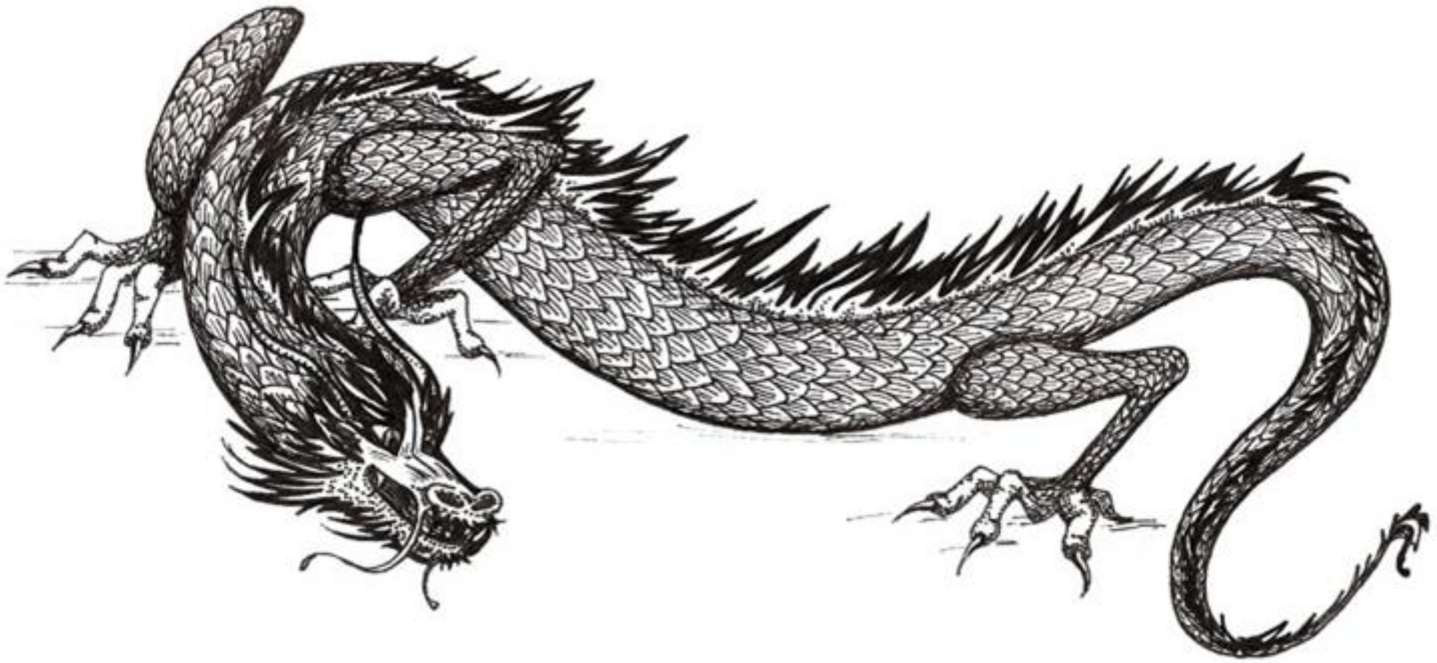


全连接层

主 讲 人: 龙 良 曲

Outline

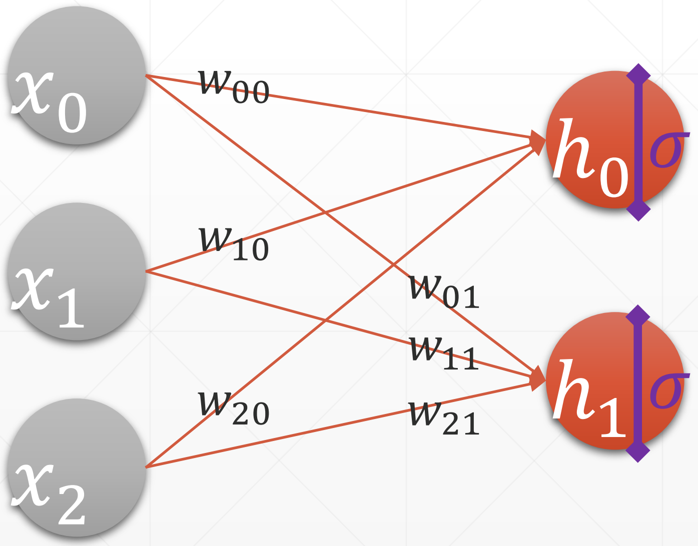
- Matmul
- Neural Network
- Deep Learning
- Multi-Layer

Recap

- $out = \boldsymbol{f}(X@W + b)$
- $out = \boldsymbol{relu}(X@W + b)$

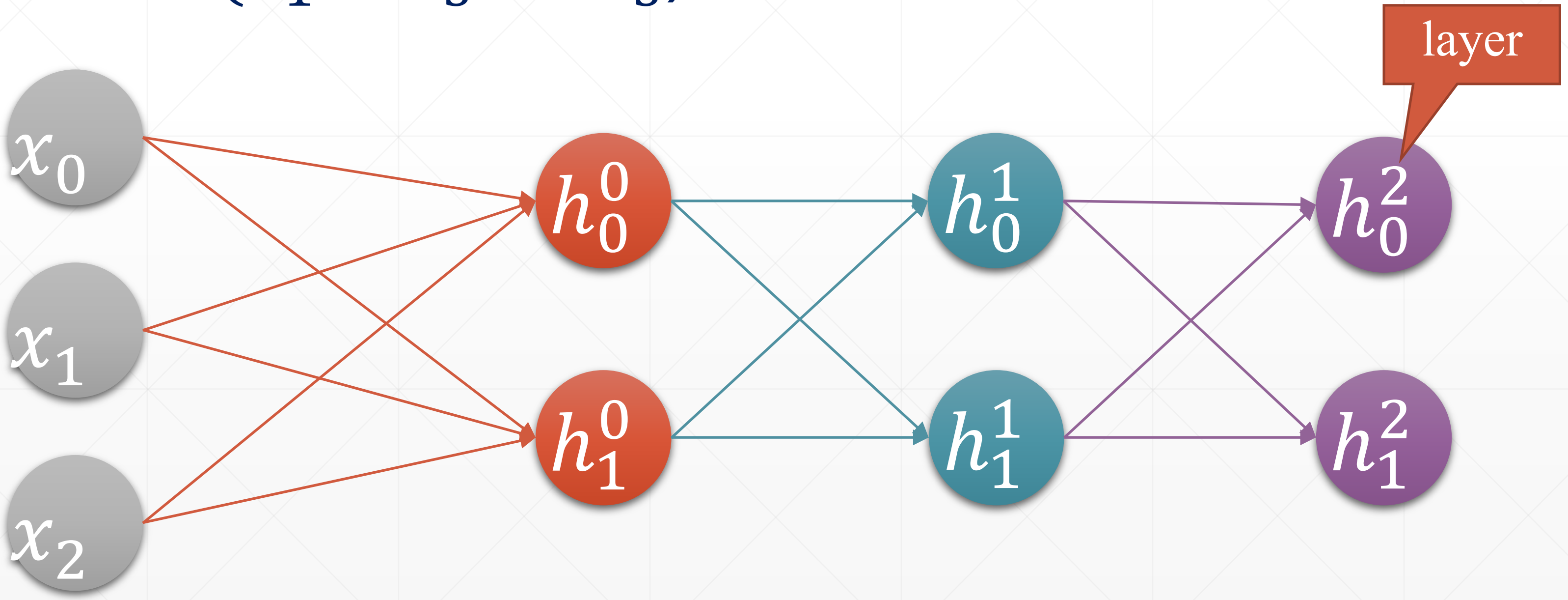
X@W+b

- $h = \boldsymbol{relu}(X@W + b)$
- $\begin{bmatrix} h_0^0 & h_1^0 \\ h_0^1 & h_1^1 \end{bmatrix} = relu(\begin{bmatrix} x_0^0 & x_1^0 & x_2^0 \\ x_0^1 & x_1^1 & x_2^1 \end{bmatrix} @ \begin{bmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \\ w_{20} & w_{21} \end{bmatrix} + [b_0 \quad b_1])$



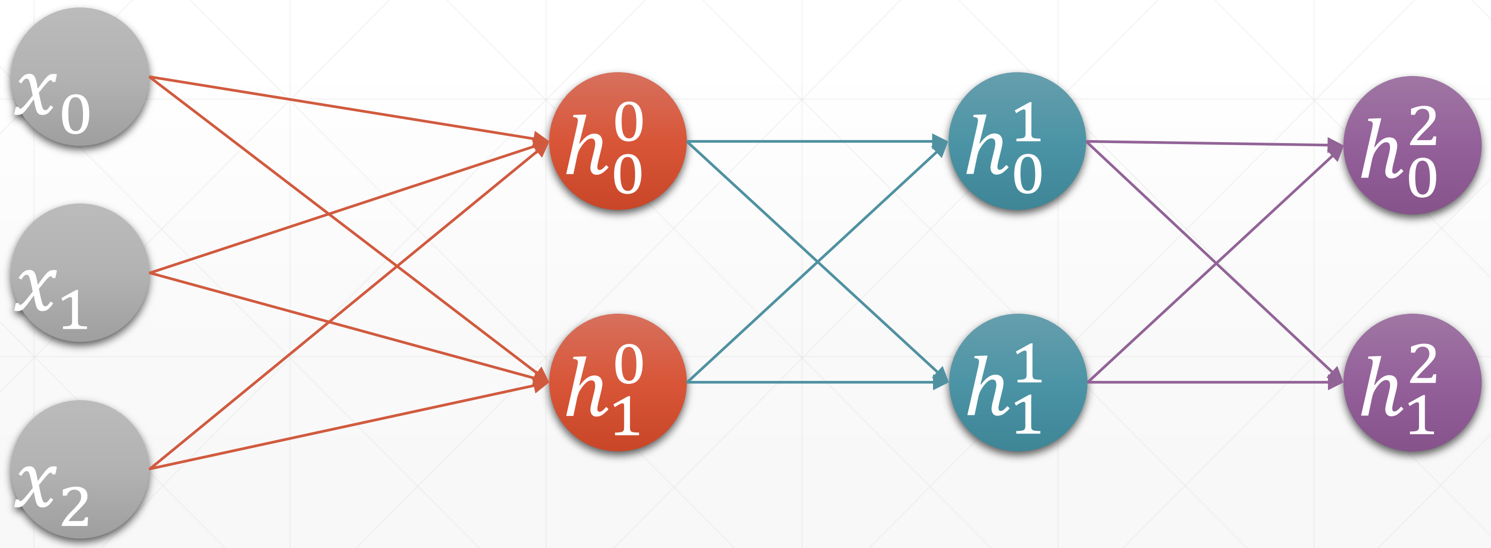
Black Magic!

- $h_0 \;\; = \boldsymbol{relu}(X@W_1 \; + \; b_1)$
- $h_1 \;\; = \boldsymbol{relu}(h_0@W_2 \; + \; b_2)$
- $out = \boldsymbol{relu}(h_1@W_3 \; + \; b_3)$



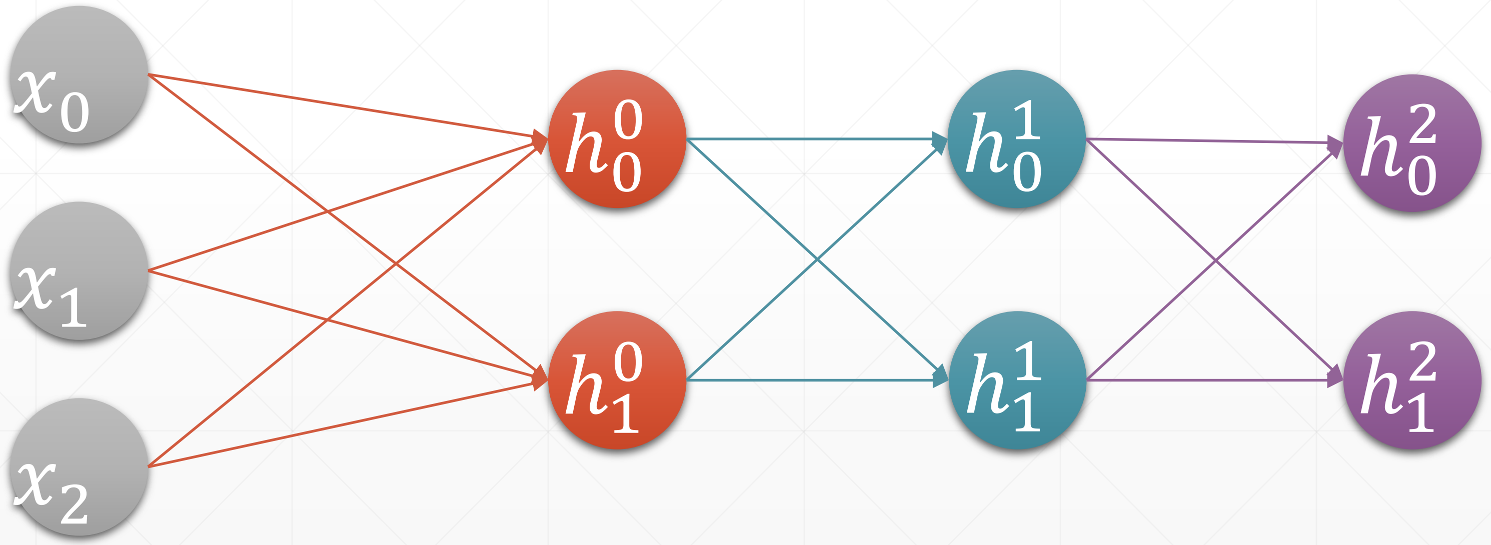
Layers

- Input
- Hidden
- Output



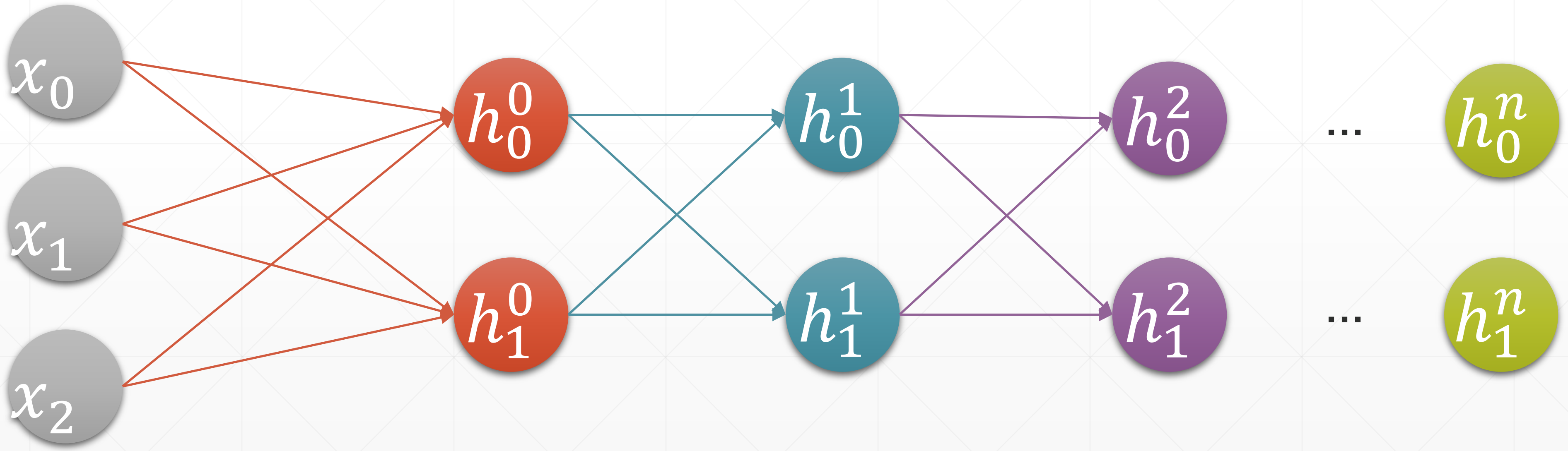
Here comes Deep Learning！

- Neural Network in the 1980s
  - 3~5 layers



Here comes Deep Learning!

- Deep Learning now
  - $n \approx 1200\ layers$



Why?

- 486 PC with DSP32C
  - 20Mflops, 4MB RAM

- Telsa V100
  - 32GB HBM2, 100Tflops

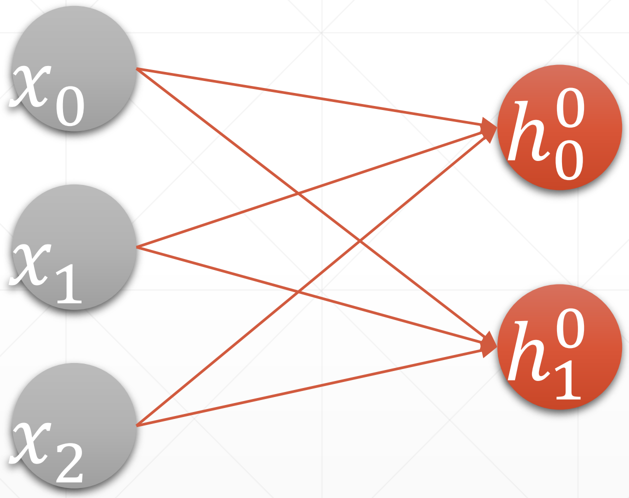
https://www.youtube.com/watch?v=FwFduRA_L6Q

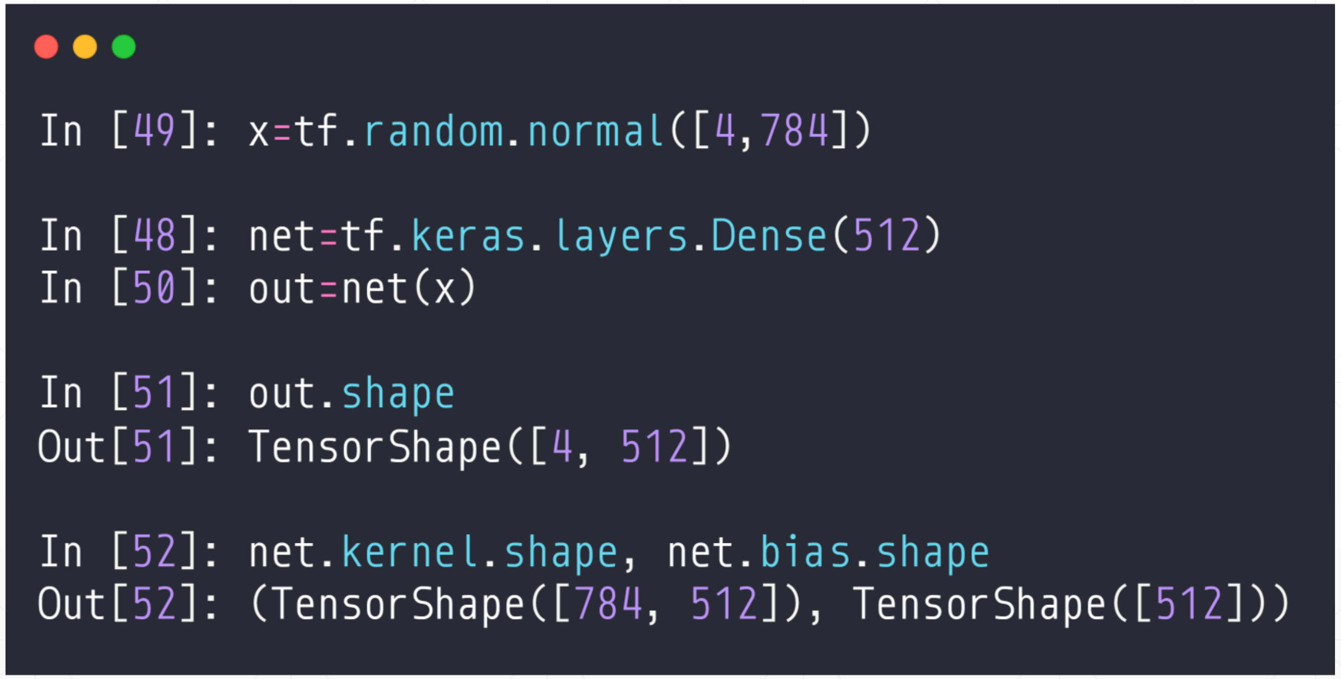
Heroes

- BigDATA
- ReLU
- Dropout
- BatchNorm
- ResNet
- Xavier Initialization
- Caffe/TensorFlow/PyTorch
- …

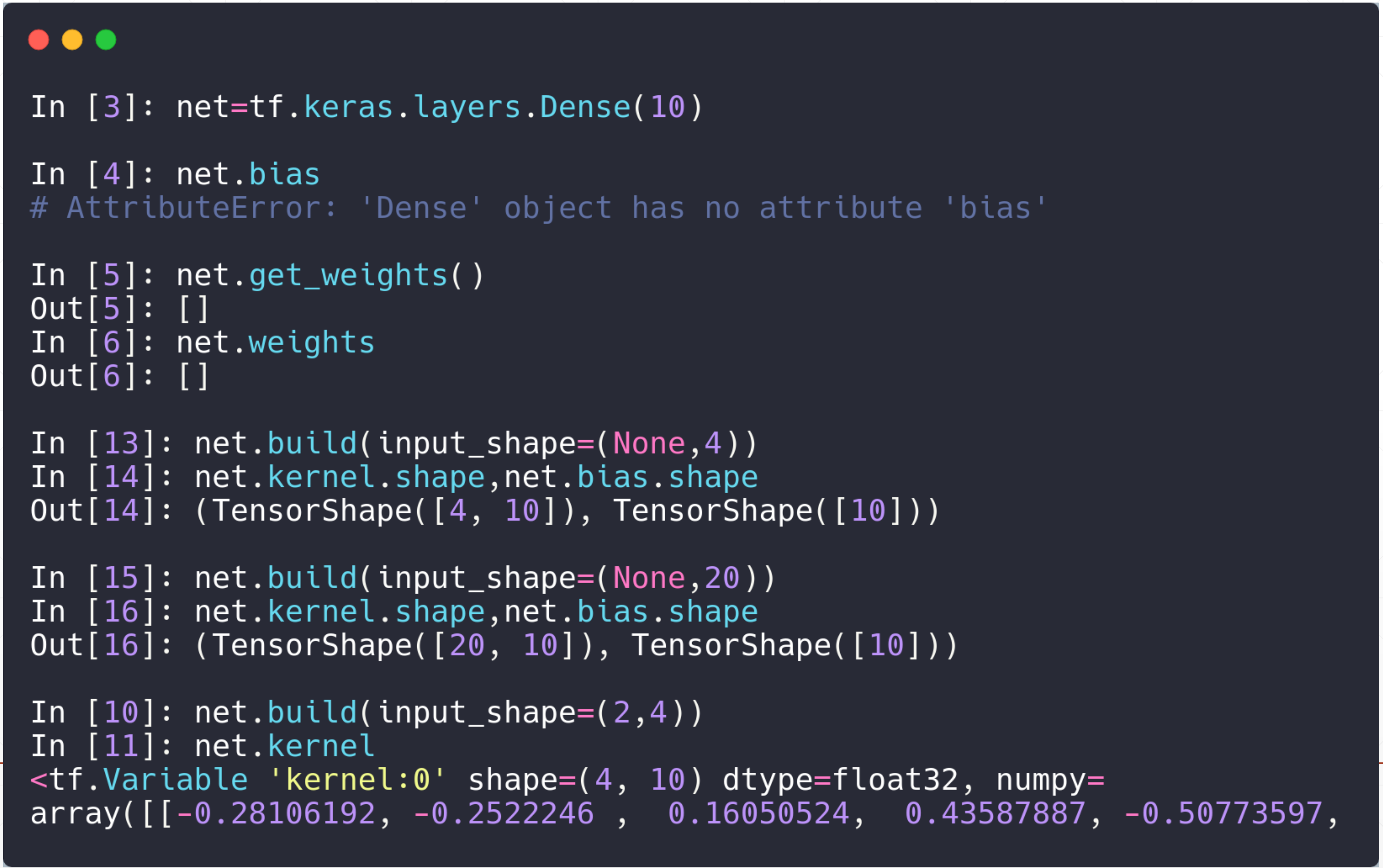
http://capacitybc.com/mini-series-part-3-the-hero1

Fully connected layer

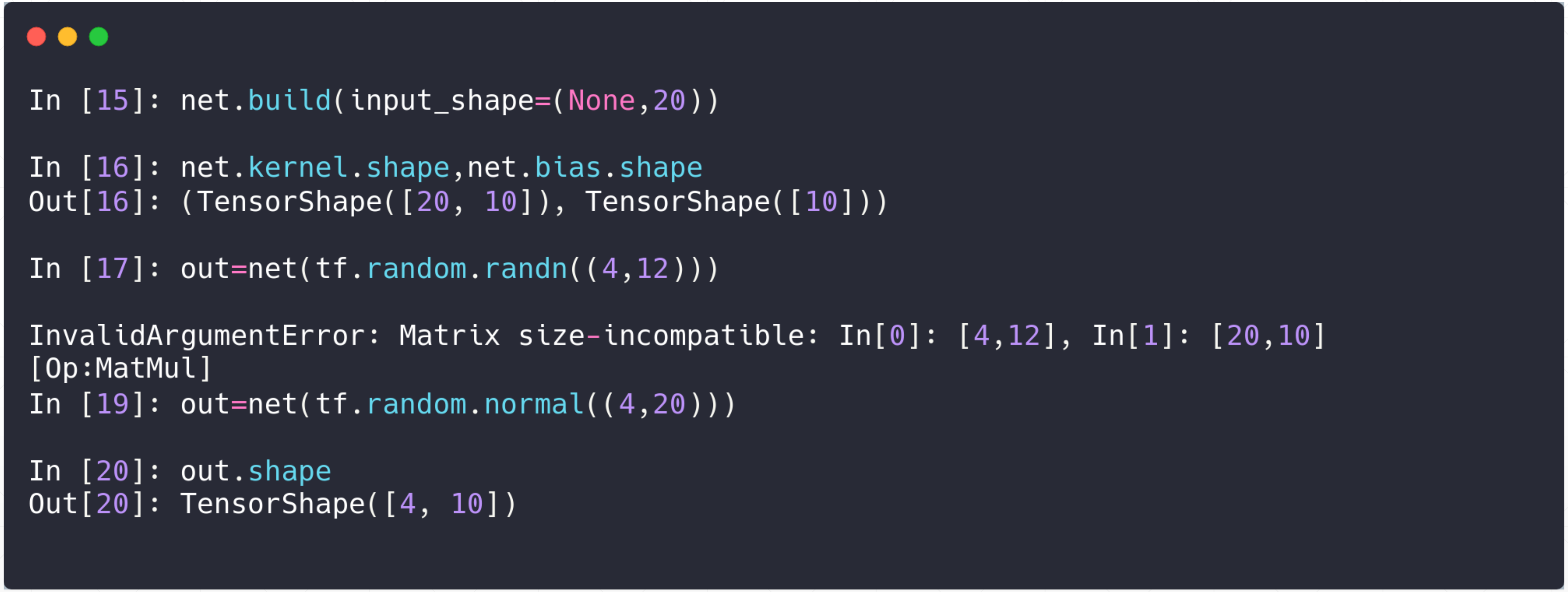




```
In [49]: x=tf.random.normal([4,784])

In [48]: net=tf.keras.layers.Dense(512)
In [50]: out=net(x)

In [51]: out.shape
Out[51]: TensorShape([4, 512])

In [52]: net.kernel.shape, net.bias.shape
Out[52]: (TensorShape([784, 512]), TensorShape([512]))
```



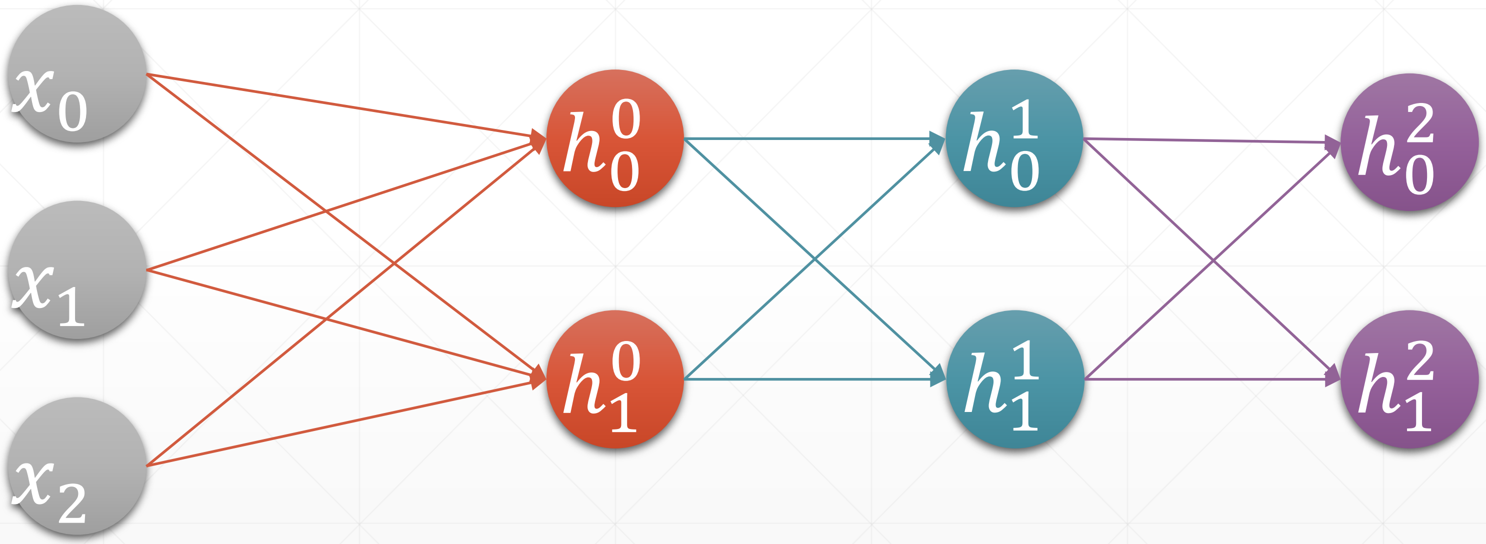
```
In [3]: net=tf.keras.layers.Dense(10)

In [4]: net.bias
# AttributeError: 'Dense' object has no attribute 'bias'

In [5]: net.get_weights()
Out[5]: []
In [6]: net.weights
Out[6]: []

In [13]: net.build(input_shape=(None,4))
In [14]: net.kernel.shape,net.bias.shape
Out[14]: (TensorShape([4, 10]), TensorShape([10]))

In [15]: net.build(input_shape=(None,20))
In [16]: net.kernel.shape,net.bias.shape
Out[16]: (TensorShape([20, 10]), TensorShape([10]))

In [10]: net.build(input_shape=(2,4))
In [11]: net.kernel
<tf.Variable 'kernel:0' shape=(4, 10) dtype=float32, numpy=
array([[-0.28106192, -0.2522246 ,  0.16050524,  0.43587887, -0.50773597,
```



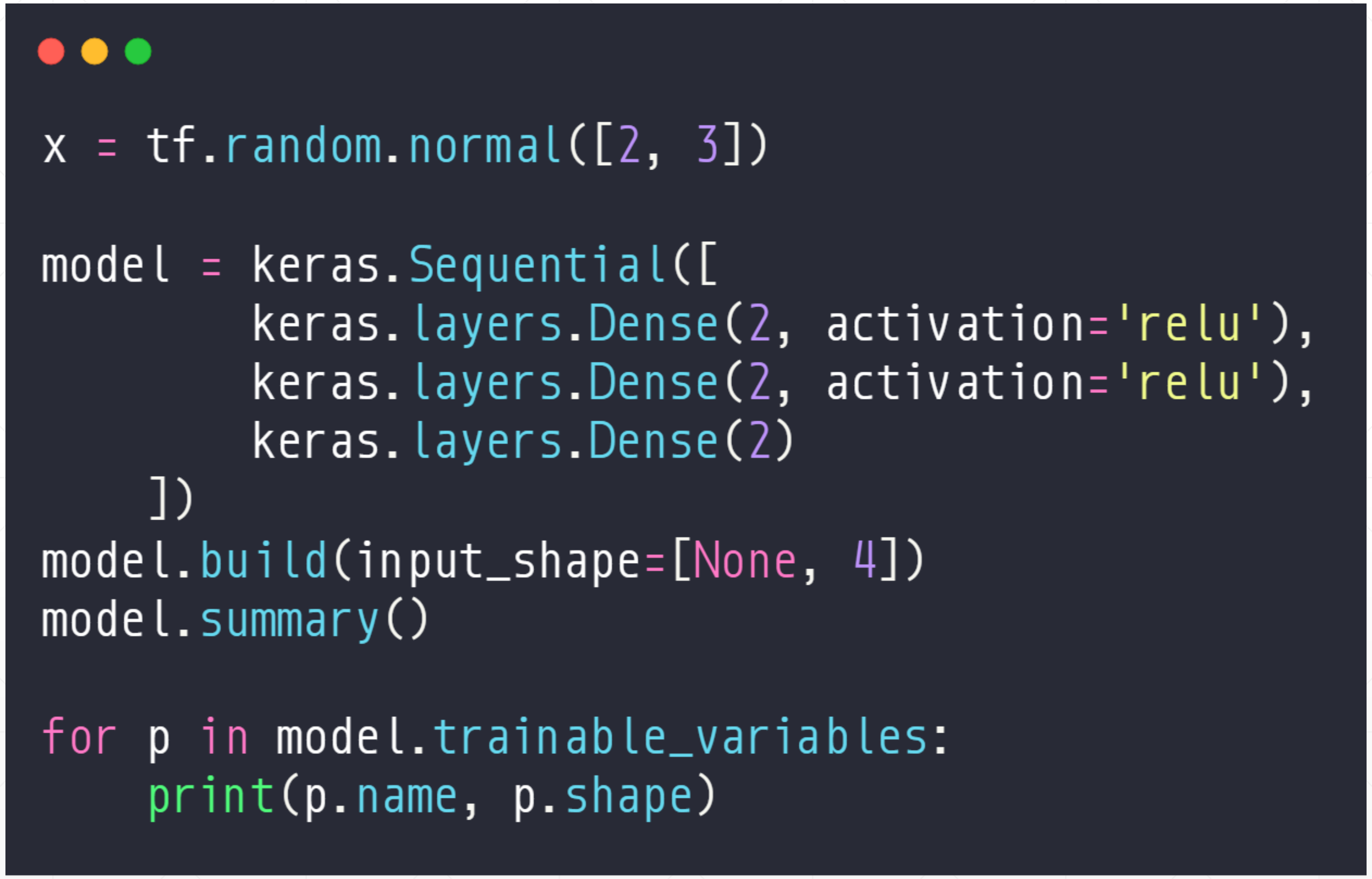
```
In [15]: net.build(input_shape=(None,20))

In [16]: net.kernel.shape,net.bias.shape
Out[16]: (TensorShape([20, 10]), TensorShape([10]))

In [17]: out=net(tf.random.randn((4,12)))

InvalidArgumentError: Matrix size-incompatible: In[0]: [4,12], In[1]: [20,10]
[Op:MatMul]
In [19]: out=net(tf.random.normal((4,20)))

In [20]: out.shape
Out[20]: TensorShape([4, 10])
```

Multi-Layers

- keras.Sequential([layer1, layer2, layer3])



Sequential



```
x = tf.random.normal([2, 3])

model = keras.Sequential([
        keras.layers.Dense(2, activation='relu'),
        keras.layers.Dense(2, activation='relu'),
        keras.layers.Dense(2)
    ])
model.build(input_shape=[None, 4])
model.summary()

for p in model.trainable_variables:
    print(p.name, p.shape)
```

JUST
DO
IT.

https://www.facebook.com/deepdreamgenerator/photos/a.892441237472223/1449674838415524/?type=1&theater

Next

神经网络层与训练方法
图片识别
文本理解
艺术创作
自动决策
等等

免费才是世上最昂贵的东西！ --马云

- 超高水准教程
- 持续更新算法
- 及时答疑解惑
- 交流学习平台
- 学习规划指导

ClosedAI

Going Deeper!

人工智能101学院

- https://study.163.com/provider/480000001847407/index.htm?share=2&shareId=480000001847407

Scan me

Thank You.