

USPS 手書き数字認識データセットのパターン認識

手書き数字認識のデータセット USPS に対して、パターン認識を、K-NN 法 ($k = 1$)、マハラノビス法の二つで行った。

K-NN 法で作成したプログラムを図 1 に、マハラノビス法で作成したプログラムを図 2 に示す。

```
% K-nearest neighbor rule %
clear all; close all;

r=13;      % dimensionality of each subspace
imgnum=1;  % test sample number for displaying Uc
nclass=10; % number of classes

% loading data-file
load('./usps_resampled/usps.mat');
[d,ndata]=size(trai);

% normalization
for ii = 1 : ndata
    trai(:,ii)=trai(:,ii)./norm(trai(:,ii));
end

%K-NN 法をすべてのテストサンプルに行う
labels = zeros(4649, 1);
for n = 1 : 4649
    %テストサンプルと各訓練データのユークリッド距離を計算する
    D = zeros(1, 4649);
    for i = 1 : 4649
        D(1, i) = norm(test(:, n) - trai(:, i));
    end
    %もっともユークリッド距離の小さいクラスを求める
    [D_sorted, index] = sort(D);
    class_num = trai_label(index(1, 1), 1);
    labels(n, 1) = class_num;
end

%混同行列を作成する
conf_matrix = zeros(10, 10);
```

```

for j = 1 : 4649
    %行番号と列番号を取得する
    line_num = test_label(j, 1) + 1;
    column_num = labels(j, 1) + 1;
    %混同行列内の値を1増やす
    conf_matrix(line_num, column_num) += 1;
end

%クラス平均認識率を求める
A = diag(conf_matrix);
B = sum(conf_matrix, 2);
accuracy = A ./ B;
accuracy_mean = mean(accuracy);
printf('accuracy = %f\n', accuracy_mean);

```

図1 K-NN法のプログラム

```

% Mahalanobis Distance rule %
warning('off', 'all');
clear all; close all;

% loading data-file
load('./usps_resampled/usps.mat');
[d,ndata]=size(trai);

% normalization
for ii = 1 : ndata
    trai(:,ii)=trai(:,ii)./norm(trai(:,ii));
    test(:,ii)=test(:,ii)./norm(test(:,ii));
end

%各クラスのデータをまとめた行列を求める
class_count = zeros(1, 10);
class_data = zeros(256, 800, 10);
for k = 0 : 9
    for m = 1 : 4649
        if trai_label(m, 1) == k

```

```

        class_count(1, k + 1) += 1;
        class_data(:, class_count(1, k + 1), k + 1) = trai(:, m);
    endif
endfor
endfor

%マハラノビス法をすべてのテストサンプルに行う
labels = zeros(4649, 1);
for i = 1 : 4649
    D = zeros(1, 10);
    for n = 1 : 10
        %各クラスのデータで不要な列を削除する
        data = class_data(:, :, n);
        data(:, class_count(1, n) + 1 : 800) = [];
        %テストサンプルと各訓練クラスデータのマハラノビス距離を計算する
        inv_cov = inv(cov(data'));
        mean_class = mean(data, 2);
        D(1, n) = (test(:, i) - mean_class)' * inv_cov * (test(:, i) - mean_class);
    end
    %もっともマハラノビス距離の小さいクラスを求める
    [D_sorted, index] = sort(D);
    class_num = index(1, 1) - 1;
    labels(i, 1) = class_num;
end

%混同行列を作成する
conf_matrix = zeros(10, 10);
for j = 1 : 4649
    %行番号と列番号を取得する
    line_num = test_label(j, 1) + 1;
    column_num = labels(j, 1) + 1;
    %混同行列内の値を1増やす
    conf_matrix(line_num, column_num) += 1;
end

%クラス平均認識率を求める

```

```
A = diag(conf_matrix);  
B = sum(conf_matrix, 2);  
accuracy = A ./ B;  
accuracy_mean = mean(accuracy);  
printf('accuracy = %f\n', accuracy_mean);
```

図2 マハラビノス距離でパターン認識するプログラム

二つのアルゴリズムでは、混同行列を `conf_matrix`、クラス平均認識率を `accuracy_mean` として変数に与えた。

また、それぞれのプログラムを実行した結果、K-NN 法では、クラス平均認識率 0.9672、マハラノビス法では、クラス平均認識率 0.4736 となった。

この結果より、K-NN 法の方が優れた認識率を示したが、マハラノビス法では、想定よりも認識率がかなり小さくなってしまった。原因として、Octave GNU に条件数の逆数を示す `rcond` の値が 0 に近いという warning が出ていたことから、生成した各クラスのデータを集めた行列が正則だが、計算誤差の影響を大きく受けるため、正しい逆行列を `inv()` 関数を使って求められていないことが考えられる。