Chapter_2_Section_1_Defining_Tensors

February 2, 2019

1 Ch 02: Concept 01

1.1 Defining tensors

Import TensorFlow and Numpy:

```
In [1]: import tensorflow as tf
        import numpy as np
```

Now, define a 2x2 matrix in different ways:

```
In [2]: m1 = [[1.0, 2.0],
              [3.0, 4.0]]

        m2 = np.array([[1.0, 2.0],
                       [3.0, 4.0]], dtype=np.float32)

        m3 = tf.constant([[1.0, 2.0],
                          [3.0, 4.0]])
```

Let's see what happens when we print them:

```
In [3]: print(type(m1))
        print(type(m2))
        print(type(m3))

<class 'list'>
<class 'numpy.ndarray'>
<class 'tensorflow.python.framework.ops.Tensor'>
```

So, that's what we're dealing with. Interesting.

By the way, there's a function called `convert_to_tensor(...)` that does exactly what you might expect.

Let's use it to create tensor objects out of various types:

```
In [4]: t1 = tf.convert_to_tensor(m1, dtype=tf.float32)
        t2 = tf.convert_to_tensor(m2, dtype=tf.float32)
        t3 = tf.convert_to_tensor(m3, dtype=tf.float32)
```

Ok, ok! Time for the reveal:

```
In [5]: print(type(t1))
        print(type(t2))
        print(type(t3))

<class 'tensorflow.python.framework.ops.Tensor'>
<class 'tensorflow.python.framework.ops.Tensor'>
<class 'tensorflow.python.framework.ops.Tensor'>
```