Soft Computing Assignment No. 4

Manas Gaur, Computer Science, U00856829

Answer 1a. How I designed my dataset?

My process of generating the dataset uses random data generator. I have used normal continuous random variable to generate 800 values of 6 dimensional random datasets. My data has been synthesized using Cholesky decomposition matrix. The covariance matrix created from synthesized dataset matrix has positive determinant.

Answer 1b. What Principal components should be produced via a correct eigenvector extraction from the data?

The principal components that should be produced via eigenvector extraction from the data are 4 dimensions of the 6 dimensions of the data which has the highest Eigen value. This states that principal components of 2 dimension contributed least towards classification. Hence, have been ignored.

Answer 1c. How have you designed the dataset so those components would be extracted?

Following are the steps for verification:

1. Dataset designed and I developed covariance matrix using it.
2. Calculated the lower triangular matrix
3. The Data generated was random and in small range. I have kept small range of random number generator so as to facilitate convergence and prevent oscillation in learning.
4. I trained my dataset through Generalized Hebbian learning so that it picked 4 important principal components. The dimensions selected have highest Eigen values.

Having said that, I verified my results as follows:

1. Eigen vector of my covariance matrix were calculated.
2. Calculate Eigen pairs (values and vector) of synthesized dataset.
3. There was infinitesimal small difference in the value.
4. Hence, I obtained 4 dimensions from my dataset which possess highest Eigen values.

Execution Details and Results

>> gcc simple_eigenfilter.c
>> ./a.out

-0.677941 0.735206
-0.735118 -0.677852

```
The six dimensions eigen vectors matrix is:

[[-0.48395757 -0.12466338 -0.33718094  0.6873909  -0.40496468 -0.00710462]
 [-0.47097739 -0.11480396  0.57168897 -0.38692264 -0.5353371   0.04332572]
 [-0.43671235 -0.12113094  0.46707604  0.29888714  0.68035554 -0.1556676 ]
 [-0.43610086 -0.42641424 -0.53932963 -0.49026578  0.2664932   0.16040988]
 [-0.30733988  0.60268595 -0.2233455  -0.21929367  0.00718467 -0.66654879]
 [-0.26192733  0.64151923 -0.02368633  0.00091284  0.12435389  0.70980376]]
The six dimensions synthesised dataset eigen values matrix is :

[ 3.68220604+0.j  1.35798415+0.j  0.66821991+0.j  0.04979281+0.j
  0.11920662+0.j  0.40811955+0.j]
The six dimensions synthesised dataset eigen vector matrix is :

[[-0.47487584 -0.20025751 -0.37660257  0.31680043  0.69988472  0.04857193]
 [-0.46655765 -0.20871321 -0.5058687  -0.56398357 -0.39854897  0.07707699]
 [-0.43058271 -0.04330975  0.67792737 -0.46990958  0.28833658 -0.22177528]
 [-0.42520752 -0.39220343  0.33478148  0.55343207 -0.48002902  0.12876911]
 [-0.31813134  0.55733771 -0.14825695  0.23257945 -0.19362494 -0.6889145 ]
 [-0.2990091   0.6708287   0.09286055  0.01909852 -0.01623667  0.6718117 ]]
The four most important dimensions eigen vectors are:

0.471986932203 0.460477975393 0.433113417771 0.417736914249 0.329357650186 0.307821477252
0.208313889911 0.214179512578 0.045495050785 0.391389324069 -0.555500867798 -0.668260061665
0.372864701474 0.506633743684 -0.677095902297 -0.336734517763 0.15273406587 -0.0922319497588
-0.0494185609021 -0.0771692343351 0.22118445752 -0.130766751269 0.68846465336 -0.671572925843
```

Answer 2a: I have read and understood the *som_pak.pdf.* Because of my interest in python programming, I have used the concept and methodology defined in *som_pak.pdf* for creating my self organizing map.

Answer 2b: I downloaded *gnuplot* in mac OSX and tested it by executing shell script files mentioned in the folder. The source used for downloading the gnuplot package is http://www.gnuplot.info and https://www.youtube.com/watch?v=baUQ1g_UySM

Answer 2c: After executing CREATE_MESH_FROM_FOO.sh from the folder, I obtained lattice type map visualization. I understood the the way content was stored in ".cod" files and ".dat" files. The lattice map was meaningful and sensible.

Answer 2d: The two shell script files present in "*som_pak-3.1_WSU_mods"* were easy to understand and was used in designing my self organizing map python code. The shell script file first defines the dimension of the map and the type of neighborhood (Gaussian because of convergence). The "vsom" function is responsible for self organizing map creation using learning rate, "cod" file generated suing "randinit" function and distance specified using radius parameter. The self organizing map created is sent to "*gnuplot*" for plotting.

Answer 2e: The file "*Gaur_animal_map.cod"* will be attached in the assignment folder. This file is different from the ". cod" files present in the *som_pak-3.1_WSU_mods* folder in that *"Gaur_animal_map.cod"* contains weight vectors of the neurons without the labels.

Answer 2f: The self organization map formed has following characteristics:

1. Learning Rate: 0.02

2. Iterations: 10000
3. Dimensions: x dimension → 10 and y dimension → 10
4. Labels: 16 labels
5. Neighborhood Function: Gaussian

Table 1: Labels and Shapes

Label	**Color**	**Shape**
dove	Blue	
hen	Green	
duck	Red	
goose	Cyan	
owl	Pink	
hawk	Yellow	
eagle	White	
fox	Blue	
dog	Green	
wolf	Red	
cat	Cyan	
tiger	Pink	
lion	Yellow	
horse	White	
zebra	Blue	
cow	Green	

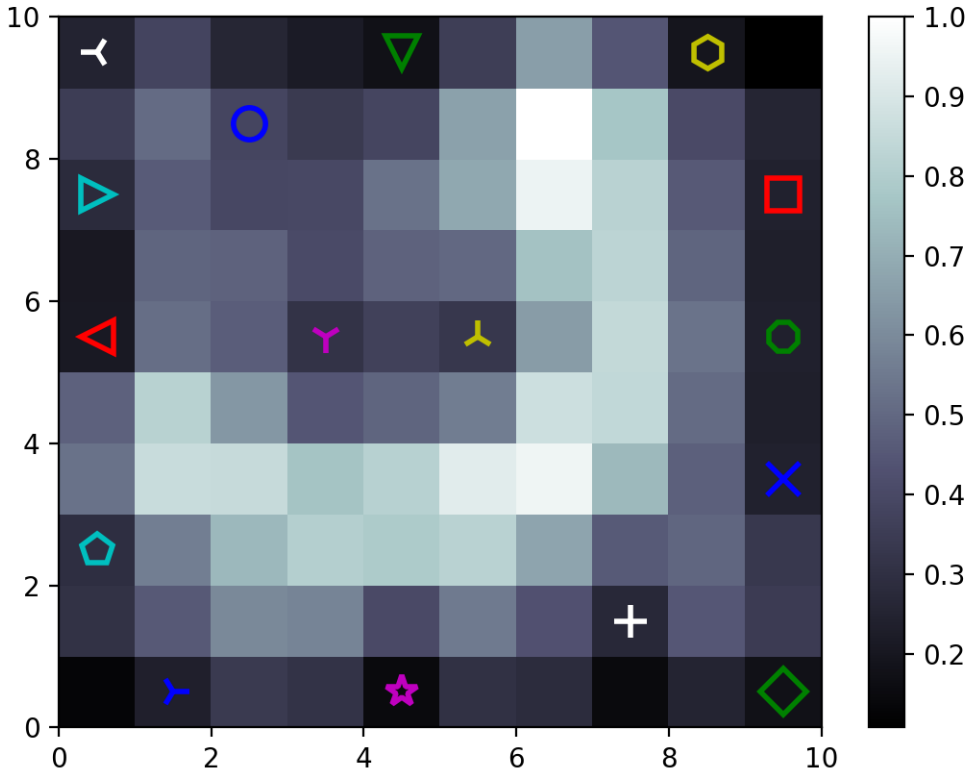

Figure 1: Self Organizing Map of Animals. See table 1 for labels of these symbols.