SNS COLLEGE OF TECHNOLOGY

An Autonomous Institution

Coimbatore-35

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

19ECT303-ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

III YEAR/ V SEMESTER

UNIT 4 – NEURONS AND NEURAL NETWORKS

4.6 Back Propagation –Dimensionality Reduction

BACK PROPAGATION

Introduction

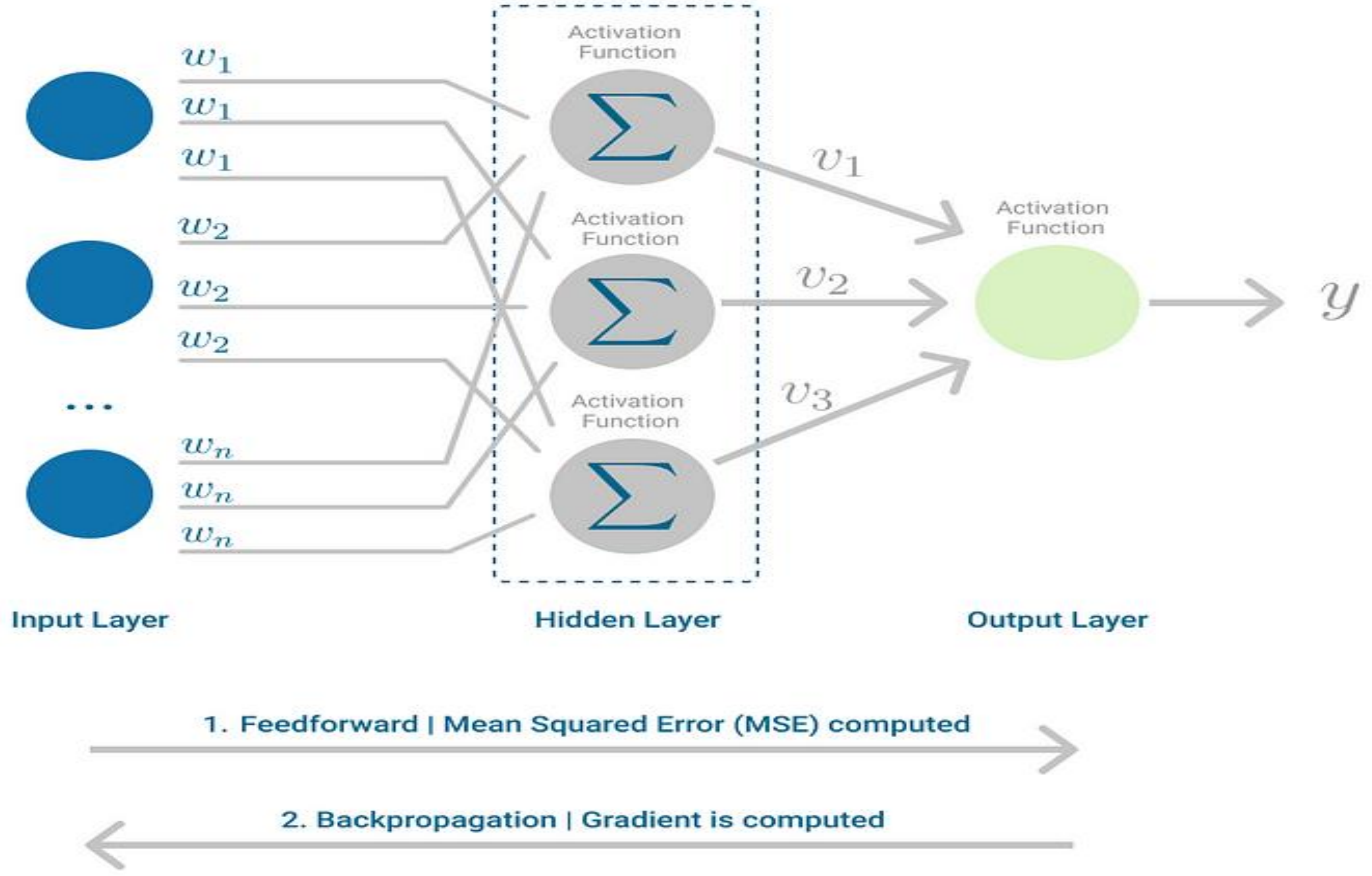
•learning mechanism that allows the Multilayer Perceptron to iteratively adjust the weights in the network, with the goal of minimizing the cost function.

•There is one hard requirement for backpropagation to work properly.

•The function that combines inputs and weights in a neuron, for instance the weighted sum, and the threshold function, for instance ReLU, must be differentiable.

•These functions must have a **bounded derivative**, because Gradient Descent is typically the optimization function used in MultiLayer Perceptron.

BACK PROPAGATION



BACK PROPAGATION

•In each iteration, after the weighted sums are forwarded through all layers, the gradient of the **Mean Squared Error** is computed across all input and output pairs.

•Then, to propagate it back, the weights of the first hidden layer are updated with the value of the gradient. That's how the weights are propagated back to the starting point of the neural network!

$$\underbrace{\Delta_w(t)}_{\text{Gradient Current Iteration}} = -\overset{\text{Bias}}{\varepsilon}\,\frac{\overbrace{dE}^{\text{Error}}}{\underbrace{dw_{(t)}}_{\text{Weight vector}}} + \overset{\text{Learning Rate}}{\alpha}\,\underbrace{\Delta_{w(t-1)}}_{\text{Gradient Previous Iteration}}$$

BACK PROPAGATION

This process keeps going until gradient for each input-output pair has converged, meaning the newly computed gradient hasn't changed more than a specified ***convergence threshold***, compared to the previous iteration.

Back Propagation Algorithm

It is used in a Multilayer perceptron neural network to increase the accuracy of the output by reducing the error in predicted output and actual output.

1. After calculating the output from the Multilayer perceptron neural network, calculate the error.
2. This error is the difference between the output generated by the neural network and the actual output. The calculated error is fed back to the network, from the output layer to the hidden layer.
3. Now, the output becomes the input to the network.
4. The model reduces error by adjusting the weights in the hidden layer.
5. Calculate the predicted output with adjusted weight and check the error. The process is recursively used till there is minimum or no error.
6. This algorithm helps in increasing the accuracy of the neural network.

BACK PROPAGATION

Difference Between Multilayer Perceptron Neural Network and Conventional Neural Network

	Multi Layer Perceptron Neural Network	**Convolutional Neural Network**
Types of Input	It takes vector inputs.	It takes both vectors and matrices as input.
Network Type	It is a fully connected Neural network	It is a spatially connected neural network.
Focus Problem	It can deal with non-linear problems.	Can only deal with linear problems.
Application	It is good for simple image classification.	It is mostly used for complex image classification.

BACK PROPAGATION

Advantages of MultiLayer Perceptron Neural Network

1. MultiLayer Perceptron Neural Networks can easily work with non-linear problems.
2. It can handle complex problems while dealing with large datasets.
3. Developers use this model to deal with the fitness problem of Neural Networks.
4. It has a higher accuracy rate and reduces prediction error by using backpropagation.
5. After training the model, the Multilayer Perceptron Neural Network quickly predicts the output.

Disadvantages of MultiLayer Perceptron Neural Network

1. This Neural Network consists of large computation, which sometimes increases the overall cost of the model.
2. The model will perform well only when it is trained perfectly.
3. Due to this model’s tight connections, the number of parameters and node redundancy increases.

DIMENSIONALITY REDUCTION

Technique of reducing the feature space to obtain a stable and statistically sound machine learning model avoiding the Curse of dimensionality.

1. Feature Selection

To subset important features and remove collinear or not-so-important features. One can read more about it, here.

2. Feature Transformation.(Feature Extraction tries)

To project the high-dimensional data into lower dimensions. Some Feature Transformation techniques are **PCA (already discussed)** , Matrix-Factorisation, **Autoencoders, t-SNE, UMAP, etc.**

UMAP - ***Uniform Manifold Approximation And Projection***

•nonlinear dimensionality reduction technique.

•Visually, it is similar to t-SNE, but it assumes that the data is uniformly distributed on a locally connected Riemannian manifold and that the Riemannian metric is locally constant or approximately locally constant.

DIMENSIONALITY REDUCTION

Autoencoder

To learn nonlinear dimension reduction functions and codings together with an inverse function from the coding to the original representation.

t-SNE (*t-distributed Stochastic Neighbor Embedding)*

•nonlinear dimensionality reduction technique useful for visualization of high-dimensional datasets.

• It is not recommended for use in analysis such as clustering or outlier detection since it does not necessarily preserve densities or distances well.

Applications

• Neuroscience is maximally informative dimensions, which finds a lower-dimensional representation of a dataset such that as much information as possible about the original data is preserved.

• noise reduction,

•data visualization,

•cluster analysis, or as an intermediate step to facilitate other analyses.

•Feature selection

DIMENSIONALITY REDUCTION