QE.
Artificial Neural Networks and Deep Learning

학번: 20256145
이름: 윤원태

1. Gradient Descent (8점)

다변량 함수 $g(x, y) = x^2 + y^2 + xy - 4x - 5y + 11$ 에 대해 아래의 설명을 잘 읽고 단계별로 해를 구하라. 계산과정에서 나오는 소수에 대해서는 소수점 이하 둘째 자리에서 반올림하여 소수점 첫째 자리까지만 표기하라.

1-1. 함수 g의 편도함수 $\frac{\partial g}{\partial x}$, $\frac{\partial g}{\partial y}$를 각각 구하라.

답:
먼저 x에 대해 편미분을 수행합니다.

$$\begin{aligned}\frac{\partial g}{\partial x} &= \frac{\partial}{\partial x}(x^2 + y^2 + xy - 4x - 5y + 11) \\ &= \frac{\partial}{\partial x}(x^2) + \frac{\partial}{\partial x}(y^2) + \frac{\partial}{\partial x}(xy) - \frac{\partial}{\partial x}(4x) - \frac{\partial}{\partial x}(5y) + \frac{\partial}{\partial x}(11) \\ &= 2x + 0 + y - 4 + 0 + 0 \\ &= 2x + y - 4\end{aligned}$$

다음으로 y에 대해 편미분을 수행합니다.

$$\begin{aligned}\frac{\partial g}{\partial y} &= \frac{\partial}{\partial y}(x^2 + y^2 + xy - 4x - 5y + 11) \\ &= \frac{\partial}{\partial y}(x^2) + \frac{\partial}{\partial y}(y^2) + \frac{\partial}{\partial y}(xy) - \frac{\partial}{\partial y}(4x) - \frac{\partial}{\partial y}(5y) + \frac{\partial}{\partial y}(11) \\ &= 0 + 2y + x - 0 - 5 + 0 \\ &= 2y + x - 5\end{aligned}$$

1-2. Gradient Descent의 갱신 규칙(업데이트 룰)을 아래와 같이 쓴다. 이를 x와 y에 적용한 결과를 각각 제시하라.

$$x_{k+1} = x_k - \alpha \frac{\partial g}{\partial x}$$

$$y_{k+1} = y_k - \alpha \frac{\partial g}{\partial y}$$

답:
위에서 구한 편도함수를 대입하면:

$x_{k+1} = x_k - \alpha(2x_k + y_k - 4)$

$y_{k+1} = y_k - \alpha(2y_k + x_k - 5)$

1-3. (x, y)의 초기값 $(x_0, y_0) = (4, 4)$로 설정할 때 (x_1, y_1)을 구하라.

답:
학습률 α를 그대로 두고 일반식으로 계산합니다.

$x_1 = x_0 - \alpha(2x_0 + y_0 - 4)$
$y_1 = y_0 - \alpha(2y_0 + x_0 - 5)$

초기값 $(x_0, y_0) = (4, 4)$를 대입하면,

$x_1 = 4 - \alpha(2 \times 4 + 4 - 4) = 4 - 8\alpha$
$y_1 = 4 - \alpha(2 \times 4 + 4 - 5) = 4 - 7\alpha$

따라서 $(x_1, y_1) = (4 - 8\alpha,\ 4 - 7\alpha)$입니다.

1-4. 두 기울기의 크기가 모두 0.2 이하가 될 때까지 (x, y)의 경로를 수행하고 최종 근사값을 제시하라.

답:
두 기울기의 크기가 모두 0.2 이하가 될 때까지 gradient descent를 수행합니다. 학습률 $\alpha = 0.1$로 설정하겠습니다.

초기값: $(x_0, y_0) = (4.00, 4.00)$

1단계:

- $\frac{\partial g}{\partial x}(4.00, 4.00) = 2 \times 4.00 + 4.00 - 4 = 8.00$
- $\frac{\partial g}{\partial y}(4.00, 4.00) = 2 \times 4.00 + 4.00 - 5 = 7.00$
- $x_1 = 4.00 - 0.1 \times 8.00 = 3.20$
- $y_1 = 4.00 - 0.1 \times 7.00 = 3.30$

2단계:

- $\frac{\partial g}{\partial x}(3.20, 3.30) = 2 \times 3.20 + 3.30 - 4 = 6.40 + 3.30 - 4 = 5.70$
- $\frac{\partial g}{\partial y}(3.20, 3.30) = 2 \times 3.30 + 3.20 - 5 = 6.60 + 3.20 - 5 = 4.80$
- $x_2 = 3.20 - 0.1 \times 5.70 = 3.20 - 0.57 = 2.63$
- $y_2 = 3.30 - 0.1 \times 4.80 = 3.30 - 0.48 = 2.82$

3단계:

- $\frac{\partial g}{\partial x}(2.63, 2.82) = 2 \times 2.63 + 2.82 - 4 = 5.26 + 2.82 - 4 = 4.08$
- $\frac{\partial g}{\partial y}(2.63, 2.82) = 2 \times 2.82 + 2.63 - 5 = 5.64 + 2.63 - 5 = 3.27$
- $x_3 = 2.63 - 0.1 \times 4.08 = 2.63 - 0.41 = 2.22$
- $y_3 = 2.82 - 0.1 \times 3.27 = 2.82 - 0.33 = 2.49$

4단계:

- $\frac{\partial g}{\partial x}(2.22, 2.49) = 2 \times 2.22 + 2.49 - 4 = 4.44 + 2.49 - 4 = 2.93$
- $\frac{\partial g}{\partial y}(2.22, 2.49) = 2 \times 2.49 + 2.22 - 5 = 4.98 + 2.22 - 5 = 2.20$
- $x_4 = 2.22 - 0.1 \times 2.93 = 2.22 - 0.29 = 1.93$
- $y_4 = 2.49 - 0.1 \times 2.20 = 2.49 - 0.22 = 2.27$

5단계:

- $\frac{\partial g}{\partial x}(1.93, 2.27) = 2 \times 1.93 + 2.27 - 4 = 3.86 + 2.27 - 4 = 2.13$
- $\frac{\partial g}{\partial y}(1.93, 2.27) = 2 \times 2.27 + 1.93 - 5 = 4.54 + 1.93 - 5 = 1.47$
- $x_5 = 1.93 - 0.1 \times 2.13 = 1.93 - 0.21 = 1.72$
- $y_5 = 2.27 - 0.1 \times 1.47 = 2.27 - 0.15 = 2.12$

6단계:

- $\frac{\partial g}{\partial x}(1.72, 2.12) = 2 \times 1.72 + 2.12 - 4 = 3.44 + 2.12 - 4 = 1.56$
- $\frac{\partial g}{\partial y}(1.72, 2.12) = 2 \times 2.12 + 1.72 - 5 = 4.24 + 1.72 - 5 = 0.96$
- $x_6 = 1.72 - 0.1 \times 1.56 = 1.72 - 0.16 = 1.56$
- $y_6 = 2.12 - 0.1 \times 0.96 = 2.12 - 0.10 = 2.02$

7단계:

- $\frac{\partial g}{\partial x}(1.56, 2.02) = 2 \times 1.56 + 2.02 - 4 = 3.12 + 2.02 - 4 = 1.14$
- $\frac{\partial g}{\partial y}(1.56, 2.02) = 2 \times 2.02 + 1.56 - 5 = 4.04 + 1.56 - 5 = 0.60$
- $x_7 = 1.56 - 0.1 \times 1.14 = 1.56 - 0.11 = 1.45$
- $y_7 = 2.02 - 0.1 \times 0.60 = 2.02 - 0.06 = 1.96$

8단계:

- $\frac{\partial g}{\partial x}(1.45, 1.96) = 2 \times 1.45 + 1.96 - 4 = 2.90 + 1.96 - 4 = 0.86$
- $\frac{\partial g}{\partial y}(1.45, 1.96) = 2 \times 1.96 + 1.45 - 5 = 3.92 + 1.45 - 5 = 0.37$
- $x_8 = 1.45 - 0.1 \times 0.86 = 1.45 - 0.09 = 1.36$
- $y_8 = 1.96 - 0.1 \times 0.37 = 1.96 - 0.04 = 1.92$

9단계:

- $\frac{\partial g}{\partial x}(1.36, 1.92) = 2 \times 1.36 + 1.92 - 4 = 2.72 + 1.92 - 4 = 0.64$
- $\frac{\partial g}{\partial y}(1.36, 1.92) = 2 \times 1.92 + 1.36 - 5 = 3.84 + 1.36 - 5 = 0.20$
- $x_9 = 1.36 - 0.1 \times 0.64 = 1.36 - 0.06 = 1.30$
- $y_9 = 1.92 - 0.1 \times 0.20 = 1.92 - 0.02 = 1.90$

10단계:

- $\frac{\partial g}{\partial x}(1.30, 1.90) = 2 \times 1.30 + 1.90 - 4 = 2.60 + 1.90 - 4 = 0.50$
- $\frac{\partial g}{\partial y}(1.30, 1.90) = 2 \times 1.90 + 1.30 - 5 = 3.80 + 1.30 - 5 = 0.10$
- $x_{10} = 1.30 - 0.1 \times 0.50 = 1.30 - 0.05 = 1.25$
- $y_{10} = 1.90 - 0.1 \times 0.10 = 1.90 - 0.01 = 1.89$

11단계:

- $\frac{\partial g}{\partial x}(1.25, 1.89) = 2 \times 1.25 + 1.89 - 4 = 2.50 + 1.89 - 4 = 0.39$
- $\frac{\partial g}{\partial y}(1.25, 1.89) = 2 \times 1.89 + 1.25 - 5 = 3.78 + 1.25 - 5 = 0.03$
- $x_{11} = 1.25 - 0.1 \times 0.39 = 1.25 - 0.04 = 1.21$
- $y_{11} = 1.89 - 0.1 \times 0.03 = 1.89 - 0.00 = 1.89$

12단계:

- $\frac{\partial g}{\partial x}(1.21, 1.89) = 2 \times 1.21 + 1.89 - 4 = 2.42 + 1.89 - 4 = 0.31$
- $\frac{\partial g}{\partial y}(1.21, 1.89) = 2 \times 1.89 + 1.21 - 5 = 3.78 + 1.21 - 5 = -0.01$
- $x_{12} = 1.21 - 0.1 \times 0.31 = 1.21 - 0.03 = 1.18$
- $y_{12} = 1.89 - 0.1 \times (-0.01) = 1.89 + 0.00 = 1.89$

13단계:

- $\frac{\partial g}{\partial x}(1.18, 1.89) = 2 \times 1.18 + 1.89 - 4 = 2.36 + 1.89 - 4 = 0.25$
- $\frac{\partial g}{\partial y}(1.18, 1.89) = 2 \times 1.89 + 1.18 - 5 = 3.78 + 1.18 - 5 = -0.04$
- $x_{13} = 1.18 - 0.1 \times 0.25 = 1.18 - 0.03 = 1.15$
- $y_{13} = 1.89 - 0.1 \times (-0.04) = 1.89 + 0.00 = 1.89$

14단계:

- $\frac{\partial g}{\partial x}(1.15, 1.89) = 2 \times 1.15 + 1.89 - 4 = 2.30 + 1.89 - 4 = 0.19$
- $\frac{\partial g}{\partial y}(1.15, 1.89) = 2 \times 1.89 + 1.15 - 5 = 3.78 + 1.15 - 5 = -0.07$
- $x_{14} = 1.15 - 0.1 \times 0.19 = 1.15 - 0.02 = 1.13$
- $y_{14} = 1.89 - 0.1 \times (-0.07) = 1.89 + 0.01 = 1.90$

이제 $\frac{\partial g}{\partial x} = 0.19$, $\frac{\partial g}{\partial y} = -0.07$로 두 기울기의 크기가 모두 0.2 이하가 되었습니다.

따라서 최종 근사값은 $(x, y) = (1.1, 1.9)$입니다.

2. Least-Square Method (4)

Least-Square Method(최소 제곱법)을 이용하여 주어진 단일 실수 값 특징으로 이루어진 데이터 세트에 가장 잘 맞는 가설 $h_w(x) = w_0 + w_1 x$ 을 찾고자 한다. 아래 물음에 답하라.

2-1. 단일 example (x, y)에 대한 제곱 손실은 무엇인가?

답:

$$L((x, y), \mathbf{w}) = (h_w(x) - y)^2 = (w_0 + w_1 x - y)^2$$

2-2. 제곱 손실을 최소화하기 위해 경사 하강법을 사용한다면, w_0와 w_1에 대한 갱신 규칙(Update Rule)은 어떻게 표현되는가?

답: 학습률(learning rate)을 η라고 할 때,

$$w_0 \leftarrow w_0 - \eta \frac{\partial L}{\partial w_0} = w_0 - \eta \cdot 2(w_0 + w_1 x - y)$$
$$w_1 \leftarrow w_1 - \eta \frac{\partial L}{\partial w_1} = w_1 - \eta \cdot 2(w_0 + w_1 x - y)x$$

3. Logistic Regression (4)

우측 그림에 나타난 단순 데이터에 대해 로지스틱 회귀 모델 $h_{\mathbf{w}}(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}\cdot\mathbf{x}}}$를 유도하려고 한다. 여기서 ● 와 ○는 각각 positive example과 negative example을 나타낸다. 이 두 차원 데이터에 대해, $\mathbf{w} = (w_0, w_1, w_2)$이고 $\mathbf{x} = (x_0, x_1, x_2)$이며 $x_0 = 1$이다. 또한, 우도(Likelihood) 함수 P는 아래와 같이 정의된다.

$$P(d|h_{\mathbf{w}}) = \prod_j P(d_j|h_{\mathbf{w}}) = \prod_j \hat{p}(\mathbf{x}_j)^{y_j} \left(1 - \hat{p}(\mathbf{x}_j)\right)^{1-y_j}$$

3-1. 이 모델에 대한 두 가지 파라미터 후보 $\mathbf{w}_1 = (0, -1, 1)$과 $\mathbf{w}_2 = (-3, -1, 3)$에 대해, 우도를 계산하여 비교하여 더 나은 것을 결정하라.

답:

- Positive 예제(●): $(1, 2)$, $(2, 3)$
- Negative 예제(○): $(3, 1)$, $(4, 2)$

$\mathbf{w}_1 = (0, -1, 1)$에 대한 우도 계산:

- $(1, 2)$: $\hat{p} = \frac{1}{1+e^{-(0\cdot1+(-1)\cdot1+1\cdot2)}} = \frac{1}{1+e^{-1}} = 0.731$
- $(2, 3)$: $\hat{p} = \frac{1}{1+e^{-(0\cdot1+(-1)\cdot2+1\cdot3)}} = \frac{1}{1+e^{-1}} = 0.731$
- $(3, 1)$: $\hat{p} = \frac{1}{1+e^{-(0\cdot1+(-1)\cdot3+1\cdot1)}} = \frac{1}{1+e^{-(-2)}} = \frac{1}{1+e^{2}} = 0.119$
- $(4, 2)$: $\hat{p} = \frac{1}{1+e^{-(0\cdot1+(-1)\cdot4+1\cdot2)}} = \frac{1}{1+e^{-(-2)}} = \frac{1}{1+e^{2}} = 0.119$

$P(d|h_{\mathbf{w}_1}) = 0.731^1 \times 0.731^1 \times (1-0.119)^1 \times (1-0.119)^1 = 0.731 \times 0.731 \times 0.881 \times 0.881 \approx 0.421$

$\mathbf{w}_2 = (-3, -1, 3)$에 대한 우도 계산:

- $(1, 2)$: $\hat{p} = \frac{1}{1+e^{-(-3\cdot1+(-1)\cdot1+3\cdot2)}} = \frac{1}{1+e^{-2}} = 0.881$
- $(2, 3)$: $\hat{p} = \frac{1}{1+e^{-(-3\cdot1+(-1)\cdot2+3\cdot3)}} = \frac{1}{1+e^{-4}} = 0.982$
- $(3, 1)$: $\hat{p} = \frac{1}{1+e^{-(-3\cdot1+(-1)\cdot3+3\cdot1)}} = \frac{1}{1+e^{-(-5)}} = \frac{1}{1+e^{5}} = 0.007$
- $(4, 2)$: $\hat{p} = \frac{1}{1+e^{-(-3\cdot1+(-1)\cdot4+3\cdot2)}} = \frac{1}{1+e^{-(-5)}} = \frac{1}{1+e^{5}} = 0.007$

$P(d|h_{\mathbf{w}_2}) = 0.881^1 \times 0.982^1 \times (1-0.007)^1 \times (1-0.007)^1 = 0.881 \times 0.982 \times 0.993 \times 0.993 \approx 0.856$

$\mathbf{w}_2 = (-3, -1, 3)$가 더 좋은 파라미터입니다.

3-2. $\mathbf{w}_2 = (-3, -1, 3)$에 대한 결정 경계(Decision Boundary)를 그려라.

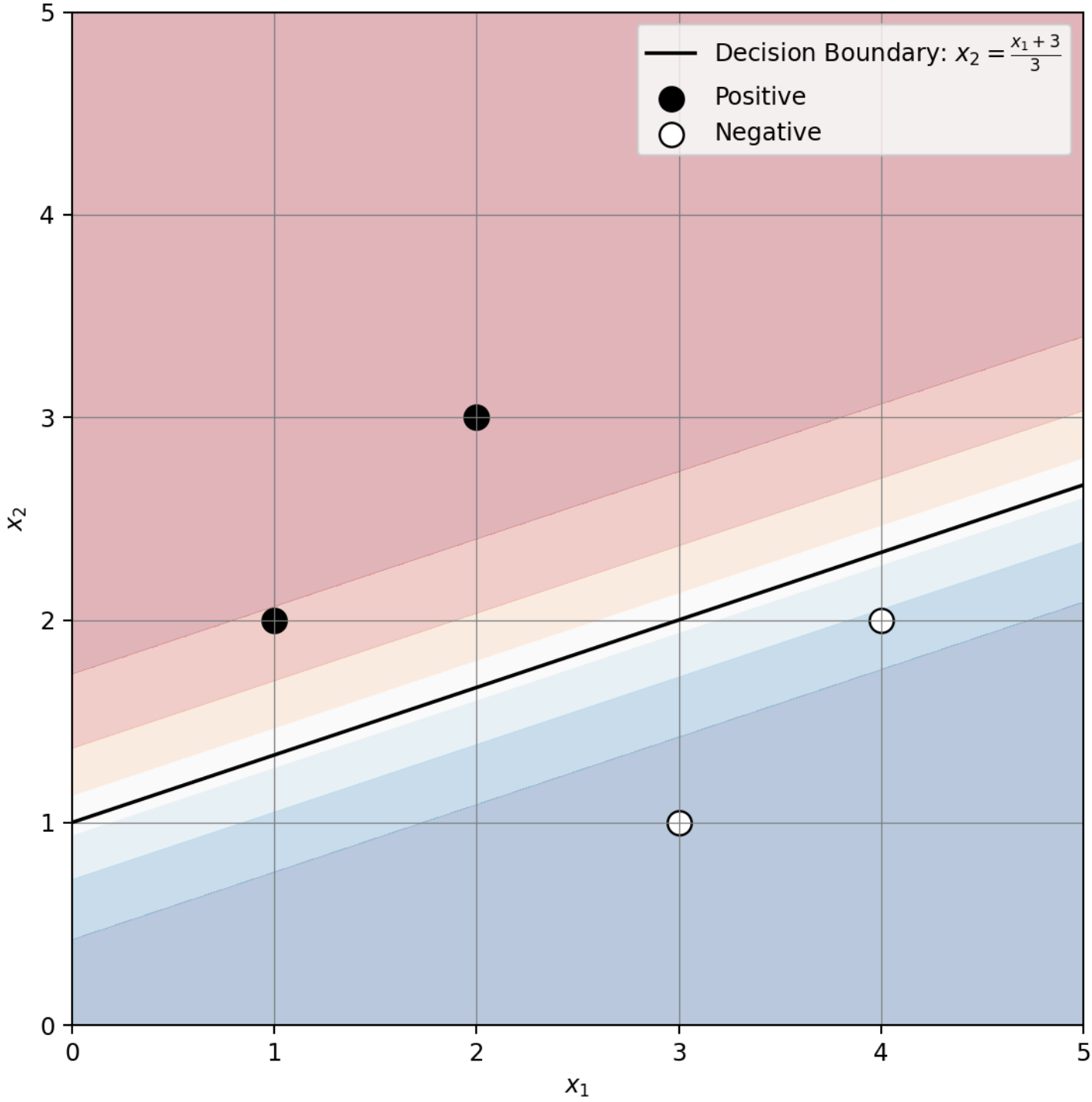
답:

$\hat{p}(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}\cdot\mathbf{x}}} = 0.5$

$1 + e^{-\mathbf{w}\cdot\mathbf{x}} = 2$
$e^{-\mathbf{w}\cdot\mathbf{x}} = 1$
$-\mathbf{w}\cdot\mathbf{x} = 0$
$\mathbf{w}\cdot\mathbf{x} = 0$

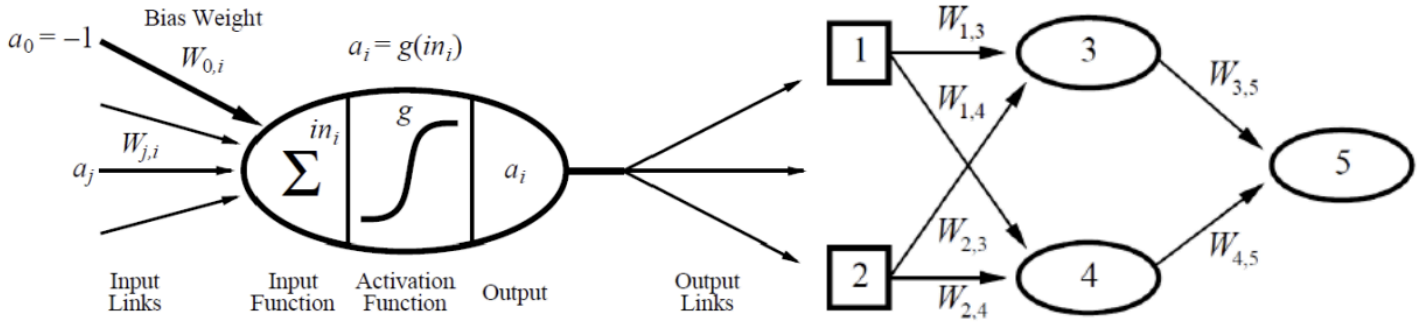
$\mathbf{w}_2 = (-3, -1, 3)$와 $\mathbf{x} = (1, x_1, x_2)$를 대입
$(-3)\cdot 1 + (-1)\cdot x_1 + 3\cdot x_2 = 0$
$-3 - x_1 + 3x_2 = 0$
$3x_2 - x_1 = 3$
$x_2 = \frac{x_1+3}{3}$

답 : $x_2 = \frac{x_1+3}{3}$


Decision Boundary: $x_2 = \frac{x_1 + 3}{3}$
Positive
Negative
x_1
x_2
0
1
2
3
4
5

4. Multi-Layer Perceptron: Feed-Forward Network (5)

아래 좌측 그림은 신경망의 기본 단위인 McCulloch-Pitts Unit을 나타낸다. 요약하면 어떠한 신경망 노드는 편향(bias)을 포함한 모든 입력과 이에 대한 가중치를 곱한 결과를 입력으로 하는 활성화함수(activation function) 값을 출력한다는 의미이다. 이를 참고하여 다음의 물음에 답하여라.



4-1. 우측 신경망의 최종 출력 a_5를 전개하라. 즉, a_5를 입력 a_1, a_2와 각 입력의 매개변수(parameter), 활성함수(activation)를 이용하여 나타내어라. 모든 신경망 노드에서의 활성함수는 g로 동일하다고 가정한다. (3)

답:

$$a_3 = g(w_{13}a_1 + w_{23}a_2)$$

$$a_4 = g(w_{14}a_1 + w_{24}a_2)$$

$$a_5 = g(w_{35}a_3 + w_{45}a_4)$$

따라서,

$$a_5 = g\left(w_{35} \cdot g(w_{13}a_1 + w_{23}a_2) + w_{45} \cdot g(w_{14}a_1 + w_{24}a_2)\right)$$

4-2. 우측의 신경망에는 은닉층에 대한 편향(bias)만 존재하며, 각각 b_1, b_2라 하면 4-1의 결과는 어떻게 바뀌는가?

답:

$$a_3 = g(w_{13}a_1 + w_{23}a_2 + b_1)$$

$$a_4 = g(w_{14}a_1 + w_{24}a_2 + b_2)$$

$$a_5 = g(w_{35}a_3 + w_{45}a_4)$$

따라서,

$$a_5 = g\left(w_{35} \cdot g(w_{13}a_1 + w_{23}a_2 + b_1) + w_{45} \cdot g(w_{14}a_1 + w_{24}a_2 + b_2)\right)$$

5. Deep Neural Networks: DNN, CNN, RNN (6)

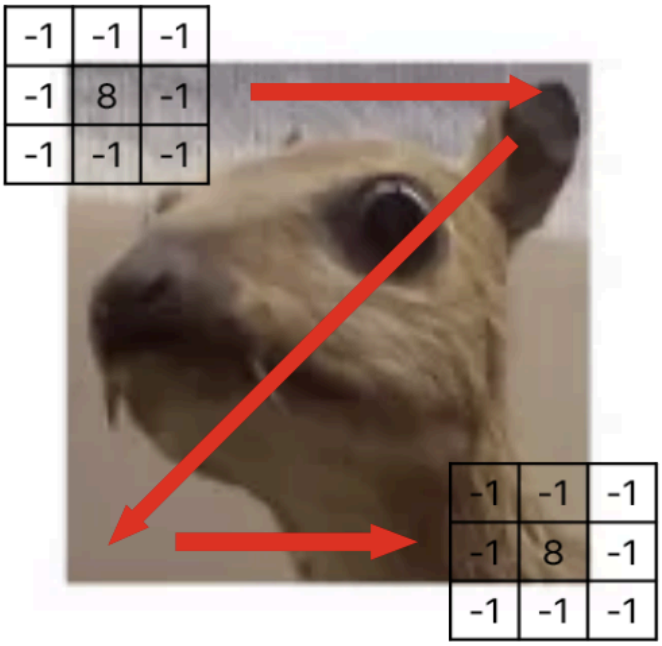
5-1. 다음과 같은 깊은 신경망이 있다고 가정하자. **입력층 노드 수: 6, 은닉층 노드 수: [5, 4], 출력층 노드 수: 3**. 이 신경망의 총 파라미터 수(가중치와 편향을 모두 포함)를 계산하라. 최종 계산 결과를 도출할 필요는 없으며 곱과 합으로 식을 표현하여도 좋다.

답:

- 입력층(6개 노드: $a_1, a_2, a_3, a_4, a_5, a_6$)에서 첫 번째 은닉층(5개 노드)으로 연결되는 가중치: $w_{ij}^{(1)}$ (i=1,2,...,6; j=1,2,...,5)
  - 가중치 수: $6 \times 5 = 30$개
  - 편향 수: $b_j^{(1)}$ (j=1,2,...,5) = 5개
- 첫 번째 은닉층(5개 노드)에서 두 번째 은닉층(4개 노드)으로 연결되는 가중치: $w_{jk}^{(2)}$ (j=1,2,...,5; k=1,2,...,4)
  - 가중치 수: $5 \times 4 = 20$개
  - 편향 수: $b_k^{(2)}$ (k=1,2,...,4) = 4개
- 두 번째 은닉층(4개 노드)에서 출력층(3개 노드)으로 연결되는 가중치: $w_{kl}^{(3)}$ (k=1,2,...,4; l=1,2,3)
  - 가중치 수: $4 \times 3 = 12$개
  - 편향 수: $b_l^{(3)}$ (l=1,2,3) = 3개

총 파라미터 수 = $(6 \times 5 + 5) + (5 \times 4 + 4) + (4 \times 3 + 3) = 30 + 5 + 20 + 4 + 12 + 3 = 74$개

5-2. 아래 그림은 컨볼루션 연산을 수행하여 특징 지도(Feature Map)를 얻는 과정에 관한 개략적 예시이다. 그림을 참고하여 다음의 물음에 답하라.



입력 이미지가 크기 64×64인 특징에 크기 5×5인 커널로 컨볼루션 연산을 적용한다. 이후, 풀링 과정을 수행하는데 가능한한 모든 영역의 특징 지도를 얻기 위해 적절한 크기의 패딩(padding)을 수행해야 한다. 그러나 한 구조의 간결성을 위해 보폭(stride)의 크기는 1로 제한한다. 이렇게 생성된 특징 지도의 크기는 얼마인가? 그림을 개략적으로 그리고 결과를 함께 보여라. 커널 연산과 풀링 연산이 되었다가 복원은 동일하다고 가정한다.

답:

컨볼루션 연산 후 특징 지도의 크기를 계산하기 위해 다음 공식을 사용합니다:

$$\text{출력크기} = \frac{(\text{입력크기} - \text{커널크기} + 2 \times \text{패딩})}{\text{스트라이드}} + 1$$

주어진 조건:

- 입력 이미지 크기: 64×64
- 커널 크기: 5×5
- 스트라이드: 1
- 패딩: 계산 필요

출력 크기가 입력 크기와 동일하게 유지되려면(가능한 한 모든 영역의 특징 지도를 얻기 위해):
$64 = \frac{(64 - 5 + 2p)}{1} + 1$

이 식을 풀면:
$64 = 64 - 5 + 2p + 1$
$64 = 60 + 2p$
$4 = 2p$
$p = 2$

따라서 패딩 크기는 2이고, 이를 적용하면:

- 입력 이미지(64×64)에 패딩(p=2)을 적용하면 68×68 크기가 됩니다.
- 이 이미지에 5×5 커널, 스트라이드 1로 컨볼루션을 적용하면 출력 크기는 64×64가 됩니다.

특징 지도의 최종 크기: 64×64

[개략도]
입력 이미지(64×64)
→ 패딩 적용(68×68)
→ 컨볼루션 연산(5×5 커널, stride=1)
→ 특징 지도(64×64)

5-3. RNN에서 시계열 시퀀스의 길이가 길어진다면 은닉층의 숫자가 1개일 경우라도 Gradient Vanishing 문제가 발생하는 이유는 무엇인가?

답:

RNN에서는 시간 단계마다 동일한 가중치 행렬이 반복적으로 곱해지는 구조로 인해 기울기 소실 문제가 발생합니다. 활성화 함수(tanh, sigmoid)의 미분값이 1보다 작아 여러 시간 단계를 거치며 기울기가 급격히 감소합니다. 은닉층이 하나라도 시간적으로 펼쳐진 구조에서는 시퀀스 길이만큼 깊이가 증가하는 효과가 있습니다.

이러한 문제로 인해 시퀀스의 초기 정보가 후반부에 영향을 미치지 못하고 장기 의존성 학습이 어려워집니다. 결과적으로 은닉층의 수와 관계없이 시퀀스 길이가 길어질수록 기울기 소실 문제가 심화되어 학습 성능이 저하됩니다. LSTM이나 GRU와 같은 게이트 메커니즘을 도입한 모델들은 이러한 문제를 완화하기 위해 개발되었습니다.

6. Additional Problem: DNN, CNN, RNN (8)

6-1. 경사 하강(Gradient Descent)법의 학습 속도를 개선하기 위한 대표적 방법에는 모멘텀(Momentum)과 적응형 학습률(Adaptive Learning Rate)이 있다. 이 중, 모멘텀 방식이 경사 하강법에 어떤 개념을 도입하여 최적화 과정을 개선하는지 설명하라. 또한, 이 방법이 왜 필요한지, 어떤 문제를 해결하려고 하는지에 대해 서술하라.

답:

모멘텀 방식은 물리학의 운동량 개념을 경사 하강법에 도입하여 최적화 과정을 개선하는 방법입니다. 이 방법은 이전 단계의 그래디언트 방향을 일정 비율로 현재 업데이트에 반영함으로써 학습 과정에 관성을 부여합니다. 이를 통해 학습 과정이 지속적으로 같은 방향으로 진행될 때 더 빠르게 이동하고, 방향이 자주 바뀌는 경우에는 진동을 감소시키는 효과를 얻을 수 있습니다.

모멘텀 방식은 특히 경사가 가파른 협곡이나 지역 최소값이 많은 복잡한 손실 함수 지형에서 필요합니다. 이 방법은 지역 최소값에 빠지는 문제를 완화하고, 안장점에서의 느린 수렴 문제를 해결하는 데 효과적입니다. 또한 그래디언트의 방향이 자주 변하는 경우에 발생하는 지그재그 현상을 줄여 더 빠르고 안정적인 수렴을 가능하게 합니다.

6-2. Gradient Vanishing 문제의 해결방안을 제시하라.

답:

Gradient Vanishing 문제 해결을 위한 첫 번째 방안은 활성화 함수의 변경으로, 시그모이드나 tanh 대신 ReLU와 같은 함수를 사용하여 그래디언트가 사라지는 현상을 방지할 수 있습니다. 배치 정규화를 적용하면 각 층의 입력 분포를 정규화하여 그래디언트 흐름을 안정화시키고 학습 속도를 향상시킬 수 있습니다. 또한 잔차 연결을 통해 그래디언트가 네트워크의 이전 층으로 직접 흐를 수 있는 지름길을 제공함으로써 그래디언트 소실 문제를 완화할 수 있습니다.

LSTM이나 GRU와 같은 게이트 메커니즘을 가진 순환 신경망 구조를 사용하면 장기 의존성을 효과적으로 학습하고 그래디언트 소실 문제를 해결할 수 있습니다. 가중치 초기화 방법으로 Xavier나 He 초기화를 사용하면 각 층의 활성화 값이 적절한 분포를 유지하도록 하여 그래디언트 소실을 방지할 수 있습니다. 마지막으로 그래디언트 클리핑을 적용하여 그래디언트 폭발 문제를 방지하고 안정적인 학습을 가능하게 할 수 있습니다.

6-3. CNN에서 풀링(Pooling)이 이미지를 처리하는데 어떠한 이점을 제공하는지 설명하라. 또한, 가능하다면 Pooling 과정에 있어 최대(Max)가 아닌 평균(Average)를 사용했을 경우 어떠한 차이가 생기는지에 대해서도 설명하라.

답:

풀링은 CNN에서 특성 맵의 공간적 크기를 줄이는 연산으로, 주요 목적은 연산량을 줄이고 과적합을 방지하며 위치 변화에 대한 불변성을 확보하는 것입니다. 또한 풀링은 네트워크의 수용 영역을 확장시켜, 보다 넓은 영역의 특징을 추출할 수 있게 합니다.

최대 풀링(Max Pooling): 주어진 영역에서 가장 큰 값을 선택함으로써, 가장 강한 특징을 강조하고 잡음에 강한 특성을 보입니다.

평균 풀링(Average Pooling): 영역 내 모든 값의 평균을 취하여, 배경 정보나 전체적인 분포를 더 잘 보존합니다.

일반적으로 Max Pooling은 객체의 존재 여부나 경계 등 명확한 특징이 중요한 작업(예: 객체 검출)에 적합하며, Average Pooling은 전체적인 질감이나 패턴 분포가 중요한 작업(예: 장면 분류)에서 사용될 수 있습니다.

최근에는 풀링 대신 Strided Convolution을 사용하는 경우도 늘고 있으며, 이는 네트워크의 파라미터와 학습 가능성을 늘리는 방향으로 발전하고 있습니다.

6-4. RNN 모델의 각 타임 스텝에서 입력 데이터가 은닉층을 통해 출력층으로 어떻게 흐르는지 설명하라. 또한, RNN의 "재귀적(recurrent)" 특성은 무엇인지, 그리고 이러한 특성이 시계열 데이터를 처리할 때 어떤 이점을 제공하는지 설명하라. 이 때, RNN의 신경망 모델을 반드시 이용하라.

답:

RNN 모델에서는 각 타임 스텝 t에서 입력 데이터 x_t가 은닉층에 전달되어 이전 타임 스텝의 은닉 상태 h_{t-1}와 결합된 후 활성화 함수를 통과하여 새로운 은닉 상태 h_t를 생성합니다. 이 은닉 상태 h_t는 출력층으로 전달되어 현재 타임 스텝의 예측값 y_t를 생성하는 동시에, 다음 타임 스텝의 계산을 위해 저장됩니다. 이 과정은

$$h_t = tanh(W_{hx}x_t + W_{hh}h_{t-1} + b_h) 와 y_t = W_{yh}h_t + b_y$$

와 같은 수식으로 표현되며, 여기서 W와 b는 각각 가중치와 편향을 나타냅니다.

RNN의 재귀적 특성은 은닉 상태가 이전 타임 스텝의 정보를 현재 계산에 반영하는 순환 연결 구조를 의미합니다. 이러한 특성은 시계열 데이터에서 시간적 의존성을 모델링할 수 있게 하여, 텍스트나 음성과 같은 순차적 데이터에서 문맥 정보를 유지하는 데 중요한 역할을 합니다. 또한 이 구조는 가변 길이의 입력 시퀀스를 처리할 수 있게 하여 자연어 처리, 음성 인식, 시계열 예측 등 다양한 응용 분야에서 유연성을 제공합니다.