SVMの説明書
～理論編～

@salinger01101

今回の目的

SVMをきちんと使えるようになる

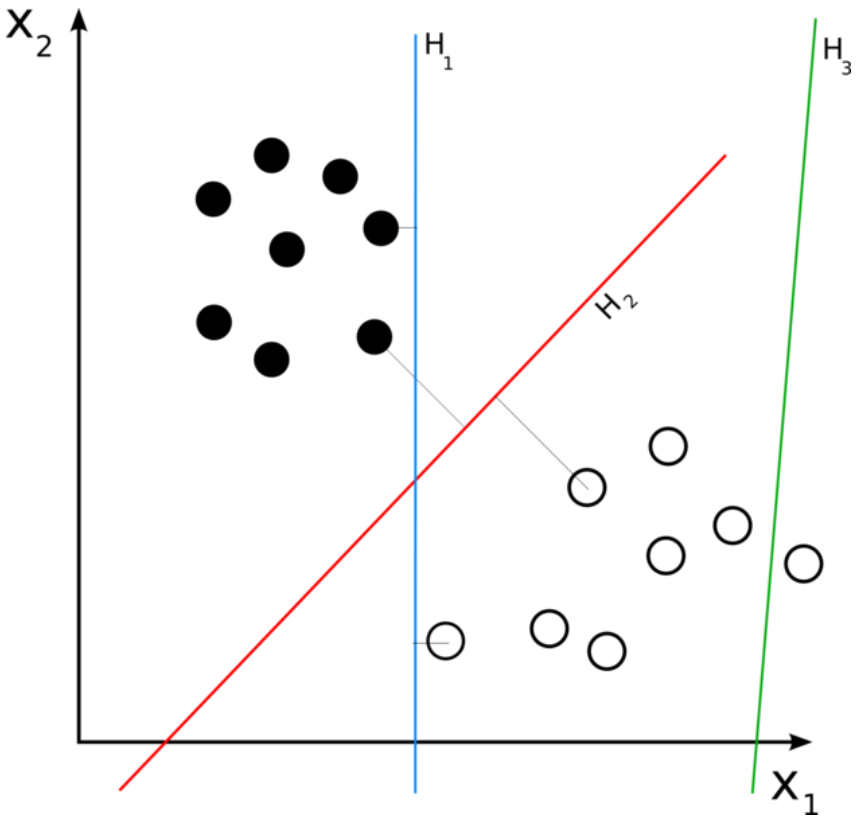
目次

- SVMの理論
- 最適化

SVMの理論

SVMとは？

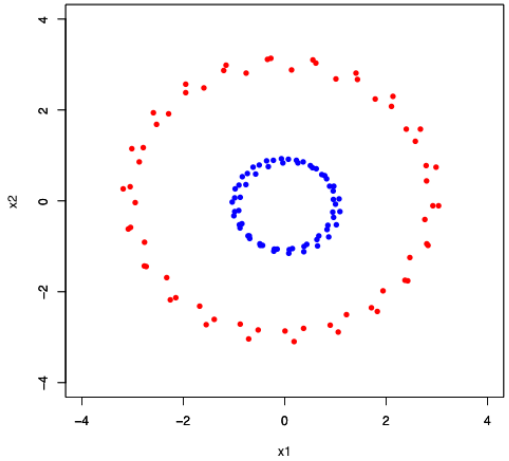
- **SVM（サポートベクターマシン）**
  - 教師あり学習
  - マージン最大化学習を行う2値分類器
  - この図のように、2つのグループ間のマージンが最大になるようなH_2を決定する。
  - 初期はこのような線形分類にしか適用できなかった。



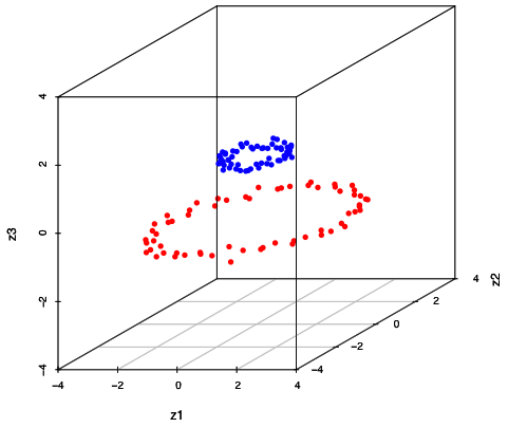
カーネル法の話

- **写像された高次元空間における内積を計算する関数**
  - 非線形なカーネル関数により、非線形な識別関数を学習可能。
  - カーネル関数を取り入れた一連の手法では、どのような写像が行われるか知らずに計算できることから、カーネルトリックと呼ばれている。

線形分離不可能



線形分離可能



カーネル関数の例

- 線形
（Linear）

$$K(x_i, x_j) = x_i^T x_j$$

- 多項式
（Polynomial）

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^{\mathrm{d}} \quad (\gamma > 0)$$

- RBF: “Gaussian”
（Radial Basis Function）

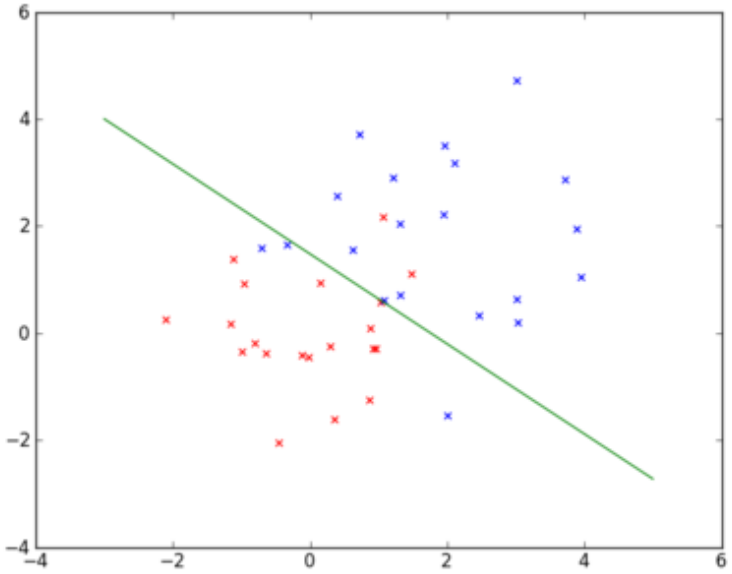
$$K(x_i, x_j) = \exp(-\gamma \left\| x_i^T - x_j \right\|^2) \quad (\gamma > 0)$$

- シグモイド(Sigmoid)

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$$

マージンの話

- **実際に問題を解く際には、どの程度誤りを許容するかが問題となる。**
  - コストパラメータ:C で決定する。
    - C が大きい ⇒ 誤りを許容しない
    - C が小さい ⇒ 誤りを許容する



厳密にやりすぎると、汎化能力（未知のものに対する予測性能）が低下する。

分類精度を向上させるためには適当さも必要

SVMのモジュール

- **LIBSVM**
  - SVMのモジュール。基本的にはこれで問題ない。
  - 各言語用のバインディングあり。
  - http://www.csie.ntu.edu.tw/%7Ecjlin/libsvm/
- **LIBLINEAR**
  - 線形カーネルのみだが、高速。
  - 各言語用のバインディングあり。
  - http://www.csie.ntu.edu.tw/~cjlin/liblinear/
- **SVMLight**
  - 大きなデータセットを高速に処理可能。
  - http://svmlight.joachims.org

最適化

SVMのダメな使い方

1. データをSVMで使えるように整形。
2. デフォルトのパラメータで試行。
3. 「精度悪いなぁ…」
4. 適当に選択したカーネルとパラメータで試行。
5. 「精度悪いなぁ…別の手法試すか…」

ダメ！絶対！

BETTERな手順

1. データをSVMで使えるように整形。
2. 素性の選択
3. データのスケーリング。
4. RBFカーネルを利用。
5. 交差検定により、最適なコストパラメータ：CとRBFカーネルの γ パラメータを調べる。
6. 最適なパラメータを用いて、モデルの生成を行う。
7. テストデータで試行。

素性の選択(1)

- **数値データ**
  - そのまま使用　　(10.0, 2.5, 6.4, -8.2)
  - 範囲ごとに分割　(10, 1, 5, -10)
  - バイナリ化　　(1, 1, 1, 0)
- **テキストデータ**
  - 単語の出現回数（n-gram）
  - 品詞情報　等
- **画像・音声データ**
  - 元データをそのまま行列に
  - フィルタリング
  - 圧縮して単純化
  - フーリエ変換・ウェーブレット変換 等

素性の選択(2)

- **次元の呪い(curse of dimensionality)**
  - 超高次元になるとモデルが複雑になりすぎ、学習データ不足になる。
  - 球面集中現象により，次元の増加に伴って，いろいろなデータ間の距離が互いに等しくなっていく。
  - まとめられるものはまとめる ⇒ 特徴選択・次元削減
    - Ex. 単語の出現回数
      - そのまま使用せず、何らかの手法で事前にグルーピング。
- **SVMは素性を実数値として扱う**
  - n種類の値を取る素性 ⇒ n個のバイナリ素性に
    - Ex. {red, green, blue}
      - (0), (1), (2) とせずに、(1,0,0), (0,1,0), (0,0,1) としたほうが結果が安定する事が多い。
- **素性の選択は経験がモノを言うので、事前にどのような素性を選べば良いかしっかり調査するの大事。**

スケーリングの話

- **SVMでは、値のとりうる範囲が大きい素性が支配的になる。**
  - 正規化したほうが良い結果になる場合がある。
    - Ex. 0 <= x <= 1 or -1 <= x <= 1

- **情報落ち誤差を防ぐためにも必要。**
  - 基本的なカーネル関数では素性ベクトルの内積計算を用いるので、スケーリングを行なわないと誤差が発生する恐れあり。

モデル選択の話

1. カーネル関数の決定
2. パラメータの決定
   （コストパラメータ ＆ カーネルパラメータ）

カーネル関数の決定（1）

- **最初に試すのはRBFカーネルが無難**
  - 高次元の非線形空間
  - 線形カーネルはRBFカーネルの特殊系
  - シグモイドカーネルもRBFカーネルとほぼ同じように動作
  - γパラメータ ＋ Cパラメータ のみの調整で良い

- **じゃ他のカーネルを使う場合はあるの？**

カーネル関数の決定（2）

- **線形カーネルの利点**
  - LIBSVMの代わりに高速なLIBLINEARが使用出来る。
  - Cパラメータのみの調整で良い

1. **事例数 << 素性数 の場合**
   - 素性が高次元なので写像する必要がない。
   - 線形カーネルを使うべき
2. **事例数 >> 素性数 の場合**
   - 非線形カーネルを利用して高次元に写像すべき。
3. **事例数も素性数も大きい の場合**
   - 学習に時間がかかる。LIBSVMが苦手なケース。
     - 線形カーネル & LIBLINEARの利用を検討。

パラメータの決定

- **RBFカーネルを使用する場合**
  - コストパラメータ：C
  - カーネルパラメータ：γ

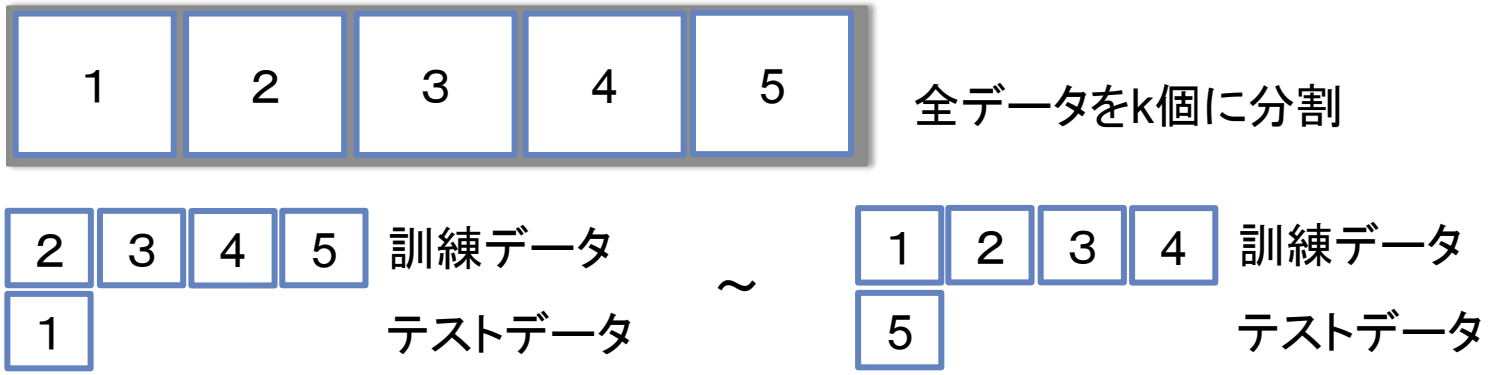
  を調整しなければならない。

- **最適なパラメータは？**
  - 交差検定
  - グリッドサーチ

  を利用して決定する。

交差検定

- 訓練データとテストデータの分割方法

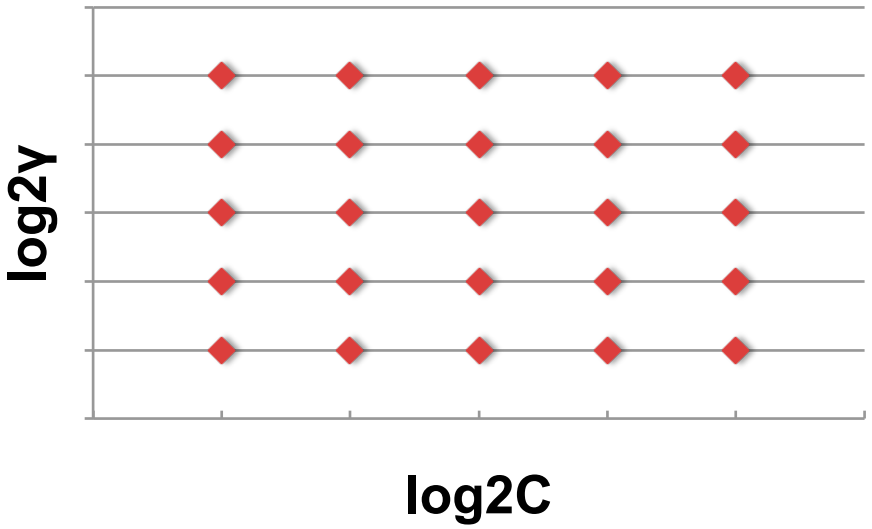


このようにk回試行し、その平均を利用する。

こうすることで、テストデータは常に未知のデータになる。
⇒過剰適応（Over fitting）を防げる

グリッドサーチ

- **2種類のパラメータを網羅的に探索**
  - グラフの赤い点を網羅的に試す。
    - 荒い探索の後、細かい探索。
  - 指数増加列がよい。
    - Ex. $C = 2^n$ (n = -5 ~ 15), $\gamma = 2^m$ (m = -15 ~ 3)



BETTERな手順（再掲）

1. データをSVMで使えるように整形。
2. 素性の選択
3. データのスケーリング。
4. RBFカーネルを利用。
5. 交差検定により、最適なコストパラメータ：CとRBFカーネルの γ パラメータを調べる。
6. 最適なパラメータを用いて、モデルの生成を行う。
7. テストデータで試行。

まとめ

- 素性の選択大事！
- 迷ったらRBF！
- スケーリングとパラメータ調整大事！

参考文献

・SVM実践ガイド (A Practical Guide to Support Vector Classification)

http://d.hatena.ne.jp/sleepy_yoshi/20120624/p1

・カーネル法

http://www.eb.waseda.ac.jp/murata/research/kernel

・TAKASHI ISHIDA　HomePage SVM

http://www.bi.a.u-tokyo.ac.jp/~tak/svm.html

・使用したソースコード等

https://github.com/Salinger/iris-svm