

Questão 1

Resposta salva

Vale 2,00
ponto(s).

Mude a função seno para cosseno (no exemplo da aula) e verifique qual o grau do polinômio mais adequado para 20 observações (10 no conjunto de teste e 10 no conjunto de treinamento). Considere o código abaixo para gerar os dados.

```
import numpy as np
from matplotlib import pyplot as plt

np.random.seed(101)

def function(x):
    y = np.cos(2 * np.pi * x)
    return y

# training set
N_train = 10
x_train=np.linspace(0, 1,N_train)
y_train = function(x_train) + np.random.normal(0,0.1, N_train)

fig = plt.figure(figsize=(8, 6))
plt.scatter(x_train, y_train, facecolor="none", edgecolor="b", s=100, label="training data")

# test set
N_test = 10
x_test=np.linspace(0, 1,N_test)
y_test = function(x_test) + np.random.normal(0,0.1, N_test)

# Theoretical curves
xt=np.linspace(0,1,100)
yt = function(xt)
plt.plot(xt,yt, '-r', label="Theoretical curve")
plt.legend()
plt.show(True)
```

Escolha uma:

- ☐ a. 8
- ☐ b. 1
- ☐ c. 2
- ☐ d. 9
- ☒ e. 4

[Limpar minha escolha](#)

Questão **2**
Resposta salva
Vale 2,00
ponto(s).

Mude a função seno para x^4 (no exemplo da aula) e verifique qual grau do polinômio mais adequado para 20 observações (10 no conjunto de teste e 10 no conjunto de treinamento). Use o código abaixo para gerar os dados:

```
import numpy as np
from matplotlib import pyplot as plt
np.random.seed(101)
def function(x):
    y = x**4
    return y
# training set
N_train = 10
x_train=np.linspace(0, 1,N_train)
y_train = function(x_train) + np.random.normal(0,0.1, N_train)
fig = plt.figure(figsize=(8, 6))
plt.scatter(x_train, y_train, facecolor="none", edgecolor="b", s=100, label="training data")
# test set
N_test = 10
x_test=np.linspace(0, 1,N_test)
y_test = function(x_test) + np.random.normal(0,0.1, N_test)
# Theoretical curves
xt=np.linspace(0,1,100)
yt = function(xt)
plt.plot(xt,yt, '-r', label="Theoretical curve")
plt.legend()
plt.show(True)
```

Escolha uma:

- ☐ a. 1
- ☐ b. 3
- ☐ c. 4
- ☐ d. 8
- ☒ e. 2

[Limpar minha escolha](#)

Questão **3**

Resposta salva

Vale 2,00
ponto(s).

Considere 100 pontos (50 no conjunto de treinamento e 50 no conjunto de teste) gerados a partir da função cosseno com erro. Qual o grau mais adequado do polinômio? Considere o código abaixo.

```
import numpy as np
from matplotlib import pyplot as plt
np.random.seed(101)

def function(x):
    y = np.cos(2 * np.pi * x)
    return y

# training set
N_train = 50
x_train=np.linspace(0, 1,N_train)
y_train = function(x_train) + np.random.normal(0,0.1, N_train)

fig = plt.figure(figsize=(8, 6))
plt.scatter(x_train, y_train, facecolor="none", edgecolor="b", s=100, label="training data")

# test set
N_test = 50
x_test=np.linspace(0, 1,N_test)
y_test = function(x_test) + np.random.normal(0,0.1, N_test)
# Theoretical curves
xt=np.linspace(0,1,100)
yt = function(xt)
plt.plot(xt,yt, '-r', label="Theoretical curve")
plt.legend()
plt.show(True)
```

Escolha uma:

- ☐ a. 12
- ☐ b. 14
- ☒ c. 4
- ☐ d. 5
- ☐ e. 8

[Limpar minha escolha](#)

Questão **4**

Resposta salva

Vale 2,00
ponto(s).

Considerando a base Vehicle, encontre o melhor número de vizinhos. Qual é a acurácia aproximada usando validação cruzada com 10 folds, usando o melhor número de vizinhos encontrado? Repita o que foi feito na aula para a base iris.

Escolha uma:

- ☐ a. 0.3
- ☒ b. 0.6
- ☐ c. 1.0
- ☐ d. 0.2
- ☐ e. 0.9

[Limpar minha escolha](#)

Questão **5**

Resposta salva

Vale 2,00
ponto(s).

Quais são os principais ingredientes em um problema de aprendizado supervisionado?

Escolha uma:

- ☐ a. conjunto de dados, validação cruzada e modelo
- ☐ b. conjunto de dados, normalização e projeção.
- ☒ c. conjunto de dados, modelo e função de custo.
- ☐ d. modelo, função de custo e seleção de atributos.
- ☐ e. nenhuma das alternativas desse exercício.

[Limpar minha escolha](#)

