Introduction to Kohonen Self-Organizing Maps

Kohonen Self-Organizing Maps (SOMs) are a type of artificial neural network invented by Professor Teuvo Kohonen. These maps are used to visualize high-dimensional data by mapping it onto a two-dimensional representation. The key features of Kohonen SOMs include a set of neurons that compete with each other to represent input patterns, with each neuron determining its output based on a weighted sum of its inputs.The neurons' weights and inputs are normalized, ensuring that their magnitudes are equal to one. During the competition, the neuron with the largest output is declared the winner and its output is set to one, while all other neurons in the layer are inhibited, having their outputs set to zero. This process allows different input patterns to activate different winner neurons, while identical patterns are classified by the same winning neuron.The Kohonen network consists of two layers: the input layer and the Kohonen output layer. The learning process in SOMs involves adjusting the weights of the neurons based on the input patterns. This is done through a process that models lateral inhibition and excitation, focusing on the neuron with the maximum output and designating it as the winner. Training is then applied to all neurons within a certain neighborhood size of the winning neuron, while those outside this neighborhood do not participate in the weight update.An example of a Kohonen network might include an input layer with two nodes and an output layer with several nodes, each representing a potential winning neuron. The initial step in the learning process involves randomizing the weight values, preferably within the range of the input patterns, to start the training process.As the network learns, the neighborhood size and the learning rate (alpha) are typically reduced, refining the network's ability to classify input patterns and improving its representation of the data. The SOM's ability to reduce dimensions while preserving the topological properties of the input space makes it a powerful tool for pattern recognition and data visualization.

Normalization of Vectors in SOM

Normalization of vectors is a crucial step in the process of training Self-Organizing Maps (SOM). It involves scaling the input vectors to a common scale without distorting differences in the ranges of values. In SOM, normalization is typically done to ensure that each input vector contributes equally to the training process.To normalize a vector, you calculate the magnitude of the vector and then divide each component of the vector by this magnitude. The magnitude of a vector A with components x, y, and z is calculated using the Euclidean norm, which is the square root of the sum of the squares of the components: Magnitude $(n) = 1 / \sqrt{(x^2 + y^2 + z^2)}$Once the magnitude is calculated, the normalized vector A' is obtained by multiplying each component of the original vector by the calculated magnitude (n):$A' = (xn, yn, zn)$This normalization process is applied to the weight vectors in the input to the Kohonen feature map layer. By normalizing the weight vectors, the SOM algorithm can compare the distances between vectors in a consistent manner, which is essential for the map's learning and organization.During the training of the SOM, the normalized weight vectors are used to calculate the activation of the nodes in the output layer. The node with the highest activation is considered the winning neuron, and its weight vector is updated along with the weight vectors of its neighbors, depending on the neighborhood size. The normalization ensures that the weight vectors are kept within a reasonable range, facilitating the convergence of the network during training.

Lateral Inhibition in Neural Networks

Lateral inhibition is a key process in some biological neural networks, which involves the formation of lateral connections between neurons within a given layer. These connections play a crucial role in the way neurons interact with each other. The primary function of lateral inhibition is to enhance the contrast in the signal being processed, which helps in emphasizing the most stimulated areas of the neural network.In the context of neural networks, particularly those modeled after biological processes, lateral inhibition works by strengthening the signal of the most active neuron, often referred to as the 'winner', while simultaneously weakening the signals of its neighboring neurons. This is achieved by forming excitatory connections that support the activity of the winner neuron, and inhibitory connections that suppress the activity of its neighbors.The strength of these lateral connections is typically inversely related to the distance between neurons. Neurons that are closer to the winner neuron are more strongly inhibited than those further away. This creates a competitive environment where only the strongest signals prevail, leading to a sparse and efficient representation of the input data.During the training phase of a neural network, such as a Kohonen map, lateral inhibition is modeled by identifying the neuron with the maximum output and designating it as the winner. The outputs of all other neurons are then inhibited, which is often implemented by setting their outputs to zero. This ensures that the training or weight update is focused on the winner neuron and its immediate neighbors within a specified neighborhood size. Neurons outside this neighborhood are not updated, which reinforces the local feature mapping of the network. The concept of lateral inhibition is not only fundamental to the operation of biological neural networks but also to artificial neural networks where it is used to refine learning and feature detection. By simulating this biological process, artificial networks can achieve more nuanced and detailed pattern recognition.

The Mexican Hat Function

The Mexican Hat Function is a graphical representation that illustrates the range of influence a neuron has during the training process in a Kohonen network. It specifies the distance of participation in training and weight vector changes, effectively determining which neurons will be affected by a given input. Neurons located within the specified distance from the winning neuron are allowed to participate in the training and have their weight vectors adjusted. Conversely, neurons that fall outside this distance do not participate in the training process.As the training progresses, the neighborhood size, which is influenced by the Mexican Hat Function, typically decreases. This gradual reduction ensures that the learning becomes more focused and refined, eventually leading to a situation where only the single winning neuron is updated. This process helps the network to fine-tune its responses to the input data and to form distinct categories or classes based on the input patterns it has been exposed to.

Training Law for the Kohonen Map

The Training Law for the Kohonen Map is a fundamental principle that guides the adjustment of weights in the network. This law states that the change in the weight vector for a given output neuron is determined by a gain constant, denoted as alpha (α), which is multiplied by the difference between the input vector and the old weight vector. The formula for updating the weight is as follows:Wnew = Wold + α * (Input - Wold)Here, Wnew represents the new weight vector, Wold is the old weight vector, α is the gain constant, and Input is the current input vector. The gain constant α is a crucial parameter that influences the learning rate of the network. It typically ranges between 0.01 and 1.The purpose of this training law is to adjust the weights of the neurons in the network so that they become closer to the input vectors. This process is essentially a form of learning or memorization, where the network adapts its weights to better classify the input data set into different classes. By repeatedly applying this law during the training phase, the Kohonen Map can effectively organize itself to map input patterns to specific output neurons, thereby creating a feature map that can classify input patterns based on their similarities.

Neighborhood Size and Learning Rate (Alpha) in SOM

In Self-Organizing Maps (SOM), two critical parameters that influence the learning process are the neighborhood size and the learning rate, often denoted as alpha. During the training phase, the neighborhood size determines the range of influence a winning neuron has over its neighbors. Initially, a larger neighborhood size allows for global ordering of the map, but as learning progresses, the neighborhood size is gradually decreased. This decrement can occur every set number of periods, which can be determined by the user or programmer. The reduction in neighborhood size ensures that the learning becomes more localized, allowing for fine-tuning of the map's topology.The learning rate, alpha, is another dynamic parameter that is crucial for the convergence of the SOM. It dictates the degree to which the winning neuron and its neighbors are adjusted during each learning cycle. Alpha is typically set to a higher value at the beginning of the training to allow for significant adjustments to the neuron weights, facilitating faster initial learning. As the training cycles continue, alpha is decremented by a fixed amount, such as 0.1, after every cycle set of patterns. The decrement continues until alpha reaches a lower limit, which is often set to 0. This gradual reduction in the learning rate helps to stabilize the network as it converges to a solution.The training process of the SOM is considered complete when one of two conditions is met: either the neighborhood size has been reduced to 0, signifying that only the winning neuron is updated, or the average distance pattern, calculated as the total distance in a cycle divided by the number of patterns per cycle, is less than or equal to a required distance threshold, typically around 0.05. These conditions ensure that the map reflects the input data's topology with sufficient accuracy before the training is halted.

Summary of Kohonen SOM Features

The Kohonen Self-Organizing Map (SOM) is a neural network model that features neurons in competition. Each neuron calculates its output as a weighted sum of its inputs, with both weights and inputs being normalized to a magnitude of one. The neuron with the highest output is deemed the winner and is assigned an output value of one, while all other neurons in the layer are assigned an output of zero. This process allows the SOM to classify different input patterns by activating distinct winner neurons, while identical patterns are classified to the same winning neuron.The learning rate, denoted as alpha, is reduced by 0.1 after each cycle of pattern presentations until it reaches zero. The neighborhood size, which influences the learning of surrounding neurons, is also decreased periodically at an interval determined by the user or programmer.The SOM training process concludes when the neighborhood size becomes zero or when the average distance between the input pattern and the winning neuron is less than or equal to a predefined threshold, typically 0.05. The average distance for a cycle is calculated by dividing the total distance in that cycle by the number of patterns presented during the cycle.