

情報生命科学演習

AdaBoost の実装

11月2, 9日

森下

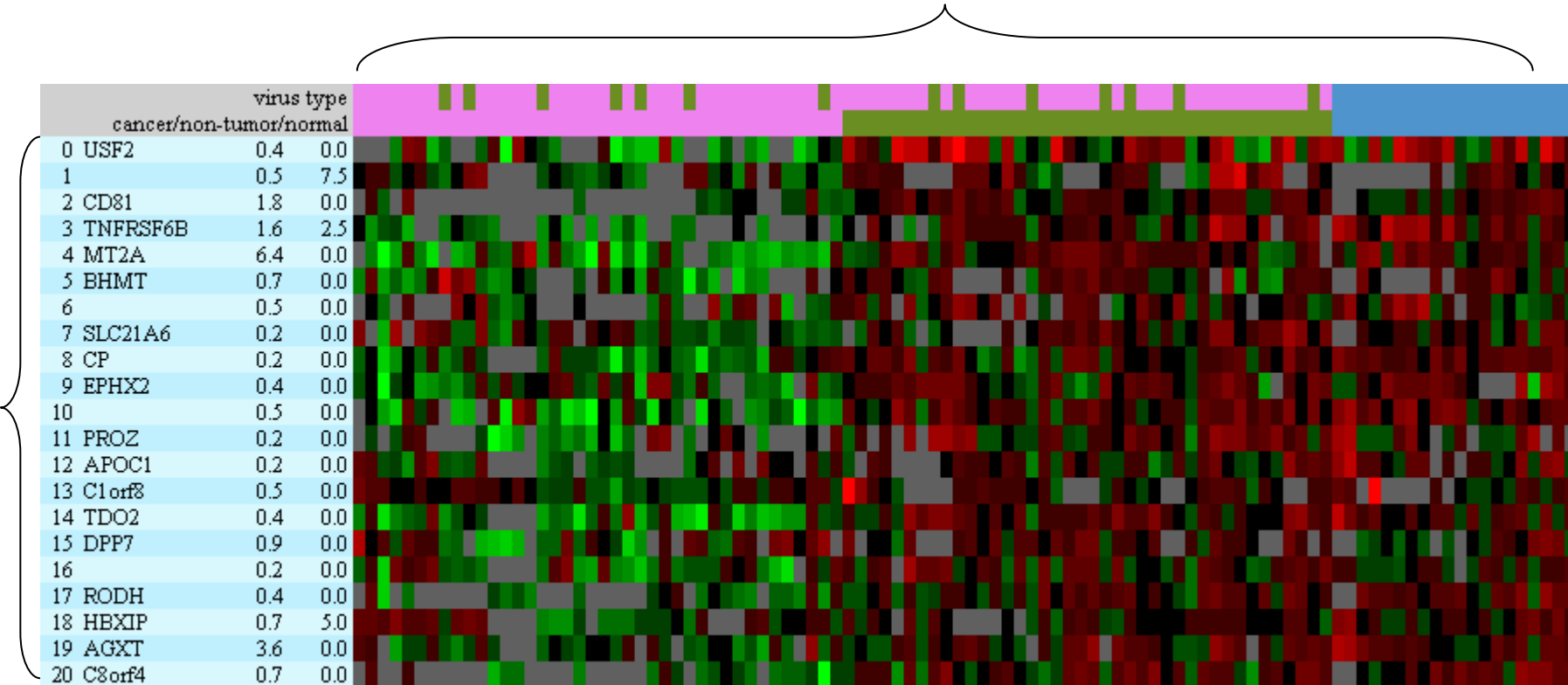
遺伝子発現, 疾患, 表現型

組織 肝癌細胞

virus type: ■, HCV; ■, HBV ; ■, non-B,C.

tumor~normal: ■, tumor ; ■, non-tumor ; ■, normal.

遺伝子



CGED (Cancer Gene Expression Database)

<http://cged.hgc.jp/>

加藤菊也先生 大阪府立成人病センター

AdaBoost の応用例

Marcel Dettling and Peter Bühlmann

Boosting for tumor classification with gene expression data

Bioinformatics, Jun 2003; 19: 1061 - 1069.

Jinyan Li, Huiqing Liu, See-Kiong Ng, and Limsoon Wong

Discovery of significant rules for classifying cancer diagnosis data

Bioinformatics, Sep 2003; 19: 93 - 102.

Manuel Middendorf, Anshul Kundaje, Chris Wiggins, Yoav Freund, and Christina Leslie

Predicting genetic regulatory response using classification

Bioinformatics, Aug 2004; 20: i232 - i240.

Pål Sætrom

Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming

Bioinformatics, Advance Access published on June 16, 2004; doi:
10.1093/bioinformatics/bth364.

演習内容

- **11月2日の講義**
AdaBoost アルゴリズムの説明
プログラム化のヒント
- **11月9日の講義**
プログラム例を紹介
- **11月9日までに自分のプログラムを完成できなかった人は、
プログラムの例の一部を穴埋め**
- **レポートのメ切 11月16日**

三人寄れば文殊の知恵

Boosting とは正答率の低いクラス分類器を組合せて、
高い正答率を示すクラス分類器を構成する技法。

参考文献

Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119-139, 1997.

訓練用データ

	T1	T2	T3	T4	目標属性	
$\vec{x}_1$	1	0	1	1	1	y_1
	1	0	1	1	1	
	1	1	1	1	1	
	1	1	1	0	0	
	1	0	1	0	0	
	1	1	0	1	0	
	1	0	0	1	0	
	1	1	0	1	0	
	0	1	0	1	1	
	0	0	1	1	1	
	0	1	0	1	1	
	0	1	0	1	1	
	0	0	0	1	1	
	0	0	1	0	0	
	0	1	0	0	0	
	0	0	1	0	0	

$(\vec{x}_1, y_1)$
 $\vdots$
 $(\vec{x}_N, y_N)$

AdaBoost の基本的考え方

- **初期状態では各レコードに均等な重みを割当てる**
- **次のステップを繰り返す**
 1. **ランダムに推測するよりは正答率の高い
(50%を超える) クラス分類器を生成**
 2. **目標属性の値の予測を誤ったレコードの重みを
相対的に上げる
(予測が困難であることを覚えておく)**

註: 参考論文中ではクラス分類器(classifier)のことを仮説(hypothesis)と呼んでいる

クラス分類器

T1	T2	T3	T4	目標	重み	if T1=1 then Ob=0 else Ob=1	新しい 重み
1	0	1	1	1		0	
1	0	1	1	1		0	
1	1	1	1	1		0	
1	1	1	0	0		0	
1	0	1	0	0		0	
1	1	0	1	0		0	
1	0	0	1	0		0	
1	1	0	1	0		0	

 の大きさが重みを表現

新たな
クラス分類器

T1	T2	T3	T4	目標	重み	if <u>T3</u> =1 then Ob=1 else Ob=0	新しい 重み
1	0	1	1	1		1	
1	0	1	1	1		1	
1	1	1	1	1		1	
1	1	1	0	0		1	
1	0	1	0	0		1	
1	1	0	1	0		0	
1	0	0	1	0		0	
1	1	0	1	0		0	

新たな
クラス分類器

T1	T2	T3	T4	目標	重み	if <u>T4</u> =1 then Ob=1 else Ob=0	新たな 重み
1	0	1	1	1	■	1	■
1	0	1	1	1	■	1	■
1	1	1	1	1	■	1	■
1	1	1	0	0	■	0	■
1	0	1	0	0	■	0	■
1	1	0	1	0	■	1	■
1	0	0	1	0	■	1	■
1	1	0	1	0	■	1	■

クラス分類器

if T1=1
then Ob=0
else Ob=1

if T3=1
then Ob=1
else Ob=0

if T4=1
then Ob=1
else Ob=0

単純な
多数決

T1 T2 T3 T4 Ob

1	0	1	1	1	0	1	1	1
1	0	1	1	1	0	1	1	1
1	1	1	1	1	0	1	1	1
1	1	1	0	0	0	1	0	0
1	0	1	0	0	0	1	0	0
1	1	0	1	0	0	0	1	0
1	0	0	1	0	0	0	1	0
1	1	0	1	0	0	0	1	0

AdaBoost **は重みを使った多数決**

AdaBoost アルゴリズム

入力

訓練データ $(\vec{x}_1, y_1), \dots, (\vec{x}_N, y_N)$

初期の重み $w_i^1 = D(i) = \frac{1}{N}$ for each $i = 1, 2, \dots, N$

WeakLearn : エラー率が0.5未満の
クラス分類器を常に出力する学習アルゴリズム

T : 最終的に使うクラス分類器の個数

各 $t=1, \dots, T$, について以下のステップを繰り返す:

1: **各データの重みを正規化し、分布 p_i^t を計算**

$$p_i^t = w_i^t / \sum_{i=1}^N w_i^t$$

2: **WeakLearn を呼び出し次の条件を満たすクラス分類器 h_t を生成**

$$\varepsilon_t = \sum_{i=1}^N p_i^t |h_t(\vec{x}_i) - y_i| < 1/2$$

$$|h_t(\vec{x}_i) - y_i| = \begin{cases} 0 & h_t \text{ が } \vec{x}_i \text{ の正解を返す場合} \\ 1 & \text{otherwise} \end{cases}$$

3: **重みを更新 重みは正解だと軽くなり、不正解だとそのまま**

$$\beta_t = \varepsilon_t / (1 - \varepsilon_t) \quad \beta_t < 1 \quad w_i^{t+1} = w_i^t \beta_t^{1 - |h_t(\vec{x}_i) - y_i|}$$

$$\varepsilon_t \rightarrow 0, \quad \beta_t \rightarrow 0 \quad \varepsilon_t \rightarrow 1/2, \quad \beta_t \rightarrow 1$$

出力: 最終のクラス分類器 h_f (h_t の重みつき多数決)

$$h_f(\vec{x}) = \begin{cases} 1 & \text{if } \sum_{t=1}^T (-\log \beta_t) h_t(\vec{x}) \geq \sum_{t=1}^T (-\log \beta_t) \frac{1}{2} \quad \dots (*) \\ 0 & \text{otherwise} \end{cases}$$

$$\varepsilon_t \rightarrow 0, \quad \beta_t \rightarrow 0, \quad (-\log \beta_t) \rightarrow +\infty \quad \varepsilon_t \rightarrow 1/2, \quad \beta_t \rightarrow 1, \quad (-\log \beta_t) \rightarrow 0$$

定義 $\varepsilon = \sum_{\{i \mid h_f(\vec{x}_i) \neq y_i\}} D(i)$

**最終クラス分類器 h_f の
初期分布に対するエラー率**

定理 $\varepsilon \leq \prod_{t=1}^T 2\sqrt{\varepsilon_t(1-\varepsilon_t)} = \prod_{t=1}^T \sqrt{1-(1-2\varepsilon_t)^2}$

$$\varepsilon_t < 1/2, \quad 1-2\varepsilon_t < 1, \quad \sqrt{1-(1-2\varepsilon_t)^2} < 1$$

AdaBoost の実装

訓練データとテストデータの読み込みとデータ構造

ヒント 理学部・生物情報科学・学部教育特別プログラム

瀬々潤助手のサンプルプログラムの利用

<http://mlab.cb.k.u-tokyo.ac.jp/~sesejun/upbsb/jikken/2004/>

データの属性: 2値, 数値, テキスト

本演習では2値が目標、できれば数値

クラス分類器のデータ構造

簡単なクラス分類器 1個の属性 $A=1$, $A > \text{閾値}$ など

一般には Support Vector Machine, 決定木, k Nearest Neighbor

WeakLearn の実装, 各レコードの重みの更新

サンプルデータ (2値)

T1	T2	T3	T4	目標属性
1	0	1	1	1
1	0	1	1	1
1	1	1	1	1
1	1	1	0	0
1	0	1	0	0
1	1	0	1	0
1	0	0	1	0
1	1	0	1	0

T1	T2	T3	T4	目標属性
1	0	1	1	1
1	0	1	1	1
1	1	1	1	1
1	1	1	0	0
1	0	1	0	0
1	1	0	1	0
1	0	0	1	0
1	1	0	1	0
0	1	0	1	1
0	0	1	1	1
0	1	0	1	1
0	1	0	1	1
0	0	0	1	1
0	0	1	0	0
0	1	0	0	0
0	0	1	0	0

遺伝子発現量データ (数値)

MIT, Broad Institute, Cancer Program Data Sets

<http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi>

Princeton University, Gene Expression Project

<http://microarray.princeton.edu/oncology/>

Nara Institute of Science and Technology,

Taisho Laboratory of Functional Genomics

http://love2.aist-nara.ac.jp/laboratory/data_download.html

レポートの内容

プログラムのソースコード

データ構造と手続きの流れの説明

サンプルデータ (2 値) から計算された分類器

重み、エラー率の変動についての解析

オプション 数値データの解析

参考資料 Boosting 性の証明

定義 $\varepsilon = \sum_{\{i \mid h_f(\vec{x}_i) \neq y_i\}} D(i)$

最終クラス分類器 h_f の
初期分布に対するエラー率

定理
$$\varepsilon \leq \prod_{t=1}^T 2\sqrt{\varepsilon_t(1-\varepsilon_t)} = \prod_{t=1}^T \sqrt{1-(1-2\varepsilon_t)^2}$$

$\varepsilon_t < 1/2, \quad 1-2\varepsilon_t < 1, \quad \sqrt{1-(1-2\varepsilon_t)^2} < 1$

証明のロードマップ

$$\varepsilon \left(\prod_{t=1}^T \beta_t \right)^{1/2} \leq \sum_{i=1}^N w_i^{T+1} \quad \text{補題 2}$$

$$\leq \left(\sum_{i=1}^N w_i^T \right) \times 2\varepsilon_t \quad \text{補題 3} \quad \sum_{i=1}^N w_i^{t+1} \leq \left(\sum_{i=1}^N w_i^t \right) \times 2\varepsilon_t$$

$\vdots$

$$\leq \prod_{t=1}^T 2\varepsilon_t \quad \sum_{i=1}^N w_i^1 = 1$$

$$\varepsilon \leq \prod_{t=1}^T \left(2\varepsilon_t \times \beta_t^{-1/2} \right) = \prod_{t=1}^T 2\sqrt{\varepsilon_t(1-\varepsilon_t)}$$

$\beta_t = \varepsilon_t / (1 - \varepsilon_t)$ に注意

$$\text{補題 2} \quad \sum_{i=1}^N w_i^{T+1} \geq \varepsilon \left(\prod_{t=1}^T \beta_t \right)^{1/2}$$

$$\begin{aligned} \sum_{i=1}^N w_i^{T+1} &\geq \sum_{\{i \mid h_f(\bar{x}_i) \neq y_i\}} w_i^{T+1} && h_f \text{ が誤るレコードの重みの総和} \\ &= \sum_{\{i \mid h_f(\bar{x}_i) \neq y_i\}} \left(D(i) \prod_{t=1}^T \beta_t^{1-|h_t(\bar{x}_i)-y_i|} \right) && w_i^{t+1} = w_i^t \beta_t^{1-|h_t(\bar{x}_i)-y_i|} \quad \text{より} \\ &\geq \sum_{\{i \mid h_f(\bar{x}_i) \neq y_i\}} \left(D(i) \left(\prod_{t=1}^T \beta_t \right)^{1/2} \right) && \text{補題 1 より} \\ &= \left(\sum_{\{i \mid h_f(\bar{x}_i) \neq y_i\}} D(i) \right) \left(\prod_{t=1}^T \beta_t \right)^{1/2} \end{aligned}$$

$$h_f(\vec{x}) = \begin{cases} 1 & \text{if } \sum_{t=1}^T (-\log \beta_t) h_t(\vec{x}) \geq \sum_{t=1}^T (-\log \beta_t) \frac{1}{2} \quad \dots (*) \\ 0 & \text{otherwise} \end{cases}$$

補題1 $h_f(\vec{x}_i) \neq y_i$ ならば $\prod_{t=1}^T \beta_t^{1-|h_t(\vec{x}_i)-y_i|} \geq \left(\prod_{t=1}^T \beta_t \right)^{1/2}$

コメント 最終のクラス分類機で予測が失敗すると重みが十分に減らない

$h_f(\vec{x}_i) = 1, y_i = 0$ ならば: $\sum_{t=1}^T (\log \beta_t)$ を (*) の両辺に加算.

$$\sum_{t=1}^T (\log \beta_t) (1 - h_t(\vec{x}_i)) \geq \sum_{t=1}^T (\log \beta_t) \frac{1}{2}$$

さらに $1 - h_t(\vec{x}_i) = 1 - |h_t(\vec{x}_i) - y_i|$ に注意

$h_f(\vec{x}_i) = 0, y_i = 1$ ならば: $\sum_{t=1}^T (-\log \beta_t) h_t(\vec{x}_i) < \sum_{t=1}^T (-\log \beta_t) \frac{1}{2}$

$$\sum_{t=1}^T (\log \beta_t) h_t(\vec{x}_i) > \sum_{t=1}^T (\log \beta_t) \frac{1}{2}$$

さらに $h_t(\vec{x}_i) = 1 - |h_t(\vec{x}_i) - y_i|$ に注意

補題3

$$\sum_{i=1}^N w_i^{t+1}$$

$$= \sum_{i=1}^N w_i^t \beta_t^{1-|h_t(\vec{x}_i) - y_i|}$$

$$\leq \sum_{i=1}^N w_i^t (1 - (1 - \beta_t)(1 - |h_t(\vec{x}_i) - y_i|))$$

$$\begin{array}{l} \alpha^\gamma \leq 1 - (1 - \alpha)\gamma, \\ \text{if } \alpha \leq 1 \text{ and } \gamma = 0 \text{ or } 1. \end{array}$$

$$= \sum_{i=1}^N w_i^t - (1 - \beta_t) \sum_{i=1}^N w_i^t (1 - |h_t(\vec{x}_i) - y_i|)$$

正解する重みの和

$$\begin{aligned} 1 - \varepsilon_t &= \sum_{i=1}^N p_i^t - \sum_{i=1}^N p_i^t |h_t(\vec{x}_i) - y_i| \\ &= \sum_{i=1}^N p_i^t (1 - |h_t(\vec{x}_i) - y_i|) \\ &= \sum_{i=1}^N w_i^t (1 - |h_t(\vec{x}_i) - y_i|) / (\sum_{i=1}^N w_i^t) \end{aligned}$$

$$= (\sum_{i=1}^N w_i^t) (1 - (1 - \beta_t)(1 - \varepsilon_t))$$

$$= (\sum_{i=1}^N w_i^t) \bullet 2\varepsilon_t$$

$$1 - \beta_t = 1 - \frac{\varepsilon_t}{1 - \varepsilon_t} = \frac{1 - 2\varepsilon_t}{1 - \varepsilon_t}$$