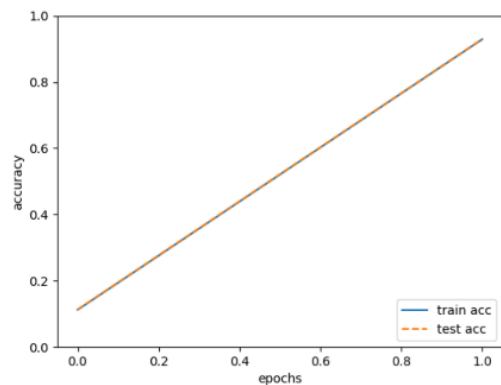


딥러닝 실제 9주차 과제

[1]

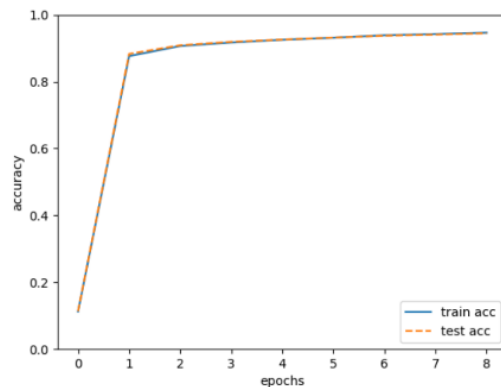
교재에서 제공된 `train_neuralnet.py` 를 수정하여 Batch 사이즈를 변화 시켜가며, 훈련데이터와 테스트데이터에 대한 정확도 추이가 어떻게 바뀌는지 그래프와 함께 기술하시오.



Batch size 10

Train acc : 0.929

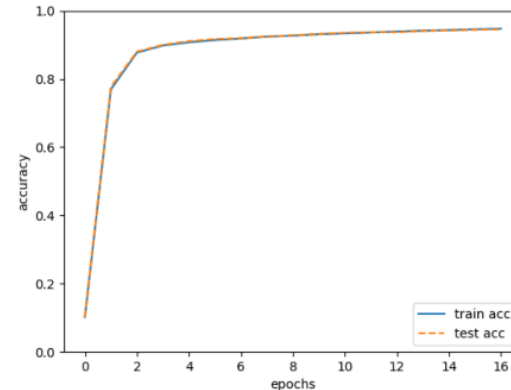
Test acc : 0.929



Batch size 50

Train acc : 0.946

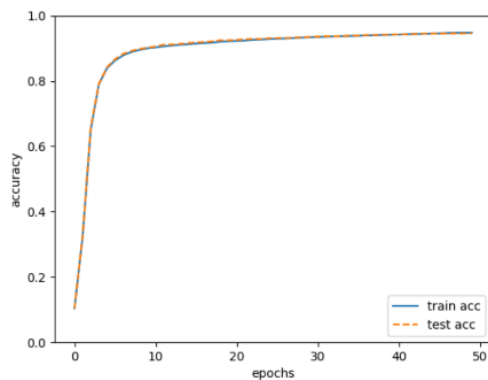
Test acc : 0.945



Batch size 100

Train acc : 0.946

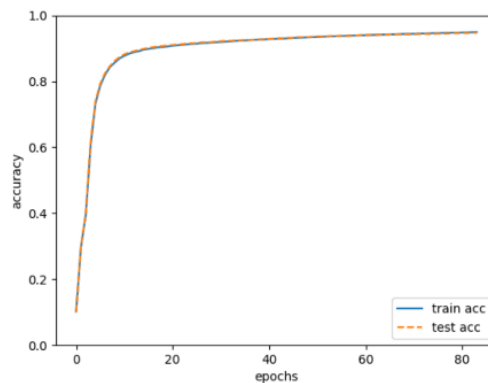
Test acc : 0.945



Batch size 300

Train acc : 0.947

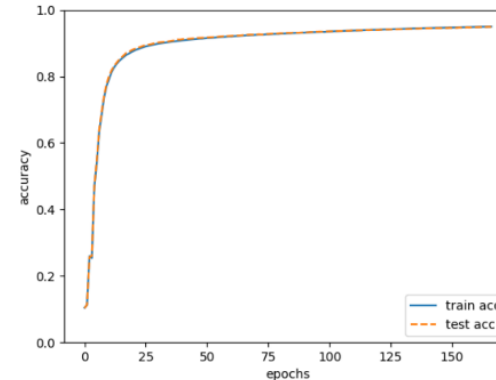
Test acc : 0.946



Batch size 500

Train acc : 0.948

Test acc : 0.946



Batch size 1000

Train acc : 0.948

Test acc : 0.948

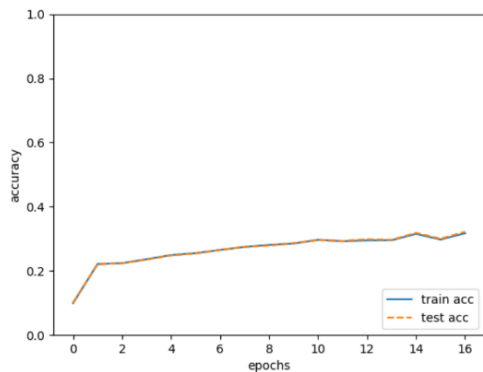
[1]

교재에서 제공된 `train_neuralnet.py` 를 수정하여 Batch 사이즈를 변화 시켜가며, 훈련데이터와 테스트데이터에 대한 정확도 추이가 어떻게 바뀌는지 그래프와 함께 기술하시오.

- Batch size : Mini Batch 실행 시 random하게 선정되어 학습되는 훈련데이터의 개수.
- Batch Size는 최소10, 최대 1000으로 설정하였으며, Batch Size 100을 기준으로 테스트를 진행했습니다
- Batch size를 줄일 경우 trainset과 testset의 정확도 그래프의 기울기가 급격하게 증가하였습니다.
- Batch size를 늘릴 경우 정확도 및 학습시간이 증가하였습니다.

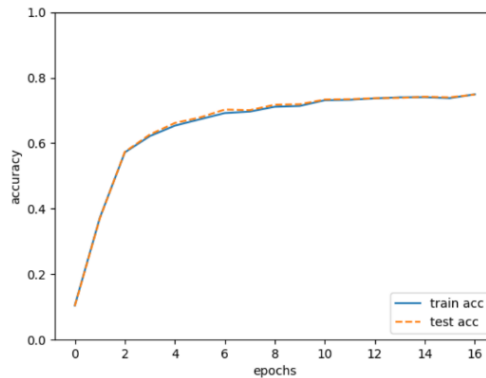
[2]

Hidden neuron의 개수를 변화 시켜가면서,
훈련데이터와 테스트데이터의 정확도 추이가 어떻게 바뀌는지 그래프와 함께 기술하시오.



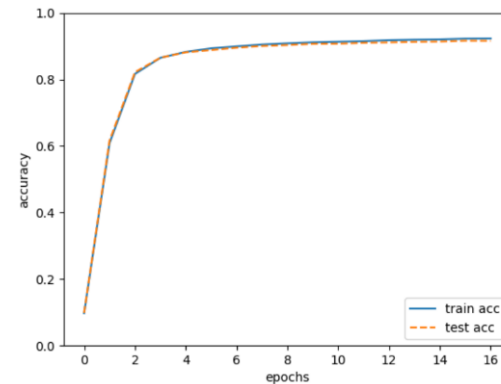
Hidden Neuron 1

Train acc : 0.317
Test acc : 0.321



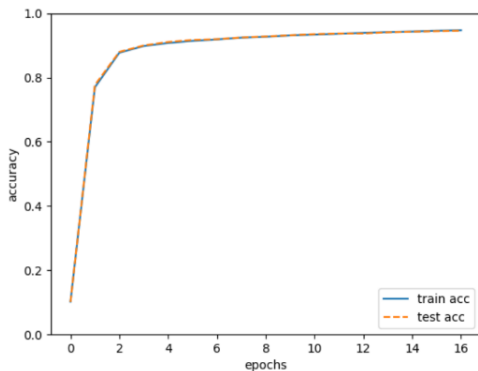
Hidden Neuron 3

Train acc : 0.749
Test acc : 0.748



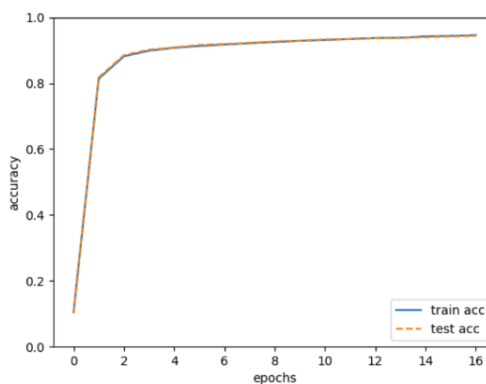
Hidden Neuron 10

Train acc : 0.922
Test acc : 0.916



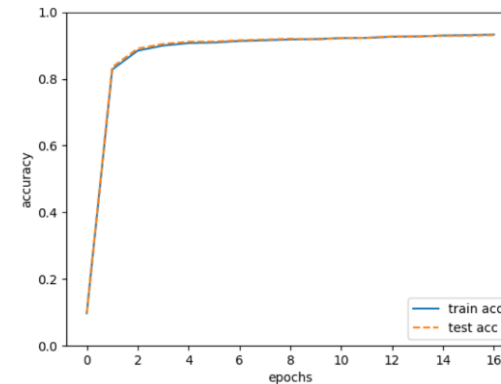
Hidden Neuron 50

Train acc : 0.946
Test acc : 0.945



Hidden Neuron 100

Train acc : 0.945
Test acc : 0.944



Hidden Neuron 500

Train acc : 0.932
Test acc : 0.930

[2]

Hidden neuron의 개수를 변화 시켜가면서,
훈련데이터와 테스트데이터의 정확도 추이가 어떻게 바뀌는지 그래프와 함께 기술하시오.

- Hidden Neuron : 은닉층의 뉴런
- Hidden Neuron의 개수는 최소1, 최대 500으로 설정하였으며
epoch 16, Hidden Neuron 50개를 기준으로 설정하여 테스트를 진행했습니다.
- Hidden Neuron 개수를 줄일 경우 학습시간이 감소했으며, Trainset과 Testset의 정확도가 떨어졌습니다.
- Hidden Neuron 개수를 늘릴 경우 학습 시간이 증가했으며, 50 이상 넘어가자 정확도가 떨어졌습니다.
따라서 학습 시 Hidden Neuron의 개수를 적절하게 설정하는 것이 매우 중요함을 알 수 있습니다.

[3]

위의 [1][2]에서 오버피팅 문제를 기술하시오.

- 오버피팅 : 학습데이터를 과하게 학습하는 것
- 학습 데이터에 대해서는 오차가 감소하지만, 학습에 사용되지 않은 테스트 데이터에 대해서는 오차 증가
- 복잡한 모델 사용 시 오버피팅 문제 초래 Ex) 과도하게 많은 Hidden Neuron의 개수
복잡한 모델 사용 시 분산은 증가하며, 편향은 감소하는 모습을 보인다.

[3]

위의 [1][2]에서 오버피팅 문제를 기술하시오.

[1]

large batch size	small batch size
정확한 기울기 추정 (낮은 분산)	기울기의 noise 생성(부정확) (높은 분산)
1 iteration 반복 당 높은 컴퓨터 비용	1 iteration 반복 당 낮은 계산 비용
병렬 처리의 높은 가용성 (빠른 학습)	낮은 병렬 처리 가용성 (느린 학습)

- 배치 사이즈가 작은 경우, 분산이 높아지는 현상을 보입니다.
- 분산이란 학습 데이터에 내재된 작은 변동으로 인하여 학습된 모델들 간의 차이를 말하며, 높은 분산은 오버피팅을 초래합니다.
- 이를 해결하기 위해 배치정규화를 수행합니다.
배치정규화는 정규화 레이어를 통해 평균과 분산을 정규화 하여, 분산이 높아지지 않도록 조절하는 것으로, 파라미터의 update가 mini-batch 단위로 일어나기 때문에 배치 정규화 역시 미니 배치 단위로 수행됩니다..

[2]

- hidden neuron의 층이 증가할수록, 모델이 복잡해지며, 복잡한 모델은 오버피팅을 초래합니다.
- 학습데이터를 늘려 복잡한 모델로 인한 오버피팅을 줄일 수 있습니다.