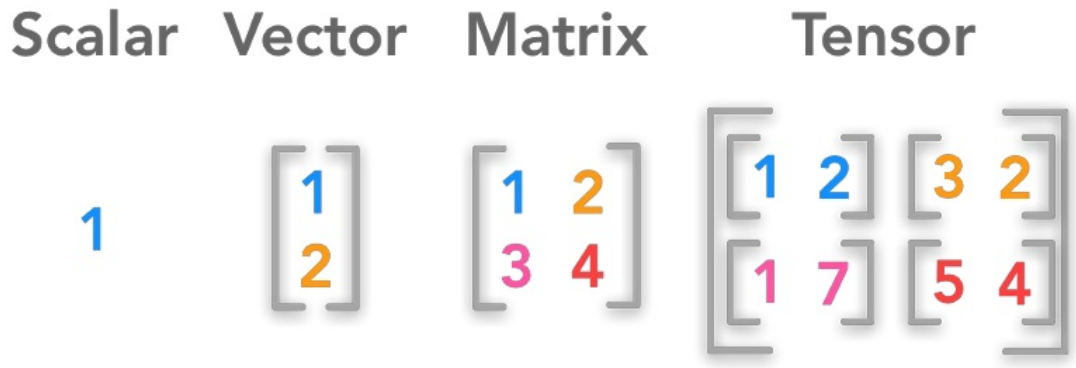
Tensors Introduction

May 22, 2021

1 Introduction to Tensor Notes :

A tensor is a container which can house data in N dimensions, along with its linear operations, though there is nuance in what tensors technically are and what we refer to as tensors in practice.



[4]:
```
#Import Tensorflow
import tensorflow as tf
print(tf.__version__)
import numpy
```

```
2.3.0
```

[3]:
```
#Create tesnors with tf.constant()
scalar = tf.constant(7)
scalar
```

[3]:
```
<tf.Tensor: shape=(), dtype=int32, numpy=7>
```

[6]:
```
#Check the number of dimensions of tensor using ndim
scalar.ndim
```

[6]:
```
0
```

[7]:
```
#Create a vector
vector = tf.constant([10,10])
vector
```

[7]:
```
<tf.Tensor: shape=(2,), dtype=int32, numpy=array([10, 10])>
```

[8]:
```
vector.ndim
```

[8]:
```
1
```

[10]:
```
#Create a matrix
matrix = tf.constant([[10,7],
                      [7,10]]
                      )
matrix
```

[10]:
```
<tf.Tensor: shape=(2, 2), dtype=int32, numpy=
array([[10,  7],
       [ 7, 10]])>
```

[11]:
```
matrix.ndim
```

[11]:
```
2
```

At this point, we can relate that ndim represents the number of elements in the shape tuple

[13]:
```
#Create another matric
matrix_2 = tf.constant([[10.,7.],
                        [3.,2.],
                         [1.,2.]],dtype=tf.float16) #Here we use float16 since␣
 ↪our numbers are small
matrix_2
```

[13]:
```
<tf.Tensor: shape=(3, 2), dtype=float16, numpy=
array([[10.,  7.],
       [ 3.,  2.],
       [ 1.,  2.]], dtype=float16)>
```

[14]:
```
matrix_2.ndim
```

[14]:
```
2
```

[18]:
```
#Create a tensor
tensor = tf.constant([[[1,2,3],
                       [4,5,6]],
                        [[7,8,9],
                         [10,11,12]],
                        [[13,14,15],
                         [16,17,18]]])
tensor
```

[18]:
```
<tf.Tensor: shape=(3, 2, 3), dtype=int32, numpy=
array([[[ 1,  2,  3],
        [ 4,  5,  6]],
```

```
        [[ 7,  8,  9],
         [10, 11, 12]],

        [[13, 14, 15],
         [16, 17, 18]]])>
```

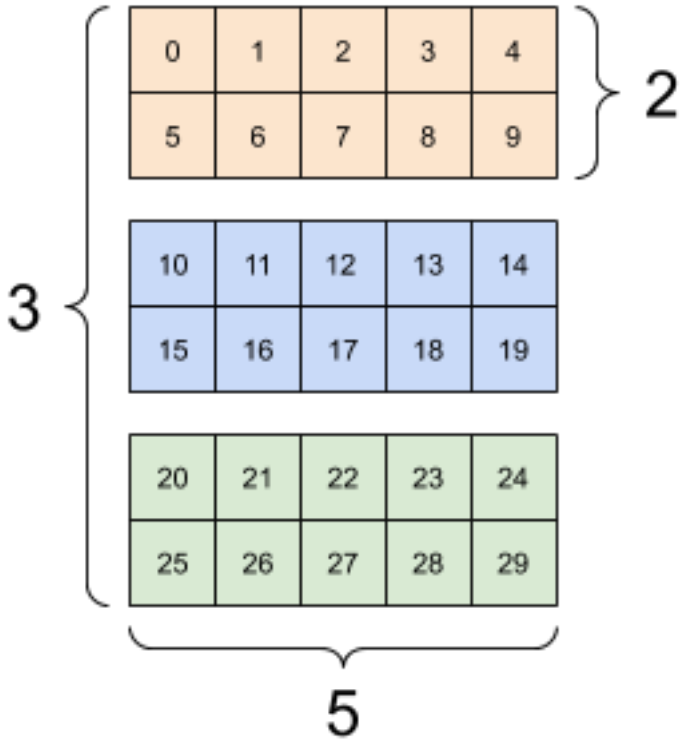
[19]:
```
rank_3_tensor = tf.constant([
  [[0, 1, 2, 3, 4],
   [5, 6, 7, 8, 9]],
  [[10, 11, 12, 13, 14],
   [15, 16, 17, 18, 19]],
  [[20, 21, 22, 23, 24],
   [25, 26, 27, 28, 29]],])

print(rank_3_tensor)
```

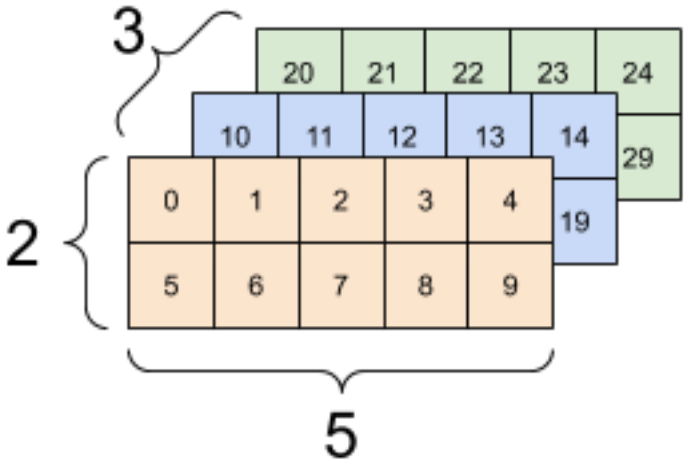
```
tf.Tensor(
[[[ 0  1  2  3  4]
  [ 5  6  7  8  9]]

 [[10 11 12 13 14]
  [15 16 17 18 19]]

 [[20 21 22 23 24]
  [25 26 27 28 29]]], shape=(3, 2, 5), dtype=int32)
```



The above example for a 3 dimensional tensor represents : Number of matrices , Number of rows in a matrices and Number of columns in a matrices i.e. Shape = (3,2,5). You can also visualise this as matrices stacked on top of each other to produce a 3D structure as shown below



2 Summary so far :

1. Scalar : Single number
2. Vector : A number with both direction and magnitude
3. Matrix : A 2 dimensional array of numbers
4. Tensor : A n-dimensional array of numbers which can constitude all of the above as well.