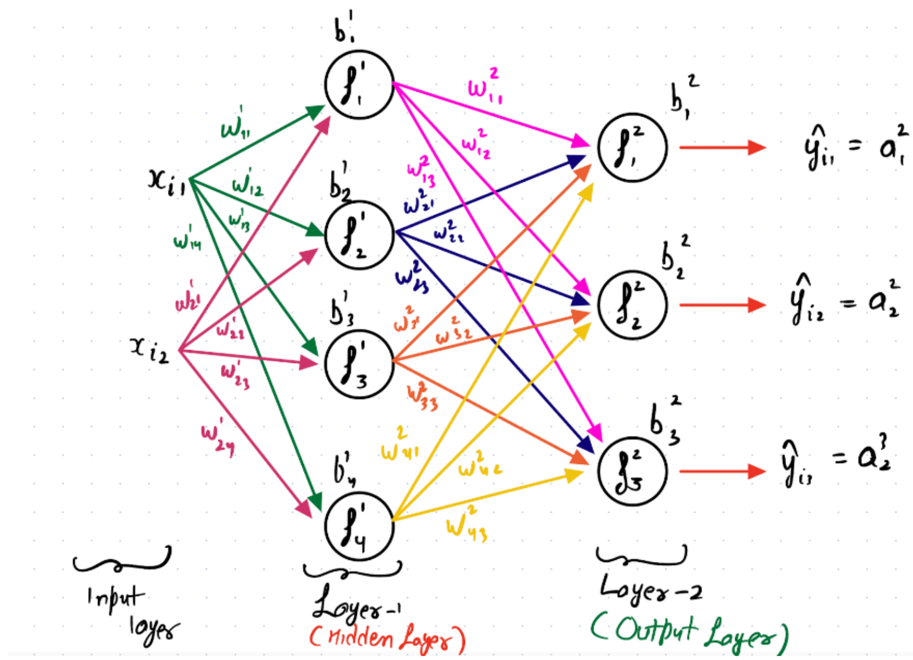
Neural network 3

- **NN-3: Backpropagation and Activation functions**
  What are MLPs?
  If we wish to get a more complex decision boundary, we need to add another layer of neurons in the model. This is known as the **hidden layer**.
    - These models are known as MLPs (Multi Layer Perceptrons). Or N-layered NN
    - They are based on the idea of function composition.
    - We can have as many hidden layers as we want, however, the greater the number, the higher the risk of overfitting.
    - The activation of hidden layers also needs to always be **non-linear,** otherwise, we will not be able to get complex features.

  A 2-layer model looks like:-



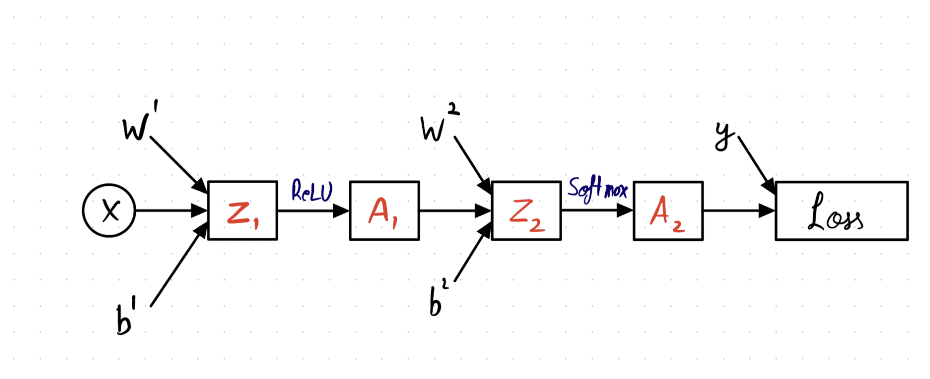
  Since there are multiple layers, we modify the notation to cater to it
    - Weights: $\mathbf{W^{L}_{ij}}$
    - Bias: $\mathbf{b^{L}_{i}}$

  Note:-
    - Weights in layer-1 will be stored as a 2x4 matrix: $W^{1}_{2\,x\,4}$
    - Biases in layer-1 will be stored as a 1x4 matrix: $b^{1}_{1\,x\,4}$
    - Weights in layer-2 will be stored as a 4x3 matrix: $W^{2}_{4\,x\,3}$

- Biases in layer-2 will be stored as a 1x3 matrix: $b^{2}_{1\,x\,3}$

Why do we need to add a hidden layer to increase complexity?
The idea is that Stacking a non-linearity over a linear function, and repeating the process helps create complex features

What does forward propagation look like for this NN?
Let h -> no of neurons in the hidden layer.
m -> no of training examples
d -> no of features
n -> no of classes/neurons in the output layer



Forward propagation:-

$$Z^{1}_{m\,x\,h} = X_{m\,x\,d} . W^{1}_{d\,x\,h} + b^{1}_{1\,x\,h}$$
$$A^{1}_{m\,x\,h} = f^{1}(Z^{1}_{m\,x\,h})$$
$$Z^{2}_{m\,x\,n} = A^{1}_{m\,x\,h} . W^{2}_{h\,x\,n} + b^{2}_{1\,x\,n}$$
$$A^{2}_{m\,x\,n} = f^{2}(Z^{2}_{m\,x\,n})$$