

# Elastic Weight Consolidation (EWC)

## 🎯 EWC Weight Importance Mechanism

Task A (Previous Task)  
Weight Importance:  
 High (Fixed) Low

Fisher Information  
Matrix F

Task B (New Task)  
Learning Constraints:  
 Constrained Free

## ⚡ EWC Loss Function

$$L_{\text{total}} = L_{\text{new}} + \lambda \cdot L_{\text{EWC}}$$

New task loss + Important weight preservation penalty = Balanced learning

## 🌟 Effects

- ✓ 중요 지식 보존
- ✓ 유연한 학습
- ✓ 메모리 효율적

## 🎯 EWC Mechanism

- Fisher Information Matrix: Identify important weights
- Weight importance calculation: Identify parameters important to previous tasks
- Add constraints: Limit important weight changes
- Maintain flexibility: Freely update less important weights

## Loss Function

- $L_{\text{total}} = L_{\text{new}} + \lambda * L_{\text{EWC}}$
- $L_{\text{EWC}} = \sum F_i * (\theta_i - \theta^*_i)^2$
- $F_i$ : Fisher information (weight importance)
- $\lambda$ : Regularization strength (typically 1-1000)

## Penalty Application

- High penalty for important weight changes
- Freely learn less important weights
- Cumulative penalty across multiple tasks
- Memory efficient: Store only Fisher matrix