EEC 289Q Data Analytics for Computer Engineers Homework 5

Ahmed Mahmoud

June, 7th 2018

PCA Whitening:

The following listing shows our code for PCA, PCA whitening and ZCA whitening implementation applied to MNIST data set.

```matlab
1  %% Step 0a: Load data
2  clear;
3  close all;
4  addpath(genpath('../common'))
5  x = loadMNISTImages('../common/train-images-idx3-ubyte');
6  figure('name','Raw images');
7  randsel = randi(size(x,2),200,1);
8  display_network(x(:,randsel));
9  %%===============================================================
10 %% Step 0b: Zero-mean the data (by row)
11 %%% YOUR CODE HERE %%%
12 avg = mean(x, 1);
13 x = x - repmat(avg, size(x, 1), 1);
14 %%===============================================================
15 %% Step 1a: Implement PCA to obtain xRot
16 %%% YOUR CODE HERE %%%
17  sigma = x * x' / size(x, 2);
18  [U,S,V] = svd(sigma);
19  xRot = U'*x;
20 %%===============================================================
21 %% Step 1b: Check your implementation of PCA
22 %%% YOUR CODE HERE %%%
23 covar = xRot*xRot'/(size(xRot,2)-1);
24
25 % Visualise the covariance matrix. You should see a line across the
26 % diagonal against a blue background.
27 figure('name','Visualisation of covariance matrix');
28 imagesc(covar);
29 %%===============================================================
30 %% Step 2: Find k, the number of components to retain
31 %%% YOUR CODE HERE %%%
32 S_diag = diag(S);
33 k = sum(cumsum(S_diag)/sum(S_diag)<=0.99);
34 %%===============================================================
```

```matlab
35  %% Step 3: Implement PCA with dimension reduction
36  %%% YOUR CODE HERE %%%
37  xHat = U*[V(:,1:k)'*x;zeros(size(x,1)-k,size(x,2))];
38
39  % Visualise the data, and compare it to the raw data
40  % You should observe that the raw and processed data are of comparable
41  % For comparison, you may wish to generate a PCA reduced image which
42  % retains only 90% of the variance.
43
44  figure('name',['PCA processed images ',...
45      sprintf('(%d / %d dimensions)', k, size(x, 1)),'']);
46  display_network(xHat(:,randsel));
47  figure('name','Raw images');
48  display_network(x(:,randsel));
49  %%================================================================
50  %% Step 4a: Implement PCA with whitening and regularisation
51  epsilon = 1e-1;
52  %%% YOUR CODE HERE %%%
53  xPCAwhite = diag(1./sqrt(diag(S) + epsilon)) * xRot;
54
55  %% Step 4b: Check your implementation of PCA whitening
56  %%% YOUR CODE HERE %%%
57  covar=xPCAwhite*xPCAwhite'/size(xPCAwhite,2);
58
59  % Visualise the covariance matrix. You should see a red line across the
60  % diagonal against a blue background.
61  figure('name','Visualisation of covariance matrix');
62  imagesc(covar);
63  %%================================================================
64  %% Step 5: Implement ZCA whitening
65  %%% YOUR CODE HERE %%%
66  xZCAWhite = U * xPCAwhite;
67
68  % Visualise the data, and compare it to the raw data.
69  % You should observe that the whitened images have enhanced edges.
70  figure('name','ZCA whitened images');
71  display_network(xZCAWhite(:,randsel));
72  figure('name','Raw images');
73  display_network(x(:,randsel));
```

Figure 1 shows the series of images produces by the code for an example image from the input data set.

(a) Raw input

(b) $xRot$ covariance Matrix

(c) PCA whitened (99% variance)

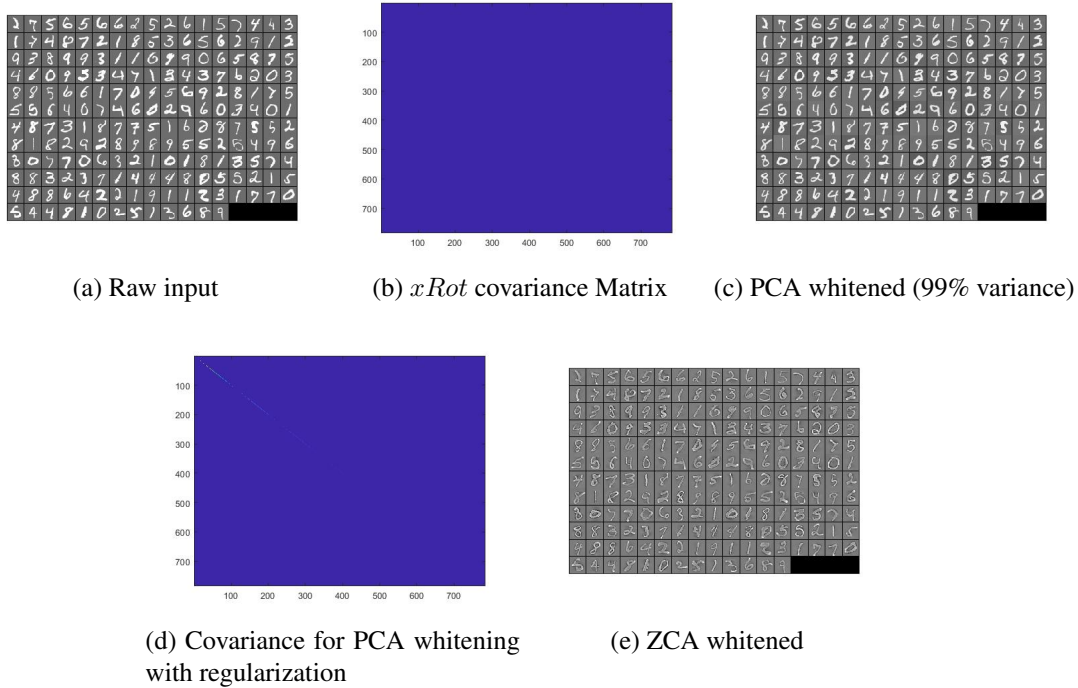(d) Covariance for PCA whitening with regularization

(e) ZCA whitened

Figure 1: Showing the progress of the code on a single image from the input dataset.