Exercise 2: The functions and parameters of the algorithm

In this exercise you will relate the observations made at the end of Exercise 1 to some function and parameter that characterizes the SOM algorithm. In Exercise 1 You observed qualitatively how the training point “attracts” the winning neuron and the other neurons of the network, and how the resulting displacement changes during the training.

The reason of this behavior can be understood observing the terms of the weight update formula and the rationale behind the choice of the winning neuron.

Weight update formula

$$\text{"}w_p(n+1)=w_p(n)+\eta(n)h_{p,p_{winner}}(x(n)-w_p(n))\text{"}$$

Winner selection criterion

$$p_{winner}=p:\min_{p}(x-w_p)^2$$

Neighborhood function

$$\left\{h_{p,q}(n)=e^{-\frac{e^{\frac{2n}{\tau_1}}d[ip,jp,kp,iq,jq,kq]^2}{2\sigma_0^2}},\ \sigma_0=2.82843\right\}$$

$$d[ip,jp,kp,iq,jq,kq]=\sqrt{(ip-iq)^2+(jp-jq)^2+(kp-kq)^2}$$

Learning Rate

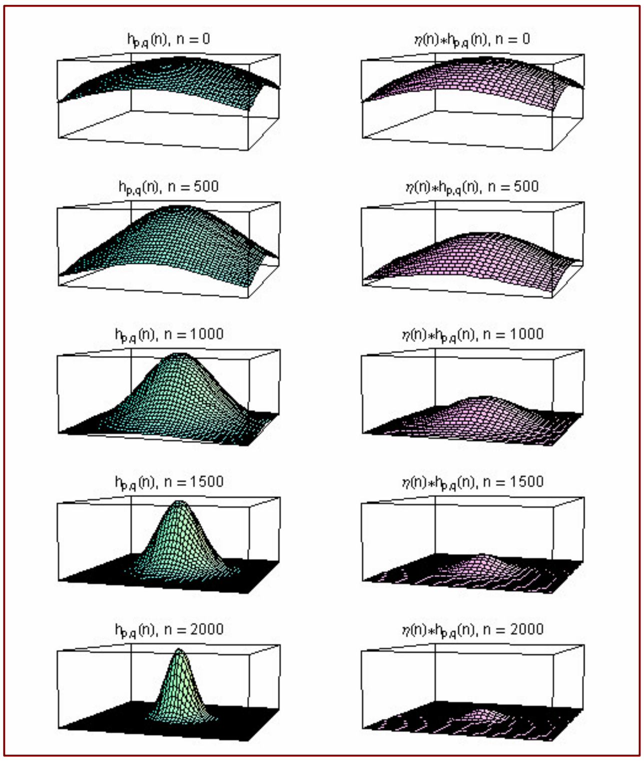
$$\eta(n)=e^{-\frac{n}{\tau_2}}\eta_0$$

Changing the parameters of these function (or the function themselves) you can influence the policy of updating of the weights. In particular, there are three parameters with which you are asked to experiment. They are defined in the following two pink cells

```
setNeighborhoodFunction[tau1 → 1000];
```

```
setLearningRateFunction[eta0 → 0.1, tau2 → 1000];
```

Look at the weight update formula and try to understand the meaning and effect of these parameters



$h_{p,q}(n)$, n = 0

$\eta(n)*h_{p,q}(n)$, n = 0

$h_{p,q}(n)$, n = 500

$\eta(n)*h_{p,q}(n)$, n = 500

$h_{p,q}(n)$, n = 1000

$\eta(n)*h_{p,q}(n)$, n = 1000

$h_{p,q}(n)$, n = 1500

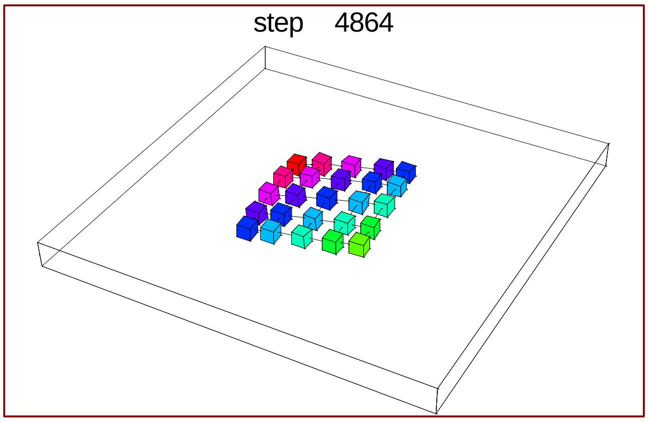
$\eta(n)*h_{p,q}(n)$, n = 1500

$h_{p,q}(n)$, n = 2000

$\eta(n)*h_{p,q}(n)$, n = 2000

You can obtain some hint looking at the shapes of the of neighborhood function and on the product of the neighborhood and of the function that determines the learning rate (represented in the bi-dimensional case).

```
setLearningRateFunction[eta0 → 0.1, tau2 → 100];
```



When you have understood the meaning of the functions and parameters, change the default values and re-evaluate the notebook.

Observe first the effect of the variation of the parameters on the shape of the neighborhood function and on the product of the neighborhood and of the function that determines the learning rate

Then, observe the effect of the unfolding of the lattice of neurons.

```
setLearningRateFunction[eta0 → 1, tau2 → 1000];
```