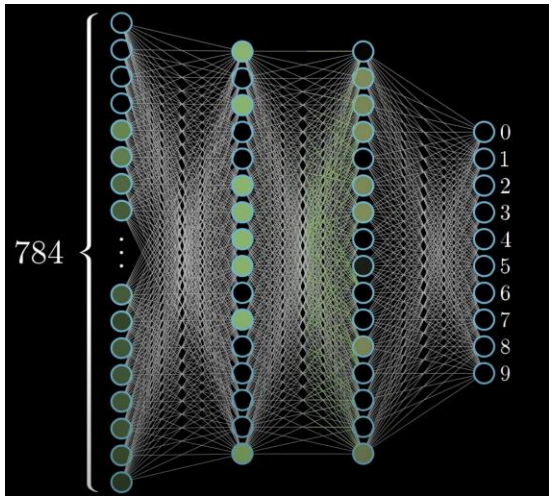
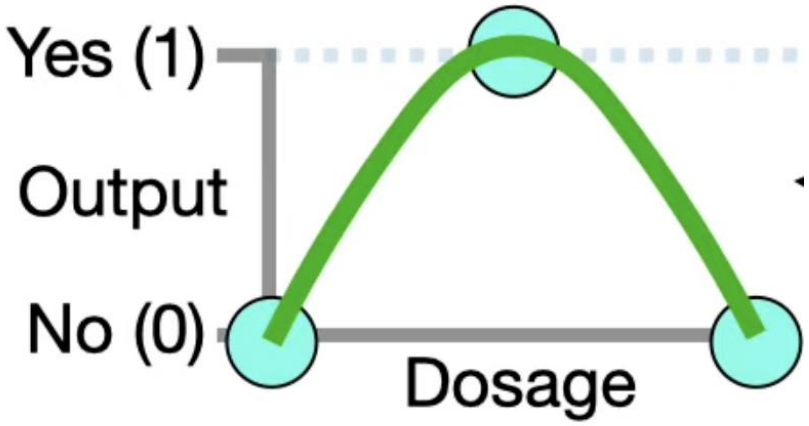
Link to handout: https://bit.ly/NN-DFS-Backprop



PA03: TRAINING NEURAL NETWORKS USING

DFS-BASED BACKPROPAGATION

Neural Net to predict effectiveness of a drug

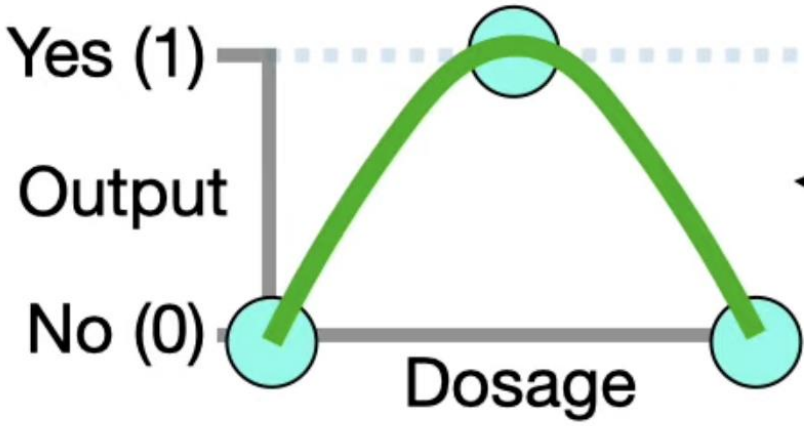


Draw the neural network that was used to predict whether a drug's dosage is effective against a virus.

How many parameters does the NN have?

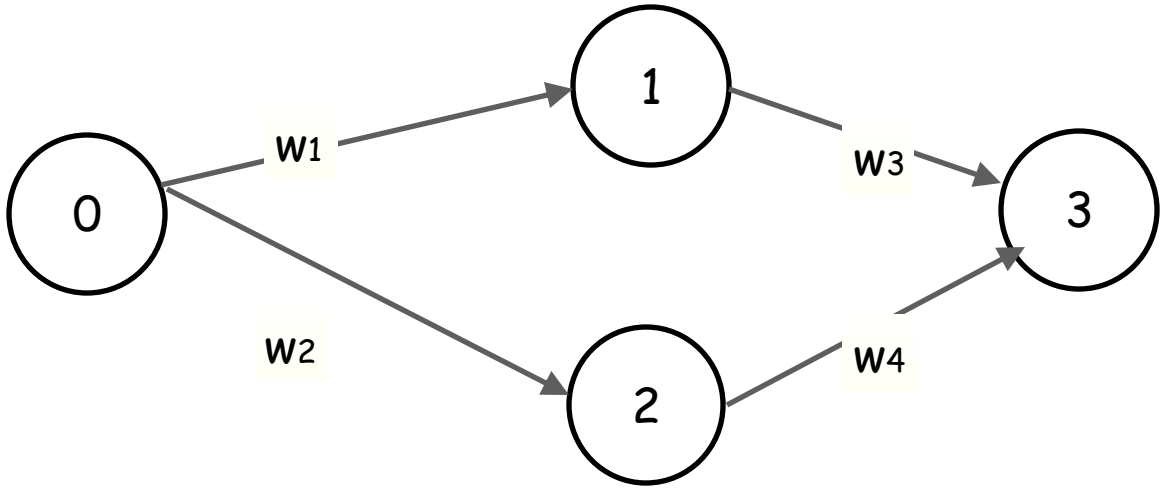
Pre-lecture video on Backpropagation in NNs: https://youtu.be/IN2XmBhILt4?si=bnDft-3T4DQ2iO9X

What is “training” the neural net?



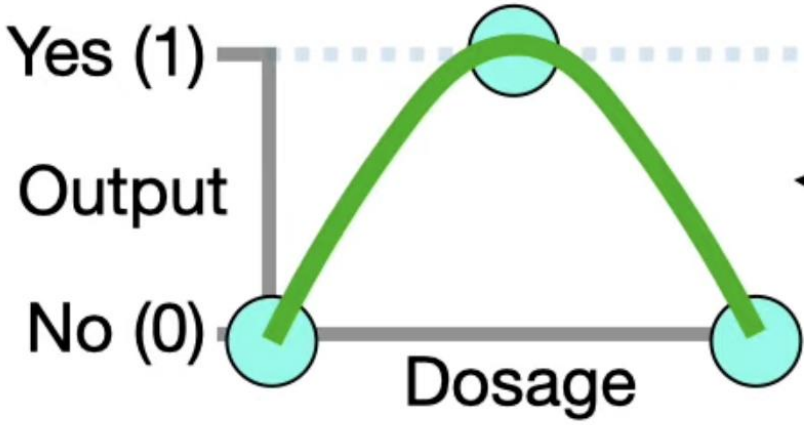
Training a neural net means adjusting its parameters (weights and biases) so it learns to map inputs to the correct outputs.

Show the neural net different example inputs and outputs



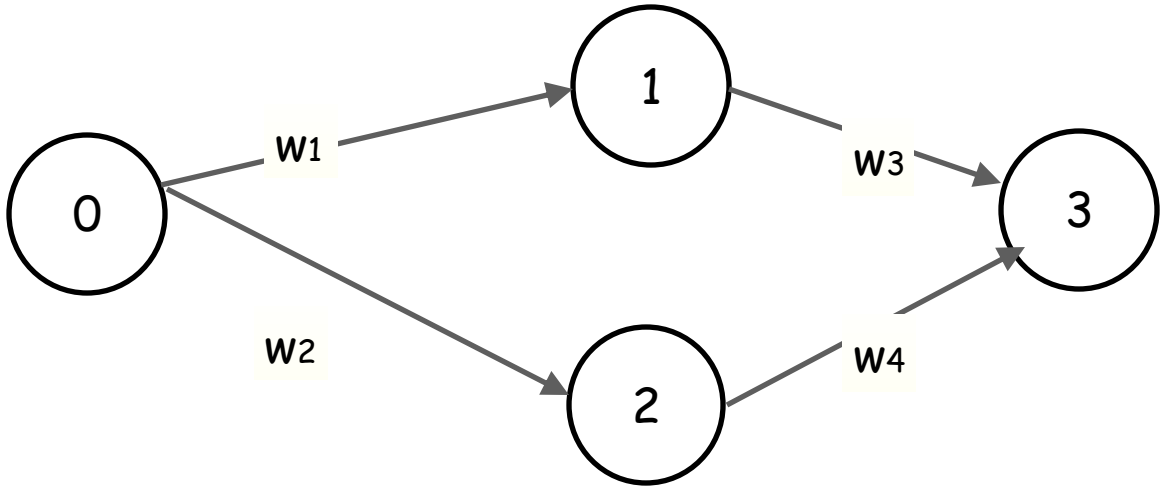
Pre-lecture video on Backpropagation in NNs: https://youtu.be/IN2XmBhILt4?si=bnDft-3T4DQ2iO9X

Backpropagation = smart feedback



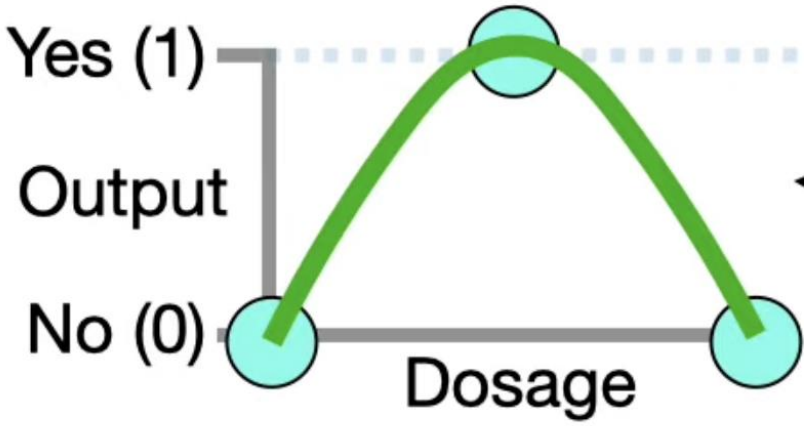
Backprop: Guiding feedback that tells each weight and bias how to slightly nudge to reduce the error.

In PA03, that feedback is abstracted into a call to contribute(node, y, p)



PA03 Starter: https://github.com/ucsb-cs24-s25/STARTER-pa03/blob/main/NeuralNetwork.cpp

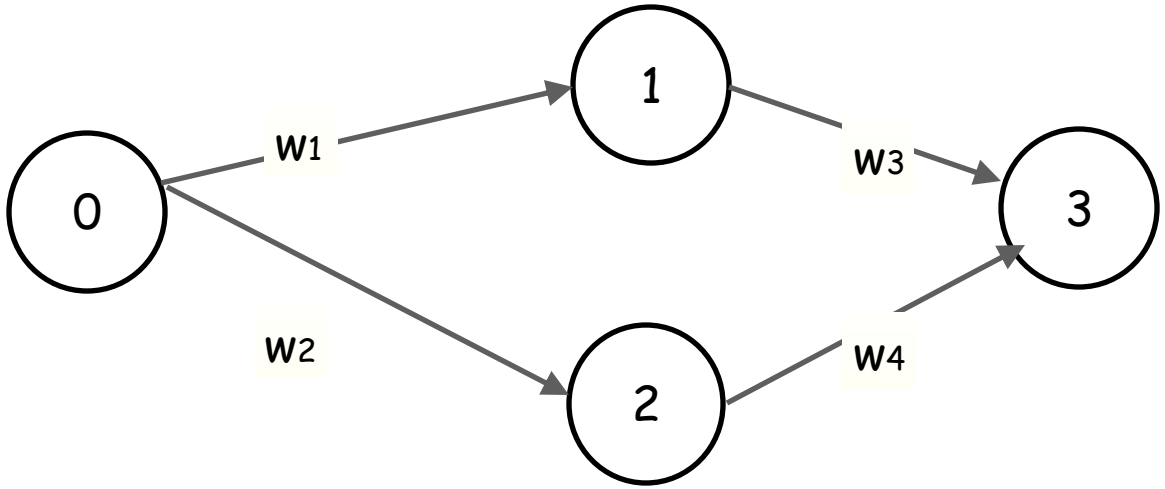
Backpropagation: PA03 contribute()



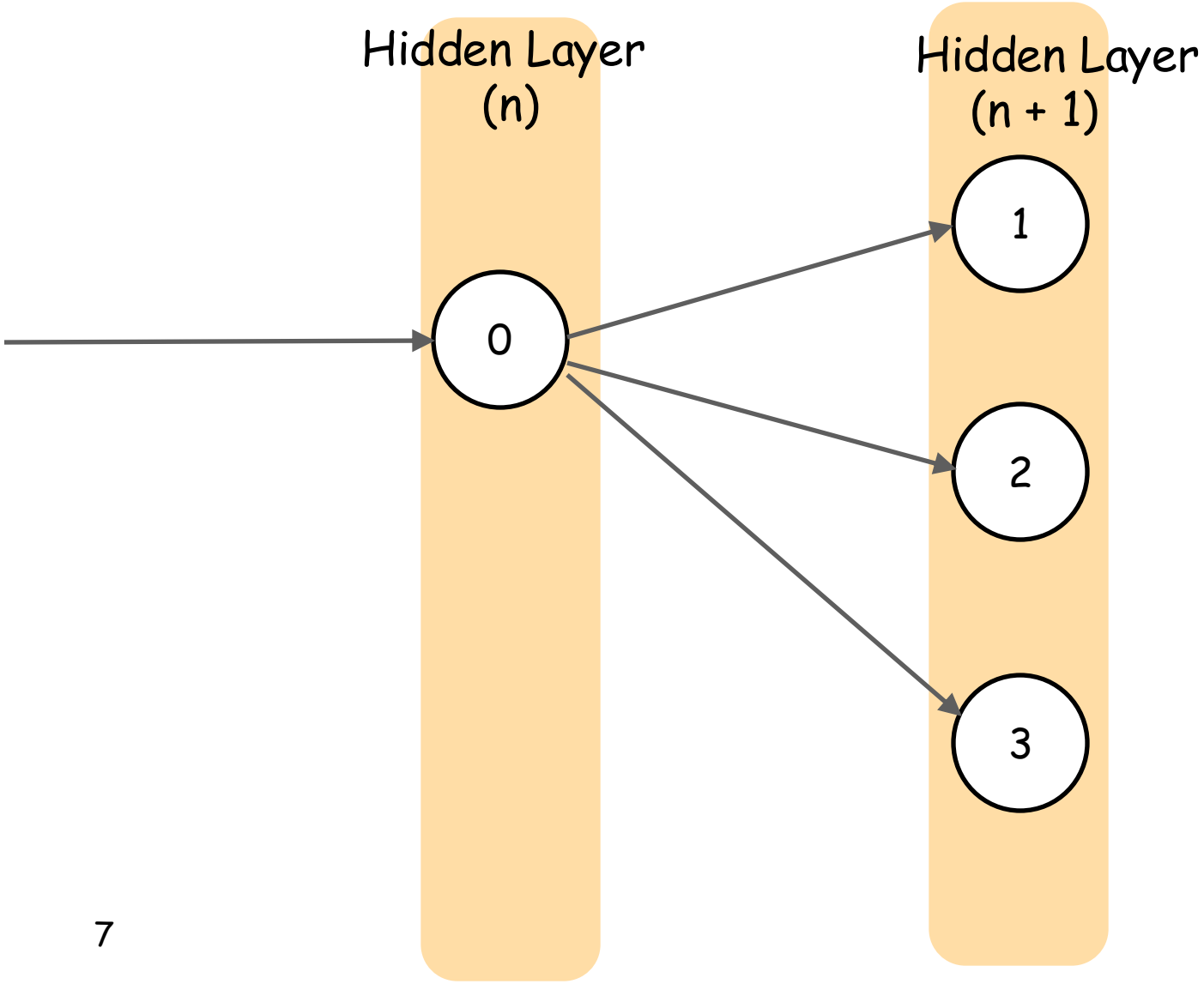
Node 3's contribution = Nudge to node 3's bias

Contributions in one layer are used to compute the contributions in the previous layer

Node 3's outgoing contribution is used to compute the contributions of which nodes?



Backpropagation: Order of operations for one node



Backprop using DFS (you need to implement)

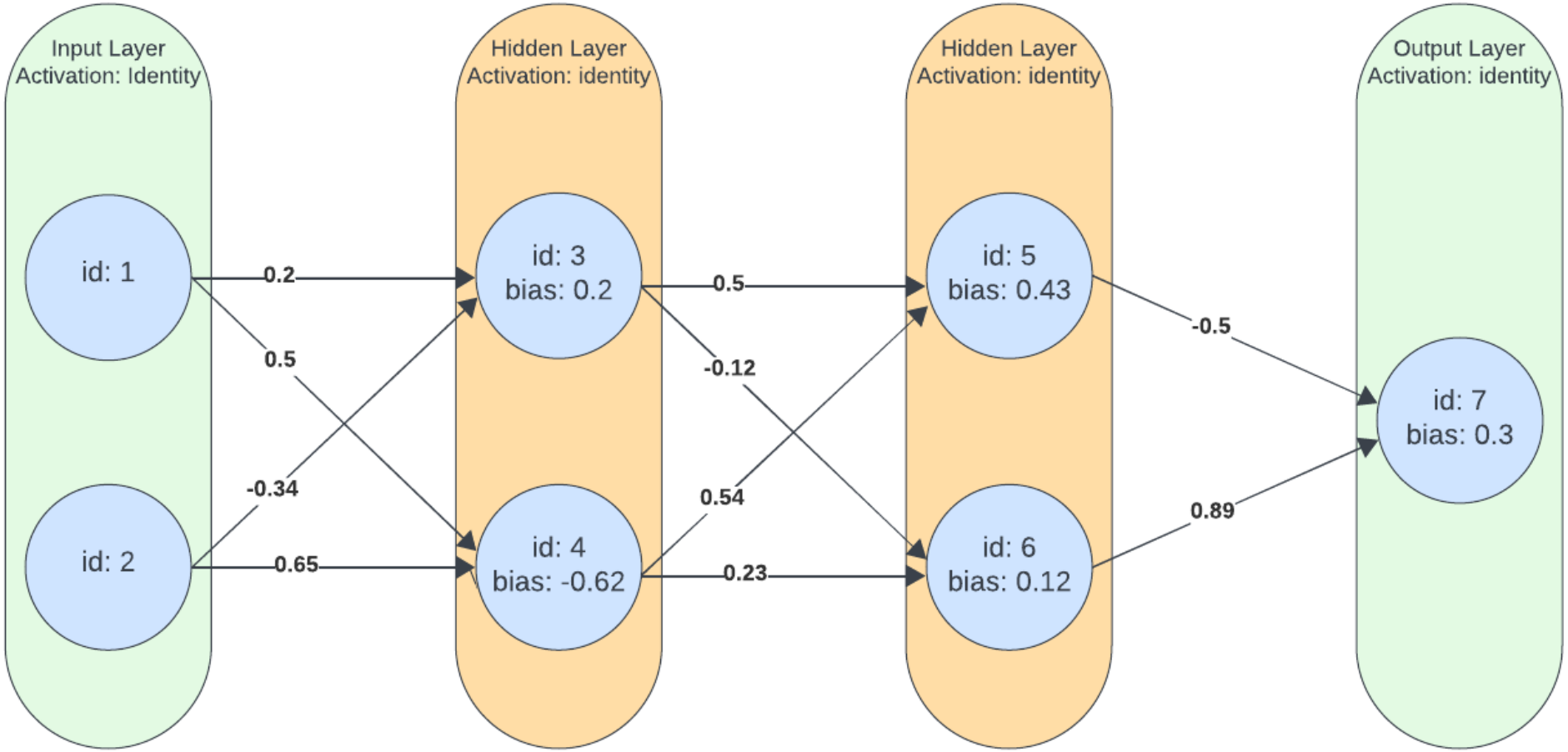
```
contribute(int nodeId,
           const double& y,
           const double& p)
```

Helper visit functions that do the math for you:

```
visitContributeNeighbor(
  Connection& c,
  double& incomingContribution,
  double& outgoingContribution)

visitContributeNode(
  int vId,
  double& outgoingContribution)
```

Activity: Trace Data Flow for Backpropagation (contribute)



Trace the order in which **contribute(...)**, **visitContributeNode (…)**, and **visitContributeNeighbor(…)** are called. For each node, write down: when it's visited, what contributions it receives, and when those are passed on to others.