

# Matplotlib and Random Cheat Sheet

By Michelle Cristina de Sousa Baltazar

# Biblioteca Random

Para usar a biblioteca random, primeiro é necessário importá-la. No início do programa inserimos:

*from random import \**

Também podemos rodar o comando `help(random)` no interpretador python para ver quais funções a biblioteca `random` fornece:

```
$ python
```

```
from random import random
```

```
?? help(random)
```

O código em `randomOps.py` contém alguns exemplos das funções mais úteis desta biblioteca:

|                              |   |
|------------------------------|---|
| <i>random()</i> :            | obtem o próximo número aleatório no intervalo [0.0, 1.0]    |
| <i>random(começo, fim)</i> : | obter o próximo número aleatório no intervalo [começo, fim] |
| <i>random(stop)</i> :        | obtem o próximo número aleatório no intervalo [0, fim]      |

# Biblioteca Matplotlib

Para usar a biblioteca Matplotlib, comece importando estes módulos Python:

```
import numpy as np
```

```
import pandas as pd
```

```
from pandas import DataFrame, Series
```

```
import matplotlib.pyplot as plt
```

```
import matplotlib
```

**Pyplot** é uma coleção de funções no estilo de comandos que fazem a biblioteca `matplotlib` funcionar como o `MatLab`. Cada função `pyplot` faz alguma alteração na plotagem do gráfico.

## Exemplo básico Matplotlib:

Exemplo básico de plotagem de gráfico:

```
import matplotlib.pyplot as plt
plt.plot([1,2,3,4])
plt.ylabel('Números de Exemplo')
plt.show()
```

Neste exemplo, foi gerado um valor para Y baseado no valor de X informado.

## Matplotlib com dois eixos:

Podemos também informar o valor dos dois eixos.

```
import matplotlib.pyplot as plt
plt.plot([1,2,3,4], [1,4,9,16], 'ro')
plt.axis([0, 6, 0, 20])
plt.show()
```

Neste caso, temos os 2 eixos mais o terceiro argumento opcional em formato de string *'ro'* que indica a cor e o tipo de linha da plotagem (vide quadro a seguir).

A linha *ptl.axix* mostra quais são os pontos que deverão ser marcados no gráfico.

## Matplotlib - Comandos de Texto Básicos

Os comandos a seguir são usados para criar texto na interface Pyplot:

|                         |   |
|-------------------------|---|
| <code>text()</code>     | adiciona texto em um local específico dos eixos.      |
| <code>xlabel()</code>   | adiciona uma legenda para o eixo x.                   |
| <code>ylabel()</code>   | adiciona uma legenda para o eixo y.                   |
| <code>title()</code>    | adiciona um título para os eixos.                     |
| <code>figtext()</code>  | adiciona texto em um local específico da figura.      |
| <code>suptitle()</code> | adiciona um título à figura.                          |
| <code>annotate()</code> | adiciona uma anotação ao eixos com uma seta opcional. |

All of these functions create and return a `matplotlib.text.Text()` instance, which can be configured with a variety of font and other properties.

## Matplotlib - Propriedades para Plotagem

| Propriedade               | Tipo de Valor   |
|---------------------------|---|
| alpha                     | float   |
| animated                  | [True — False]  |
| antialiased or aa         | [True — False]  |
| clip_box                  | uma instância matplotlib.transforms.Bbox                            |
| clip_on                   | [True — False]  |
| clip_path                 | uma instancia de caminho e uma instancia de transformação, um Patch |
| color or c                | qualquer cor matplotlib   |
| contains                  | a função de teste de acertos  |
| dash_capstyle             | ['final' — 'turno' — 'projeção']                                    |
| solid_capstyle            | ['final' — 'turno' — 'projeção']                                    |
| dash_joinstyle            | ['topo' — 'turno' — 'corte']  |
| solid_joinstyle           | ['topo' — 'turno' — 'corte']  |
| dashes                    | sequencia de liga/desliga cor nos pontos                            |
| data                      | (np.arranjo dadox, np.arranjo da-doy)                               |
| figure                    | uma instância matplotlib.image.Figure                               |
| label                     | qualquer string   |
| linestyle or ls           | ['-', '-', '-', '.', ':', '-.', -passos'-...]                       |
| linewidth or lw           | valores tipo float nos pontos                                       |
| lod                       | [True — False]  |
| marker                    | ['+', ',', '. ', .', '1'—'2'—'3'—'4']                               |
| markeredgewidth<br>or mec | qualquer cor matplotlib   |
| markedgewidth<br>or mew   | valor tipo float value nos pontos                                   |
| markerfacecolor<br>or mfc | qualquer cor matplotlib   |
| markersize or ms          | float   |
| markevery                 | [ nada — inteiro — (startind,<br>stride) ]                          |
| picker                    | usado na seleção da linha interativa                                |
| pickradius                | a amplitude de seleção da linha escolhida                           |
| transform                 | uma instância matplotlib.transforms.Transform                       |
| visible                   | [True — False]  |
| xdata                     | np.arranjo  |
| ydata                     | np.arranjo  |
| zorder                    | qualquer número   |