
MAT Consulting for Chem R US by Thomas Wagner, Alexander Allen, MingYi Wang

Table of Contents

DatasetA processing	1
define positive class and calculate mean and covariance	1
define negative class and calculate mean and covariance	4
Define testing and training sets	6
Mean Method on DatasetA	6
Fisher method on DatasetA	9
KNN classifier on Test set	12
FisherMedian DatasetA	13
DatasetV Analysis	16
DA ErrorsFisher	19

March 19 2015

DatasetA processing

```
%read in D
Doriginal = csvread('DatasetA.csv');

%Break D into id's, class, and features
IDA=Doriginal(:,1); %id column
Class=Doriginal(:,end); % Y contains the class labels 1 or -1
DA=Doriginal(:,2:(end-1)); % All the rest are the features
```

define positive class and calculate mean and covariance

```
DAP = DA(Class==1,:); %Class 1 of DSA;
[mp,np]=size(DAP);
DAP_mean = (1/np)*ones(1,np)*DAP

figure
imagesc(DAP_mean)
title('Mean Vector of Class 1 DatasetA')
colormap(gray)
colorbar

DAP_centered= DAP - (1/np)*(ones(np,np)*DAP);
```

```
CovAp = (1/(mp-1))*DAP_centered'*DAP_centered; %Covariance of class 1
```

```
figure
imagesc(CovAp)
title('Covariance Matrix of Class 1 DatasetA')
colormap(gray)
colorbar
```

DAP_mean =

Columns 1 through 7

4.4792	3.0673	0.1039	0.0056	0.2135	0.0337	0.5955
--------	--------	--------	--------	--------	--------	--------

Columns 8 through 14

34.4736	1.2219	2.2444	0.3792	-0.3183	3.1956	1.6539
---------	--------	--------	--------	---------	--------	--------

Columns 15 through 21

9.4908	3.6124	1.0073	1.1367	0	0.0056	0
--------	--------	--------	--------	---	--------	---

Columns 22 through 28

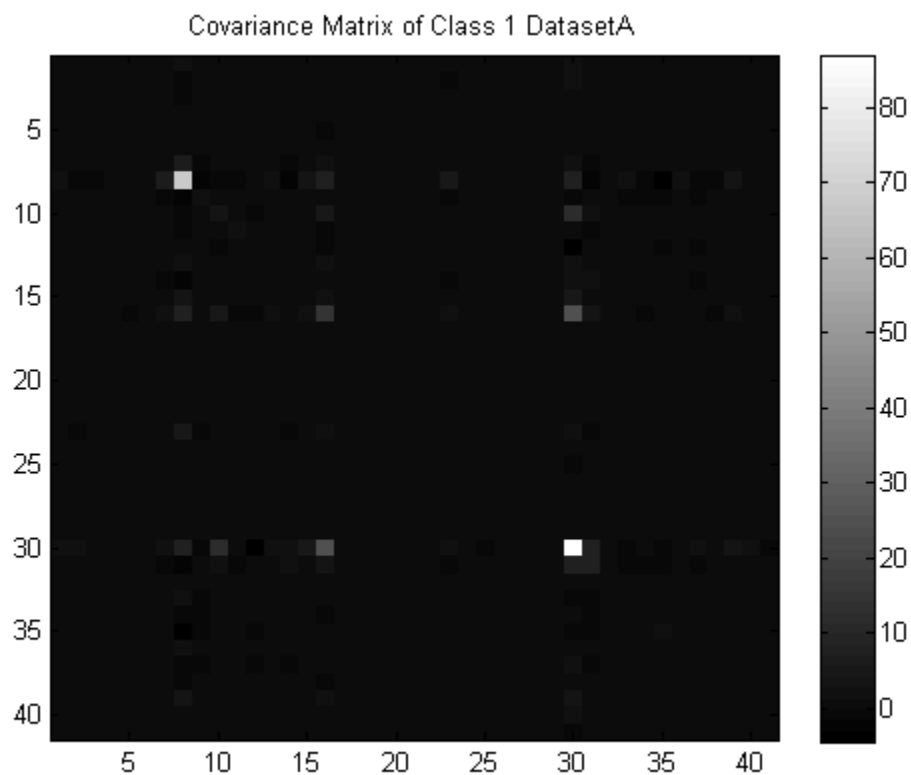
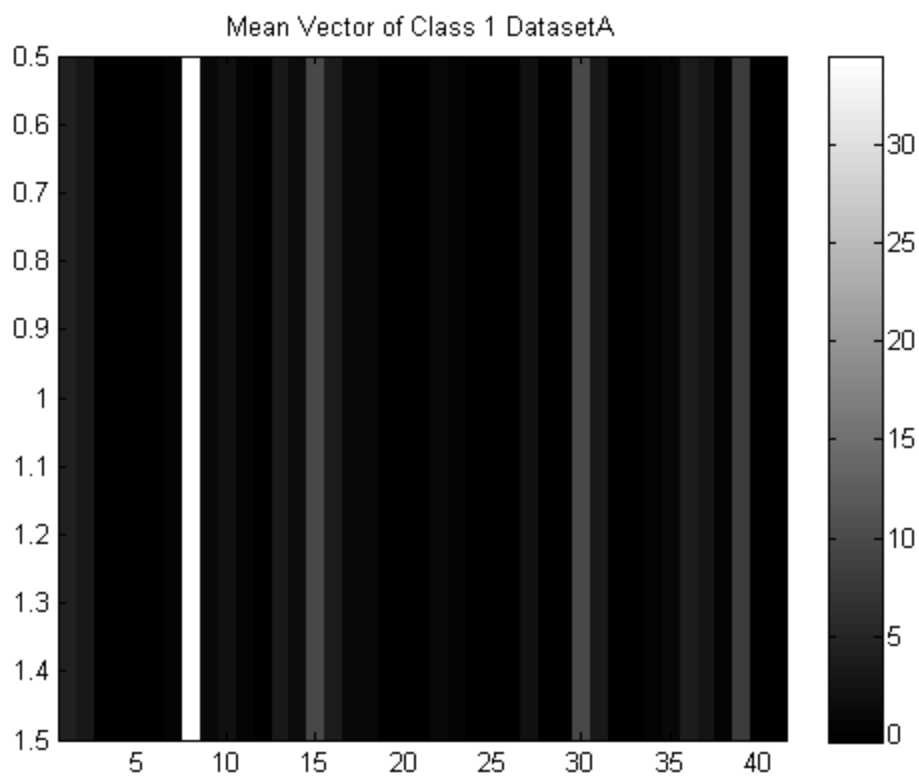
1.1884	0.6236	0.0084	0.0225	0.0056	2.0921	-0.0068
--------	--------	--------	--------	--------	--------	---------

Columns 29 through 35

0.0056	9.6742	3.1780	0.0337	0.2051	0.4185	0.9129
--------	--------	--------	--------	--------	--------	--------

Columns 36 through 41

3.5128	2.6619	0.2865	7.9915	0.1180	0.0506
--------	--------	--------	--------	--------	--------



define negative class and calculate mean and covariance

```
DAn = DA(Class== -1, :);  
[mn, nn]=size(DAn);  
DAn_mean=(1/mn)*ones(1, mn)*DAn  
  
figure  
imagesc(DAn_mean)  
title('Mean Vector of Class -1 DatasetA')  
colormap(gray)  
colorbar  
  
DAn_centered=DAn - (1/mn)*(ones(mn, mn)*DAn);  
  
CovAn = (1/(mn-1))*DAn_centered'*DAn_centered;  
  
figure  
imagesc(CovAn)  
title('Covariance Matrix of Class -1 DatasetA')  
colormap(gray)  
colorbar
```

DAn_mean =

Columns 1 through 7

4.9372	3.0706	1.0286	0.0615	1.3705	0.4206	2.1817
--------	--------	--------	--------	--------	--------	--------

Columns 8 through 14

38.3707	1.4549	1.5794	1.9757	-0.1354	3.6201	1.1963
---------	--------	--------	--------	---------	--------	--------

Columns 15 through 21

10.1648	3.6395	1.0163	1.1283	0.0129	0.1087	0.0443
---------	--------	--------	--------	--------	--------	--------

Columns 22 through 28

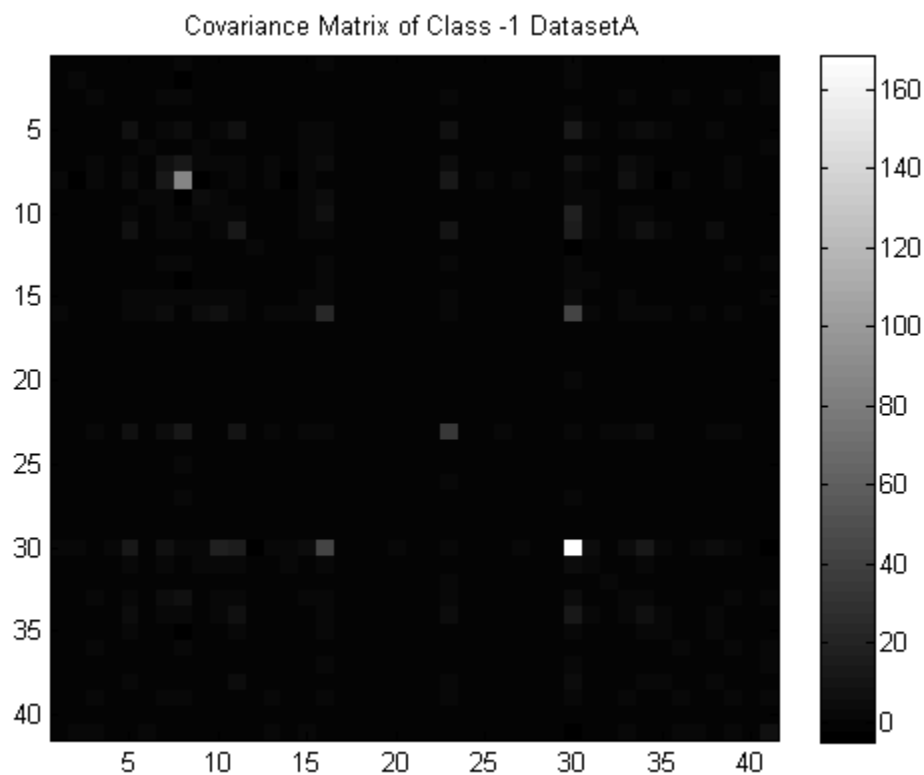
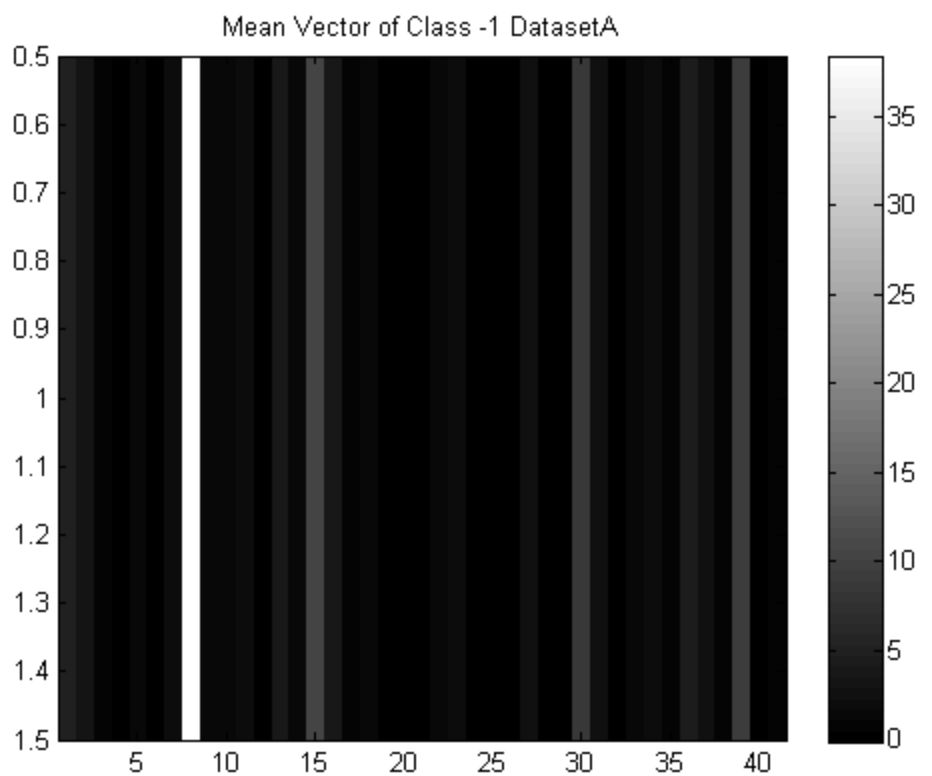
1.2643	1.8040	0.0558	0.2117	0.0443	2.2785	0.0016
--------	--------	--------	--------	--------	--------	--------

Columns 29 through 35

0.0372	8.3254	2.4088	0.1788	1.2289	1.7110	0.9857
--------	--------	--------	--------	--------	--------	--------

Columns 36 through 41

4.1247	2.5057	0.8898	8.9544	0.0172	1.0658
--------	--------	--------	--------	--------	--------



Define testing and trianing sets

```
% Training and testing matrices for DatasetA

% Classp_train  := Class 1 training data
% Classm_train  := Class -1 training data
% Classp_test   := Class 1 testing data
% Classm_test   := Class -1 testing data


% Set random number to an initial seed
[r,c]=size(DA);
s=RandStream('mt19937ar','Seed',550);
%generate a permutation of the data
p=randperm(s,r);
DA=DA(p,:);
Y=Class(p);
%Use trainpct percent of the data for training and the rest for testing.
trainpct=.90;
train_size=ceil(r*trainpct);

% Grab training and test data
Train = DA(1:train_size,:);
Test = DA(train_size+1:end,:);
YTrain = Y(1:train_size,:);
YTest = Y(train_size+1:end,:);

%Break them up into Class 1 and Class -1
Classp_train = Train(YTrain==1,:);
Classm_train = Train(YTrain==-1,:);

Classp_test = Test(YTest==1,:);
Classm_test = Test(YTest==-1,:);
```

Mean Method on DatasetA

```
%Calculate the mean classifier

% Calculate w as difference of the class means
meanp=mean(Classp_train);
meanm=mean(Classm_train);
w=(meanp-meanm)';
w=w/norm(w);

t= (meanp+meanm)./2*w;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Count the number of training points missclassified in each class.

MeanPosErrorTrain = sum(Classp_train*w <= t);
MeanNegErrorTrain = sum(Classm_train*w >= t);
```

```
MeanTrainError = ((MeanPosErrorTrain + MeanNegErrorTrain)/(size(Train,1)))
```

```
%Calculate the testing error of the Mean Method
```

```
MeanPosErrorTest = sum(Classp_test*w <= t);
```

```
MeanNegErrorTest = sum(Classm_test*w >= t);
```

```
MeanTestError = ((MeanPosErrorTest + MeanNegErrorTest)/(size(Test,1)))
```

```
%Histogram of Mean Training Results
```

```
HistClass(Classp_train,Classm_train,w,t,...  
  'Mean Method Training Results',MeanTrainError);
```

```
%Histogram of Mean Testing Results
```

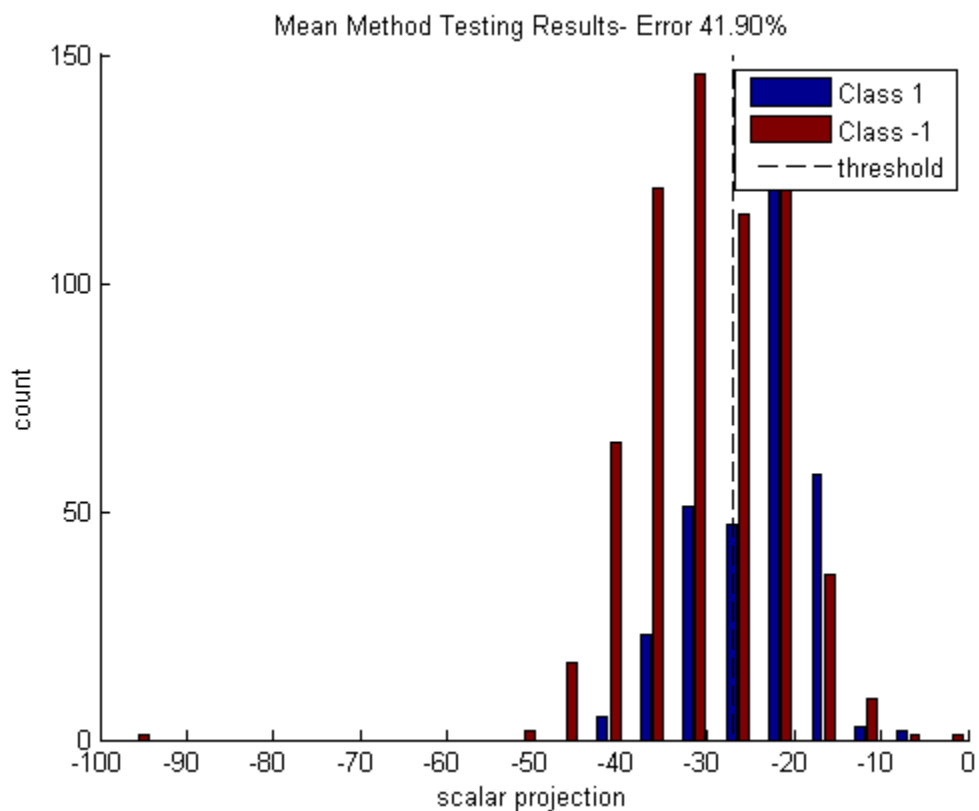
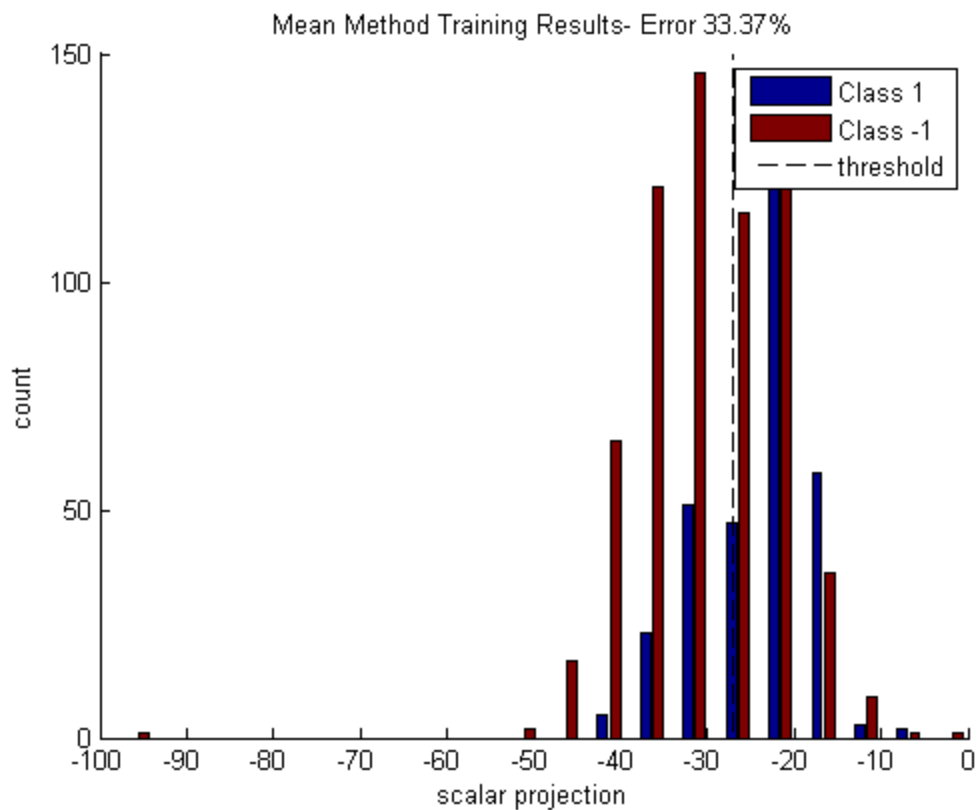
```
HistClass(Classp_train,Classm_train,w,t,...  
  'Mean Method Testing Results',MeanTestError);
```

```
MeanTrainError =
```

```
0.3337
```

```
MeanTestError =
```

```
0.4190
```



Fisher method on DatasetA

```
meanp=mean(Classp_train);
meanm=mean(Classm_train);

psize=size(Classp_train,1)
nsize=size(Classm_train,1)
Bp=Classp_train-ones(psize,1)*meanp;
Bn=Classm_train-ones(nsize,1)*meanm;

Sw=Bp'*Bp+Bn'*Bn;
wfisher = Sw\((meanp-meanm)');
wfisher=wfisher/norm(wfisher)

tfisher=(meanp+meanm)./2*wfisher
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Analyze training data  results of the Fisher Linear Discriminant

FisherPosErrorTrain = sum(Classp_train*wfisher <= tfisher);
FisherNegErrorTrain = sum(Classm_train*wfisher >= tfisher);

FisherTrainError= ((FisherPosErrorTrain + FisherNegErrorTrain)/(size(Train,1)))

% Histogram of Fisher Training Results
HistClass(Classp_train,Classm_train,wfisher,tfisher,...
    'Fisher Method Training Results',FisherTrainError);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

FisherPosErrorTest = sum(Classp_test*wfisher <= tfisher);
FisherNegErrorTest = sum(Classm_test*wfisher >= tfisher);

FisherTestError= ((FisherPosErrorTest + FisherNegErrorTest)/(size(Test,1)))

% Histogram of Fisher Testing Results
HistClass(Classp_test,Classm_test,wfisher,tfisher,...
    'Fisher Method Testing Results',FisherTestError);

psize =

    315

nsize =

    635

wfisher =

    -0.1355
```

-0.0189
0.0073
0.0372
-0.0022
-0.0098
-0.0106
0.0022
0.0045
0.0205
0.0054
0.0085
-0.0744
0.0266
0.1076
-0.0055
-0.6455
-0.7027
-0.0909
-0.0331
-0.0002
-0.0729
0.0007
0.1079
-0.0414
0.0225
-0.0551
-0.0083
-0.0614
0.0009
0.0013
-0.0125
0.0028
-0.0086
0.0006
0.0518
0.0790
-0.0150
-0.0589
0.0477
0.0136

tfisher =

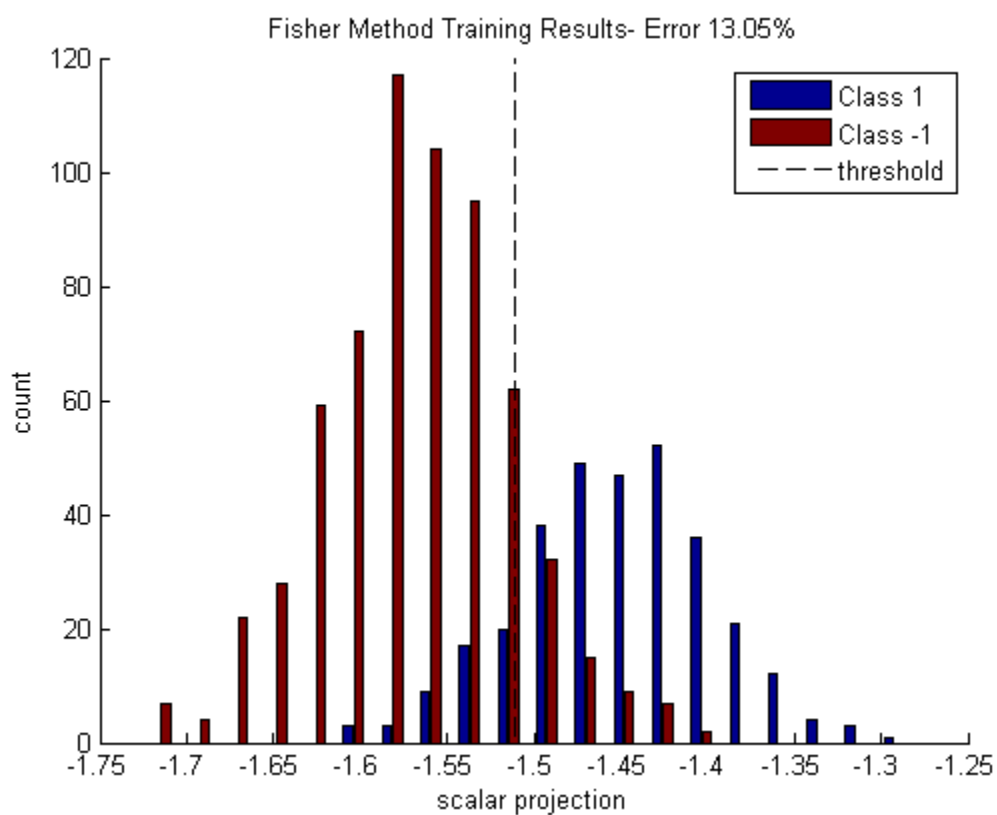
-1.5104

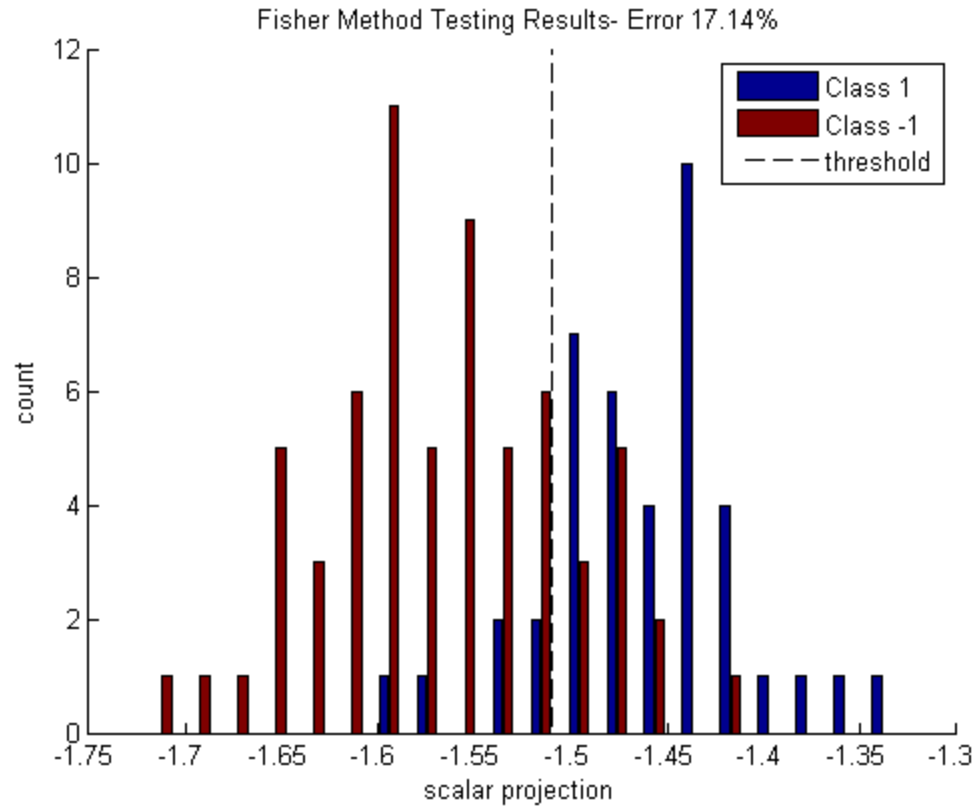
FisherTrainError =

0.1305

FisherTestError =

0.1714





KNN classifier on Test set

```
classifier=knnsearch(Train,Test);  
total_error=0;  
[s,z]=size(Test)  
for i=1:s,  
    if(YTest(i)~=YTrain(classifier(i)))  
        total_error=total_error+1;  
    end  
end  
KNNtesterror = total_error/s
```

s =

105

z =

41

KNNtesterror =

0.2571

FisherMedian DatasetA

```
medianp=median(Classp_train);
medianm=median(Classm_train);

BMp=Classp_train-ones(psize,1)*medianp;
BMn=Classm_train-ones(nsize,1)*medianm;

Sw=BMp'*BMp+BMn'*BMn;
wFishMed = Sw\(medianp-medianm)';
wFishMed=wFishMed/norm(wFishMed)

tFishMed=(medianp+medianm)./2*wFishMed

MedFishPosErrorTrain = sum(Classp_train*wFishMed <= tFishMed);
MedFishNegErrorTrain = sum(Classm_train*wFishMed >= tFishMed);
MedFishTrainError=((MedFishPosErrorTrain + MedFishNegErrorTrain)...
/(size(Train,1)))

MedFishPosErrorTest = sum(Classp_test*wFishMed <= tFishMed);
MedFishNegErrorTest = sum(Classm_test*wFishMed >= tFishMed);
MedFishTestError= ((MedFishPosErrorTest + MedFishNegErrorTest)...
/(size(Test,1)))

HistClass(Classp_test,Classm_test,wFishMed,tFishMed,...
'MedianFisher Method Testing Results',MedFishTestError);

HistClass(Classp_train,Classm_train,wFishMed,tFishMed,...
'MedianFisher Training Results',MedFishTrainError);

wFishMed =

-0.2013
-0.0320
0.0185
0.0443
-0.0029
0.0060
-0.0301
0.0040
0.0058
0.0307
0.0071
0.0213
-0.0684
0.0278
0.1941
-0.0116
-0.8250
```

-0.3367
-0.0907
-0.0416
0.0080
-0.0898
0.0011
0.1394
-0.0428
0.0509
-0.2176
-0.0117
-0.0843
0.0039
-0.0157
-0.0278
0.0221
-0.0140
0.0072
0.0706
0.0573
-0.0194
-0.0809
0.0533
0.0132

tFishMed =

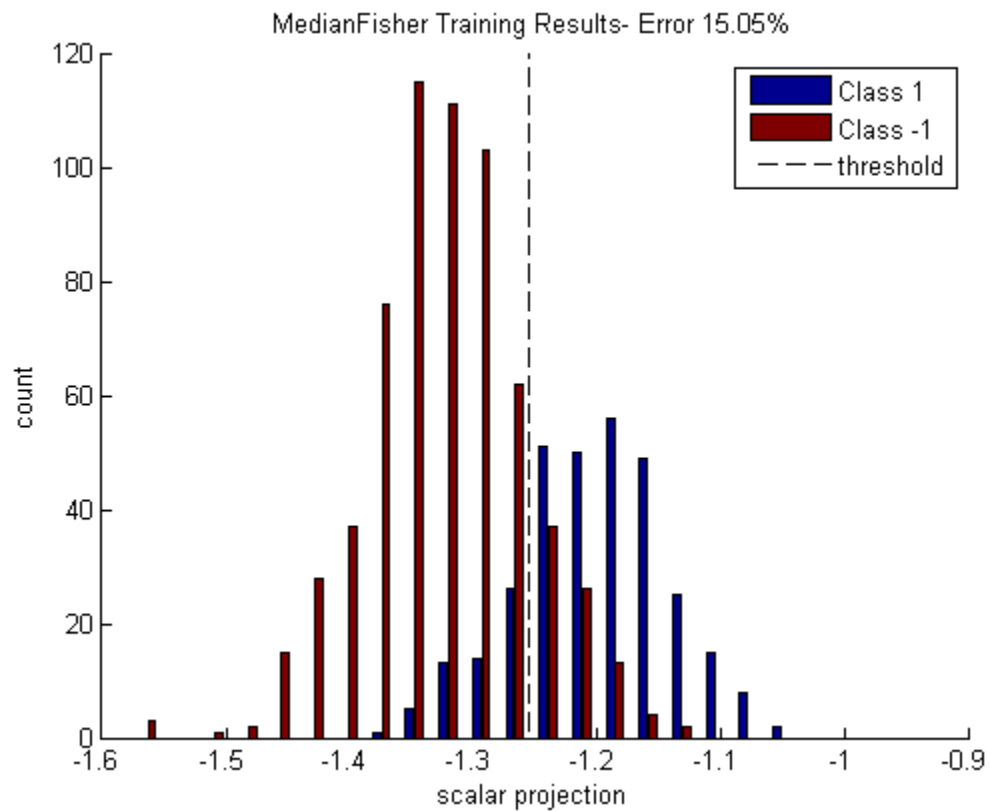
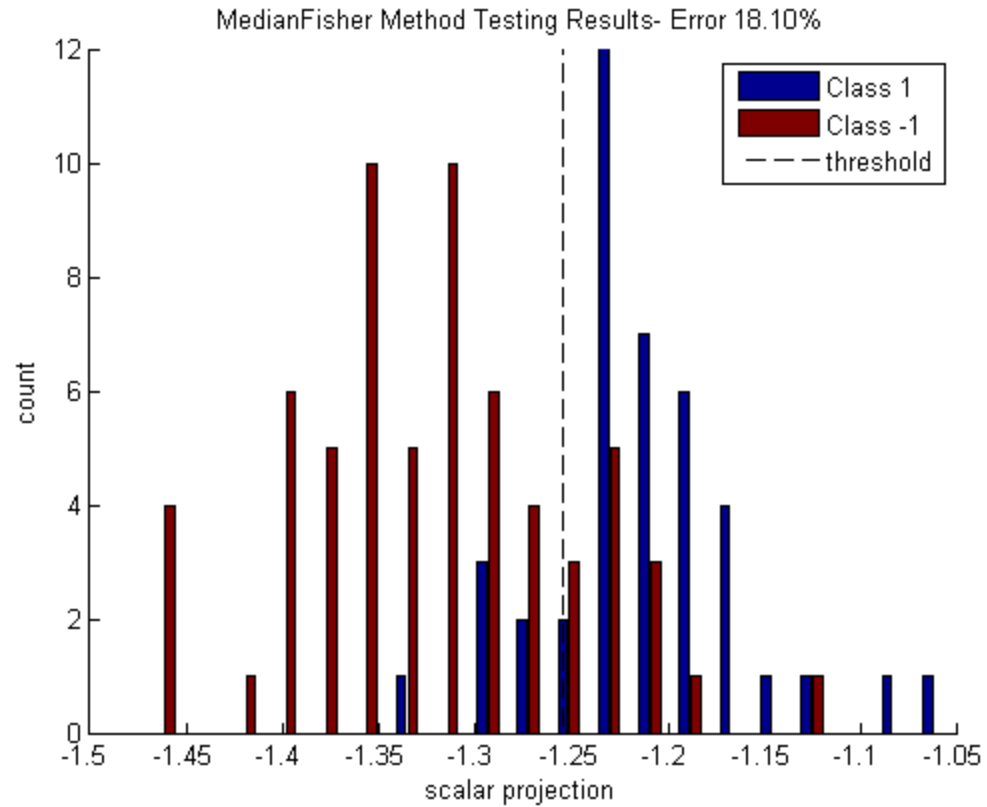
-1.2551

MedFishTrainError =

0.1505

MedFishTestError =

0.1810



DatasetV Analysis

```
DV = csvread('DatasetV.csv');

IDV=DV(:,1); %id column
DV=DV(:,2:end);

[m,n] = size(DV);

DVmean = (1/m)*ones(1,m)*DV %mean for DatasetV

% image for mean
figure
imagesc(DVmean)
title('Mean Vector DatasetV')
colormap(gray)
colorbar

%Prediction by Fisher LDA method
PClassCount = sum(DV*wfisher > tfisher)
NClassCount = sum(DV*wfisher < tfisher)

classes=ones(m,1);
for i=1:m,
    if(DV(i,:)*wfisher <= tfisher)
        classes(i,1)=-1;
    end
end

DVLabels = cat(2,IDV,classes);
csvwrite('MAT_Consulting_DSV_prediction.csv',DVLabels);

%Covariance of DV
DV_centered = (DV - (1/m)*(ones(m,m)*DV));
CovDV = (1/(m-1))*DV_centered'*DV_centered;

%image of covariance of DatasetV
figure
imagesc(CovDV)
title('Covariance Matrix of DatasetV')
colormap(gray)
colorbar

DV_labeled = csvread('MAT_Consulting_DSV_prediction.csv');

DVPos=sum(DV_labeled(:,2)==1) %Number of points in DV estimated to be class 1
DVNeg=sum(DV_labeled(:,2)==-1) %Number of points in DB estimated to be class -1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```


DVmean =

Columns 1 through 7

4.9229	2.8658	0.8775	0.1775	3.1175	0.4000	2.9325
--------	--------	--------	--------	--------	--------	--------

Columns 8 through 14

38.7750	1.4825	2.1925	3.7525	-0.2778	3.8215	1.4113
---------	--------	--------	--------	---------	--------	--------

Columns 15 through 21

10.3688	4.6825	1.0140	1.1301	0.0050	0.1400	0.0350
---------	--------	--------	--------	--------	--------	--------

Columns 22 through 28

1.2498	2.4525	0.0650	0.2000	0.0600	2.2886	0.0065
--------	--------	--------	--------	--------	--------	--------

Columns 29 through 35

0.0625	12.5599	3.6270	0.1350	1.7450	3.2825	1.1225
--------	---------	--------	--------	--------	--------	--------

Columns 36 through 41

4.1658	2.5212	1.4200	9.1148	0.0250	1.0650
--------	--------	--------	--------	--------	--------

PClassCount =

120

NClassCount =

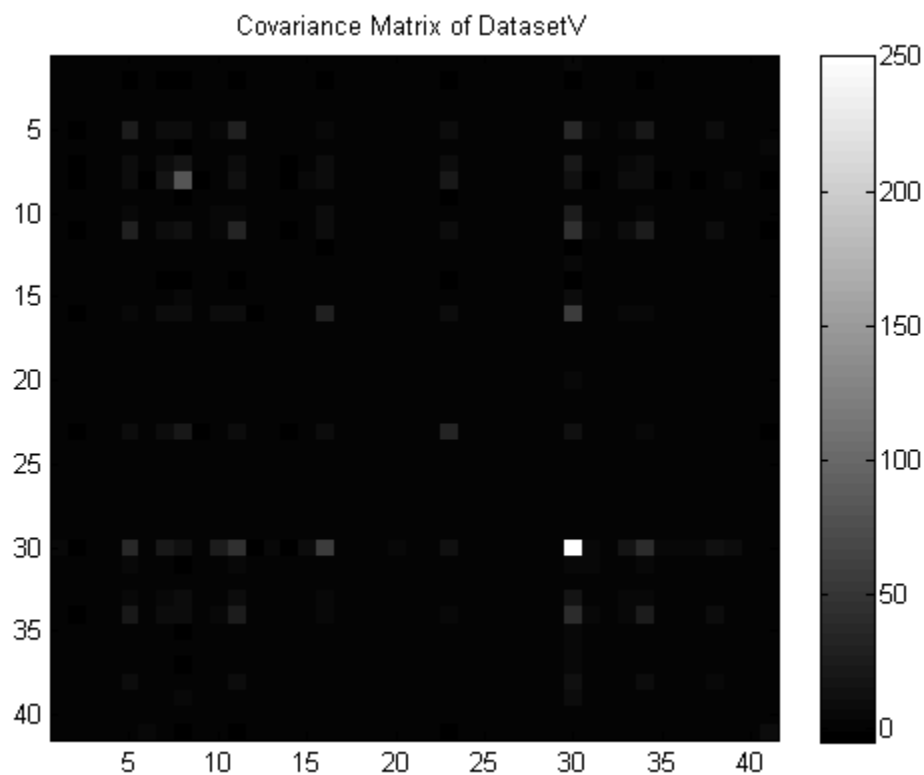
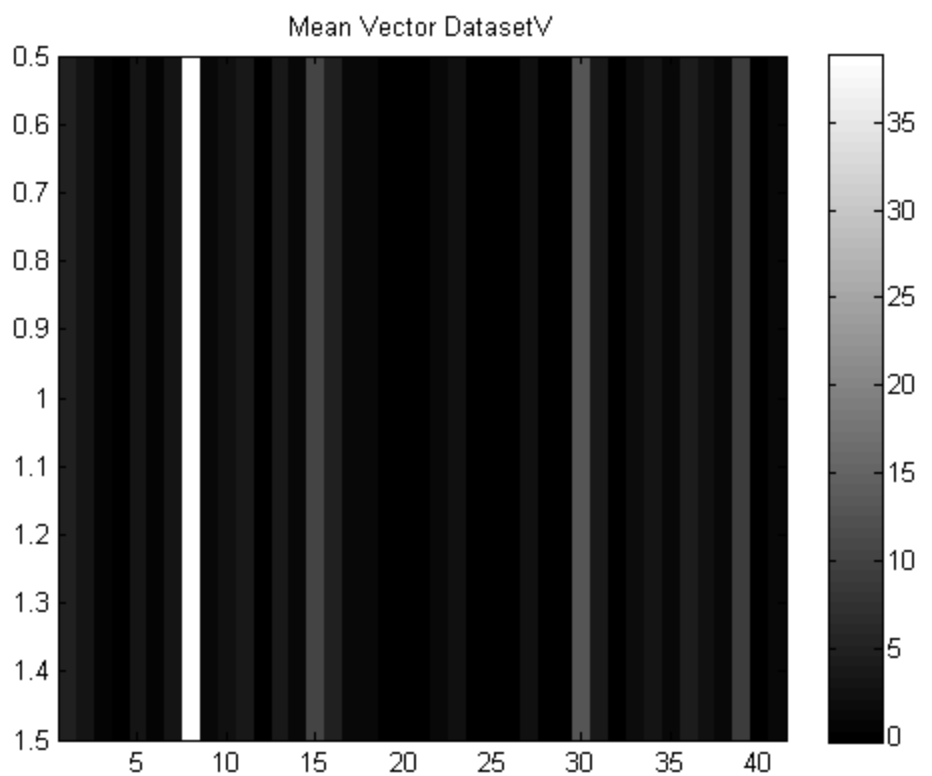
280

DVPos =

120

DVNeg =

280



DA ErrorsFisher

```
DApmis=sum(DAp*wfisher <= tfisher); %Number of erroneous class 1 points
DAnmis=sum(DAn*wfisher >= tfisher); %Number of erroneous class -1 points
DAPosError = DApmis/size(DAp,1)
DANegError = DAnmis/size(DAn,1)

TotalDAError = (DApmis+DAnmis)/size(DA,1)%Total error of Fisher on DatasetA

DAPosError =

    0.1433

DANegError =

    0.1302

TotalDAError =

    0.1346
```

Published with MATLAB® R2014a