# Pattern Recognition (Advanced)
# パターン認識特論 その 2

Kyoto University, ACCMS
京都大学 学術情報メディアセンター

Mori Shinsuke
森 信介

http://www.ar.media.kyoto-u.ac.jp/members/mori/

regularization
# 正　則　化

- linear programming classifier
  **線 形 計 画 識 別 器**

  let regularization term be　　norm
  **正　則　化　項　を** $L_1$ **ノルム** $||\boldsymbol{w}||_1 = \sum_{i=1}^{d'} |w_i|$ **にする**

  $$\min_{\boldsymbol{w},b} \frac{1}{n} \sum_{i=1}^{n} c(f(\boldsymbol{x}_i), y_i) + \lambda ||\boldsymbol{w}||_1$$

  sparse solution　　parameters near 0 become 0
  **スパースな解 (0に近い** $w_i$ **を0に押し切る)**

- various variations
  **様々なヴァリエーション**

  - first term　loss function
    **第 1 項: 損失関数**
  - second term　regularization term
    **第 2 項: 正　則　化　項**

sequential minimal optimization

# 逐次最小最適化アルゴリズム (SMO algorithm) その1

$$\boxed{\max W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad \text{s.t. } 0 \le \alpha_i \le C,\ \sum_{i=1}^{n} \alpha_i y_i = 0}$$

only two variable parameters

**1. 変化させるパラメータを2つに限定**

without loss of generality　　we choose

**一般性を失わずに $\alpha_1$, $\alpha_2$ と仮定**

before　　after

**変更前** $(\alpha_1^{\mathtt{old}}, \alpha_2^{\mathtt{old}}) \Rightarrow$ **変更後** $(\alpha_1^{\mathtt{new}}, \alpha_2^{\mathtt{new}})$

constant

$$\alpha_1^{\mathtt{new}} y_1 + \alpha_2^{\mathtt{new}} y_2 = \text{定数} = \alpha_1^{\mathtt{old}} y_1 + \alpha_2^{\mathtt{old}} y_2 \quad \left( \because\ \sum_{i=1}^{n} \alpha_i y_i = 0 \right)$$

is considered as the variable

**2. $\alpha_2^{\mathtt{new}}$ を変数とみなす**

$$\alpha_1^{\mathtt{new}} = \alpha_1^{\mathtt{old}} + (\alpha_2^{\mathtt{old}} - \alpha_2^{\mathtt{new}}) y_1 y_2$$

sequential minimal optimization
# 逐次最小最適化アルゴリズム (SMO algorithm) その2

$$\boxed{\max W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad \text{s.t.} \;\; 0 \le \alpha_i \le C, \;\; \sum_{i=1}^{n} \alpha_i y_i = 0}$$

from the box constraints

- **ボックス制約** $0 \le \alpha_i \le C$ **から**

$$U \le \alpha_2^{\mathtt{new}} \le V$$

**1.** $y_1 \ne y_2$ **の場合** (case)

$$\begin{aligned} U &= \mathbf{max}(0, \alpha_2^{\mathtt{old}} - \alpha_1^{\mathtt{old}}) \\ V &= \mathbf{min}(C, C - \alpha_1^{\mathtt{old}} + \alpha_2^{\mathtt{old}}) \end{aligned}$$

**2.** $y_1 = y_2$ **の場合** (case)

$$\begin{aligned} U &= \mathbf{max}(0, \alpha_1^{\mathtt{old}} + \alpha_2^{\mathtt{old}} - C) \\ V &= \mathbf{min}(C, \alpha_1^{\mathtt{old}} + \alpha_2^{\mathtt{old}}) \end{aligned}$$

sequential minimal optimization
# 逐次最小最適化アルゴリズム (SMO algorithm) その3

$$\boxed{\max W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad \text{s.t. } 0 \le \alpha_i \le C, \ \sum_{i=1}^{n} \alpha_i y_i = 0}$$

notetions for simplicity

- **簡略化のための表記法**

difference between the class and the output of the discriminant function

1. **(ある途中段階での) 識別関数の出力とクラスの差**

$$E_i = f(\boldsymbol{x}_i) - y_i = \left( \sum_{j=1}^{n} \alpha_j y_i K(\boldsymbol{x}_j, \boldsymbol{x}_i) + b \right) - y_i \quad i = 1, 2$$

second-order derivative along with the diagonal line

2. **対角直線に沿う目的関数の2次導関数 ($-\kappa$)**

$$\begin{aligned} \kappa &= K(\boldsymbol{x}_1, \boldsymbol{x}_1) + K(\boldsymbol{x}_2, \boldsymbol{x}_2) - 2K(\boldsymbol{x}_1, \boldsymbol{x}_2) \\ &= ||\boldsymbol{\phi}(\boldsymbol{x}_1) - \boldsymbol{\phi}(\boldsymbol{x}_2)||^2 \end{aligned}$$

# sequential minimal optimization
# 逐次最小最適化アルゴリズム (SMO algorithm) その4

$$\boxed{\max W(\boldsymbol{\alpha}) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad \text{s.t.} \;\; 0 \le \alpha_i \le C, \; \sum_{i=1}^{n}\alpha_i y_i = 0}$$

- the analitic solution in the two variable case
  **変化するパラメータが2の場合の解**

**1.** $\frac{\partial W}{\partial \alpha_2} = 0$ **から**

$$\alpha_2^{\texttt{tmp}} = \alpha_2^{\texttt{old}} + \frac{y_2(E_1 - E_2)}{\kappa}$$

← to be checked **要 確 認**

**2.** $U \le \alpha_2^{\texttt{new}} \le V$ therefore **なので**

$$\alpha_2^{\texttt{new}} = \begin{cases} V & \textbf{if} \quad V < \alpha_2^{\texttt{tmp}} \\ \alpha_2^{\texttt{tmp}} & \textbf{if} \quad U \le \alpha_2^{\texttt{tmp}} \le V \\ U & \textbf{if} \quad \alpha_2^{\texttt{tmp}} < U \end{cases}$$

**3.** finally **最後に** $\alpha_1^{\texttt{new}} = \alpha_1^{\texttt{old}} + (\alpha_2^{\texttt{old}} - \alpha_2^{\texttt{new}})y_1 y_2$

sequential minimal optimization
# 逐次最小最適化アルゴリズム (SMO algorithm) その5

$$\max W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad \text{s.t. } 0 \leq \alpha_i \leq C, \ \sum_{i=1}^{n} \alpha_i y_i = 0$$

select two parameters

- 変化させるパラメータ$\alpha_1$,$\alpha_2$ (に対応する点$\boldsymbol{x}_i$,$\boldsymbol{x}_j$ (の添字))の選択

search for points which do not satisfy KKT condition

1. KKT(Karush-Kuhn-Tucker)条件を満たさない点を探す

   let one to be $\alpha_1$ if found

   見つかればそれを$\alpha_1$とする

   select $\boldsymbol{x}_2$ which miximizes $|E_1 - E_2|$

2. $|E_1 - E_2|$が最大化となる$\boldsymbol{x}_2$を選択

   checking the sign

   (a) 符合チェック

   full search

   (b) 全探索

# sequential minimal optimization
# 逐次最小最適化アルゴリズム (SMO algorithm) その6

$$\boxed{\max W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad \text{s.t. } 0 \le \alpha_i \le C, \ \sum_{i=1}^{n} \alpha_i y_i = 0}$$

**select variable parameters**

**1. 変化させるパラメータ$\alpha_1$, $\alpha_2$の選択**

**renew**

**2. $\alpha_1$, $\alpha_2$の更新**

**check the stop condition**

**3. 停止条件のチェック**

maximum entropy method
# 最大エントロピー法

- distorted trihedron
  いびつな3面体

  probability of each surface
  各面の出る確率を $\boldsymbol{p} = (p_1,\ p_2,\ p_3)$ とする

  condition
  条　件：

  1. $p_1 + p_2 + p_3 = 1$
  2. $p_3 = 2p_1$

- what probabilities are reasonable
  各面の出る確率をいくらとするのが妥当か?

maximum entropy method
# 最大エントロピー法 (ME)

maximum entropy principle

- 最大エントロピー原理

  select the model of the maximum entropy

  エントロピーが最大となるモデルを選択

  in the distorted trihedron case

- いびつな3面体の例では

  obj. func.

  – 目的関数

  $$\max_{\boldsymbol{p}} H \quad \text{where ここで } H = \sum_{i=1}^{3} -p_i \log p_i$$

  s.t.

  – 制約条件

  **1.** $p_1 + p_2 + p_3 = 1$

  **2.** $p_3 = 2p_1$

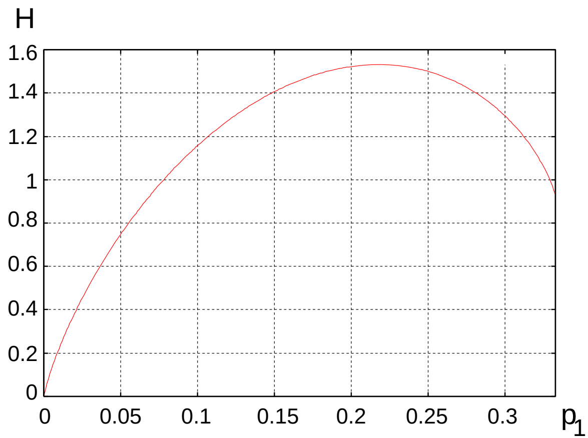
maximum entropy method
# 最大エントロピー法 (MaxEnt)

from the constraints
- **制約条件から**

$$p_3 = 2p_1,\ p_2 = 1 - 3p_1$$

obj. func
- **目的関数**

$$\frac{dH}{dp_1} = \sum_{i=1}^{3} \frac{\partial H}{\partial p_i}\frac{dp_i}{dp_1} = -\sum_{i=1}^{3} (\log p_i + 1)\frac{dp_i}{dp_1}$$




from the graph
- **グラフから (数値計算)**

$$\hat{\boldsymbol{p}} = \operatorname*{\mathbf{argmax}}_{\boldsymbol{p}} H = (0.218,\ 0.346,\ 0.436)$$
$$\hat{H} = \max_{\boldsymbol{p}} H = 1.5310$$

# constraint expression with feature functions
# 素性関数を用いた制約表現 (その1)

- training examples
  **学習データ**

$$\{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$$

- feature functions
  **素性関数**

$$\phi_i : (x, y) \mapsto \{0, 1\}, \quad i \in \{1, 2, \cdots, n\}$$

- empirical distribution
  **経験的確率分布**

$$\tilde{P}(x, y) = \frac{C(x, y)}{N}$$

constraint expression with feature functions

# 素性関数を用いた制約表現 (その 2)

expectation value of $f_i$ based on the empirical distribution

- 経験的確率分布による素性 $f_i$ の期待値

$$E_{\tilde{P}}[\phi_i] = \sum_{x,y} \tilde{P}(x,y)\phi_i(x,y)$$

- (推定すべき) モデル $P(x,y)$ による素性 $\phi_i$ の期待値

$$E_P[\phi_i] = \sum_{x,y} P(x,y)\phi_i(x,y)$$

these must be equal

- 両者は一致しているべき

$$\sum_{x,y} P(x,y)\phi_i(x,y) = \sum_{x,y} \tilde{P}(x,y)\phi_i(x,y)$$

# maximum entropy method
# 最大エントロピー法

- obj. func. maximize
  **目的関数 (最大化)**

$$H(P) = -\sum_{x,y} P(x,y) \log P(x,y)$$

- constraint
  **制　約**

$$\sum_{x,y} P(x,y)\phi_i(x,y) = \sum_{x,y} \tilde{P}(x,y)\phi_i(x,y) \quad \forall i \in \{1, 2, \cdots, n\}$$

$$P(x,y) \geq 0 \qquad \forall (x,y)$$

$$\sum_{x,y} P(x,y) = 1$$

**制約を満たしつつ、$P(x,y)$ をできるだけ散らす**

# Lagrange multiplier method
# ラグランジュの未定乗数法

introduce Lagrange multipliers

- ラ グ ラ ン ジュ乗 数 $w_i\ (\geq 0)$ を導入

$$
\begin{aligned}
& L(P, \boldsymbol{w}) \\
&= H(P) + \sum_{i=1}^{n} w_i (E_P[\phi_i] - E_{\tilde{P}}[\phi_i]) \\
&= -\sum_{(x,y)} P(x,y) \log P(x,y) + \sum_{i=1}^{n} w_i \sum_{x,y} \{P(x,y) - \tilde{P}(x,y)\} \phi_i(x,y)
\end{aligned}
$$

partial derivative is

- 偏 微 分 が 0 (必要条件)

$$
\frac{\partial L}{\partial P(x,y)} = -\log P(x,y) - 1 + \sum_{i=1}^{n} w_i \phi_i(x,y) = 0
$$

$$
\Leftrightarrow P(x,y) = C \exp\left(\sum_{i=1}^{n} w_i \phi_i(x,y)\right)
$$

十分性は省略

constant $C$ is decided by the other constraints

- 定 数 $C$ は 残 り の 制 約 か ら 決 定

parametric form

# パラメトリック形式

formation in a parametric form

- 以下のパラメトリック形式で表せる (ラグランジュ未定乗数法)

$$P_{\boldsymbol{w}}(x, y) = \frac{1}{Z_{\boldsymbol{w}}} \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x, y) \rangle)$$

$$Z_{\boldsymbol{w}} = \sum_{x,y} \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x, y) \rangle)$$

are the weights　　is the feature vector

$\boldsymbol{w}$は各素性の重み、$\phi(x, y)$は素性ベクトル

$$\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x, y) \rangle = \sum_{i=1}^{n} w_i \phi_i(x, y)$$

search algorithm

- 解探索アルゴリズム

**GIS(Generalized Iterative Scaling),**
**BFGS(Broyden-Fletcher-Goldfarb-Shanno)**

conditional maximum entropy model

# 条件付き最大エントロピーモデル

we need for discrimination

- 識別のためには$P(y|x)$が 必 要

is replaced by

$P(x,y)$ を $\tilde{P}(x)P(y|x)$で置き換える

obj. func.

- 目的関数

$$H(P) = -\sum_{x,y} \tilde{P}(x)P(y|x)\log P(y|x)$$

s.t.

- 制約

$$\underline{\sum_{x} \tilde{P}(x)} \sum_{y} P(y|x)\phi_i(x,y) = \sum_{x,y} \tilde{P}(x,y)\phi_i(x,y) \quad \forall i \in \{1,2,\cdots,n\}$$

this refers to only $x$ in the training data

↑訓練データに出現する$x$のみが対象

## パラメトリック形式の解

solution

- $x$ に依存する解

solution depending on $x$

$$P_{\boldsymbol{w}}(y|x) = \frac{1}{Z_{\boldsymbol{w}}(x)} \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle)$$

$$Z_{\boldsymbol{w}}(x) = \sum_y \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle)$$

**$\boldsymbol{w}$は各素性の重み、$\phi(x,y)$は素性ベクトル**

**are the weights** … **is the feature vector**

$$\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle = \sum_{i=1}^{n} w_i \phi_i(x,y)$$

## parameter estimation
## パラメータ推定

- probability distribution
  確率分布

$$P(y|x;\boldsymbol{w}) = \frac{1}{Z_w(x)} \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle)$$

- minimize the following
  以下を最小化

$$\hat{\boldsymbol{w}} = \mathop{\mathbf{argmin}}_{\boldsymbol{w}} \sum_{i=1}^{n} -\log P(y_i|x_i;\boldsymbol{w})$$

**the frequency of $x$ is considered because of a sum over $i$**
**※$i$に対する合計なので$x$の頻度が考慮される**

# ME as a classifier
# 識別器としてのME

estimate the label $y$ with the highest prob. for an input $x$

- 入力$x$に対する確率最大のラベル$y$の推定

  prob. distribution

  – 確率分布

$$P(y|x;\boldsymbol{w}) = \frac{1}{Z_w(x)} \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle)$$

  classification　　can be $n$-ary

  – 識別 (※$y$は多値)

$$\begin{aligned}
\hat{y} &= \operatorname*{\mathbf{argmax}}_y P(y|x;\boldsymbol{w}) \\
&= \operatorname*{\mathbf{argmax}}_y \log P(y|x;\boldsymbol{w}) \\
&= \operatorname*{\mathbf{argmax}}_y \log \frac{1}{Z_w(x)} \exp(\langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle) \\
&= \operatorname*{\mathbf{argmax}}_y \langle \boldsymbol{w} \cdot \boldsymbol{\phi}(x,y) \rangle - \log Z_w(x)
\end{aligned}$$

# MEの正則化 (regularization)

- 正則化項を加えて過学習を防ぐ (avoid over training by a regularization term)

- $L_2$ノルム (norm)

$$\min_{\boldsymbol{w}} \sum_{i=1}^{n} - \log P(y_i | x_i; \boldsymbol{w}) + \frac{C}{2} ||\boldsymbol{w}||_2$$

事前分布にガウス分布$\exp(-w_i^2/2\sigma^2)$を仮定 (assume a Gaussian dist. as the prior)

- $L_1$ノルム (norm)

$$\min_{\boldsymbol{w}} \sum_{i=1}^{n} - \log P(y_i | x_i; \boldsymbol{w}) + C ||\boldsymbol{w}||_1$$

事前分布にラプラス分布$\exp(-\sigma |w_i|)$を仮定 (assume a Laplacian dist. as the prior)

loss function comp.
# 損失関数の比較 (SVM, ME, etc.)

- loss func. regularization term
  損失関数 + 正則化項

$$\min_{\boldsymbol{w}} \frac{1}{n}\sum_{i=1}^{n} \mathbf{Loss}(f(\boldsymbol{x}_i), y_i) + CR(\boldsymbol{w})$$

※ $f(\boldsymbol{x})$ は (includes) $w$を含む

1. **SVM**

   $$\mathbf{HingeLoss}(f(\boldsymbol{x}), y) = \mathbf{max}(1 - yf(\boldsymbol{x}), 0)$$

2. **ME** ($y \in \{-1, 1\}$)

   $$\mathbf{LogLoss}(f(\boldsymbol{x}), y) = \log\left(1 + e^{-yf(\boldsymbol{x})}\right)$$

3. **AdaBoost** ($y \in \{-1, 1\}$)

   $$\mathbf{ExpLoss}(f(\boldsymbol{x}), y) = e^{-yf(\boldsymbol{x})}$$

4. square loss
   **2乗誤差** ($y \in \{-1, 1\}$)

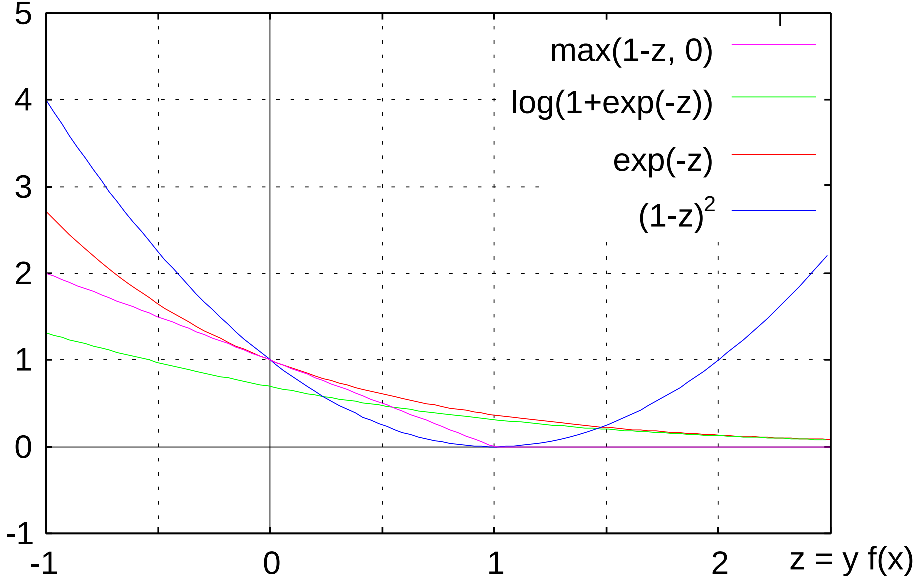
   $$\mathbf{SquareLoss}(f(\boldsymbol{x}), y) = (1 - yf(\boldsymbol{x}))^2 \qquad \because\ y^2 = 1$$

graph of the loss functions

# 損失関数のグラフ

$$\textbf{SVM:}\quad \mathbf{max}(1 - yf(\boldsymbol{x}), 0) \qquad \textbf{AdaBoost:}\quad e^{-yf(\boldsymbol{x})}$$

$$\textbf{ME:}\quad \log(1 + e^{-yf(\boldsymbol{x})}) \qquad \textbf{MeanSquare:}\quad (1 - yf(\boldsymbol{x}))^2$$



# report
# 課 題

- friday deadline
  7月23日(金曜) 〆 切
- hand written
  手 書 き
- staple together multiple pages
  複数枚の場合は綴じて提出

1. you attended
   if 講義参加者

   - derive the objective function in the soft margin case
     ソフトマージンの場合の$\alpha$に対する目的関数の導出
   - derive in SMO
     SMOにおける$U \leq \alpha_2^{\mathtt{new}} \leq V$の導出

   make your own questions of equal or greater difficulty and answer them
   Else それぞれについて同等かそれ以上難しい課題とその回答
   must be different from all questions in this or previous years for full credit
   (今年分も含め既出との重複は減点)

2. questionnaire
   アンケート

   (a) what interested you
   興味を持ったトピック
   (b) what you want to learn more about
   もっと詳しく掘下げて欲しいトピック
   (c) other topics you want to learn
   新たに扱って欲しいトピック