使用TensorFlow 之前你需要了解关于TensorFlow 的以下基础知识:

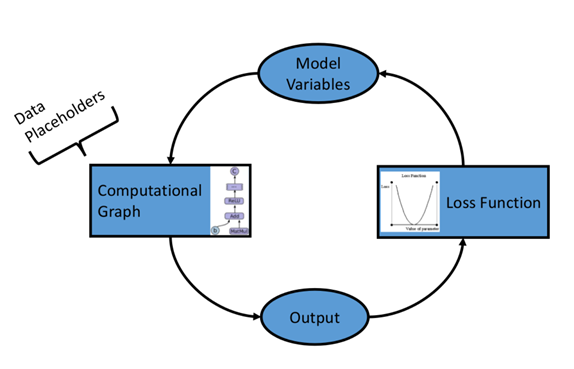
• 使用图(graphs) 来表示计算.

• 在会话(Session) 中执行图.

• 使用张量(tensors) 来代表数据.

• 通过变量(Variables) 维护状态.

• 使用供给(feeds) 和取回(fetches) 将数据传入或传出任何操作.



1. 图

What?

结点和有向边

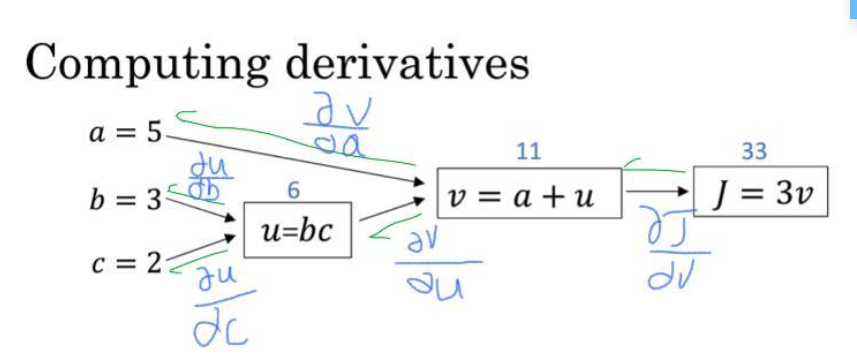
比如说 J=a+bc 在图计算的表示中：

1.节点表示某种运算

2.有向边表示数据和数据的流向

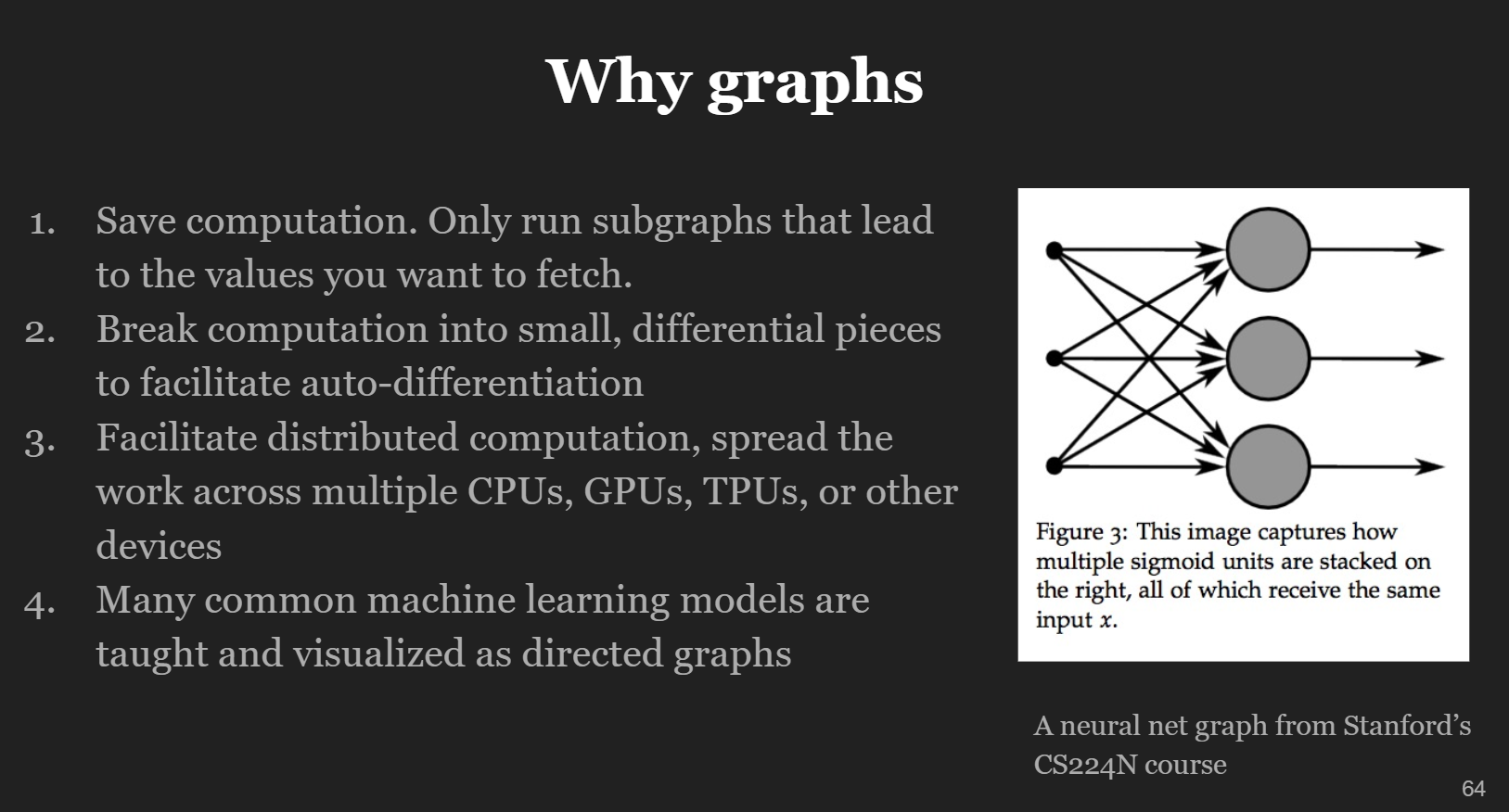
比如 节点u=a+u，有两条边流进来，就说明这个节点是需要这两条边的数据。这是“需”

而 a=5 会自发的流向v=a+u，这是“供”（5，3，2也都是节点）



Why?

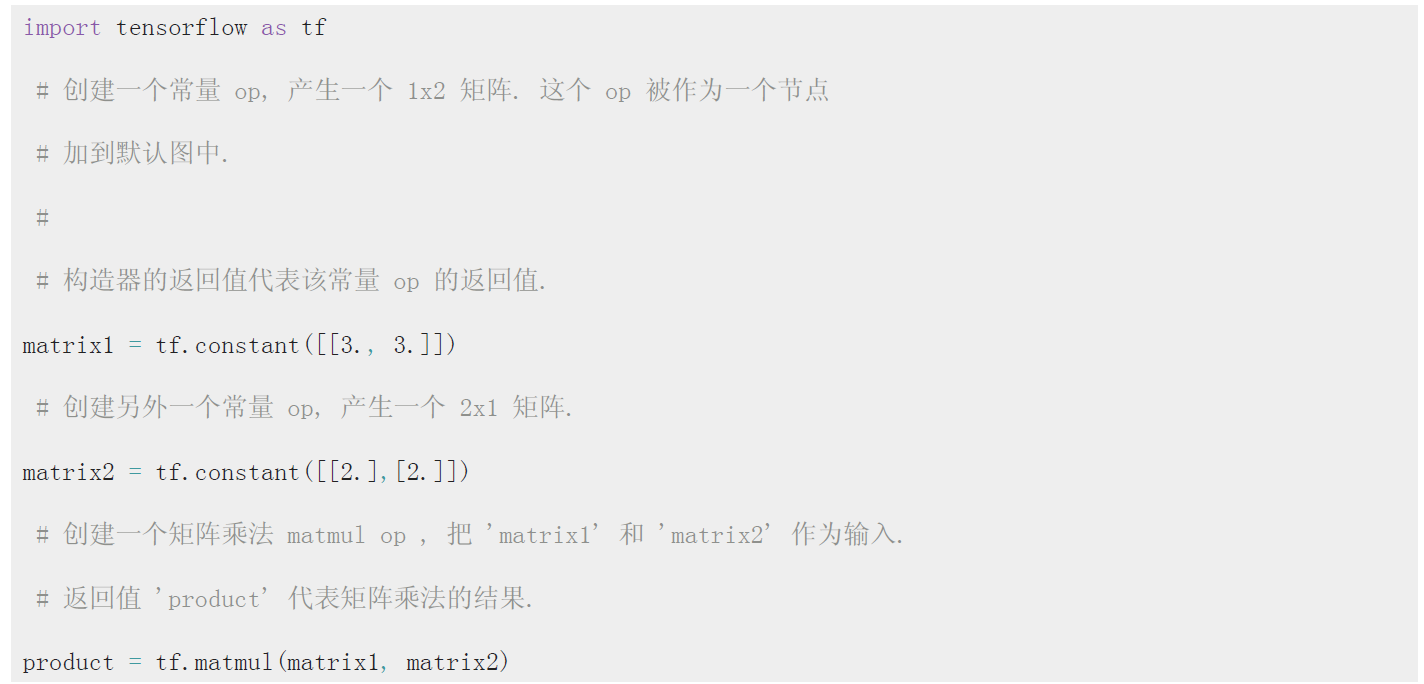
利用这个图来表示计算而不是一般的计算是因为，正向可以方便取中间值求结果，反向可以很方便的求偏导。



How?

创建节点（输出）和边（输入）

Implement?



这样图就设计好了，之后应该怎么获取图的值呢？-Create a session.

1. 在一个会话中启动图

What?

通过会话启动图

Why?

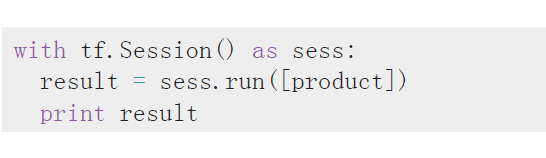
获取图的值

How?

第一步是创建一个Session 对象, tf.Session(),

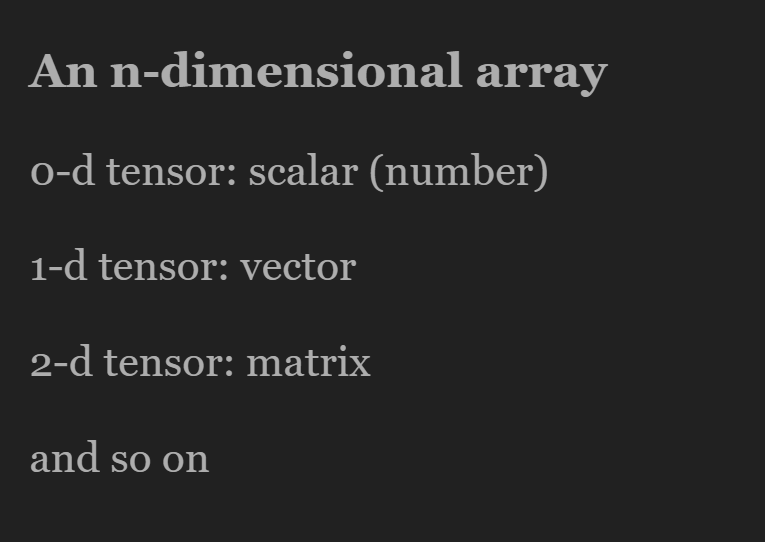
第二步通过run()方法feed，然后fetch到该图的值

Implement:用with as语句简化代码

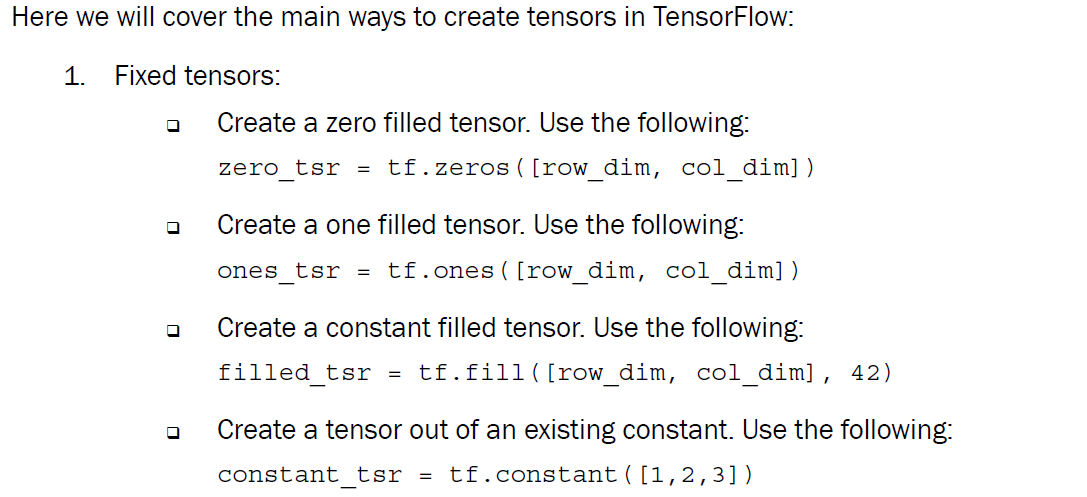


1. Tensor

What?



How?



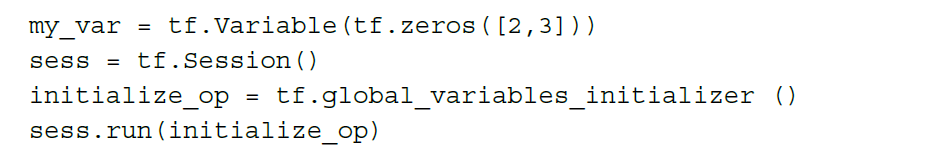
1. 变量

What?

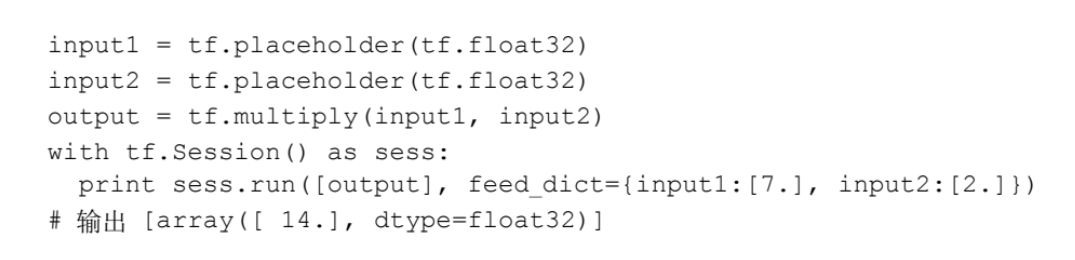
在图中有固定位置的变量（一般为需要更新的参数），不会像tensor那样流动

How?

1. 创建变量对象Variable()
2. 需要初始化



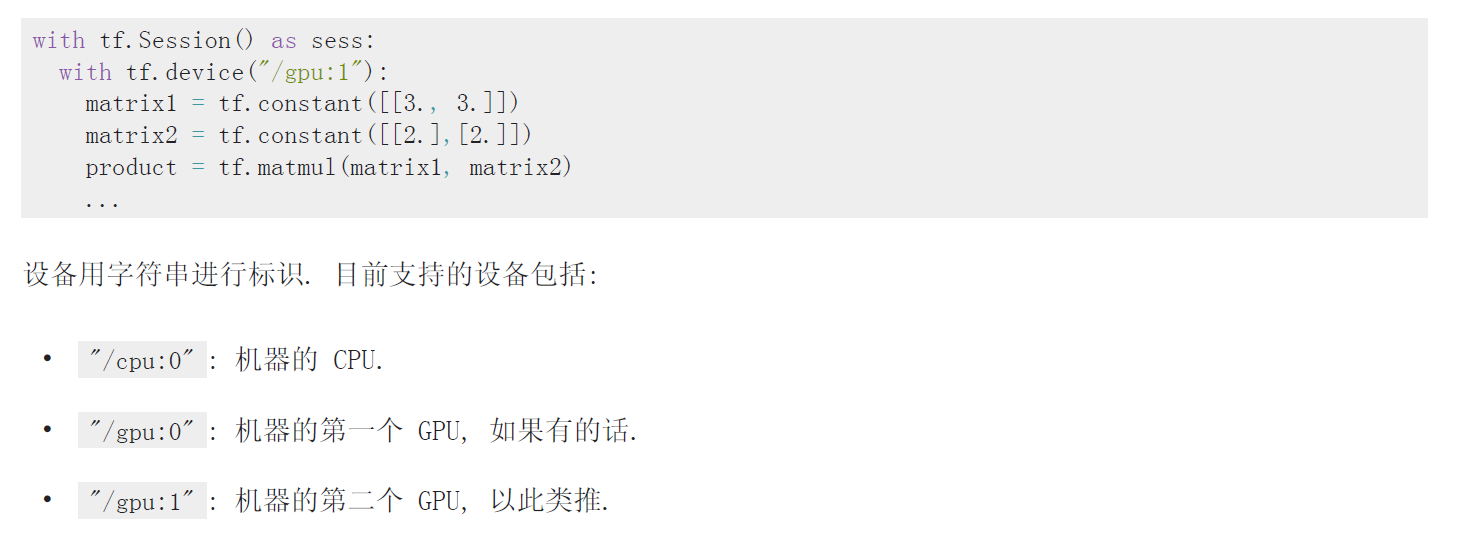
TensorFlow还提供了填充机制，可以在构建图时使用tf.placeholder()临时替代任意操作的张量，在调用Session对象的run()方法去执行图时，使用填充数据作为调用的参数，调用结束后，填充数据就消失。代码示例如下：



1. 设备

如果机器上有超过一个可用的 GPU, 除第一个外的其它 GPU 默认是不参与计算的. 为了让 TensorFlow 使用这些

GPU, 你必须将 op 明确指派给它们执行. with...Device 语句用来指派特定的 CPU 或 GPU 执行操作:



What is Tensorflow?

Goole machine learning framework.

Why?

Because the computational graph concept, automatic differentiation, and the adaptability of the Tensorflow python API structure.

How TensorFlow Works?

1. Import or generate data.

either generate data or use an outside source of data. Sometimes it is better to rely on generated data because we will want to know the expected outcome.

1. Transform and normalize data.

The data is usually not in the correct dimension or type that our Tensorflow algorithms expect. We will have to transform our data before we can use it. Most algorithms also expect normalized data and we will do this here as well. Tensorflow has built in functions that can normalize the data for you.

1. Set algorithm parameters.

For example, this can be the number of iterations, the learning rate, or other fixed parameters of our choosing.

1. Initialize variables and placeholders.

