Contents


```
clear; close all; clc
```

Performing clustering with only a "Competitive Layer" (based on LVQ)

Competitive learning is a form of unsupervised learning in artificial neural networks, in which nodes compete for the right to respond to a subset of the input data. As a variant of Hebbian learning, competitive learning works by increasing the specialization of each node in the network. It is well suited to finding clusters within data. (from Wikipedia)

By the word "only", we mean that a one-layer neural net is the only thing that represents our model. For this pupose, the number of clusters should be given to the "competlayer" function as the first input argument.

Part 1- Creating a fake dataset

```
nRow_fake_dataset = 3000; % it should actually be 30000, but for elapsed
% running time of the code, 3000 is chosen.
nCol_fake_dataset = 600; % it should actually be 6000, but for elapsed
% running time of the code, 600 is chosen.

temp0 = randn(ceil(nRow_fake_dataset/5), ceil(nCol_fake_dataset/5));
temp1 = repmat(temp0, 5, 5);
fake_dataset = temp1(1:nRow_fake_dataset, 1:nCol_fake_dataset);
```

Part 2- Performing linear dimensionality reduction (or noise suppression) alg.

```
[coeff, score, ~, ~, explained, ~] = pca(fake_dataset);
explained_variance_threshold = 99.5; % for the FFT dataset, a threshold of
% 99 or 99.5 is thought to be more suitable.

cumulative_explained = cumsum(explained);
numFeatures = find(cumulative_explained >= explained_variance_threshold);
numFeatures = numFeatures(1);

reduced_dataset_by_pca = score(:, 1:numFeatures);
size(reduced_dataset_by_pca)
```

```
Warning: Columns of X are linearly dependent to within machine precision.
Using only the first 120 components to compute TSQUARED.

ans =
```

```
3000         119
```

Part 3- Performing LVQ-based clustering

Part 3-1- Preparing (transposing) the dataset matrix

```
inputs = reduced_dataset_by_pca.';
```

Part 3-2- Creating a "network" object and configuring its parameters

```
numClusters = 3;  numEpochs = 100;
net = competlayer(numClusters);
configure(net, inputs);
net.trainParam.epochs = numEpochs;
```

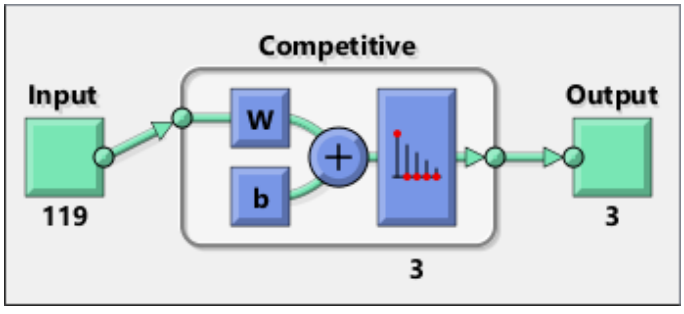
Part 3-3- Training the network and clustering the dataset

```
[net, tr] = train(net, inputs,'CheckpointFile', ...
    'compete_layer_chk', 'CheckpointDelay', 5);

view(net)

oneHotClasses = net(inputs);
classes = vec2ind(oneHotClasses);
```

```
31-Oct-2019 16:59:04 First Checkpoint #1: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:09 Write Checkpoint #2: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:15 Write Checkpoint #3: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:21 Write Checkpoint #4: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:26 Write Checkpoint #5: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:32 Write Checkpoint #6: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:37 Write Checkpoint #7: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:43 Write Checkpoint #8: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:48 Write Checkpoint #9: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:53 Write Checkpoint #10: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 16:59:59 Write Checkpoint #11: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 17:00:04 Write Checkpoint #12: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 17:00:09 Write Checkpoint #13: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 17:00:15 Write Checkpoint #14: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 17:00:20 Write Checkpoint #15: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
31-Oct-2019 17:00:22 Final Checkpoint #16: C:\Users\M_Ali\Desktop\sadra-prj\phase-2\code\compete_layer_chk.mat
```



Part 3-4- Visualization of the cluster proposals for some samples

```
numOfSamples = 5;
disp(['Proposals for the first ' num2str(numOfSamples) ' samples:'])
classes(1:numOfSamples)
```

```
Proposals for the first 5 samples:

ans =

     1     1     3     3     1
```

Published with MATLAB® R2018b