

情報意味論 (課題3)

慶應義塾大学理工学部
櫻井 彰人

レポート課題3

- レポートは、前回同様、web を通じて行ってください。
- レポートは電子的に作成してください。TeX, MsWord で作成して結構です。提出は pdf 形式でも結構です。
- 書くべき内容に関しては特には述べません。すべて常識的に判断してください。
- 締め切りは、1/30 一杯とします。
- 万が一の緊急の連絡のため、**電子メールアドレス**を書いて下さい。
- 課題1、課題2について修正・追記されたい方は、再提出して下さい。

3.1 SVM

2次元のデータを考える。
SVM分類器で、カーネル関数を使い、中心(0,0), 半径1の円の内部の点と外部の点を分ける方法を考える。

- まず、非線形関数を用いて、高次元の特徴空間へ写像しよう。
この状況に適した、最も簡単な非線形関数は何か? A hint: 3次元への写像
- それに対応するカーネル関数を定めよ A hint: 上記の関数の内積
(計算が簡単になるように定める必要はない。機械的に定めて下さい)
- 高次元の特徴空間に写像したあとの、分離超平面を1つあげよ。
- 「中心(0,0)、半径1の円」を考えたが、これを「中心(0,0)、長軸(x軸としよう)の長さ4、短軸(y軸としよう)の長さ2の楕円であったら、円の場合と同じカーネル関数は使えるか?
- 使えないなら、使えるカーネル関数は何か。

A hint: 上記(1)(2)の手順

3.1 SVM

- (6) 上記1つまたは2つのカーネル関数が目論見通り機能するか、試してみよ。
なお、比較のため、
- ・線形カーネル
 - ・RBFカーネル
 - ・多項式カーネル(2次)
- A hint: 学習誤りが0になればよいを試してみよ。

Rの kernlab パッケージを使うことにする。
なお、以下のプログラム例のデータは、上記の問題が要求する、円で2クラスに分離されたものではない。
<http://cran.r-project.org/web/packages/kernlab/vignettes/kernlab.pdf>
のplotの項にあるプログラム例を少し手直ししたものである。

3.1 SVM

```
library(kernlab)
set.seed(123)
x <- rbind(matrix(rnorm(120), , 2), matrix(rnorm(120, sd=4), , 2))
y <- matrix(c(rep(1, 60), rep(-1, 60)))
svp <- ksvm(x, y, type = "C-svc", kernel="rbfdot")
plot(svp, data = x)
```

A hint: kernelやデータを変えてこのプログラムを実行してみればよい

プロット結果の諧調は、svm で予測した時のdecision valueに従っている。
decision valueは、
predict(svp, x, type="decision")
とすれば取り出せる。なお、出力クラスは、単に、
predict(svp, x)
とすれば得られる。
定義済のカーネル関数(線形等)については、上記マニュアルを参照されたい。

3.1 SVM

カーネル関数のユーザ定義は、次のように行う。
なお、定義するカーネル関数の引数には、それぞれ、今回の例では、長さ2(次元数)のベクトルが渡される。

```
k2 <- function(x, y) { カーネル関数値(スカラー)を計算する }
class(k2) <- "kernel"
set.seed(200)
x <- rbind(matrix(rnorm(120, sd=1), , 2), matrix(rnorm(120, sd = 4), , 2))
y <- matrix(c(rep(1, 60), rep(-1, 60)))
svp <- ksvm(x, y, type = "C-svc", kernel=k2 )
plot(svp, data = x)
decisionValues <- predict(svp, x, type="decision")
```

A hint: 仮引数の x, y は共に2次元のベクトル(要素2個の1次元配列)である

3.2 frequent itemsets

- 次のデータに対し、Aprioriアルゴリズムを適用し k-frequent itemsets を得て下さい。
 - Minimum support は 40% として下さい。
 - 5-frequent itemsets は空になるはずです。

取引ID	トロ	酢	醤油	茶	米
1	1	1	1	0	0
2	1	1	1	1	1
3	1	0	1	1	0
4	1	0	1	1	1
5	1	1	1	1	0

3.3 簡単な証明

- Adaboost の訓練誤差の上限を証明してください。すなわち、次の式を証明してください。

$$\text{training error}(H_{\text{final}}) \leq \exp\left(-2 \sum_i \gamma_i^2\right)$$

- 講義のスライドの式変形の部分を埋めれば結構です。当然ながら、式の個数があう必要は全くありません。
 - ヒントは、スライド中にある式です