

위 문서를 요약하자면

위 문서의 내용은 가장 성능이 좋은 Hidden Layer 개수에 관한 글입니다.

그리고 문서에서는 하나의 Hidden Layer를 가진 Neural Network에서 가장 잘 작동한다고 합니다.

간혹 2개의 Hidden Layer를 가진 것에서 성능향상을 하는 경우도 간혹 발견된다고 합니다.

그 외에는 성능이 좋지 않다고 합니다.

↑ 단, computer vision과 sequence data 제외

실제로 1개의 Hidden Layer를 가진 MLP를 구현해보세요.

2개, 3개, 4개로 Hidden Layer를 늘려보시고 정확도를 비교해 보세요.

보통은 1개에서 가장 잘 작동하고, 2개도 어떤 경우는 정확도가 약간 증가하기도 합니다.

그러나 3개, 4개로 갈수록 정확도가 내려가게 됩니다.

faq 문서에서는 이 이유를 local minimum의 문제가 layer를 증가시킬수록 심해지는데 있다고 추정합니다.

layer가 깊어질수록 local minimum에서 빠져나오기 힘들다는 것이죠.

바로 이 부분이 기존의 MLP 연구가 줄어들게 된 부분입니다.

하지만 아직도 여러 분야에서 Hidden Layer 1, 2개로 구성된 MLP은 많이 사용하고 있습니다.

Problems that require more than two hidden layers were rare prior to deep learning. Two or fewer layers will often suffice with simple data sets. However, with complex datasets involving time-series or computer vision, additional layers can be helpful. The following table summarizes the capabilities of several common layer architectures.

Table: Determining the Number of Hidden Layers

Num Hidden Layers	Result
none	Only capable of representing linear separable functions or decisions.
1	Can approximate any function that contains a continuous mapping from one finite space to another.
2	Can represent an arbitrary decision boundary to arbitrary accuracy with rational activation functions and can approximate any smooth mapping to any accuracy.
>2	Additional layers can learn complex representations (sort of automatic feature engineering) for layer layers.