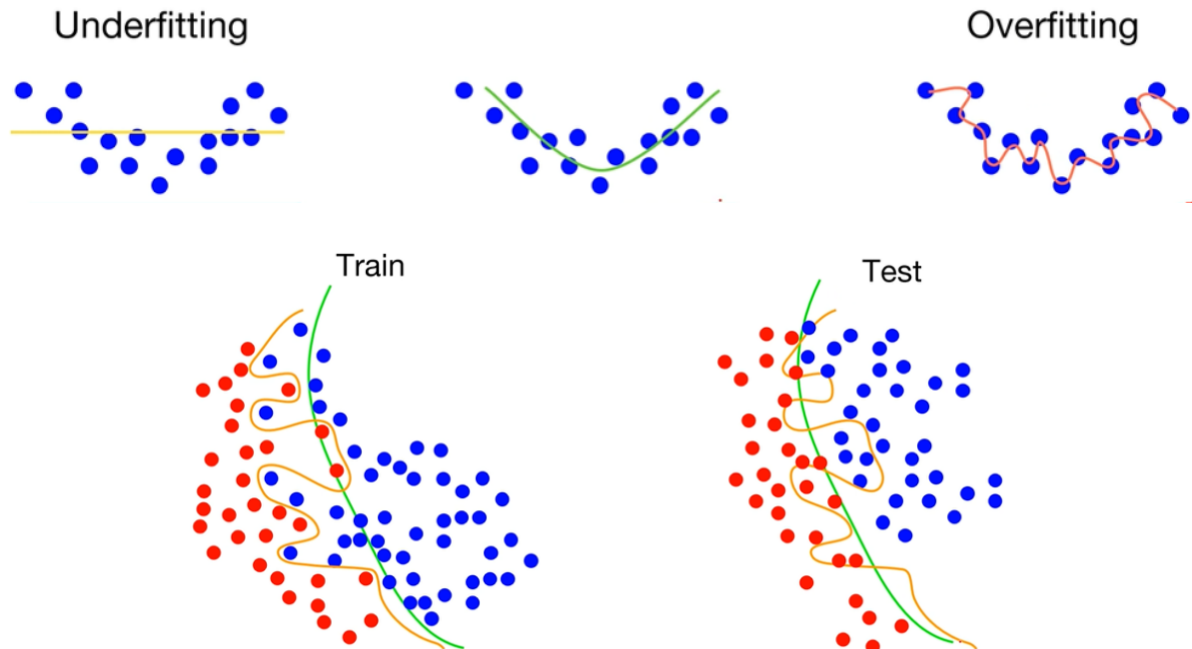


Dropout

Overfitting(과적합)

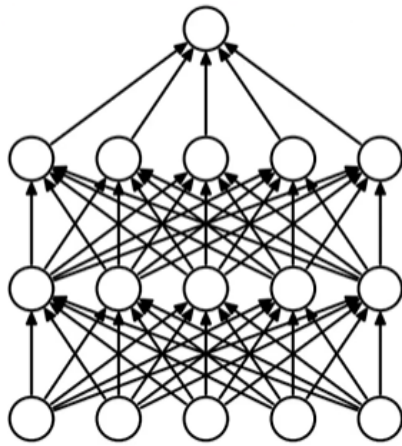


- Very high accuracy on the training dataset
- Poor accuracy on the test dataset

Solutions for overfitting

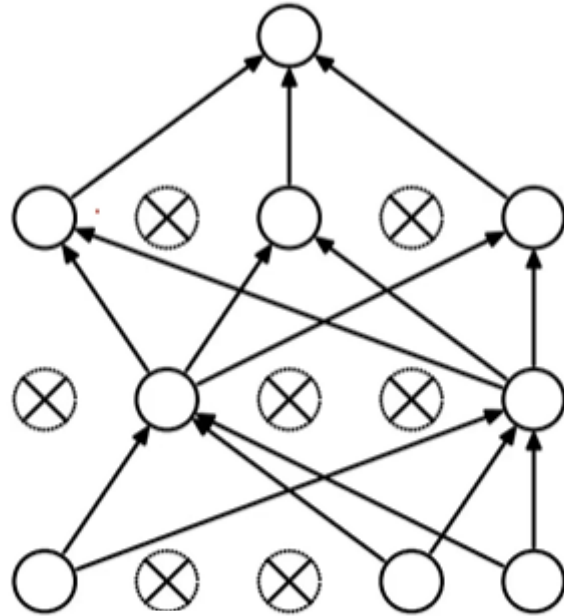
- More training data
- Reduce the number of features
- Regularization
- **Dropout**

Dropout



(a) Standard Neural Net

평범한 뉴런 네트워크



(b) Applying dropout.

Drop-out Rate = 0.5

[딥러닝] Drop-out(드롭아웃)은 무엇이고 5

본 포스팅에서는 딥러닝에 있어서 Generalization 기법 중 하나인 Drop-out(드롭아웃)에 대해 알아봅니다. 목차 1. 사전에 설정된 특정 비율에 따라 드롭아웃 적용 Drop-out 개념 2. Drop-out 사용이유 3. Mini-batch 학습 시

👉 <https://heytech.tistory.com/127>

<https://heytech.tistory.com/>

: 서로 연결된 연결망(layer)에서 0~1 사이의 확률로 뉴런을 제거(drop)하는 기법

→ 오버피팅 방지

→ 매번 다른 형태의 네트워크가 매번 다른 결과를 내고 그것을 학습

⇒ 네트워크 앙상블 효과를 얻을 수 있다.

```

# nn Layers
linear1 = torch.nn.Linear(784, 512, bias=True)
linear2 = torch.nn.Linear(512, 512, bias=True)
linear3 = torch.nn.Linear(512, 512, bias=True)
linear4 = torch.nn.Linear(512, 512, bias=True)
linear5 = torch.nn.Linear(512, 10, bias=True)
relu = torch.nn.ReLU()
dropout = torch.nn.Dropout(p=drop_prob)

# model
model = torch.nn.Sequential(linear1, relu, dropout,
                             linear2, relu, dropout,
                             linear3, relu, dropout,
                             linear4, relu, dropout,
                             linear5).to(device)

```

dropout 코드 추가, p=확률

```

...
total_batch = len(data_loader)
model.train()    # set the model to train mode (dropout=True)
for epoch in range(training_epochs):
    ...

    ...
    # Test model and check accuracy
    with torch.no_grad():
        model.eval()    # set the model to evaluation mode (dropout=False)
    ...

```

학습을 할 때의 방식 / 실제 test를 할 때는 모든 노드를 전부 사용해서 학습하게 됨

model.train() → 이 모델에 있는 dropout 적용하겠다

model.eval() → 이 모델에 있는 dropout을 지금부터 사용하지 않겠다

train/evaluation이라는 걸 선언을 하고 들어가야 정확한 결과를 얻을 수 있다