よく目にするサポートベクトルマシン，ブースティングって？

青山学院大学　理工学部
経営システム工学科
小野田　崇

チュートリアルの内容

- サポートベクトルマシン(SVM)って何してるの？
  - SVMの学習って？
  - SVMをちょっと定式化？
- AdaBoostって何してるの？
  - AdaBoostのアルゴリズムって？
  - AdaBoostの学習って？

チュートリアルの内容

- サポートベクトルマシン(SVM)って何してるの？
  - SVMの学習って？
  - SVMをちょっと定式化？
- AdaBoostって何してるの？
  - AdaBoostのアルゴリズムって？
  - AdaBoostの学習って？

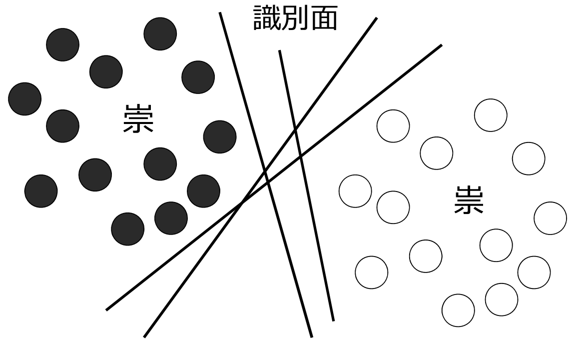
SVMの学習(1)

- SVM→線形識別器

$$f(\boldsymbol{x}) = \sum_{j=1}^{d} w_j x_j + b = \boldsymbol{w}^T \boldsymbol{x} + b$$

  - サンプル$\boldsymbol{x}^T = (x_1, \cdots, x_d)$
  - w_j 重みパラメータ，$\boldsymbol{w}$重みベクトル
  - bはバイアス項
  - $f(x) = 0$を満たす点の集合(識別面)は$d-1$次元の超平面

SVMの学習(2)
分離可能な場合：2次元空間



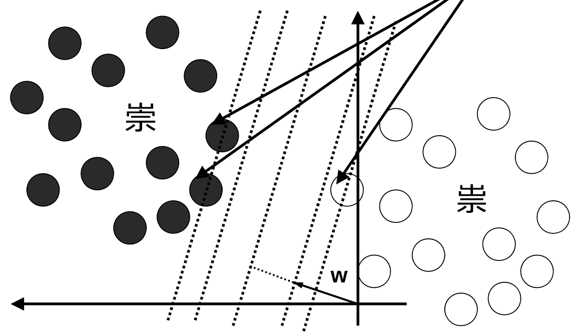
SVMの学習(3)

- パターン識別の目的
  - 未知のテストサンプルを正しく識別すること
    - 訓練サンプルを完全に識別する超平面の中で，テストサンプルを識別するのに，最も優れた識別超平面はどれか？
- SVMでの識別最適超平面
  - 2つのクラスの「真中」を通る超平面
    - 2つのクラスの「真中」を通る超平面を選ぶには，どのような評価関数を設定すればよいのだろうか？

SVMの学習(4)

- SVMの評価関数
  - 識別超平面と訓練サンプルとの最短距離を評価関数
  - この評価関数を最大化するように識別超平面を決定
- SVMの識別面．．．
  - 識別面は，各クラスの訓練サンプルの凸包(convex hull)を結ぶ最短の線分の中点を通り，直交する超平面.

SVMの学習(5)



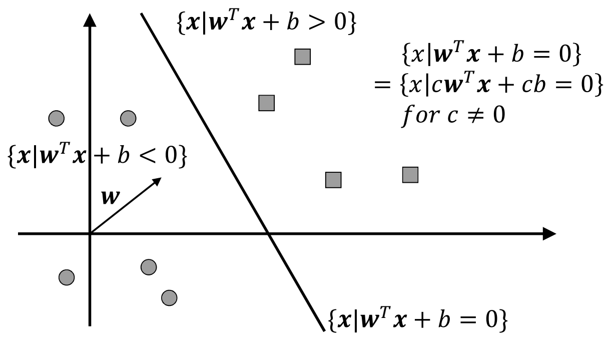
学習の定式化(1)

- SVMの学習問題→二次計画問題
- 2クラス問題
  - 訓練サンプル $\boldsymbol{x}_1, \dots, \boldsymbol{x}_l$
  - 各クラスラベル $y_1, \dots, y_l$
    - 訓練サンプルがクラス崇に属していれば $y = +1$
    - 訓練サンプルがクラス崇に属していれば $y = -1$

学習の定式化(2)

- SVMの識別超平面→$f(\boldsymbol{x}) = \sum_{j=1}^{d} w_j x_j = \boldsymbol{w}^T\boldsymbol{x} + b$
  - $\boldsymbol{w}, b$は，冗長性を持っている
    - → $\boldsymbol{w}, b$ は，定数倍しても表現する超平面が変わらない
    - →学習結果が一意に定まらない

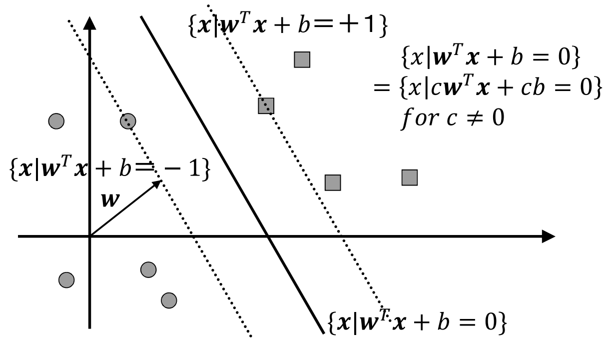
一意に決まらない



学習の定式化(2)

- SVMの識別超平面→$f(\boldsymbol{x}) = \sum_{j=1}^{d} w_j x_j = \boldsymbol{w}^T\boldsymbol{x} + b$
  - $\boldsymbol{w}, b$は，冗長性を持っている
    - → $\boldsymbol{w}, b$ は，定数倍しても表現する超平面が変わらない
    - →学習結果が一意に定まらない
- 制約条件を付加
  - $\min_{i=1,\cdots,l} |\boldsymbol{w}^T\boldsymbol{x}_i + b| = 1$

制約を付加



学習の定式化(2)

- SVMの識別超平面→$f(\boldsymbol{x})=\sum_{j=1}^{d} w_j x_j=\boldsymbol{w}^T\boldsymbol{x}+b$
  - $\boldsymbol{w},b$は，冗長性を持っている
    - → $\boldsymbol{w},b$ は，定数倍しても表現する超平面が変わらない
    - →学習結果が一意に定まらない
- 制約条件を付加

$$\min_{i=1,\cdots,l}|\boldsymbol{w}^T\boldsymbol{x}_i+b|=1$$

- 訓練サンプルと識別超平面の最小距離

$$\min_{i=1,\cdots,l}\frac{|\boldsymbol{w}^T\boldsymbol{x}_i+b|}{\|\boldsymbol{w}\|}\to\frac{1}{\|\boldsymbol{w}\|}$$

学習の定式化(3)

- SVMでは…
  - $\boldsymbol{w},b$は訓練サンプルを<u>完全識別</u>するものの中から，訓練サンプルと識別超平面との最小距離を最大にするように決める．
- 目的関数：
  - $\frac{\boldsymbol{1}}{\boldsymbol{2}}\|\boldsymbol{w}\|^2$ →最小化
- 制約条件：
  - $y_i(w^Tx_i+b)\geq 1, i=1,\cdots,l$

このまま解くのは難しいなー

学習の定式化(4)

- 双対問題：ラグランジュ乗数法
  - ラグランジュ乗数$\alpha_i(\geq 0)$を導入

$$L(\boldsymbol{w},b,\boldsymbol{\alpha})=\frac{1}{2}\|\boldsymbol{w}\|^2-\sum_{i=1}^{l}\alpha_i(y_i(\boldsymbol{x}_i{}^T\boldsymbol{w}+b)-1)$$

  - $\boldsymbol{w},b$に関してLを最小化し，$\boldsymbol{\alpha}$に関して最大化
  - 最適解ではLの勾配が0

$$\frac{\partial L}{\partial b}=0,\frac{\partial L}{\partial \boldsymbol{w}}=0\to\sum_{i=1}^{l}\alpha_i y_i=0\,,w=\sum_{i=1}^{l}\alpha_i y_i x_i$$

学習の定式化(5)

- 双対問題：$\boldsymbol{\alpha}$のみの最大化問題
  - 目的関数：

$$\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j x_i x_j$$

  - 制約条件：

$$\alpha_i\geq 0(i=1,\cdots,l),\sum_{i=1}^{l}\alpha_i y_i=0$$

$$\alpha_i\neq 0\to Support\ Vector$$

学習の定式化(6)

- 判別関数

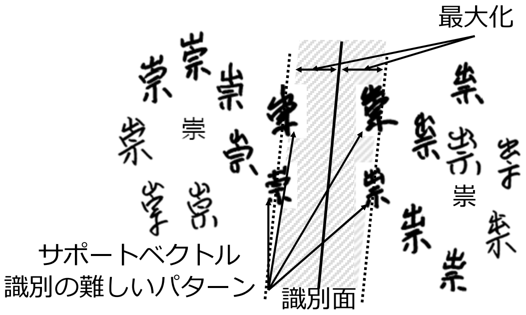
$$f(\boldsymbol{x})=\boldsymbol{w}^T\boldsymbol{x}+b=\sum_{i=1}^{l}\alpha_i y_i(\boldsymbol{x}\cdot\boldsymbol{x}_i)+b$$

$$b=\frac{1}{2}(\boldsymbol{x}_{circle}+\boldsymbol{x}_{square})$$

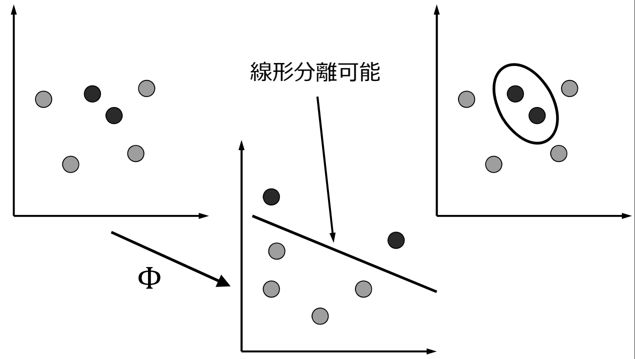
  - $\boldsymbol{x}_{circle},\boldsymbol{x}_{square}$: 丸，四角に属するSupport Vectors

$$\mathrm{sign}\left(\sum_{i=1}^{l}\alpha_i y_i(\boldsymbol{x}\cdot\boldsymbol{x}_i)+b\right)$$

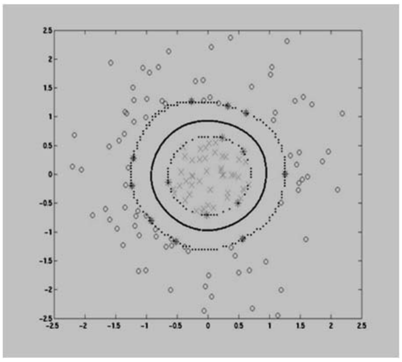
SVMのイメージ



SVMの非線形化(1)



SVMの非線形化(2)



カーネル：
RBFカーネル
$\sigma = 1.0$

実線：識別面
点線：$f(x) = \pm 1$

チュートリアルの内容

- サポートベクトルマシン(SVM)って何してるの？
  - SVMの学習って？
  - SVMをちょっと定式化？
- AdaBoostって何してるの？
  - AdaBoostのアルゴリズムって？
  - AdaBoostの学習って？

AdaBoost アルゴリズム(1)

- 入力:
  - l個の訓練サンプル$\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i), i = 1, \cdots l$
    - ただし，$y_i = \pm 1$
- 初期化:
  - 各訓練サンプルの加重を$w_1(z_i) = {}^1\!/_l$に設定
    - サンプル全体で各サンプルがどの程度存在するか
- 繰り返し$(t = 1, \ldots, T)$:
  - 訓練サンプル加重w_tに従って例題を学習
  - t番目の学習仮説$h \rightarrow \{\pm 1\}$を生成

AdaBoost アルゴリズム(2)

- 学習仮説h_tの学習誤差を計算

$$\varepsilon_t = \sum_{i=1}^{l} w_t(\boldsymbol{z}_i) \frac{I(y_i \neq h_t(\boldsymbol{z}_i)) + 1}{2}$$

  - ただし，$I(true) = +1, I(false) = -1$ なる関数
  - 例題を学習できなかった程度
- 学習仮説h_tの加重を計算

$$\alpha_t = \frac{1}{2}\left(\log\frac{\varepsilon_t}{1-\varepsilon_t} - \log\frac{\phi}{1-\phi}\right)$$

  - 学習仮説h_tの発言力
  - オリジナルのAdaBoostでは，$\phi = {}^1\!/_2$

AdaBoost アルゴリズム(3)

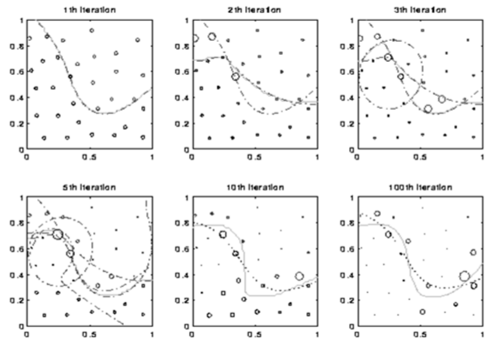
- 各サンプルの加重を更新

$$w_{t+1}(\boldsymbol{z}_i) = \frac{w_t(\boldsymbol{z}_i)\exp\{\alpha_t I(h_t(\boldsymbol{x}_i) \neq y_i)\}}{\sum_{j=1}^{l} w_t(\boldsymbol{z}_j)\exp\{\alpha_t I(h_t(\boldsymbol{x}_j) \neq y_j)\}}$$

  - 学習仮説h_tが苦手とするサンプルの加重を増加
  - 新たなサンプル加重を生成
- 出力(T 回の繰返しの後):
  - 最終学習仮説$f(\boldsymbol{x})$を生成

$$f(\boldsymbol{x}) = \frac{1}{\|\boldsymbol{\alpha}\|_1}\sum_{t=1}^{T}\alpha_t h_t(\boldsymbol{x})$$

AdaBoost 学習例



AdaBoost アルゴリズム(4)

- AdaBoostアルゴリズムの特徴
  - 学習誤差$\varepsilon_t < \frac{1}{2}$の学習仮説を生成
  - 有限回の学習仮説生成で，学習誤差0の最終学習仮説を生成
- 例えば…受験英単語の暗記
  - 暗記できなかった単語を注視して，暗記を繰り返すといった手順で，得意な英単語は繰り返しの中で適当に学習し，学習しずらい英単語をより注視して，相対的な学習のレベルを一致させる努力をすることにより，英単語集全ての英単語を暗記する学習と一致

AdaBoost の学習(1)

- 一つの入出力の組$\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i), i = 1, \cdots l$ に対してmargin$\rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$を次のように定義

$$\rho(\boldsymbol{z}_i, \boldsymbol{\alpha}) = y_i \sum_{t=1}^{T} \frac{\alpha_t}{\|\boldsymbol{\alpha}\|_1} h_t(\boldsymbol{x}_i) = y_i f(\boldsymbol{x}_i)$$

$$\rho(\boldsymbol{z}_i, \boldsymbol{\alpha}) \in [-1,1]$$

  - $h_t(\boldsymbol{x}_i)$はt 番目の学習仮説に入力ベクトル$\boldsymbol{x}_i$が入力された際の出力クラス
  - このmarginはSVMのmarginと同等

AdaBoost の学習(2)

- fに対する最小marginρの定義

$$\rho = \min_{z_i} \rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$$

  - fによる識別クラス

$$\mathrm{sign}\left(\sum_{t=1}^{T} \frac{\alpha_t}{\boldsymbol{\alpha}} h_t(\boldsymbol{x}_i)\right) = \mathrm{sign}(f(\boldsymbol{x}_i))$$

  - margin $\rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$ の値が正→正しい分類
  - margin $\rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$ の値が大→正しさが増加

受験時の英単語の暗記と同じ

AdaBoost の学習(3)

- AdaBoostはmargin $\rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$ の関数$G(\boldsymbol{\alpha})$を最小化する過程

$$G(\boldsymbol{\alpha}) = \sum_{i=1}^{l} \exp\left\{-\|\boldsymbol{\alpha}\|_1(\rho(\boldsymbol{z}_i, \boldsymbol{\alpha}) - (\frac{1}{2} - \phi))\right\}$$

- 例題サンプルの更新された加重$w_{t+1}(\boldsymbol{z}_i)$ は$G(\boldsymbol{z}_i)$のmargin $\rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$ に対する勾配

$$w_{t+1}(\boldsymbol{z}_i) = \frac{\partial G(\boldsymbol{\alpha}^t)/\partial\rho(\boldsymbol{z}_i, \boldsymbol{\alpha}^t)}{\sum_{j=1}^{l} \partial G(\boldsymbol{\alpha}^t)/\partial\rho(\boldsymbol{z}_j, \boldsymbol{\alpha}^t)}$$

AdaBoost の学習(4)

- 学習仮説h_tの加重α_tは$G(\boldsymbol{\alpha}^t)$を最小化するように選択

$$\alpha_t = \underset{\alpha_t}{\operatorname{argmin}}\, G(\boldsymbol{\alpha}^t)$$
$$= \underset{\alpha_t}{\operatorname{argmin}} \sum_{i=0}^{l} \exp\left\{-\|\boldsymbol{\alpha}\|_1 (\rho(\boldsymbol{z}_i, \boldsymbol{\alpha}) - (\frac{1}{2} - \phi))\right\}$$
$$= \frac{1}{2}\left(\log\frac{\varepsilon_t}{1-\varepsilon_t} - \log\frac{\phi}{1-\phi}\right) \leftarrow \frac{\partial G(\alpha^t)}{\partial \alpha_t} = 0$$

AdaBoost の学習(5)

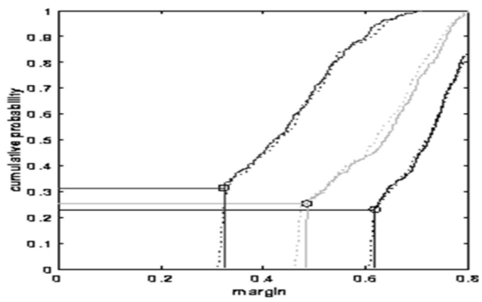
- $w_{t+1}(\boldsymbol{z}_i)$は損失関数$G(\boldsymbol{\alpha}^t)$の勾配方向
- α_tは損失関数を最小化するステップ幅
  - 基本学習アルゴリズムが与える仮説空間での gradient descent法に一致
  - 損失関数$G(\boldsymbol{\alpha})$をmargin$\rho(\boldsymbol{z}_i, \boldsymbol{\alpha})$をパラメータとして最小化
- $w_{t+1}(\boldsymbol{z}_i)$の更新→パラメータ$\|\boldsymbol{\alpha}^t\|_1$を有するSoftmax関数の学習
  - Softmax関数の学習：アニーリング過程

AdaBoost の学習(6)

- パラメータ$\|\boldsymbol{\alpha}^t\|_1$小→各訓練サンプル均等な加重
- パラメータ$\|\boldsymbol{\alpha}^t\|_1$大→識別の難しい訓練サンプルに大きな加重
  - 極限では最も識別の難しい訓練サンプルのみを考慮
- 例えば…受験英単語の暗記
  - 1回目の単語学習で暗記できた単語が，2回目の単語学習後に暗記できていない．英単語の暗記の場合，最初，2回目，3回目くらいは，暗記と共に暗記しづらい単語の見極めを行っている．つまり，初期段階で暗記できた，できなかった単語が，必ずしも本当に暗記しやすい，しづらい単語とは一致しない．

AdaBoost の学習(7)

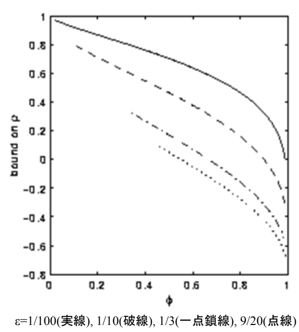
- 繰返し回数Tを大きくしたら



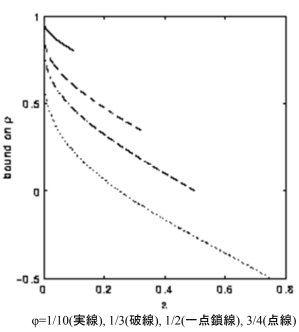
AdaBoost の学習(8)

- 最小marginρの下界
  - 最小marginってどんな値になるの?
  - $T \to \infty$で，
$$\rho \geq \frac{\ln(\phi\varepsilon^{-1}) + \ln((1-\phi)(1-\varepsilon)^{-1})}{\ln(\phi\varepsilon^{-1}) - \ln((1-\phi)(1-\varepsilon)^{-1})}$$
  - $\varepsilon = \max_t \varepsilon_t$であり，$\phi > \varepsilon$を満たす．

AdaBoost の学習(9)



ε=1/100(実線), 1/10(破線), 1/3(一点鎖線), 9/20(点線)

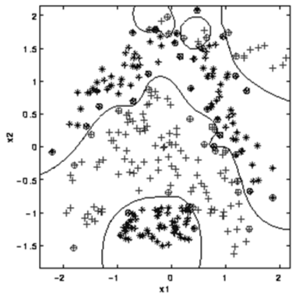


φ=1/10(実線), 1/3(破線), 1/2(一点鎖線), 3/4(点線)

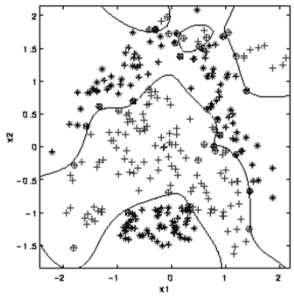
AdaBoost の学習(10)

- 例えば…(受験英単語の暗記)
  - 暗記力の優れた学生(εが小さい)の方が，そうでない学生(εが大きい)より確実な暗記を達成できる．また，学習レベルの目標を高くした方が(ϕを小さくした場合)，そうでない場合(ϕを大きくした場合)より確実な暗記を達成できる．しかし，自分の学習能力εを過信し，目標学習レベルϕを自分の能力より高くすると，学習が効率的にならないことを示している．

AdaBoost と SVM



AdaBoost's decision line



SVM's decision line

まとめ

- サポートベクトルマシン(SVM)って何してるの？
  - SVMの学習
  - SVMをちょっと定式化
- AdaBoostって何してるの？
  - AdaBoostのアルゴリズム
  - AdaBoostの学習
  - AdaBoostとSVMの比較

ご清聴ありがとうございました

onoda@ise.aoyma.ac.jp