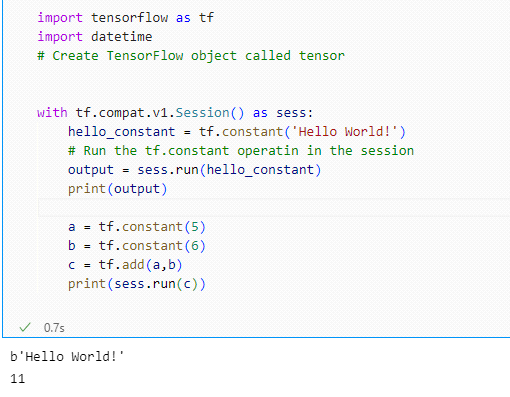
* **Tensorflow:**

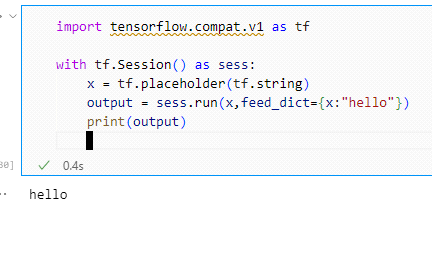
Trong tensorflow, dữ liệu không được lưu dưới dạng int, float hay string. Các giá trị này được đóng gói trong một đối tượng gọi là tensor.

tf.constant() được gọi là 1 tensor ko đổi vì giá trị của tensor không bao giờ thay đổi.

Tensorflow phân biệt rạch ròi việc định nghĩa và tính toán trong quá trình thực thi. Bao gồm:  
 + Xây dựng, định nghĩa đồ thị(Graph).   
 + Sử dụng một Session để thực thi các tính toán trong đồ thị  
sess.run() đánh giá tensor và trả về kết quả.



tf.placeholder() trả về 1 tensor nhận giá trị được truyền vào từ tf.session.run() thông qua feed\_dict, tf.placeholder() cho phép đặt kiểu dữ liệu đầu vào ngay trước khi run()



* **Phép tính trong tensorflow:**

Phép cộng: x =tf.add(5,7)

Phép trừ: y = tf.subtract(10,4)

Phép nhân: z = tf.multiply(2,5)

Phép chia: t = tf.divide(10,2)

Tính từ trong ra ngoài:

Vd: tf.subtract(tf.cast(tf.constant(2.0), tf.int32), tf.constant(1)) # 1

* **Biến trong tensor:**

x= tf.variable(5)

lớp tf.variable tạo ra 1 tensor giá trị ban đầu và có thể thay đổi giá trị.

(x1.initializer) điều này nghĩa là khởi tạo x với giá trị bằng 5

init=tf.global\_variables\_initializer()#khởi tạo trạng thái của tất cả cácbiến

with tf.Session() as sess:

sess.run(init)

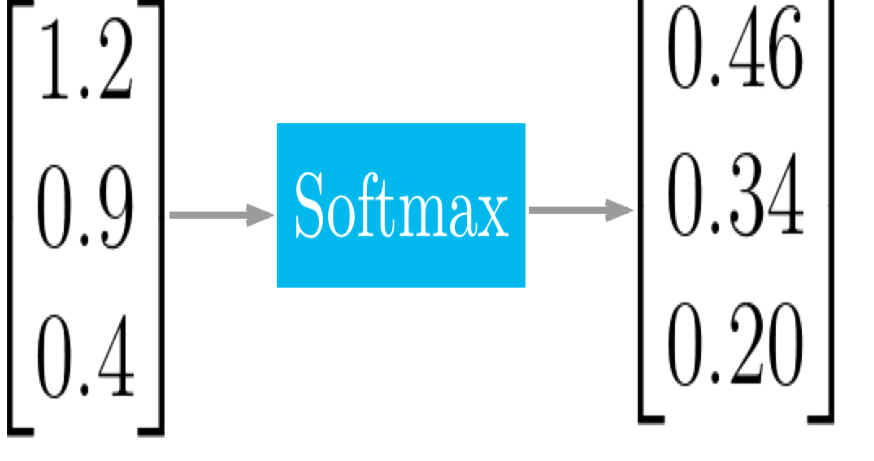
* tf.truncated\_normal()

hàm trả về 1 tensor có giá trị ngẫu nhiên, độ lớn không quá 2

* tf.zeros()

trả về 1 tensor với tất cả các số 0.

* Tensorflow softmax:

Chuẩn hóa đầu ra sao cho tất cả có tổng là 1:  


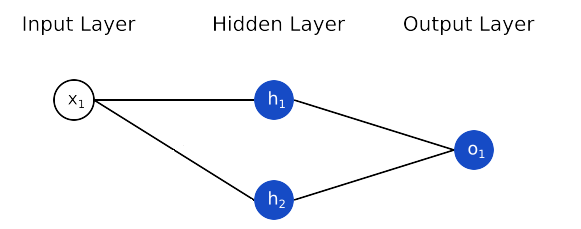
**2, trình bày hiểu biết về học sâu:**

TensorFlow cung cấp một nền tảng học máy toàn diện cung cấp cả khả năng cấp cao và cấp thấp để xây dựng và triển khai các mô hình học máy. Tuy nhiên, nó có một đường cong học tập dốc. Nó được sử dụng tốt nhất khi bạn có nhu cầu:

* Nghiên cứu học tập sâu
* Mạng nơ-ron phức tạp
* Làm việc với bộ dữ liệu lớn
* Các mô hình hiệu suất cao

Cách xác định mạng thần kinh bằng API tuần tự của Keras

API tuần tự là một khuôn khổ để tạo các mô hình dựa trên các phiên bản của lớp **sequential ()**. Mô hình có một biến đầu vào, một lớp ẩn với hai nơ-ron và một lớp đầu ra với một đầu ra nhị phân. Các lớp bổ sung có thể được tạo và thêm vào mô hình.



# Xác định mô hình:

from keras.models import Sequential

from keras.layers import Dense

model = Sequential()

model.add (Dense (2, input\_dim=1,activation=’relu'))

model.add (Dense (1, activation='sigmoid'))

Hàm **Summary ()**được sử dụng để tạo và in tóm tắt trong bảng điều khiển Python:

from keras.models import Sequential

from keras.layers import Dense

model = Sequential()

model.add(Dense(2, input\_dim=1, activation='relu'))

model.add(Dense(1, activation='sigmoid'))

print(model.summary())

