

Tutorial-1

Cristian Muñoz.

Instalação Tensorflow



1. Com GPU:

- Instalação de CUDA:

Site: <https://developer.nvidia.com/cuda-downloads>

2. Instalar miniconda:

Site: <https://conda.io/miniconda.html>

3. Criar um ambiente em miniconda para tensorflow:

Comando: `conda create -n tensorflow python=3.5 anaconda`

Comando: `activate tensorflow`

4. Procurar na rede a ultima versão de Tensorflow:

Site: <https://www.tensorflow.org/install/>

Comando: `pip install --ignore-installed --upgrade <path_tensorflow>`

Para começar...

- Jupyter:
 - activate tensorflow
 - cd <pasta do tutorial>
 - jupyter notebook



Sumario

- Tipos de variáveis
 - Constant
 - Placeholders
 - Variable
- Criando e visualizando um modelo de aprendizagem
- Exercício: Treinar o modelo

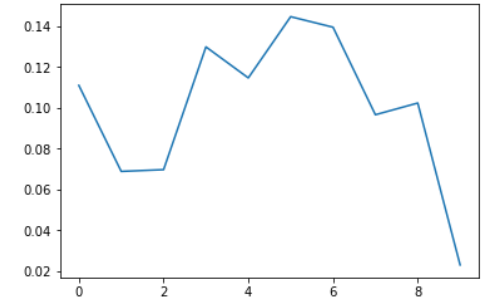
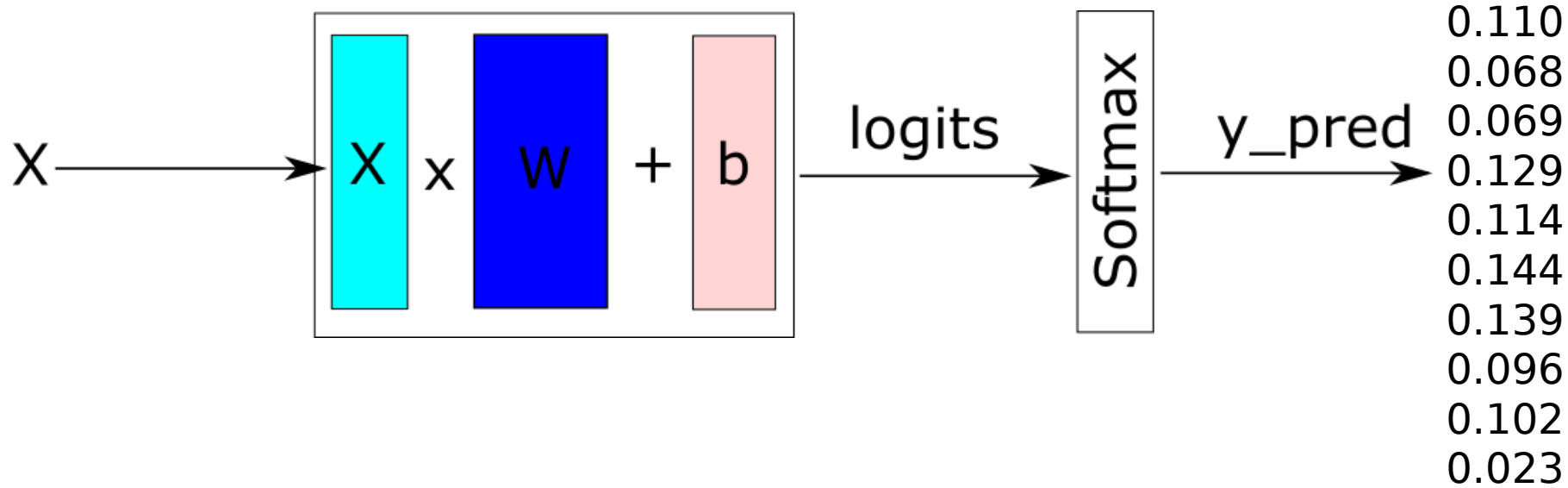
Sumario

- Tipos de variáveis
 - Constant
 - Placeholders
 - Variable
- Criando e visualizando um modelo de aprendizagem
- Exercício: Treinar o modelo

Sumario

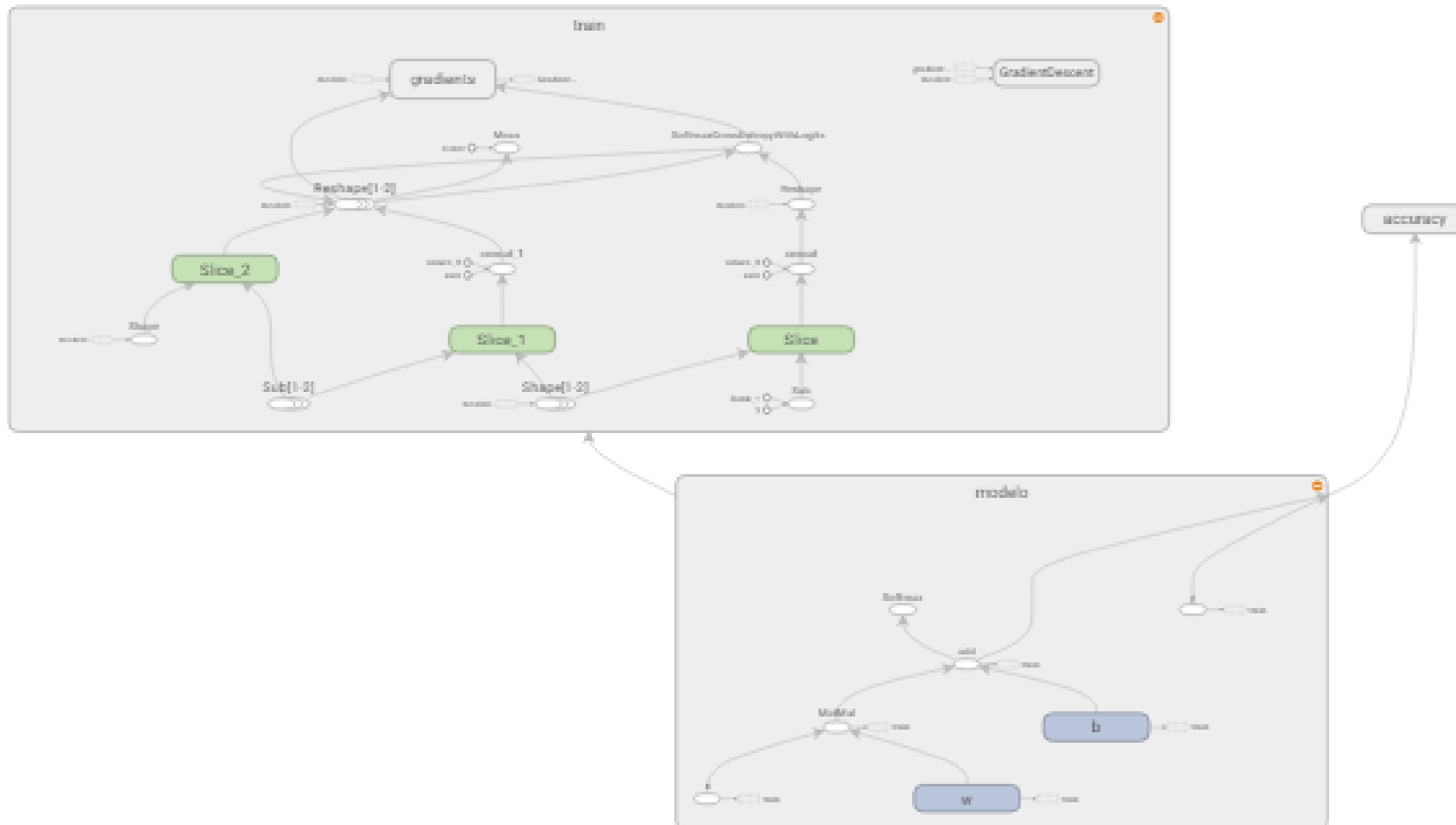
- Tipos de variáveis
 - Constant
 - Placeholders
 - Variable
- Criando e visualizando um modelo de aprendizagem
- Exercício: Treinar o modelo

Criando um modelo de aprendizagem



logits: Unscaled log probabilities

Melhorando a visualização



Visualização do modelo de aprendizagem

- Podemos encapsular processos com o comando `with tf.name_scope('modelo'):`

.....

- Também podemos rotular algumas variáveis adicionando o atributo *name*:

```
w = tf.Variable( ..... , name= 'w')
```

Sumario

- Tipos de variáveis
 - Constant
 - Placeholders
 - Variable
- Criando e visualizando um modelo de aprendizagem
- Exercício: Treinar o modelo

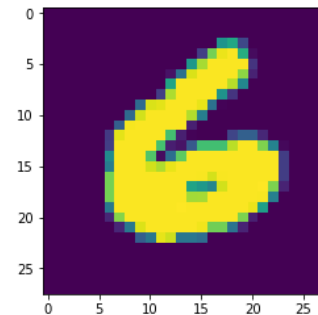
Exercício

- Em Jupyter importamos os módulos de python que precisamos para este tutorial:

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
from tensorflow.examples.tutorials.mnist import input_data
```

- Abrir a base de dados MNIST e visualizar uma imagem aleatória:

```
data = input_data.read_data_sets("data/mnist", one_hot=True)
img_flat = data.train.images[18]
img = np.reshape(img_flat, (28,28))
plt.imshow(img)
```



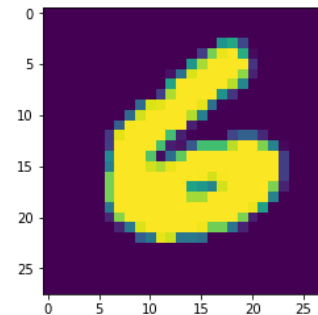
Exercício

- Em Jupyter importamos os módulos de python que precisamos para este tutorial:

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
from tensorflow.examples.tutorials.mnist import input_data
```

- Abrir a base de dados MNIST e visualizar uma imagem aleatória:

```
data = input_data.read_data_sets("data/mnist", one_hot=True)
img_flat = data.train.images[18]
img = np.reshape(img_flat, (28,28))
plt.imshow(img)
```



Exercício

- Criar um *session* para treinar nosso algoritmo de aprendizagem:

```
session = tf.Session()  
session.run(tf.global_variables_initializer())
```

- Criar métodos que ajudem a treinar e a medir acurácia:

```
def print_accuracy():  
    acc = session.run(accuracy, feed_dict={x:data.test.images, y: data.test.labels})  
    print("accuracy: %f" % acc)
```

```
def train(iteration=1):  
    for _ in range(iteration):  
        images , labels = data.train.next_batch(batch_size=128)  
        session.run(optimizer, feed_dict={x:images, y: labels})
```

Exercício

- Analisar o desempenho inicial da rede
- Treinar 1 iteração e analisar desempenho
- Treinar 10 iterações e analisar desempenho
- Treinar 1000 iterações e analisar desempenho