情報認識：Octaveによる手書き文字認識*

東京工業大学 計算工学専攻
杉山 将
sugi@cs.titech.ac.jp　*http://sugiyama-www.cs.titech.ac.jp/~sugi/*

平成18年11月27日

1　手書き数字データの読み込みと表示

まずは *http://sugiyama-www.cs.titech.ac.jp/~sugi/data/digit.mat* より，手書き数字のデータファイル digit.mat をダウンロードする．そして次のようにして，手書き数字のデータを読み込む．

```
> load digit.mat
```

そうすると，X と T という二つの変数が読み込まれる．X には訓練用の文字データが，T にはテスト用の文字データがそれぞれ格納されている．定義されている変数の一覧は who コマンドで確認できる．

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

X と T は3次元の超行列である．これは引数を3つ取る行列のことであり，3×3 行列ではない．一つの手書き文字データは256次元のベクトルで表されている．これは 16×16 画素の画像データを一列に並べたベクトルであり，それぞれの要素は -1 から1までの実数を取る．値が -1 のとき画素は黒で，値が1のとき画素は白である．変数 X には '0' から '9' からまでの各数字が500文字ずつ，変数 T には '0' から '9' からまでの各数字が200文字ずつ含まれている．例えば，23番目の訓練用の手書き数字 '5' のデータを変数 x に取り出すときは

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

とすればよい．手書き数字 '0' のデータは，3番目の引数が10の場合に対応するので注意すること．

取り出した手書き文字のデータの画像は，以下のようにして表示することができる．

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

2　演習1

手書きの '1' と '2' を分類する文字認識器を作ってみよう．まずは，メモリをクリアして，手書き文字のデータをファイルから読み直そう．

```
> clear all
> load digit.mat
```

ここでは，'1' と '2' の各カテゴリがそれぞれ分散共分散行列の等しい正規分布に従うと仮定し，テストパターン t が与えられたもとでの各カテゴリの事後確率 $p(\omega_i|t)$ を計算しよう．そのためには，各カテゴリの平均を推定する必要がある．これは次のようにして計算できる．

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

また，両方のカテゴリに共通の分散共分散行列も推定する必要がある．これは次のようにして計算できる．

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

これらを用いて，あるテストパターン t に対する各カテゴリの事後確率 $p(\omega_i|t)$ から定数を無視した値は

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

となる．inv(S) のところで

```
warning: inverse: matrix singular...
```

という警告が出るかもしれないが，とりあえずは無視することにしよう．テストパターンの識別には事後確率の差の符号を調べればよい．

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

この例では，テストパターンはカテゴリ2（即ち数字 '2'）に分類される．つまり，テストパターンは正しく認識されている．sign(pdiff) の値が1のときはテストパターンはカテゴリ1（即ち数字 '1'）に分類される．

カテゴリ2に属すると分かっているテストパターン全ての識別結果は

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

によって計算できる．今回は p1 と p2 は横ベクトルであることに注意せよ．結果をまとめると次のようになる．

*Octave のより詳しい説明は，例えば *http://www.obihiro.ac.jp/~suzukim/masuda/octave/hdbk/octhdbk.pdf* を参照せよ．

```
> sum(result==-1)
ans = 198
> sum(result~=-1)
ans = 2
```

即ち，200 個の手書きの ‘1’ のテストパターンのうち，198 個は正しく識別され，残りの 2 つは誤って識別されている．即ち 99%の正解率である．

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

とすることによって，誤識別されたのは 69 番目と 180 番目のテストパターンであることが分かる．これらのパターンがどのような文字であるかは，

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

などとすることにより確認できる．ちなみに 69 番目のテストパターンは人間が見ても誤識別しそうな文字である．

3 課題 1

上記の演習の設定の基づいて，カテゴリ 1 に属すると分かっているテストパターン全ての識別結果を調べよ．また，誤識別されたパターンがどのようなものかを確認せよ．

4 演習 2

次に，‘1’, ‘2’, ‘3’ の 3 つのカテゴリのデータを用いて，手書き数字認識を行なう．方法は演習 1 の場合とほぼ同様である．即ち，各カテゴリの平均と，共通の分散共分散行列を訓練パターンから推定し，テストパターンに対する各カテゴリの事後確率を計算する．そして，事後確率が最大のカテゴリにテストパターンを分類する．

実験結果は，**混同行列**(confusion matrix) にまとめると見通しがよい．混同行列とは，カテゴリ i に属すべきパターンをカテゴリ j に属すると判定した回数を (i, j) 要素に持つ行列である．

以下にサンプルプログラムを示す．

```
clear all
load digit.mat X T
[d,n,nc]=size(X);

nc=3;
S=zeros(d,d);
for c=1:nc
  mu(:,c)=mean(X(:,:,c),2);
  S=S+cov(X(:,:,c)')/nc;
end
invS=inv(S);

for ct=1:nc
  for c=1:nc
    muc=mu(:,c);
    t=T(:,:,ct);
    p(ct,:,c)=muc'*invS*t-muc'*invS*muc/2;
  end
end

[pmax P]=max(p,[],3);
for ct=1:nc
  for c=1:nc
    C(ct,c)=sum(P(ct,:)==c);
  end
end
```

これにより得られる混同行列は

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

となり，認識結果は以下のようにまとめられる．

- ‘1’ のテストパターンは 199 個が正しく ‘1’ と認識され，残りの 1 個は ‘2’ と認識された．
- ‘2’ のテストパターンは 192 個が正しく ‘2’ と認識され，残りの 8 個は ‘3’ と認識された．
- ‘3’ のテストパターンは 198 個が正しく ‘3’ と認識され，残りの 2 個は ‘2’ と認識された．

5 課題 2

‘0’ から ‘9’ 全てのカテゴリのデータを用いて手書き数字認識を行ない，認識結果を混同行列にまとめよ．

6 課題 3

今回の Octave 演習に対する感想，苦情，意見等を述べよ．よかった点，分かりにくかった点や改善すべき点は**できるだけ具体的に**述べよ．

7 小レポート（第 8 回）

上記の課題 1 から 3 をレポートにまとめ提出せよ．課題 3 も評価対象なのでしっかり記述すること．