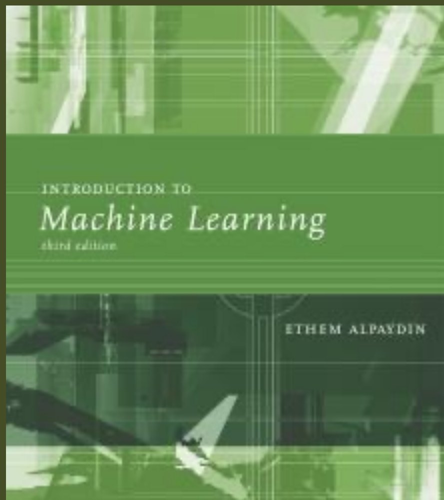


Lecture Slides for

INTRODUCTION TO MACHİNE LEARNİNG

3RD EDİTİON

ETHEM ALPAYDIN

CHAPTER 10:

LİNEAR DİSCRİMİNATİON

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:

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

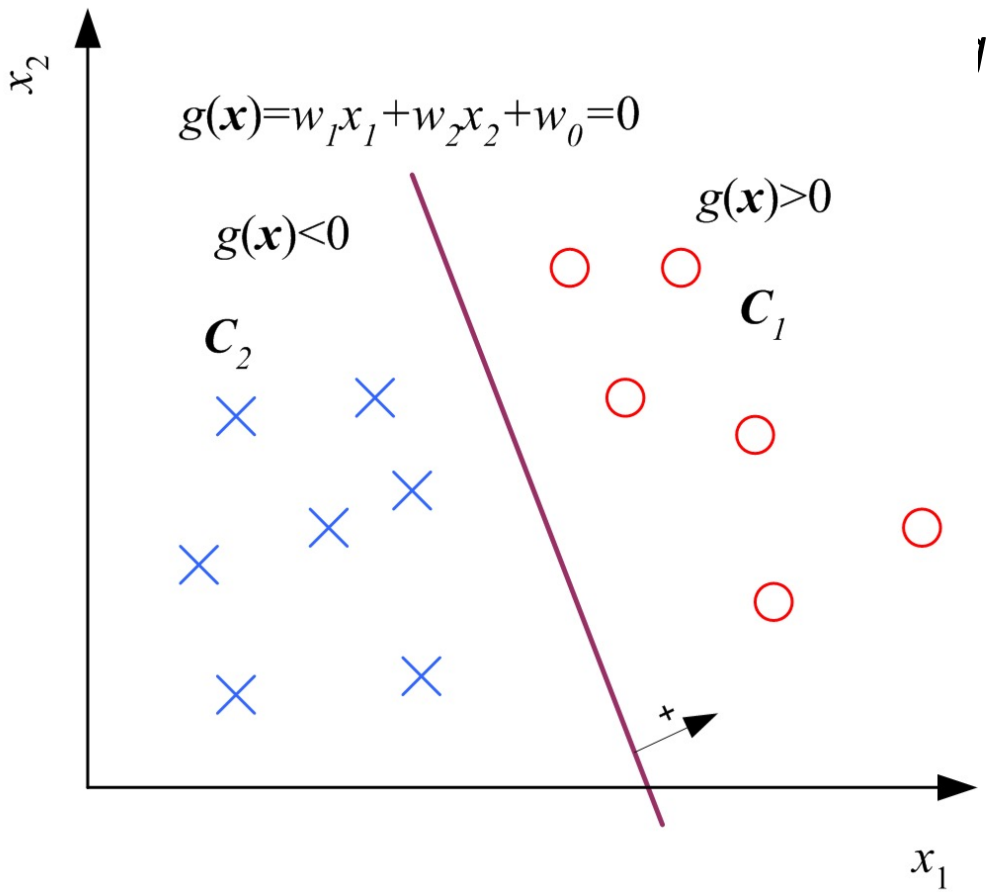
- Higher-order (product) terms:

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

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

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

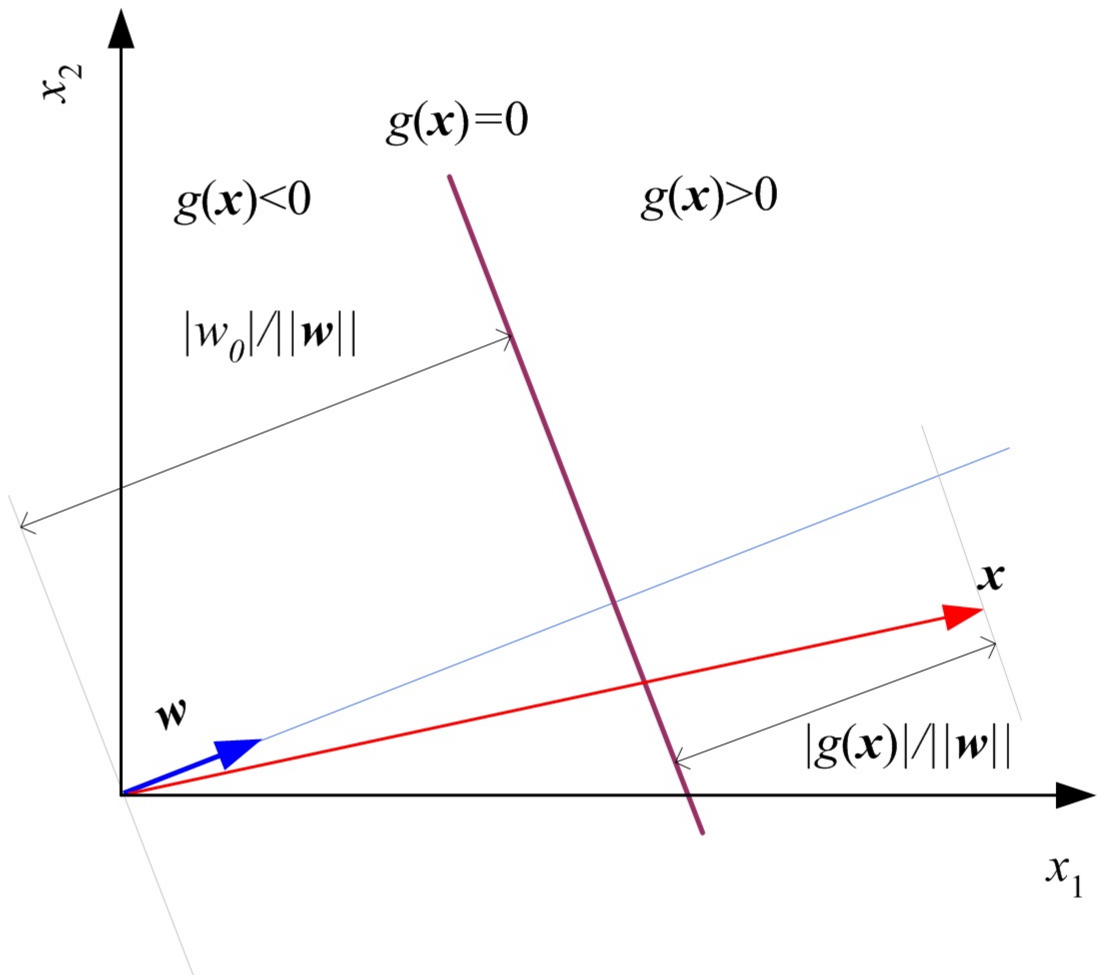
Two Classes



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

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

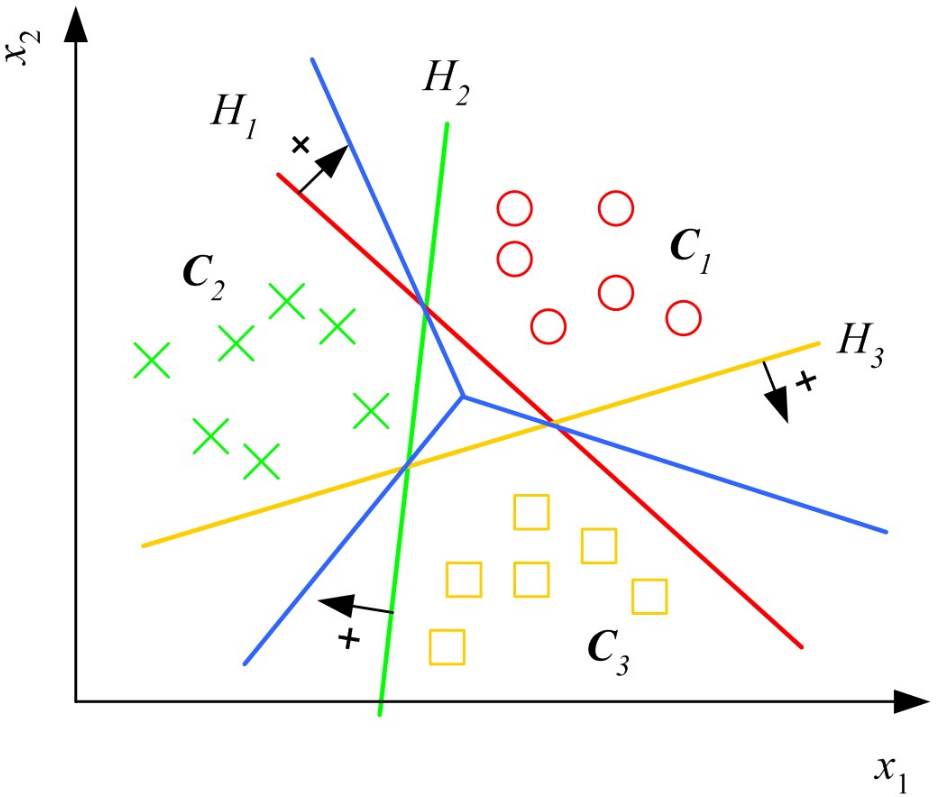
Geometry



x_2
$g(\boldsymbol{x})=0$
$g(\boldsymbol{x})<0$
$g(\boldsymbol{x})>0$
$|w_0|/||\boldsymbol{w}||$
$\boldsymbol{x}$
$\boldsymbol{w}$
$|g(\boldsymbol{x})|/||\boldsymbol{w}||$
x_1

Multiple Classes

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

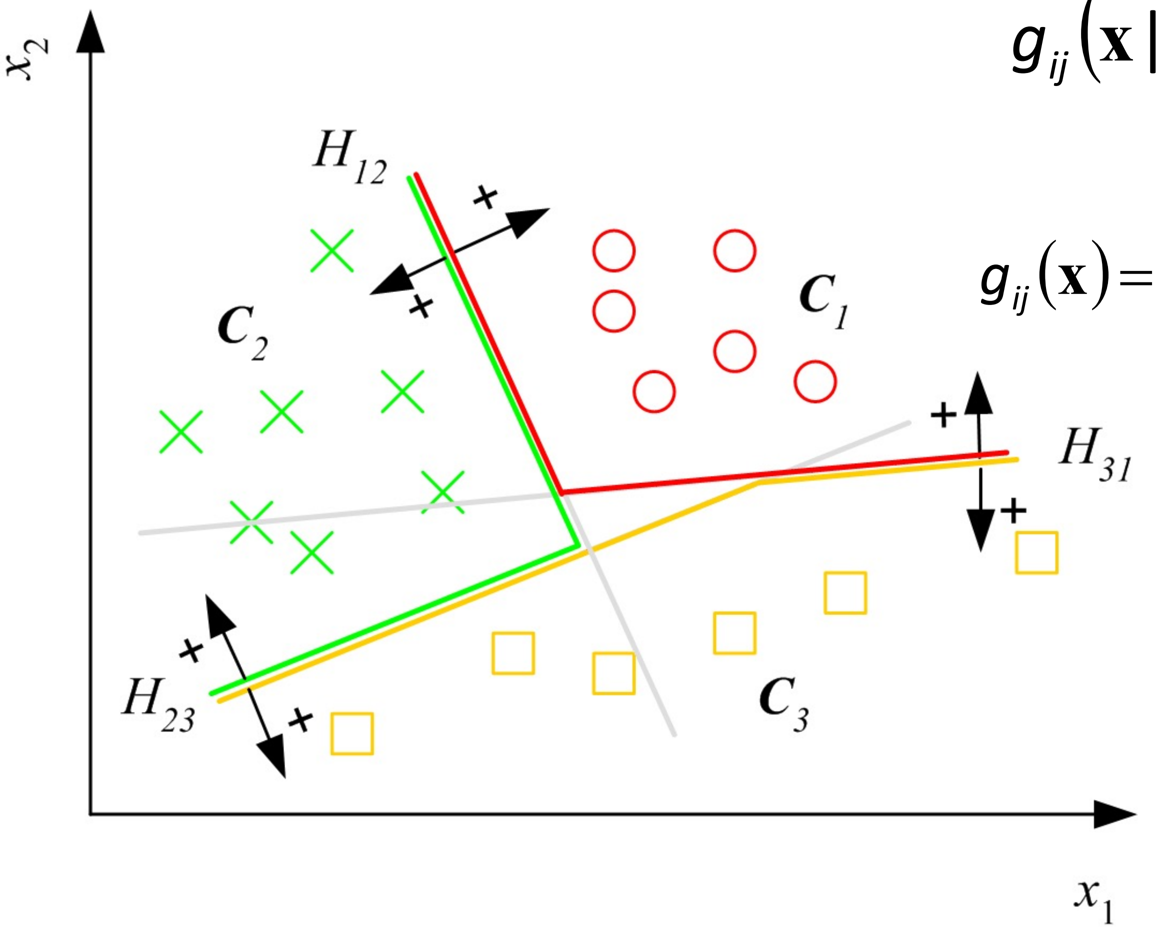


Choose C_i if

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

Classes are
linearly separable

Pairwise Separation



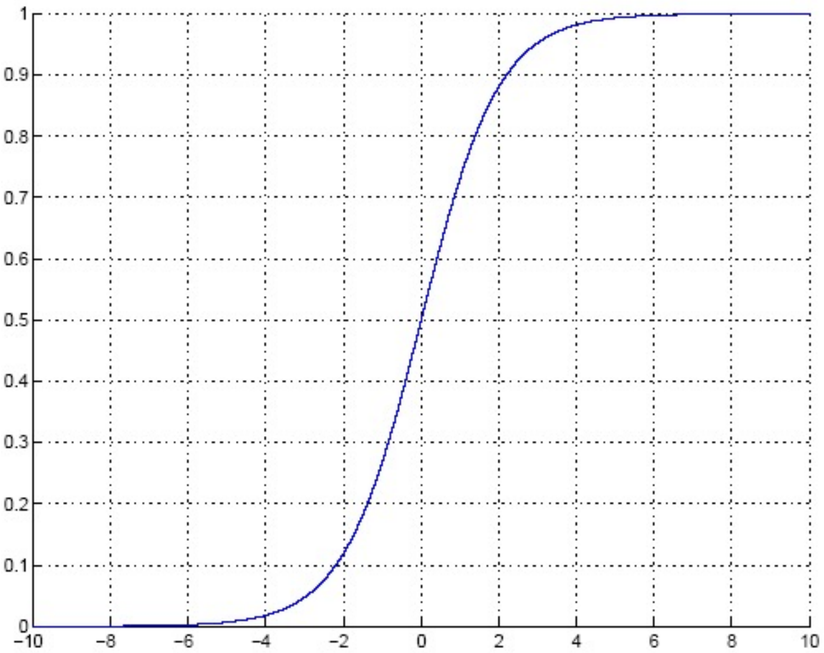
$$g_{ij}\left(\mathbf{x} \mid \mathbf{w}_{ij}, w_{ij0}\right) = \mathbf{w}_{ij}^T \mathbf{x} + w_{ij0}$$

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

choose C_i if

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

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

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

- Gradient

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

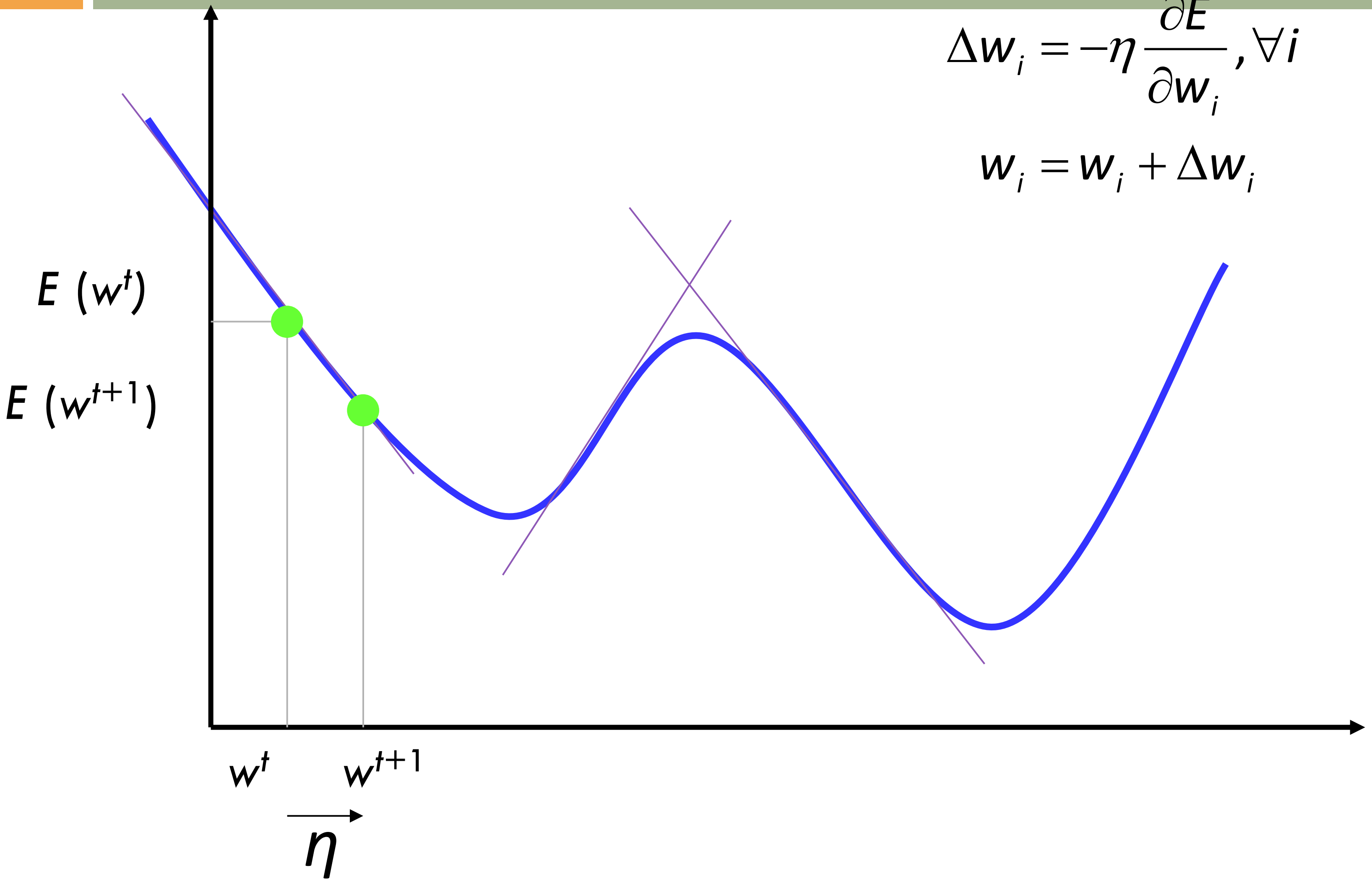
- Gradient-descent:

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

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

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

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

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

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

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

Training: Two Classes

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

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

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

$$E = -\log l$$

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

Training: Gradient-Descent

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

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

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

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

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

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 $j = 0, \ldots, d$

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

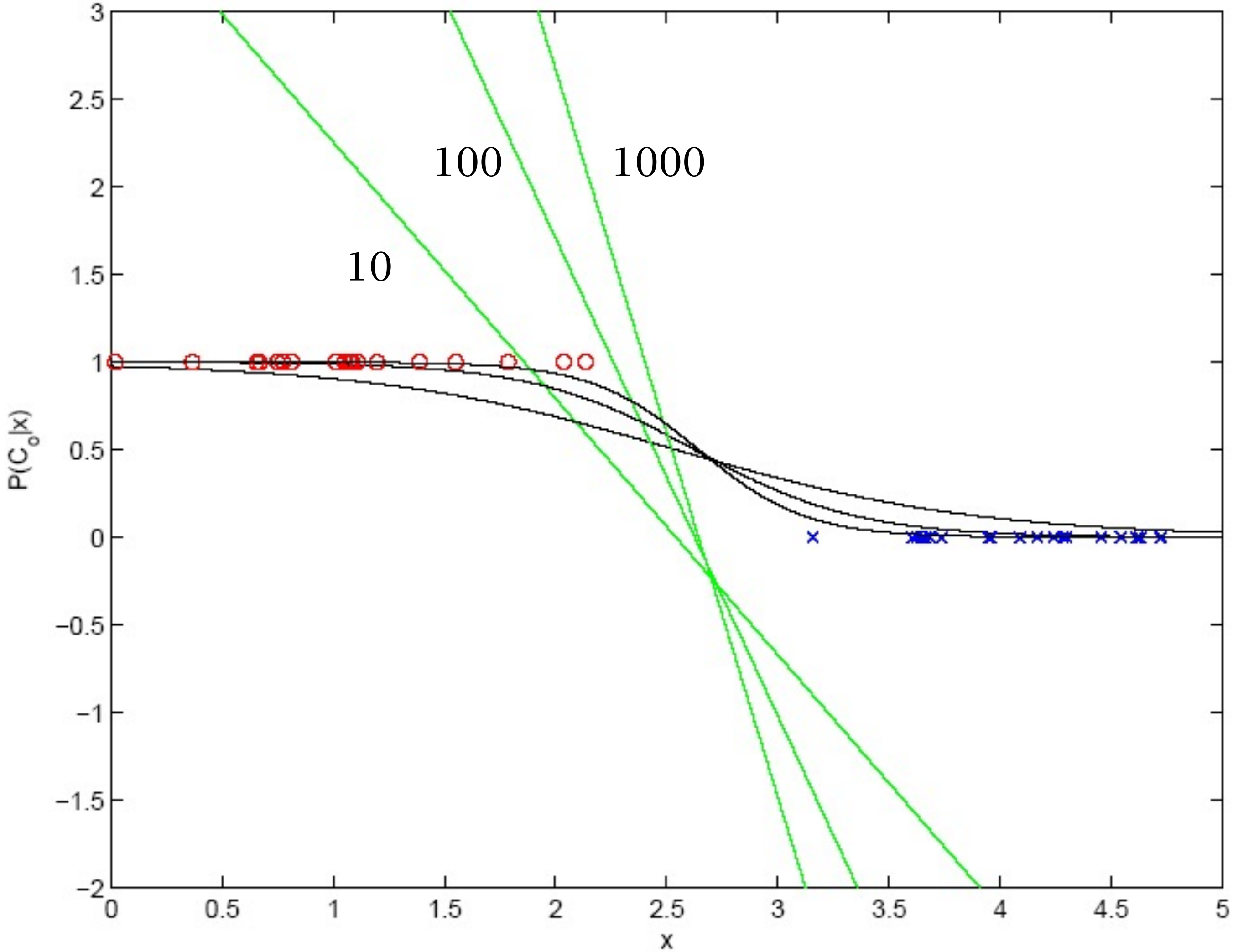
$\quad\quad y \leftarrow \mathrm{sigmoid}(o)$

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

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

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

Until convergence



3
2.5
2
1.5
1
0.5
0
−0.5
−1
−1.5
−2
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
x
$P(C_o|x)$
10
100
1000

K>2 Classes

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

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

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

softmax

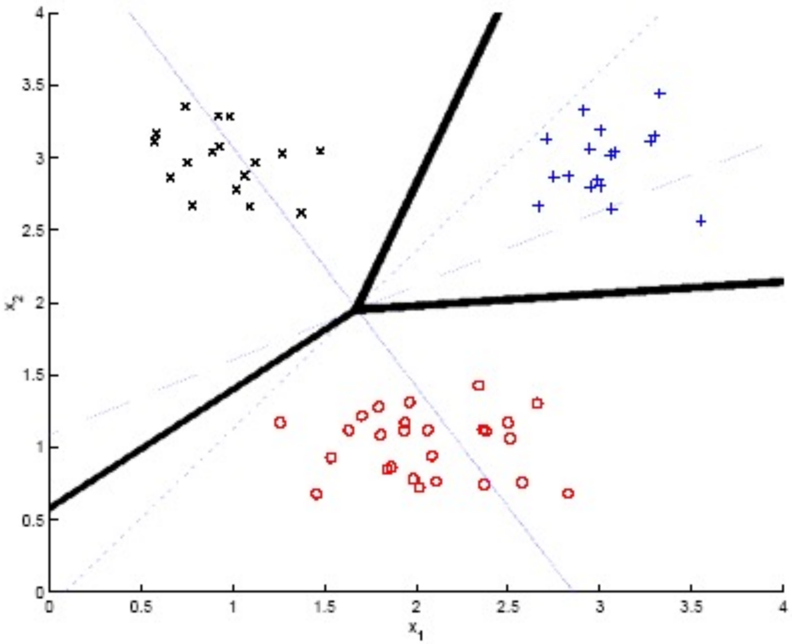
$$l\left(\{\mathbf{w}_i, w_{i0}\}_i \mid \mathcal{X}\right) = \prod_t \prod_i \left(y_i^t\right)^{\left(r_i^t\right)}$$

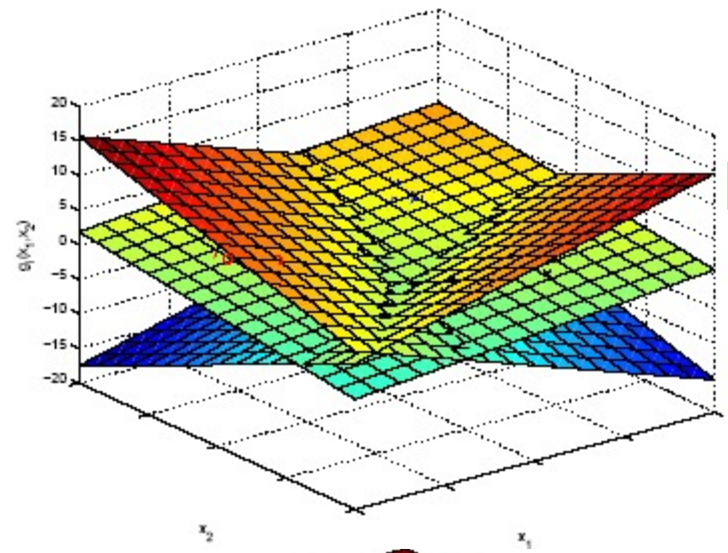
$$E\left(\{\mathbf{w}_i, w_{i0}\}_i \mid \mathcal{X}\right) = -\sum_t r_i^t \log y_i^t$$

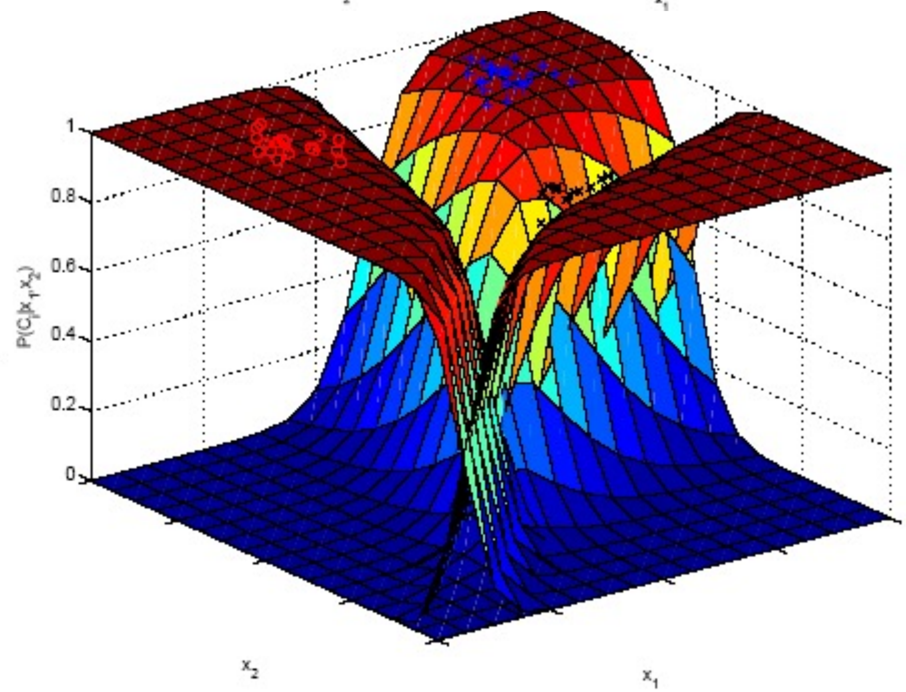
$$\Delta \mathbf{w}_j = \eta \sum_t \left(r_j^t - y_j^t\right) \mathbf{x}^t \quad \Delta w_{j0} = \eta \sum_t \left(r_j^t - y_j^t\right)$$

```
For $i = 1, \ldots, K$, For $j = 0, \ldots, d$, $w_{ij} \leftarrow$ 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
```

Example


0
0.5
1
1.5
2
2.5
3
3.5
4
x_1
x_2


20
15
10
5
0
-5
-10
-15
-20
$g_i(x_1,x_2)$
x_2
x_1


1
0.8
0.6
0.4
0.2
0
$P(C_i|x_1,x_2)$
x_2
x_1

Generalizing the Linear Model

- Quadratic:

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

- Sum of basis functions:

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

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