CS303B: Lab – Dimensionality Reduction

Aim: To understand how to perform Dimensionality Reduction using PCA and LDA in Matlab.

Taking the iris dataset as the example:

1: **First visualize the data simply using the first two measures of the samples**:

```
load fisheriris
gscatter(meas(:,1), meas(:,2), species,'rgb','osd');
xlabel('Sepal length');
ylabel('Sepal width');
```

2: **Visualizing the data using PCA**:

```
load('fisheriris');
m= mean(meas);
meas = meas - repmat(m, size(meas,1), 1);
covar = cov(meas);
[v, d] = eigs(covar);
```

Questions: What is the meaning of the above code? Do you understand each line of the codes?

Variable *v* contains that eigen vectors (each column in *v* is an eigen vector). The corresponding eigen values are stored in *d*. select the first two *leading* eigen vectors.

Now computing projection scores on the first two principle vectors:

```
score = meas*v(:,1:2); %% get the projection scores on the first two leading vectors.

figure(2)
gscatter(score(:,1), score(:,2), species, 'rgb','osd')
title('Projected Iris data'), grid on

%Or you can try the following code to plot:
figure(3)
hold on;
plot(score (1:50, 1), score(1:50, 2), 'o');
plot(score (51:100, 1), score(51:100, 2), '*');
plot(score (101:150, 1), score(101:150, 2), '+');
```

```
hold off;
legend('Setosa', 'Versicolour','Virginica');
```

Questions: Compare figure1 and figure 2, what could you conclude?

3. If we don't use the leading two Eigen vectors to project the original data, what will happen? E.g, using the two eigenvectors corresponding to the second and third largest eigenvalues. Plot the figure to verify your thoughts

4. **LDA (we only consider two classes)**

```
% preparing the vectors for 'virginica' against the rest, i.e., 'virginica' as class 1 %
and others as class '0'

load fisheriris
a = zeros(size(species));
for i=1:max(size(a))
a(i) = isequal(species{i},'virginica');
%% 1 means 'virginica'; and '0' means others
end
label_vector = a;
instance_matrix = meas;
class1 = meas(a==1,:)
class0 = meas(a==0,:);

% class means
m1 = mean(class1);
m0= mean(class0);

% class covariance matrix
s1 = cov(class1);
s0 = cov(class0);

% within class scatter matrix
sw = s0 + s1;

% computing the LDA projection vector

v = inv(sw)*(m1-m0)';

% computing the projection score:
score  = meas*v;
```

gscatter(score(:,1), score(:,1), a, 'rg','os') %% look at only one direction will be fine.

Or the following code to plot data after LDA projection:

```
% the class 1:
x1 = score(a==1);

lm1 = mean(x1);
lstd1 = std(x1);
class1_pdf = mvnpdf(x1,lm1,lstd1);
% the  class 0
x0 = score(a==0);
lm0 = mean(x0);
lstd0 = std(x0);
class0_pdf = mvnpdf(x0, lm0, lstd0);

figure(1); hold on; plot(x1, class1_pdf, 'r.'); plot(x0,class0_pdf,'g.'); hold off
```

Questions and Extensions: Would you be able to compare the PCA vs. LDA using iris dataset? i.e., plot the projected score of the first principal component and then compare to LDA?