(SUBJECT:ARTIFICIAL NEURAL NETWORK) (SUBJECT CODE:24SBT113)

AI Individual Task on

"TRAIN A PERCEPTRON O AND/OR TASKS"

Submitted in the partial fulfillment of the requirements for the fourth semester

BACHELOR OF TECHNOLOGY IN AIML

Submitted By,

VANDANA S V(01SU24AI113)

UNDER THE GUIDANCE OF

Prof. Mahesh Kumar V B

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

SRINIVAS UNIVERSITY INSTITUTE OF ENGINEERING & TECHNOLOGY MUKKA, MANGALURU-574146

2025-26

SRINIVAS UNIVERSITY
INSTITUTEOFENGINEERING&TECHNOLOGY

DEPARTMENT ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

CERTIFICATE

This is to certify that VANDANA S V (01SU24AI113) has satisfactorily completed the assessment (Individual-Task – Module 2) in **"ARTIFICIAL NEURAL NETWORK"** prescribed by the Srinivas University for the 4st semester B. Tech course during the year **2025-26**.

MARKS AWARDED

Staff In charge

Name: Prof. Mahesh Kumar V B

Assistant Professor ,Department Of AIML

Abstract

This report demonstrates the implementation of the Perceptron model using the Error-Correction learning rule. The Perceptron is trained to perform binary classification on AND and OR logic gate problems. The learning process updates weights based on classification error. The decrease in error over epochs is plotted to study convergence behavior. The impact of learning rate on convergence is also discussed.

The learning process involves adjusting the weights based on the difference between the target output and the predicted output. This difference is called the error. During training, the total error decreases over epochs, showing convergence. The effect of the learning rate on convergence speed and stability is also analyzed. The experiment confirms that the Perceptron successfully learns AND and OR functions using the error-correction rule.

Introduction

The Perceptron is a single-layer neural network used for solving linearly separable binary classification problems. It uses a step activation function and updates weights using the error-correction rule.

The learning process involves adjusting the weights based on the difference between the target output and the predicted output. This difference is called the error. During training, the total error decreases over epochs, showing convergence. The effect of the learning rate on convergence speed and stability is also analyzed. The experiment confirms that the Perceptron successfully learns AND and OR functions using the error-correction rule..

One of the earliest and simplest neural network models is the Perceptron, introduced by Frank Rosenblatt in 1958. The Perceptron is a single-layer neural network used for solving binary classification problems. It can classify inputs into two categories, such as 0 or 1, Yes or No, True or False.

The Perceptron works effectively when the dataset is linearly separable, meaning the data points can be separated using a straight line (in two dimensions) or a hyperplane (in higher dimensions). It calculates a weighted sum of the input features and passes the result through a step activation function to generate the final output.

The learning process of a Perceptron is based on the error-correction learning rule. During training, the model compares its predicted output with the actual target output. If there is an error, the weights are adjusted proportionally to the error and the input values. This iterative process continues until the model correctly classifies all training samples or reaches a maximum number of epochs.

In this experiment, the Perceptron model is trained to learn two basic logic gates:

- **AND logic gate**
- **OR logic gate**

These logic gates are fundamental digital operations in computer systems and are commonly used to demonstrate the learning capability of neural networks. Since both AND and OR functions are linearly separable, they are ideal examples for training a single-layer Perceptron.

The main objective of this report is to demonstrate how the error-correction learning rule helps the Perceptron adjust its weights, reduce classification error over time, and achieve convergence. Additionally, the effect of learning rate on training speed and stability is analyzed through experimental observation.

Perceptron Learning Rule

Weight Update Rule:

W(new) = W(old) + η * (Target - Output) * Input

Where, **η is the learning rate.**

Truth Tables

AND Gate Truth Table:

X1	**X2**	**Output**
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate Truth Table:

X1	X2	Output
0	0	0
0	1	1
1	0	1
1	1	1

Training Results

The Perceptron was trained for 10 epochs with learning rate $\eta = 0.1$. The total error per epoch is shown in the following figures.

GRAPH RSULTS:

Figure 1: AND Gate Error Convergence

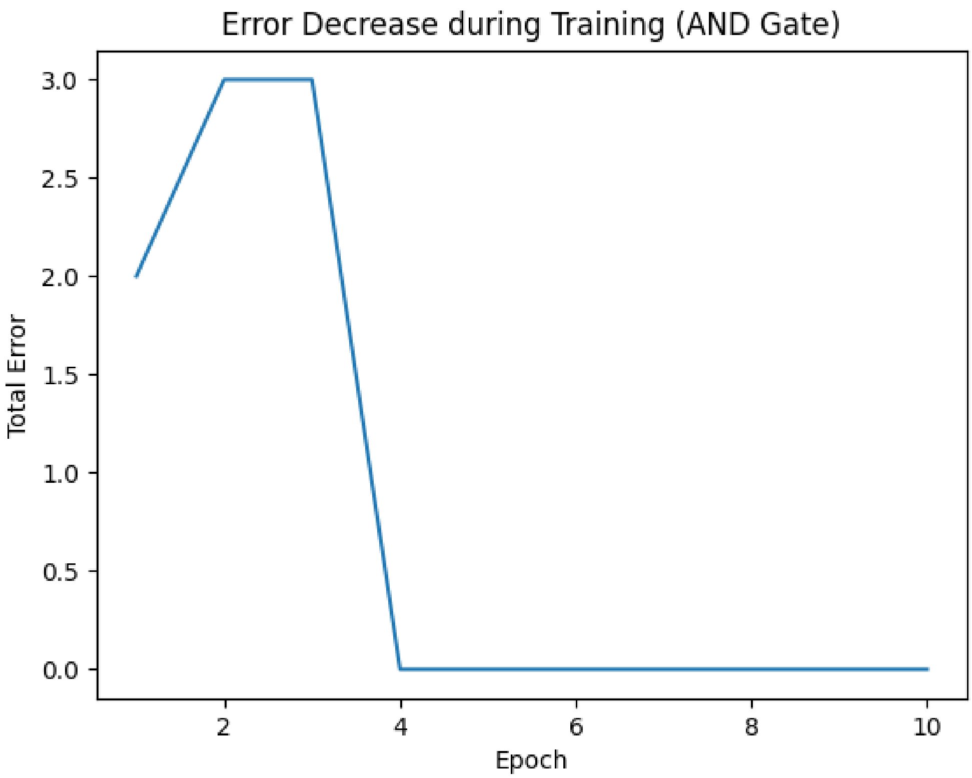
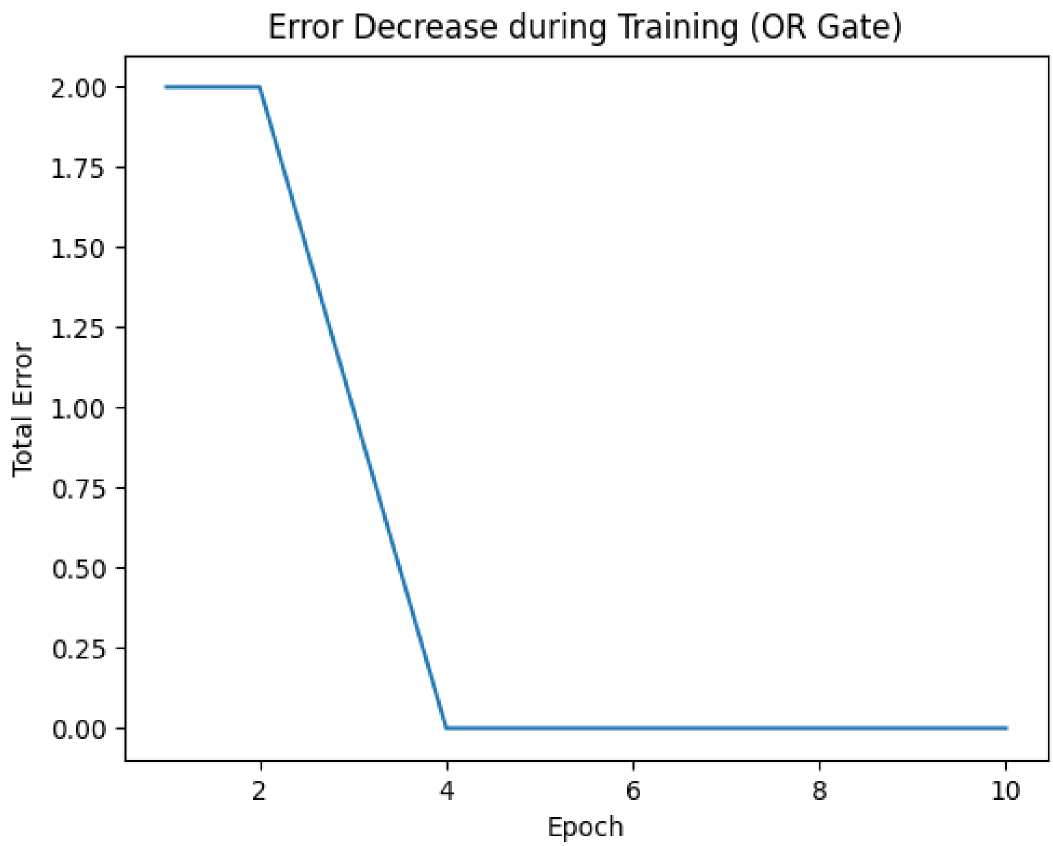


Figure 2: OR Gate Error Convergence



Learning Rate and Convergence Analysis

In this experiment, the Perceptron model was trained using the error-correction learning rule with a learning rate (η) of 0.1. The learning rate is a crucial parameter in neural network training because it determines the magnitude of weight updates during each iteration.

According to the Perceptron learning rule:

W=(new) = W(old)+ η (Target-Ouput)*Input

the learning rate η directly influences how much the weights are adjusted whenever a misclassification occurs. If the predicted output differs from the target output, the error term (Target – Output) becomes non-zero, and the weights are updated proportionally to η.

Convergence in AND and OR Tasks

The AND and OR logic gates are linearly separable problems. This means that their input-output pairs can be separated using a straight line in a two-dimensional space.

Because of this property:

- The Perceptron is guaranteed to converge.
- The error eventually becomes zero.
- No further weight updates occur after convergence.

In the provided graphs:

- The AND gate required a few epochs before reaching zero error.
- The OR gate converged slightly faster due to simpler separation.
- Once convergence is achieved, the weights stabilize and the decision boundary correctly classifies all training samples

Overall Observation

In this experiment, $\eta = 0.1$ provided:

- Stable training behavior
- Smooth error reduction
- Convergence within 10 epochs
- No oscillation or instability

Therefore, the learning rate played a critical role in ensuring efficient and successful training of the Perceptron model.

Conclusion

This experiment successfully demonstrated the implementation of a Perceptron model using the error-correction learning rule for solving binary classification problems. The Perceptron was trained on two fundamental logic gates — AND and OR — which are linearly separable problems. The training process showed that the model was able to adjust its weights iteratively based on classification error and eventually achieve zero misclassification.

The results clearly indicate that the error decreased progressively over epochs, as observed in the convergence graphs. During the initial training stages, the model produced some incorrect outputs due to randomly initialized weights. However, through repeated weight updates using the error-correction rule, the Perceptron gradually learned the correct decision boundary. Once all training samples were correctly classified, the total error became zero and remained constant, confirming convergence.

The experiment also highlighted the importance of the learning rate parameter. A properly chosen learning rate ensured smooth and stable training without oscillations. In this case, $\eta = 0.1$ provided balanced weight updates, allowing the model to converge efficiently within a limited number of epochs. This demonstrates that learning rate selection plays a critical role in determining both convergence speed and stability.

Furthermore, the experiment reinforces the fundamental concept that the Perceptron can only solve linearly separable problems. Since AND and OR gates satisfy this condition, the model successfully converged. However, for non-linearly separable problems such as XOR, a single-layer Perceptron would fail to converge. This limitation led to the development of multi-layer neural networks and more advanced learning algorithms.

Overall, this experiment provides a clear understanding of:

- The working mechanism of a single-layer Perceptron
- The application of the error-correction learning rule
- The role of learning rate in training dynamics
- The concept of convergence in linearly separable problems

Thus, the Perceptron serves as a foundational model in Artificial Neural Networks and helps in understanding more complex neural network architectures. The successful training on AND and OR tasks confirms that the Perceptron is an effective and simple model for basic binary classification problems.