

The background image shows a vast mountain range under a sky transitioning from deep blue to vibrant orange and pink. In the foreground, a bright orange tent is set up on a rocky, grassy slope, its light reflecting softly. A small body of water is visible in the middle ground.

DEEPMLEARNING

# TENSORFLOW

<https://www.tensorflow.org>

Rafael Vieira Coelho

# TENSORFLOW

INTRODUÇÃO  
E PRIMEIRO EXEMPLO



# O QUE É TENSORFLOW?

- Biblioteca implementada pelo Google
- Utiliza Python para criar um front-end para a implementação em C++
- É utilizada para processamento de linguagem, redes neurais recorrentes, classificação digital de escrita e reconhecimento de imagens.
- Roda em iOS, Android, GPUs, CPUs, máquinas locais e dispositivos de cluster em nuvem.

# INSTALANDO A BIBLIOTECA

- pip install pip --upgrade
- pip install tensorflow

Para instalar a versão GPU:

- pip install -U tensorflow-gpu

```
coelho — pip install tensorflow==2.0.0-alpha0 — 90x35
mac:~ coelho$ !pip install tensorflow==2.0.0-alpha0
-bash: !pip: event not found
mac:~ coelho$ pip install tensorflow==2.0.0-alpha0
Collecting tensorflow==2.0.0-alpha0
  Downloading tensorflow-2.0.0a0-cp36-cp36m-macosx_10_11_x86_64.whl (64.3 MB)
    |████████████████████████████████| 64.3 MB 343 kB/s
Collecting protobuf>=3.6.1
  Downloading protobuf-3.12.1-cp36-cp36m-macosx_10_9_x86_64.whl (1.3 MB)
    |████████████████████████████████| 1.3 MB 27.9 MB/s
Collecting keras-preprocessing>=1.0.5
  Downloading Keras_Preprocessing-1.1.2-py2.py3-none-any.whl (42 kB)
    |████████████████████████████████| 42 kB 2.7 MB/s
Collecting google-pasta>=0.1.2
  Downloading google_pasta-0.2.0-py3-none-any.whl (57 kB)
    |████████████████████████████████| 57 kB 8.3 MB/s
Collecting keras-applications>=1.0.6
  Downloading Keras_Applications-1.0.8-py3-none-any.whl (50 kB)
    |████████████████████████████████| 50 kB 7.0 MB/s
Collecting tb-nightly<1.14.0a20190302,>=1.14.0a20190301
  Downloading tb_nightly-1.14.0a20190301-py3-none-any.whl (3.0 MB)
    |████████████████████████████████| 3.0 MB 1.5 MB/s
Requirement already satisfied: grpcio>=1.8.6 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow==2.0.0-alpha0) (1.12.0)
Collecting absl-py>=0.7.0
  Downloading absl-py-0.9.0.tar.gz (104 kB)
    |████████████████████████████████| 104 kB 26.8 MB/s
Requirement already satisfied: six>=1.10.0 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow==2.0.0-alpha0) (1.11.0)
Collecting tf-estimator-nightly<1.14.0.dev2019030116,>=1.14.0.dev2019030115
  Downloading tf_estimator_nightly-1.14.0.dev2019030115-py2.py3-none-any.whl (411 kB)
    |████████████████████████████████| 411 kB 1.9 MB/s
Requirement already satisfied: astor>=0.6.0 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow==2.0.0-alpha0) (0.6.2)
Collecting numpy<2.0,>=1.14.5
```

# ATUALIZANDO A BIBLIOTECA

- pip install tensorflow -U

```
mac:~ coelho$ pip install tensorflow --upgrade
Collecting tensorflow
  Downloading tensorflow-2.2.0-cp36-cp36m-macosx_10_11_x86_64.whl (175.3 MB)
    |████████| 175.3 MB 30.0 MB/s
Requirement already satisfied, skipping upgrade: wrapt>=1.11.1 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (1.11.2)
Requirement already satisfied, skipping upgrade: termcolor>=1.1.0 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (1.1.0)
Collecting astunparse==1.6.3
  Downloading astunparse-1.6.3-py2.py3-none-any.whl (12 kB)
Requirement already satisfied, skipping upgrade: google-pasta>=0.1.8 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (0.2.0)
Requirement already satisfied, skipping upgrade: protobuf>=3.8.0 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (3.12.1)
Collecting opt-einsum>=2.3.2
  Downloading opt_einsum-3.2.1-py3-none-any.whl (63 kB)
    |████████| 63 kB 5.3 MB/s
Collecting scipy==1.4.1; python_version >= "3"
  Downloading scipy-1.4.1-cp36-cp36m-macosx_10_6_intel.whl (28.5 MB)
    |████████| 28.5 MB 31.9 MB/s
Collecting tensorboard<2.3.0,>=2.2.0
  Downloading tensorboard-2.2.1-py3-none-any.whl (3.0 kB)
    |████████| 3.0 kB 1.6 MB/s
Requirement already satisfied, skipping upgrade: grpcio>=1.8.6 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (1.12.0)
Requirement already satisfied, skipping upgrade: keras-preprocessing>=1.1.0 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (1.1.2)
Requirement already satisfied, skipping upgrade: wheel>=0.26; python_version >= "3" in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from tensorflow) (0.30.0)
Collecting h5py<2.11.0,>=2.10.0
  Downloading h5py-2.10.0-cp36-cp36m-macosx_10_6_intel.whl (3.0 kB)
    |████████| 3.0 kB 25.8 MB/s
```

# COMO VERIFICAR SE FOI INSTALADO?

```
1 import tensorflow as tf
2
3 # Testando se a instalação do Tensorflow foi bem sucedida.
4
5 def main():
6     msg = tf.constant('Hello, TensorFlow!')
7     tf.print(msg)
8
9 if __name__ == '__main__':
10    main()

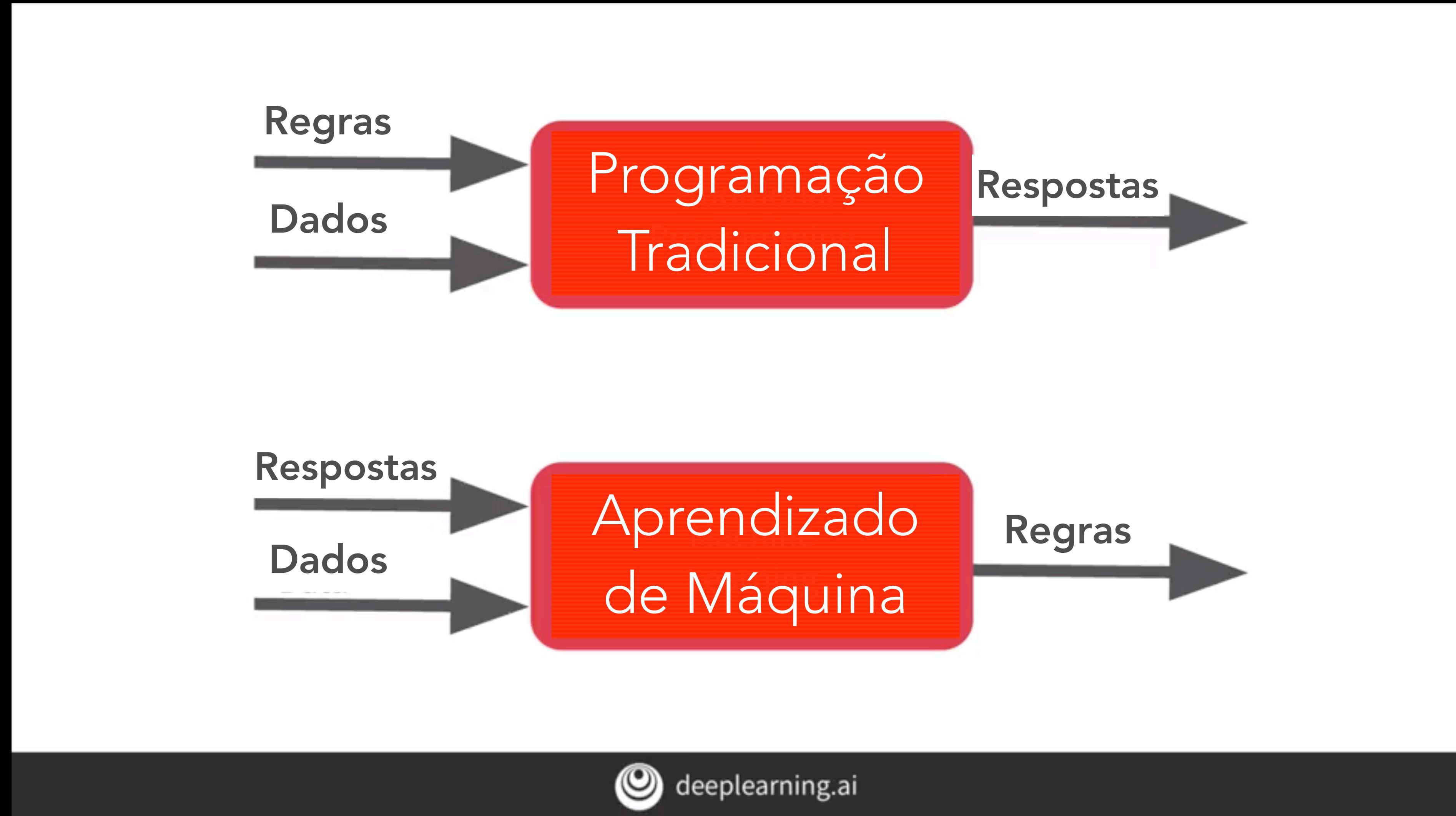
2020-05-27 20:18:18.251908: I tensorflow/core/platform/cpu_feature_guard.cc:143] Your CPU
supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2020-05-27 20:18:18.279591: I tensorflow/compiler/xla/service/service.cc:168] XLA service
0x13d3566b0 initialized for platform Host (this does not guarantee that XLA will be used).
Devices:
2020-05-27 20:18:18.279622: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor
device (0): Host, Default Version
Hello, TensorFlow!

-----
(program exited with code: 0)
Press return to continue
```

# PROJETO DE APRENDIZADO DE MÁQUINA

1. Olhar para o quadro geral;
2. Obter os dados;
3. Descobrir e visualizar os dados para obter informações;
4. Preparar os dados para os algoritmos;
5. Selecionar e treinar um modelo;
6. Ajustar o modelo;
7. Apresentar sua solução;
8. Lançar, monitorar e manter seu sistema.

# NOVO PARADIGMA DE PROGRAMAÇÃO



## EXEMPLO BÁSICO

- Qual o raciocínio por traz da sequência de números Y que é gerado a partir dos números X?

$$\begin{aligned}X &= -1, \quad 0, \quad 1, \quad 2, \quad 3, \quad 4 \\Y &= -3, \quad -1, \quad 1, \quad 3, \quad 5, \quad 7\end{aligned}$$

## EXEMPLO BÁSICO

- Qual o raciocínio por traz da sequência de números Y que é gerado a partir dos números X?

X = -1,	0,	1,	2,	3,	4
Y = -3,	-1,	1,	3,	5,	7

```
float hw_function(float x){  
    float y = (2 * x) - 1;  
    return y;  
}
```

## 1) CRIA O MODELO SEQUENCIAL COM 1 CAMADA (1 NEURÔNIO) E 1 DADO DE ENTRADA

```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1]))
```

OBS: KERAS É UMA BIBLIOTECA IMPLEMENTADA EM PYTHON E RODA UMA CAMADA ACIMA DO TENSORFLOW. É AMIGÁVEL, DE FÁCIL EXTENSÃO E AMIGÁVEL.



# INSTALANDO A BIBLIOTECA KERAS

```
>Last login: Wed May 27 20:18:12 on ttys001
You have new mail.
[mac:~ coelho$ pip install keras -U
Collecting keras
  Downloading Keras-2.3.1-py2.py3-none-any.whl (377 kB)
    |████████████████████████████████| 377 kB 182 kB/s
Requirement already satisfied, skipping upgrade: pyyaml in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (3.12)
Requirement already satisfied, skipping upgrade: six>=1.9.0 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (1.15.0)
Requirement already satisfied, skipping upgrade: keras-preprocessing>=1.0.5 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (1.1.2)
Requirement already satisfied, skipping upgrade: scipy>=0.14 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (1.4.1)
Requirement already satisfied, skipping upgrade: keras-applications>=1.0.6 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (1.0.8)
Requirement already satisfied, skipping upgrade: h5py in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (2.10.0)
Requirement already satisfied, skipping upgrade: numpy>=1.9.1 in /Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages (from keras) (1.18.4)
Installing collected packages: keras
  Attempting uninstall: keras
    Found existing installation: Keras 2.1.6
    Uninstalling Keras-2.1.6:
      Successfully uninstalled Keras-2.1.6
Successfully installed keras-2.3.1
```

## 2) DEFINE A FORMA COMO VAI SER CALCULADO O ERRO (MÉDIO QUADRÁTICO)

```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1]))  
model.compile(optimizer='sgd', loss='mean_squared_error')
```



### 3) INICIALIZA OS DADOS DE ENTRADA COMO DOIS ARRAYS

```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1]))  
model.compile(optimizer='sgd', loss='mean_squared_error')  
  
xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype=float)  
ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0], dtype=float)
```



## BIBLIOTECA NUMPY

- **NumPy** é uma biblioteca Python usada largamente para operações com listas/arrays. Você pode pensar em um array NumPy como um tipo especial de lista com uma série de funções adicionais.

Para criar um array NumPy você deve especificar o tamanho e um método de inicialização. Veja um exemplo:

```
import numpy as np  
nparray = np.zeros((5,5))
```

<https://docs.scipy.org/doc/numpy/reference/>

```
[[0.  0.  0.  0.  0.]  
 [0.  0.  0.  0.  0.]  
 [0.  0.  0.  0.  0.]  
 [0.  0.  0.  0.  0.]  
 [0.  0.  0.  0.  0.]]
```

## 4) DEFINE O MODELO (EXEMPLOS E ÉPOCAS) PARA O TREINAMENTO

```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1]))  
model.compile(optimizer='sgd', loss='mean_squared_error')  
  
xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype=float)  
ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0], dtype=float)  
  
model.fit(xs, ys, epochs=500)
```



## 5) FAZ A PREDIÇÃO DO MODELO PARA O VALOR DE ENTRADA 10

```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1]))  
model.compile(optimizer='sgd', loss='mean_squared_error')  
  
xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype=float)  
ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0], dtype=float)  
  
model.fit(xs, ys, epochs=500)  
  
print(model.predict([10.0]))
```



# EXECUTANDO A REDE (PREDIÇÃO PARA O VALOR 10)

```
/usr/local/bin/python3.7 /Users/coelho/Desktop/tensorflow_examples/tensorflow1.py
2020-05-25 09:54:46.520818: I tensorflow/core/platform/cpu_feature_guard.cc:143] Your CPU supports instructions that this TensorFlow version was not compiled to use: AVX2
2020-05-25 09:54:46.539364: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x7fbad9b7d4a0 initialized for platform Host (this process)
2020-05-25 09:54:46.539382: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default Version
Epoch 1/500
1/1 [=====] - 0s 327us/step - loss: 6.5086
Epoch 2/500
1/1 [=====] - 0s 245us/step - loss: 5.3074
Epoch 3/500
1/1 [=====] - 0s 246us/step - loss: 4.3585
Epoch 4/500
1/1 [=====] - 0s 362us/step - loss: 3.6082
Epoch 5/500
1/1 [=====] - 0s 249us/step - loss: 3.0142
Epoch 6/500
1/1 [=====] - 0s 285us/step - loss: 2.5433
Epoch 7/500
1/1 [=====] - 0s 245us/step - loss: 2.1692
Epoch 8/500
1/1 [=====] - 0s 261us/step - loss: 1.8715
Epoch 9/500
1/1 [=====] - 0s 264us/step - loss: 1.6339
Epoch 10/500
1/1 [=====] - 0s 263us/step - loss: 1.4436
Epoch 11/500
1/1 [=====] - 0s 269us/step - loss: 1.2907
Epoch 12/500
1/1 [=====] - 0s 894us/step - loss: 1.1672
Epoch 13/500
1/1 [=====] - 0s 275us/step - loss: 1.0669
Epoch 14/500
```

The screenshot shows the PyCharm IDE interface with the project 'tensorflow\_examples' open. The 'Run' tool window is active, displaying the output of the script 'tensorflow1.py'. The output shows the training progress of a neural network, with the loss value decreasing from approximately 6.5 to 1.0669 over 14 epochs. The terminal also displays system logs and TensorFlow compilation information.

# EXECUTANDO A REDE (PREDIÇÃO PARA O VALOR 10)

The screenshot shows the PyCharm IDE interface with the following details:

- Title Bar:** tensorflow\_examples [~/Desktop/tensorflow\_examples] - .../tensorflow1.py
- Project View:** Shows a single file named tensorflow1.py.
- Run Tab:** Set to "tensorflow1".
- Output Window:** Displays the execution log of the script. The log shows the training process for 500 epochs, with each epoch taking approximately 1ms per step. The loss value decreases from 4.0538e-05 to 3.0799e-05. The final prediction output is shown in red: array([[18.983807]], dtype=float32).
- Status Bar:** Shows the following information: Python Console, Terminal, Run, TODO, Event Log, Packages installed successfully: Installed packages: 'tensorflow' (3 minutes ago), 1008:1 LF UTF-8 4 spaces Python 3.7 (2), and 336 of 990M.

# CÓDIGO-FONTE COMPLETO DO EXEMPLO

The screenshot shows the Geany IDE interface with the file `tensorflow1.py` open. The window title is `tensorflow1.py - /Users/coelho/Desktop/tensorflow_examples - Geany`. The toolbar includes icons for New, Open, Save, Undo, Redo, Back, Forward, Compile, Build, Execute, Color Selector, Find, Go To, and Exit. The code editor displays the following Python script:

```
1 from tensorflow import *
2 import numpy as np
3
4 """
5 Problem:
6 Y = 2 * X - 1
7 """
8
9 def main():
10    # cria o modelo sequencial com 1 camada (1 neurônio) e 1 dado de entrada
11    model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1])])
12    model.compile(optimizer='sgd', loss='mean_squared_error')
13
14    # inicializa os dados de entrada como dois arrays
15    xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype=float)
16    ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0], dtype=float)
17
18    # define o modelo (exemplos e épocas para treinamento)
19    model.fit(xs, ys, epochs=500)
20
21    # faz a predição do modelo para o valor de entrada 10
22    print(model.predict([10.0]))
23
24
25 if __name__ == '__main__':
26    main()
```

## EXPERIMENTO UM POUCO!

- A. Altere o valor de entrada
  - B. Diminua bastante o número de épocas e execute o código
  - C. Aumente bastante o número de épocas e execute o código
- 
- **O que aconteceu em cada teste?**

## LINKS ÚTEIS

- <https://github.com/lmoroney/dlaicourse>
- <https://github.com/lmoroney/dlaicourse/blob/master/Course%201%20-%20Part%202%20-%20Lesson%202%20-%20Notebook.ipynb>
- <https://www.tensorflow.org>
- <https://www.youtube.com/tensorflow>

## EXERCÍCIO

1) Você deve criar uma rede neural que prevê preços de casas baseado em uma simples equação linear.

Uma casa de 1 quarto custa 100k.

Uma casa de 2 quartos custa 150k.

Uma casa de 3 quartos custa 200k.

...

Como você resolveria este problema com o Tensorflow?

## EXERCÍCIO

Dica: Para uma casa de 7 quartos (predição para entrada da rede como 7.0), o preço deve ser estimado em [[4.0088816]] com 1000 épocas de treinamento.

```
import tensorflow as tf
import numpy as np
from tensorflow import keras
model = # Your Code Here#
model.compile(# Your Code Here#)
xs = # Your Code Here#
ys = # Your Code Here#
model.fit(# Your Code here#)
print(model.predict([7.0]))
```

DÚVIDAS?

RAFAELVC2@GMAIL.COM

