**Working with image data**

You are given a black-and-white image of a letter, which has been encoded as a tensor, letter. You want to determine whether the letter is an X or a K. You don't have a trained neural network, but you do have a simple model, model, which can be used to classify letter.

The 3x3 tensor, letter, and the 1x3 tensor, model, are available in the Python shell. You can determine whether letter is a K by multiplying letter by model, summing over the result, and then checking if it is equal to 1. As with more complicated models, such as neural networks, model is a collection of weights, arranged in a tensor.

Note that the functions reshape(), matmul(), and reduce\_sum() have been imported from tensorflow and are available for use.

**Instructions**

**100 XP**

* The model, model, is 1x3 tensor, but should be a 3x1. Reshape model.
* Perform a matrix multiplication of the 3x3 tensor, letter, by the 3x1 tensor, model.
* Sum over the resulting tensor, output, and assign this value to prediction.
* Print prediction using the .numpy() method to determine whether letter is K.

# Reshape model from a 1x3 to a 3x1 tensor

model = reshape(model, (3, 1))

# Multiply letter by model

output = matmul(letter, model)

# Sum over output and print prediction using the numpy method

prediction = reduce\_sum(output)

print(prediction.numpy())

Excellent work! Your model found that prediction=1.0 and correctly classified the letter as a K. In the coming chapters, you will use data to train a model, model, and then combine this with matrix multiplication, matmul(letter, model), as we have done here, to make predictions about the classes of objects.