Assignment 33

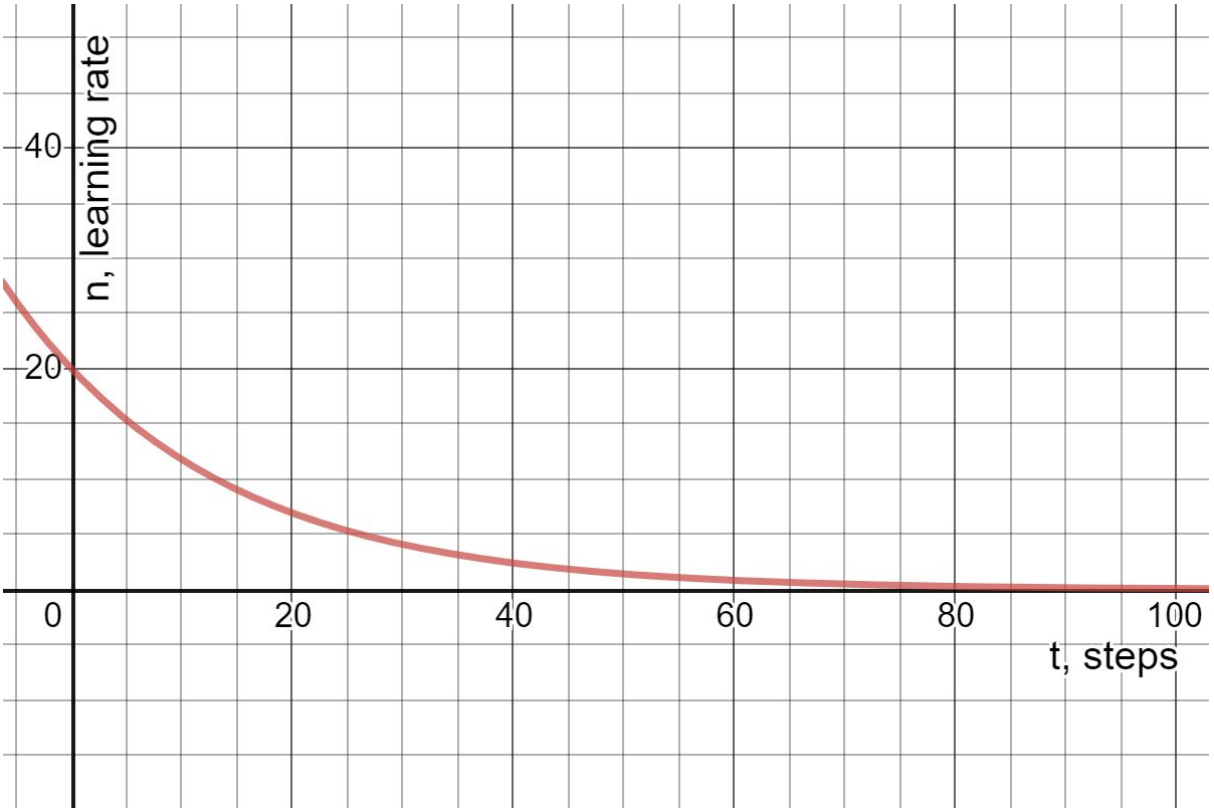
The major steps of training a SOM are:

1. **Initialization SOM**: this initial step is useful to:
    a. learn more about the task
    b. extract training data from the data set
    c. get information about the chosen pattern (like their labels)
    d. define the grid structure which specify which neurons are close to each other. The most used topology is a 2-dimensional rectangular grid structure, but it can be chosen wherever structure is prefered and the only limitation is given by the number of dimensions (g <= 3)
    e. define the network size, this specify the number of neurons that the grid has to include
    f. initialization of the learning parameter
        i. the learning rate (0.0 < **η** < 1.0 **)**,
        ii. the distance **s** that specify the dimension of the neighbourhood,
        iii. the neighbour function (Gaussian, step function, triangular, mexican hat).
    g. initialization of the centers vectors $\mathbf{C_k}$, this can be done completely randomly with respect of the area V, randomly with respect of the training data or with a starting configuration in case a prior knowledge is available.
2. **Selection and application of a stimulus**: a pattern, called stimulus $^{P}\mathbf{X}$, is chosen based on a probability distribution, after the selection is then applied to the learning process.
3. **Calculation of the response**: the stimulus is then used to calculate the similarity (euclidean distance is the most used) between the former and all the center $\mathbf{r_k}$.
4. **Determination of the winner**: knowing the distance from each centers, the winner is neuron with the lowest value (shortest distance: $|| \mathbf{C_i} - {}^{P}\mathbf{X} ||_2 \leq || \mathbf{C_j} - {}^{P}\mathbf{X} ||_2$ for i, j = 1…K).
5. **Learning rule**: once the winner is determined, its and all of the neighbours around its have to move in the direction of the pattern. Here it is computed the delta value that indicates the changing distance for each center: $\mathbf{\Delta C_j = \eta(t) * H(dist(i,j), t) * ({}^{P}X - C_j)}$. Where **i** is the index of the winner neuron, **j** is the index of the learning neuron, **η** is the learning rate and **t** specify the decaying of the learning rate, **h(...)** is the neighbourhood function, **dist(i,j)** is the distance between the neuron i and j defined on the grid, $^{P}\mathbf{X}$ is the stimulus, $\mathbf{C_j}$ is the center vector of the neuron j.
6. **Update step**: after computing the delta value of the winner and for the neighbors, they are summed to their old centers.
7. **Take the decision to continue**: here has to be decided whether to end the process or not. The process ends if: a predefined quality is reached, if the iteration has converged and the centers are not longer changing, after a predefined number of timesteps or if the user is bored.
8. **Labeling of the SOM neurons**: finally, all the neurons are labeled in order to assign a meaning to their position in the specific space.

Assignment 34

$$\eta(t) = \eta_{init} * (\eta_{final}/\eta_{init})^{t/t_{max}}$$

Where n(t) is the learning rate at step t, n_{init} is is the initial value of the learning rate, n_{final} is is the target value to which learning rate converges and t_{max} is the number of learning steps.
The graph below depicts the decay of n(t) with n_{init} = 20, n_{final} = 0.1 and t_{max} = 100



Assignment 36

The conditions that have to be met, to guarantee that the same neuron will be the winner after representing the same pattern used in the training process, are:

- **η(t)**: the learning rate has to decay in respect of the passed time. This allow to influence always less the movement of the centers from their position after t iteration in order to preserve the consistency of the previous learning steps.
- **dist(i, j)**: in order to consider a center neighbour of the winner, the distance between them has to be between 0 and S, where S is a predefined value that specifies the minimum distance.
- **h(dist(i, j), t)**: the distance between the neighbours of the winner center has to be computed by the neighbourhood function and the obtained value must be between 0 and 1.

Assignment 37

Clustering is a process performed on a dataset to group its entries based on given similarity criteria (Aggarwal & Yu, 1999), e.g. distance between data points. This means segmenting the data, or assigning group membership to each data point. In other words, this is unsupervised learning, which does not produce or predict any output value for patterns and just sorts the data during the learning and guesses group membership for unseen data points (Aggarwal, 2015). This is the aim of such clustering algorithms as K-means and such networks as SOM.

Classification also groups dataset into different clusters, or *classes.* But for each data point there is two types of information:

- "class label", or the desired discrete output indicating the class membership. It is either provided additionally or selected from the features in the input data
- "feature attributes", meaning the rest of the features for each data point which are not a class label (Aggarwal & Yu, 1999).

During the learning the relationship between "feature attributes" and the label is being modeled. Then the model can be used to predict not the group relationship, but the label, or the class of the input data (Aggarwal & Yu, 1999). This is done by supervised learning algorithms, where teacher values for each pattern are labels, and it can be done by RBF or MLP networks, or decision trees (Mitchell, 1999).

Example.

In insurance applications, the input for clustering might consist of vectors with demographic and medical information on persons. Clustering would reveal groups of people with similar information, which then can be treated as distinct types of clients.

The input for classification might also include the label - level of medical risk for each person. Then it might be learned as a 'function' of all other features, being equal for persons of the same group. Then, the level of risk can be predicted for new clients (Aggarwal & Yu, 1999).

Aggarwal, C. C., & Yu, P. S. (1999). Data Mining Techniques for Associations, Clustering and Classification. *Methodologies for Knowledge Discovery and Data Mining Lecture Notes in Computer Science*, 13–23. doi: 10.1007/3-540-48912-6_4

Aggarwal, C. C. (2015). Mining Text Data. *Data Mining*, 429–455. doi: 10.1007/978-3-319-14142-8_13

Mitchell, T. M. (1999). Machine learning and data mining. *Communications of the ACM*, *42*(11), 30–36. doi: 10.1145/319382.319388