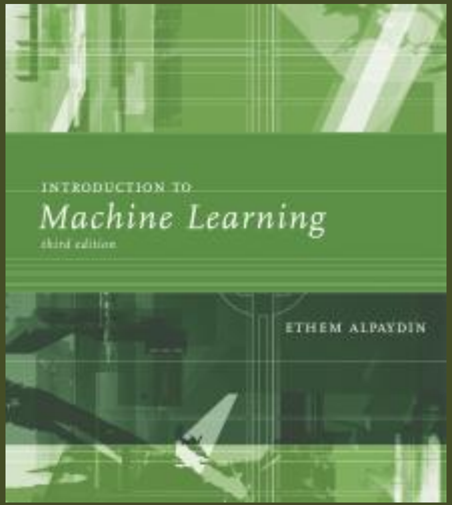


Lecture Slides for

INTRODUCTION TO MACHINE LEARNING

3RD EDITION

ETHEM ALPAYDIN

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml3e

CHAPTER 10:

LINEAR DISCRIMINATION

Likelihood- vs. Discriminant-based Classification

- Likelihood-based: Assume a model for $p(\mathbf{x}|C_i)$, use Bayes' rule to calculate $P(C_i|\mathbf{x})$

  $$g_i(\mathbf{x}) = \log P(C_i|\mathbf{x})$$

- Discriminant-based: Assume a model for $g_i(\mathbf{x}|\Phi_i)$; no density estimation
- Estimating the boundaries is enough; no need to accurately estimate the densities inside the boundaries

Linear Discriminant

- Linear discriminant:

$$g_i(\mathbf{x}|\mathbf{w}_i,w_{i0}) = \mathbf{w}_i^T\mathbf{x} + w_{i0} = \sum_{j=1}^{d} w_{ij}x_j + w_{i0}$$

- Advantages:
  - Simple: $O(d)$ space/computation
  - Knowledge extraction: Weighted sum of attributes; positive/negative weights, magnitudes (credit scoring)
  - Optimal when $p(\mathbf{x}|C_i)$ are Gaussian with shared cov matrix; useful when classes are (almost) linearly separable

Generalized Linear Model

- Quadratic discriminant:

$$g_i(\mathbf{x} \mid \mathbf{W}_i, \mathbf{w}_i, w_{i0}) = \mathbf{x}^T \mathbf{W}_i \mathbf{x} + \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

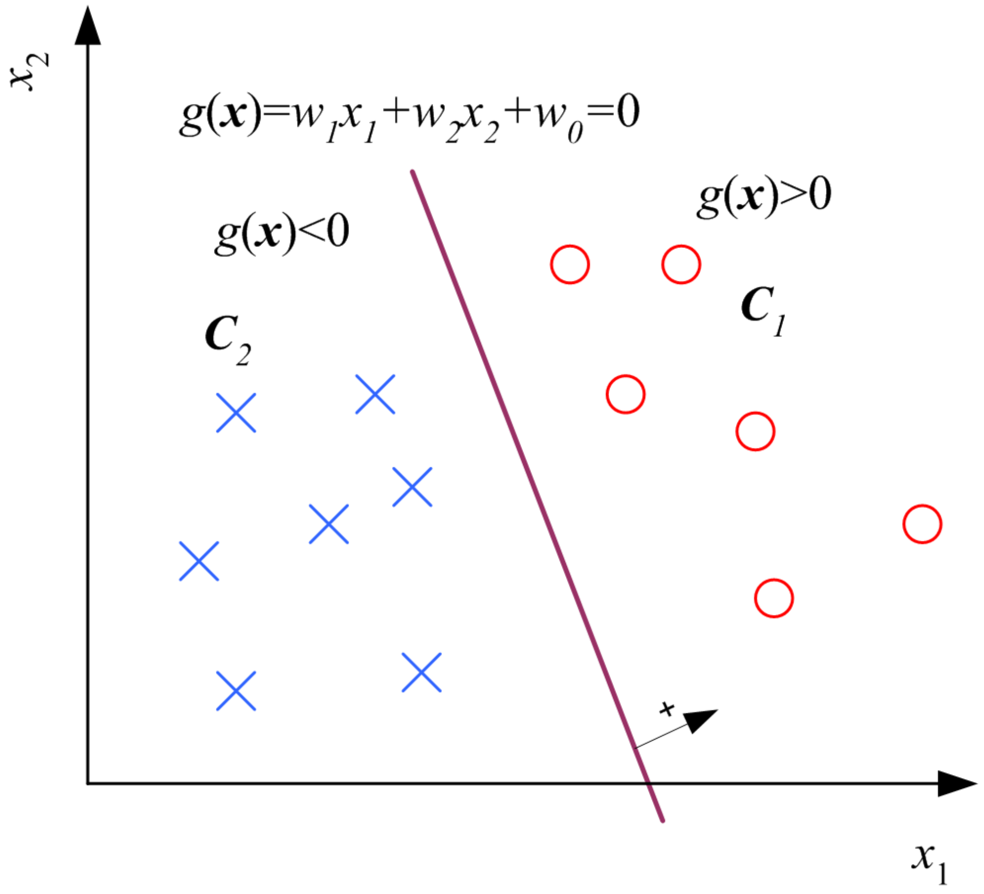
- Higher-order (product) terms:

$$z_1 = x_1,\ z_2 = x_2,\ z_3 = x_1^2,\ z_4 = x_2^2,\ z_5 = x_1 x_2$$

Map from ***x*** to ***z*** using nonlinear basis functions and use a linear discriminant in ***z***-space

$$g_i(\mathbf{x}) = \sum_{j=1}^{k} w_{ij} \phi_j(\mathbf{x})$$

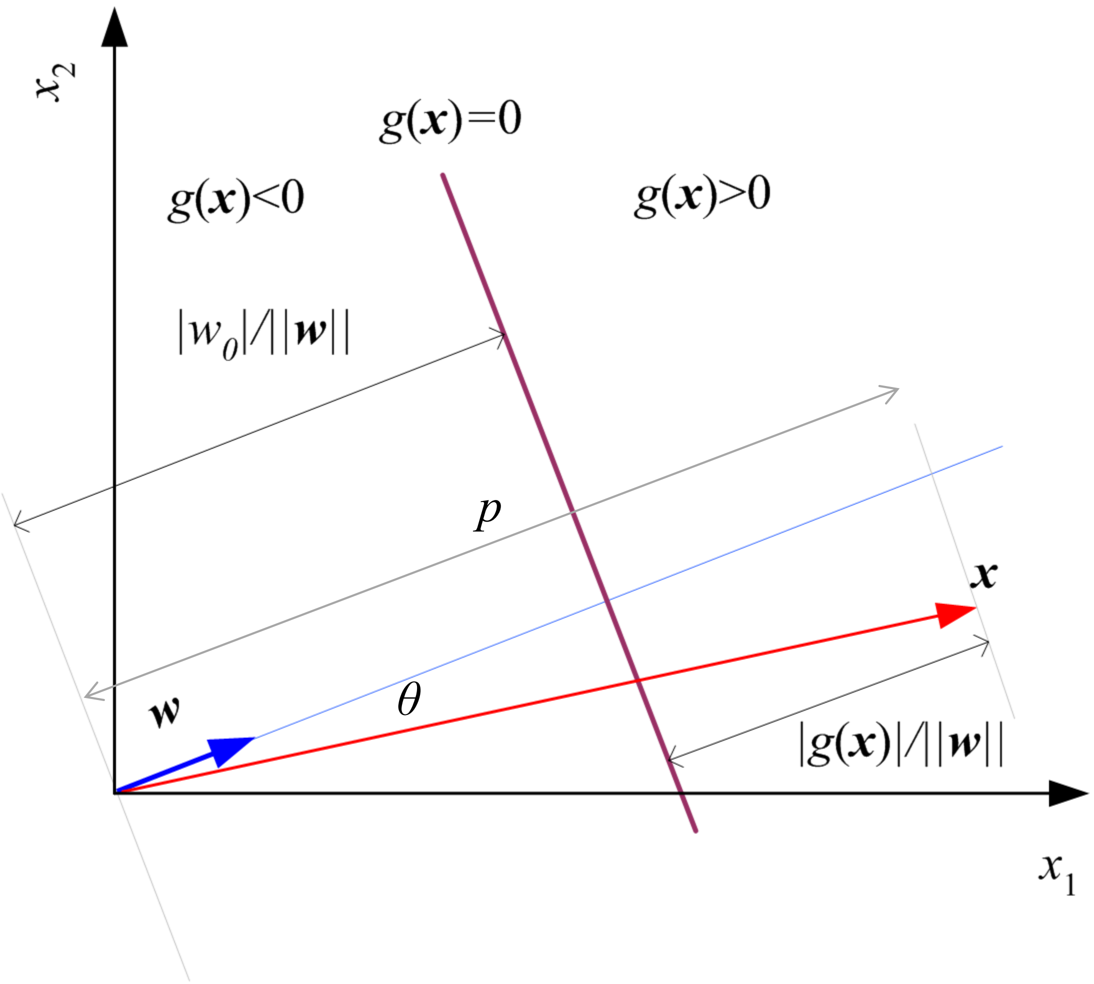
Two Classes



$$
\begin{aligned}
g(\mathbf{x}) &= g_1(\mathbf{x}) - g_2(\mathbf{x}) \\
&= \left(\mathbf{w}_1^T\mathbf{x} + w_{10}\right) - \left(\mathbf{w}_2^T\mathbf{x} + w_{20}\right) \\
&= \left(\mathbf{w}_1 - \mathbf{w}_2\right)^T\mathbf{x} + \left(w_{10} - w_{20}\right) \\
&= \mathbf{w}^T\mathbf{x} + w_0
\end{aligned}
$$

$$
\text{choose} \begin{cases} C_1 & \text{if } g(\mathbf{x}) > 0 \\ C_2 & \text{otherwise} \end{cases}
$$

Geometry



$g(\boldsymbol{x}) > 0$ iff $\boldsymbol{w}^T\boldsymbol{x} + w_0 > 0$

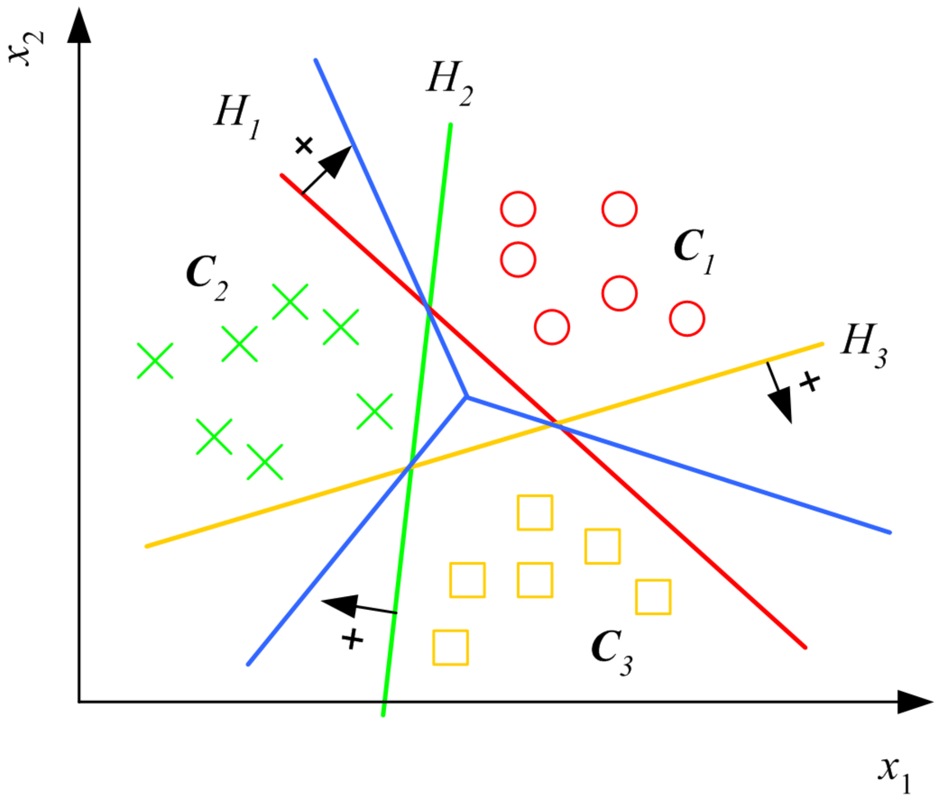
iff $\boldsymbol{w}^T\boldsymbol{x} > -w_0$

iff $\|\boldsymbol{w}\|\|\boldsymbol{x}\| \cos\theta > -w_0$

iff $p \equiv \|\boldsymbol{x}\| \cos\theta > -w_0/\|\boldsymbol{w}\|$

(edited by BJM)

Multiple Classes



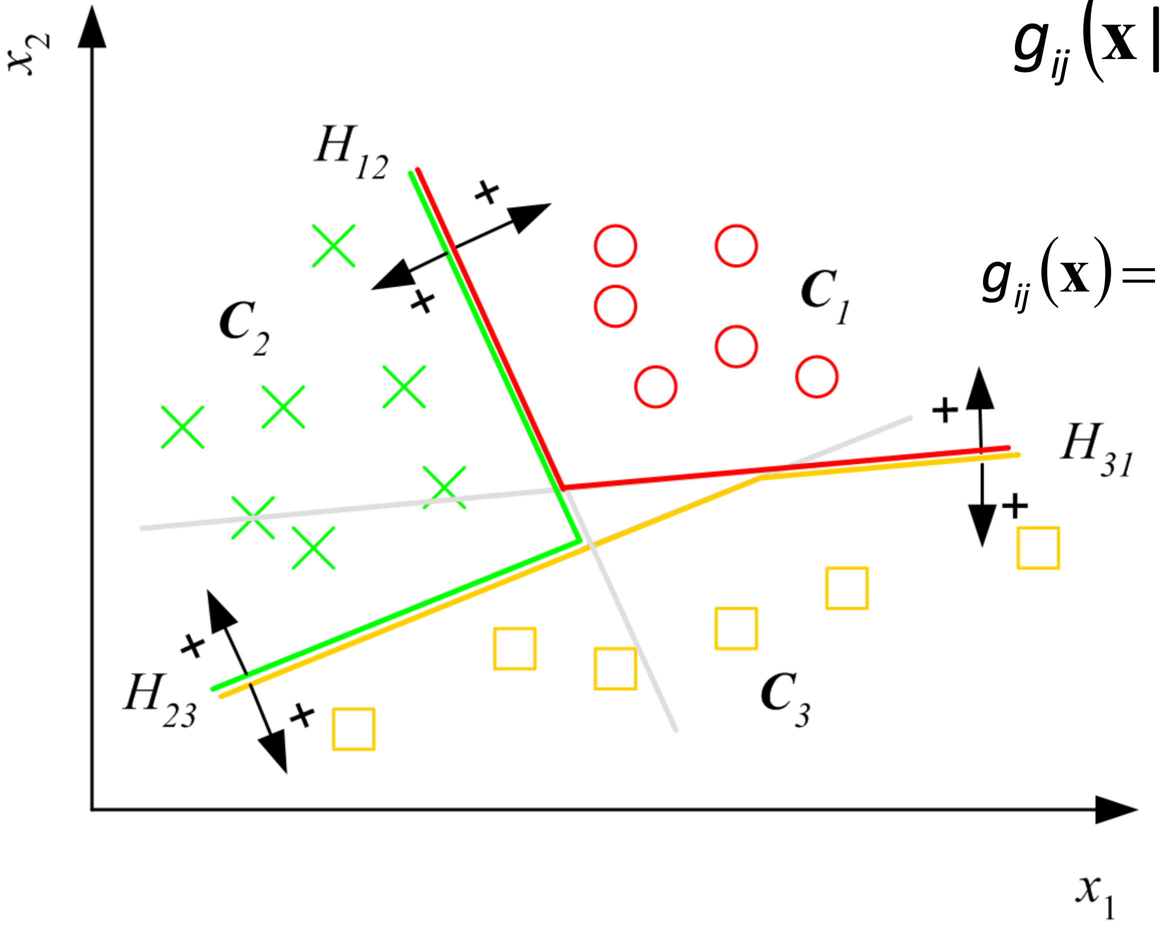
$$g_i(\mathbf{x} \mid \mathbf{w}_i, w_{i0}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

Choose C_i if

$$g_i(\mathbf{x}) = \max_{j=1}^{K} g_j(\mathbf{x})$$

Classes are linearly separable

Pairwise Separation



$$g_{ij}(\mathbf{x}|\mathbf{w}_{ij}, w_{ij0}) = \mathbf{w}_{ij}^T\mathbf{x} + w_{ij0}$$

$$g_{ij}(\mathbf{x}) = \begin{cases} > 0 & \text{if } \mathbf{x} \in C_i \\ \leq 0 & \text{if } \mathbf{x} \in C_j \\ \text{don't care} & \text{otherwise} \end{cases}$$

choose C_i if

$$\forall j \neq i, g_{ij}(\mathbf{x}) > 0$$

From Discriminants to Posteriors

When $p(\mathbf{x} \mid C_i) \sim \mathcal{N}(\boldsymbol{\mu}_i, \Sigma)$

$$g_i(\mathbf{x} \mid \mathbf{w}_i, w_{i0}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

$$\mathbf{w}_i = \Sigma^{-1}\mu_i \quad w_{i0} = -\frac{1}{2}\mu_i^T \Sigma^{-1}\mu_i + \log P(C_i)$$

$$y \equiv P(C_1 \mid \mathbf{x}) \text{ and } P(C_2 \mid \mathbf{x}) = 1 - y$$

$$\text{choose } C_1 \text{ if} \begin{cases} y > 0.5 \\ y/(1-y) > 1 \\ \log[y/(1-y)] > 0 \end{cases} \text{ and } C_2 \text{ otherwise}$$

$$\text{logit}(P(C_1 \mid \mathbf{x})) = \log \frac{P(C_1 \mid \mathbf{x})}{1 - P(C_1 \mid \mathbf{x})} = \log \frac{P(C_1 \mid \mathbf{x})}{P(C_2 \mid \mathbf{x})}$$

$$= \log \frac{p(\mathbf{x} \mid C_1)}{p(\mathbf{x} \mid C_2)} + \log \frac{P(C_1)}{P(C_2)}$$

$$= \log \frac{(2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left[-(1/2)(\mathbf{x} - \mu_1)^T \Sigma^{-1} (\mathbf{x} - \mu_1)\right]}{(2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left[-(1/2)(\mathbf{x} - \mu_2)^T \Sigma^{-1} (\mathbf{x} - \mu_2)\right]} + \log \frac{P(C_1)}{P(C_2)}$$

$$= \mathbf{w}^T \mathbf{x} + w_0$$

where $\mathbf{w} = \Sigma^{-1}(\mu_1 - \mu_2)$ $\quad w_0 = -\frac{1}{2}(\mu_1 + \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) + \log \frac{P(C_1)}{P(C_2)}$

The inverse of logit

$$\log \frac{P(C_1 \mid \mathbf{x})}{1 - P(C_1 \mid \mathbf{x})} = \mathbf{w}^T \mathbf{x} + w_0$$

$$P(C_1 \mid \mathbf{x}) = \text{sigmoid}(\mathbf{w}^T \mathbf{x} + w_0) = \frac{1}{1 + \exp\left[-(\mathbf{w}^T \mathbf{x} + w_0)\right]}$$

$$\log \frac{P(C_1|x)}{1-P(C_1|x)} \equiv \log \frac{P}{1-P} = w^T x + w_0$$

$$\therefore \frac{P}{1-P} = \exp(w^T x + w_0)$$

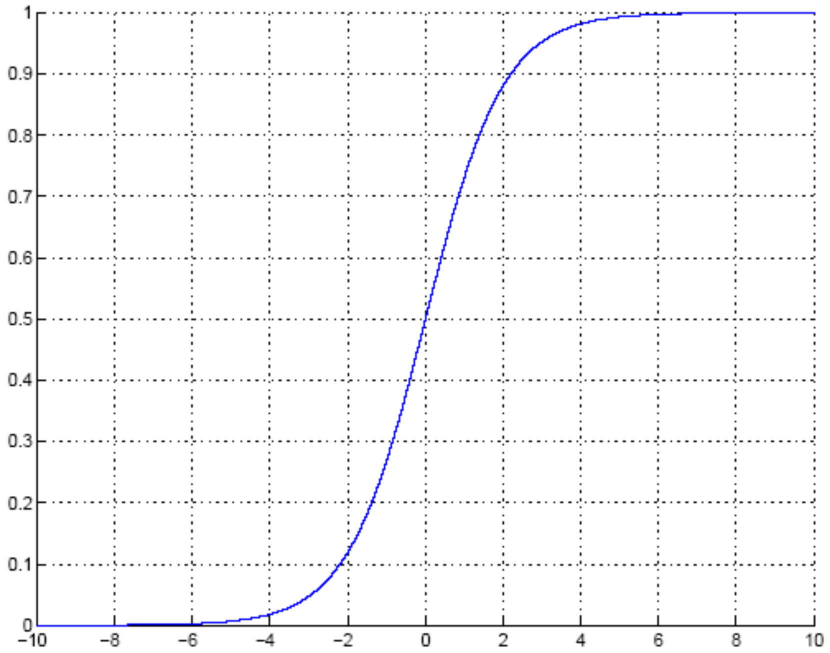
$$P = (1 - P) \exp(w^T x + w_0)$$

$$= \exp(w^T x + w_0) - P \exp(w^T x + w_0)$$

$$P[1 + \exp(w^T x + w_0)] = \exp\,(w^T x + w_0)$$

$$P = \frac{\exp(w^T x + w_0)}{1 + \exp(w^T x + w_0)} = \frac{1}{1 + \exp[-(w^T x + w_0)]}$$

Sigmoid (Logistic) Function



Calculate $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ and choose C_1 if $g(\mathbf{x}) > 0$, or

Calculate $y = \text{sigmoid}\left(\mathbf{w}^T\mathbf{x} + w_0\right)$ and choose C_1 if $y > 0.5$

Gradient-Descent

- $E(\mathbf{w} | \mathcal{X})$ is error with parameters $\mathbf{w}$ on sample $\mathcal{X}$

  $$\mathbf{w}^* = \arg\min_{\mathbf{w}} E(\mathbf{w} \mid \mathcal{X})$$

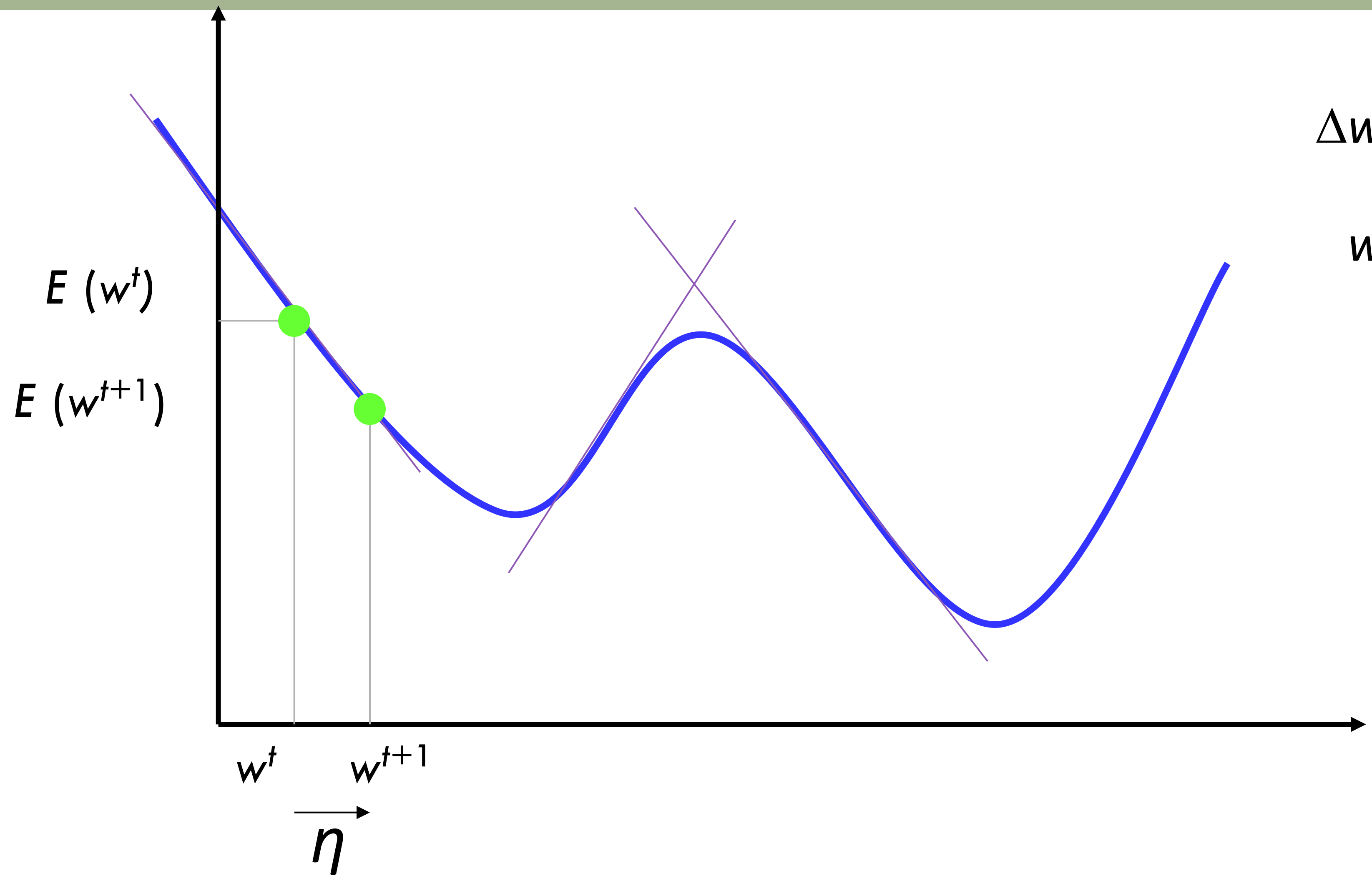
- Gradient

  $$\nabla_w E = \left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \ldots, \frac{\partial E}{\partial w_d} \right]^T$$

- Gradient-descent:

  Starts from random $\mathbf{w}$ and updates $\mathbf{w}$ iteratively in the negative direction of gradient (steepest descent)

Gradient-Descent



$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}, \forall i$$

$$w_i = w_i + \Delta w_i$$

Logistic Discrimination

Two classes: Assume log likelihood ratio is linear

$$\log\frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} = \mathbf{w}^T\mathbf{x} + w_0^o$$

$$\operatorname{logit}(P(C_1|\mathbf{x})) = \log\frac{P(C_1|\mathbf{x})}{1-P(C_1|\mathbf{x})} = \log\frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} + \log\frac{P(C_1)}{P(C_2)}$$

$$= \mathbf{w}^T\mathbf{x} + w_0$$

$$\text{where } w_0 = w_0^o + \log\frac{P(C_1)}{P(C_2)}$$

$$y = \hat{P}(C_1|\mathbf{x}) = \frac{1}{1+\exp\left[-\left(\mathbf{w}^T\mathbf{x} + w_0\right)\right]}$$

Training: Two Classes

$$\mathcal{X} = \{\mathbf{x}^t, r^t\}_t \quad r^t \mid \mathbf{x}^t \sim \text{Bernoulli}(y^t)$$

$$y = P(C_1 \mid \mathbf{x}) = \frac{1}{1 + \exp\left[-(\mathbf{w}^T \mathbf{x} + w_0)\right]}$$

$$l(\mathbf{w}, w_0 \mid \mathcal{X}) = \prod_t (y^t)^{(r^t)} (1 - y^t)^{(1 - r^t)}$$

$$E = -\log l$$

$$E(\mathbf{w}, w_0 \mid \mathcal{X}) = -\sum_t r^t \log y^t + (1 - r^t) \log (1 - y^t)$$

CROSS ENTROPY

- The optimal code for signal s_k with probability q_k has length $-\lg q_k$ bits
- The average length is the entropy $H(q) = E_q[-\lg q_k] = -\sum_k q_k \lg q_k$ bits
- But suppose the actual probabilities are p_k
- Then the average length is given by the <u>cross-entropy</u>:
  $H(p \parallel q) = E_p[-\lg q_k] = -\sum_k p_k \lg q_k$ bits
- The cross-entropy is minimized when the probability distributions are equal:
  $H(p \parallel q) = H(p) = H(q)$

Training: Gradient-Descent

$$E(\mathbf{w}, w_0 | \mathcal{X}) = -\sum_t r^t \log y^t + \left(1 - r^t\right) \log\left(1 - y^t\right)$$

$$\text{If } y = \text{sigmoid}(a) \quad \frac{dy}{da} = y(1-y)$$

$$\Delta w_j = -\eta \frac{\partial E}{\partial w_j} = \eta \sum_t \left(\frac{r^t}{y^t} - \frac{1-r^t}{1-y^t} \right) y^t \left(1 - y^t\right) x_j^t$$

$$= \eta \sum_t \left(r^t - y^t\right) x_j^t, \, j = 1, \ldots, d$$

$$\Delta w_0 = -\eta \frac{\partial E}{\partial w_0} = \eta \sum_t \left(r^t - y^t\right)$$

$$\frac{\partial}{\partial w_j} r^t \log y^t = \frac{r^t}{y^t} \frac{\partial y^t}{\partial w_j}$$

$$= \frac{r^t}{y^t} \frac{\partial}{\partial w_j} \text{sigmoid}(a)$$

$$= \frac{r^t}{y^t} y^t (1 - y^t) \frac{\partial a}{\partial w_j}$$

$$= \frac{r^t}{y^t} y^t (1 - y^t) \frac{\partial (\mathbf{w}^T \mathbf{x}^t + w_0)}{\partial w_j}$$

$$= \frac{r^t}{y^t} y^t (1 - y^t) \mathbf{x}_j^t$$

(edited by BJM)

For $j = 0, \ldots, d$

$\quad w_j \leftarrow$rand(-0.01,0.01)

Repeat

$\quad$ For $j = 0, \ldots, d$

$\quad\quad \Delta w_j \leftarrow 0$

$\quad$ For $t = 1, \ldots, N$

$\quad\quad o \leftarrow 0$

$\quad\quad$ For $j = 0, \ldots, d$

$\quad\quad\quad o \leftarrow o + w_j x_j^t$

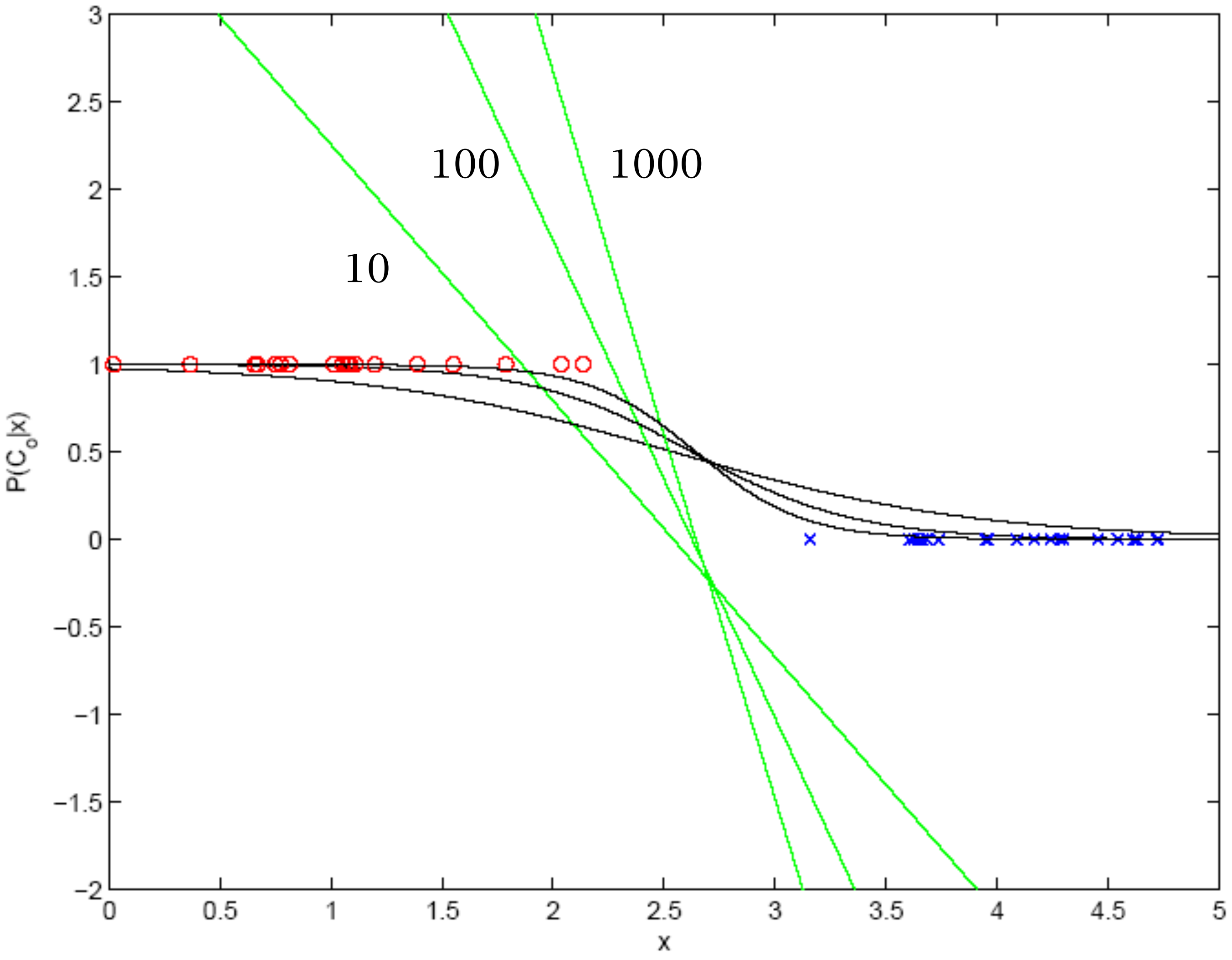
$\quad\quad y \leftarrow \mathrm{sigmoid}(o)$

$\quad\quad \Delta w_j \leftarrow \Delta w_j + (r^t - y) x_j^t$

$\quad$ For $j = 0, \ldots, d$

$\quad\quad w_j \leftarrow w_j + \eta \Delta w_j$

Until convergence



100
1000
10
P(C_o|x)
x

$K>2$ Classes

$$\mathcal{X} = \{\mathbf{x}^t, \mathbf{r}^t\}_t \quad r^t \mid \mathbf{x}^t \sim \text{Mult}_K\left(1, \mathbf{y}^t\right)$$

$$\log \frac{p(\mathbf{x} \mid C_i)}{p(\mathbf{x} \mid C_K)} = \mathbf{w}_i^T \mathbf{x} + w_{i0}^o$$

$$y = \hat{P}(C_i \mid \mathbf{x}) = \boxed{\frac{\exp\left[\mathbf{w}_i^T \mathbf{x} + w_{i0}\right]}{\sum_{j=1}^{K} \exp\left[\mathbf{w}_j^T \mathbf{x} + w_{j0}\right]}}, i = 1, \ldots, K$$

softmax

$$l\left(\{\mathbf{w}_i, w_{i0}\}_i \mid \mathcal{X}\right) = \prod_t \prod_i \left(y_i^t\right)^{\left(r_i^t\right)}$$

$$E\left(\{\mathbf{w}_i, w_{i0}\}_i \mid \mathcal{X}\right) = \boxed{-\sum_t r_i^t \log y_i^t}$$

cross-entropy

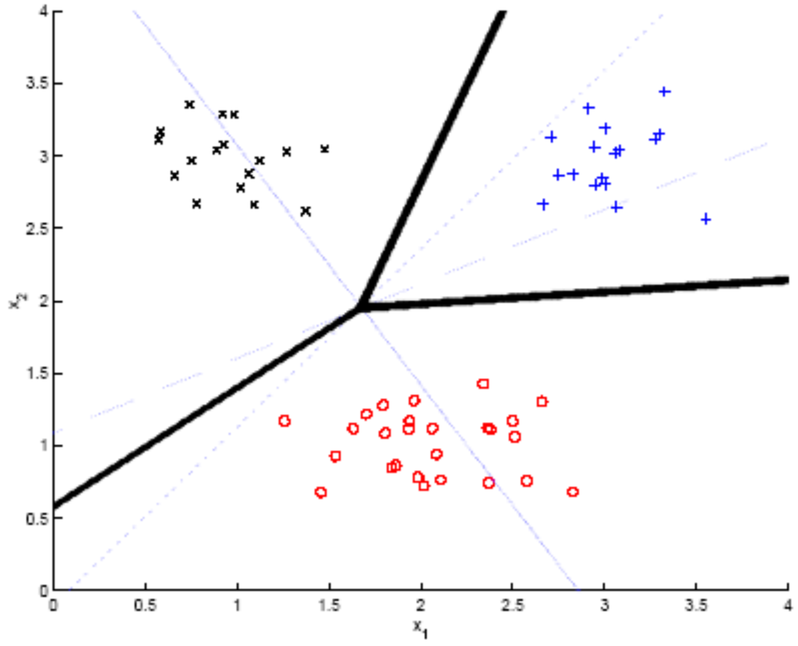
$$\Delta \mathbf{w}_j = \eta \sum_t \left(r_j^t - y_j^t\right) \mathbf{x}^t \quad \Delta w_{j0} = \eta \sum_t \left(r_j^t - y_j^t\right)$$

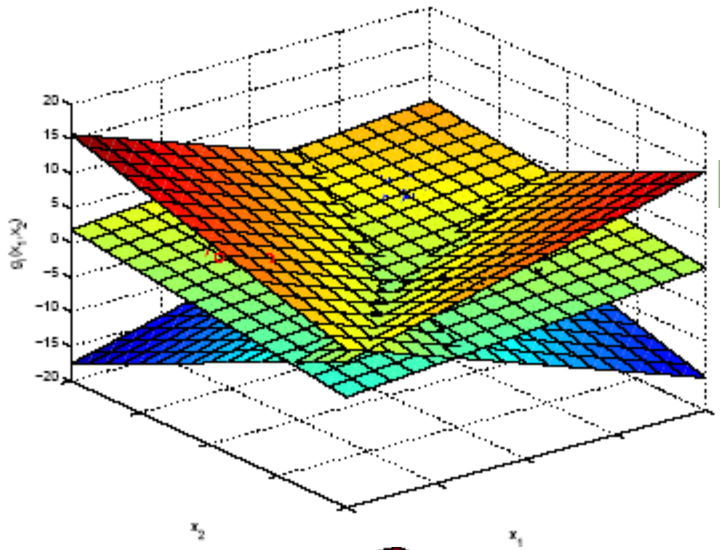
For $i = 1, \ldots, K$, For $j = 0, \ldots, d$, $w_{ij} \leftarrow \text{rand}(-0.01, 0.01)$
Repeat
    For $i = 1, \ldots, K$, For $j = 0, \ldots, d$, $\Delta w_{ij} \leftarrow 0$
    For $t = 1, \ldots, N$
        For $i = 1, \ldots, K$
            $o_i \leftarrow 0$
            For $j = 0, \ldots, d$
                $o_i \leftarrow o_i + w_{ij} x_j^t$
        For $i = 1, \ldots, K$
            $y_i \leftarrow \exp(o_i) / \sum_k \exp(o_k)$
        For $i = 1, \ldots, K$
            For $j = 0, \ldots, d$
                $\Delta w_{ij} \leftarrow \Delta w_{ij} + (r_i^t - y_i) x_j^t$
    For $i = 1, \ldots, K$
        For $j = 0, \ldots, d$
            $w_{ij} \leftarrow w_{ij} + \eta \Delta w_{ij}$
Until convergence

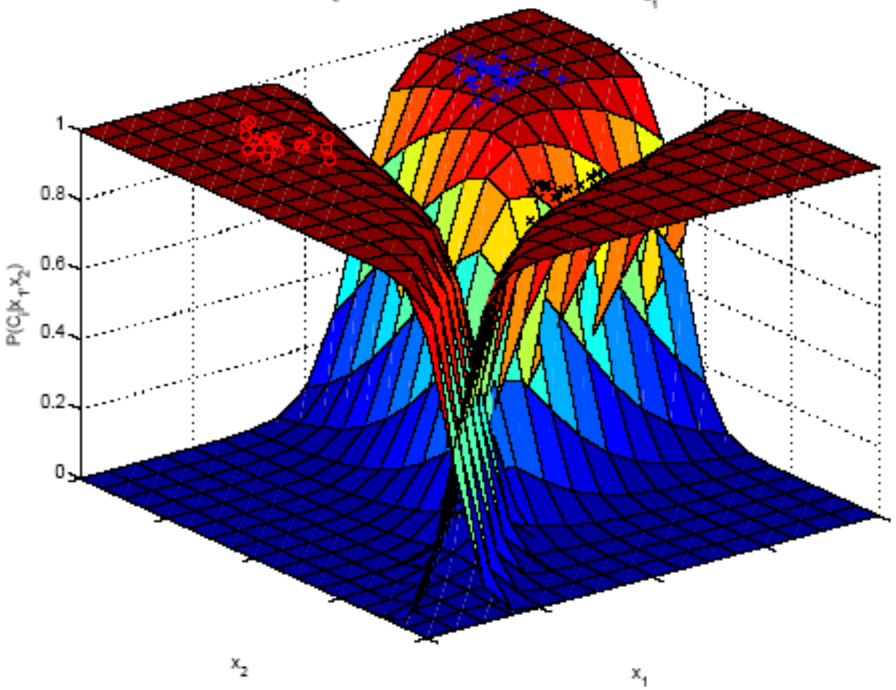
$\mathbf{W} \leftarrow \text{rand}_{K\times d}(-0.01, +0.01)$
Repeat
  $\Delta\mathbf{W} \leftarrow \text{zeroes}(K, d)$
  For $t = 1, \dots, N$
    $\mathbf{o} \leftarrow \mathbf{W}\mathbf{x}^t$
    $\mathbf{z} \leftarrow \exp(\mathbf{o})$ (component-wise)
    $\mathbf{y} \leftarrow \mathbf{z}/\text{sum}(\mathbf{z})$
    $\Delta\mathbf{W} \leftarrow (\mathbf{r}^t - \mathbf{y})(\mathbf{x}^t)^T$
  $\mathbf{W} \leftarrow \mathbf{W} + \Delta\mathbf{W}$
until convergence

Given $\mathbf{X}$ $(N\times d)$ and $\mathbf{R}$ $(N\times K)$
$\mathbf{W} \leftarrow \text{rand}_{K\times d}(-0.01, +0.01)$
Repeat
  $\mathbf{O} \leftarrow \mathbf{X}\mathbf{W}^T$ $(N\times k)$
  $\mathbf{Z} \leftarrow \exp(\mathbf{O})$ (component-wise)
  $\mathbf{S} \leftarrow \mathbf{Z}\mathbf{1}_{N\times k}$ $(N\times k)$
  $\mathbf{Y} \leftarrow \mathbf{Z}/\mathbf{S}$ (component-wise)
  $\Delta\mathbf{W} \leftarrow (\mathbf{R} - \mathbf{Y})^T\mathbf{X}$ $(K\times d)$
  $\mathbf{W} \leftarrow \mathbf{W} + \Delta\mathbf{W}$
until convergence

Example







Generalizing the Linear Model

- Quadratic:

$$\log \frac{p(\mathbf{x}|C_i)}{p(\mathbf{x}|C_K)} = \mathbf{x}^T \mathbf{W}_i \mathbf{x} + \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

- Sum of basis functions:

$$\log \frac{p(\mathbf{x}|C_i)}{p(\mathbf{x}|C_K)} = \mathbf{w}_i^T \phi(\mathbf{x}) + w_{i0}$$

  where $\varphi(\mathbf{x})$ are basis functions. Examples:
  - Hidden units in neural networks (Chapters 11 and 12)
  - Kernels in SVM (Chapter 13)

Discrimination by Regression

- Classes are NOT mutually exclusive and exhaustive

$$r^t = y^t + \varepsilon \text{ where } \varepsilon \sim \mathcal{N}(0, \sigma^2)$$

$$y^t = \text{sigmoid}(\mathbf{w}^T \mathbf{x}^t + w_0) = \frac{1}{1 + \exp[-(\mathbf{w}^T \mathbf{x}^t + w_0)]}$$

$$l(\mathbf{w}, w_0 \mid \mathcal{X}) = \prod_t \frac{1}{\sqrt{2\pi\sigma}} \exp\left[-\frac{(r^t - y^t)^2}{2\sigma^2}\right]$$

$$E(\mathbf{w}, w_0 \mid \mathcal{X}) = \frac{1}{2} \sum_t (r^t - y^t)^2$$

$$\Delta \mathbf{w} = \eta \sum_t (r^t - y^t) y^t (1 - y^t) \mathbf{x}^t$$

Learning to Rank

- Ranking: A different problem than classification or regression
- Let us say $\mathbf{x}^u$ and $\mathbf{x}^v$ are two instances, e.g., two movies

  We prefer u to v implies that $g(\mathbf{x}^u) > g(\mathbf{x}^v)$

  where $g(\mathbf{x})$ is a <u>score function</u>, here linear:

  $$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$$

- Find a direction $\mathbf{w}$ such that we get the desired ranks when instances are projected along $\mathbf{w}$

Ranking Error

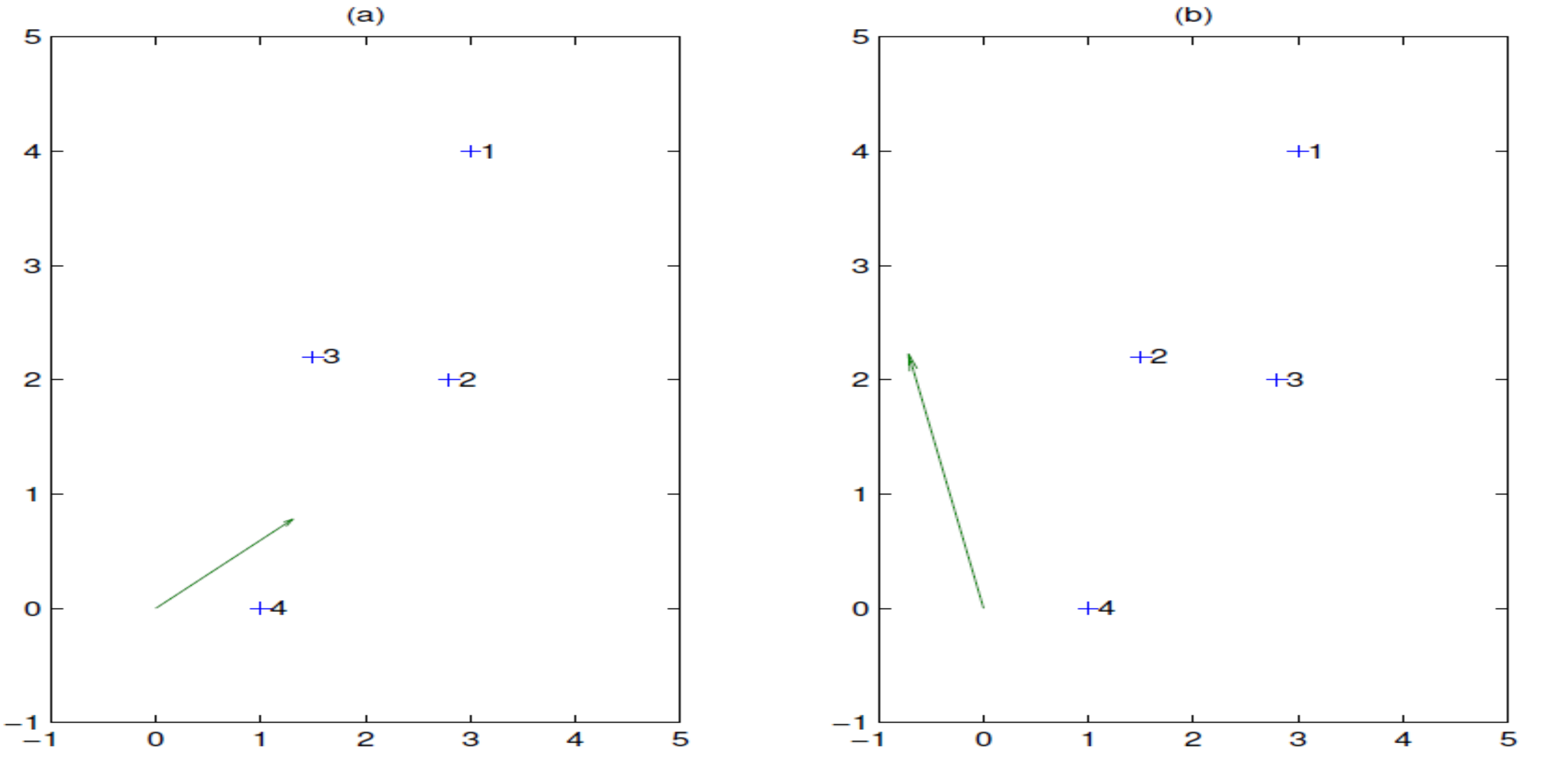
- We prefer *u* to *v* implies that $g(\boldsymbol{x}^u) > g(\boldsymbol{x}^v)$, so error is $g(\boldsymbol{x}^v) - g(\boldsymbol{x}^u)$, if $g(\boldsymbol{x}^u) < g(\boldsymbol{x}^v)$

$$E(\mathbf{w}|\{r^u, r^v\}) = \sum_{r^u \prec r^v} [g(\mathbf{x}^v|\theta) - g(\mathbf{x}^u|\theta)]_+$$

where a_+ is equal to a if $a \geq 0$ and 0 otherwise.

- Linear case: $E(\boldsymbol{w}|\{r^u, r^v\}) = \sum_{r^u \prec r^v} [\boldsymbol{w}^T(\boldsymbol{x}^v - \boldsymbol{x}^u)]_+$
- Gradient descent update when ranked wrong:
  $\Delta w_j = -\eta\left(x_j^v - x_j^u\right), j = 1, \ldots, d$

(edited by BJM)


(a)
(b)
+1
+2
+3
+4
5
4
3
2
1
0
−1
−1
0
1
2
3
4
5

READ ALPAYDIN CH. 11