

第6章 線形判別分析による手書き文字 認識

6.1 手書き数字データの読み込みと表示

6.2 線形判別分析の実装

6.3 多カテゴリ文字認識の結果評価

07t4072f 岡崎 駿

目的

- ・線形判別分析(P61参照)をOctaveに実装
- ・その有用性を手書き文字認識を通じて説明

※今回はWeb上で本書から配布されている手書き数字データファイル'digit.mat'を使用

手書き数字データの読み込みと表示 (1)

- ・ファイル読み込み

```
> load digit.mat
```

- ・変数の内容確認

```
> who -variables -long
```

手書き数字データの読み込みと表示 (2)

確認結果

```
> who -variables -long
```

```
*** local user variables:
```

Prot	Name	Size	Bytes	Class
====	=====	====	=====	=====
rwd	T	256x200x10	4096000	matrix
rwd	X	256x500x10	10240000	matrix
rw-	__margin__	1x1	8	scalar

入力データの内容

- ・変数X, 変数Tについて
 - 3階のテンソル(引数を3つ取る配列)
 - 1つの手書き文字データが256次元(16×16画素)
 - 要素は-1~1の間(-1が黒、1が白)
- ・変数Xには'0'から'9'の文字が各500文字
- ・変数Tには'0'から'9'の文字が各200文字

文字データの参照と画像表示

- 23番目の訓練用手書き数字'5'のデータを変数xに取り出す

```
> x=X(:,23,5);
```

- 取り出した手書き文字データの画像の表示

```
> imagesc(reshape(x,[16 16]))
```

線形判別分析の実装(1)

目的

手書きの'1'と'2'を分類する文字認識器の作成

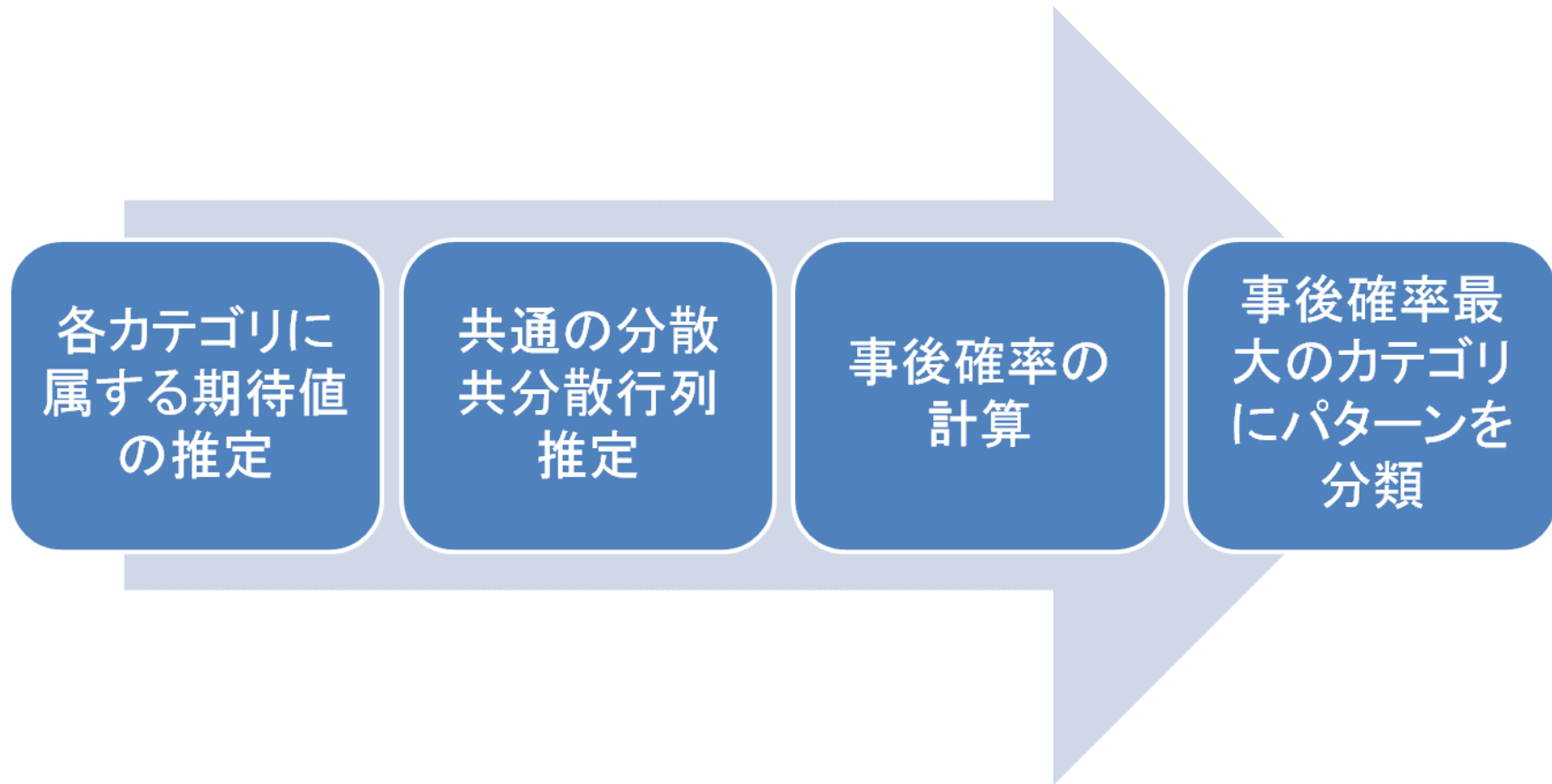
仮定

'1'と'2'の各カテゴリに属するパターンが
期待値は異なるが、
分散共分散行列が等しい正規分布に
それぞれ独立に従う

$$\{\mathbf{x}_i\}_{i:y_i=1} \stackrel{i.i.d.}{\sim} N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}), \{\mathbf{x}_i\}_{i:y_i=2} \stackrel{i.i.d.}{\sim} N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$$

線形判別分析の実装(2)

- 実装プロセス



線形判別分析の実装(3)

Octaveの関数について

- `mean(x)`: x の平均値を計算
- `cov(x)`: 共分散を計算
- `inv(x)`: x の逆行列
- `sign(x)`: 符号関数を計算
- `find(x)`: x を満たす配列要素を返す
- `sum(x)`: x の全体の合計(答えはスカラー)
- `x'`: x の転置行列

事後確率の計算(1)

1.各カテゴリの訓練標本の平均の推定

```
> mu1=mean(X(:,1),2);  
> mu2=mean(X(:,2),2);
```

2.両方のカテゴリに共通の分散共分散行列の推定

```
> S=(cov(X(:,1)')+cov(X(:,2)'))/2;
```

事後確率の計算(2)

- 3.「訓練標本の平均」、「分散共分散行列」より、あるテストパターン(1番目)に対する、各カテゴリの「事後確率」を計算

```
> t = T(:,1,2);  
> invS = inv(S);  
> p1 = t' * invS * mu1 - mu1' * invS * mu1 / 2  
> p2 = t' * invS * mu2 - mu2' * invS * mu2 / 2
```

線形判別分析による識別結果

4. それぞれの事後確率の差の符号を調べる

```
> sign(p1 - p2)  
ans = -1
```

→この例ではテストパターンはカテゴリ2(数字の2)に分類

→もし $ans = 1$ ならばカテゴリ1(数字の1)に分類

識別結果の考察(1)

- ・正しく識別された個数(カテゴリ2に分類された全ての識別結果)の検出 ※p1、p2は横ベクトル

```
> t = T(:, :, 2);  
> p1 = mu1' * invS * t - mu1' * invS * mu1 / 2  
> p2 = mu2' * invS * t - mu2' * invS * mu2 / 2  
> result = sign(p1 - p2);  
> sum(result == -1)  
ans = 198  
> sum(result ~= -1)  
ans = 2
```

識別結果の考察(2)

- ・ 誤認識したテストパターンの番号を知る

```
> find(result ~= -1)
ans =
    69    180
```

- ・ 目視の確認(69番目)

```
> imagesc(reshape(t(:,69),[16 16]))'
```

多カテゴリ文字認識の結果評価(1)

- 多カテゴリの文字認識のプロセスも2カテゴリの場合(本スライド8ページ)と同様である
- 実験結果は混合行列に表すと見通しが良い
 - 混合行列
 - カテゴリ y に属すべきパターンをカテゴリ y' に属すると判定した回数を (y, y') 要素に持つ行列

多カテゴリ文字認識の結果評価(2)

- 混合行列の例

> C			
C =			
	199	1	0
	0	192	8
	0	2	198

- '1'のテストパターンは199個が正しく認識、1個が'2'と認識
- '2'のテストパターンは192個が正しく認識、8個が'3'と認識
- '3'のテストパターンは198個が正しく認識、2個が'2'と認識