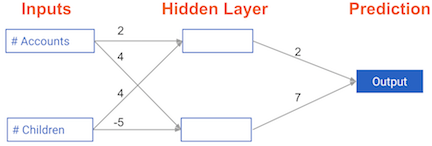
**Lecture1**

**Coding the forward propagation algorithm**

In this exercise, you'll write code to do forward propagation (prediction) for your first neural network:



Each data point is a customer. The first input is how many accounts they have, and the second input is how many children they have. The model will predict how many transactions the user makes in the next year. You will use this data throughout the first 2 chapters of this course.

The input data has been pre-loaded as input\_data, and the weights are available in a dictionary called weights. The array of weights for the first node in the hidden layer are in weights['node\_0'], and the array of weights for the second node in the hidden layer are in weights['node\_1'].

The weights feeding into the output node are available in weights['output'].

NumPy will be pre-imported for you as np in all exercises.

* Calculate the value in node 0 by multiplying input\_data by its weights weights['node\_0'] and computing their sum. This is the 1st node in the hidden layer.
* Calculate the value in node 1 using input\_data and weights['node\_1']. This is the 2nd node in the hidden layer.
* Put the hidden layer values into an array. This has been done for you.
* Generate the prediction by multiplying hidden\_layer\_outputs by weights['output'] and computing their sum.