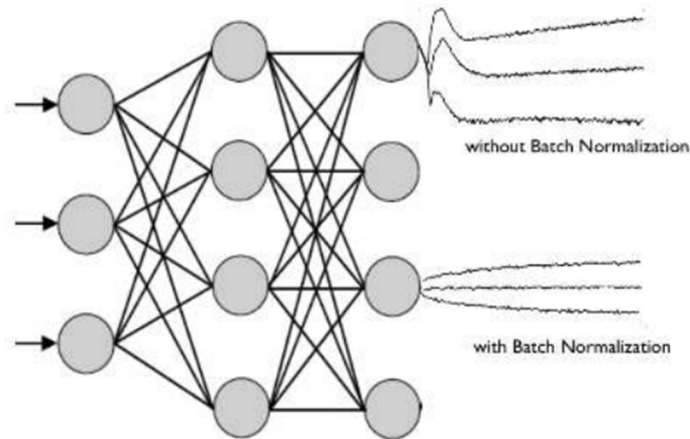


3차 일석이조 조별보고서	
작성일 : 2023년 11월 14일	작성자 : 탁성재
조 모임 일시 : 11월 14일	모임 장소 : 구글미트
참석자 : 이준용, 유정훈, 김동규, 이학빈, 탁성재	조원 : 이준용, 유정훈, 김동규, 이학빈, 탁성재
구분	내용
학습 범위와 내용	1. 11주차 온라인 강의 내용 2. 교재 5장 내용
논의 내용 (조별 모임 전에 조장이 지시)	5.1 목적함수 : 교차 엔트로피와 로그우도 5.1.1 평균제곱 오차를 다시 생각하기 5.1.2 교차 엔트로피 목적함수 5.1.3 softmax 활성화함수와 로그우도 목적함수 5.2 성능 향상을 위한 요령 5.2.1 데이터 전처리 5.2.2 가중치 초기화 5.2.3 모멘텀 5.2.4 적응적 학습률 5.2.5 활성화함수 5.2.6 배치 정규화
질문 내용 (모임 전 공지된 개별 학습 범위에서 이해된 것과 못한 것)	<p>Q1</p> <p>내부 공변량 변화와 배치 정규화 간의 관계에서 배치 정규화의 stablization은 정말로 공변 량 시프트 현상을 누그러뜨리는지 자세히 알고 싶다.</p> <p>A1</p> <p>배치 정규화의 경우, 깊은 레이어의 입력값의 변화(공변량 변화)를 줄이기 위해, 활성화함수 입력값, 혹은 활성화함수 출력값을 대상으로 평균 0, 분산 1이 되도록 정규화 시키는 것을 의미 한다.</p>



그러나 시그모이드 함수의 경우, 평균 0, 분산 1 을 갖는 입력값의 경우, $[-1, 1]$ 의 범위를 가지 는데, 이 범위의 sigmoid 함수 값은 0.5 를 중심으로 0.2~0.7 사이의 거의 직선인 선형 영역이 다. 따라서 활성화함수의 가장 중요한 역할인 비선형성을 잃어버릴 수 있어서,

이 점을 보완하기 위해 정규화된 값에 scale 팩터(γ)와 shift 팩터(β)을 도입하고 정규화된 값 에 곱하고 더해준다. 두 인자는 오차 역전파 과정에서 학습해 준다. 이를 통해 활성화함수로 들 어가는 값의 범위를 바꾸어줌에 의해 비선형성을 부여할 수 있다.

$$BN(x_i) = \gamma \left(\frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \right) + \beta$$

Q2

교차엔트로피와 softmax활성 함수에 대해 팀원들과 더 조사해봤습니다

A2

만일 확률분포 q가 확률분포 p와 동일하다면 q의 표본은 최적의 상태이다. 반면 q와 p가 상이하다면 표본 당 정보의 양이 새넨 엔트로피에 의한 최소 값(또는 최적 값)에 비해 증가한다. 다시 말해 p와 q 확률분포 사이의 교차 엔트로피가 p의 새넨 엔트로피보다 크다면, 확률분포 q가 p를 제대로 예측하지 못했음을 뜻한다. 또한 교차 엔트로피와 새넨 엔트로피의 차는 이러한 오차를 보정하기 위해 얼마만큼의 정보가 필요할지 알려주는 역할을 한다. 즉 새넨 엔트로피와 교차 엔트로피의 차를 p와 q 두 확률분포 사이의 거리 척도로 볼 수 있다. 이때 거리의 단위는 '표본 당 정보(information per sample)', 즉 q에서 표본을 분석하기 위해 추가적으로 필요한 정보의 양이다.

교차 엔트로피는 로지스틱 회귀와 같은 딥러닝 분류 모델 안에서 손실 함수로 활용된다. 어떤 사건에 대한 실제 확률분포 p를 모르는 상태에서, p를 예측하는 확률분포 q를 구한 후 p와 q의 교차 엔트로피를 구해 두 확률분포 사이의 차이를 좁히고 p의 근사치로서의 q를 구할 수 있기 때문이다. 딥러닝 모델의 훈련 데이터에서는 실제 확률분포 p를 알 수 있으므로 p와 q의 교차 엔트로피를 계산할 수 있다.

p와 q의 교차 엔트로피 $H(p, q)$ 는 다음과 같다. p_i 는 특정 확률에 대한 실제 값이고, q_i 는 현재 학습한 확률값이다. 연속형 확률분포에 대해서는 Σ 대신 \int 이 사용된다.

$$\begin{aligned} H(p, q) &= \sum_i p_i \log_2 \frac{1}{q_i} \\ &= - \sum_i p_i \log_2 q_i \end{aligned}$$

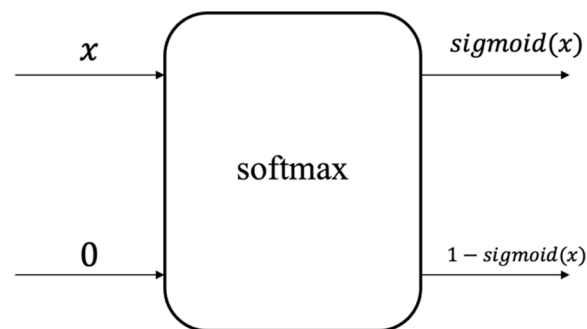
Softmax 함수는 N 가지 출력값을 갖는 함수로써 입력값을 N 가지 클래스 중 하나로 분류하는 Multi-class Classification 에 주로 사용된다. 수식은 아래와 같다.

$$\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

Softmax 함수의 특징은 다음과 같다.

- 출력값이 N 개
- 입력값을 각각 지수함수로 취하고, 이를 정규화(=총합을 1로 만듦)
- 정규화로 인해 각 출력값은 0~1 값을 가짐
- 모든 출력값의 합은 반드시 1
- N 가지 중 한 가지에 속할 확률 표현 가능(=Multi-class Classification)

앞서 다룬 Sigmoid 함수는 하나의 입력을 0으로 고정한 2-Class Softmax(0) 함수와 동일하다. 이를 쉽게 이해할 수 있도록 표현하면 아래의 그림 7과 같다.



<https://heytech.tistory.com/>

이를 활용하여 Simoid 함수와 Softmax 함수를 수식으로 표현하면 아래와 같다.

$$\text{sigmoid}(x) = \text{softmax}([x, 0])_{\{0\}} = \frac{e^x}{e^x + e^0} = \frac{1}{1+e^{-x}}$$

모든 출력값의 합은 1 이므로 sigmoid 함수의 다른 출력값은 $(1 - \text{sigmoid}(x))$ 와 같이 표현할 수 있습니다.

이를 통해 Softmax 는 2 개의 클래스를 구분하기 위해 2 개의 입력값을 받아왔었지만, 1 개의 입력값으로도 2 가지 클래스를 분류할 수 있다는 것을 알 수 있다.

Q3

데이터 전처리의 주요 단계에 대해서 조사해봤다.

A3

1. 결측치 처리 (Handling Missing Data)

데이터셋에 결측치(누락된 값)가 있는 경우, 이를 다루는 것이 중요하다. 결측치를 채우거나 해당 샘플을 제거하는 등의 방법을 사용하여 데이터를 완전하게 만들 수 있다.

2. 이상치 처리 (Outlier Handling)

이상치는 데이터셋에서 일반적인 패턴에서 벗어난 값으로, 모델의 성능을 왜곡할 수 있다. 이상치를 감지하고 수정하거나 제거하여 모델의 안정성을 향상시킨다.

3. 데이터 정규화/표준화 (Data Normalization/Standardization)

데이터를 일정한 범위나 분포로 조정하는 작업이다. 정규화는 데이터의 범위를 $[0, 1]$ 또는 $[-1, 1]$ 과 같은 특정 범위로 조정하고, 표준화는 데이터를 평균이 0 이고 표준 편차가 1 인 분포로 변환한다.

4. 특성 스케일링 (Feature Scaling)

데이터의 특성 간에 크기 차이가 클 경우, 모델의 성능에 영향을 줄 수 있다. 이를 해결하기 위해 특성 간의 스케일을 조정하여 동등한 중요도를 갖도록 한다.

5. 범주형 데이터 인코딩 (Categorical Data Encoding)

머신 러닝 모델은 주로 숫자로 된 데이터를 처리한다. 범주형 데이터(문자열 또는 카테고리 값)를 숫자로 변환하여 모델에 입력할 수 있도록 인코딩한다.

6. 데이터 분할 (Data Splitting)

데이터셋을 학습, 검증 및 테스트 세트로 나누는 작업이다. 일반적으로 학습에 사용되는 데이터와 모델 성능을 평가하기 위한 데이터를 분리한다.

7. 특성 추출 또는 생성 (Feature Extraction/Engineering)

데이터에서 유용한 특성을 추출하거나 새로운 특성을 만들어내는 작업으로, 모델의 성능을 향상시키는 데 도움을 준다.

8. 데이터 정제 (Data Cleaning)

데이터셋에서 노이즈, 중복된 값 또는 불필요한 정보를 제거하여 데이터의 질을 개선한다.

9. 시계열 데이터 처리 (Time Series Data Processing)

시간에 따라 기록된 데이터를 다룰 때, 시간 간격을 조정하거나 트렌드를 분석하는 등의 처리가 필요하다.

이러한 데이터 전처리 단계들은 데이터의 품질을 향상시켜 모델이 더 정확한 예측을 할 수 있도록 돕는다. 데이터 전처리는 머신 러닝 모델의 성능에 큰 영향을 미치므로 신중하게 진행되어야 한다.

Q4

모멘텀이란?

A4

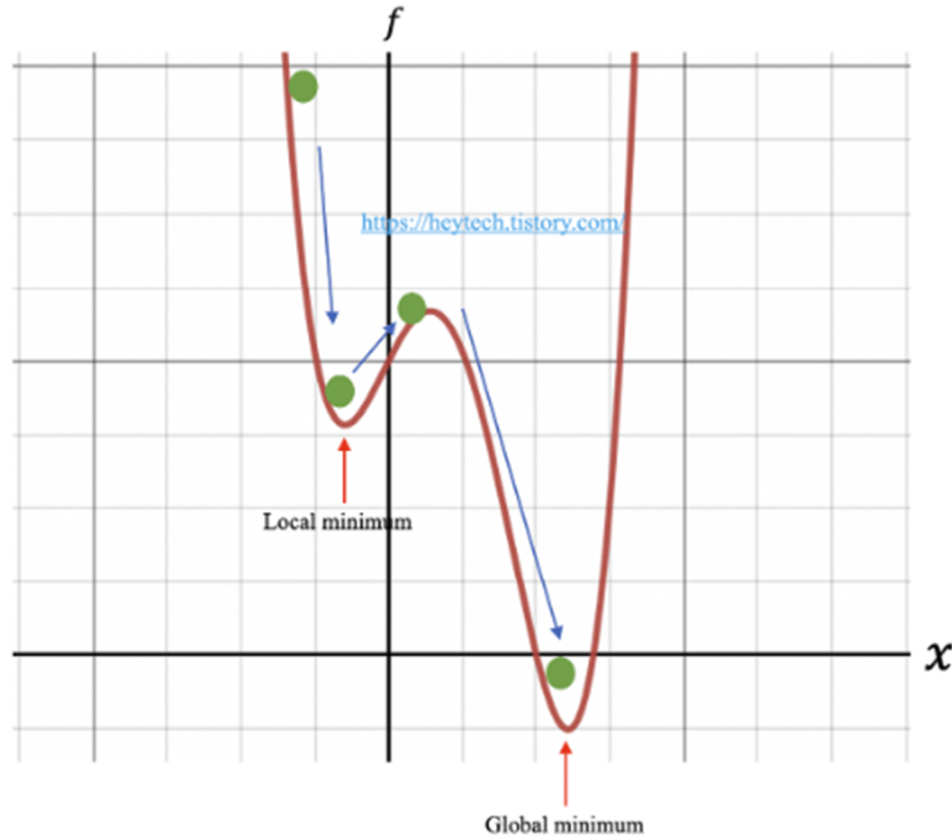
경사하강법과 같이 사용되는 학습률을 이용한 기계학습의 일종입니다.

경사 하강법

- 손실 함수를 통한 값의 비교
- 손실 함수 = 실제값 - 예측값
- 손실 함수값을 최소로 만들 수 있는 부분을 찾아감

모멘텀

- 어떠한 값에서 가지는 기울기를 해당 방향으로 지속적으로 이동시킴
- 어느 최소점을 지나더라도, 해당 방향으로 조금 더 움직임
- 덕분에 local maximum, local minimum 을 회피하기가 더 쉬움



모멘텀의 수식은

1. v_t (t 번째 시간단위에서의 x 의 이동벡터)
2. r (관성계수. 이전 벡터를 얼마나 적용받는가)
 1. 관성계수는 일반적으로 1 보다 작은 값을 받는다.
 2. 관성계수가 커질수록 예측 그래프는 부드러워진다.

3. n (학습률)

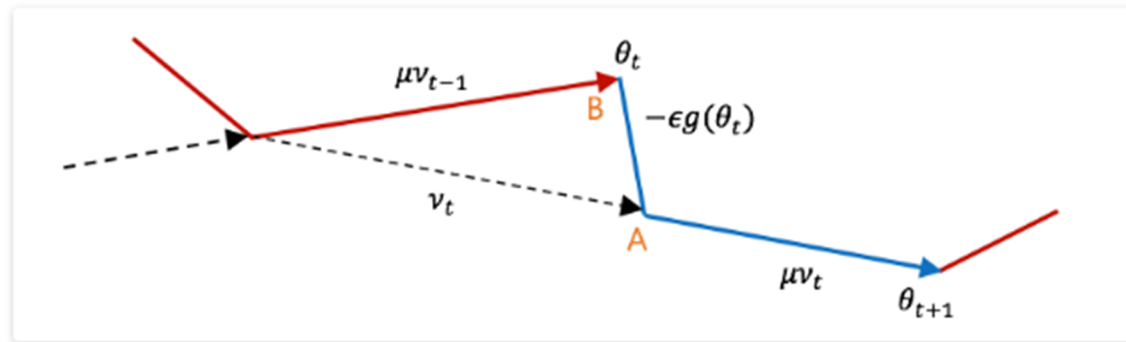
$$v_t = \gamma v_{t-1} + \eta \nabla f(x_{t-1})$$

이때 v_t 는 이전 이동벡터(v_{t-1})과 관성계수의 곱과 학습률과 $t-1$ 번째의 기울기(그래디언트)를 곱한 후 더한다.

네스테로프 모멘텀

v_t 의 예측에 v_{t-1} 와 v_t 의 기울기를 사용해서 계산을 줄임

따라서 해당 벡터는 v_{t-1} 과 v_t 의 기울기가 아닌, v_{t-1} 와 v_t 의 기울기를 따라 이동하게 되므로, 경로의 갯수가 줄어들게 되고, 속도도 빨라진다.



Q5

수업 중 언급된 가중치 초기화 사례들에 대한 조사

	<p>A5</p> <ol style="list-style-type: none"> 1. Saxe et al., 2014: 수직 초기화 (Orthogonal Initialization) <ul style="list-style-type: none"> ○ 방법: 가중치 벡터가 수직이 되도록 초기화합니다. ○ 특징: 가중치가 서로 직교하게끔 초기화되어, 특성 간의 의존성을 줄이고 모델의 안정성을 높입니다. ○ 사례: 특히 RNN 과 같이 시계열 데이터를 다루는 모델에서 효과적으로 사용됩니다. 2. Sussillo et al., 2014: Random Walk 초기화 <ul style="list-style-type: none"> ○ 방법: 랜덤 워크(random walk)를 통해 초기화를 수행합니다. ○ 특징: 가중치 초기화의 무작위성을 유지하면서도 모델의 수렴을 도모합니다. ○ 사례: 주로 복잡한 모델 구조에서 사용되며, 초기 무작위성이 중요한 경우에 활용됩니다. 3. Sutskever et al., 2013: 모멘텀 최적화를 이용한 초기화 <ul style="list-style-type: none"> ○ 방법: 가중치 초기화와 모멘텀 최적화를 동시에 사용하여 최적화를 수행합니다. ○ 특징: 모멘텀 최적화를 초기화와 함께 사용하여 빠른 수렴을 도모하고 안정성을 향상시킵니다. ○ 사례: 다양한 신경망 구조에서 모멘텀과 초기화의 상호 작용을 고려할 때 사용됩니다. 4. Mishkin et al., 2016: 노드 출력값 분포 일정화 <ul style="list-style-type: none"> ○ 방법: 가중치 분포가 아니라 노드의 출력값 분포를 일정하게 유지하도록 초기화합니다. ○ 특징: 훈련 중 가중치의 변화로 인한 노드 출력값의 분포가 일정하게 유지되어 학습의 안정성을 높입니다. ○ 사례: 딥 네트워크에서 불안정한 학습을 개선하고자 할 때 사용됩니다.
질문내용	<p>Q: 기존에 사용하던 Sigmoid function을 ReLU가 대체하게 된 이유 중 가장 큰 것이 Gradient Vanishing 문제의 해결이다. ReLU활성함수를 사용하게 되면 input data에서 음수 값을 입력받으면 항상 0으로 반환 되는것인데, 그렇다면 기울기(미분값)이 0이 되어 가중치가 업데이트가 안될 수 있을 것 같다. 결론으로 가중치가 업데이트가 안되면 결과로 아무것도 변하는것이 없는데 이부분은 어떻게 해결해야 하나요?</p>

학과	컴퓨터 전자시스템 공학	학번	201904458	이름	이준용
구분	내용				
학습 범위	기계학습 5장 딥러닝 최적화 5.1 목적함수: 교차 엔트로피와 로그우도 5.2 성능 향상을 위한 요령				
학습 내용	<p>딥 러닝의 최적화 과학과 공학에서 최적화 기계학습에 비해서 딥 러닝의 최적화가 훨씬 간단함 일반화 능력이 뛰어나야 함을 전제로 최적화 훈련집합과 테스트집합 간의 관계(대리자 관계)가 최적화를 어렵게 하고 있음 목적함수의 비볼록적인 성질, 고차원 특징 공간, 데이터의 희소성 등 긴 시간이 소요</p> <p>목적함수: 교차 엔트로피와 로그 우도 MSE 목적함수(평균 제곱 오차) -> 오차가 클수록 e의 값이 커지니 벌점으로 표기하기는 알맞음 한계: 벌점의 학습이 더디기 때문에 벌점의 상대적인 수치가 적절하지 않을 수 있음</p> <p>교차 엔트로피 목적함수 딥러닝은 적어도 분류 문제에서는 로스함수로 MSE대신 CE(Cross Entropy)를 주로 사용한다. (물론, 이를 확률분포로 생각한다면, 연속적인 값에 대한 손실함수는 가우시안 분포를 얻는다고 가정하는 MSE, 이산적인 값에 대해서는 멀티놀이 확률분포를 따를 것이라 가정하는 Cross entropy를 쓰는 것이 맞다.)</p> <p>Softmax 활성화함수 Max를 모방하여 만들어진 함수로, 최댓값은 더욱 활성화되고 작은 값은 억제되는 특징을 가진다</p> <p>데이터 전처리 데이터는 각 종류별로 데이터의 규모를 같은 선상에서 가늠하기 힘들다. 키와 몸무게를 기준으로 하였을 때 m와 kg를 비 교할 경우 kg가 100배 정도 더 큰 규모이기 때문에 첫 번째 특징의 가중치는 100배 정도 느리게 학습된다.</p> <p>1. 가중치 초기화 처음에 0 이 되면 문제발생. 다 균일한 값을 주면 backpropagation 시 두 노드가 같은 일을 하는 중복성 발생한다. 따라서 난수로 가중치를 초기화한다. 가우시언 분포 또는 균일 분포에서 난수를 추출한다.</p>				

2. 모멘텀

Gradient의 잡음 현상을 의미하며, 기계 학습은 훈련 집합을 이용하여 Gradient를 추정하므로 잡음이 생길 가능성이 높다. 모멘텀은 그래디언트 스무딩을 가하여 잡음을 줄인다. -> 수렴 속도 빨라진다.

모멘텀의 효과는 오버슈팅 현상을 누그러뜨린다.

3. 적응적 학습률

학습률 p 는 너무 크게 주어질 경우 Overshooting 현상으로 인한 진자 현상이 일어날 가능성이 높고 너무 작으면 수렴하는 속도가 매우 느려진다.

이럴 때 Gradient에 학습률 p 를 곱하여 모든 매개변수가 같은 크기의 학습률을 사용하게 유도하여 step이 진행됨에 따라 값의 진행을 조절한다.

- ✓ AdaGrad – Parameter 중 Element의 움직임이 많아질수록 학습률이 낮아지게 하여 element마다 학습률을 다르게 감소시키는 방법.
- ✓ RMSProp – 기울기를 단순 누적하는 것이 아니라 지수 가중 이동 평균을 사용하여 최신 기울기들이 AdaGrad에 비해 더 크게 반영하도록 하는 것.
- ✓ Adam – AdaGrad와 RMSProp을 융합한 형태로, 두 기법에서 AdaGrad와 RMSProp의 계수가 초기에 0으로 biased 되는 문제를 해결하기 위하여 고안된 방법
- ✓

4. 활성화함수

활성값 z 를 계산하여 나온 결과 r 를 적용하는 과정.

초기에는 선형의 활성화함수를 사용하였으나 비선형 문제를 해결하기 위해 퍼셉트론에서 계단식 함수를 사용하였다. 그 이후 tanh 형태에서 현재 쓰이고 있는 ReLU 형태로 진화하였다.

Tanh는 활성화값이 커지면 포화 상태가 되고 그래디언트는 0에 가까워진다. 매개변수 갱신이 매우 느린 요인이다. Relu 함수 살짝 negative 하게 하려면 Leaky Relu 사용한다.

5. 배치 정규화

공변량 시프트 현상

학습이 진행되면서 층 1의 매개변수가 바뀔에 따라 결과값이 따라 바뀔 -> 층 2에 입력되는 데이터의 분포가 변경됨 -> 층3....으로 갈수록 더 심각해진다
이는 학습을 방해하는 요인으로 작용

Gradient Vanishing / Exploding Problem이 발생하지 않도록 하면서 Learning Rate 값을 크게 설정해야 함. 이를 누그러뜨리기 위해 이전의 데이터 정규화를 모든 층에 적용하는 기법. 입력 데이터가 아닌 중간 결과 데이터에 적용하는 것이 더 유리하다

-> 다른 곳에 해도 성능 향상에는 도움이 됨.

	<p>훈련 집합보다는 미니 배치에 적용하는 것이 훨씬 유리하다.</p> <p>-> 어느 방향으로 업데이트 할 것인지 계산하는 쪽에 정규화를 하는 것이 더 성능에 도움이 된다.</p> <p>효과: Training 결과와 Test 결과 분포가 다를 경우 Training Data Set에 대한 결과가 Test Data Set의 결과에 맞춰지게 됨.</p> <ul style="list-style-type: none"> ● CNN에서는 노드 단위가 아니라 특징 맵 단위로 학습, 테스트를 적용 ● 가중치 초기화에 덜 민감함 ● 학습률을 크게 하여 수렴 속도가 향상 ● Sigmoid를 활성화함수로 사용하는 깊은 신경망도 학습이 이루어짐 ● Drop Out을 따로 적용하지 않아도 높은 성능이 이루어짐.
--	---

학과	철학과	학번	201802344	이름	유정훈
구분	내용				
학습 범위	동영상 11주차				
학습 내용	<p>1. 평균 제곱 오차</p> <p>(1) 평균 제곱 오차는 오차가 클수록 e 값이 크므로 벌점으로 훌륭</p> <p>(2) 단점 발견: 계수값이 커져 영향이 작아지며 더 많은 오류를 범하더라도 더 낮은 벌점을 받음. 이에 학습이 더딘 부정적 효과를 초래</p> <p>(3) 오류가 큰 상황에 더 낮은 벌점을 주더라도 그레디언트가 오류를 줄이는 방향을 제대로 알려주기에 최저점을 찾아가는 목적 자체에는 지장이 없음.</p> <p>2. 교차엔트로피</p> <p>(1) 만일 확률분포 q 가 확률분포 p 와 동일하다면 q 의 표본은 최적의 상태이다. 반면 q 와 p 가 상이하다면 표본 당 정보의 양이 새넌 엔트로피에 의한 최소 값(또는 최적 값)에 비해 증가한다. 다시 말해 p 와 q 확률분포 사이의 교차 엔트로피가 p 의 새넌 엔트로피보다 크다면, 확률분포 q 가 p 를 제대로 예측하지 못했음을 뜻한다. 또한 교차 엔트로피와 새넌 엔트로피의 차는 이러한 오차를 보정하기 위해 얼마만큼의 정보가 필요할지 알려주는 역할을 한다. 즉 새넌 엔트로피와 교차 엔트로피의 차를 p 와 q 두 확률분포 사이의 거리 척도로 볼 수 있다. 이때 거리의 단위는 '표본 당 정보(information per sample)', 즉 q 에서 표본을 분석하기 위해 추가적으로 필요한 정보의 양이다.</p> <p>교차 엔트로피는 로지스틱 회귀와 같은 딥러닝 분류 모델 안에서 손실 함수로 활용된다. 어떤 사건에 대한 실제 확률분포 p를 모르는 상태에서, p 를 예측하는 확률분포 q를 구한 후 p와 q의 교차 엔트로피를 구해 두 확률분포 사이의 차이를 좁히고 p의 근사치로서의 q를 구할 수 있기 때문이다. 딥러닝 모델의 훈련 데이터에서는 실제 확률분포 p를 알 수 있으므로 p와 q의 교차 엔트로피를 계산할 수 있다.</p> <p>p와 q의 교차 엔트로피 $H(p, q)$는 다음과 같다. p_i는 특정 확률에 대한 실제 값이고, q_i는 현재 학습한 확률값이다. 연속형 확률분포에 대해서는 Σ 대신 \int이 사용된다.</p> <p>3. softmax 활성화함수</p> <p>(1) Softmax 함수는 N 가지 출력값을 갖는 함수로써 입력값을 N 가지 클래스 중 하나로 분류하는 Multi-class Classification 에 주로 사용됩니다. 수식은 아래와 같습니다.</p> $\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$ <p>Softmax 함수의 특징은 다음과 같습니다.</p> <ul style="list-style-type: none"> - 출력값이 N개 - 입력값을 각각 지수함수로 취하고, 이를 정규화(=총합을 1로 만듦) - 정규화로 인해 각 출력값은 0~1 값을 가짐 				

	<ul style="list-style-type: none"> - 모든 출력값의 합은 반드시 1 - N가지 중 한 가지에 속할 확률 표현 가능(=Multi-class Classification)) <p>4. 로그 우도 목적함수</p> <p>(1) 모든 출력 노드값을 사용하는 MSE 나 교차엔트로피와 달리 \log 라는 하나의 노드만 사용</p> <p>5. 데이터 전처리시 주의할 점</p> <p>(1) 모든 특징이 양수일 경우 가중치가 뭉치로 증감하면 최저점을 찾아가는 경로가 갈팡질팡하며 느린 수렴을 한다.</p> <p>6. 모멘텀</p> <p>(1) 그레이디언트 잡음 현상: 기계학습은 훈련집합을 이용해 그레이디언트 추정: 잡음가능성 높음</p> <p>7. 배치정규화</p> <p>(1) Covariate shift 현상: 학습이 진행되면서 층 1 개의 매개변수가 바꿈에 따라 x 가 따라서 바뀐다. 층 2 층에서 보면 자신에게 입력되는 데이터의 분포가 수시로 바뀌게 되는 것이다. 최적화 후에는 각 노드에 전체 평균과 분산을 저장한다.</p>
--	---

학과	컴퓨터 전자시스템 공학	학번	201702899	이름	이학빈
구분	내용				
학습 범위	11주차 강의				
학습 내용	<p>-교차 엔트로피 목적함수</p> <p>분류 문제에서 사용되는 손실 함수 중 하나입니다. 이는 모델의 예측값과 실제 레이블 간의 차이를 측정하여 모델을 학습시키는 데 사용됩니다.</p> <p>분류 문제에서는 모델이 주어진 입력에 대해 클래스 또는 카테고리를 예측하는데 관심이 있습니다. 이때 모델이 출력하는 값은 일반적으로 확률 분포로 해석됩니다. 예를 들어, 소프트맥스(softmax) 함수를 통과한 출력은 각 클래스에 속할 확률로 이루어진 벡터가 됩니다.</p> <p>교차 엔트로피 목적 함수는 이러한 출력 확률 분포와 실제 레이블의 분포 사이의 차이를 측정하여 모델을 학습합니다. 다음은 교차 엔트로피 손실 함수의 수식입니다.</p> <p>정답 레이블을 y로 표현하고, 모델의 예측을 p로 표현했을 때, 두 분포 간의 교차 엔트로피 손실 함수는 다음과 같이 정의됩니다.</p> $J(y,p)=-\sum_i y_i \cdot \log(p_i)$ <p>여기서 y_i는 실제 레이블의 i번째 클래스에 대한 값이며, p_i는 모델의 예측값에서 해당 클래스에 속할 확률입니다. 위의 식에서 주로 y는 원-핫 인코딩된 실제 레이블이고, p는 소프트맥스 함수를 통과한 모델의 출력입니다.</p> <p>이 손실 함수는 모델의 예측이 실제 레이블과 얼마나 다른지 측정합니다. 모델이 정확한 예측을 할 때, 교차 엔트로피 손실은 최소화됩니다. 따라서 모델은 이 손실 함수를 최소화하는 방향으로 학습하여 예측을 개선하게 됩니다.</p> <p>요약하면, 교차 엔트로피 목적 함수는 분류 문제에서 사용되는 손실 함수로, 모델의 예측 확률 분포와 실제 레이블 간의 차이를 측정하여 모델을 학습시키는 데 사용됩니다. 이 함수는 모델이 더 정확한 예측을 하도록 하기 위해 최적화됩니다.</p> <p>-데이터 전처리</p> <p>데이터 전처리는 데이터를 분석하거나 머신 러닝 모델에 입력으로 사용하기 전에 데이터를 준비하고 정리하는 과정을 말합니다. 이는 데이터를 더 적절하게 사용할 수 있는 형태로 가공하여 모델의 성능을 향상시키고, 잘못된 데이터나 노이즈를 제거하여 모델의 학습 과정을 최적화하는 데 도움을 줍니다.</p> <p>-가중치 초기화</p>				

	<p>가중치 초기화(Weight Initialization)는 신경망 모델을 학습하기 위해 가중치 파라미터를 초기화하는 과정을 의미합니다. 초기화는 모델이 학습을 시작할 때 가중치가 어떤 값으로 설정되어야 하는지 결정하는 중요한 단계입니다. 적절한 가중치 초기화는 모델의 수렴 속도와 성능에 영향을 미칩니다.</p> <p>-모멘텀</p> <p>모멘텀(Momentum)은 경사 하강법(Gradient Descent)의 한 변종으로, 학습 알고리즘에서 사용되는 최적화 기법 중 하나입니다. 이 방법은 기존의 경사 하강법에 비해 수렴 속도를 높이고, 지역 최솟값에 빠지지 않도록 도와줍니다.</p> <p>경사 하강법은 현재의 가중치에서 손실 함수의 그레디언트(기울기)에 비례하여 가중치를 업데이트합니다. 그러나 가중치 업데이트 시 이전의 업데이트 방향을 고려하지 않고 단순히 현재의 기울기만 고려합니다. 모멘텀은 이러한 단점을 보완하여 더욱 효율적인 업데이트를 수행합니다.</p> <p>-적응적 학습률</p> <p>적응적 학습률(Adaptive Learning Rate)은 머신 러닝과 최적화 알고리즘에서 사용되는 학습률(learning rate)을 동적으로 조절하여 모델의 수렴 속도를 최적화하는 방법입니다. 학습률은 모델의 가중치를 업데이트할 때 사용되는 값으로, 가중치의 조정 정도를 결정합니다.</p> <p>일반적으로 고정된 학습률을 사용하는 경우, 모든 파라미터 업데이트에 동일한 학습률을 적용합니다. 그러나 적응적 학습률은 각각의 파라미터에 대해 독립적으로 학습률을 조정합니다. 이를 통해 모델의 수렴 속도를 높이고, 학습의 안정성을 향상시킬 수 있습니다.</p> <p>-활성 함수</p> <p>활성화 함수(Activation Function)는 인공 신경망의 각 뉴런에서 입력 신호의 가중치 합을 계산한 후, 그 결과를 다음 층의 뉴런에 전달하는 비선형 함수입니다. 이 함수는 신경망이 복잡한 패턴을 학습할 수 있도록 돕고, 모델의 출력을 결정합니다.</p> <p>시그모이드(Sigmoid) 함수, 하이퍼볼릭 탄젠트(Tanh) 함수, 렐루(Rectified Linear Unit, ReLU) 함수, 리키 렐루(Leaky ReLU) 함수, 소프트맥스(Softmax) 함수 등이 있습니다.</p> <p>-배치 정규화</p> <p>배치 정규화(Batch Normalization)는 딥러닝 신경망에서 사용되는 기술 중 하나로, 신경망의 각 층에서 입력 데이터의 분포를 정규화하여 학습을 안정화하는 방법입니다. 이를 통해 학습 과정의 안정성을 높이고, 더 빠르게 수렴하며, 더 높은 성</p>
--	---

능을 달성할 수 있습니다.

배치 정규화는 주로 다음과 같은 과정을 거칩니다.

1. 미니배치 단위의 정규화: 각 미니배치의 입력 데이터에 대해 평균과 분산을 계산합니다.
2. 정규화된 데이터 계산: 각 입력을 해당 미니배치의 평균과 분산으로 정규화합니다.
3. 스케일 조정 및 이동: 정규화된 데이터에 대해 스케일(scale)과 이동(shift)을 수행합니다. 이 과정에서 학습 가능한 가중치와 편향을 사용합니다.
4. 학습: 배치 정규화 계층은 학습 중에 적절한 가중치와 편향을 학습하며, 이 정보는 역전파(backpropagation)를 통해 업데이트됩니다.

배치 정규화는 신경망에서 중요한 기법 중 하나이며, 일반적으로 모든 은닉층에 적용하여 모델의 성능을 향상시키는 데 도움을 줍니다. 하지만 모든 상황에서 항상 도움이 되는 것은 아니며, 특히 작은 데이터셋이나 RNN(순환 신경망)과 같이 시퀀스 데이터에 적용할 때에는 주의가 필요합니다.

학과	철학과	학번	201803758	이름	탁성재
구분	내용				
학습 범위	11주차 강의				
학습 내용	<p>딥러닝 최적화는 학습 알고리즘을 효율적으로 조정하여 모델의 성능을 향상시키는 핵심 과정입니다. 그러나 딥러닝은 여러 어려움을 겪고 있습니다. 고차원 특징, 데이터의 희소성, 그리고 학습 시간이 오래 걸린다는 점이 그 중 일부입니다.</p> <p>평균제곱 오차는 주로 회귀 문제에서 사용되며, 실제 결과와 모델의 예측 결과 간의 제곱 오차를 최소화하는 방향으로 가중치와 바이어스를 조정합니다. 그러나 학습이 느릴 수 있으며, 이는 예측값이 특정 범위를 벗어날 때 기울기가 0에 가까워져 느린 학습이 발생하는 현상이 있습니다.</p> <p>교차 엔트로피 목적 함수는 주로 분류 문제에서 사용되며, 정확한 예측에 대한 보상 및 부정확한 예측에 대한 패널티를 제공합니다. 이를 통해 모델은 빠르게 학습하고 정확한 분류를 수행하도록 유도됩니다.</p> <p>활성 함수인 softmax는 출력값을 정규화하여 확률 분포의 형태로 변환시킵니다. 이는 각 클래스에 대한 확률을 나타내므로 분류 모델에서 중요한 역할을 합니다.</p> <p>로그우도 목적 함수는 확률적 모델에서 사용되며, 출력값과 실제 결과 간의 차이를 로그 함수를 통해 계산합니다. 이를 최소화하여 모델을 향상시킵니다.</p> <p>성능 향상을 위한 데이터 전처리는 각 데이터의 단위와 크기가 다르기 때문에 필수적입니다. 특히, 정규화를 통해 데이터의 평균을 0으로, 표준편차를 1로 조정하여 모델이 일관된 학습을 진행할 수 있도록 합니다.</p> <p>가중치 초기화는 모델의 학습에 중요한 영향을 미치는데, 대칭성을 파괴하고 학습을 빠르게 진행하기 위해 적절한 초기화 방법을 선택해야 합니다.</p> <p>모멘텀 및 적응적 학습률(Adam)은 학습 속도를 개선하기 위한 중요한 기법 중 하나입니다. 모멘텀은 현재 위치와 이동 방향을 고려하여 이전에 이동한 정도를 반영하여 빠르게 수렴하도록 돕고, Adam은 적응적 학습률을 통해 효율적으로 학습을 진행합니다.</p> <p>적응적 기울기(AdaGrad)와 RMSProp은 변수별로 학습률을 다르게 조절하여 효율적으로 학습할 수 있도록 돕습니다. 이러한 최적화 기법들을 통해 딥러닝 모델의 학습 성능을 향상시키고 더 빠르게 수렴할 수 있습니다.</p>				

학과	컴퓨터전자시스템 공학과	학번	201800615	이름	김동규
구분	내용				
학습 범위	기계학습 5장 5.1 딥러닝 최적화 5.2 성능 향상을 위한 요령				
학습 내용	<p>딥러닝 최적화</p> <p>딥러닝이 어려운 이유</p> <p>고차원 특징, 데이터의 희소성, 오랜 시간</p> <p>평균제곱 오차</p> <p>오차=실제결과y-계산결과o</p> <p>계산결과=가중치w,바이어스b를가지고계산한값</p> <p>이때 가중치를 기준으로 미분할수도, 바이어스를 기준으로 미분할 수도 있음</p> <p>또한 예측값이 일정 범위를 넘어가기 시작하면 기울기가 0에 가까워져 느린 학습이 발생함</p> <p>교차 엔트로피 목적함수</p> <p>$y \log o = y$의 사건이 발생하고, 예측이 정확하게 y로 예측된 경우</p> <p>$(1-y) \log (1-o) = y$의 사건이 발생하지 않고, 예측이 y를 예측하지 않은 경우</p> <p>오차가 클수록 더 큰 벌점이 부여되어 느린 학습이 발생하지 않음</p> <p>softmax 활성화함수</p> <p>영향을 적게 미치는 값을 억제, 영향을 크게 미치는 값을 극대화시킴 모든값을 더하면 1이 되어 확률과 비슷한 형태를 가지게 됨</p> <p>로그우도 목적함수</p> <p>에러 값은 로그 함수에서의 값을 가져옴</p> <p>출력값이 결과(1)과의 차이를 계산한 후(x축의 값), \log를 취해 값을 추산함(y값의 값)</p> <p>성능 향상을 위한요령</p> <p>데이터 전처리</p> <p>각 데이터별 단위, 크기가 달라서 영향이 일정하지 않음</p> <p>모든 특징이 양수인 문제?</p> <p>해당 데이터 셋의 평균을 0으로, 표준편차를 1로 바꾸어 사용함 - 정규화</p> <p>가중치 초기화</p> <p>대칭성파괴 - 모든 값들이 동일한 값으로 연결된 상태층을 통과해도 동일한 결과가 나오기에 부쉬야 함</p>				

	<p>크기 r 안에서 $(-r, r)$ 범위의 난수를 사용해 생성 난수를 통한 가중치 초기화 가중치 행렬의 수직 초기화</p> <p>파괴방법 모멘텀 현재 위치를 일종의 벡터로 봄 다음 벡터는 기계학습의 방법을 골라 그래디언트를 추출함 현재 위치와 다음으로 이동하는 그래디언트를 활용해 계속 이동</p> <p>적응적 학습률 각 방식의 설명은 여기에 존재함 https://velog.io/@daniel1025/AdamAdaptive-Momentum 학습률이 높을 때, 검은색으로 표현함 학습률이 낮을 때, 파란색으로 표현함 그래디언트를 추출한 후, 이동하는 거리를 나타냄</p> <p>adaGrad(적응적 기울기) 각 요소의 특징별로 학습률을 다르게 조정함 학습이 진행됨에 따라 학습률이 점점 0에 가까워짐 이전까지의 이동량이 큼 = 이번 이동량은 전체와 비교해서 절대량이 상대적으로 작아짐=조금만 이동 이전까지의 이동량이 작음 = 이번 이동량은 전체와 비교해서 절대량이 상대적으로 커짐=많이이동</p> <p>RMSProp adaGrad에서 이동이 얼마나 이전에 발생했는지에 따라 다르게 조정함 최근의 기울기는 많이 반영, 과거의 기울기는 적게 반영함 변수마다 다른 학습률을 적용할 수 있어서 효율적임 adaGrad보다 오래 학습할 수 있음</p>
--	--