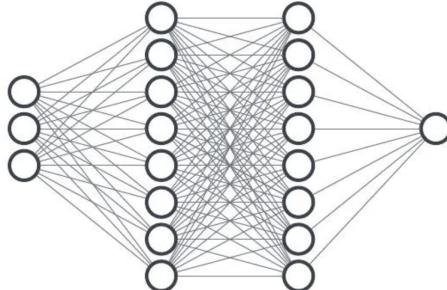


Practice Questions: Neural Networks

1. Draw the neural network diagrams and say the number of parameters for:
 - (a) a linear regression model with 5 predictors
 - (b) a logistic regression model with 2 predictors
 - (c) a multiclass logistic regression model with 3 predictors and 5 classes.
2. Suppose I had a neural network for a regression problem with 2 hidden layers. Suppose further that I let all activation functions equal the identity function, $h(z) = z$. What model does this collapse down to? Can you prove your claim?
3. What are the **final** activation functions for the following problems:
 - (a) A three hidden layer neural network for binary classification.
 - (b) A two hidden layer neural network for multiclass classification.
 - (c) A two hidden layer neural network for regression.
4. Write the equations and PyTorch code for each of the following neural networks. Also specify the number of parameters in each model.
 - (a) a neural network with two hidden layers which accepts 10 inputs, has 4 features at each hidden layer, and 3 output classes, and sigmoid activation functions
 - (b) the neural network described by this diagram, where the activation functions for the hidden layers are ReLUs and the problem is for a regression task



5. Draw the NN diagram for each of the following neural networks. Also specify the number of parameters in each model.

- (a) a neural network for binary classification with three hidden layers which takes 4 inputs, has 2 features at each hidden layer, and ReLU activation functions
- (b) the neural network described by this code:

```
model = torch.nn.Sequential(  
    torch.nn.Linear(4,3),  
    torch.nn.ReLU(),  
    torch.nn.Linear(3,2),  
    torch.nn.ReLU(),  
    torch.nn.Linear(2,2),  
    torch.nn.ReLU(),  
    torch.nn.Linear(2,2),  
    torch.nn.Sigmoid()  
)
```

6. Suppose you fit a neural network model for a regression task which takes 10 predictors as inputs and has 10 hidden layers each with 10 neurons. Your training data consists of 150 observations.
- (a) How many parameters does this model have?
 - (b) When you apply your fitted model to a validation set, you see a validation loss much higher than your training loss. What does this make you suspicious of?
 - (c) Why might this model be encountering this problem?
 - (d) What modifications could you make to your architecture to resolve the issue?