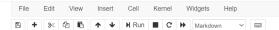


Not Trusted Python 3 O



Ungraded Lab: Activation in Custom Layers

In this lab, we extend our knowledge of building custom layers by adding an activation parameter. The implementation is pretty straightforward as you'll see below.

Imports

Adding an activation layer

To use the built-in activations in Keras, we can specify an activation parameter in the __init__() method of our custom layer class. From there, we can initialize it by using the tf.keras.activations.get() method. This takes in a string identifier that corresponds to one of the <u>available activations</u> in Keras. Next, you can now pass in the forward computation to this activation in the call() method.

We can now pass in an activation parameter to our custom layer. The string identifier is mostly the same as the function name so 'relu' below will get tf.keras.activations.relu .