

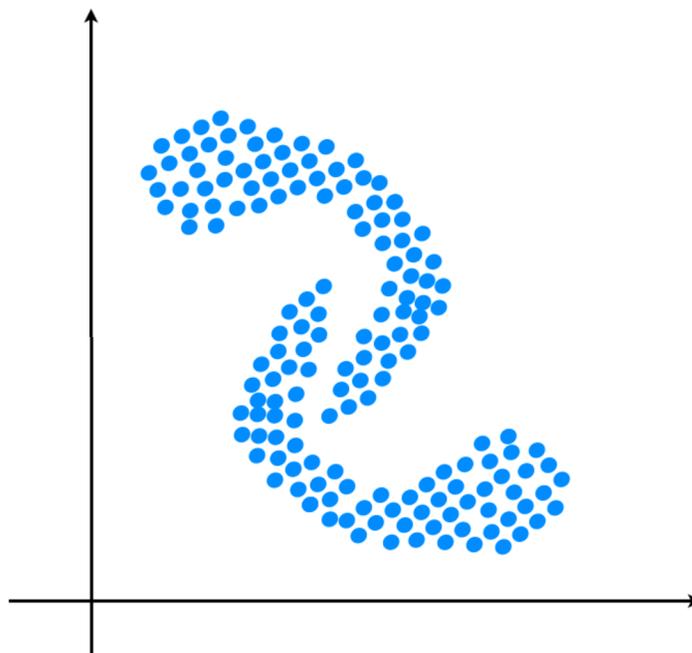
# INF3490/INF4490 Exercise Solutions - Unsupervised Learning

November 1, 2018

ℙ marks the programming exercises, we strongly recommend using the python programming language for these. Exercises may be added/changed after publishing.

## 1 k-means clustering

The following graph shows 2D data points. It is clear from the graphs that there are two clusters. Will  $k$ -means clustering be able to find these two, if we define  $k = 2$ ? If not, why?



*Answer:*

$k$ -means will struggle to find these clusters because they are non-convex shaped. Some points (e.g. in the center of the graph) that lie in one cluster (from the two visibly obvious clusters) can be closer to the cluster center of the other cluster than the points in the other cluster. The closeness between data points, since it is defined as the Euclidian distance, does not inform the  $k$ -means algorithm about the non-convex nature of the layout of these data points.

## 2 Self organizing maps

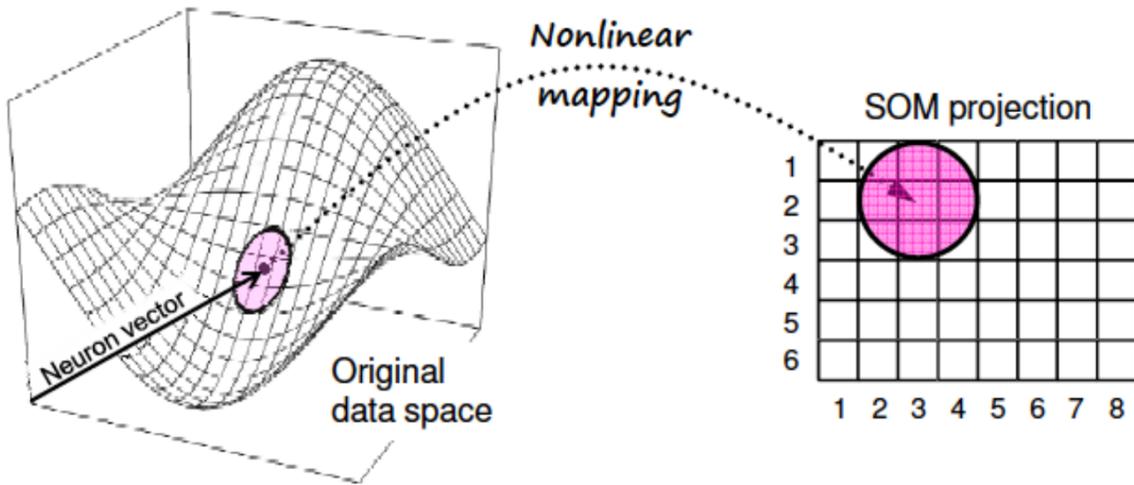
In SOMs, what role do the predefined topological (neighborhood) relationships between neurons in the map space play in the discovery of topological (neighborhood) relationships between input vectors in the data space?

*Answer:*

The map layer defines a topological ordering over the neurons. The SOM algorithm works by making a map space neighbor neurons zone in on to similarities in the data space. This is because the weights associated with

all the map space neighbor neurons are moved towards the data points closest to the best matching neuron. This makes weights of map space neighbor neurons adapt towards similar weights in the data space (weights having the same dimensionality as data, makes weight be elements of and move around in the data space). Thus, similarities in the data are captured by map space neighbor neurons. Visualization techniques are needed to see the dissimilarities in data space. These techniques can allow seeing the separation between groups of neurons that represent different data clusters.

The relationship between map space and data space can be seen in the figure below. The data points lying nearby the highlighted neuron (and its neighborhood) in the data space (left hand side graph), will be represented by spatially close neurons in the map space (right hand side graph).



Ideally, the winning and second place neurons will be close neighbors in the map space, because their weight vectors are similar. Weight vectors that are similar are similar because they get adapted towards similar data points, i.e. points within some cluster.

### 3 Gaussian function in SOMs

In SOMs, a Gaussian function, as defined below, can be used to define the neighborhood relation

$$N(i, j) = e^{-\frac{\|i-j\|^2}{2\sigma^2(t)}} \tag{1}$$

where  $i$  and  $j$  are two neurons, whose position on the lattice are given by (row, column), and  $\|i-j\|$  is the Euclidian distance between them. So, if  $i$ 's position is (2, 2) and  $j$ 's position is (3, 3) and  $\sigma(t) = 1$ ,  $N(i, j)$  will be  $e^{-1}$ .

As can also be seen, the further a neuron  $i$  is from the winning neuron  $j$  on the lattice the smaller  $N(i, j)$  is. What would happen if  $N(i, j)$  is set to and remains zero for all neurons except the winning neuron? What would happen if  $N(i, j)$  is set to and remains 1 for all neurons including the winning neuron? Why is it important to have  $N(i, j)$  large for distant (on the lattice/map space) neurons in the beginning of the learning process i.e.  $j$  should have a larger neighborhood, and smaller as time goes by? This can be controlled by  $\sigma(t) = \sigma_0 e^{-t/T}$ , where  $\sigma_0$  is the initial value of  $\sigma$ , and  $T$  is a constant.

Example Matlab script that can be run to see how the neighborhood function may change with time:

```

figure;
[X,Y] = meshgrid(-5:.1:5, -5:.1:5);
sigma0 = 5;
for i=1:100
    Z = (exp(-(X.^2 + Y.^2)/(2*sigma^2)));
    surf(X,Y,Z);
    sigma = sigma0 * exp(-i/10);
    surf(X, Y, Z);
    pause(0.01);
end

```

Describe one other neighborhood relationship function that can be used for SOMs. In what way can it influence the learning process? Explain.

*Answer:*

There are quite a few neighborhood relationship functions that have been investigated in SOM research. In general, the neighborhood should be larger (somewhere half the size of the larger dimension in map space i.e. height or width of the map, whichever is greater) in size in the beginning of learning and gradually become smaller. As long as such a reduction in time of the neighborhood is taken into consideration, different popular neighborhood functions lead to more or less similar learning results.

Typical functions other than Gaussian are:

- **Bubble/flat function:** value is 1 up to a certain distance in the map neighborhood, otherwise 0.

$$N(i, j) = \begin{cases} 1 & \text{if } \|i - j\| \leq \sigma \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

- **Cut Gaussian function:** the Gaussian function is multiplied by the flat function so that it is smooth for close neurons and then cuts straight to zero at a certain distance:

$$N(i, j) = \begin{cases} e^{-\frac{\|i-j\|^2}{2\sigma^2(i)}} & \text{if } \|i - j\| \leq \sigma_{flat} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

- **Epanechicov function:** piecewise polynomial function is parabolic for near neighbors and zero above a certain distance:

$$N(i, j) = \max \left\{ 0, 1 - \left( \frac{\|i - j\|}{\sigma} \right)^2 \right\} \quad (4)$$

## Contact

The exercises are the same as those in the **GitHub repository**. If there are any suggestions to corrections of grammar, language or any additional suggestions, we appreciate all feedback! The TA's can be reached at [lonnekes@ifu.uio.no](mailto:lonnekes@ifu.uio.no), [hermankn@ifu.uio.no](mailto:hermankn@ifu.uio.no), [a.s.skage@econ.uio.no](mailto:a.s.skage@econ.uio.no) or [sharanak@ifu.uio.no](mailto:sharanak@ifu.uio.no)