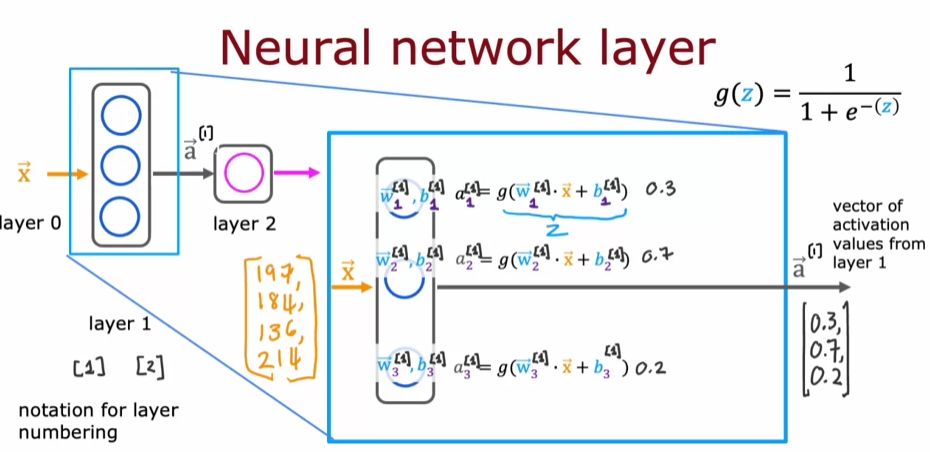
**NEURAL NETWORK MODEL**

**NEURAL NETWORK LAYER**

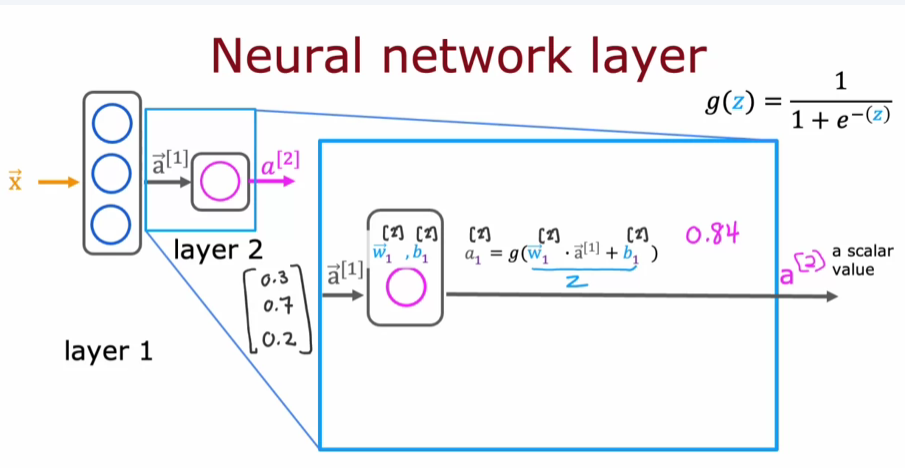
**Understanding layers in neural networks**

* **A neural network typically consists of multiple layers, with each layer containing neurons that process input data. The first layer is the input layer, followed by hidden layers, and finally the output layer.**
* **Each neuron in a hidden layer performs logistic regression, taking input features and producing an activation value that indicates the likelihood of a certain outcome.**

****

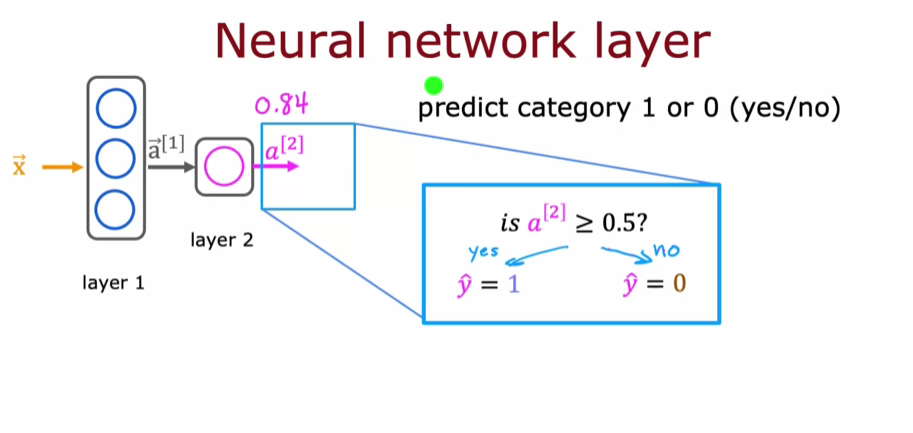
**Computation in Hidden and Output Layers**

* **In the hidden layer, each neuron computes an activation value using parameters (weights and biases) and the logistic function, resulting in a vector of activation values that is passed to the next layer.**
* **The output layer takes the activation values from the hidden layer and computes a final output, which can be a probability or a binary prediction based on a threshold.**

****

**Notation and Layer Indexing**

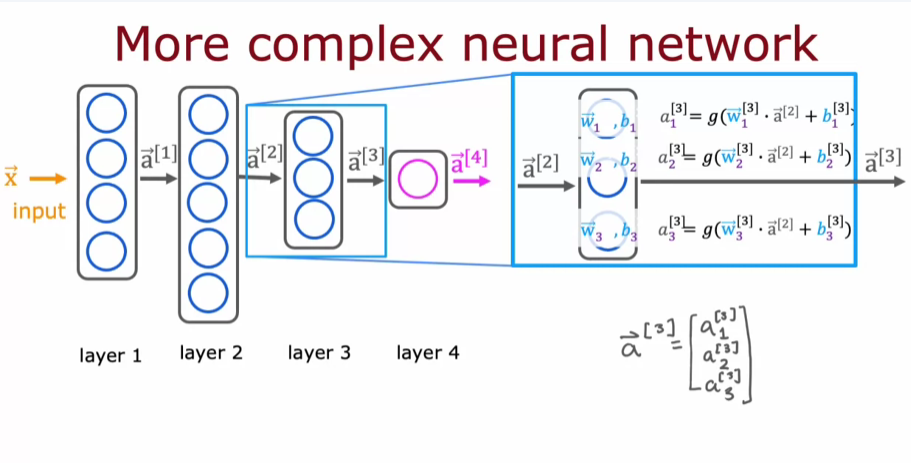
* **To keep track of different layers and their parameters, a notation system using superscripts and subscripts is introduced. For example,  denotes the output of layer 1, while  refers to the weights of layer 2.**
* **This notation helps in organizing and understanding the computations as the data flows through the layers of the neural network.**

****

**MORE COMPLEX NEURAL NETWORKS**

**Understanding Neural Network Layers**

* **A neural network consists of multiple layers, including input, hidden, and output layers. In this example, there are four layers, with three hidden layers and one output layer.**
* **Each layer processes input vectors and produces output vectors, with the computations involving weights and biases associated with each neuron.**

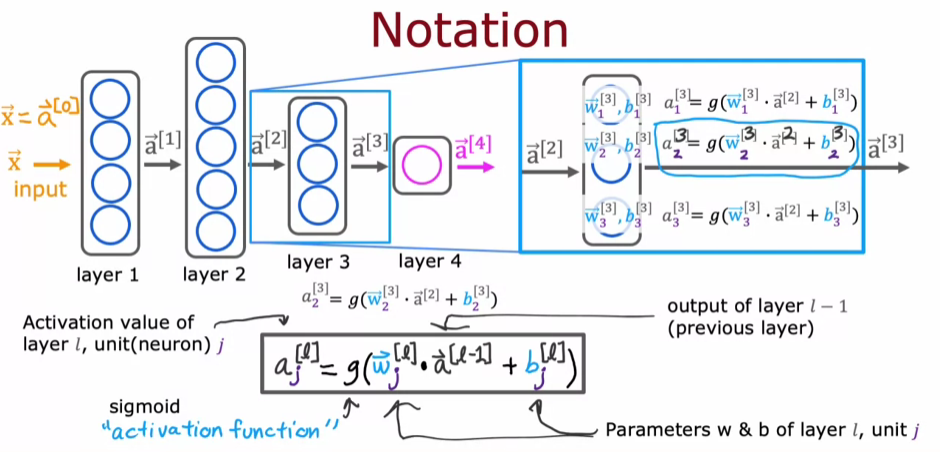
****

**Computations in Layer 3**

* **Layer 3 takes the output from Layer 2 as input and computes its activations using the sigmoid function applied to the weighted sum of inputs plus biases.**
* **The notation used helps clarify which parameters and activations belong to which layer, making it easier to understand the flow of data through the network.**

**General Form of Activation Computation**

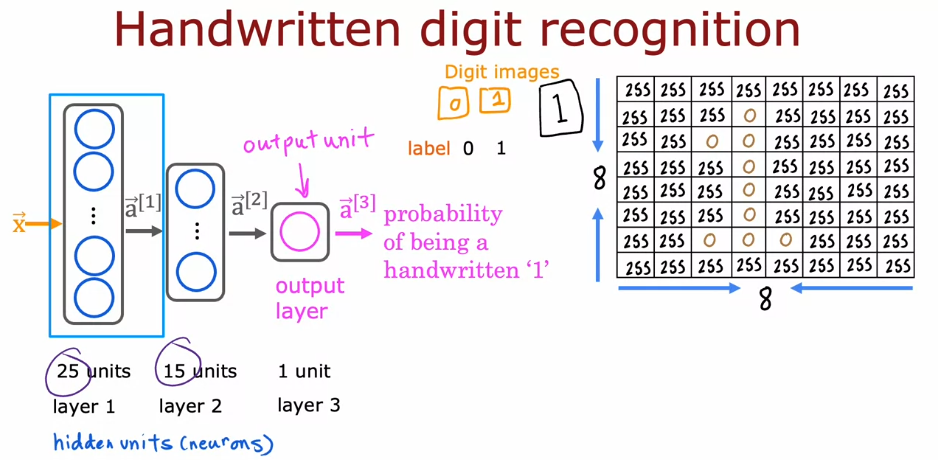
* **The general formula for computing activations in any layer involves the weights, biases, and the activations from the previous layer.**
* **The activation function, often the sigmoid function, is crucial for determining the output of each neuron, and this concept will be expanded upon in future lessons.**

****

**INFERENCE: MAKING PREDICTIONS (FORWARD PROPAGATION)**

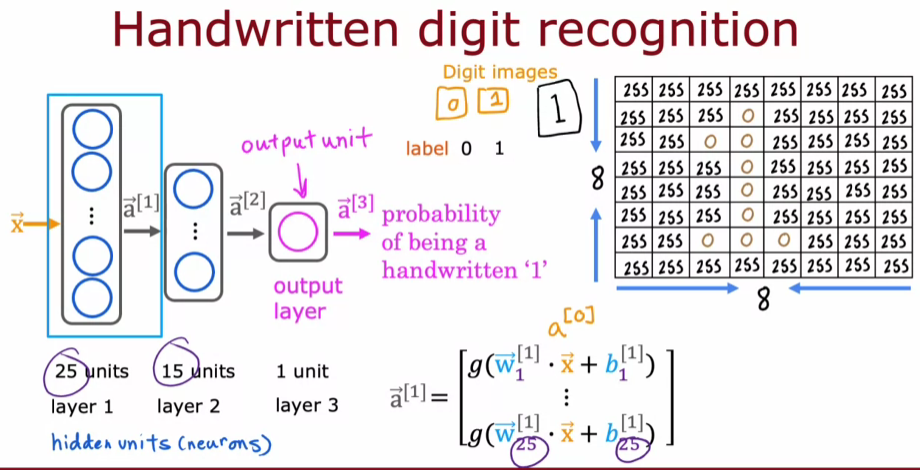
**Understanding Forward Propagation**

* **Forward propagation is the process of computing the output of a neural network by passing input data through its layers, starting from the input layer to the output layer.**
* **In this example, we distinguish between the digits zero and one using an 8x8 image, which consists of 64-pixel intensity values.**

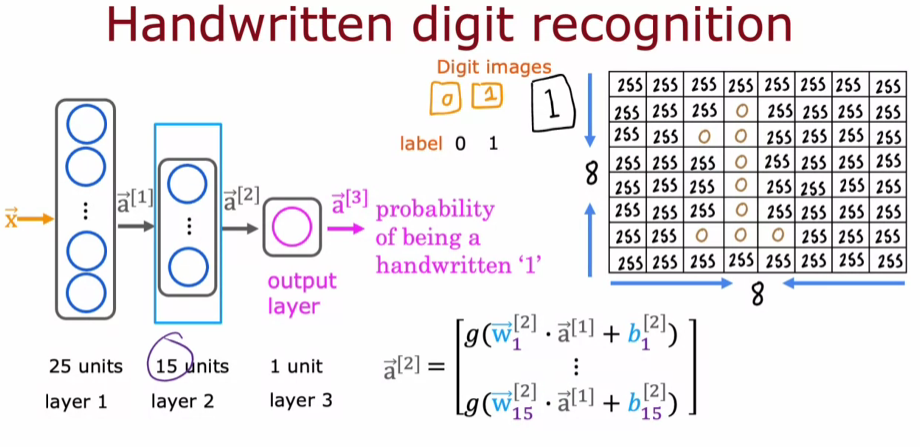


**Layer Computations**

* **The first hidden layer computes activations (a1) from the input features (x) using weights (w) and biases (b), resulting in 25 activations.**

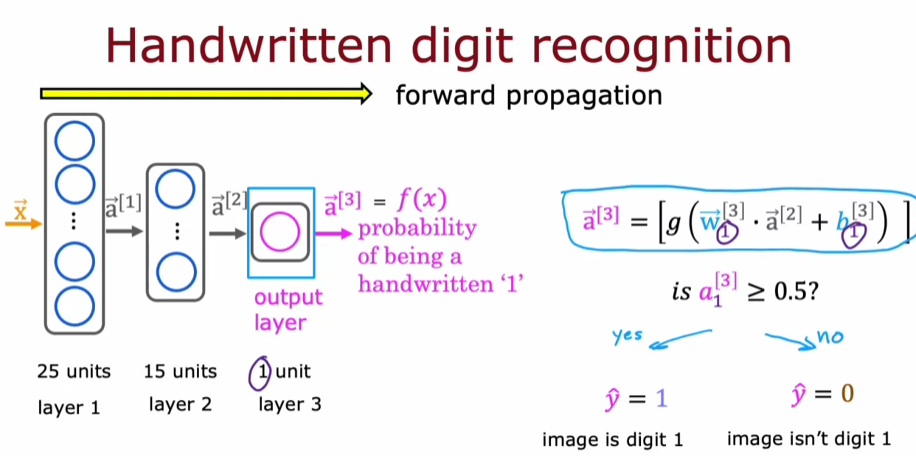
****

* **The second hidden layer computes activations (a2) from the first layer's outputs (a1), resulting in 15 activations, using a similar weighted sum and activation function.**

****

**Final Output Calculation**

* **The output layer computes the final activation (a3), which is a scalar representing the probability of the input being the digit one.**
* **The output can be thresholded to classify the input as either a zero or one, completing the forward propagation process.**

****