# Ficha 9 - DC

João Nunes (A82300)
Luís Braga (A82088)

17/04/2020

## Conteúdo

# 1 Parte I

## 1.1 Quais as características dos atributos de um dataset que podem levá-lo a escolher uma metodologia de Data Mining de árvore de decisão, em vez de uma abordagem de regressão linear? Porquê?

Se os atributos de um dataset forem de natureza categórica ou se o conjunto de dados for misto devemos usar uma metodologia de árvore de decisão. Isto porque as árvores de decisão têm a capacidade de tratar de forma eficaz atributos que têm valores em falta ou que são inconsistentes.

## 1.2 Para que servem as percentagens de confiança, e por que razão é importante considerá-las, para além de considerar apenas o atributo de previsão?

É importante ter em consideração a percentagem de confiança para além de somente o atributo de previsão, porque a percentagem de confiança traduz o quão confiantes podemos estar na previsão feita. Assim, sabemos a probabilidade de aquela previsão ser a que irá na realidade acontecer.

## 1.3 Como é que é possível manter um atributo, como o nome ou número de identificação de uma pessoa, que não deve ser considerado como preditivo no modelo de um processo, mas que é útil ter nos resultados de Data Mining?

Manter um atributo como o nome ou número de identificação é útil, pois permite realizar um Data Mining mais personalizado, isto é, tomando como exemplo o caso dado na aula prática, não é apenas possível saber que tipo de estratégias de marketing temos de tomar, mas sabemos exactamente quais os alvos dessa estratégia.

## 1.4 Quais as principais vantagens apresentadas na utilização de árvores de decisão comparativamente com outras técnicas de Data Mining?

Árvores de decisão apresentam vantagens em relação a outros modelos de previsão, nomeadamente, vantagem em lidar com atributos que têm valores em falta e atributos e com atributos que são inconsistentes.

# 2 Parte II

## 2.1 Faça download do dataset "titanic-training". Importe os dados para o repositório do RapidMiner. Execute a fase de Data Understanding.

### 2.1.1 Qual foi a percentagem de passageiros sobreviventes?


Figura 1: Estatísticas relativas aos sobreviventes.

| Index | Nominal value | Absolute count | Fraction |
|---|---|---|---|
| 1 | false | 549 | 0.616 |
| 2 | true | 342 | 0.384 |

Figura 2: Detalhes dos sobreviventes.

Através da análise estatística dos dados é possível afirmar que apenas sobreviveram cerca de 38% das pessoas que embarcaram no Titanic.

### 2.1.2 Qual era a principal faixa etária dos passageiros que estavam no Titanic?

A principal faixa etária dos passageiros que se encontravam no Titanic era dos 16 aos 24 anos, seguido dos 24 aos 32 anos como se poderá verificar no seguinte histograma.

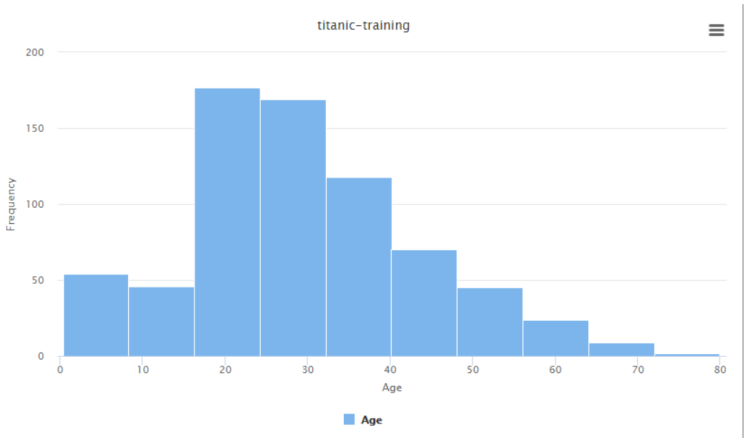


Figura 3: Faixas etárias dos passageiros no Titanic.

### 2.1.3 Sobreviveram mais crianças ou mais adultos?

No Titanic, e como se poderá verificar no seguinte histograma, sobreviveram mais adultos do que crianças em valor total (uma vez que existiam mais adultos dentro do titanic do que crianças).

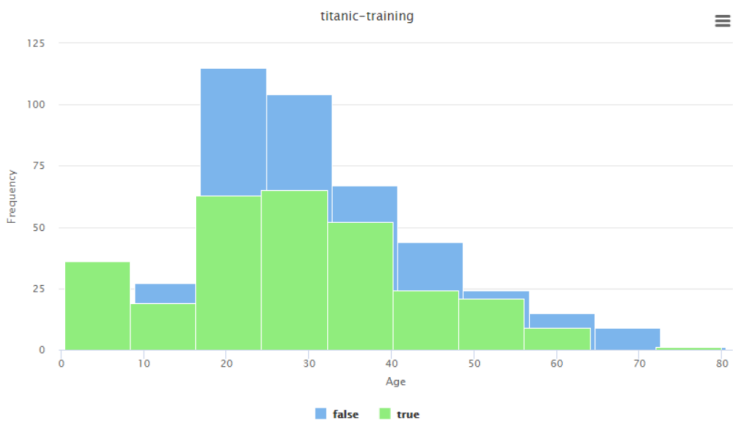


Figura 4: Faixas etárias dos sobreviventes.

## 2.2 Efectue a etapa de Data Preparation. Não se esqueça de colocar o operador Set Role nos atributos que justifiquem a sua aplicação.

No dataset de treino foi possível observar que existem 891 instâncias com 12 atributos distintos.

ExampleSet (891 examples, 0 special attributes, 12 regular attributes)

Figura 5: Informações gerais sobre o dataset.

Depois de analisar o *dataset*, foi possível identificar alguns atributos como *label* e como *id*. Ou seja, o atributo *PassengerId* irá possuir o *target role* de *id*. O *role Survived* do *dataset* de treino será o *label* e é o atributo a prever.

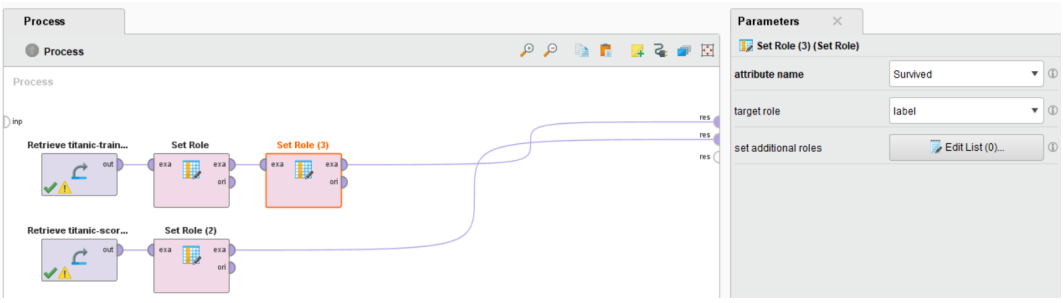


Figura 6: Modelo construído para a etapa de preparação dos dados.

## 2.3 Usando o RapidMiner, crie um primeiro processo utilizando um operador de otimização de parâmetros para descobrir valores otimizados para os parâmetros do operador de Decision Tree, tal como se encontra descrito nos slides das aulas.

Foi construído mais um modelo desta vez de otimização de parâmetros, como tal para o *dataset* de treino foi mais uma vez aplicado os dois *roles* anteriores. Após aplicar o modelo de otimização de parâmetros foi necessário criar um novo submodelo.

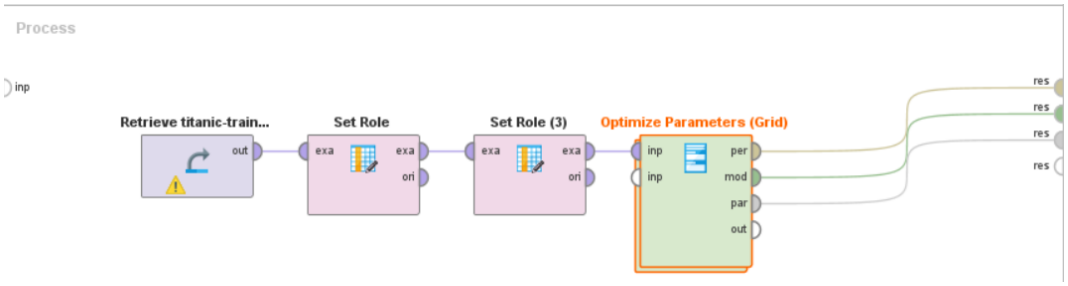


Figura 7: Modelo principal construído.

De seguida no sub processo, os dados foram divididos em treino e teste com uma proporção 70/30. Foi aplicado de seguida o modelo de *Decision Tree* com os parâmetros *standard*. Foi também adicionado um operador de *performance* que visa analisar estatisticamente a *performance* do modelo.

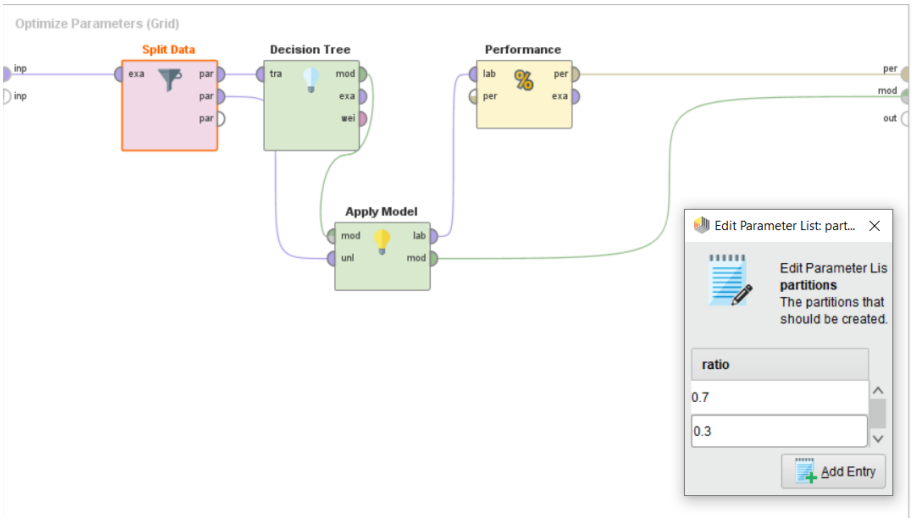


Figura 8: Sub modelo construído.

Agora sim é necessário escolher os parâmetros a otimizar do modelo *Decision Tree*, tendo em conta também que no *criterion* é necessário retirar os valores de *accuracy* e de *least_square*.

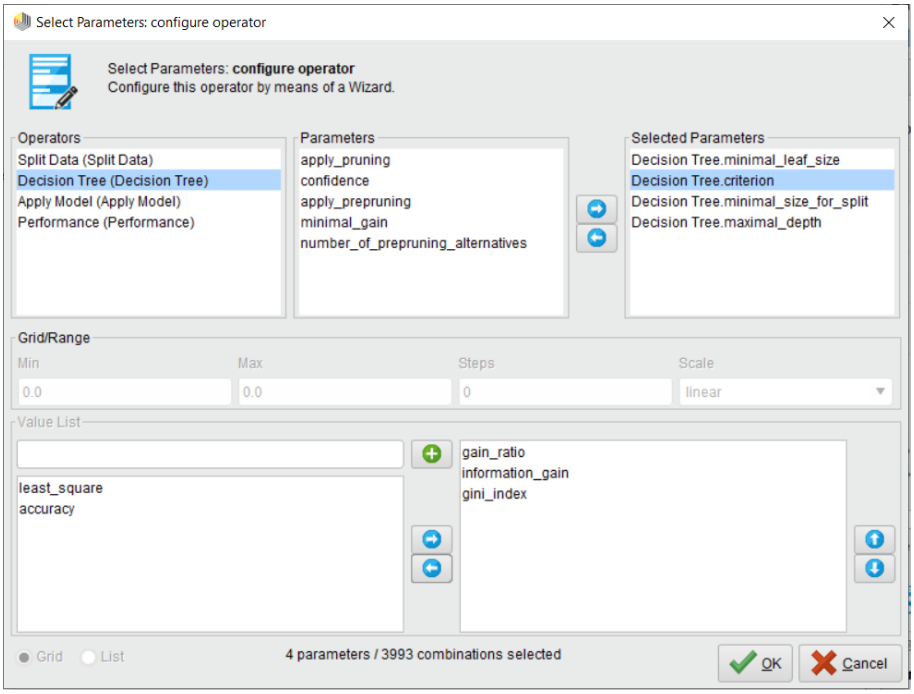


Figura 9: Parâmetros de otimização.

Após correr este modelo foi possível obter os melhores valores para cada parâmetro do modelo *Decision Tree*, os parâmetros otimizados para o algoritmo encontram-se destacados.

```
-----accuracy: 85.77%
ConfusionMatrix:
True:   false   true
false:  160     31
true:   7       69
-----precision: 90.79% (positive class: true)
ConfusionMatrix:
True:   false   true
false:  160     31
true:   7       69
-----recall: 69.00% (positive class: true)
ConfusionMatrix:
True:   false   true
false:  160     31
true:   7       69
-----AUC (optimistic): 0.921 (positive class: true)
-----AUC: 0.856 (positive class: true)
-----AUC (pessimistic): 0.791 (positive class: true)
]
Decision Tree.minimal_leaf_size = 11
Decision Tree.criterion = information_gain
Decision Tree.minimal_size_for_split    = 80
Decision Tree.maximal_depth     = 29
```

Figura 10: Melhores parâmetros do algoritmo.

## 2.4 Numa folha Excel inclua algumas pessoas no dataset de teste (titanic-scoring.csv) (pode até usar informações de pessoas que conheça). Guarde esta folha Excel como um ficheiro CSV. Importe-o para o repositório do RapidMiner.

Na folha *excel* foram adicionados duas novas entradas.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1310 | 1 | Boy, Johnr | male | 21 | 1 | 1 | 2665 | 8,222 | C105 | S |
| 1311 | 2 | Bragin, Lev | male | 22 | 0 | 1 | 34563 | 59,2 | C81 | Q |

Figura 11: Dois novos passageiros no titanic.

## 2.5 Num novo processo, repita os passos no RapidMiner tal como descritos nos slides da aula para aplicar o modelo de árvore de decisão ao dataset de teste ("titanic-scoring").

Para aplicar o modelo de árvore de decisão, foi necessário criar um modelo com dois *roles* para o *training*, incluindo o *target role* de id e *label*, bem como o próprio operador do algoritmo. No dataset de teste é necessário mais um *role* com o *target role* de id. E por fim é necessário executar o modelo com os dados de teste através do *Apply Model*.

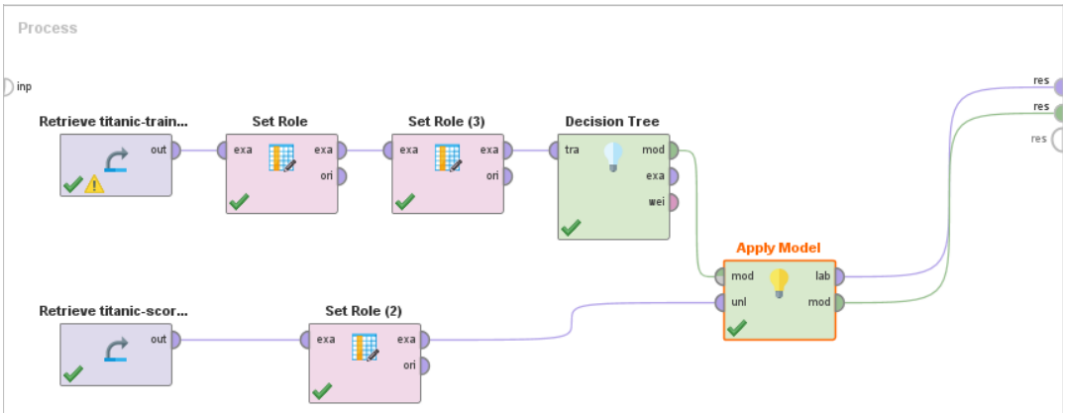


Figura 12: Modelo criado.

### 2.5.1 Execute o modelo usando os parâmetros default. Após executar o modelo, na secção dos resultados, examine as previsões e as percentagens de confiança no conjunto de teste. Relate os nós da árvore, e discuta se as pessoas que inseriu seriam sobreviventes, falecidos ou desconhecidos.

Após executar o modelo com os parâmetros *default* do operador da árvore de decisão, foi possível obter uma árvore de decisão bastante grande, tendo sido também possível observar as seguintes previsões do algoritmo para o *dataset* de teste juntamente com a confiança.

| Row No. | PassengerId | prediction(S... | confidence(f... | confidence(t... | Pclass | Name | Sex | Age | SibSp |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 892 | false | 0.896 | 0.104 | 3 | Kelly, Mr. Jam... | male | 34.500 | 0 |
| 2 | 893 | false | 0.857 | 0.143 | 3 | Wilkes, Mrs. J... | female | 47 | 1 |
| 3 | 894 | false | 0.896 | 0.104 | 2 | Myles, Mr. Th... | male | 62 | 0 |
| 4 | 895 | false | 0.896 | 0.104 | 3 | Wirz, Mr. Albert | male | 27 | 0 |
| 5 | 896 | false | 0.679 | 0.321 | 3 | Hirvonen, Mrs... | female | 22 | 1 |
| 6 | 897 | false | 0.896 | 0.104 | 3 | Svensson, Mr... | male | 14 | 0 |
| 7 | 898 | false | 1 | 0 | 3 | Connolly, Mis... | female | 30 | 0 |
| 8 | 899 | false | 0.896 | 0.104 | 2 | Caldwell, Mr. ... | male | 26 | 1 |
| 9 | 900 | true | 0.286 | 0.714 | 3 | Abrahim, Mrs.... | female | 18 | 0 |
| 10 | 901 | false | 0.896 | 0.104 | 3 | Davies, Mr. J... | male | 21 | 2 |
| 11 | 902 | false | 0.958 | 0.042 | 3 | Ilieff, Mr. Ylio | male | ? | 0 |

Figura 13: Previsão para os passageiros no dataset de teste.

No que toca aos dois novos passageiros adicionados, foi também possível verificar que o algoritmo preveu que estes não iriam sobreviver.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 419 | 1310 | false | 1 | 0 | 1 | Boy, Johnny | male | 21 | 1 |
| 420 | 1311 | false | 1 | 0 | 2 | Bragin, Lewis | male | 22 | 0 |

Figura 14: Previsão para os dois novos passageiros.

O algoritmo determinou para esses dois novos casos com 100% da certeza que os passageiros novos não iriam sobreviver. No caso, por exemplo, do passageiro 1304 existe bastante confiança (mas não toda) que o passageiro irá sobreviver.

Olhando para a árvore gerada, e para os seus nodos, é possível tirar as seguintes conclusões.

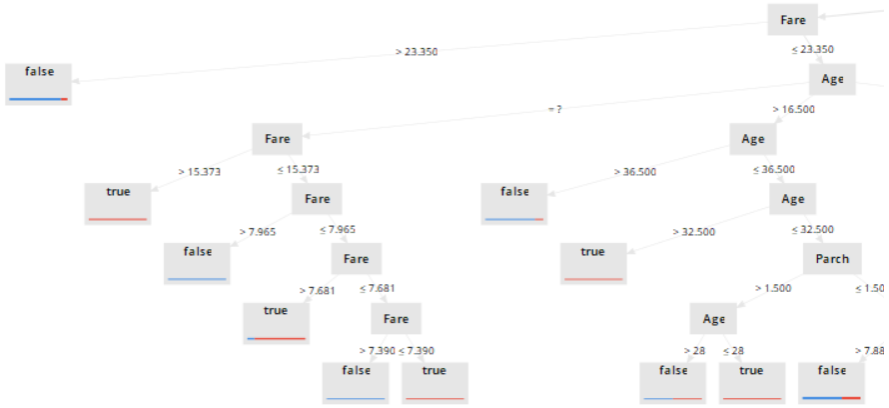


Figura 15: Árvore gerada com os parâmetros standard.

Ou seja, analisando o fluxo até chegar ao atributo *Fare*, é possível verificar que se a *Fare* do passageiro for menor 23350 então o passageiro não irá sobreviver.

### 2.5.2 Volte a executar o modelo, mas agora usando os valores dos parâmetros encontrados no exercício 3. Relate as diferenças na estrutura da sua árvore. Discuta se as suas hipóteses de sobrevivência e das pessoas que conhece aumentam.

Após alterar os parâmetros para os parâmetros do exercício três, foi possível verificar a existência de uma árvore completamente diferente, sendo esta mais compacta (e com menos nodos).

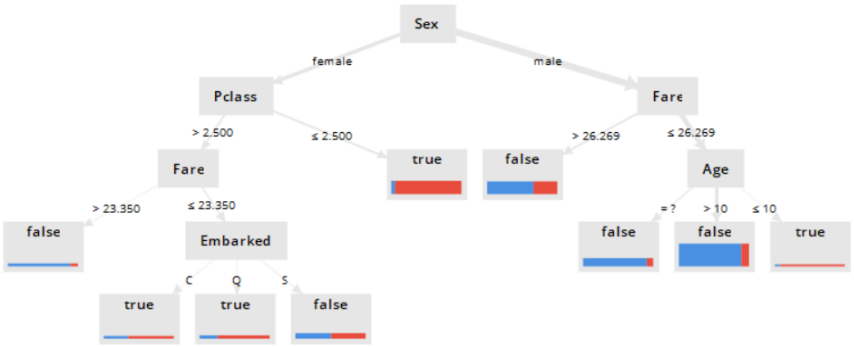


Figura 16: Árvore gerada com os parâmetros otimizados.

Como é possível verificar, após alterar os parâmetros para os otimizados, foi possível gerar uma árvore com um menor número de decisões. Voltando a verificar os dois novos passageiros adicionados, foi possível verificar que com estes parâmetros não existe tanta certeza acerca da não sobrevivência destes mesmos, mesmo assim o algoritmo determinou que os dois novos passageiros não iriam sobreviver. Portanto, aumentaram as chances de sobrevivência dos novos passageiros adicionados.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 419 | 1310 | false | 0.891 | 0.109 | 1 | Boy, Johnny | male | 21 | 1 |
| 420 | 1311 | false | 0.660 | 0.340 | 2 | Bragin, Lewis | male | 22 | 0 |

Figura 17: Previsão com parâmetros otimizados para dois novos passageiros.

### 2.5.3 Repita os exercícios 3 e 4(b) até que fique satisfeito com os resultados obtidos. Apresente detalhadamente todas as tentativas, bem como os resultados obtidos e respetivas comparações.

De modo a averiguar o funcionamento da hiper parametrização com um maior número de parâmetros, primeiramente foi utilizado o modelo apresentado anteriormente, onde os dados foram divididos num *split 60/40*.

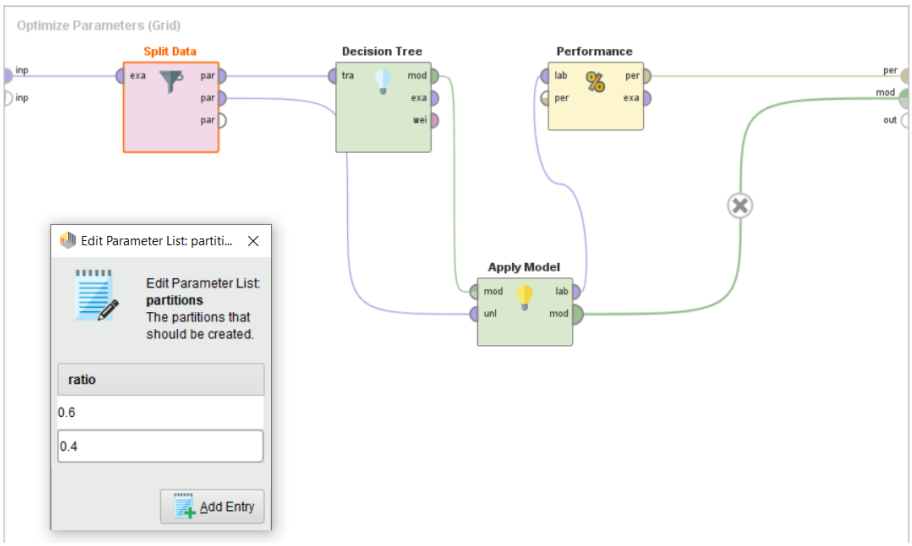


Figura 18: Split do dataset em treino e teste.

Foi também aumentado o número de parâmetros da hiper parametrezição, tendo sido escolhidos os seguintes parâmetros.

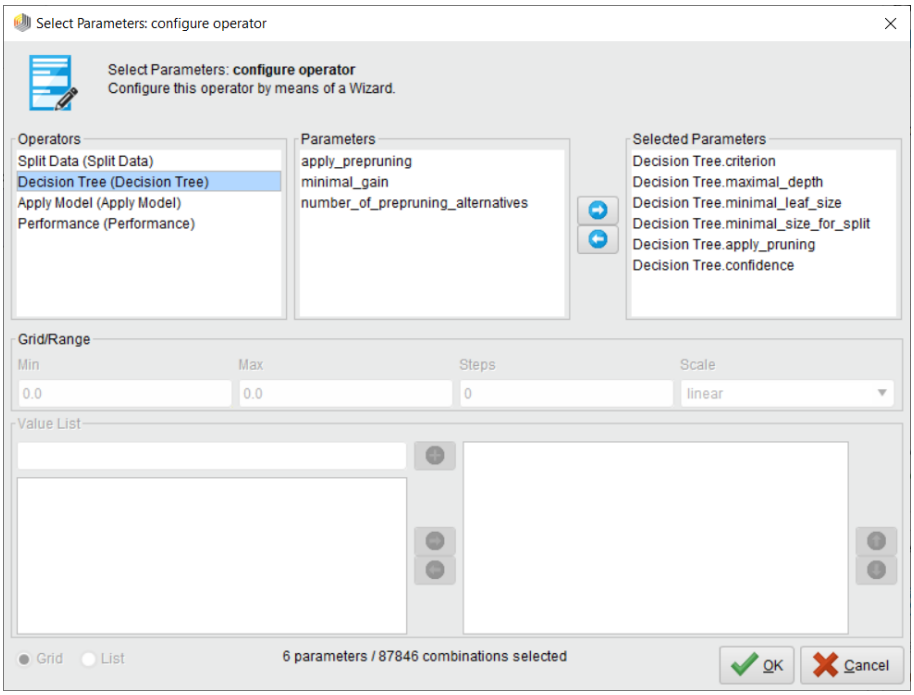


Figura 19: Novos parâmetros.

Após executar o modelo, foi possível obter os seguintes resultados ideais da hiper parametrização.

```
-----precision: 86.55% (positive class: true)
ConfusionMatrix:
True:   false   true
false:  206     31
true:   16      103
-----recall: 76.87% (positive class: true)
ConfusionMatrix:
True:   false   true
false:  206     31
true:   16      103
-----AUC (optimistic): 0.916 (positive class: true)
-----AUC: 0.879 (positive class: true)
-----AUC (pessimistic): 0.851 (positive class: true)
]
Decision Tree.criterion = gini_index
Decision Tree.maximal_depth     = 90
Decision Tree.minimal_leaf_size = 1
Decision Tree.minimal_size_for_split    = 11
Decision Tree.apply_pruning     = true
Decision Tree.confidence        = 0.15000006999999999
```

Figura 20: Resultados da nova execução.

Como tal, é agora necessário aplicar esses parâmetros ideais no operador de *Decision Tree* no modelo construído na pergunta 2.5. Pelo que a configuração deste mudou agora consideravelmente.

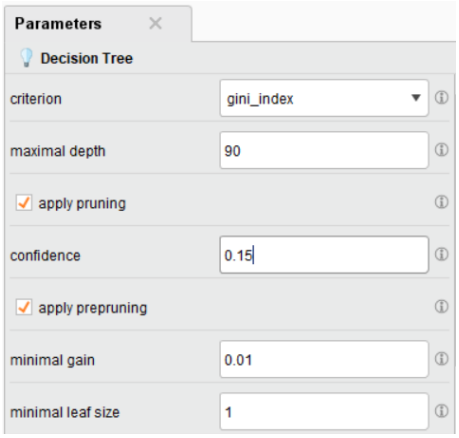


Figura 21: Novos parâmetros ideias do decision tree.

Após mudar os parâmetros e executar o modelo foi possível obter uma árvore de decisão bastante maior, tendo agora muitas mais decisões possíveis (um bocado à semelhança da execução com os parâmetros standard).

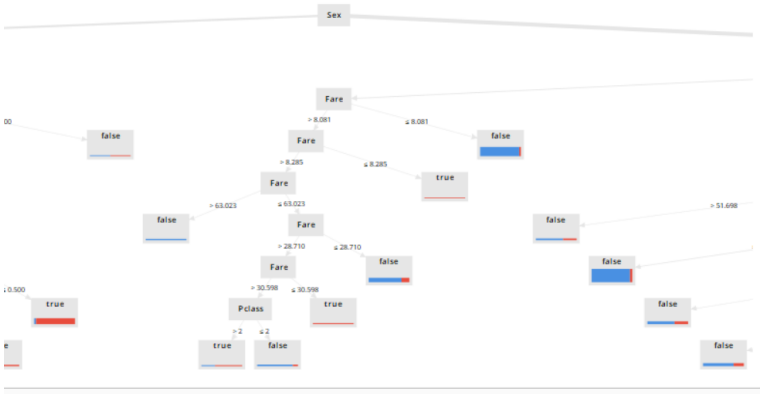


Figura 22: Nova árvore gerada após nova otimização dos parâmetros.

Ora, e verificando de novo os novos passageiros adicionado foi possível verificar que para o primeiro existe de novo 100% de certeza que não sobreviveu, para o segundo passageiro existe alguma dúvida ainda sobre se sobreviveu ou não, contudo o passageiro foi classificado como não tendo sobrevivido.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 419 | 1310 | false | 1 | 0 | 1 | Boy, Johnny | male | 21 | 1 |
| 420 | 1311 | false | 0.667 | 0.333 | 2 | Bragin, Lewis | male | 22 | 0 |

Figura 23: Previsão com os parâmetros novamente otimizados para os dois novos passageiros.

Testou-se dados com novas percentagens no *split data* em 80/20 e como se pode ver na figura exposta abaixo.

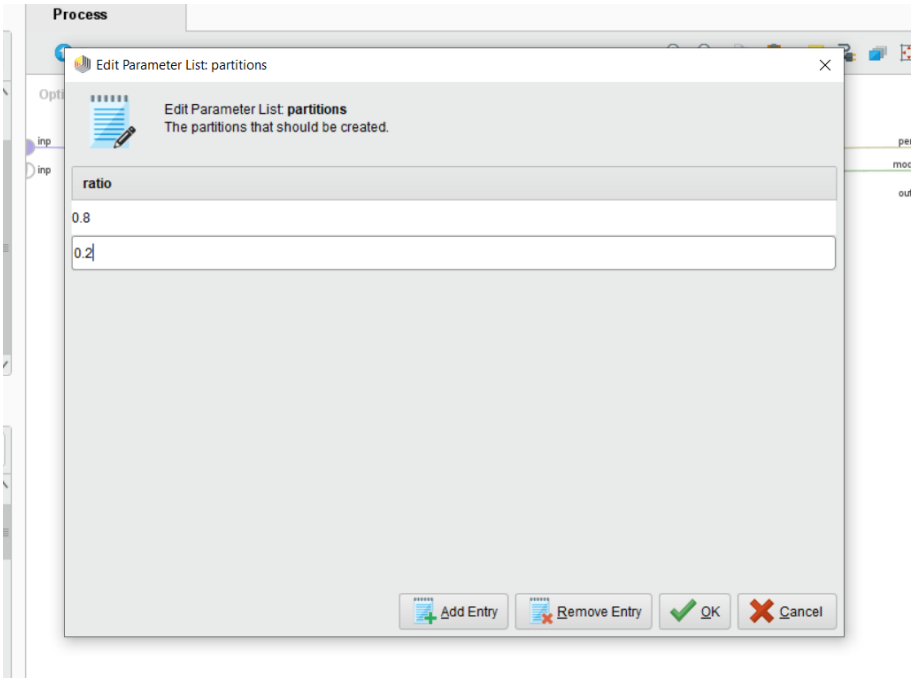


Figura 24: Percentagens dados para treino e teste.

Depois de correr o modelo com estes dados obtiveram-se os seguintes parâmetros optimizados.

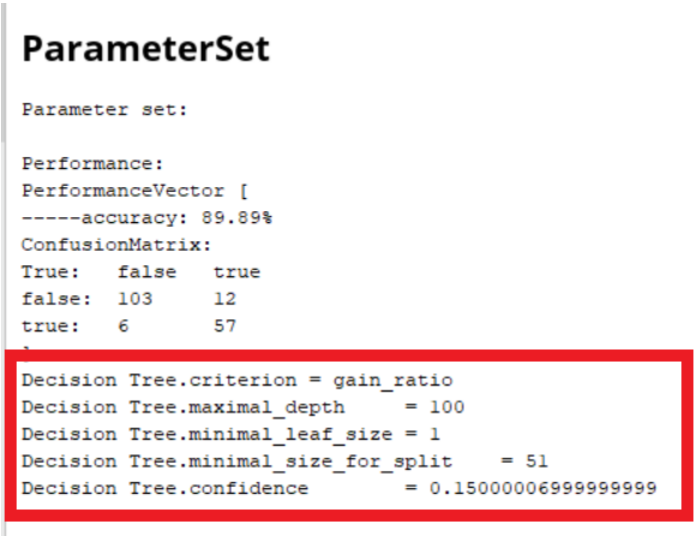


Figura 25: Parâmetros optimizados.

Substituíram-se então os parâmetros default pelos parâmetros optimizados e obtiveram-se os seguintes resultados para os dados adicionados ao dataset de scoring.

| | | | | | |
|---|---|---|---|---|---|
| 418 | 1309 | false | 0.893 | 0.107 | 3 |
| 419 | 1310 | false | 1 | 0 | 1 |
| 420 | 1311 | false | 0.667 | 0.333 | 2 |

Figura 26: Resultados para os dados inseridos no modelo após atualizar os parâmetros.

Após ter exausto todas as possibilidades de alterações de parâmetros, ou seja, não faria mais sentido alterar o data split na hyperparameterização uma vez que poderia causar dependência excessiva dos dados de treino. O que causaria overfitting dos dados. No que toca aos parâmetros a optimizar no Optimize Parameters também já foram escolhidos a maior parte dos parâmetros a utilizar na decision tree. Dá-se então por concluída a fase de experimentação.