CHAPTER 32

Other Considerations for Training the Network

In this chapter, we will cover some other important techniques to consider when training a deep neural network.

Weight Initialization

Weight initialization is a technique for assigning initial values to the weights (parameters) of the neural network before training (see Figure 32-1). Proper weight initialization may mitigate the effects of vanishing and exploding gradients when training the network. It may also speed up the training process. Two commonly used methods for weight initializations are the Xavier and the He techniques. We will not go into the technical explanation of these initialization strategies. However, they are implemented in the standard deep learning framework libraries such as TensorFlow and Keras. In TensorFlow 2.0, the dense layer in **'tf.keras.layers.Dense()'** has the Glorot uniform initializer, also called Xavier uniform initializer as its default kernel initializer.

E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*,
https://doi.org/10.1007/978-1-4842-4470-8_32

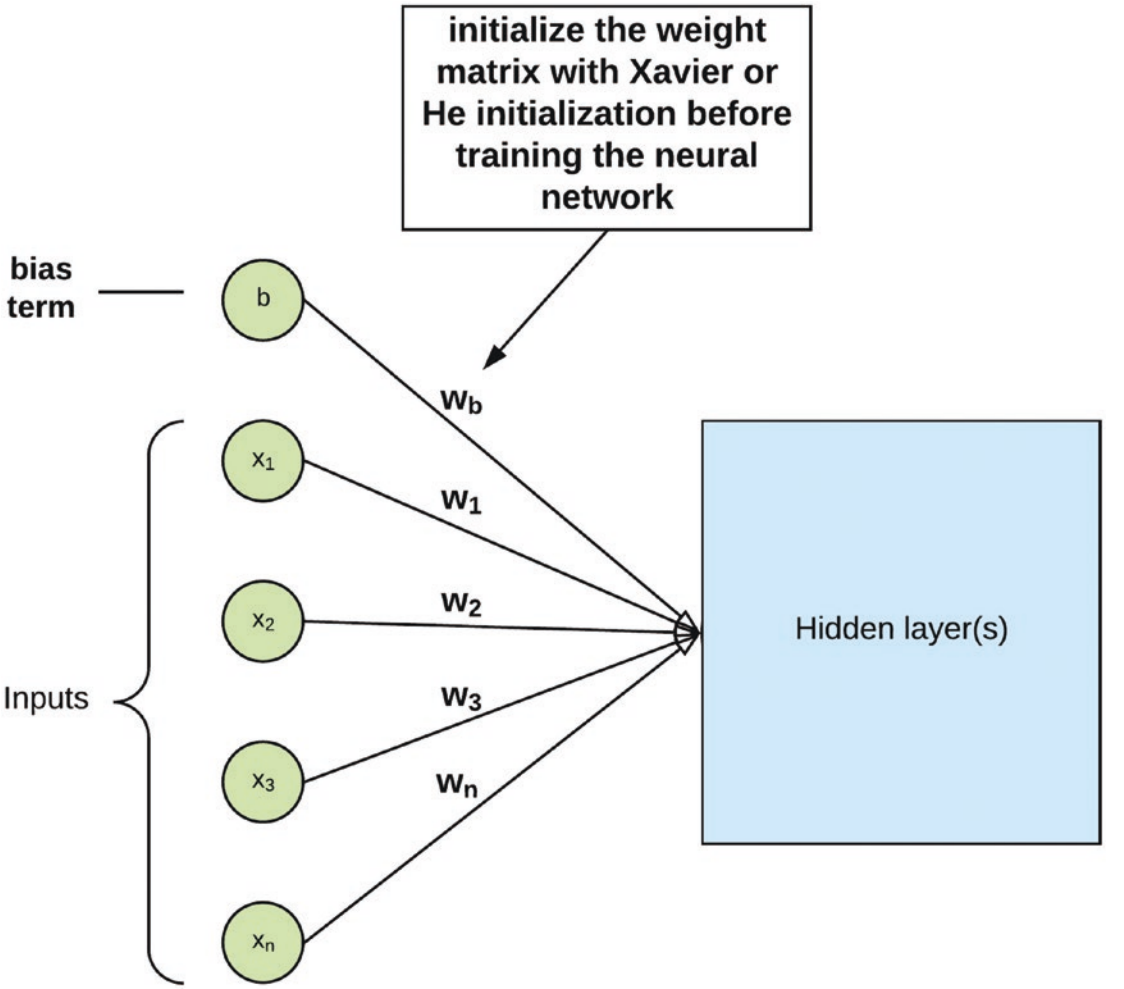

Figure 32-1. *Weight initialization*

Batch Normalization

The technique of batch normalization involves normalizing the data (to have zero mean and unit variance), as well as scaling and shifting the data batch at each layer of the neural network during the training phase. Batch normalization occurs after the affine transformation of the input matrix and their weights, but before passing the transformation into the activation function (see Figure 32-2).