Using the Neural Gas Algorithm

This is the Spring 2021 version using Matlab. This semester, we'll be using code authored by S. Mostapha Kalami Heris, and available for download from `yarpiz.com` (Project YPML111).

I'm using this code because it is cleanly written and nicely self-contained. It is very useful to be able to write your own code, and I would highly encourage you to do it some time.

For now, we'll take a look and see how to use the code to get a Neural Gas clustering.

Note on Packages

The first time you run the code, you may see an error (the error is similar on Octave-Online). You'll need to either install a package (Use Octave's Help menu), or you may need to "load" the package: `pkg load statistics`

In our case, the author is using `unifrnd` to choose data from a uniform distribution between two given numbers (to initialize the weights).

The Data

The data should be organized as "number of points" by dimension. Typically, that will mean the data is a tall matrix, although it doesn't have to be. The data is passed in as the first parameter.

The Parameters

We'll be passing parameters in a new way- by creating a Matlab *data structure*. Our data structure will have the name `params`, and we'll be able to attach all kinds of data to it using a dot. We do have lots of parameters to pass in- Here's a list (and why it's good to use these script files when you're starting your own clustering).

```
params.N = 30;       % Number of clusters
params.MaxIt = 50;   % Number of passes through the data

params.tmax = 10000; % See comment below.  Used for maximum number of iterations
params.epsilon_initial = 0.4; % Epsilon is the learning rate
params.epsilon_final = 0.02;

params.lambda_initial = 2;  % Lambda is the neighborhood size
params.lambda_final = 0.1;  %Good value

params.T_initial = 5;   % These are used to set T, which is used
params.T_final = 10;    %    to determine if a connection should be removed.
```

Looking over the parameters, remember that ϵ, λ, T are updated every iteration, but our author is updating these after each data point has been selected, so the `tmax` ought to be set automatically (as `MaxIt * nData`) rather than by user choice. We'll look into these values later.

The Output

What information do we want to come out of Neural Gas? We want to know the cluster center locations, and the connections between them. These are given in the output data structure `net`. That is, if

```
net = NeuralGasNetwork( data, params, plotflag)
```

then the output is:

- `net.w` is the matrix of weights (number of weights by dimension).
- `net.c` is the connection matrix (number of weights by number of weights).
- `net.t` is the time matrix (although I'm not sure what we would do with that).

Sample Files

We have three examples showing the clustering, the main program file, three example data sets, and the plotting routine.

- `app1.m`: Script file for the two rings data.
- `app2.m`: Script file for the two horseshoes data
- `app3.m`: Script file for the spiral data
- `NGDdata.mat`: Data file for the three examples
- `NeuralGasNetwork.m`: The main function you call.
- `plotResults.m` Helper function to plot the data, centers and connections in a nice way. Once the clustering is finished, you can plot the results: `PlotResults(X,net.w,net.C)` (note lower case "w" and upper case "C").