

# TP555 - AI/ML

## Lista de Exercícios #0

Utilize o Jupyter para resolver os exercícios desta lista. Siga as instruções do link a seguir caso você queira instalá-lo em seu computador pessoal:

<https://docs.anaconda.com/anaconda/install/>

1. Crie um repositório no github com seu nome seguido de seu número de matrícula mais a palavra tp555. Exemplo: felipe-124103-tp555. Este repositório servirá para que você versione seus exercícios/trabalhos e os entregue para avaliação. Dentro do repositório, crie uma pasta com o nome lista0 e dentro desta pasta guarde os notebooks com os códigos dos exercícios. Faça o mesmo para todas as outras listas de exercícios. Ao final desta lista, me envie o link para o seu repositório.
2. Execute cada um dos exemplos dos slides. Crie um notebook Jupyter para cada um deles. No Windows, digite Jupyter na barra de buscas e selecione Jupyter Notebook. No Linux, abra um terminal e digite `jupyter notebook`.
3. Neste exercício você irá plotar um gráfico 2D. Este tipo de gráfico é comumente utilizado para se analisar os dados de entrada e saída de um modelo de aprendizado de máquina. Crie um vetor coluna,  $y$ , com  $M = 1000$  elementos, onde  $y$  é dado pela seguinte equação

$$y = 1.2 + 2.3 \cdot x + 10 \cdot w,$$

onde  $x$  é um vetor coluna com  $M$  elementos retirados de uma distribuição aleatória uniforme e  $w$  é um vetor coluna com  $M$  elementos retirados de uma distribuição aleatória Gaussiana normal, i.e., com média 0 e variância unitária.

- a. Plote um gráfico com os vetores  $x$  e  $y$  sendo os eixos  $x$  e  $y$ , respectivamente.

**Dica:**

- Use o módulo `random` da biblioteca `numpy` para gerar números aleatórios  
<https://docs.scipy.org/doc/numpy-1.14.0/reference/routines.random.html>
- Use a biblioteca `matplotlib` para plotar gráficos  
[https://matplotlib.org/3.1.3/gallery/lines\\_bars\\_and\\_markers/simple\\_plot.html](https://matplotlib.org/3.1.3/gallery/lines_bars_and_markers/simple_plot.html)

4. Neste exercício você vai plotar o histograma de um vetor criado através da soma de variáveis aleatórias. Histogramas são utilizados para se verificar a distribuição de um determinado conjunto de dados. Crie um vetor coluna  $x$  com  $M = 10000$  amostras retiradas de uma distribuição aleatória uniforme. Em seguida, crie outro vetor coluna  $y$ , também com  $M = 10000$  amostras retiradas de uma distribuição aleatória uniforme. Na sequência, obtenha o vetor  $z$ , que é definido pela seguinte equação

$$z = x + y.$$

- a. Plote o histograma normalizado de  $z$ .

**Dica:**

- Use o método hist da biblioteca matplotlib  
[https://matplotlib.org/3.1.3/api/\\_as\\_gen/matplotlib.pyplot.hist.html](https://matplotlib.org/3.1.3/api/_as_gen/matplotlib.pyplot.hist.html)  
[https://matplotlib.org/3.1.3/gallery/statistics/histogram\\_features.html](https://matplotlib.org/3.1.3/gallery/statistics/histogram_features.html)
5. Neste exercício você irá plotar um gráfico 3D. Este tipo de gráfico pode ser utilizado para visualizar superfícies de erro, as quais são comumente encontradas em problemas de otimização. Crie 2 vetores, x1 e x2, respectivamente, com valores uniformemente espaçados entre -10 e 11 com passos de 0.25 unidades. Em seguida crie o vetor y, o qual é definido pela seguinte equação
- $$y = x1^2 + x2^2.$$
- a. Plote um gráfico 3D com x1, x2 e y sendo plotados nos eixos x, y e z, respectivamente.

**Dica:**

- Use o método arange da biblioteca numpy para gerar valores uniformemente espaçados com passos predefinidos.  
<https://docs.scipy.org/doc/numpy/reference/generated/numpy.arange.html>
- Estude o seguinte exemplo para entender como plotar gráficos 3D  
<https://matplotlib.org/3.1.3/gallery/mplot3d/surface3d.html>

**Referências**

- [1] Python.org, “BeginnersGuide”, <https://wiki.python.org/moin/BeginnersGuide/Programmers>  
[2] Mark Pilgrim, “Dive into Python”, <https://diveintopython3.problemsolving.io/>  
[3] Nerd Paradise, “4 Minute Python Crash Course”  
<https://nerdparadise.com/programming/python4minutes/>