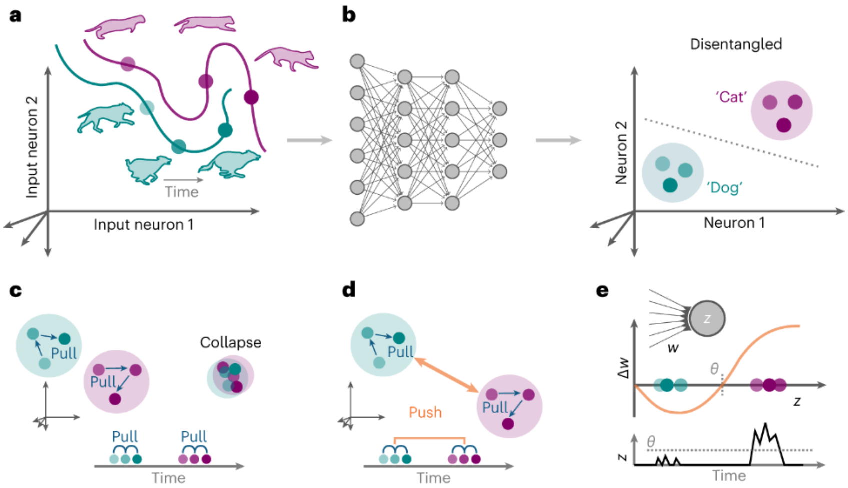
MODULE 3 : GROUP TASK

Introduction

Learning is the fundamental mechanism through which artificial neural networks adapt and improve. A learning rule defines how model parameters are updated during training. Over decades, multiple learning paradigms have emerged, each suited for different problem types.



This report focuses on four important learning strategies:

- Hebbian Learning
- Error-Correction Learning
- Reinforcement Learning
- Stochastic Learning

The comparison evaluates them on:

- Learning dynamics (how updates occur)
- Stability (robustness of training)
- Convergence (guarantee and speed)

Understanding these differences helps in selecting appropriate algorithms for real-world applications.

Hebbian Learning

2.1 Concept

Hebbian learning is based on the biological principle:

“Neurons that fire together, wire together.”

This rule was introduced by Donald Hebb in 1949.

It is an unsupervised learning rule, meaning it does not require target output.

2.2 Mathematical Formulation

Weight update:

$$\Delta w_{ij} = \eta x_i y_j$$

Where:

- (x_i) = input neuron
- (y_j) = output neuron
- (η) = learning rate

Weights increase when input and output are both active.

2.3 Learning Dynamics

- Strengthens correlations.
- Does not minimize error.
- Purely associative learning.

2.4 Stability

- Can cause unbounded weight growth.

- Requires normalization to stabilize.

2.5 Convergence

- No explicit convergence guarantee.
- May diverge without constraints.

3. Error-Correction Learning

3.1 Concept

Error-correction learning is supervised learning.

Weights are adjusted based on:

$$e = d - y$$

Where:

- (d) = desired output
- (y) = predicted output

3.2 Update Rule

$$\Delta w_i = \eta (d - y) x_i$$

3.3 Learning Dynamics

- Reduces classification error.
- Moves decision boundary toward correct separation.
- Iterative refinement.

3.4 Stability

- Stable if learning rate properly chosen.
- Too high learning rate causes oscillation.

3.5 Convergence

- Guaranteed convergence for linearly separable data.
- Faster convergence than Hebbian learning.

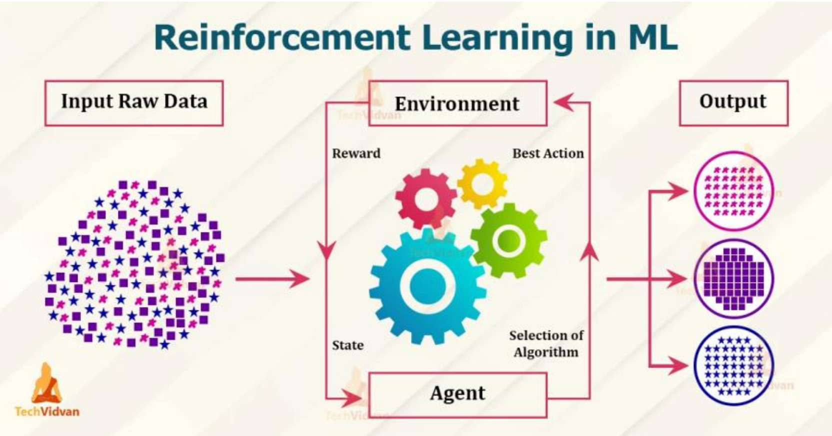
Hebbian vs Error-Correction Learning (Comparative Analysis)

Feature	Hebbian	Error-Correction
Type	Unsupervised	Supervised
Uses Target Output	No	Yes
Objective	Strengthen correlations	Reduce classification error
Stability	Can diverge	Stable with proper η
Convergence	No guarantee	Guaranteed (linear cases)
Application	Feature learning	Classification tasks

Key Insight:

Hebbian learning captures patterns but does not ensure correct classification. Error-correction explicitly minimizes error.

Reinforcement Learning



5.1 Concept

Reinforcement learning (RL) is learning by reward and punishment.

An agent interacts with environment:

- Takes action
- Receives reward
- Updates policy

5.2 Mathematical Representation

Typical update:

$$Q(s,a) = Q(s,a) + \eta [r + \gamma \max Q(s',a') - Q(s,a)]$$

Where:

- (r) = reward
- (γ) = discount factor

5.3 Learning Dynamics

- Delayed feedback.

- Exploration vs exploitation trade-off.
- Trial-and-error learning.

5.4 Stability

- Can be unstable with high learning rate.
- Sensitive to reward structure.

5.5 Convergence

- Converges under specific theoretical conditions.
- May require many iterations.

Stochastic Learning

6.1 Concept

Stochastic learning updates weights using randomly selected data samples.

Common example:

- Stochastic Gradient Descent (SGD)

6.2 Update Rule (SGD Example)

[
w = w - \eta \nabla L(w)
]

Using one sample at a time.

6.3 Learning Dynamics

- Faster updates.

- Noisy but efficient.
- Escapes local minima.

6.4 Stability

- Fluctuates due to randomness.
- Averaging improves stability.

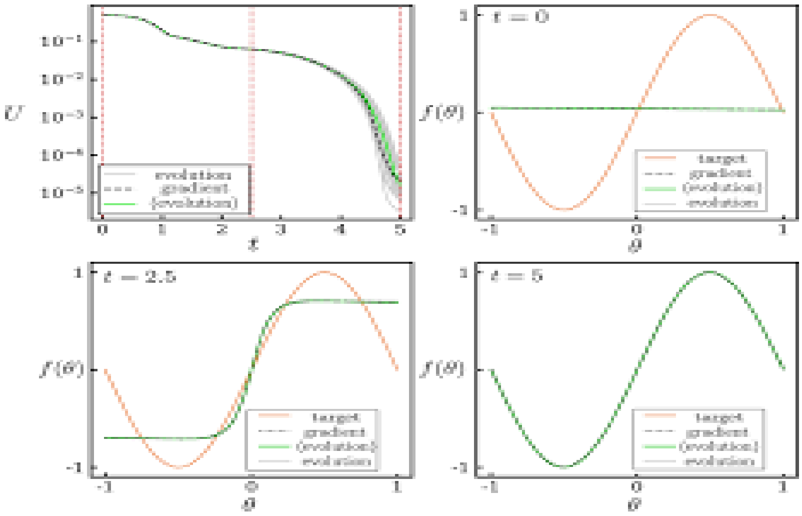
6.5 Convergence

- Converges in expectation.
- Faster than batch learning.

Reinforcement vs Stochastic Learning (Comparative Analysis)

Feature	Reinforcement Learning	Stochastic Learning
Feedback Type	Reward signal	Loss gradient
Supervision	Indirect	Direct
Update Style	Policy-based	Gradient-based
Stability	Reward-dependent	Noisy but manageable
Convergence	Slower	Faster
Application	Robotics, Games	Deep learning models

Learning Dynamics Comparison



Hebbian

- Strength-based learning.
- Local update rule.
- Correlation-driven.

Error-Correction

- Error-driven.
- Goal-oriented.
- Global supervision.

Reinforcement

- Reward-driven.
- Sequential decision making.
- Environment interaction.

Stochastic

- Randomized gradient-based.
- Data-efficient.
- Scalable.

Stability Analysis

Learning Rule	Stability Level	Risk Factors
Hebbian	Low	Unbounded growth
Error-Correction	High	High η
Reinforcement	Moderate	Poor reward design
Stochastic	Moderate-High	High variance

Convergence Comparison

Learning Rule	Convergence Guarantee
Hebbian	No guarantee
Error-Correction	Guaranteed (linear separable case)
Reinforcement	Theoretical guarantee under constraints
Stochastic	Converges in expectation

Poster Summary Section

🔵 Hebbian vs Error-Correction

Hebbian

- Unsupervised
- Correlation-based
- No explicit error minimization
- May diverge

Error-Correction

- Supervised
- Error-based updates
- Converges for linear problems
- Stable with proper learning rate

🟢 Reinforcement vs Stochastic

Reinforcement

- Reward-driven
- Delayed feedback
- Exploration needed
- Slower convergence

Stochastic

- Random gradient updates
- Faster learning
- Slightly noisy
- Widely used in deep learning

Applications in Real Systems

Learning Rule	Real-World Application
Hebbian	Feature extraction, associative memory
Error-Correction	Perceptron, logistic regression
Reinforcement	Game AI, robotics
Stochastic	Deep neural networks

Advantages and Limitations

Hebbian

- Simple
- Biologically inspired
  – No error correction
  – Unstable without normalization

Error-Correction

- Accurate classification
- Convergence guarantee
  – Requires labeled data

Reinforcement

- Learns complex policies
- No labeled dataset needed
  – Computationally expensive

Stochastic

- Efficient for large datasets
- Escapes local minima
  – Noisy updates

Conclusion

This comparative study of learning rules—Hebbian learning, error-correction learning, reinforcement learning, and stochastic learning—provides a comprehensive understanding of how different neural learning paradigms influence training behavior, stability, and convergence properties. Each learning rule represents a distinct philosophy of adaptation, and their differences significantly impact model performance and application suitability.

Reinforcement learning introduces a fundamentally different approach by relying on reward signals rather than explicit target labels. Learning occurs through interaction with an environment, where actions are reinforced positively or negatively. This delayed feedback mechanism makes reinforcement learning suitable for complex sequential decision-making problems such as robotics, autonomous navigation, and game playing. However, convergence in reinforcement learning depends on exploration strategies, reward design, and algorithmic constraints. Improper reward structures may lead to unstable or suboptimal learning behavior.

From a convergence perspective, error-correction learning offers strong theoretical guarantees in specific conditions (linearly separable data). Stochastic learning provides convergence guarantees in expectation under diminishing learning rates. Reinforcement learning converges under more restrictive mathematical assumptions, and Hebbian learning lacks a formal error-minimization convergence framework.

- For associative pattern recognition → Hebbian learning is suitable.
- For supervised classification → Error-correction learning is effective.
- For sequential decision-making → Reinforcement learning is ideal.
- For large-scale optimization → Stochastic learning is preferred.

In conclusion, this study highlights that learning rules define not only how a neural network updates its parameters but also how it behaves dynamically during training. Stability, convergence speed, and robustness are all deeply influenced by the chosen learning mechanism. A thorough understanding of these learning paradigms is essential for designing efficient, scalable, and reliable artificial intelligence systems.