space:
  decision surface is a $(D-1)$ dimensional hyperplane

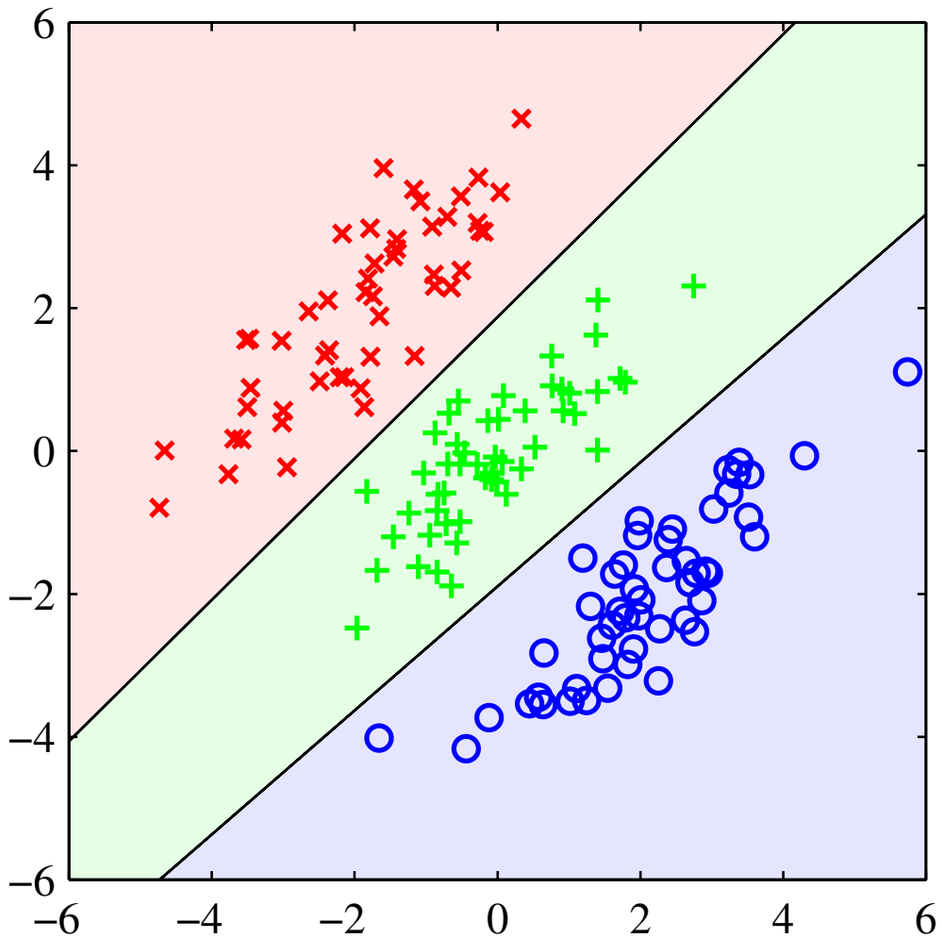


Figure: Linearly separable dataset (Bishop 4.5)

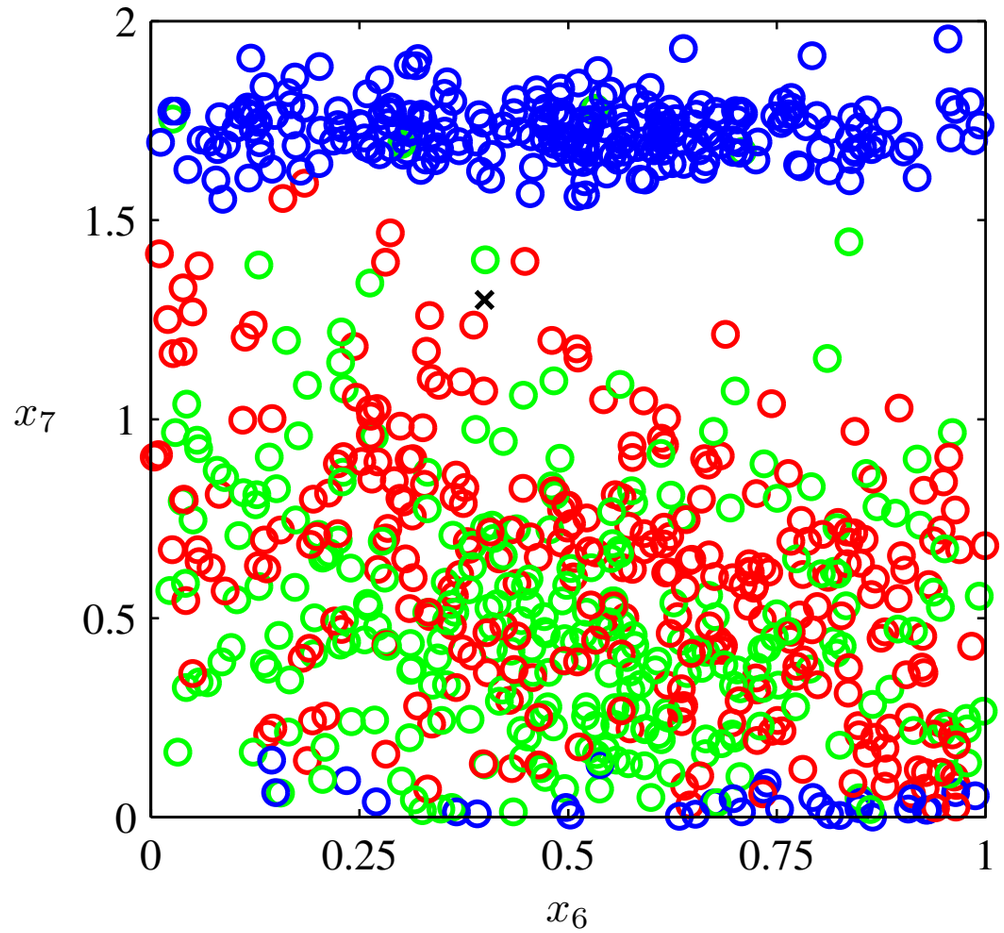


Figure: Not linearly separable dataset (Bishop 1.19)

Classification Strategies

- *Discriminant functions*

Direct mapping of input to target $t = y(x)$

- *Probabilistic generative models* → model joint $p(t, x)$

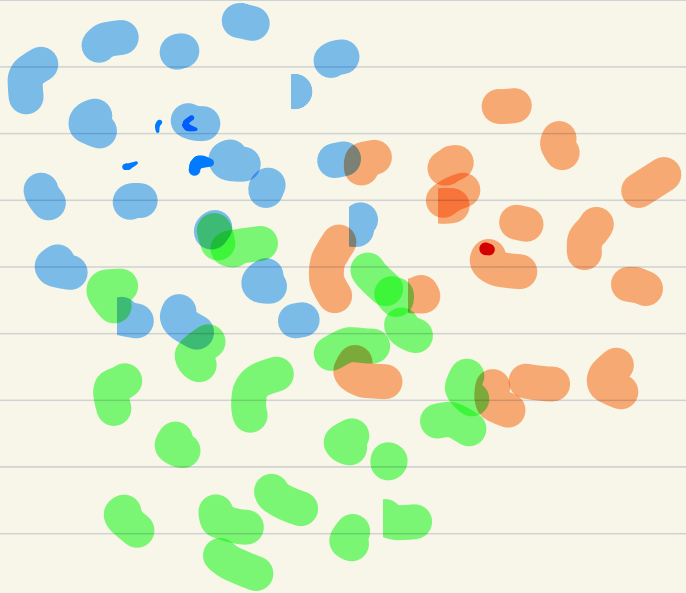
Marginal class probabilities: $p(C_1), \ldots, p(C_K)$

Class-conditional probabilities: $p(x|C_i)$

- *Probabilistic discriminative models*

Responsibilities: model $p(C_i|x)$

(don't care about marginal on $p(x)$ on x)



Overview

1. Linear Discriminant Analysis (LDA)
2. Logistic Regression

Overview

1. **Linear Discriminant Analysis (LDA)**
2. Logistic Regression

One-hot encoding of K classes

- Consider a discrete variable t with a value one of K classes:

$$\{C_1, \ldots, C_K\}$$

1, ..., K

- Then we can encoder t as a vector:

$t = c_1$

c_K

$t = (\mathbb{1}_{c_1}(t), \ldots, \mathbb{1}_{c_K}(t))$

$(1, 0, \ldots, 0)$ if $t = c_1$

$(0, 1, 0 \ldots, 0)$ if $t = c_2$

$\vdots$

$(0, 0, \ldots 0, 1)$ if $t = c_K$

Probabilistic Generative Models: K=2

- Class-conditional densities: $p(x|C_k)$
- Marginal class probabilities: $p(C_k)$ $k=1,\dots,K=2$
- Joint distribution: $\rightsquigarrow p(x,t) = p(x|t) \cdot p(t)$
- Point-conditional class probabilities ("responsibilities"): K=2

$$p(C_1|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_1)p(C_1) + p(\mathbf{x}|C_2)p(C_2)}$$

$$= \frac{1}{1 + \frac{p(x|C_2)\,p(C_2)}{p(x|C_1)\,p(C_1)}} = \frac{1}{1+\exp(-a)}$$

$\sigma(a)$

- $a = \ln \frac{\sigma}{1-\sigma} =$ $\quad a = \log \frac{p(x|C_1)\,p(C_1)}{p(x|C_2)\,p(C_2)}$

Logistic Sigmoid Function

$$\sigma(a) = \frac{1}{1 + \exp(-a)}$$

Symmetry:

$$\sigma(-a) = 1 - \sigma(a)$$

Derivative:

$$\sigma'(a) = \sigma(a)(1 - \sigma(a))$$

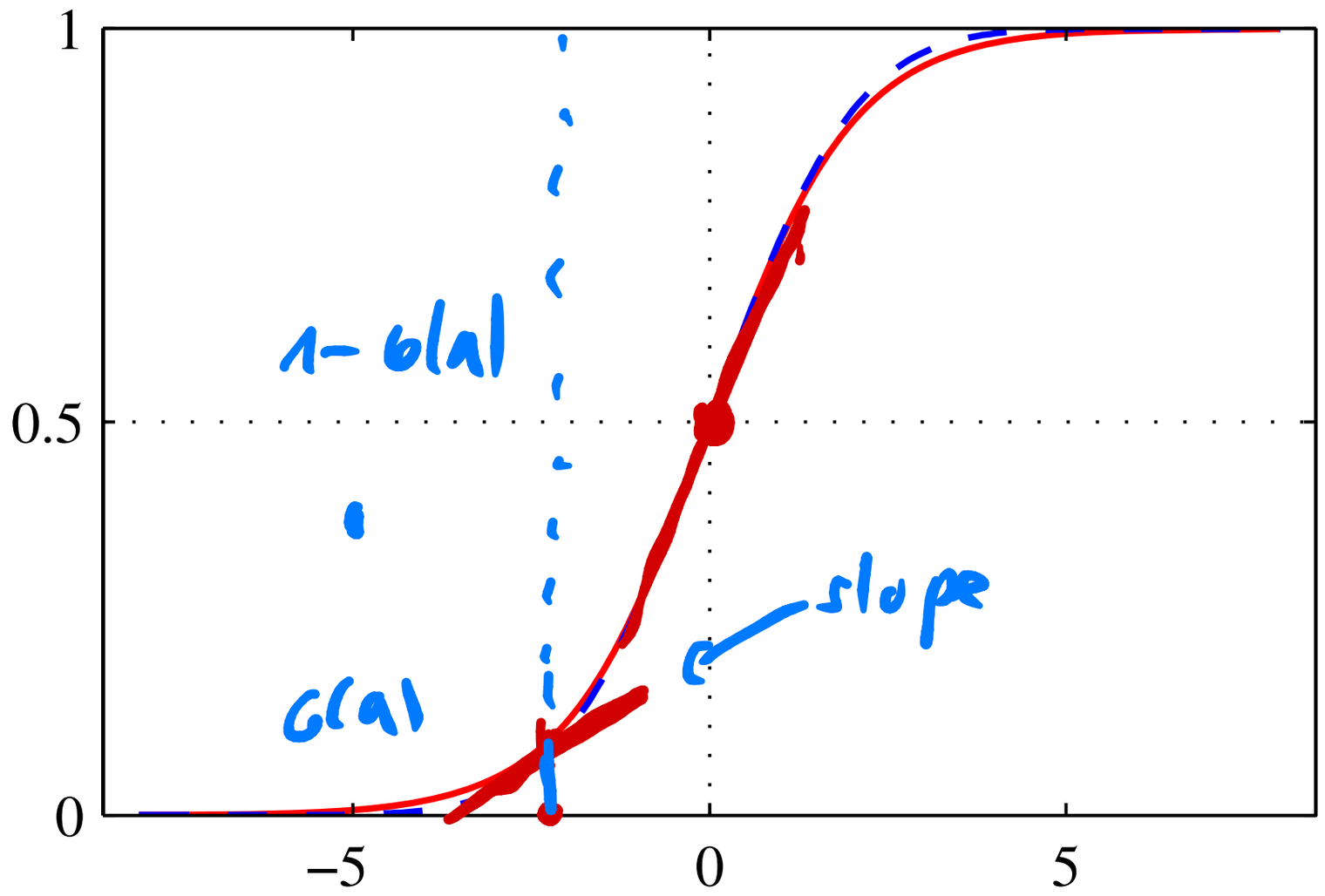


Figure: Logistic Sigmoid function (red) (Bishop 4.9)

Probabilistic Generative Models: general K

- For multiple classes (general K):

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{\sum_{j=1}^{K} p(\mathbf{x}|C_j)p(C_j)} = \frac{\exp(a_k)}{\sum_{j=1}^{K} \exp(a_j)}$$

- $a_k = \ln(p(\mathbf{x}|C_k)p(C_k))$

$a = (a_1, \ldots, a_K) \in \mathbb{R}^K$

$\text{Softmax}(a) =: \sigma(a)$

- Softmax: if $a_k >> a_j$ for all $j \neq k$:

dim a = K

- Note: for K=2:

$$p(C_1|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_1)p(C_1) + p(\mathbf{x}|C_2)p(C_2)} = \frac{1}{1 + \frac{p(\mathbf{x}|C_2)p(C_2)}{p(\mathbf{x}|C_1)p(C_1)}}$$

$$= \frac{1}{1 + \exp(-a)} = \sigma(a)$$

$$a = \ln \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)}$$

Class Conditional Densities: Continuous Inputs

- Gaussian Class-conditional densities:

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\mathbf{\Sigma}_k|^{1/2}} \exp\{\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\}$$

- Assume shared covariance matrix: $\mathbf{\Sigma}_k = \mathbf{\Sigma}$ ← LDA

LDA - assumption

- K=2 classes: $p(C_1|\mathbf{x}) = \frac{1}{1+\exp(-a)} = \sigma(a)$

$$a = \ln \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)} = \ln \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_1, \mathbf{\Sigma}) - \ln \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_2, \mathbf{\Sigma}) + \ln \frac{p(C_1)}{p(C_2)}$$

$$= -\frac{1}{2}\ln|\mathbf{\Sigma}| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_1) + \frac{1}{2}\ln|\mathbf{\Sigma}| + \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_2) + \ln \frac{p(C_1)}{p(C_2)}$$

$$= -\tfrac{1}{2}\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x} + \boldsymbol{\mu}_1^T\mathbf{\Sigma}^{-1}\mathbf{x} - \tfrac{1}{2}\boldsymbol{\mu}_1^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_1 + \tfrac{1}{2}\boldsymbol{\mu}_2^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_2 + \tfrac{1}{2}\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x} - \boldsymbol{\mu}_2^T\mathbf{\Sigma}^{-1}\mathbf{x} + \ln \frac{p(C_1)}{p(C_2)}$$

- Generalized Linear Model: $p(C_1|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$

$$\mathbf{w} = \mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$

$$w_0 = -\frac{1}{2}\boldsymbol{\mu}_1^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_2 + \ln \frac{p(C_1)}{p(C_2)}$$

linear in x

Example: Linear Discriminant Analysis for K=2

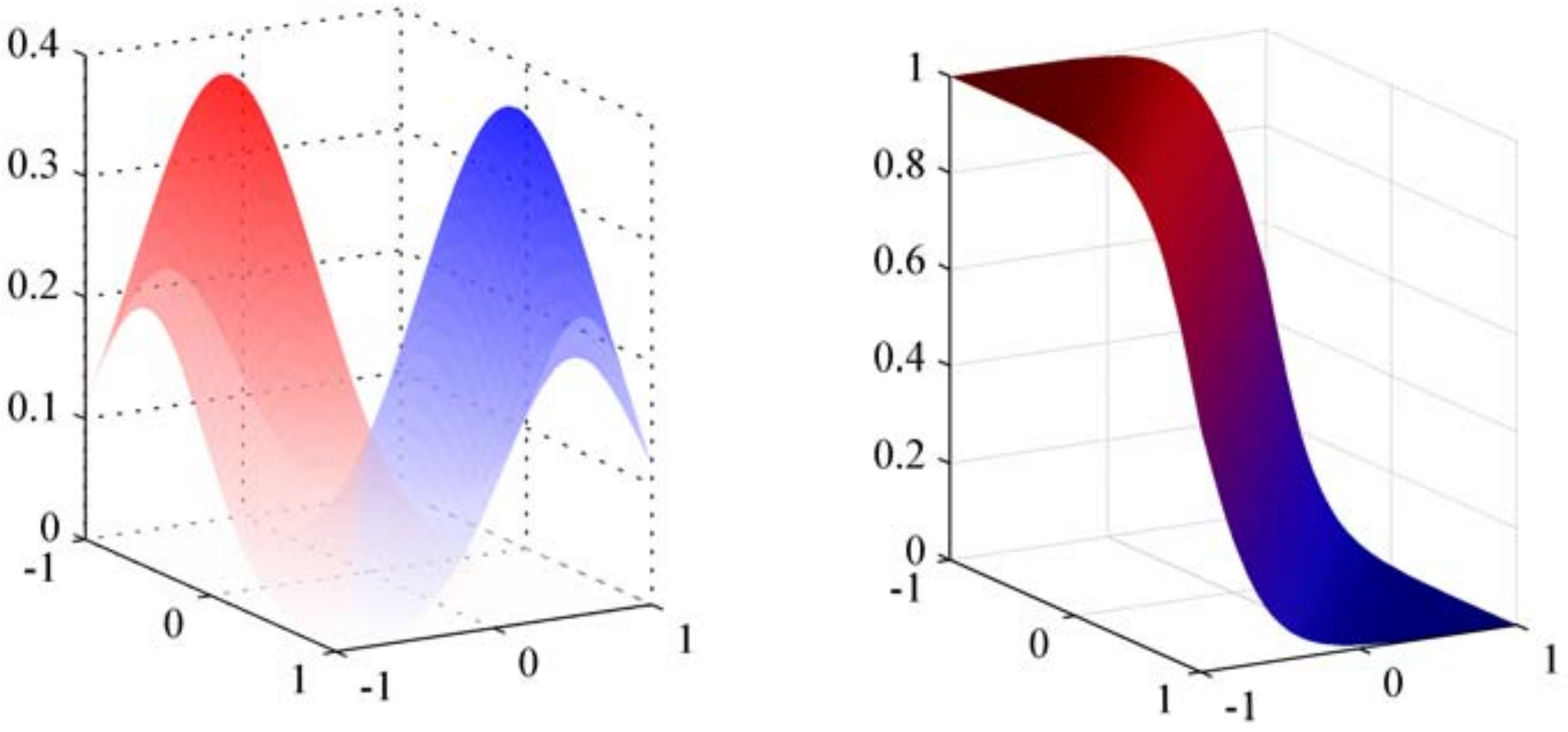


Figure: Left: class conditional densities p(x | C_k). Right: responsibility P(C_1|x) as sigmoid of linear function of x. (Bishop 4.9)

Linear Discriminant Analysis: General K

- Gaussian Class-conditional densities & fixed covariance:

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\mathbf{\Sigma}|^{1/2}} \exp\{\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\}$$

- Responsibilities:

$$p(C_k|\mathbf{x}) = \frac{\exp(a_k(\mathbf{x}))}{\sum_{j=1}^{K}\exp(a_j(\mathbf{x}))}$$

- $a_k(\mathbf{x}) = \mathbf{w}_k^T\mathbf{x} + w_{k0}$ $\qquad \mathbf{w}_k =$ $\qquad w_{k0} =$

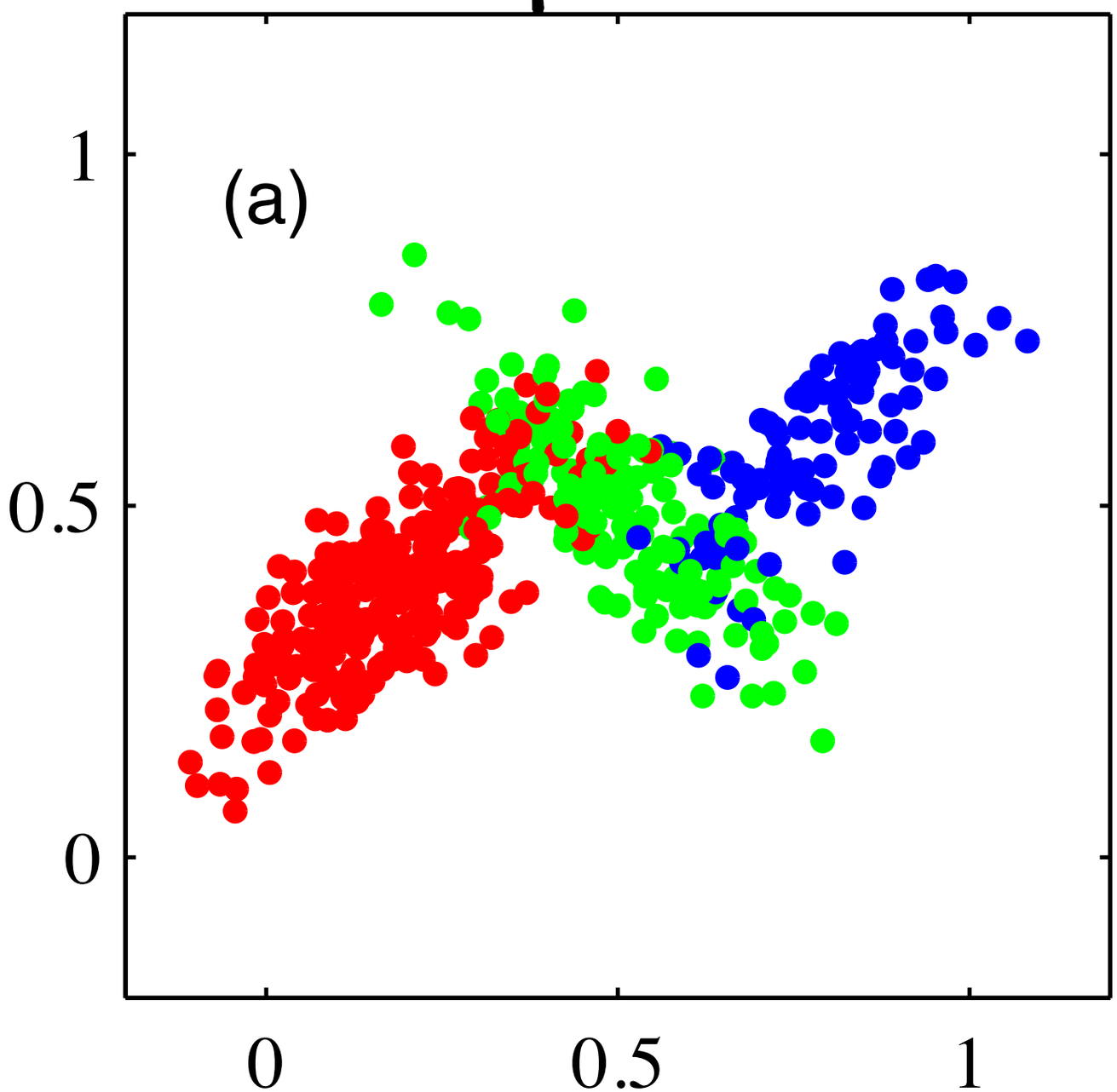
- Decision boundary:

$$p(C_k|\mathbf{x}) = p(C_j|\mathbf{x})$$

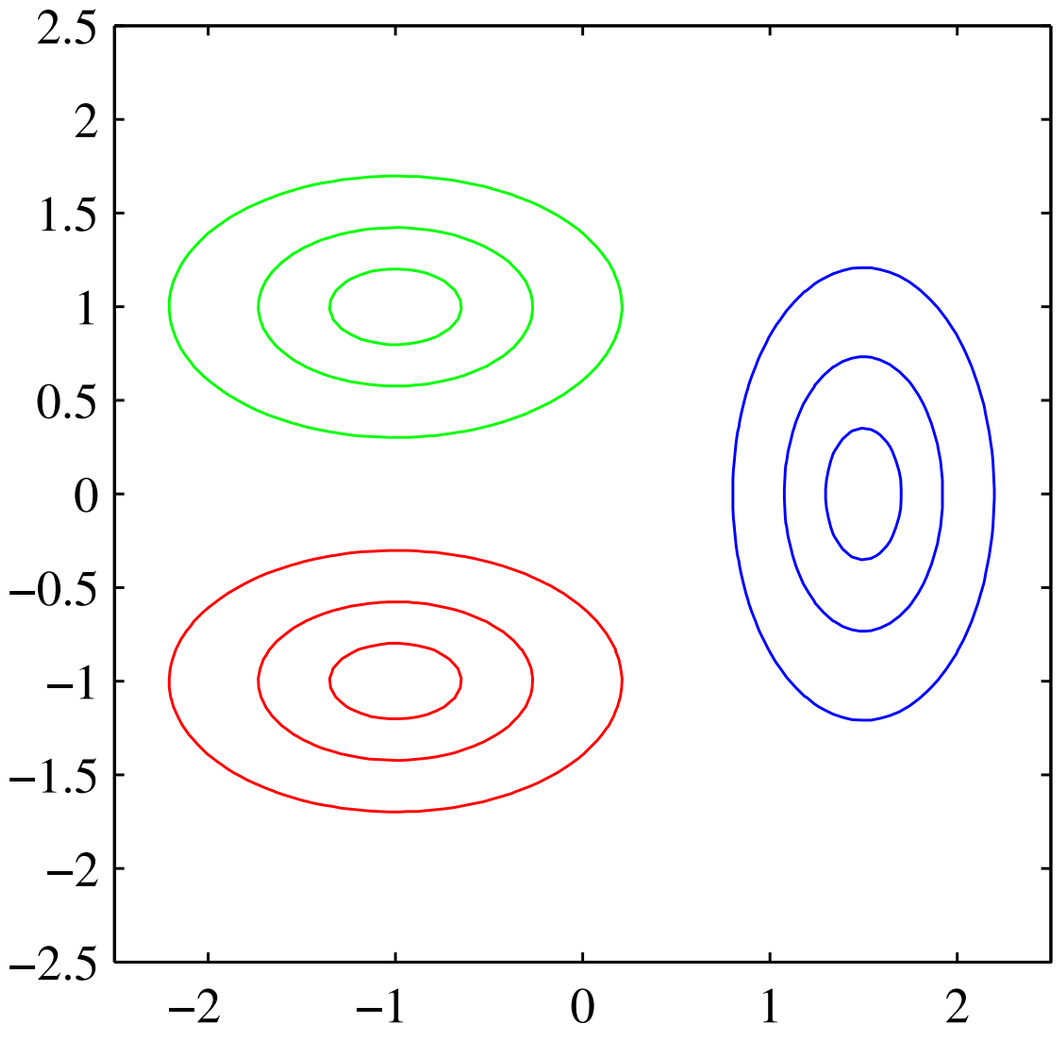
- If all covariance matrices are different $\mathbf{\Sigma}_k \neq \mathbf{\Sigma}_j$ then $a_k(\mathbf{x})$ will also contain quadratic terms in $\mathbf{x}$

Example: QDA for general K = 3

LDA for red + blue only (no green)



Example: LDA and QDA



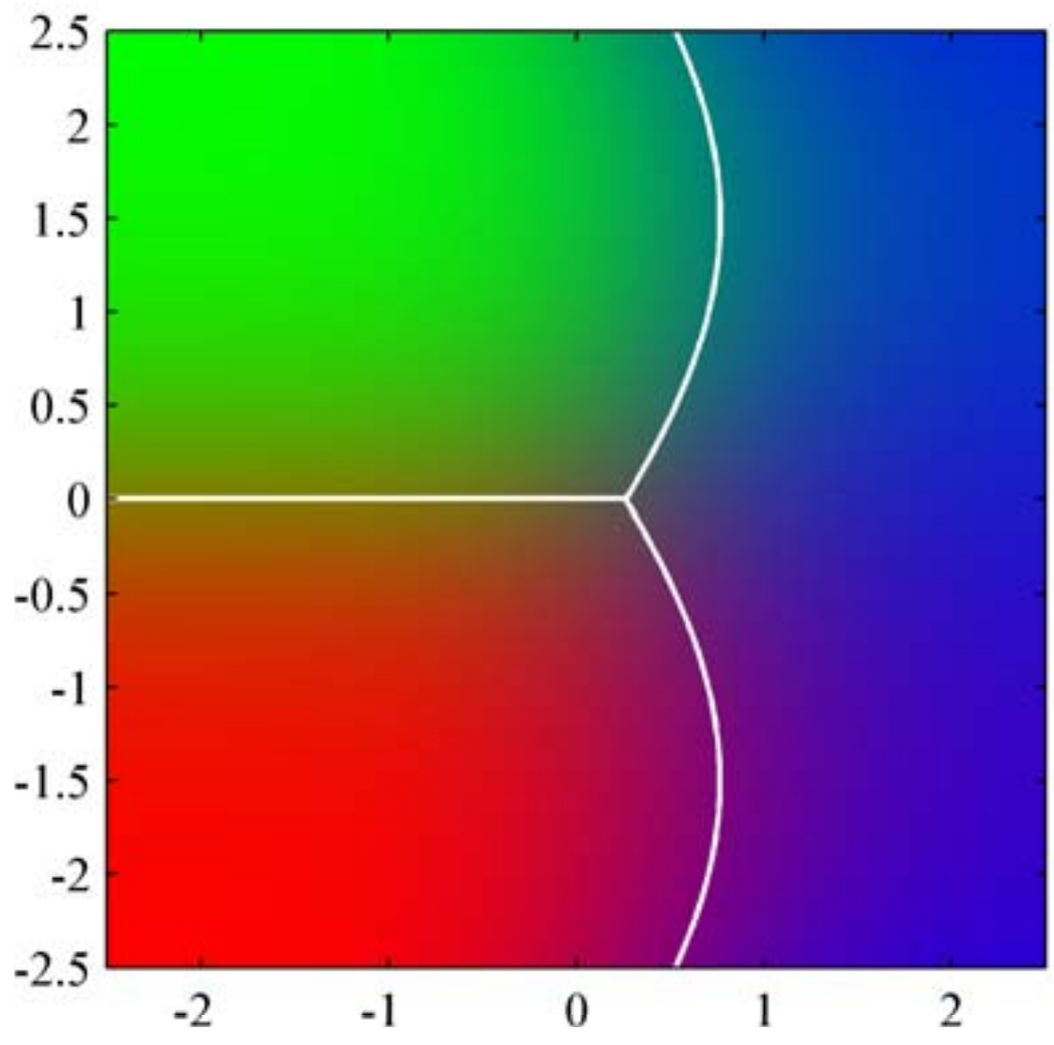


Figure: Left: Gaussian class conditional densities $p(x \mid C_k)$, red and green have same covariance matrix. Right: responsibilities $P(C_k \mid x)$ (RGB vectors) and decision boundaries. (Bishop 4.9)

LDA: Maximum Likelihood for K=2

- Dataset: input $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^T$, binary targets $\mathbf{t} = (t_1, ..., t_N)^T$

  $t_n \in \{0, 1\}$ $\quad \mathbf{x}_n \in \mathbb{R}^D$

- Gaussian conditional densities

  LDA assumption

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\mathbf{\Sigma}|^{1/2}} \exp\{\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\}$$

- Use Maximum likelihood to estimate $\boldsymbol{\mu}_k, \mathbf{\Sigma}$ and marginals $p(C_1)$ $\quad k=1,2$

- Denote $p(C_1) = q$ and $p(C_2) = 1 - q$

- For $\mathbf{x}_n$ with $t_n = 1$: $p(\mathbf{x}_n, C_1) = p(\mathbf{x}_n|C_1)p(C_1) = N(x_n|\mu_1, \Sigma) \cdot q$

- For $\mathbf{x}_n$ with $t_n = 0$: $p(\mathbf{x}_n, C_2) = p(\mathbf{x}_n|C_2)p(C_2) = N(x_n|\mu_0, \Sigma) \cdot (1-q)$

- Likelihood:

$$p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \mathbf{\Sigma}) \overset{\text{iid}}{=} \prod_{n=1}^{N} \left(q\, N(x_n|\mu_1, \Sigma)\right)^{t_n} \cdot \left((1-q) \cdot N(x_n|\mu_0, \Sigma)\right)^{(1-t_n)}$$

LDA: Maximum Likelihood for K=2

- Likelihood

$$p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \prod_{n=1}^{N} [q\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma})]^{t_n} [(1-q)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma})]^{1-t_n}$$

- Log likelihood

$$\ln p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} t_n \ln q + t_n \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}) +$$
$$+ (1-t_n) \ln(1-q) + (1-t_n) \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$$

- Estimate for q:

$$0 = \frac{\partial}{\partial q} \ln p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} t_n \cdot \frac{1}{q} + (1-t_n) \frac{1}{1-q} \cdot (-1)$$

$$p(C_1) \sim q_{\mathrm{ML}} = \frac{1}{N} \sum_{n=1}^{N} t_n = \frac{N_1}{N}$$

$N_1 = \#\{ x_n \mid t_n = 1 \}$

LDA: Maximum Likelihood for K=2

- log likelihood:

$$\ln p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} t_n \ln q + t_n \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}) + \\ + (1 - t_n) \ln(1 - q) + (1 - t_n) \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$$

- Estimate for $\boldsymbol{\mu}_1$

$$\frac{\partial}{\partial \boldsymbol{\mu}_1} \ln p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \frac{\partial}{\partial \boldsymbol{\mu}_1} \sum_{n=1}^{N} t_n \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}) =$$

$$= -\frac{1}{2} \frac{\partial}{\partial \boldsymbol{\mu}_1} \sum_{n=1}^{N} t_n (\mathbf{x}_n - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_1) =$$

$$\Rightarrow \hat{\boldsymbol{\mu}}_{1,ML} = \frac{1}{N_1} \sum_{n=1}^{N} t_n \mathbf{x}_n$$

$$\boldsymbol{\mu}_{2,\mathrm{ML}} = \frac{1}{N_2} \sum_{n=1}^{N} (1 - t_n) \mathbf{x}_n$$

LDA: Maximum Likelihood for K=2

- log likelihood:

$$\ln p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} t_n \ln q + t_n \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}) + \\ + (1 - t_n) \ln(1 - q) + (1 - t_n) \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$$

- Estimate for $\boldsymbol{\Sigma}$

$$\frac{\partial}{\partial \boldsymbol{\Sigma}} \ln p(\mathbf{t}, \mathbf{X}|q, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \frac{\partial}{\partial \boldsymbol{\Sigma}} \left[-\frac{N}{2} \ln |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{n=1}^{N} t_n (\mathbf{x}_n - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_1) \right.$$

$$\left. - \frac{1}{2} \sum_{n=1}^{N} (1 - t_n)(\mathbf{x}_n - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_2) \right] = 0$$

- ML solution:

$$\boldsymbol{\Sigma}_{\mathrm{ML}} = \frac{N_1}{N} \left[\frac{1}{N_1} \sum_{n=1}^{N} t_n (\mathbf{x}_n - \boldsymbol{\mu}_{1,\mathrm{ML}})(\mathbf{x}_n - \boldsymbol{\mu}_{1,\mathrm{ML}})^T \right] \quad \Sigma_{1,\mathrm{ML}}$$

$$+ \frac{N_2}{N} \left[\frac{1}{N_2} \sum_{n=1}^{N} (1 - t_n)(\mathbf{x}_n - \boldsymbol{\mu}_{2,\mathrm{ML}})(\mathbf{x}_n - \boldsymbol{\mu}_{2,\mathrm{ML}})^T \right] \quad \Sigma_{2,\mathrm{ML}}$$

LDA: prediction for K=2

- For new datapoint **x**':

$$p(C_1|\mathbf{x}') = \sigma(\mathbf{w}_{\mathrm{ML}}^T\mathbf{x}' + w_{0,\mathrm{ML}})$$

$$\mathbf{w}_{\mathrm{ML}} = \boldsymbol{\Sigma}_{\mathrm{ML}}^{-1}(\boldsymbol{\mu}_{1,\mathrm{ML}} - \boldsymbol{\mu}_{2,\mathrm{ML}})$$

$$w_{0,\mathrm{ML}} = -\frac{1}{2}\boldsymbol{\mu}_{1,\mathrm{ML}}^T\boldsymbol{\Sigma}_{\mathrm{ML}}^{-1}\boldsymbol{\mu}_{1,\mathrm{ML}} + \frac{1}{2}\boldsymbol{\mu}_{2,\mathrm{ML}}^T\boldsymbol{\Sigma}_{\mathrm{ML}}^{-1}\boldsymbol{\mu}_{2,\mathrm{ML}} + \ln\frac{q_{\mathrm{ML}}}{1-q_{\mathrm{ML}}}$$

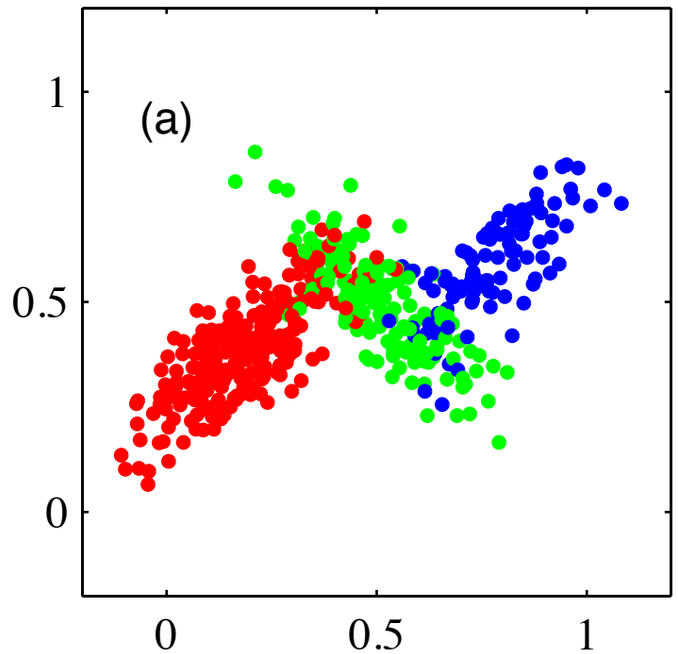
- Assign **x**' to C_1 if $p(C_1|\mathbf{x}') \geq \frac{1}{2}$
- Disadvantage of LDA:
  - Gaussian distribution is sensitive to outliers
  - Linearity/handcrafted features restrict application
  - Maximum likelihood is prone to overfitting

LDA for general K

- Model class-conditional densities: $p(\mathbf{x}|C_k)$

- For multiple classes (general K):

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{\sum_{j=1}^{K} p(\mathbf{x}|C_j)p(C_j)} = \frac{e^{a_k}}{\sum_{j=1}^{K} e^{a_j}}$$



- $a_k = \ln p(\mathbf{x}|C_k)p(C_k)$

- Softmax: if $a_k >> a_j$ for all $j \neq k$:

- Why rewrite responsibilities like this?

- The equation for the decision boundary at $p(C_k|\mathbf{x}) = p(C_j|\mathbf{x})$ is at $a_k = a_j$

Probabilistic Generative Models: Discrete

- Input: discrete feature vectors $\mathbf{x}_n = (x_1, ..., x_D)^T$

  $x_i \in \{0, 1\}$

- For D-dimensional input:

  # of parameters for $p(\mathbf{x}|C_k)$ per class:

- Naive Bayes assumption: feature values are treated as independent *when conditioned on class C_k!*

- $p(\mathbf{x}|C_k) =$

- $p(C_k|\mathbf{x}) = \frac{\exp(a_k(\mathbf{x}))}{\sum_{j=1}^{K} \exp(a_j(\mathbf{x}))}$

$$a_k(\mathbf{x}) = \ln(p(\mathbf{x}|C_k)p(C_k))$$

$$=$$

LDA: generalizations

- LDA with multiple classes K (follows similarly)
- Quadratic Discriminant Analysis (QDA), where each class has its own covariance matrix Σ_k (follows similarly)
- Use of basis functions in LDA and QDA

Example: Use of Basis Functions

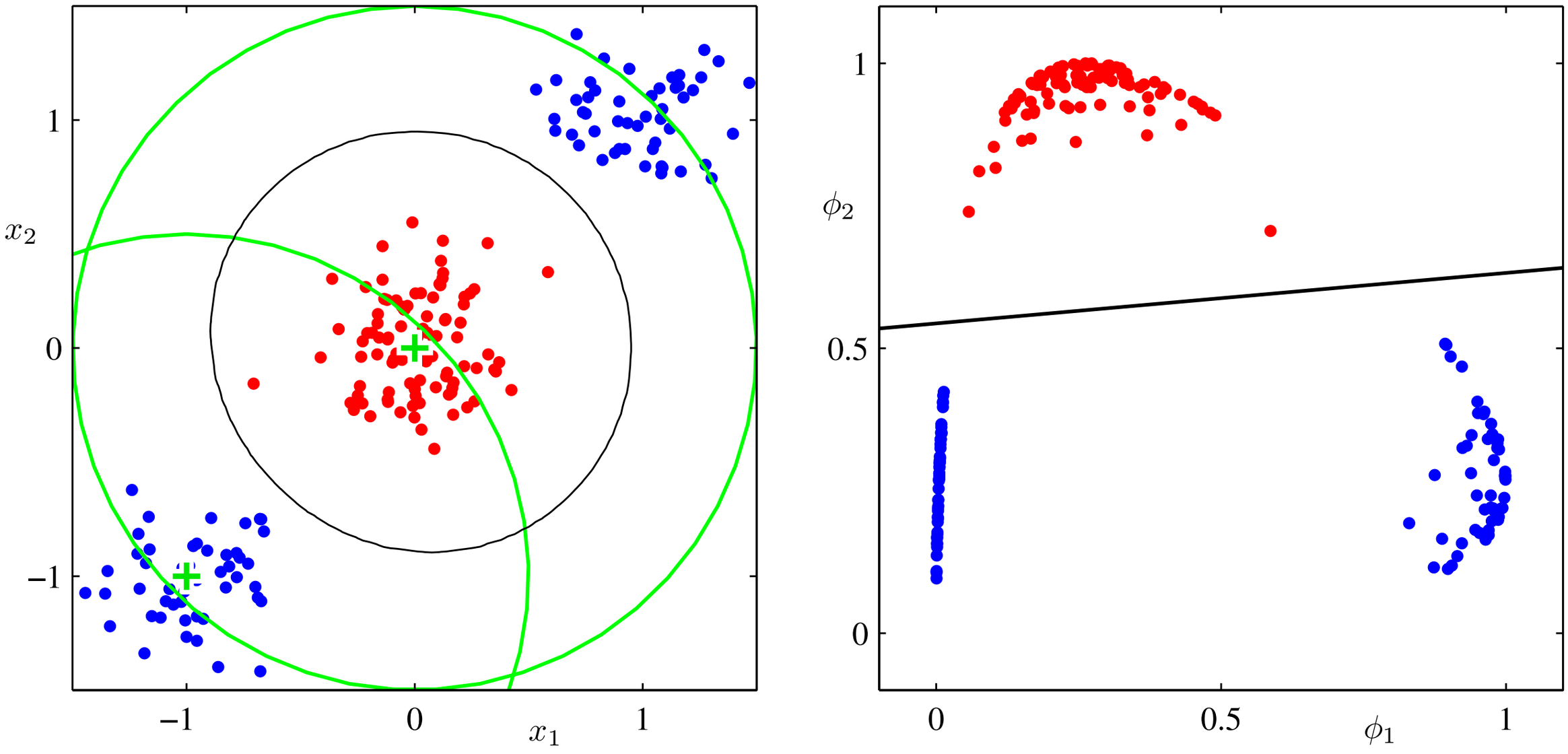


Figure: Left: original input space (x_1, x_2) , right: space of two gaussian basis functions with centres shown by the green crosses. (Bishop 4.12)

Overview

1. Linear Discriminant Analysis (LDA)
2. **Logistic Regression**

Logistic Regression for Two Classes

- Given: Dataset $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^T$ with binary targets

$$\mathbf{t} = (t_1, ..., t_N)^T \quad \text{with} \quad t_n \in \{\mathcal{C}_1, \mathcal{C}_2\} = \{1, 0\}$$

- Basis functions $\boldsymbol{\phi} = \boldsymbol{\phi}(\mathbf{x}) =$ $\begin{pmatrix} \phi_0(x) \\ \vdots \\ \phi_{M-1}(x) \end{pmatrix} \in \mathbb{R}^M$

$$\phi_0(\mathbf{x}) = 1$$

- Probabilistic Discriminative Linear Models: responsibilities $p(\mathcal{C}_k|\boldsymbol{\phi})$ are nonlinear functions with a linear function of $\boldsymbol{\phi}$ as input.

$$p(\mathcal{C}_k|\boldsymbol{\phi}, \mathbf{w}) = f(\mathbf{w}^T\boldsymbol{\phi})$$

linear in w

- Logistic regression (K=2)

$$p(\mathcal{C}_1|\boldsymbol{\phi}, \mathbf{w}) = y(\boldsymbol{\phi}) = \sigma(\mathbf{w}^T\boldsymbol{\phi})$$

$$p(\mathcal{C}_2|\boldsymbol{\phi}, \mathbf{w}) = 1 - \sigma(\mathbf{w}^T\boldsymbol{\phi})$$

$$p(t|\boldsymbol{\phi}, \mathbf{w}) = \sigma(\mathbf{w}^T\boldsymbol{\phi})^t \cdot [1 - \sigma(\mathbf{w}^T\boldsymbol{\phi})]^{(1-t)}$$

Logistic Regression for Two Classes

- Given: Dataset $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^T$ with binary targets

  $\mathbf{t} = (t_1, ..., t_N)^T$ with $t_n \in \{\mathcal{C}_1, \mathcal{C}_2\} = \{1, 0\}$

- Conditional likelihood function:

  $$p(\mathbf{t}|\mathbf{X}, \mathbf{w}) \overset{\text{iid}}{=} \prod_{n=1}^{N} \underbrace{\sigma(\mathbf{w}^T\boldsymbol{\phi}_n)}_{y_n}^{t_n} \cdot (1 - \underbrace{\sigma(\mathbf{w}^T\boldsymbol{\phi}_n)}_{y_n})^{1-t_n}$$

  $y_n = p(\mathcal{C}_1|\boldsymbol{\phi}_n)$ $\qquad$ $\boldsymbol{\phi}_n = \boldsymbol{\phi}(\mathbf{x}_n)$

  $p(\mathcal{C}_1|\mathbf{x}) = \sigma(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}))$

- Maximizing the (conditional) likelihood/minimizing the cross-entropy

  $$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}) = -\sum_{n=1}^{N} t_n \log y_n + (1-t_n)\log(1-y_n)$$

  L1 Cross-entropy

- E(**w**): convex, but no closed form solution!

  $y_n = \sigma(\mathbf{w}^T\boldsymbol{\phi}_n)$ is nonlinear in **w**

Logistic Regression (K=2): SGD

- Stochastic Gradient Descent for cross-entropy:

$$E(\mathbf{w}) = -\sum_{n=1}^{N} t_n \ln y_n + (1 - t_n) \ln(1 - y_n) = \sum_{n=1}^{N} E_n(\mathbf{w})$$

$-E_n(\mathbf{w})$

- Update rule given a random data point $(\mathbf{x}_n, t_n)$

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_n(\mathbf{w}^{(\tau)})$$

$y_n = \sigma(\mathbf{w}^T \phi_n)$

- Gradient: $\nabla E_n(\mathbf{w}) = \left(\frac{\partial E_n(\mathbf{w})}{\partial w_0}, \ldots, \frac{\partial E_n(\mathbf{w})}{\partial w_{M-1}} \right)$

- $\frac{\partial E_n(\mathbf{w})}{\partial w_j} = \frac{\partial E_n(\mathbf{w})}{\partial y_n} \cdot \frac{\partial y_n}{\partial w_j} = \left(t_n \cdot \frac{1}{y_n} + (1 - t_n) \cdot \frac{1}{1 - y_n} (-1) \right) \cdot \frac{\partial y_n}{\partial w_j}$

- $\frac{\partial y_n}{\partial w_j} = \frac{\partial}{\partial w_j} \sigma(\mathbf{w}^T \phi_n) = \sigma(\mathbf{w}^T \phi_n) \cdot (1 - \sigma(\mathbf{w}^T \phi_n)) \cdot \phi_{nj}$

$= y_n (1 - y_n) \cdot \phi_{nj}$

Logistic Regression (K=2): SGD

- $\frac{\partial y_n}{\partial w_j} = \frac{\partial}{\partial w_j}\sigma(\mathbf{w}^T\boldsymbol{\phi}_n)$ $= y_n(1-y_n)\phi_{nj}$
- Use $\frac{\partial \sigma(a)}{\partial a} = \sigma(a)(1-\sigma(a))$
- $\frac{\partial}{\partial w_j}\sigma(\mathbf{w}^T\boldsymbol{\phi}_n) =$ $\left(t_n \cdot \frac{1}{y_n} + (1-t_n)\frac{-1}{1-y_n}\right)$.

- $\frac{\partial E_n(\mathbf{w})}{\partial w_j} = -\frac{t_n}{y_n}\frac{\partial y_n}{\partial w_j} + \frac{1-t_n}{1-y_n}\frac{\partial y_n}{\partial w_j}$

  $= -t_n(1-y_n)\phi_{nj} + (1-t_n)\cdot y_n\phi_{nj}$

  $= (y_n - t_n)\cdot\phi_{nj}$

Logistic Regression (K=2): SGD

- Stochastic Gradient Descent for cross-entropy:

$$E(\mathbf{w}) = -\sum_{n=1}^{N} t_n \ln y_n + (1 - t_n) \ln(1 - y_n) = \sum_n E_n(w)$$

- Update rule given a random data point $(\mathbf{x}_n, t_n)$

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_n(\mathbf{w}^{(\tau)})^T$$

- $$\frac{\partial E_n(\mathbf{w})}{\partial w_j} = (y_n - t_n)\phi_j(\mathbf{x}_n)$$
- Gradient: $$\nabla E_n(\mathbf{w}) = \left(\frac{\partial E_n(\mathbf{w})}{\partial w_0}, ..., \frac{\partial E_n(\mathbf{w})}{\partial w_{M-1}} \right)^T = (y_n - t_n) \cdot \phi_n^T$$
- Update rule:

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \, (y_n - t_n) \cdot \phi_n$$

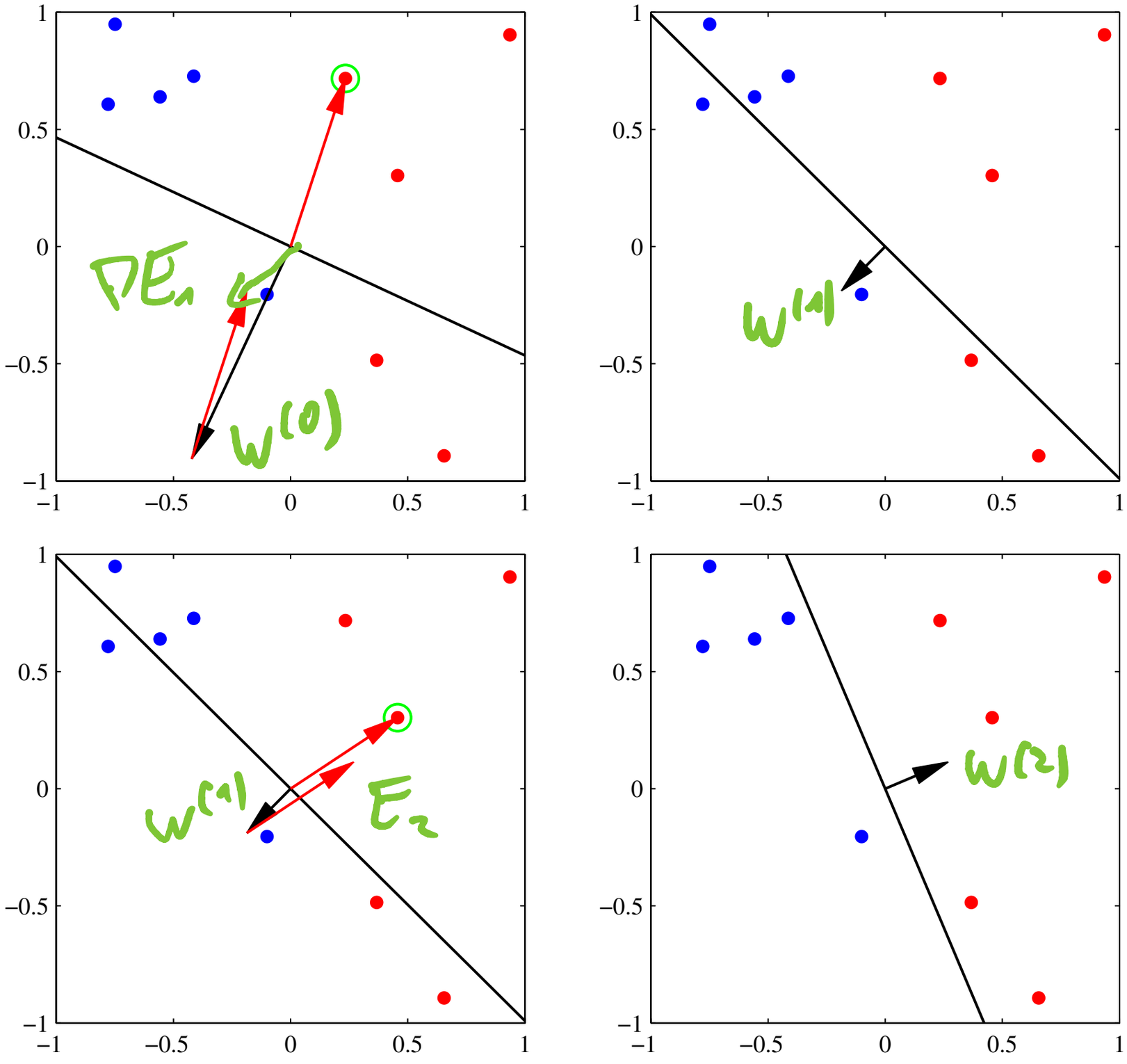
Stochastic Gradient Descent (SGD)

1. Initialize $\mathbf{w}^{(0)}$
2. Choose a learning rate η
3. While $||\mathbf{w}^{(\tau+1)} - \mathbf{w}^{(\tau)}|| > \varepsilon$
    I. Choose a random data point $(\mathbf{x}_n, t_n)$
    II. Update **w:**

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta\ (y_n^{(\tau)} - t_n)\boldsymbol{\phi}(\mathbf{x}_n)$$

- If η too large: no convergence
- If η too small: very slow convergence
- Converged **w***: estimate of minimizer of E(**w**)!

Stochastic Gradient Descent (SGD)



linearly separated

Figure: (Bishop 4.7)

Classification with Logistic Regression

- Dataset: $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^T$ with targets

  $\mathbf{t} = (t_1, ..., t_N)^T$ with $t_n \in \{\mathcal{C}_1, \mathcal{C}_2\} = \{1, 0\}$

- Basis functions $\boldsymbol{\phi}(\mathbf{x}) = (\phi_0(\mathbf{x}), ..., \phi_{M-1}(\mathbf{x}))^T$

- Responsibilities: $p(\mathcal{C}_1|\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}))$

- Minimize $E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}) = -\sum_{n=1}^{N} t_n \ln y_n + (1 - t_n)\ln(1 - y_n)$

  with stochastic gradient descent or iterative reweighted least squares (see later), to find $\mathbf{w}^*$

- New datapoint $\mathbf{x}'$ is assigned to class C_1 if

  $p(\mathcal{C}_1|\mathbf{x}', \mathbf{w}^*) = \sigma((\mathbf{w}^*)^T\boldsymbol{\phi}(\mathbf{x}')) >$ $p(C_2|x', w^*) = 1 - p(C_1|x', w^*)$

- Decision boundaries: i.e: $\frac{1}{2}$

$= \frac{1}{2} \hookrightarrow \sigma > \frac{1}{2} \rightarrowtail w^T\phi(x) = 0$

Logistic Regression for Multiple Classes

- Data: $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^T$ with targets $\mathbf{T} = (\mathbf{t}_1, ..., \mathbf{t}_N)^T$

  one-hot vectors: $\mathbf{t}_n = (0, 0, .., 1, 0, ..0)^T$

- Model assumption:

$$p(\mathcal{C}_k|\mathbf{x}, \mathbf{w}_1, ..., \mathbf{w}_K) = y_k(\boldsymbol{\phi}) = \frac{\exp(a_k)}{\sum_{j=1}^{K} \exp(a_j)} = \text{softmax}(\Psi \phi(x))$$

$$a_k = \mathbf{w}_k^T \boldsymbol{\phi}(x)$$

- Conditional likelihood

$$p(\mathbf{T}|\mathbf{X}, \mathbf{w}_1, ..., \mathbf{w}_K) = \prod_{n=1}^{N} \prod_{k=1}^{K} p(\mathcal{C}_k|\mathbf{x}_n, \mathbf{w}_1, ..., \mathbf{w}_K)^{t_{nk}} = \prod_{n,k} y_n^{t_{nk}}$$

- Cross-entropy error function:

$$E(\mathbf{w}_1, ..., \mathbf{w}_K) = -\ln p(\mathbf{T}|\mathbf{X}, \mathbf{w}_1, ..., \mathbf{w}_K) =$$

$$- \sum_{n,k} t_{nk} \log y_n$$

Logistic Regression for Multiple Classes

- Cross-entropy error:

$$E(\mathbf{w}_1, ..., \mathbf{w}_K) = -\ln p(\mathbf{T}|\mathbf{X}, \mathbf{w}_1, ..., \mathbf{w}_K) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk} \ln y_{nk}$$

- Gradient w.r.t. $\mathbf{w}_j$: $\nabla_{\mathbf{w}_j} E(\mathbf{w}_1, ..., \mathbf{w}_K) =$ $(y_n - t_{n1}) \cdot \phi_j$
- Hessian contains blocks, block (j,k) contains:

$$\frac{\partial}{\partial \mathbf{w}_k}\frac{\partial}{\partial \mathbf{w}_j^T} E(\mathbf{w}_1, ..., \mathbf{w}_K) = \sum_{n=1}^{N} y_{nk}(\mathbb{1}_{kj} - y_{nj})\boldsymbol{\phi}_n \boldsymbol{\phi}_n^T$$

- After minimizing E($\mathbf{w}_1$,…,$\mathbf{w}_K$) New datapoint **x**' is assigned to class C_k if $p(\mathcal{C}_k|\mathbf{x}', \mathbf{w}_1, ..., \mathbf{w}_K) > p(\mathcal{C}_j|\mathbf{x}', \mathbf{w}_1, ..., \mathbf{w}_K) \quad \forall j \neq k$

- Decision boundaries:

$$(\mathbf{w}_k^*)^T \boldsymbol{\phi}(\mathbf{x}') = (\mathbf{w}_j^*)^T \boldsymbol{\phi}(\mathbf{x}')$$

Bayesian Logistic Regression (K=2)

- Conditional likelihood

$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n}(1-y_n)^{1-t_n} \qquad y_n = \sigma(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n))$$

- Gaussian prior: $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0,\mathbf{S}_0)$
- Posterior: $p(\mathbf{w}|\mathbf{t},\mathbf{X}) \propto p(\mathbf{t}|\mathbf{X},\mathbf{w})p(\mathbf{w})$

$$\ln p(\mathbf{w}|\mathbf{t},\mathbf{X}) = -\frac{1}{2}(\mathbf{w}-\mathbf{m}_0)^T\mathbf{S}_0(\mathbf{w}-\mathbf{m}_0) + \sum_{n=1}^{N} t_n \ln y_n + (1-t_n)\ln(1-y_n) + \text{const}$$

- Laplace approximation to posterior:

$$\ln p(\mathbf{w}|\mathbf{t},\mathbf{X}) \approx \mathcal{N}(\mathbf{w}|\mathbf{w}_{\text{MAP}},\mathbf{S}_N)$$

$$\mathbf{S}_N^{-1} = -\nabla\nabla \ln p(\mathbf{w}|\mathbf{t},\mathbf{X})|_{\mathbf{w}=\mathbf{w}_{\text{MAP}}} = \mathbf{S}_0^{-1} + \sum_{n=1}^{N} y_n(1-y_n)\boldsymbol{\phi}_n\boldsymbol{\phi}_n^T$$

- Predictive distribution

$$p(\mathcal{C}_1|\mathbf{x}',\mathbf{t},\mathbf{X}) = \int p(\mathcal{C}_1|\mathbf{x}',\mathbf{w})\mathcal{N}(\mathbf{w}|\mathbf{w}_{\text{MAP}},\mathbf{S}_N)\mathrm{d}\mathbf{w} \approx \sigma\left((1+\frac{\pi}{8}\boldsymbol{\Phi}^T\mathbf{S}_N\boldsymbol{\Phi})^{-1/2}\mathbf{w}_{\text{MAP}}^T\boldsymbol{\phi}(\mathbf{x}')\right)$$