Lista 1 - Árvore de decisão

Vitor Lucio

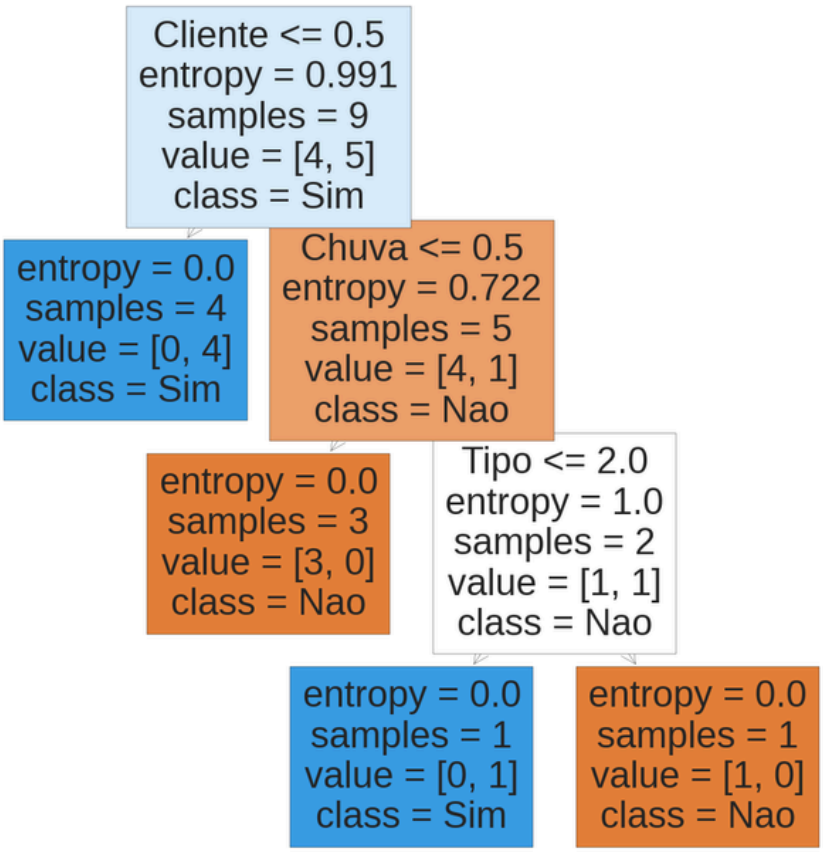
Questão 01.01

Exemplo	Alternativ	Bar	Sex/Sab	fome	Cliente	Preço	Chuva	Res	Tipo	Tempo	conc
X1	Sim	Não	Não	Sim	Alguns	RRR	Não	Sim	Francês	0-10	Sim
x2	Sim	Não	Não	Sim	Cheio	R	Não	Não	Tailandês	30-60	Não
x3	Não	Sim	Não	Não	Alguns	R	Não	Não	Hamburge	0-10	Sim
x4	Sim	Não	Sim	Sim	Cheio	R	Sim	Não	Tailandês	out/30	Sim
X5	Sim	Não	Sim	Não	Cheio	RRR	Não	Sim	Francês	>60	Não
X6	Não	Sim	Não	Sim	Alguns	RR	Sim	Sim	Italiano	0-10	Sim
X7	Não	Sim	Não	Não	Nenhum	R	Sim	Não	Hamburge	0-10	Não
X8	Não	Não	Não	Sim	Alguns	RR	Sim	Sim	Tailandês	0-10	Sim
X9	Não	Sim	Sim	Não	Cheio	R	Sim	Não	Hamburge	>60	Não
X10	Sim	Sim	Sim	Sim	Cheio	RRR	Não	Sim	Italiano	out/30	1
X11	Não	Não	Não	Não	Nenhum	R	Não	Não	Tailandês	0-10	Base
X12	Sim	Sim	Sim	Sim	Cheio	R	Não	Não	Hamburge	30-60	Sim
Ganho	0	0	0.02	0.195	0.541	0.196	0.02	0.02	0	0.207	1
					RAIZ						

Questão 01.02

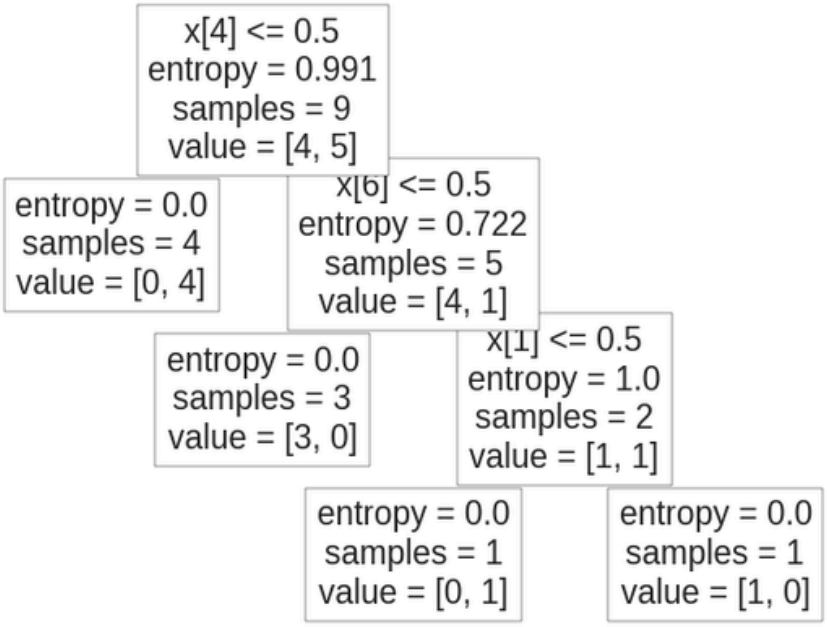
Exemplo	Alternativ	Bar	Sex/Sab	fome	Cliente	Preço	Chuva	Res	Tipo	Tempo	conc
x2	Sim	Não	Não	Sim	Cheio	R	Não	Não	Tailandês	30-60	Não
x4	Sim	Não	Sim	Sim	Cheio	R	Sim	Não	Tailandês	out/30	Sim
X5	Sim	Não	Sim	Não	Cheio	RRR	Não	Sim	Francês	>60	Não
X9	Não	Sim	Sim	Não	Cheio	R	Sim	Não	Hamburge	>60	Não
X10	Sim	Sim	Sim	Sim	Cheio	RRR	Não	Sim	Italiano	out/30	Não
X12	Sim	Sim	Sim	Sim	Cheio	R	Não	Não	Hamburge	30-60	Sim
Ganho	0.109	0	0.109	0.251	0.918	0.251	0.043	0.251	0.251	0.251	
				RAIZ 2	BASE						

Questão 02.01



Questão 02.02

Apesar das mudanças nos atributos, não notei nenhuma diferença relevante, sendo que os ganhos dos dois primeiros níveis permanecem iguais. Crio que isso ocorre devido ao pequeno tamanho da base de dados.



Questão 02.03

```
params = {
   'criterion':  ['gini', 'entropy'],
   'max_depth':  [None, 2, 4, 6, 8, 10],
   'max_features': [None, 'sqrt', 'log2', 0.2, 0.4, 0.6, 0.8],
}
```

- {'criterion': 'entropy', 'max_depth': 10, 'max_features': 0.6} 0.9499999999999998
- {'criterion': 'gini', 'max_depth': 10, 'max_features': 0.4} 0.9583333333333333
- {'criterion': 'gini', 'max_depth': 10, 'max_features': 0.8} 0.9666666666666666

```
params = {
  'criterion': ['gini', 'entropy'],
  'max_depth': [None, 2, 4, 6, 8, 10],
  'max_features': [None, 'sqrt', 'log2', 0.2, 0.4, 0.6, 0.8],
  'min_samples_split': [2, 5, 10, 20],
  'min_samples_leaf': [1, 2, 4, 6],
  'max_leaf_nodes': [None, 5, 10, 20, 30],
}
```

- {'criterion': 'gini', 'max_depth': 4, 'max_features': 'log2', 'max_leaf_nodes': 30, 'min_samples_leaf': 6, 'min_samples_split': 20} 0.9666666666666666
- {'criterion': 'gini', 'max_depth': 2, 'max_features': 'log2', 'max_leaf_nodes': 30, 'min_samples_leaf': 2, 'min_samples_split': 10} 0.9666666666666666

Questão 03.01

Diferenças entre os algoritmos ID3 e C4.5:

- **Tratamento de Dados Contínuos**: O ID3 trabalha apenas com atributos nominais, enquanto o C4.5 é capaz de lidar com atributos de valores contínuos. O C4.5 faz isso dividindo os dados em intervalos durante a construção da árvore.

- **Manejo de Valores Desconhecidos**: O ID3 não lida bem com valores ausentes ou desconhecidos, já o C4.5 pode lidar com esses valores ao estimar a informação baseada apenas nos registros onde o valor do atributo é conhecido.

- **Critério de Seleção de Atributos**: O ID3 utiliza o ganho de informação puro como critério para selecionar os atributos que irão compor os nós da árvore. O C4.5, por sua vez, utiliza a razão de ganho, que normaliza o ganho de informação para evitar que atributos com muitos valores possíveis dominem a seleção.

Questão 03.02

O C4.5 gerencia atributos numéricos contínuos ao transformar os valores contínuos em atributos discretos durante a construção da árvore. Ele faz isso ordenando os valores dos atributos e, em seguida, encontra o ponto de corte ótimo que maximiza o ganho de informação ou a razão de ganho. Para cada possível ponto de corte, ele divide os dados em dois subconjuntos: um contendo valores menores ou iguais ao ponto de corte e outro contendo valores maiores. O ponto de corte que maximiza o ganho de informação é escolhido para dividir o nó.