



Inteligência Artificial e Big Data

Aula 06

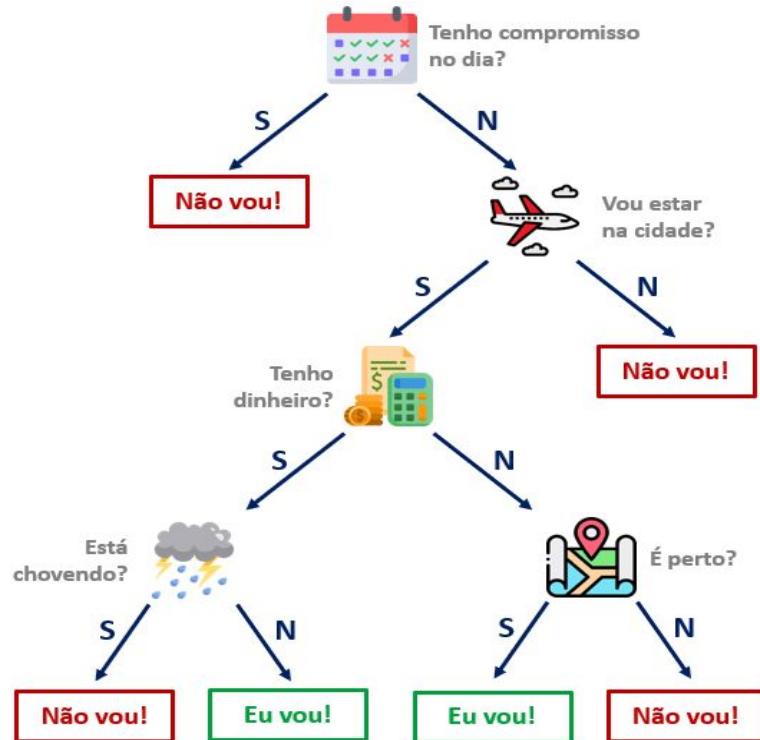
Prof. Me Daniel Vieira

Agenda

- 1- Árvore de decisão - Classificação
- 2- Árvore de decisão
- 3- Estudo de caso
- 4- Matriz de confusão
- 5- Acurácia, precisão, recall, F1 - Score
- 6- Exercícios

Árvore de decisão

É um algoritmo de aprendizado supervisionado que cria uma estrutura na forma de árvore para tomar decisões. Cada nó interno representa um teste em uma característica, cada ramo representa um resultado possível desse teste e cada folha representa uma classe ou valor de saída



Exercícios

1) Criar um script para classificar uma bebida como boa, ruim, péssima a partir das notas dos clientes

```
notas = 8, 6, 5, 9, 4, 3, 7, 2, 1, 5, 6, 9, 8] # notas dos clientes (0-10)
classes = 'boa', 'ruim', 'ruim', 'boa', 'péssima', 'péssima', 'boa', 'péssima',
'péssima', 'ruim', 'ruim', 'boa', 'boa']
```

Exercício 1

```
#dividir dados em teste e treino
notas_treino, notas_teste, classes_treino, classes_teste = train_test_split(notas.reshape(-1,1),
classes.reshape(-1,1), test_size=0.2, random_state=42)

modelo = DecisionTreeClassifier()
modelo.fit(notas_treino,classes_treino)

accuracy = accuracy_score(classes_teste,previsoes)
print(accuracy)

fig = plt.figure(figsize=(10,8))
tree.plot_tree(modelo,feature_names= notas.tolist(), class_names = classes.tolist(), filled= True)
```

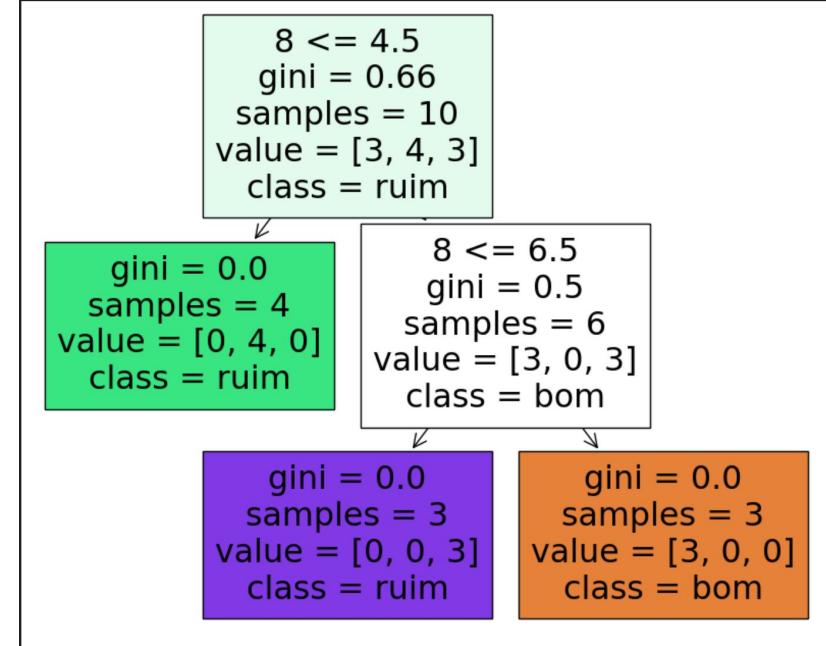
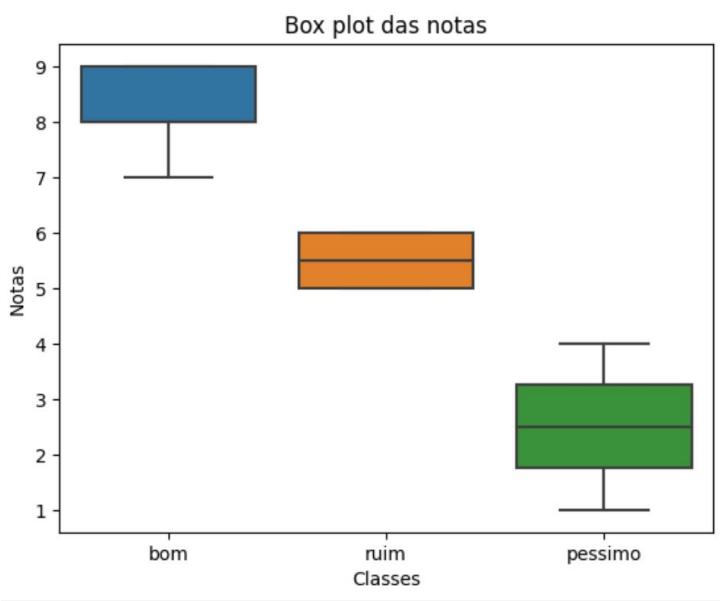
Exercício 1

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn import tree
import matplotlib.pyplot as plt
import seaborn as sns

#Dados treinamento
notas = np.array([8,6,5,9,4,3,7,2,1,5,6,9,8])
classes = np.array(['bom','ruim','ruim','bom','pessimo','pessimo',
    'bom','pessimo', 'pessimo','ruim','ruim','bom','bom'])
```

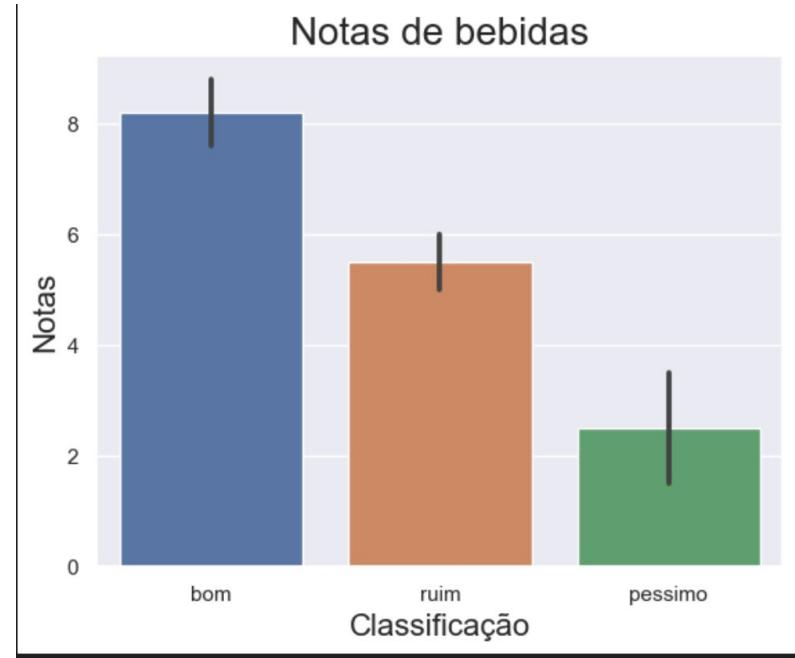
Exercício 1

```
sns.boxplot(x = classes, y = notas)  
plt.title('Box plot das notas')  
plt.ylabel('Notas')  
plt.xlabel('Classes')
```



Exercício 1

```
sns.barplot(y=notas, x = classes)
sns.set(font_scale=1)
plt.title('Notas de bebidas', fontsize=20)
plt.xlabel('Classificação', fontsize=16)
plt.ylabel('Notas', fontsize=16)
```



Matriz de confusão

		Classe esperada	
		Gato	Não é gato
Classe prevista	Gato	25 Verdadeiro Positivo	10 Falso Positivo
	Não é gato	25 Falso Negativo	40 Verdadeiro Negativo

Matriz de confusão

Neste exemplo teríamos a soma de tudo que o algoritmo acertou (sejam eles verdadeiros positivos ou verdadeiros negativos) dividido pelo total de amostras, o que daria em nosso exemplo, $65/100 = 0,65$ ou 65% .

Precisão: de todos os dados classificados como positivos, quantos são realmente positivos.

$$\text{Precisão} = \frac{\text{Verdadeiros Positivos (TP)}}{\text{Verdadeiros Positivos (TP)} + \text{Falsos Positivos (FP)}}$$

No exemplo acima teríamos: $25/(25+10) = 0,71$ ou 71% de precisão.

Ou seja, conseguimos acertar com precisão, que 71% das imagens classificadas como gatos realmente eram gatos.

DICA: em um precisão de 1.0 ou 100% significa que não houve nenhum falso positivo.

Matriz de confusão

Recall: qual a porcentagem de dados classificados como positivos comparado com a quantidade real de positivos que existem em nossa amostra.

$$\text{Recall} = \frac{\text{Verdadeiros Positivos (TP)}}{\text{Verdadeiros Positivos (TP)} + \text{Falsos Negativos (FN)}}$$

No exemplo acima teríamos um recall de $25/(25+25) = 0,5$ ou 50%.

Ou seja, nosso modelo conseguiu acertar apenas 50% dos gatos presentes na amostra.

DICA: em um recall de 1.0 ou 100% significa que não houve nenhum falso negativo.

Matriz de confusão

F1-score: essa métrica une precisão e recall afim de trazer um número único que determine a qualidade geral do nosso modelo.

$$F1 = \frac{2 * \text{precisão} * \text{recall}}{\text{precisão} + \text{recall}}$$

No exemplo acima teríamos um F1-score de $2 * (0,71 * 0,5) / (0,71 + 0,5) = 0,58$
ou 58%

Resumo:

Acurácia: qual a proporção de gatos e não gatos que foram corretamente classificados.

Precisão: qual a proporção dos dados classificados como gatos eram realmente gatos. (aqui buscamos os falsos positivos)

Recall: entre todas as amostras que realmente eram de gatos, qual a proporção classificada como gatos. (aqui buscamos os falsos negativos)

F1-score: uma maneira de observar em um único número a precisão e o recall.

Métricas para avaliar a árvore de decisão (Acurácia)

$$Acurácia = \frac{Acertos\,(A)}{Acertos\,(A) + Erros\,(E)}$$

Métricas para avaliar a árvore de decisão (Acurácia)

$$Acurácia = \frac{Verdadeiros Positivos (VP) + Verdadeiros Negativos (VN)}{Total}$$

$$\begin{aligned} Total = & \text{ } Verdadeiros Positivos (VP) + Verdadeiros Negativos (VN) \\ & + Falsos Positivos (FP) + Falsos Negativos (FN) \end{aligned}$$

Métricas para avaliar a árvore de decisão (Acurácia)

$$Acurácia = \frac{Verdadeiros Positivos (VP) + Verdadeiros Negativos (VN)}{Total}$$

$$\begin{aligned} Total = & \text{ } Verdadeiros Positivos (VP) + Verdadeiros Negativos (VN) \\ & + Falsos Positivos (FP) + Falsos Negativos (FN) \end{aligned}$$

Métricas para avaliar a árvore de decisão (Acurácia)

		Preditos	
		Falha	Não falha
Real	Falha	15	10
	Não falha	25	50

		Valor previsto	
		Positivo	Negativo
Valor Real	Positivo	Verdadeiros Positivos	Falsos Negativos
	Negativo	Falsos Positivos	Verdadeiros Negativos

$$\text{Acurácia} = \frac{\text{Verdadeiros Positivos (VP)} + \text{Verdadeiros Negativos (VN)}}{\text{Total}}$$

$$\text{Acurácia} = \frac{15 + 50}{15 + 50 + 25 + 10} = 65\%$$

Métricas para avaliar a árvore de decisão (Recall)

$$Revocação = \frac{Verdadeiros Positivos (VP)}{Verdadeiros Positivos (VP) + Falsos Negativos (FN)}$$

$$Revocação = \frac{15}{15 + 10} = 60\%$$

Métricas para avaliar a árvore de decisão (F1-Score)

$$F1 - Score = \frac{2 * Precisão * Revocação}{Precisão + Revocação}$$

$$F1 - Score = \frac{2 * 0,375 * 0,6}{0,375 + 0,6} = 46,15\%$$

Métricas para avaliar a árvore de decisão (R2-Score)

R2 score - coeficiente de determinação, indica o quanto próximo a reta da previsão está do valor real do conjunto de dados

```
#importar a biblioteca para calcular a métrica r2_score  
from sklearn.metrics import r2_score
```

```
r2_lr = r2_score(y_teste, previsao_lr)  
r2_lr
```

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Métricas para avaliar a árvore de decisão (MSE)

MSE (Mean Squared Error ou Erro Quadrático Médio): média da diferença elevada ao quadrado entre o valor real e o previsto. (penalidade sobre o erro)

```
from sklearn.metrics import mean_squared_error
```

```
y_true = [[0.5, 1],[-1, 1],[7, -6]]  
y_pred = [[0, 2],[-1, 2],[8, -5]]
```

```
mean_squared_error(y_true, y_pred, squared=True)
```

O erro quadrático médio, MSE (da sigla em inglês Mean Squared Error), é comumente usado para verificar a acurácia de modelos e dá um maior peso aos maiores erros, já que, ao ser calculado, cada erro é elevado ao quadrado individualmente e, após isso, a média desses erros quadráticos é calculada.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

Quando utilizar cada métrica ?



- **Acurácia:** indica uma performance geral do modelo. Dentre **todas** as classificações, quantas o modelo classificou corretamente;
- **Precisão:** dentre todas as classificações de classe Positivo **que o modelo fez**, quantas estão corretas;
- **Recall/Revocação/Sensibilidade:** dentre todas as situações de classe Positivo **como valor esperado**, quantas estão corretas;
- **F1-Score:** média harmônica entre precisão e recall.

Quando utilizar cada métrica ?

A **acurácia** é uma boa indicação geral de como o modelo performou. Porém, pode haver situações em que ela é enganosa. Por exemplo, na criação de um modelo de identificação de fraudes em cartões de crédito, o número de casos considerados como fraude pode ser bem pequeno em relação ao número de casos considerados legais. **Para colocar em números, em uma situação hipotética de 280000 casos legais e 2000 casos fraudulentos, um modelo simplório que simplesmente classifica tudo como legal obteria uma acurácia de 99,3%.** Ou seja, você estaria validando como ótimo um modelo que falha em detectar fraudes.

Quando utilizar cada métrica ?

A **precisão** pode ser usada em uma situação em que os Falsos Positivos são considerados mais prejudiciais que os Falsos Negativos. Por exemplo, ao classificar uma ação como um bom investimento, é necessário que o modelo esteja correto, mesmo que acabe classificando bons investimentos como maus investimentos (situação de Falso Negativo) no processo. Ou seja, o modelo deve ser preciso em suas classificações, pois a partir do momento que consideramos um investimento bom quando na verdade ele não é, uma grande perda de dinheiro pode acontecer.

Quando utilizar cada métrica ?

O ***recall*** pode ser usada em uma situação em que os **Falsos Negativos são considerados mais prejudiciais que os Falsos Positivos**. Por exemplo, o modelo deve de qualquer maneira encontrar todos os pacientes doentes, mesmo que classifique alguns saudáveis como doentes (situação de Falso Positivo) no processo. Ou seja, o modelo deve ter alto *recall*, pois classificar pacientes doentes como saudáveis pode ser uma tragédia.

Quando utilizar cada métrica ?

O **F1-Score** é simplesmente uma maneira de observar somente 1 métrica ao invés de duas (precisão e *recall*) em alguma situação. É uma média harmônica entre as duas, que **está muito mais próxima dos menores valores do que uma média aritmética simples**. Ou seja, quando tem-se um F1-Score baixo, é um indicativo de que ou a precisão ou o *recall* está baixo.

Situações de aprendizagem

1) Suponha que você está trabalhando em uma fábrica de produtos eletrônicos que produz dispositivos móveis, como smartphones e tablets. Você coletou dados sobre diferentes máquinas usadas na produção e deseja classificá-las como "Máquinas de Montagem" ou "Máquinas de Teste" com base em suas características. As características incluem a velocidade de operação, a complexidade das tarefas que executam e a quantidade de manutenção necessária. Use uma árvore de decisão para classificar as máquinas.

*

Velocidade de Operação	Complexidade da Tarefa	Manutenção Necessária	Classificação
10	Baixa	Baixa	Montagem
5	Alta	Alta	Teste
8	Média	Média	Montagem
6	Alta	Alta	Teste
12	Baixa	Baixa	Montagem
4	Alta	Média	Teste

Situações de aprendizagem

2) Imagine que você trabalha em uma usina nuclear e precisa classificar as máquinas industriais em duas categorias: "Seguras" e "Não Seguras". Você coletou dados sobre características das máquinas, como idade, histórico de manutenção, número de falhas anteriores e nível de automação. Use uma árvore de decisão para determinar se uma máquina é segura ou não com base nessas características.

Idade (anos)	Histórico de Manutenção	Número de Falhas Anteriores	Nível de Automação	Classificação
5	Bom	0	Alto	Segura
10	Ruim	3	Baixo	Não Segura
3	Excelente	0	Médio	Segura
8	Regular	2	Alto	Não Segura
1	Excelente	0	Médio	Segura
15	Ruim	5	Baixo	Não Segura

Situações de aprendizagem

3) Você é um engenheiro de qualidade em uma fábrica de automóveis e deseja classificar as máquinas de produção de acordo com sua contribuição para a qualidade dos produtos finais. As características incluem a precisão na montagem, a velocidade de produção e a taxa de retrabalho. Crie uma árvore de decisão para classificar as máquinas em "Alta Qualidade" e "Baixa Qualidade". *

Precisão na Montagem	Velocidade de Produção	Taxa de Retrabalho	Classificação
Alta	Média	Baixa	Alta Qualidade
Média	Baixa	Alta	Baixa Qualidade
Alta	Alta	Baixa	Alta Qualidade
Média	Alta	Baixa	Alta Qualidade
Baixa	Baixa	Alta	Baixa Qualidade
Baixa	Média	Alta	Baixa Qualidade

Situações de aprendizagem

4) Em uma planta de produção industrial, você deseja classificar as máquinas com base em sua eficiência energética. As características incluem o consumo de energia, o tempo de operação e o tipo de energia utilizada (por exemplo, elétrica ou a gás). Use uma árvore de decisão para classificar as máquinas em "Eficientes" e "Ineficientes" em termos de uso de energia.

Consumo de Energia (kWh)	Tempo de Operação (horas)	Tipo de Energia	Classificação
1000	200	Elétrica	Eficiente
3000	500	Gás	Ineficiente
1500	300	Elétrica	Eficiente
2500	400	Gás	Ineficiente
1200	250	Elétrica	Eficiente
3500	600	Gás	Ineficiente

Situações de aprendizagem

5) Suponha que você é responsável por determinar quando realizar a manutenção preventiva em diferentes máquinas em uma fábrica de processamento de alimentos. As características incluem a idade das máquinas, o número de horas de operação desde a última manutenção e o histórico de falhas. Use uma árvore de decisão para classificar as máquinas em "Necessidade de Manutenção" e "Não Necessidade de Manutenção".

Idade (anos)	Horas Desde a Última Manutenção	Histórico de Falhas	Classificação
2	50	Nenhuma	Necessita
6	200	2	Necessita
1	20	Nenhuma	Necessita
4	150	1	Não Necessita
3	100	3	Necessita
5	180	2	Não Necessita

Situações de aprendizagem

<https://docs.google.com/forms/d/1UzxpeNe9WNpYZ8huLn0LTURVFI0HQ0dcxw2WBWQkNvY/edit>

Obrigado!

Prof. Me Daniel Vieira

Email: danielvieira2006@gmail.com

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95

