

Type something...

01) Weight Initialization

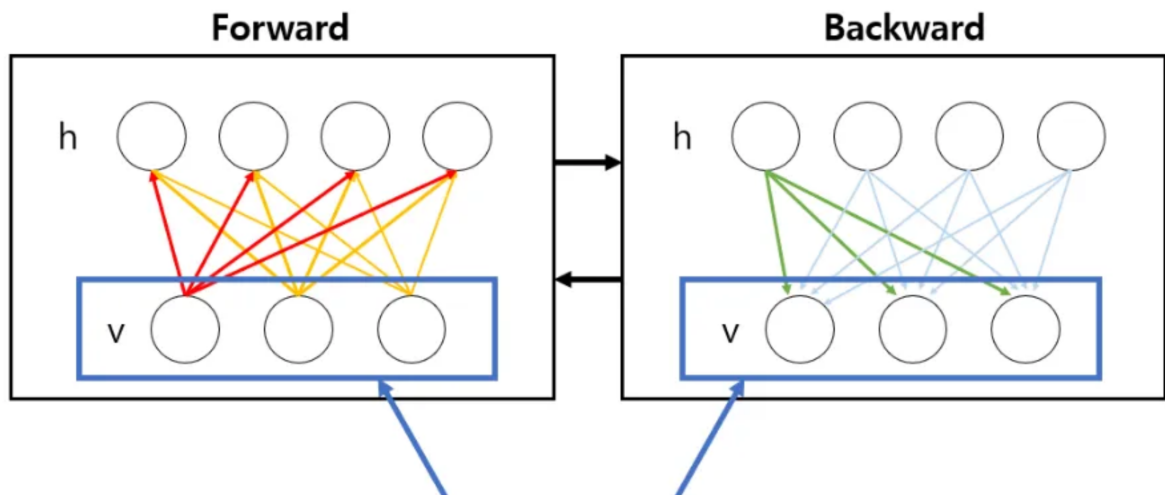
01_Restricted Boltzmann Machine(RBM)

02_Xavier/He initialization

01) Weight Initialization

01_Restricted Boltzmann Machine(RBM)

RBM이 어떤 주어진 데이터를 잘 학습했다면 sampling을 통해 얻은 visible layer의 데이터가 원래 데이터와 거의 같아야 한다.



이 두 값이 같아질 때 까지 Forward <--> Backward
반복하면서 weight, bias 조정

그림 4. RBM을 학습시킨다는 것의 의미: sampled visible layer의 값이 원래의 visible layer의 값과 같아진다.

- Forward와 Backward를 반복하면서 weight, bias를 조정하는 방식이다. 수식으로 설명이 되어 있으나 너무 복잡하여 생략하겠다.

02_Xavier/He initialization

Xavier / He initialization

- Xavier Normal initialization

$$W \sim N(0, Var(W))$$

$$Var(W) = \sqrt{\frac{2}{n_{in} + n_{out}}}$$

- Xavier Uniform initialization

$$W \sim U(-\sqrt{\frac{6}{n_{in} + n_{out}}}, +\sqrt{\frac{6}{n_{in} + n_{out}}})$$

- He Normal initialization

$$W \sim N(0, Var(W))$$

$$Var(W) = \sqrt{\frac{2}{n_{in}}}$$

- He Uniform initialization

$$W \sim U(-\sqrt{\frac{6}{n_{in}}}, +\sqrt{\frac{6}{n_{in}}})$$

- 앞의 방식이 매우 복잡하여 적용하는 것이 쉽지 않았다.

- <https://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf> 그러던 중, Understanding the difficulty of training deep feedforward neural networks라는 논문이 나오게 된다.
- 여기에 의하면 입력과 출력 또는 입력의 사이즈에 의해 초기화를 진행시키면 아주 좋은 초기화 방식이 된다는 거다.
- Xavier Normal initialization

Code: mnist_nn_xavier

...

```
# nn layers
linear1 = torch.nn.Linear(784, 256, bias=True)
linear2 = torch.nn.Linear(256, 256, bias=True)
linear3 = torch.nn.Linear(256, 10, bias=True)
relu = torch.nn.ReLU()

# xavier initialization
torch.nn.init.xavier_uniform_(linear1.weight)
torch.nn.init.xavier_uniform_(linear2.weight)
torch.nn.init.xavier_uniform_(linear3.weight)

Parameter containing:
tensor([[-0.0215, -0.0894, 0.0598, ..., 0.0200, 0.0203, 0.1212],
        [ 0.0078, 0.1378, 0.0920, ..., 0.0975, 0.1458, -0.0302],
        [ 0.1270, -0.1296, 0.1049, ..., 0.0124, 0.1173, -0.0901],
        ...,
        [ 0.0661, -0.1025, 0.1437, ..., 0.0784, 0.0977, -0.0396],
        [ 0.0430, -0.1274, -0.0134, ..., -0.0582, 0.1201, 0.1479],
        [-0.1433, 0.0200, -0.0568, ..., 0.0787, 0.0428, -0.0036]],
        requires_grad=True)
```

```
Epoch: 0001 cost = 0.249897048
Epoch: 0002 cost = 0.094330102
Epoch: 0003 cost = 0.061055195
Epoch: 0004 cost = 0.042816643
Epoch: 0005 cost = 0.032796543
Epoch: 0006 cost = 0.024419624
Epoch: 0007 cost = 0.020511184
Epoch: 0008 cost = 0.018132176
Epoch: 0009 cost = 0.015536907
Epoch: 0010 cost = 0.016846467
Epoch: 0011 cost = 0.012203062
Epoch: 0012 cost = 0.012871196
Epoch: 0013 cost = 0.011348661
Epoch: 0014 cost = 0.010990168
Epoch: 0015 cost = 0.006201488
Learning finished
Accuracy: 0.9804999828338623
```

- 이제부터 초기화는 `torch.nn.init.xavier_uniform_(linear1.weight)` 로 초기화 해주자.