

## Data Science com Python 3

### O Algoritmo K-Means

#### *Exercício para entrega*

1. Implemente a seguinte função. Ela deve devolver uma lista de comprimento  $n$  que contém tuplas no formato  $(x, y)$  com  $x, y \in \mathbb{Z}$ . Cada tupla representa a localização da residência de um funcionário da DataSciescenter.  $x$  representa a quantidade de quarteirões ao norte e  $y$  representa a quantidade de quarteirões ao leste em relação ao centro da cidade, da residência de um funcionário. A função deve gerar uma lista com as seguintes restrições.

1.1 Para o primeiro terço da lista:  $-50 \leq x \leq -40$  e  $0 \leq y \leq 10$ .

1.2 Para o segundo terço da lista:  $-45 \leq x \leq -35$  e  $5 \leq y \leq 20$ .

1.3 Para o último terço da lista:  $20 \leq x \leq 30$  e  $20 \leq y \leq 30$ .

```
def gera_base(n):  
    #sua implementação aqui
```

2. Implemente a seguinte função. Ela recebe uma lista tal qual aquela gerada pela função do exercício 1 e exibe, usando um gráfico de dispersão (scatter), os pontos em um plano. Cada terço deve ser exibido com uma cor e símbolo diferente, já que cada um deles representa uma vizinhança diferente.

```
def exibe_grafico(base):
```

3. Implemente a seguinte função. Trata-se de uma adaptação da função do exercício 2. Ela deve executar o algoritmo KMeans a fim de obter os representantes de cada grupo. Use  $k=3$ . Ao final, ela deve exibir todos os elementos da base em um gráfico de dispersão, bem como os representantes de cada um deles obtidos pelo KMeans. Os representantes devem ser exibidos com o símbolo “+”.

```
def exibe_grafico_e_representantes(base):
```

4 (**Desafio, entrega não obrigatória**) Para cada grupo obtido, desenhe um círculo com centro no seu representante e cujo raio seja suficiente para englobar todas as suas instâncias.