Batch Normalization

LATEST SUBMISSION GRADE

100%

1. What wrong with the following lines of code: **1 / 1 point**

```
class NetBatchNorm(nn.Module):

    # Constructor
    def __init__(self, in_size, n_hidden1, n_hidden2, out_size)
        super(NetBatchNorm, self).__init__()
        self.linear1 = nn.Linear(in_size, n_hidden1)
        self.linear2 = nn.Linear(n_hidden1, n_hidden2)
        self.linear3 = nn.Linear(n_hidden2, out_size)


    # Prediction
    def forward(self, x):
        x = self.bn1(torch.sigmoid(self.linear1(x)))
        x = self.bn2(torch.sigmoid(self.linear2(x)))
        x = self.linear3(x)
        return x
```

- ○ nothing
- ◉ you need to crate the BatchNorm objects bn1 and bn2

✓ **Correct**
correct

2. Consider the following Batch Norm constructor , what is the parameter *n_hidden1* represent **1 / 1 point**

```
nn.BatchNorm1d(n_hidden1)
```

- ◉ the size of the input
- ○ the input activation

✓ **Correct**
correct