

---

## Table of Contents

Data preparation .....	1
Performing PCA .....	2
Minimum Distance Criterion .....	2
Bayes criterion .....	3

## Data preparation

```
clear all
close all
clc

load('arrhythmia.mat')

A=arrhythmia;

A(:, find(sum(abs(A)) == 0)) = []; % we erase the zero columns

class_id=A(:,end); % last vector of the matrix
class_id(find(class_id>1))=2; % all the values higher than 1 are put
equal to 2
y=A;
y(:,end)=[]; % we put in y all the features but the last one
[N,F]=size(y);

%normalizing y
mean_y=mean(y,1);
stdv_y=std(y,1);

o=ones(N,1);% o is a column vector
y=(y-o*mean_y)./(o*stdv_y);% y is normalized

mean_y=mean(y,1); % checking that y matrix is properly normalized
var_y=var(y,1);

save('arrhythmia_norm.mat','y')

% we divide patients in two classes: with and without arrhythmia
y1=y(find(class_id==1),:); % patients without arrhythmia
y2=y(find(class_id==2),:); % patients with arrhythmias

n_healthy=sum(class_id==1);
n_ill=sum(class_id==2);

% define the probabilities to fall in either one of the two regions
pi_1=n_healthy/N;
pi_2=n_ill/N;
```

---

# Performing PCA

```
R_y=y'*y/N;
[U, E] = eig(R_y);

P = sum(diag(E));
percentage = 0.999; % we set the percentage of information that we
    want to keep
new_P = percentage * P;

cumulative_P = cumsum(diag(E)); % function that evaluates the
    cumulative
                                % sum of each element of the diagonal
    of A
L = length(find(cumulative_P<new_P)); % determines the first L
    features
                                % that contribute to obtain new_P
                                amount
                                % of "information"

U_L = U(:,1:L); % we only consider the first L features

Z = y * U_L;
mean_Z=mean(Z,1); % Z is zero mean
Z=Z./((sqrt(var(Z)))); % we normalize Z
```

## Minimum Distance Criterion

```
% we divide the two classes
z1=Z(find(class_id==1), :);
z2=Z(find(class_id==2), :);

% finding the representative of the two classes
w1=mean(z1,1);
w2=mean(z2,1);

wmeans=[w1;w2];
enZ=diag(Z*Z'); % |Z(n)|^2
enW=diag(wmeans*wmeans'); % |w1|^2 and |w2|^2
dotprod_2=Z*wmeans'; % matrix with the dot product between each Z(n)
    and each w
[U2,V2]=meshgrid(enW,enZ);
dist_z=U2+V2-2*dotprod_2; % |y(n)|^2+|x(n)|^2-2y(n)x(k)=|y(n)-x(k)|^2

yhat_1=find(dist_z(:,1)<=dist_z(:,2));
yhat_2=find(dist_z(:,1)>dist_z(:,2));

n_false_negative=length(find(class_id(yhat_1)==2));
n_false_positive=length(find(class_id(yhat_2)==1));
n_true_negative=length(find(class_id(yhat_1)==1));
n_true_positive=length(find(class_id(yhat_2)==2));
```

---

```

p_true_positive=100*n_true_positive/n_ill; % 87.92
p_true_negative=100*n_true_negative/n_healthy; % 93.87
p_false_positive=100*n_false_positive/n_healthy; % 6.12
p_false_negative=100*n_false_negative/n_ill; % 12.07

p_strike=100*(n_true_positive+n_true_negative)/N % 91,15

figure
hold on
b=bar(1,p_strike);
b2=bar(2,p_true_positive,'r');
b3=bar(3,p_true_negative,'g');
b4=bar(4,p_false_positive,'y');
b5=bar(5,p_false_negative,'m');

title('Classification Results: Minimum distance criterion (with PCA)')
legend('pStrike','pTruePositive','pTrueNegative','pFalsePositive','pFalseNegative')

```

## Bayes criterion

```

onevar=ones(N,1);

pis=zeros(1,2);
pis(1)=pi_1;
pis(2)=pi_2;

bayes_dist=dist_z-2*onevar*log(pis);

% taking the decision
zhat_1=find(bayes_dist(:,1)<=bayes_dist(:,2));
zhat_2=find(bayes_dist(:,1)>bayes_dist(:,2));

n_true_negative_z=length(find(class_id(zhat_1)==1));
n_true_positive_z=length(find(class_id(zhat_2)==2));
n_false_negative_z=length(find(class_id(zhat_1)==2));
n_false_positive_z=length(find(class_id(zhat_2)==1));

p_true_positive_z=100*n_true_positive_z/n_ill; % 95,9184
p_true_negative_z=100*n_true_negative_z/n_healthy; % 84,5411
p_false_positive_z=100*n_false_positive_z/n_healthy; % 4,0816
p_false_negative_z=100*n_false_negative_z/n_ill; % 15,4589

p_strike_z=100*(n_true_positive_z+n_true_negative_z)/N % 90,70

%
mset=[p_strike,p_true_positive,p_true_negative,p_false_positive,p_false_negative;
% figure
% % c = categorical({'Minimum Distance' 'Bayesian criterion'});
% b=bar(mset);
% title('Minimum distance vs MAP criterion')
%
legend('pStrike','pTruePositive','pTrueNegative','pFalsePositive','pFalseNegative')

```

---

```
figure
hold on
b=bar(1,p_strike_z);
b2=bar(2,p_true_positive_z,'r');
b3=bar(3,p_true_negative_z,'g');
b4=bar(4,p_false_positive_z,'y');
b5=bar(5,p_false_negative_z,'m');

title('Classification Results: Bayesian criterion')
legend('pStrike','pTruePositive','pTrueNegative','pFalsePositive','pFalseNegative')
```

*Published with MATLAB® R2016a*