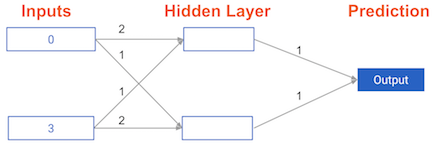
[DataCamp](https://www.datacamp.com/)

[Course Outline](https://campus.datacamp.com/courses/deep-learning-in-python/optimizing-a-neural-network-with-backward-propagation?ex=4)

**Coding how weight changes affect accuracy**

Now you'll get to change weights in a real network and see how they affect model accuracy!

Have a look at the following neural network: 

Its weights have been pre-loaded as weights\_0. Your task in this exercise is to update a **single** weight in weights\_0 to create weights\_1, which gives a perfect prediction (in which the predicted value is equal to target\_actual: 3).

Use a pen and paper if necessary to experiment with different combinations. You'll use the predict\_with\_network() function, which takes an array of data as the first argument, and weights as the second argument.