

[인공지능] 3차 결과 보고서

2018920002 컴퓨터과학부 고다현

1. 실행 환경

구분	값
운영체제	Windows 10
개발 도구	Visual Studio Code
언어	C++14

2. 코드 설명

- main.cpp
 - AND, OR, XOR에 대한 n차원의 input을 생성한다.
 - 2차원의 DONUT 데이터를 생성한다.
 - AND, OR, XOR gate, DONUT 데이터에 대한 multilayer perceptron을 만들어 학습시킨다.
 - 각 multilayer perceptron을 생성할 때, layer 수를 입력받는다.
- Perceptron.h
 - AND, OR, XOR gate, DONUT 데이터 각각에 맞게 multilayer perceptron을 학습시키도록 구현하였다.
 - 입력받은 layer 수 만큼 layer를 구현하였고, layer 당 노드 수는 input 데이터의 dimension과 같게 하였다.
 - layer 당 노드 수는 변수로 두었으나, 현재 코드는 layer 당 노드 수가 dimension과 같을 때만 동작한다.
 - Perceptron 클래스 안에 calculate 함수, learning 함수, activation 함수, D_activation 함수 등이 정의되어 있다.
 - 생성자 함수 : AND, OR, XOR gate, DONUT 데이터에 대해 target을 계산, weight값을 랜덤으로 초기화
 - calculate 함수 : 각 test case별 output을 계산
 - learning 함수 : weight를 업데이트해 perceptron을 학습시킴
 - activation 함수 : ELU 함수를 사용
 - sigmoid함수도 사용해봤으나 학습에 큰 차이는 없었음
 - D_activation 함수 : activation 함수를 미분함. backpropagation algorithm에 사용됨
 - repeat_learning 함수 : while문을 통해 calculate 함수, learning 함수 등을 실행시켜 multilayer perceptron을 학습시킴. epoch > 3000이 되면 while문을 종료하고 학습 실패로 판단.
 - isLearned 함수 : error가 정해진 값(0.01)보다 작아져서 학습이 완료됐는지 알려줌
 - Learning rate(C)는 0.1로 설정하였다.

- 초기 weight값은 [-1.0, 1.0) 범위 내의 실수로 한정하였다.

- 자료형 : float

3. 실행 결과

코드 구현에 실패하였다.

AND, OR, XOR gate

```
output : 2.63806e+029
target : 0
ex_error : 6.95934e+058
new_error : 1.00215e+059
error_diff : 3.06211e+058
```

다음은 layer=1일 때(hidden layer가 0개)의 AND gate 학습 결과 중 일부이다.

오류를 확인하기 위해 특정 test case에 대한 output, target, error값(ex_error)과 이를 토대로 weight값을 수정한 후의 error값(new_error)을 출력해보았다. 위 변수들의 값의 소수점 자리수가 너무 크고 이는 학습을 수행할수록 점점 커졌다. 결국 학습 후반부에는 아래와 같은 결과가 나타났다.

```
output : 1.98446e+161
target : 0
ex_error : inf
new_error : inf
error_diff : nan
```

값이 제대로 표기되지 않고 inf라고 뜨는데, 이는 소수점 자리수가 너무 커져서 **overflow**가 일어난 것이다. overflow를 해결할 방법을 찾지 못했고, overflow와는 별개로 epoch가 3000이 되어도 학습이 성공하지 못했기 때문에 학습은 실패한 채로 종료됐다.

OR, XOR gate도 위와 같은 문제가 발생하였다.

아래는 output, target, ex_error, new_error을 출력하지 않고 학습 실패시의 epoch 값만 출력한 결과이다.

```
dimension of input? : 2
AND gate learning
layer? : 1
Learning failed at 3001epoch.

OR gate learning
layer? : 2
Learning failed at 364epoch.

XOR gate learning
layer? : 3
Learning failed at 375epoch.
```

DONUT 데이터

DONUT 데이터 학습의 경우 weight값을 랜덤으로 초기화하는데 까진 성공했으나, 그 이후 연산에서 Segmentation fault가 발생해 프로그램이 중단됐다. 디버깅해본 결과, calculate 함수 쪽에서 벡터의 index에 접근할 때 범위 오류가 생긴 것으로 추측된다.

```
DONUT data learning
layer? : 3
-0.613684
-0.622735
-0.631495
-0.639971
-0.648172
Segmentation fault
```

4. 고찰

왜 multilayer 가 필요할까?

저번 과제에서는 1-layer perceptron으로 AND, OR, XOR gate 학습을 시도했었다. AND, OR gate는 학습에 성공했으나 XOR gate는 학습에 실패했다. 이는 AND, OR gate의 경우엔 linear problem이지만 XOR gate의 경우엔 non-linear problem이기 때문이다. 즉 input값을 2차원 평면에 나타냈을 때, 하나의 직선으로는 이를 target 값에 따라 분류할 수 없다. 따라서 여러 개의 layer를 거친 학습을 통해 알맞게 변형해 linear problem으로 만들어야 한다.

저번 과제에서 AND, OR gate는 1-layer perceptron으로 학습에 성공했기 때문에 이번 과제에서도 1-layer가 최적이고 오히려 layer가 많을수록 학습에 부정적인 영향을 끼칠 것이라고 예상했다. 하지만 결과는 그렇지 않았고 layer가 1개일 때보다 layer가 3개일 때 더 빠르게 학습 실패를 판단하는 등 학습 결과가 빠르게 나타났다.(구현에 실패하였으니 후자가 전자보다 빨리 학습에 성공하는지는 알 수 없다. 하지만 학습 실패를 느리게 판단하는 것 보단 빠르게 판단하는 것이 더 좋다고 생각했다.) 그렇다고 무조건 layer가 많을수록 학습 결과가 빠르게 나타나는 것은 아니었다.

내가 'AND, OR gate는 1-layer로 학습시키는게 최선일 것이다'라고 생각한 이유는 바로 overfitting 때문이다. 구현 초반에, layer 수를 몇으로 해야 하는지 감이 잡히지 않아 구글링해본 결과 overfitting과 underfitting에 대해 알게 되었다. overfitting은 layer 수를 너무 크게 잡았을 때 발생하고, 이로 인해 현재 데이터는 잘 분류할 수 있지만 미래의 데이터는 제대로 분류할 수 없을 수도 있다. test data 중 노이즈가 있는 경우, 이 노이즈까지 과도하게 반영해버린 것이다. underfitting은 그 반대로 layer 수를 너무 작게 잡았을 때 발생한다. 이런 개념 때문에 2-layer이상으로는 오히려 AND, OR gate 학습에서 overfitting을 일으킬 것이라고 예상한 것이다. 하지만 이번 과제에서 학습에 사용할 데이터들은 노이즈가 하나도 없다. 또 overfitting을 일으키기에 내가 설정한 layer 수(2, 3, ...)는 터무니없이 작다.

결과적으로 non-linear problem인 XOR gate와 DONUT 데이터 학습에는 multilayer가 반드시 필요하고, linear problem인 AND, OR gate 학습에는 반드시 필요한 것은 아니지만 있어서 나쁠 것은 전혀 없다.

5. 기타 제출물

- weight 값이 저장된 파일
: 학습 실패시의 weight값이라도 얻어 파일에 저장하려 했으나 weight값 또한 overflow가 일어나 값이 제대로 표현되지 않았다.
- error 그래프, 2차원 직선 그래프
: 값이 매우 많은데 이 많은 값을 반영해 그래프를 그려주는 프로그램을 모르겠다..
(저번 과제에선 값이 몇 개 되지 않아 직접 하나하나 그렸음..)