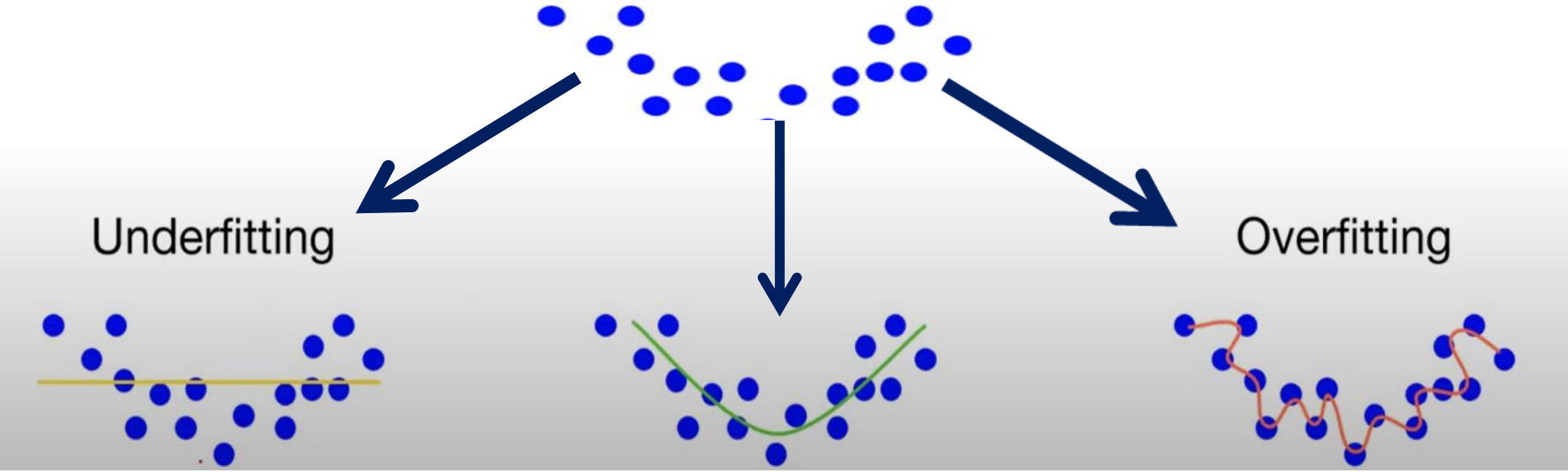
Overfitting 과 Dropout

B4 이정하

모델을 학습할 때 fitting 이란?

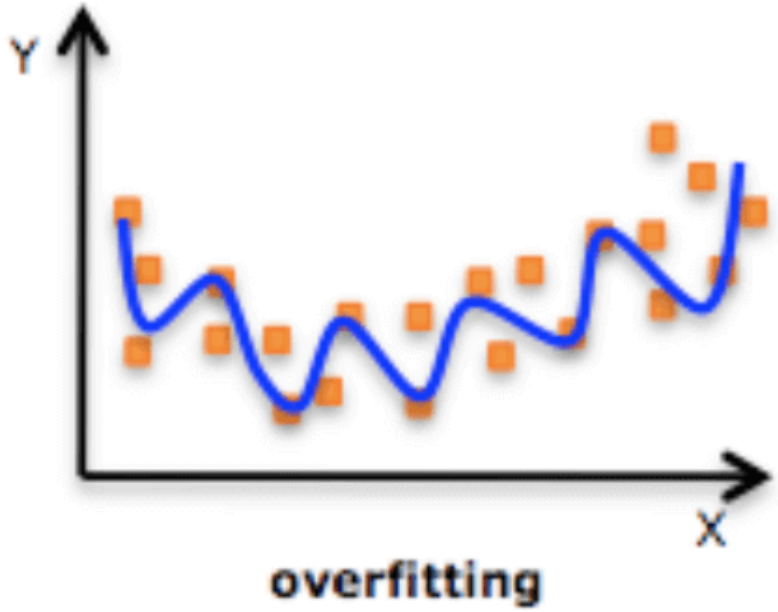


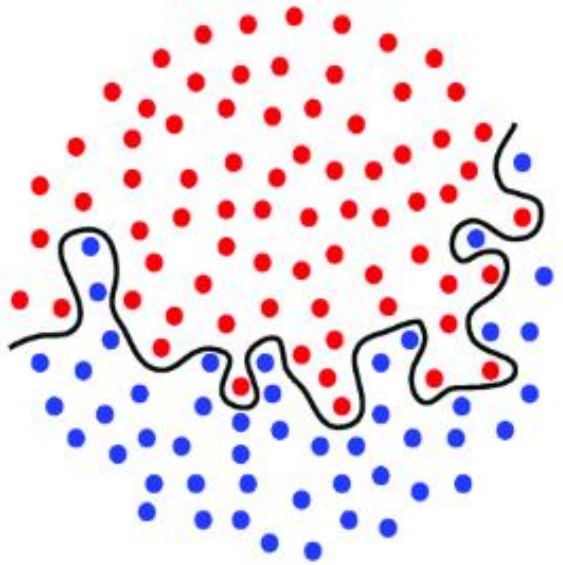
Fitting = 모델이 훈련 데이터에 얼마나 잘 맞는지의 문제


THE BEST WAY TO
EXPLAIN OVERFITTING
imgflip.com

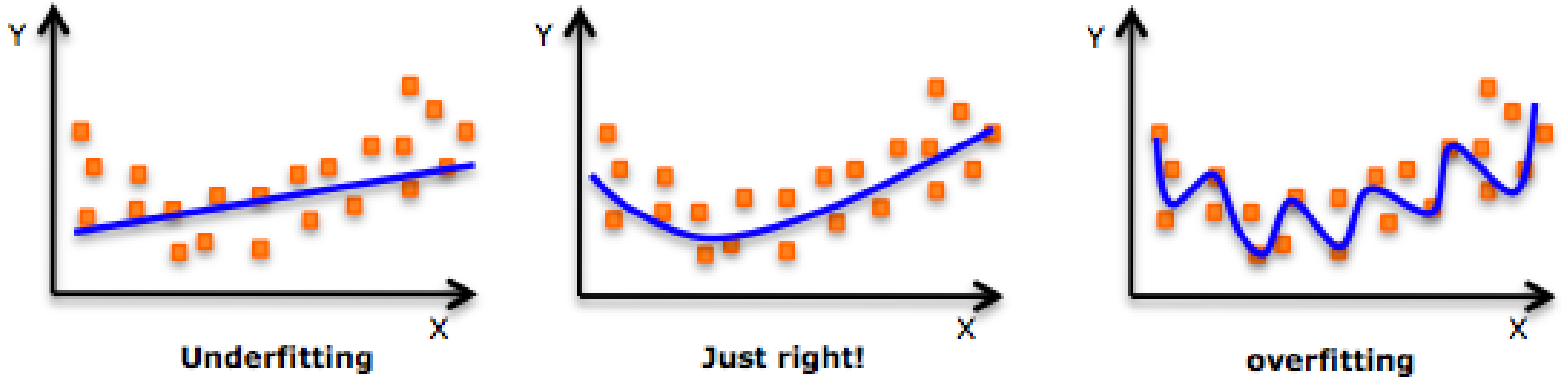
Overfitting [과대적합]

모델이 훈련데이터에 **너무 잘 맞아서** 일반성이 떨어진다는 뜻 !

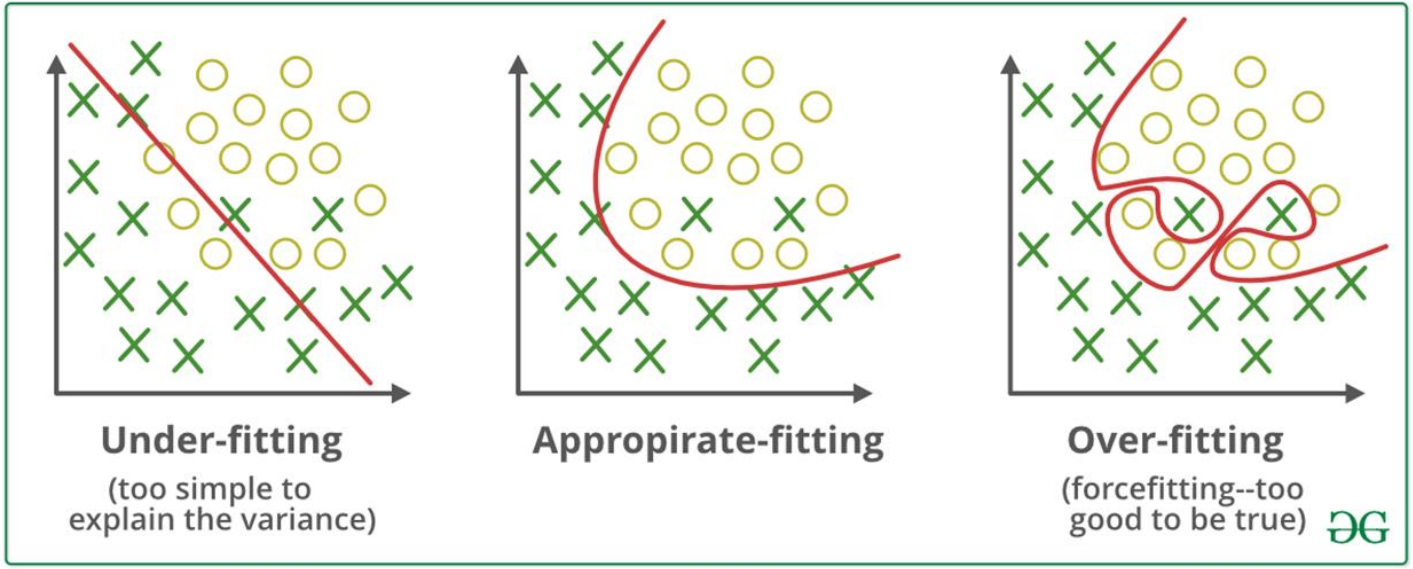




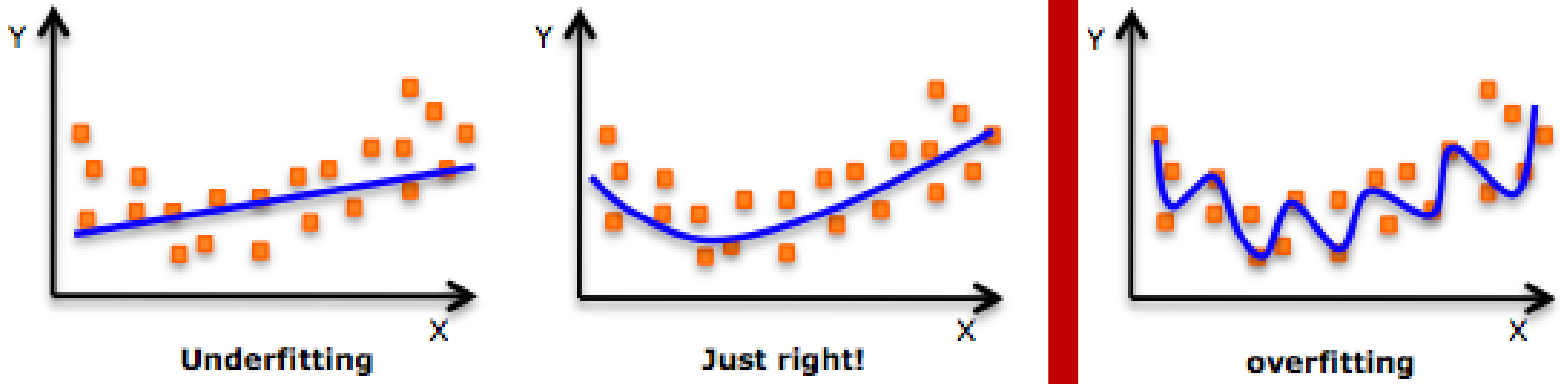
예측


Y
X
Underfitting
Y
X
Just right!
Y
X
overfitting

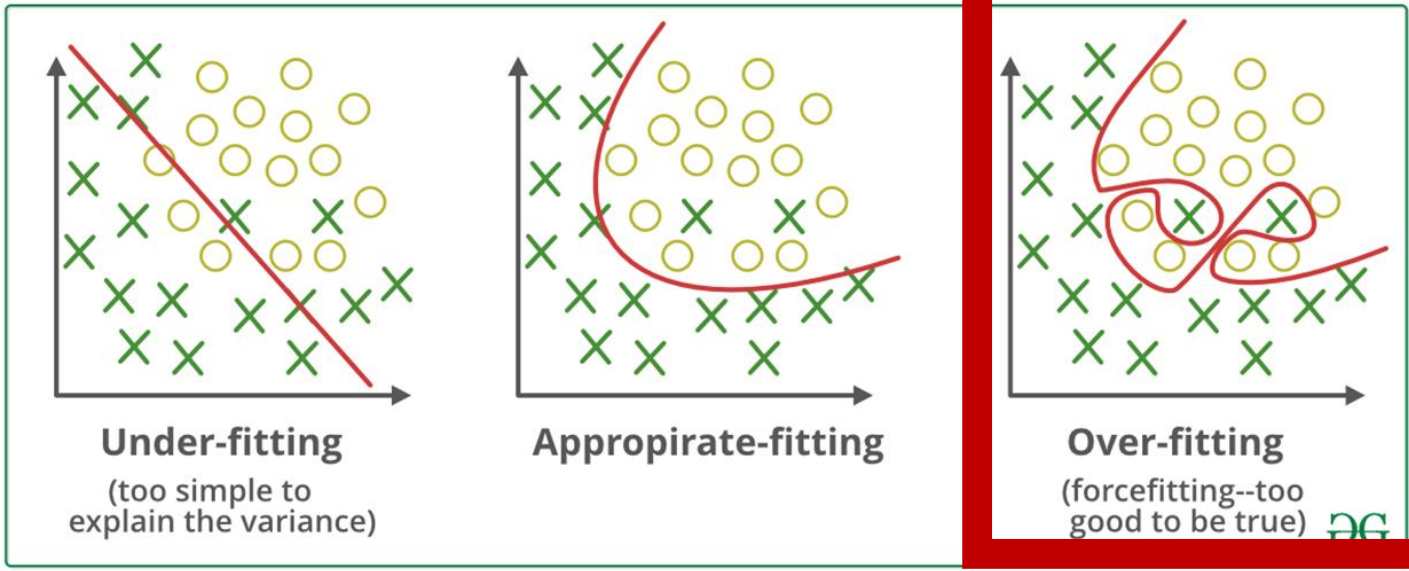
분류

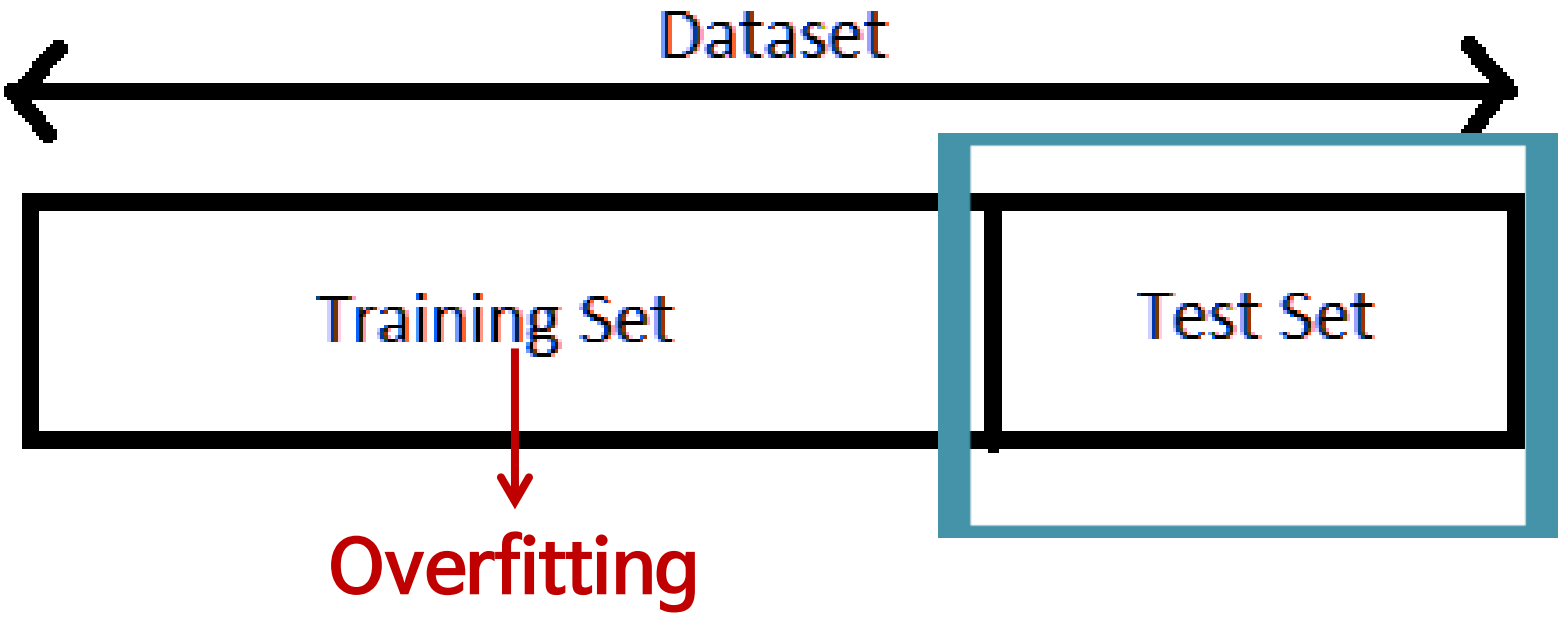

Under-fitting
(too simple to explain the variance)
Appropirate-fitting
Over-fitting
(forcefitting--too good to be true)

예측


Y
X
Underfitting
Y
X
Just right!
Y
X
overfitting

분류


Under-fitting
(too simple to explain the variance)
Appropirate-fitting
Over-fitting
(forcefitting--too good to be true)

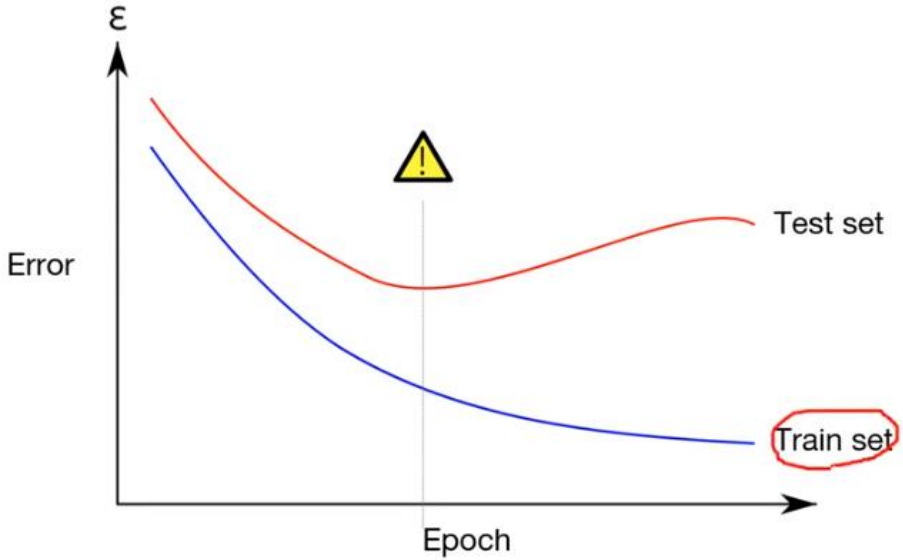


Overfitting 된 모델은
오히려 error 를 증가시킬 수 있다!

Overfitting 의 문제

High accuracy on the train set

VS Poor accuracy on the test set



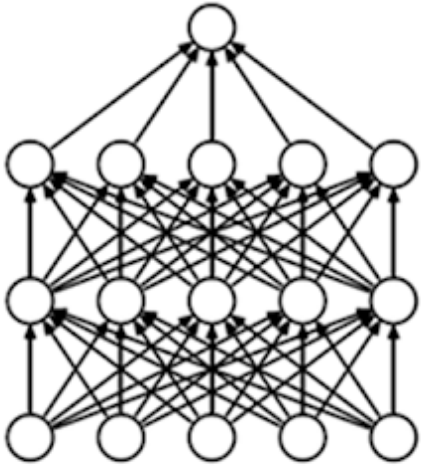
- Very high accuracy on the training dataset (e.g., 0.99)
- Poor accuracy on the test dataset (e.g., 0.85)

Overfitting 문제를 어떻게 해결할 수 있을까?

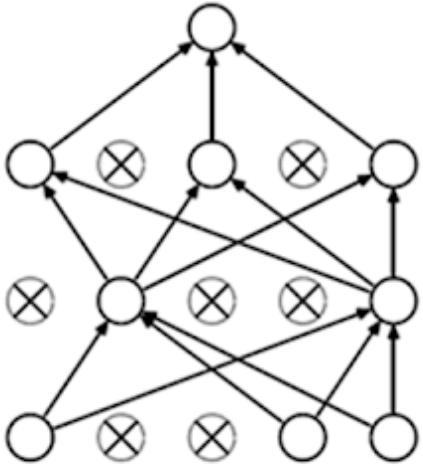
- More training data
- Reduce the number of features
- Regularization
- **Dropout !**

다양한 방법이 있지만 Dropout 방법을 알아보자.

Dropout



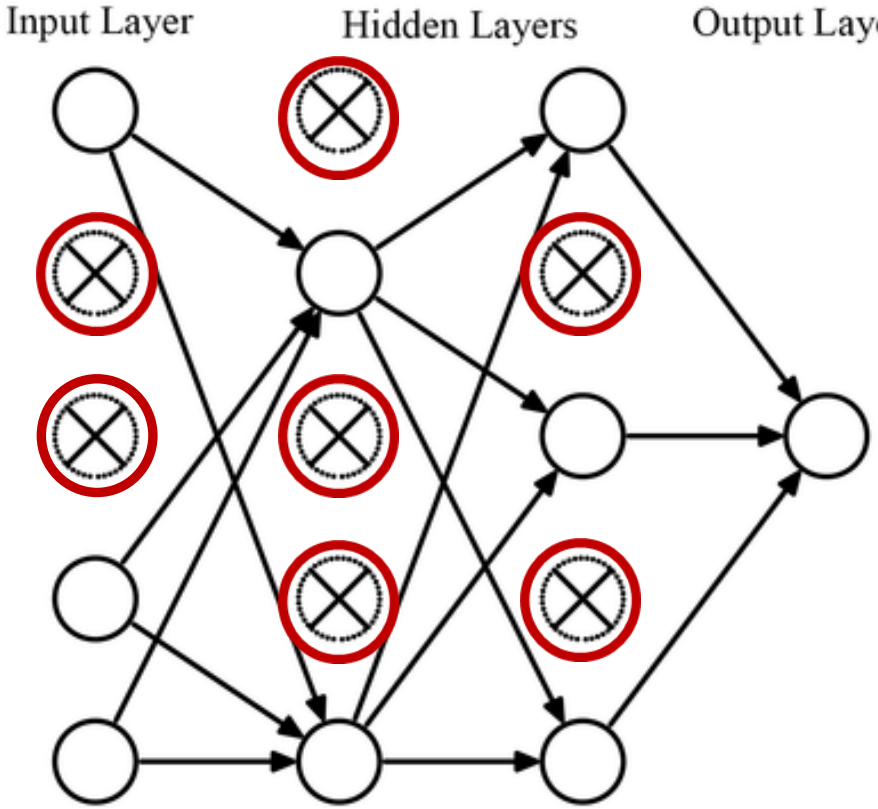
(a) Standard Neural Net



(b) After applying dropout.

사전에 설정된 어떤 확률에 따라
사용할 노드/사용하지 않을 노드를 고르고

사용할 노드의 가중치만을 이용해
그 다음 노드로 전파한다.



Dropout 의 기능

사용하지 않는 노드는
가중치를 전파하지 않으므로
Overfitting 을 방지할 수 있다

매 학습마다 사용 노드가 달라져서
네트워크 앙상블의 효과를 얻을 수 있다

Dropout 코드로 표현하기

```
# 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)
```

0.5 → 184 → 392

모델 선언

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

Dropout 시 주의할 점

Train Mode (Train Set)

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

= 모델에 있는 Dropout 모두 사용하겠다!

Evaluation Mode (Test Set)

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

= 모델에 있는 Dropout 사용하지 않겠다!
(evaluation 을 위해 전체 노드를 활용하겠다는 의미)

Train 데이터로 학습을 할 때에는 Dropout 으로 노드를 drop 할 수 있지만,
Test 데이터를 사용할 때에는 모든 노드를 전부 사용해야 한다.