제공해주신 신경망 학습 자료(5장 오차역전파, 6장 학습 기술)를 바탕으로, 핵심 개념과 이를 증명하는 수학적 풀이/수식 위주로 A4 두 페이지 분량으로 정리했습니다.

---

# 신경망 학습의 수학적 기반 및 최적화 원리 요약

## I. 오차역전파법 (Backpropagation)의 수학적 기초 (CHAPTER 5)

오차역전파법은 **연쇄 법칙(Chain Rule)**을 이용하여 신경망의 가중치에 대한 손실 함수의 기울기를 효율적으로 계산하는 방법입니다.

### 1. 연쇄 법칙 (Chain Rule)

합성 함수 $z = f(y)$이고 $y = g(x)$일 때, $z$의 $x$에 대한 미분은 다음과 같습니다.

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial x}$$
신경망에서 이 법칙은 **순전파(Forward)**의 결과를 역순으로 계산하여 국소적 미분을 상류에서 하류로 전달하는 방식으로 적용됩니다.

### 2. 기본 계층의 역전파 수식 유도 및 풀이

#### 2.1. 곱셈 노드 (MulLayer)

입력 $x, y$에 대해 $z = xy$ 일 때, 출력 $z$에 대한 손실 $L$의 미분($\frac{\partial L}{\partial z} = dout$)이 상류에서 들어옵니다.

- 역전파 수식:
  $$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial x} = dout \cdot y$$
  $$\frac{\partial L}{\partial y} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial y} = dout \cdot x$$
- 풀이 원리: 미분값이 순전파 입력의 상대방 값에 상류 미분값을 곱하여 하류로 전달됩니다.

#### 2.2. 덧셈 노드 (AddLayer)

입력 $x, y$에 대해 $z = x+y$ 일 때,

- 역전파 수식:
  $$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial x} = dout \cdot 1 = dout$$

$$\frac{\partial L}{\partial y} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial y} = dout \cdot 1 = dout$$

- 풀이 원리**:** 입력값에 관계없이 상류에서 들어온 미분값을 그대로 하류로 흘려보냅니다**.**

### 2.3. **Sigmoid** 계층

입력 $x$에 대해 출력 $y = \frac{1}{1 + e^{-x}}$ 일 때,

- 수식 유도**:**
  $$\frac{\partial y}{\partial x} = \frac{\partial}{\partial x} (1 + e^{-x})^{-1} = -1 \cdot (1 + e^{-x})^{-2} \cdot (-e^{-x}) = \frac{e^{-x}}{(1 + e^{-x})^2}$$
  $$= \frac{1}{1 + e^{-x}} \cdot \frac{e^{-x}}{1 + e^{-x}} = y \cdot (1 - y)$$
- 최종 역전파 수식**:** 상류 미분 $dout$이 들어올 때,
  $$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial y} \cdot \frac{\partial y}{\partial x} = dout \cdot y(1 - y)$$
- 풀이 원리**:** Sigmoid의 순전파 출력값($y$)만으로 역전파를 계산할 수 있습니다.

### 2.4. **Affine** 계층 **(**배치 처리**)**

입력 행렬 $X$ (N행 D열), 가중치 $W$ (D행 H열), 편향 $B$ (H열)에 대해 $Y = XW + B$ 일 때,

- 역전파 수식 **(**형상 중요**):**
  $$\frac{\partial L}{\partial X} = \frac{\partial L}{\partial Y} \cdot W^T \quad \text{(N행 D열)}$$
  $$\frac{\partial L}{\partial W} = X^T \cdot \frac{\partial L}{\partial Y} \quad \text{(D행 H열)}$$
  $$\frac{\partial L}{\partial B} = \sum_{N \text{축}} \frac{\partial L}{\partial Y} \quad \text{(1행 H열)}$$
- 풀이 원리**:** 역전파 시 $\frac{\partial L}{\partial Y}$와 행렬 곱을 수행할 때 전치**($T$)** 행렬을 사용하여 형상을 일치시켜야 합니다. 편향 $B$의 기울기는 배치($N$축)에 걸쳐 합산됩니다.

### 2.5. **Softmax-with-Loss** 계층

소프트맥스 출력 $y$와 정답 레이블 $t$에 대해 손실 $L$의 기울기는 놀랍도록 단순합니다.

- 역전파 수식**:**
  $$\frac{\partial L}{\partial X} = \frac{1}{N} (y - t)$$
  (단, $N$은 배치 크기이며, $y$와 $t$는 벡터 또는 행렬입니다.)
- 풀이 원리**:** **(예측값 - 정답 레이블)**의 형태로 오차가 계산되어 앞 계층으로 전달됩니다. 이는 학습 과정에서 직관적이고 효율적인 오차 신호를 제공합니다.

---

# II. 학습 관련 기술 및 최적화 수학 **(CHAPTER 6)**

## 1. 매개변수 갱신 방법 **(Optimizers)**

모든 갱신 방법은 손실 함수 $L$에 대한 매개변수 $W$의 기울기 $\frac{\partial L}{\partial W}$를 기반으로 합니다.

| 기법 | 갱신 수식 **(Update Rule)** | 개념 및 특징 |
|---|---|---|
| **SGD** | $W \leftarrow W - \eta \frac{\partial L}{\partial W}$ | 기울기 방향으로 일정 속도($\eta$) 이동. 비등방성(타원형) 함수에서 비효율적인 지그재그 경로. |
| **Momentum** | 1. $v \leftarrow \alpha v - \eta \frac{\partial L}{\partial W}$<br><br>2. $W \leftarrow W + v$ | 관성($\alpha$)을 이용해 이전 이동 방향을 유지. 지그재그 움직임이 완화되어 수렴 속도 개선. |
| **AdaGrad** | 1. $h \leftarrow h + \frac{\partial L}{\partial W} \odot \frac{\partial L}{\partial W}$<br><br>2. $W \leftarrow W - \eta \frac{1}{\sqrt{h} + \epsilon} \frac{\partial L}{\partial W}$ | 개별 매개변수에 적응적인 학습률 적용. 많이 변화한(기울기 큰) 매개변수의 학습률을 감소. |
| **Adam** | Momentum (방향)과 AdaGrad (적응적 학습률)의 장점을 결합. | **1**차 모멘트($\beta_1$)와 **2**차 모멘트($\beta_2$)를 보정하여 최적화. 가장 널리 쓰임. |

## **2.** 가중치 초깃값 **(Weight Initialization)**과 분포의 수학

가중치 초깃값의 표준편차 $\sigma$는 은닉층의 활성화값 분포에 큰 영향을 미치며, 학습 초기 기울기 소실**(Gradient Vanishing)** 문제를 방지해야 합니다.

| 초깃값 | 활성화 함수 | 표준편차 **σ (**수식**)** | 원리 |
|---|---|---|---|
| **Xavier** | Sigmoid, Tanh (선형) | $\frac{1}{\sqrt{n}}$ | 입력 노드 수 $n$에 대해, 순전파와 역전파 시 |

| | | | |
|---|---|---|---|
| | | | 활성화값의 분산이 유지되도록 설계. |
| **He** | ReLU (비선형) | $\sqrt{\frac{2}{n}}$ | ReLU는 음수 영역을 0으로 만들어 정보의 절반을 소실시키므로, Xavier보다 2배 더 넓은 분포를 갖도록 설계. |

## **3.** 배치 정규화 **(Batch Normalization)** 알고리즘

배치 정규화는 학습 중 **내부 공변량 변화(Internal Covariate Shift)**를 줄이는 목적으로 사용됩니다.

### **3.1.** 정규화 단계 **(Normalization)**

미니배치 $B = {x_1, x_2, \dots, x_m}$에 대해 평균 $\mu_B$와 분산 $\sigma_B^2$을 계산하여 정규화합니다.

$$\mu_B = \frac{1}{m} \sum_{i=1}^m x_i \quad \text{, } \quad \sigma_B^2 = \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

### **3.2.** 스케일 및 시프트 단계 **(Scale and Shift)**

정규화된 값 $\hat{x}_i$를 학습 가능한 매개변수 $\gamma$와 $\beta$를 사용하여 선형 변환합니다.

$$y_i = \gamma \hat{x}_i + \beta$$

- 효과**:** $\gamma=1, \beta=0$이면 표준 정규 분포, $\gamma = \sqrt{\sigma_B^2}, \beta = \mu_B$이면 정규화 전의 분포로 돌아갑니다. 이를 통해 모델은 필요에 따라 분포를 조정할 수 있는 표현력을 갖게 됩니다.

## **4.** 오버피팅 방지 **(Regularization)**

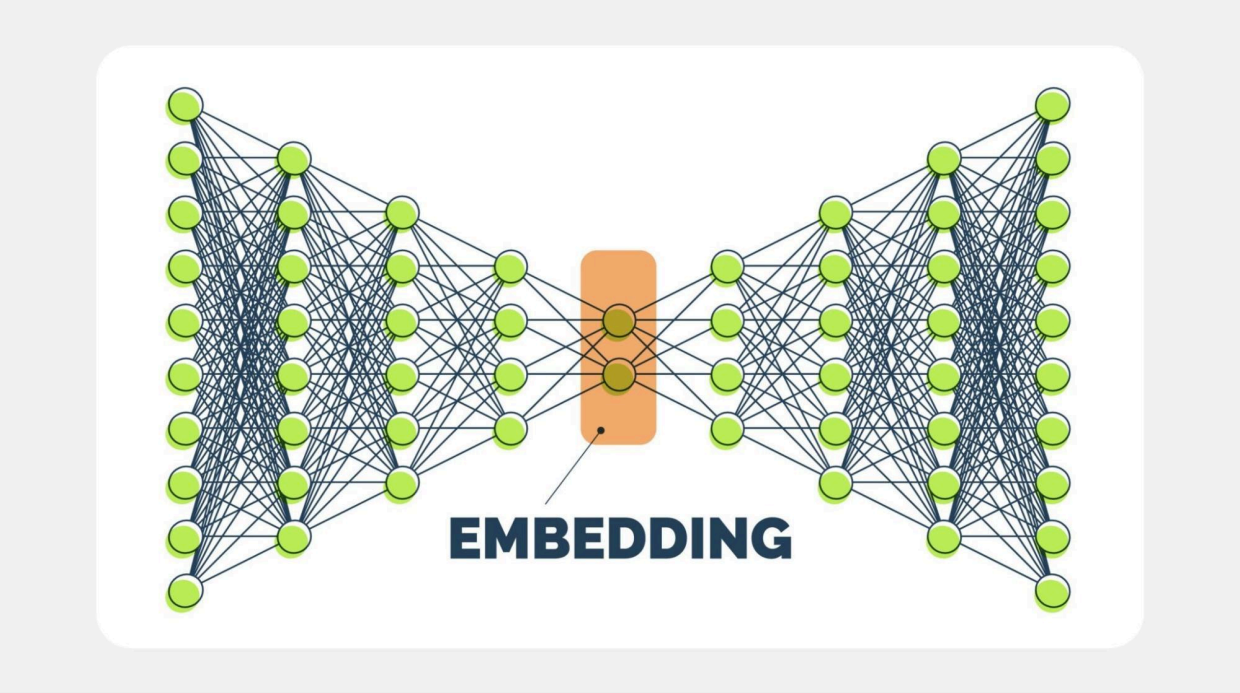
### **4.1.** 가중치 감소 **(Weight Decay, L2** 정규화**)**

손실 함수 $E$에 가중치 매개변수 $W$의 제곱 합을 추가하여 큰 가중치 값에 페널티를

부과합니다.

- 손실 함수:
  $$E' = E + \frac{1}{2} \lambda W^2$$
  (단, $\lambda$는 규제의 강도를 조절하는 하이퍼파라미터)
- 갱신 수식 **(SGD):**
  $$\frac{\partial E'}{\partial W} = \frac{\partial E}{\partial W} + \lambda W$$
  $$W \leftarrow W - \eta \left( \frac{\partial E}{\partial W} + \lambda W \right)$$
- 원리: 매 갱신마다 $W$를 $\eta \lambda W$만큼 줄여 오버피팅을 억제합니다.

### 4.2. 드롭아웃 **(Dropout)**




Getty Images

훈련 시에만 확률 $p$로 뉴런을 무작위로 비활성화하여(0으로 설정) 각 뉴런이 다른 뉴런에 덜 의존적이게 만듭니다. 이는 앙상블 학습과 유사한 효과를 냅니다.

- 훈련 시 순전파:
  $$\text{Mask} \sim \text{Bernoulli}(p)$$
  $$\text{출력} = \text{입력} \odot \text{Mask}$$
- 시험 시 순전파: 모든 뉴런을 사용하되, 훈련 시 비활성화했던 비율만큼 스케일을 조정합니다.
  $$\text{출력} = \text{입력} \cdot (1 - p)$$
  (이 과정을 인버티드 드롭아웃이라고 하며, 보통 훈련 시 $\text{출력} = \text{입력} \odot

\text{Mask} / (1-p)$로 나누고, 시험 시에는 그대로 사용하는 방식이 더 흔함.)