《 수2 임의세특 보고서 》

하이에듀

주제	머신러닝에 사용되는 경사하강법
관련 분야	#AI #머신러닝
요약	서론 : 경사하강법의 정의 본론 : 경사하강법 수식의 유도 결론 : 경사하강법의 활용 위와 같은 흐름을 추천해 드립니다.

자료1. 경사하강법

신경망은 2000년대에 들어오면서 딥러닝(deep learning)이란 새로운 이름으로 개명되었고 구글의 딥러닝 바둑 알고리즘이 세계적인 프로기사들을 이김으로써 세상의 조명을 받게 되었다. 신경망이 새롭게 태어날 수 있었던 것은 빅데이터와 프로세서의 발전 덕분이다. 빅데이터 확보로 많은 경우의 수를 테스트하고, 프로세서의 성능개선 또는 클라우드 컴퓨팅 기술로 엄청난 매개변수들을 학습할 수 있게 되었다. 물론 힌톤교수의 불굴의 노력도 중요한 역할을 하였다. 1990년대까지 신경망을 연구하던 많은 학자들이 신경망의 한계를 느끼면서 새로운 방향으로 거의 전환하였다. 그러나 힌톤교수는 신경망을 계속 연구할 수 있는 여건을 찾아서 대학교를 옮기기도 하였고 결국은 신경망을 딥러닝으로 재탄생시켰다. 딥러닝은 방대한 양의 데이터를 다루며 또 이를 위해서 구조적으로 은닉층의 수가 상당히 늘어난다. 이것은 신경세포들을 연결하는 매개변수들이 늘어나고 이를 학습하는 계산량이 방대해졌음을 의미한다. 프로세서와 클라우드 컴퓨팅 기술의 발전으로 방대한 양의 계산을 할 수 있게 되었지만 아직도 효율적으로 학습할 수 있는 방법이 필요하다.

경사하강법(gradient descent method)은 신경망을 학습하는 데 현재 가장 많이 쓰이는 알고리즘이다. 학습이란 손실함수가 최소값이 되도록 매개변수를 갱신하는 것이다. 손실함수는 실제값과 예측값의 차이를 수치화해주는 함수이다. 경사하강법은 두 값의 차이 즉 오

차가 최소화되도록 매개변수를 갱신하는데 손실함수의 기울기를 사용하는 것으로 현재 최고의 딥러닝 학습알고리즘을 제공하는 라이브러리에서 사용되고 있다. 그러나 이 알고리즘들은 블랙박스형태로 제공되고 있어서 다양한 경사하강법들의 장단점을 파악하는 것이 쉽지 않다. 따라서 효율적인 학습을 위하여 데이터의 특성에 맞는 학습알고리즘을 선택하는 것은 그리 간단하지 않다.

gradient descent 방법은 steepest descent 방법이라고도 불리는데, 함수 값이 낮아지는 방향으로 독립 변수 값을 변형시켜가면서 최종적으로는 최소 함수 값을 갖도록 하는 독립 변수 값을 찾는 방법이다.

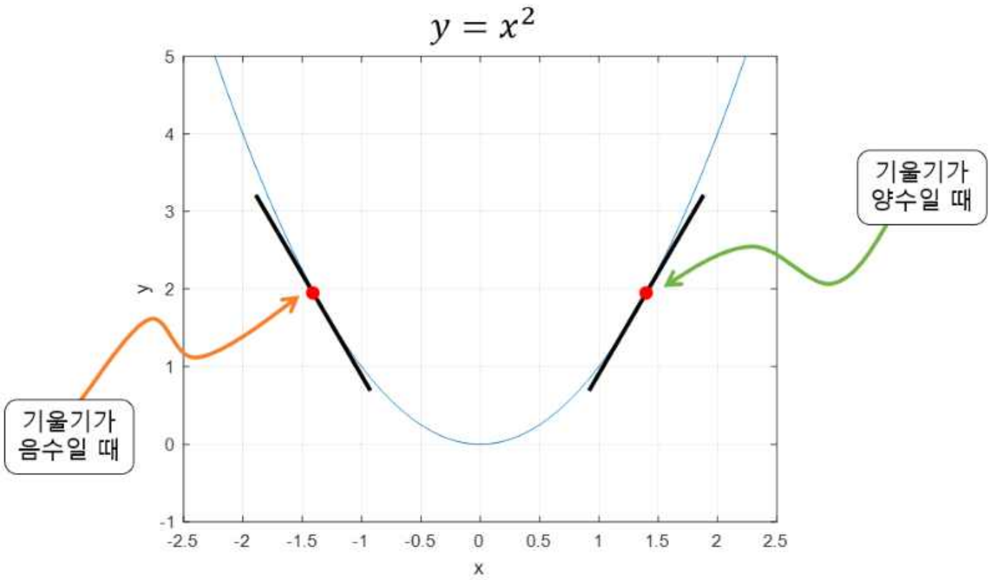
gradient descent는 함수의 최소값을 찾는 문제에서 활용된다. 함수의 최소, 최댓값을 찾으려면 “미분계수가 0인 지점을 찾으면 되지 않느냐?”라고 물을 수 있는데, 미분계수가 0인 지점을 찾는 방식이 아닌 gradient descent를 이용해 함수의 최소값을 찾는 주된 이유는 우리가 주로 실제 분석에서 맞딱드리게 되는 함수들은 닫힌 형태(closed form)가 아니거나 함수의 형태가 복잡해 (가령, 비선형함수) 미분계수와 그 근을 계산하기 어려운 경우가 많고, 실제 미분계수를 계산하는 과정을 컴퓨터로 구현하는 것에 비해 gradient descent는 컴퓨터로 비교적 쉽게 구현할 수 있기 때문이다. 추가적으로, 데이터 양이 매우 큰 경우 gradient descent와 같은 iterative한 방법을 통해 해를 구하면 계산량 측면에서 더 효율적으로 해를 구할 수 있다.

딥러닝을위한경사하강법비교.pdf 파일 참고 부탁드립니다.
https://blog.naver.com/sichang01/222399820083

자료2. 경사하강법의 수식 유도

gradient descent는 함수의 기울기(즉, gradient)를 이용해 x의 값을 어디로 옮겼을 때 함수가 최소값을 찾는지 알아보는 방법이라고 할 수 있다. 기울기가 양수라는 것은 x값이 커질 수록 함수 값이 커진다는 것을 의미하고, 반대로 기울기가 음수라면 x값이 커질 수록 함수의 값이 작아진다는 것을 의미한다고 볼 수 있다. 또, 기울기의 값이 크다는 것은 가파르다는 것을 의미하기도 하지만, 또 한편으로는 x의 위치가 최소값/최댓값에 해당되는 x 좌표로부터 멀리 떨어져있는 것을 의미하기도 한다.

이를 이용해 특정 포인트 x에서 x가 커질 수록 함수값이 커지는 중이라면 (즉, 기울기의 부호는 양수) 음의 방향으로 x를 옮겨야 할 것이고, 반대로 특정 포인트 x에서 x가 커질 수록 함수값이 작아지는 중이라면 (즉, 기울기의 부호는 음수) 양의 방향으로 x를 옮기면 된다. 이 논리를 수식으로 쓰면 다음과 같다.
x_{i+1} = x_i − 이동 거리 × 기울기의 부호
여기서 x_i와 x_{i+1}은 각각 i번째 계산된 x의 좌표와 i+1번째 계산된 x의 좌표를 의미한다.
그러면 여기서 이동거리는 어떻게 생각해야 할까? 그것은 gradient의 크기를 이용하면 된



다.

식 (1)의 문제점을 생각해보면 “이동 거리”라는 factor를 어떻게 구할지 생각해보아야 한다는 점이다. 이 문제에 대해 다시 잘 생각해보면 미분 계수(즉, 기울기 혹은 gradient)값은 극소값에 가까울 수록 그 값이 작아진다. 사실 극대값에 가까울 때에도 미분 계수는 작아지기 마련인데, gradient descent 과정에서 극대값에 머물러 있는 경우는 극히 드물기 때문에 이 문제에 대해서는 고려하지 않고자 한다. 따라서, 이동거리에 사용할 값을 gradient의 크기와 비례하는 factor를 이용하면 현재 x의 값이 극소값에서 멀 때는 많이 이동하고, 극소값에 가까워졌을 때는 조금씩 이동할 수 있게 된다. 즉, 이동거리는 gradient 값을 직접 이용하되, 이동 거리를 적절히 사용자가 조절 할 수 있게 수식을 조정해 줌으로써 상황에 맞게 이동거리를 맞춰나갈 수 있게 하면 될 것이다. 이 때, 이동 거리의 조정 값을 보통 step size라고 부르고 기호는 α로 쓰도록 하겠다. 따라서 최종 수식은 다음과 같이 계산할 수 있다.

$$x_{i+1} = x_i - \alpha \frac{df}{dx}(x_i)$$

https://angeloyeo.github.io/2020/08/16/gradient_descent.html

자료3. 경사하강법 사용 예시 (플래시 메모리)

플래시 메모리는 정해진 횟수 이상으로 데이터를 쓰고 지울 경우 마모가 발생하여 불량률

이 증가하기 때문에, 특정 횟수의 데이터 처리를 보증하는 보증기간 이후 데이터 처리가 증가할수록 지수곡선 형태로 불량이 발생한다. 그리고 이 곡선은 경사하강법의 곡선과 동일한 형태를 가지며 불량률 곡선의 순간 기울기는 손실함수에서의 비용과 유사한 특성을 갖는다.

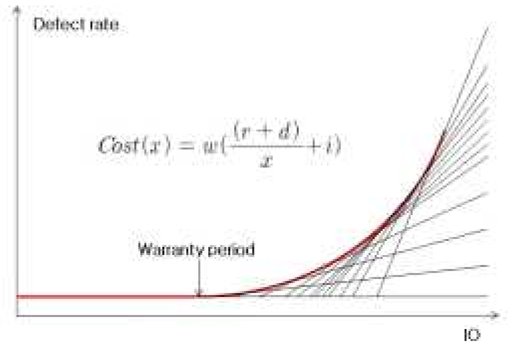


Fig. 3. Gradient Descent Model on the Rate of Flash Failure

Fig. 3는 플래시 메모리의 불량률에 경사하강법을 적용한 것을 보여주고 있다. 그림과 같이 데이터 처리가 증가하여 보증기간이 지난 순간부터 불량률은 지수의 형태로 증가를 하는데, 일정 불량률 이후부터는 더 이상 저장 장치를 사용할 수 없다. 이러한 현상을 경사하강법에 적용하면, 각 불량률이 증가하는 순간의 기울기를 계산하기 위해 손실 함수에 대응했을 때 특정 데이터 처리 횟수에서 계산된 손실비용 이후에는 고장이 더 이상 저장 장치를 사용할 수 없는 수명 마감이 발생하는 것을 예측할 수 있다. 즉, 여기서 사용하는 경사하강법은 최적화된 값을 찾기 위한 목적이 아니고, 제품의 불량률 발생 곡선을 학습하고, 이를 이용하여 수명 마감을 예측하는 데 사용된다. 본 논문에서는 Fig. 3 내부 수식과 같은 선형 기반의 비용 모델을 제안한다. 수식에서 겺는 생산제품에 따라 변경될 수 있는 미디어 구성의 특성에 대한 가중치이다. 이 가중치는 하나의 불량이 발견되었을 때 다른 하나의 예비 블록으로 대체 가능한 경우는 1이다. 그러나 하나의 블록이 불량이 발생하더라도 이웃하는 다른 블록들과 같이 여러 개의 블록을 교체해야 할 경우 증가된 교체 비용에 대한 가중치는 실제로 교체된 블록의 개수만큼 증가한다. 그리고 겳는 누적 처리된 DWPD를 의미하고, 겂은 마지막 DWPD의 데이터가 처리되는 동안 발생한 불량의 개수이다. 여기서 강는 불량률이 예상값 보다 높거나 낮을 때 이를 보정 해주기 위한 보정값이다. 마지막으로 갱가 의미하는 것은 생산 과정에서 발생할 수 있는 초기 불량을 의미한다.

자료4. 경사하강법 사용 예시

1. 비용 최소화(Cost Minimization)

상황 설명: 경사하강법은 비즈니스에서 비용을 최소화하는 데 사용될 수 있습니다. 예를 들어, 어떤 제품을 만드는 데 드는 비용(재료비, 인건비 등)을 나타내는 함수가 있을 때,

이 비용을 최소화하는 생산 수준을 찾는 것입니다.
탐구 활동: 학생들은 간단한 비용 함수를 설정하고, 경사하강법을 사용하여 비용을 최소화하는 생산 수준을 계산할 수 있습니다.

2. 물리학에서의 최적화 문제(Optimization in Physics)

상황 설명: 물리학에서는 경사하강법을 사용하여 최적의 결과를 얻을 수 있는 조건을 찾습니다. 예를 들어, 특정 물체의 최적 낙하 경로나 에너지가 최소가 되는 상태 등을 찾는 데 사용될 수 있습니다.
탐구 활동: 학생들은 간단한 물리학 문제를 설정하고, 경사하강법을 이용해 최적의 조건을 찾아보는 활동을 할 수 있습니다.

3. 기계학습에서의 매개변수 조정(Parameter Tuning in Machine Learning)

상황 설명: 기계학습에서 경사하강법은 모델의 매개변수를 조정하는 데 중요한 역할을 합니다. 예를 들어, 데이터를 가장 잘 설명하는 선형 회귀 모델의 매개변수를 찾을 때 사용됩니다.
탐구 활동: 학생들은 간단한 선형 회귀 문제를 설정하고, 경사하강법을 사용하여 최적의 매개변수를 찾는 활동을 할 수 있습니다.

4. 식품 산업에서의 레시피 최적화(Recipe Optimization in Food Industry)

상황 설명: 식품 산업에서는 경사하강법을 사용하여 최적의 맛을 내는 레시피 비율을 찾을 수 있습니다. 예를 들어, 케이크 레시피에서 각 재료의 비율을 조정하여 최상의 맛과 질감을 찾는 것입니다.
탐구 활동: 학생들은 간단한 레시피를 설정하고, 여러 변수(재료의 양 등)를 조정하며 최적의 결과를 찾는 실험을 할 수 있습니다.