

기초 인공지능 네 번째 과제

학번 : 20191583

이름 : 김태곤

1. 3번 문제를 해결하기 위해 본인이 시도한 방법

3번째 문제는 조건에 맞게 nn.Module을 상속받는 클래스를 정의하는 것이다. 주어진 코드를 살펴보면 layers에 알맞은 값을 넣어야 한다. 먼저 MLP의 첫 번째 nn.Linear layer의 input_feature과 input_data 행렬의 column은 1024로 동일하다는 조건이 있다. layers에 in_features=1024, out_features는 초기 선언 시 입력 받는 List의 첫 번째 값이 들어간 nn.Linear를 넣는다. 이후 CustomMLP class에 define 되어 있는 select_act를 이용하여 layers에 activation function을 넣는다. select_act는 초기 선언 시 입력받은 action function의 str을 nn함수로 바꿔주는 역할을 한다. 초기 Linear 입력이 끝나면 num_layers의 수만큼(초기 입력을 했으니 num_layers-1) out_feat_list의 순서대로 in_features와 out_features가 들어간 nn.Linear와 action function을 layers에 넣는다.

2. 4번 문제를 해결하기 위해 본인이 시도한 방법. 가장 성능이 높았던 방법에 대해 소개. 최종적인 test 성능

4번 문제에서는 SEED, LEARNING_RATE, optimizer를 수정하여 test 정확도 40% 이상의 모델을 만들어야 한다. 학습률(Learning rate)은 사용자가 여러 번의 학습을 통해 최적의 값을 찾아야 한다. 학습률이 작다면 학습 시간이 오래 걸리고, 최적점에 도달하지 못할 수도 있다. 반대로 학습률이 크면 데이터가 무질서하게 이탈할 수 있고, 최적점에 수렴하지 못할 수도 있다. 따라서 직접 실행시켜 보면서 최적의 값을 찾아야 한다. optimizal는 학습률에 연관이 있다. 4번 문제에서 초기 주어진 학습률 0.001, SEED 100으로 실행시키면 test 정확도는 37.15%가 나온다. 학습률을 더 크게하여 0.01로 시도하면 10%로 40%에서 많이 벗어난 값이 나온다. 학습률을 낮춰 0.0001로 시도하면 41.58%로 초기 원하던 범위 내로 들어온다. 학습률은 0.0001로 고정시키고 SEED값을 100에서 50으로 바꾸면 정확도 41.77%로 좀 더 높은 값이 나온다. 따라서 최종적으로 SEED값은 50, LEARNING_RATE는 0.0001로 선택하였다.

```
100% |██████████| 391/391 [00:07<00:00, 54.31it/s]
11 average loss: 1.735907
11 average accuracy: 38.24%
100% |██████████| 391/391 [00:07<00:00, 54.32it/s]
12 average loss: 1.714556
12 average accuracy: 38.97%
100% |██████████| 391/391 [00:07<00:00, 53.28it/s]
13 average loss: 1.694922
13 average accuracy: 39.66%
100% |██████████| 391/391 [00:07<00:00, 52.25it/s]
14 average loss: 1.675810
14 average accuracy: 40.33%
100% |██████████| 391/391 [00:07<00:00, 53.82it/s]
15 average loss: 1.658002
15 average accuracy: 41.00%
100% |██████████| 391/391 [00:07<00:00, 54.09it/s]
16 average loss: 1.641160
16 average accuracy: 41.66%
100% |██████████| 391/391 [00:07<00:00, 54.19it/s]
17 average loss: 1.625236
17 average accuracy: 42.25%
100% |██████████| 391/391 [00:07<00:00, 54.66it/s]
18 average loss: 1.610093
18 average accuracy: 42.84%
100% |██████████| 391/391 [00:07<00:00, 54.37it/s]
19 average loss: 1.595020
19 average accuracy: 43.50%
100% |██████████| 391/391 [00:07<00:00, 54.17it/s]
20 average loss: 1.580877
20 average accuracy: 44.08%
100% |██████████| 79/79 [00:01<00:00, 61.20it/s]

TEST average accuracy: 41.77%
```