

# Instalando os pacotes necessários para implementar sua primeira Rede Neural

## Introdução

Este documento é um tutorial que ensinará você a instalar os pacotes [NumPy](#)<sup>1</sup>, [Pandas](#)<sup>2</sup>, [Scikit-learn](#)<sup>3</sup> e [PyTorch](#)<sup>4</sup>, em Python, usando o PIP. Esses são os pacotes que você precisará para o Tutorial de *Como Implementar sua primeira Rede Neural usando PyTorch*.

## Usando o PIP

Nesse tutorial, usaremos o PIP, um administrador de módulos e pacotes para Python, que permite a instalação, desinstalação e atualização fácil de pacotes de *software*. O PIP está disponível para diferentes versões do Python e desde as últimas versões ele é o administrador de pacotes oficial da linguagem e já vem pré-instalado junto com o Python.

Tanto esse quanto o próximo tutorial foram preparados e testados usando a versão 3.6 do Python. Por isso sugerimos que você também utilize essa versão. Para a versão 3.6 do Python, o comando adequado para executar o PIP é:

```
1 pip3.6
```

Ou, simplesmente:

```
1 pip3
```

Caso o Python 3.6 seja a única versão do Python instalada na sua máquina, talvez o seguinte comando funcione:

```
1 pip
```

Caso mais de um desses comandos funcione na sua máquina, de preferência para eles na ordem em que foram apresentados aqui.

---

<sup>1</sup><https://numpy.org/>

<sup>2</sup><https://pandas.pydata.org/>

<sup>3</sup><https://scikit-learn.org/stable/>

<sup>4</sup><https://pytorch.org/>

Se você quiser instalar esses pacotes numa versão diferente do Python, você pode encontrar em [2] os comandos do PIP para outras versões.

## Instalando o NumPy

O Numpy é um pacote de processamento numérico eficiente para Python. Vários pacotes a seguir têm o ele como requisito para suas instalações e, por isso, é melhor começar por ele. Para instalar o NumPy usando o PIP, use o comando:

```
1 pip3.6 install numpy
```

Caso você esteja usando outro dos comandos apresentados anteriormente para chamar o PIP, não esqueça de substituí-lo aqui.

Caso você esteja usando um sistema linux e queira que a instalação valha para todos usuários, talvez você precise concatenar o comando sudo antes do comando do PIP e fornecer sua senha de administrador quando pedido.

## Instalando o pandas

O Pandas é um pacote de processamento, manipulação e análise de dados. Para instalá-lo, use o seguinte comando:

```
1 pip3.6 install pandas
```

## Instalando o scikit-learn

Já o scikit-learn é um pacote de computação científica que fornece uma série de ferramentas úteis para o aprendizado de máquina.

Para instalá-lo, faça:

```
1 pip3.6 install scikit-learn
```

Um detalhe importante é que, normalmente, após instalado o scikit-learn passa a responder pelo nome de sklearn, então para importá-lo para o seu código em python você deve fazer:

```
1 import sklearn
```

## Instalando o PyTorch

Já o PyTorch é um pacote para a implementação de modelos tensoriais e neurais e, atualmente, é uma das principais ferramentas para a implementação de redes neurais.

Placas de processamento gráfico (GPUs) são arquiteturas de *hardware* especialmente eficientes para o treinamento de Redes Neurais. Por esse motivo, o Pytorch oferece compatibilidade para diferentes versões de Placas Gráficas com CUDA. Cada uma dessas versões exige um comando de instalação próprio, por isso, vamos tentar percorrer os principais deles.

Se você não tem uma GPU instalada no seu computador, e pretende usar apenas os processadores comuns, você pode instalar o PyTorch com o seguinte comando:

```
1 pip3.6 install torch==1.5.0+cpu torchvision==0.6.0+cpu -f
2 https://download.pytorch.org/whl/torch_stable.html
```

Se você pretende usar uma GPU com CUDA 9.2, use o seguinte comando:

```
1 pip3.6 install torch==1.5.0+cu92 torchvision==0.6.0+cu92 -f
2 https://download.pytorch.org/whl/torch_stable.html
```

Se você pretende usar uma GPU com CUDA 10.1, faça:

```
1 pip3.6 install torch==1.5.0+cu101 torchvision==0.6.0+cu101 -f
2 https://download.pytorch.org/whl/torch_stable.html
```

E, por fim, se você pretende usar uma GPU com CUDA 10.2, faça:

```
1 pip3.6 install torch torchvision
```

Você pode encontrar outras opções, além dessas, em [1].

Após instalado, o PyTorch passa a responder simplesmente pelo nome de torch. Para testar sua instalação, você pode executar o seguinte programa:

```
1 from __future__ import print_function
2 import torch
3 x = torch.rand(5, 3)
4 print(x)
```

Para o qual você deveria receber uma resposta parecida com essa [1]:

```
1 tensor([[0.3380, 0.3845, 0.3217],
2         [0.8337, 0.9050, 0.2650],
```

```
3 [0.2979, 0.7141, 0.9069],  
4 [0.1449, 0.1132, 0.1375],  
5 [0.4675, 0.3947, 0.1426]])
```

## Referências

- [1] PyTorch Team. Start locally. <https://pytorch.org/get-started/locally/>, 2020. Acessado: 05/06/2020.
- [2] Michigan Technological University. Installing, uninstalling, or upgrading python modules in linux using pip. <https://servicedesk.mtu.edu/TDClient/1801/Portal/KB/ArticleDet?ID=66715>, 2020. Acessado: 05/06/2020.