



Questão 1

Ainda não respondida

Vale 2,00 ponto(s).

Considere os dados gerados pelo código abaixo. Determine o número ideal de clusters usando a medida normalized mutual information. Use o métodos k-mean para encontrar os clusters.

```
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
import numpy as np
np.random.seed(10)

import numpy as np
import matplotlib.pyplot as plt
# create blobs
k = 5
n=200
data = make_blobs(n_samples=n, n_features=2, centers=k, cluster_std=1.5, random_state=50)
X = data[0]
labels = data[1]
plt.scatter(X[:,0], X[:,1], c=labels, cmap='viridis', s=50, alpha=0.9)
plt.xlim(-15,15)
plt.ylim(-15,15)
plt.show(True)
```

Escolha uma:

- ☐ a. 8
- ☐ b. 1
- ☒ c. 5
- ☐ d. 10
- ☐ e. 2

[Limpar minha escolha](#)

Questão 2

Ainda não respondida

Vale 2,00 ponto(s).

Aplique o método k-means aos dados gerados com o código abaixo. Calcule a medida Silhouette score. Selecione a alternativa mais próxima.

```
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
np.random.seed(10)

import numpy as np
import matplotlib.pyplot as plt
# create blobs
k = 5
n=200
data = make_blobs(n_samples=n, n_features=2, centers=k, cluster_std=1.5, random_state=50)
X = data[0]
labels = data[1]
plt.figure(figsize=(8,6))
plt.scatter(X[:,0], X[:,1], c=labels, cmap='viridis', s=50, alpha=0.9)
plt.xlim(-15,15)
plt.ylim(-15,15)
plt.show(True)
```

Escolha uma:

- ☐ a. 0.2
- ☐ b. 1.0
- ☐ c. 0.9
- ☒ d. 0.5
- ☐ e. 0.1

[Limpar minha escolha](#)

Questão **3**

Ainda não respondida

Vale 2,00 ponto(s).

Aplique o algoritmo aglomerativo hierárquico com o método de Ward nos dados gerados com o código abaixo. Calcule a medida Mutual information. Selecione a alternativa mais próxima.

```
from sklearn import cluster, datasets
from sklearn.cluster import KMeans
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(10)

# create blobs
k = 2
n=200
#data = make_blobs(n_samples=n, n_features=2, centers=k, cluster_std=1.5, random_state=50)
data = datasets.make_moons(n_samples=n, noise=.05)
X = data[0]
labels = data[1]
plt.scatter(X[:,0], X[:,1], c=labels, cmap='viridis', s=50, alpha=0.9)
plt.show(True)
```

Escolha uma:

- ☒ a. 1.0
- ☐ b. 0.8
- ☐ c. 0.9
- ☐ d. 0.1
- ☐ e. 0.6

[Limpar minha escolha](#)

Questão **4**

Ainda não respondida

Vale 2,00 ponto(s).

Determine o número ideal de clusters para a base Iris usando o método k-means e a medida normalized mutual informaton.

```
from scipy.spatial.distance import cdist
from sklearn import datasets
import numpy as np
np.random.seed(10)

iris = datasets.load_iris()
X = iris.data[:, :2] # we only take the first two features.
y = iris.target
```

Escolha uma:

- ☐ a. 10
- ☐ b. 2
- ☒ c. 3
- ☐ d. 1
- ☐ e. 5

[Limpar minha escolha](#)

Questão **5**

Ainda não respondida

Vale 2,00 ponto(s).

Determine o número ideal de clusters para a base Iris usando o método de agrupamento hierárquico (Ward) e a medida normalized mutual information.

Escolha uma:

- ☐ a. 1
- ☒ b. 2
- ☐ c. 10
- ☐ d. 4
- ☐ e. 5

[Limpar minha escolha](#)