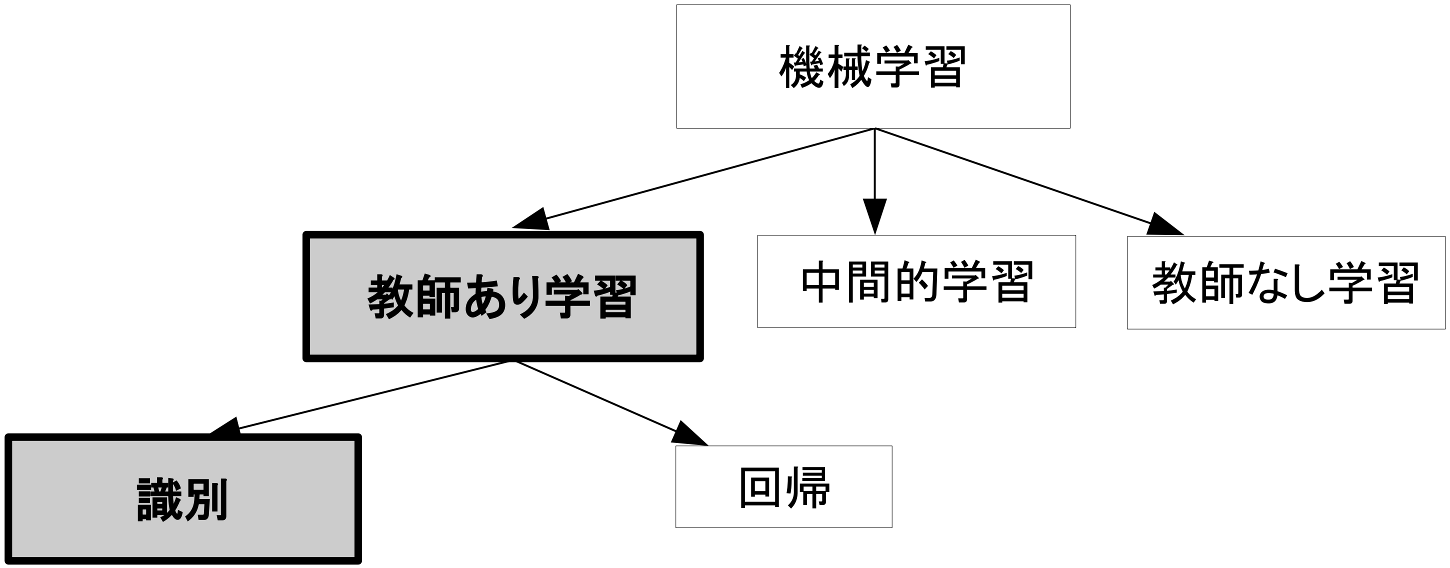
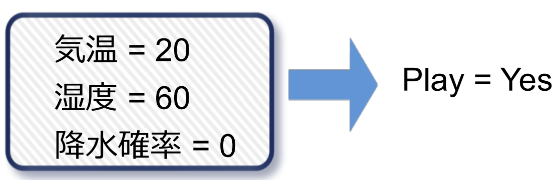
ID: 0501

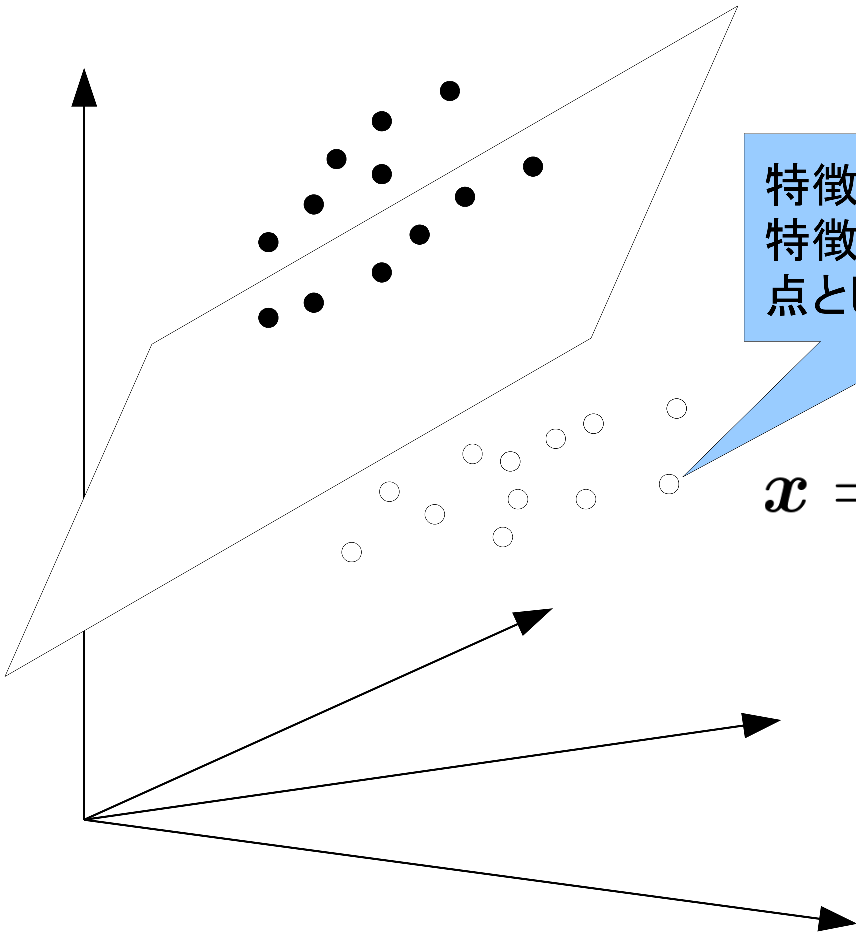
5. 識別　ー生成モデルと識別モデルー



- ラベル特徴
- 数値特徴



5.1 数値特徴に対する「教師あり・識別」問題の定義



特徴ベクトルは、
特徴空間上の
点として表現される

$$\boldsymbol{x} = (x_1, \ldots, x_d)^T$$

比較的クラスが分離している
→ 小数のパラメータで各クラスを表現可能
⇒ **統計的手法**

クラス境界が複雑
非線形識別面 ⇒ **ニューラルネット**
高次元へマッピング⇒ **SVM**

ID: 0503

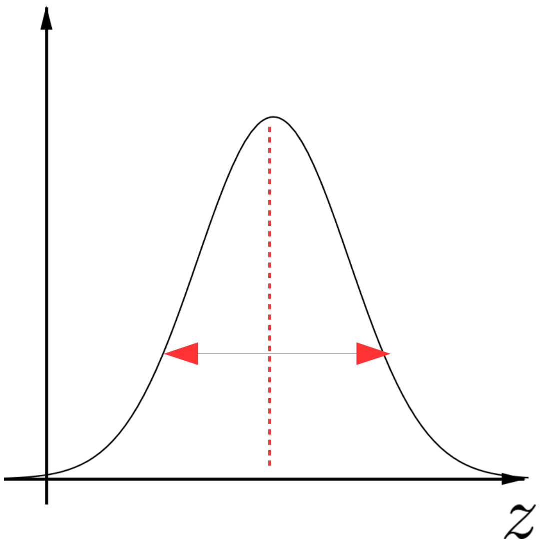
5.2 生成モデル

5.2.1 数値特徴に対するナイーブベイズ識別

$$C_{NB} = \arg\max_i P(\omega_i) \prod_{j=1}^{d} p(x_j|\omega_i)$$

- 確率密度関数 $p(x_j|\omega_i)$ の推定
  - 正規分布を仮定

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(z-\mu)^2}{2\sigma^2})$$



  - 平均 μ と分散 σ を最尤推定
    - それぞれ、学習データの平均と分散になる

5.2.2 生成モデルの考え方

- 事後確率を求めるにあたって、同時確率を求めている
  - データが生成される様子をモデル化していると見ることも出来る
    - 事前確率に基づいてクラスを選ぶ
    - そのもとで、特徴ベクトルを出力する

$$P(\omega_i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_i)P(\omega_i)}{p(\boldsymbol{x})}$$

$$= \frac{p(\omega_i, \boldsymbol{x})}{p(\boldsymbol{x})}$$

事後確率を求めるより、難しい問題を解いているのではないか？

5.3 識別モデル

- 事後確率を直接求める

正例で大きな値

負例で大きな値

あまり関係ない

$$\boldsymbol{x} = (x_1, \quad x_2, \quad x_3, \quad \ldots, \quad x_d)$$

w_1 正の大きな値

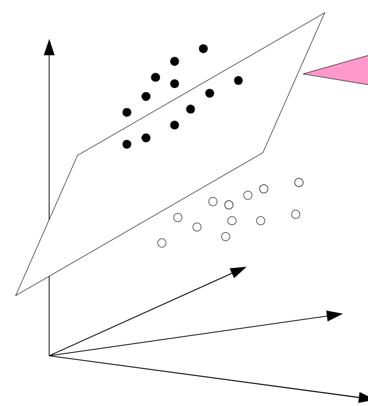
w_2 負の大きな値

w_3 小さな値

正例／負例の閾値

$$w_1 x_1 + w_2 x_2 + w_3 x_3 + \cdots + w_d x_d + w_0$$

この値が正なら正例、負なら負例となるように重み w を学習する



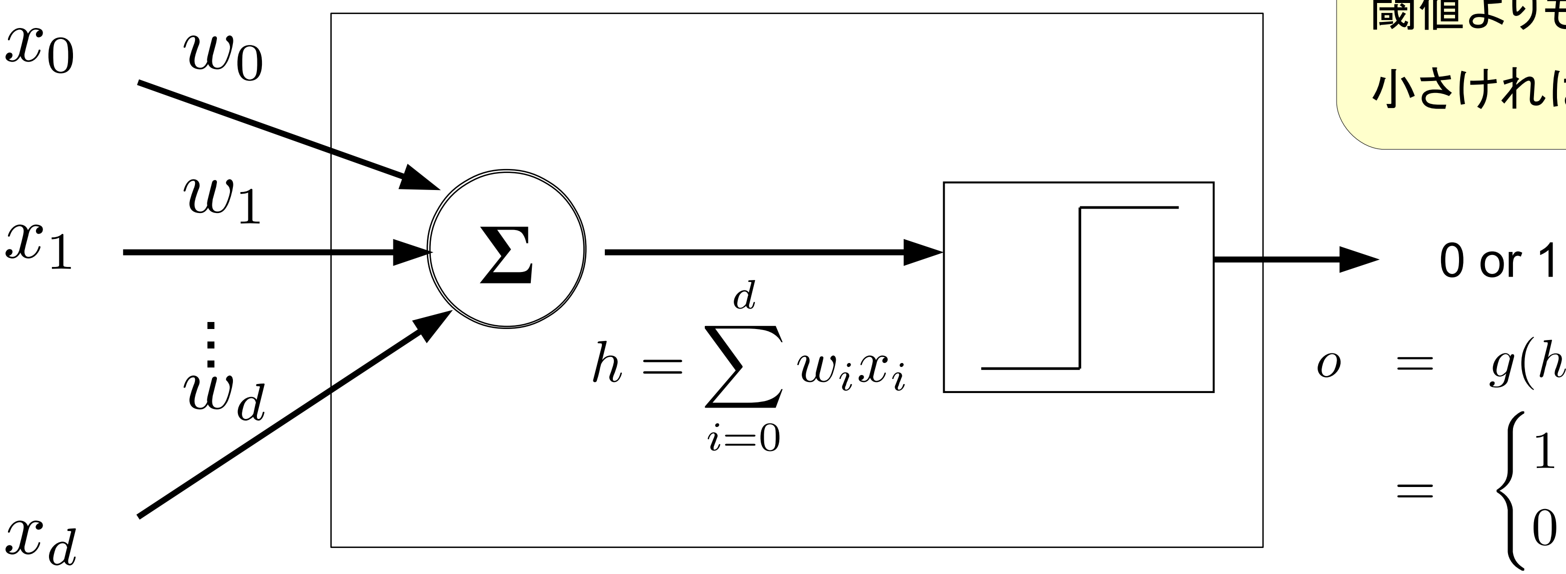
この平面を求めていることになる

確率と対応づけるには？

5.3.1 誤り訂正学習

- McCulloch & Pitts モデル

入力の重み付き和を求め、閾値よりも大きければ 1 、小さければ 0 を出力



$$
\begin{aligned}
o &= g(h) \\
&= \begin{cases} 1 & \text{if } h > 0 \\ 0 & \text{if } h \le 0 \end{cases}
\end{aligned}
$$

学習データが線形分離可能であれば、パーセプトロンの学習アルゴリズムで、すべての学習データを正しく識別できる重みの値を学習可能

5.3.1 誤り訂正学習

Algorithm 5.1 パーセプトロンの学習アルゴリズム

入力: 正解付学習データ $D\{(\boldsymbol{x}_i, y_i)\}$ $\quad i = 1, \ldots, N$ （数値特徴）

出力: 重み $\boldsymbol{w}$

重み $\boldsymbol{w}$ を適当な値で初期化

repeat

for all $\boldsymbol{x}_i \in D$ **do**

$o \leftarrow g(\boldsymbol{w} \cdot \boldsymbol{x}_i)$ $\quad$ /* $g(h)$ は閾値関数 */

if $o \neq y_i$ **then**

$\boldsymbol{w} \leftarrow \boldsymbol{w} + \eta(y_i - o)\boldsymbol{x}_i$

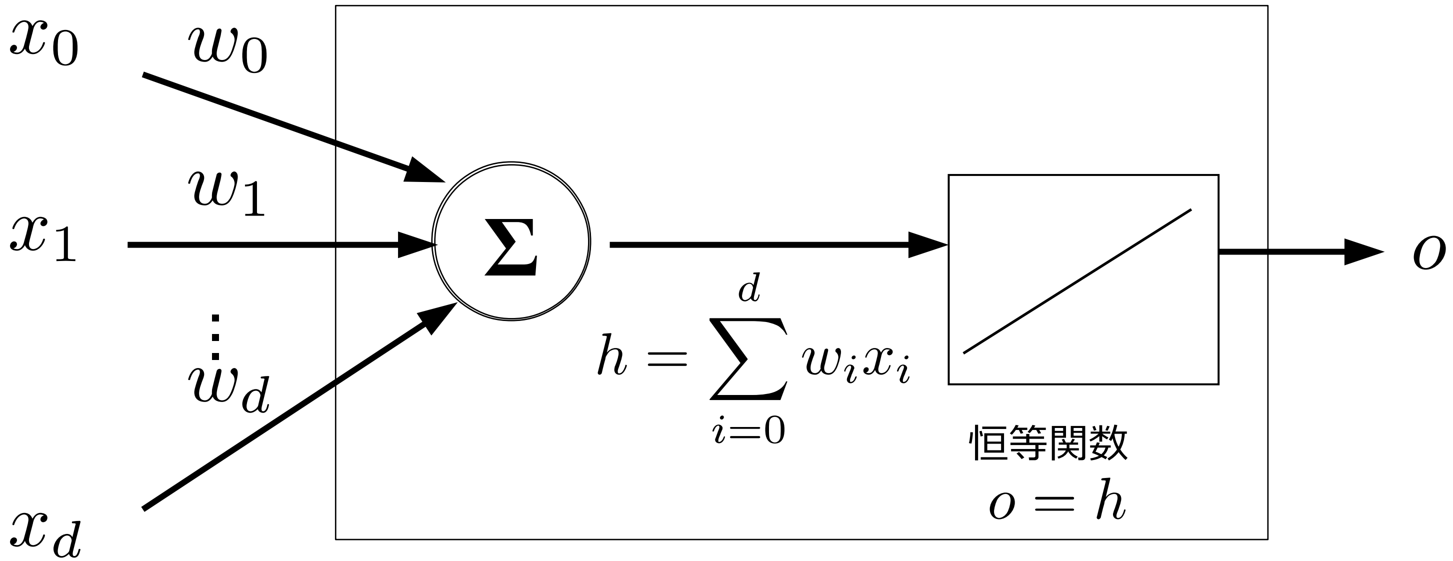
end if

end for

until エラーがなくなる

5.3.2 最小二乗法による学習

- 正解との二乗誤差が最小となるように学習



5.3.2 最小二乗法による学習

- 解析的に解が求まる
  - 求める関数

$$\hat{c}(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x}$$

  - 誤差関数

$$E(\boldsymbol{w}) = \sum_{i=1}^{N} (y_i - \hat{c}(\boldsymbol{x}_i))^2 = \sum_{i=1}^{N} (y_i - \boldsymbol{w} \cdot \boldsymbol{x}_i))^2$$

  - パターン行列による誤差関数の表現

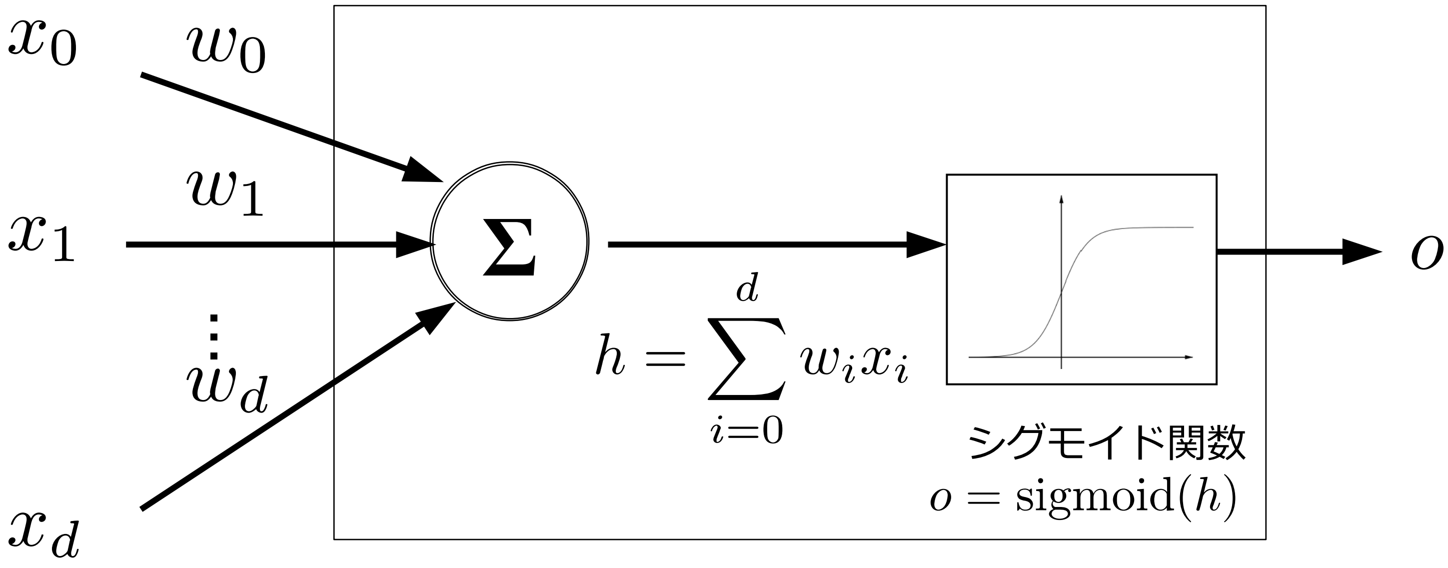
$$E(\boldsymbol{w}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})$$

  - 重み $\boldsymbol{w}$ で微分し、 0 と置いて極値を求める

$$\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) = 0 \Leftrightarrow \boldsymbol{w} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

5.3.3 識別モデルの考え方

- 事後確率 $p(\omega_i \mid \boldsymbol{x})$ を直接求める
  - 入力のすべての次元の情報を使う必要がない
    → 特徴選択を行っていることになる

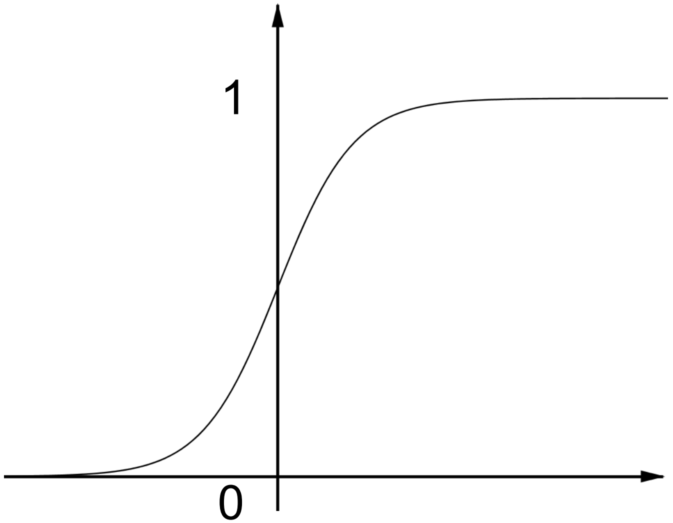


5.3.4 ロジスティック識別

- 入力が正例である確率

$$P(\oplus|\boldsymbol{x}) = \frac{1}{1+\exp(-(\boldsymbol{w}\cdot\boldsymbol{x}+w_0))}$$

-∞ ～ +∞ の値域を持つものを、順序を変えずに 0 ～ 1 にマッピング



シグモイド関数

5.3.4 ロジスティック識別

- 最適化対象＝モデルが学習データを生成する確率

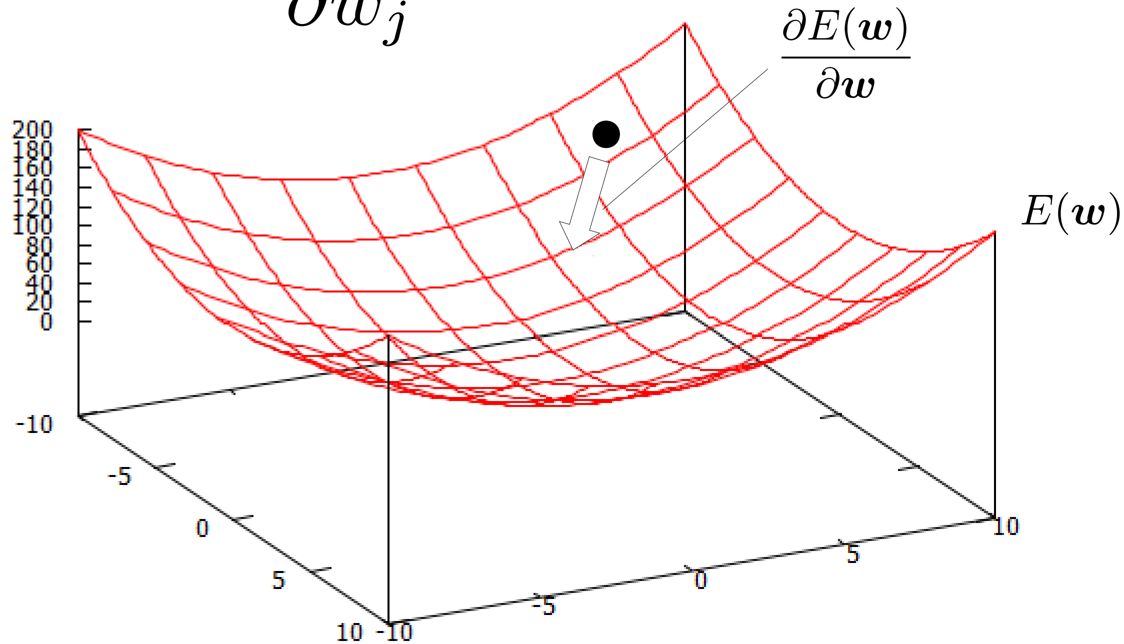
$$E(\boldsymbol{w}) = -\log P(D|\boldsymbol{w}) = -\log \prod_{\boldsymbol{x}_i \in D} o_i^{y_i}(1-o_i)^{(1-y_i)}$$

$o = P(\oplus|\boldsymbol{x})$
$y = 0 \;\; or \;\; 1$

正解ラベル

- $E(\boldsymbol{w})$ を最急勾配法で最小化

$$w_j \leftarrow w_j - \eta \frac{\partial E(\boldsymbol{w})}{\partial w_j}$$



5.3.4 ロジスティック識別

- 重み更新量の計算

$$\frac{\partial E(\boldsymbol{w})}{\partial w_j} = \sum_{\boldsymbol{x}_i \in D} \left(\frac{y_i}{o_i} - \frac{1 - y_i}{1 - o_i}\right) o_i (1 - o_i) x_{ij}$$
$$= \sum_{\boldsymbol{x}_i \in D} (y_i - o_i) x_{ij}$$

- 重みの更新式

$$w_j \leftarrow w_j - \eta \sum_{\boldsymbol{x}_i \in D} (y_i - o_i) x_{ij}$$

5.3.5 確率的最急勾配法

- 最急勾配法の問題点
  - １回の更新に全データに対する計算が必要
  - 大規模データでは学習時間が長くなる
- 確率的最急勾配法
  - 重みの更新を、個別のデータに対して行う

$$w_j \leftarrow w_j - \eta(y_i - o_i)x_{ij}$$

多クラスのロジスティック識別

- ソフトマックス関数

$$P(\omega_i|\boldsymbol{x}) = \frac{\exp(\boldsymbol{w}_i \cdot \boldsymbol{x})}{\sum_{j=1}^{c} \exp(\boldsymbol{w}_j \cdot \boldsymbol{x})}$$

  - $0 < P(\omega_i \,|\, \boldsymbol{x}) < 1$
  - $\Sigma\, P(\omega_i \,|\, \boldsymbol{x}) = 1$
  - マックス関数（最大値の要素のみ１、他は０）をソフトにしたもの

```
deff('y=sm(x)','y=exp(x) ./ sum(exp(x))')
sm([1 2 3 4 6])
```

  - ２クラスの場合はシグモイド関数に一致

一般的な識別モデル

- 対数線形モデル

$$P(y|\boldsymbol{x}) = \frac{1}{Z}\exp(\boldsymbol{w}\cdot\boldsymbol{\phi}(\boldsymbol{x},y))$$

Z: 確率の和を 1 にするための正規化項

- 素性ベクトル
  - 素性：入力と出力から作った識別に役立つ情報

$$\boldsymbol{\phi}(\boldsymbol{x},y) = (\phi_1(\boldsymbol{x},y),\cdots,\phi_d(\boldsymbol{x},y))$$

- 生成モデルとの違い
  - 素性は、この出力ならばこの特徴というような組み合わせで作ることができる
  - あるクラスの確率が増えれば、残りのクラスの確率が減る