

# 3~5주차

- Clustering (군집화)

- 개체들을 비슷한 것끼리 그룹을 나누는 것

1. K-means algorithm

- 주어진 데이터를 k개의 클러스터로 묶는 알고리즘
- 분류할 클러스터의 수를 미리 정함
- 제대로 작동하지 않는 경우
  - 군집의 크기가 다를 경우
  - 군집의 밀도가 다를 경우
  - 데이터 분포가 특이할 경우

2. Expectation-Maximization (EM)

- 기댓값 최대화 알고리즘 → 관측되지 않는 잠재변수에 의존
- 최대우도값을 갖는 매개변수를 찾는 반복 알고리즘
- E-step
  - 매개변수 값에 관한 잠재변수  $z$ 값 추정
- M-step
  - E-step에서 얻은 잠재변수 값을 이용하여 매개변수  $\theta$ 값을 다시 추정

3. DBSCAN

- 밀도 기반 데이터 클러스터링 알고리즘
  - core point를 설정하고 지정거리 안에 설정한 데이터 개수만큼 있으면 조건 만족 → 군집화
  - 더 이상 연결된 밀도 높은 지역이 없을 때(core point가 없는 경우)까지 확장
  - Border point : 연결된 밀도 높은 지역들을 통해 군집을 확장하다가 해당 경계 포인트에서 확장을 멈추는 지점
  - Noise point : 어떤 군집에도 속하지 못한 데이터
-

- Neural Network
  - Perceptron : 다수의 신호를 입력으로 받아 하나의 값을 출력
  - Multi-layered Perceptron : 입력과 출력 사이에 하나 이상의 hidden layer 추가
    - 은닉층 : 특징 공간을 분류하는데 유리한 새로운 특징 공간으로 변환시키는 역할
    - 활성화함수 : 시그모이드 함수
    - 오류 역전파 알고리즘 사용
  - 머신러닝의 목표
    - 모든 샘플을 옳게 분류하는 함수를 찾는 것을 목표로 함

- Loss function
  - 지도 학습 시 알고리즘이 예측한 값과 실제 정답의 차이를 비교하는 함수
    - 평균 제곱 오차 (Mean Square Error, MSE)
      - $1/n \sum_{i=1}^n (y_i - t_i)^2$
      - 0에 가까울수록 오차 적음 → 좋은 모델
    - 경사하강법
      - 미분해서 미분계수가 0인 지점을 찾는 방법
      - $e = 1/2 ||y - o||^2 = 1/2 (y - \sigma(wx + b))^2$
      - 문제점
        - 적절한 학습률 설정
          - step size가 너무 큰 경우 : global minimum을 가로질러 건너뛰어 최소값에서 멀어질 수 있음
          - step size가 너무 작은 경우 : 학습시간이 오래 걸리고 local minimum에 수렴할 수 있음
      - local minimum
        - 알고리즘이 시작되는 파라미터 위치가 랜덤이므로 global minimum이 아닌 local minimum에 빠져 나오지 못하는 경우 발생
  - 시그모이드 함수의 미분최대치는 0.3으로 그 범위에 들어가지 않으면 전체 곱의 기울기가 소멸된다 → 기울기가 0이 되어 학습이 멈춤 ⇒ ReLU 함수 사용

## ■ 교차 엔트로피

- 이진 분류
- Y가 0과 1일 때의 P(정답레이블)와 Q(출력값)의 확률분포
- $H(P, Q) = - \sum_{y \in \{0,1\}} P(y) \log_2 Q(y)$
- 목적함수 :  $e = -(y \log_2 o + (1 - y) \log_2 (1 - o))$ 
  - $e = \sigma(Z)$ ,  $Z = wx + b$
- $H(P, Q) \geq H(P)$  (Q와 P가 비슷할수록 H(P,Q)는 H(P)와 가까워짐)

## ■ Softmax

- $$o_j = \frac{e^{s_j}}{\sum_{i=1,c} e^{s_j}}$$

## ■ Log Likelihood

- $e = -\log_2 o_y$

---

## • 딥러닝의 최적화 기법

### 1. 데이터 전처리

- 정규화
  - 데이터 값들을 0~1 사이의 공통 간격으로 재배치
  - $MinMax = \frac{data - data.min}{data.max - data.min}$
- 표준화
  - 데이터 값들을 공통된 근거나 기준인 척도를 변경
  - $Zscore = \frac{data - data.mean}{data.std}$
- 명칭값 one-hot 코드로 변환
  - 값의 개수만큼 비트를 부여

### 2. 가중치 초기화

- 대칭적 가중치(가중치에 랜덤성 없음)에서 같은 값으로 갱신(error 가능성 높아짐)되어 두 노드가 같은 일을 하게 되는 중복성 발생 → 난수로 초기화 하여 대칭 파괴

### 3. Momentum

- 학습 방향이 바로 바뀌지 않고 일정한 방향을 유지하려는 성질

- Movement = Negative of Gradient + Momenutm
  - 기울기에 관성을 부과하여 작은 기울기는 쉽게 넘어갈 수 있도록 만들어 줌  
→ local minimum 탈출

#### 4. Adaptive Methods

- 학습률이 너무 작은 경우 : 학습에 많은 시간이 소요됨
- 학습률이 너무 큰 경우 : 오버슈팅에 따른 진자 현상 발생

#### 5. Epoch

- 훈련 데이터 셋에 포함된 모든 데이터들이 한 번씩 모델을 통과한 횟수
- Batch : data의 총 개수
- Batch size : 연산 한 번에 들어가는 데이터 크기
- mini Batch : 1 Batch size에 해당하는 데이터 셋
  - Batch size가 너무 큰 경우 : 한번에 처리해야 할 데이터의 양이 많아 학습 속도가 느려지고, 메모리 부족 문제가 발생할 수 있음
  - Batch size가 너무 작은 경우 : 적은 데이터를 대상으로 가중치를 업데이트 하고 이 업데이트가 자주 발생하므로 훈련이 불안정

#### 6. Batch Normalization

- Internal Covariate Shift 현상
  - 학습 과정에서 계층 별로 입력의 데이터 분포가 달라지는 현상
  - 현재 층에 전달되는 입력데이터의 분포가 현재 층이 학습했던 시점의 분포와 차이가 발생
- 학습 과정에서 각 배치 단위 별로 데이터가 다양한 분포를 가지더라도 각 배치 별로 평균과 분산을 이용해 정규화하는 과정

#### 7. ReLU 활성화함수

- x가 0보다 크면 미분값이 1이 되기 때문에 경사가 소실되지 않음

#### 8. Stochastic Pooling

- 최대값 또는 평균값 대신 확률에 따라 적절한 activation 선택

---

#### • Regulation principle

- 과대 적합을 피하기 위해 여러 규제 기법 사용

##### 1. Loss Function → Weight Decay

- 모델의 weight의 제곱합(L2 Norm)을 패널티 텀으로 주어 loss를 최소화
- $Loss(\theta; \mathbb{X}, \mathbb{Y}) = Loss(\theta; \mathbb{X}, \mathbb{Y}) + \frac{1}{2}\gamma\|\theta\|^2$
- 규제항 ( $\frac{1}{2}\gamma\|\theta\|^2$ )은 가중치를 작은 값으로 유지하므로 모델의 용량을 제한하는 역할

## 2. 조기 종료

- 일정 시간이 지나면 훈련 데이터를 단순히 암기하여 과대적합 현상이 나타남  
→ 모델이 과적합되기 전 훈련을 멈춤

## 3. 데이터 증대

- MNIST에 Affine 변환
- 모핑(Morphing)을 이용한 변형
- 자연영상 확대
- 잡음을 섞어 확대

## 4. Dropout

- 신경망 전체를 다 학습시키지 않고 일부 노드만 무작위로 골라 학습
- 학습하는 중간에 일정 비율로 노드들의 출력을 0으로 만들어 신경망의 출력을 계산