Universidade do Minho
Mestrado em Matemática e Computação

Sistemas Baseados em Similaridade - TP2

Simão Pedro Batista Caridade Quintela - PG52257

Outubro
2023

Universidade do Minho
Mestrado em Matemática e Computação

Relatório

Relatório realizado no âmbito do TP2 da UC Sistemas Baseados em Similiaridade do Mestrado em Matemática e Computação.

Outubro
2023

Conteúdo

1 Contextualização

Para a realização do TP2 foi proposto analisar dados, criar e comparar modelos que consigam prever o grau de satisfação de um cliente no setor das telecomunicações. Alguns motivos para o descontentamento podem ser o preço ou um mau serviço prestado. Uma variável a ter em conta é a possibilidade de *churn* de um cliente, ou seja, se *churn*=0 significa que o cliente permaneceu na operadora, por outro lado, se *churn*=1 significa que o cliente abandonou a operadora.

Para a realização deste estudo foram fornecidos dois datasets distintos, um com dados acerca de chamadas dos clientes e outro com dados contratuais.

Nas seguintes páginas estarão presentes as minhas resoluções às tarefas propostas.

2 Tarefas

2.1 Tarefa 1

Enunciado: Carregar, no Knime, ambos os datasets. Utilizar um nodo Joiner para agregar, por "area code" e "phone", os dados provenientes das duas readers. Transformar o atributo Churn em nominal.

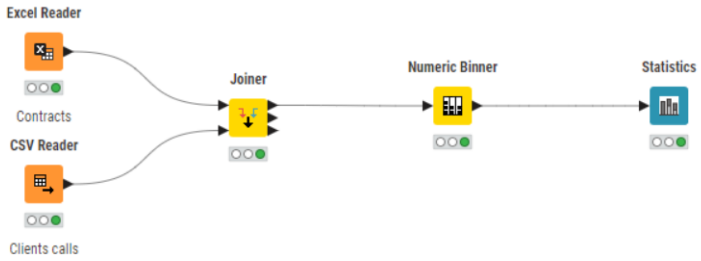

Figura 1: Circuito da tarefa 1

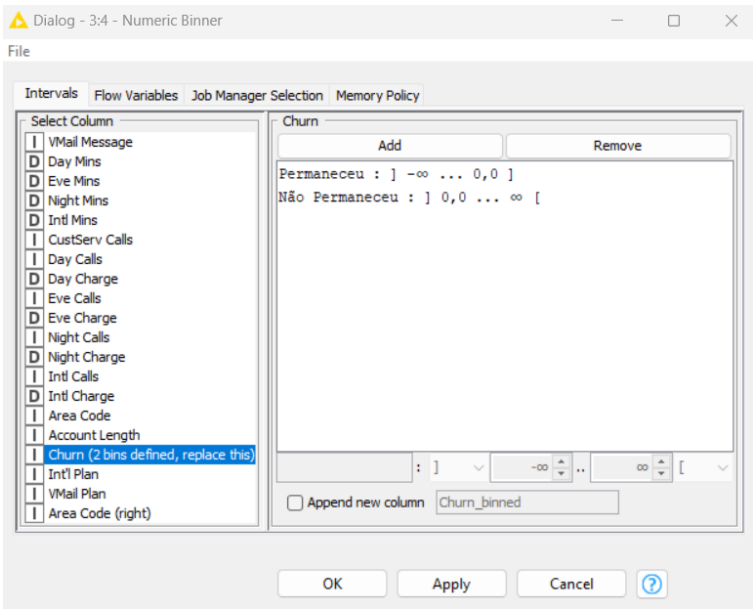

Figura 2: Configuração do Numeric Binner

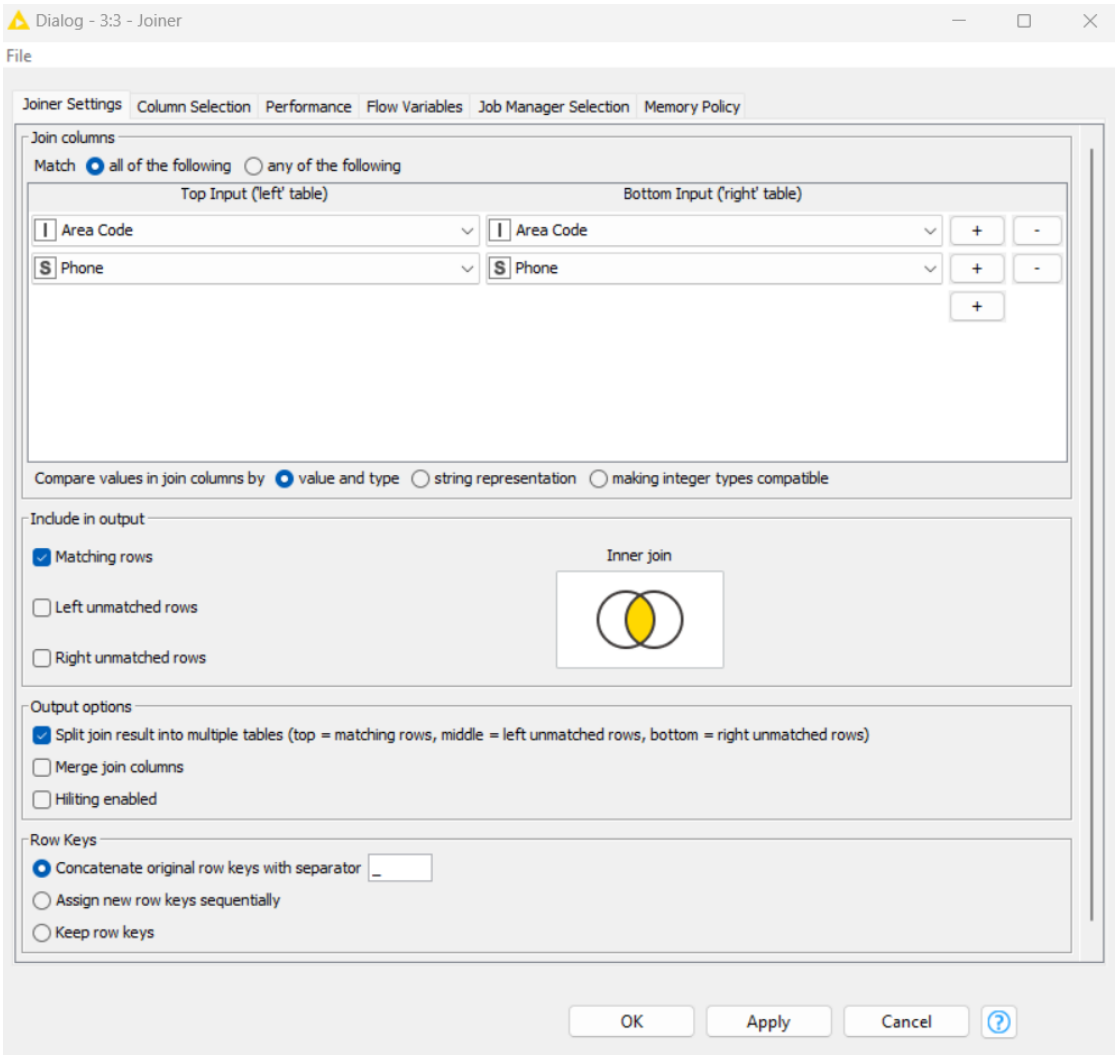

Figura 3: Configuração do Joiner

Na Figura 1 vemos a configuração do Joiner, onde realizamos a agregação com base nos campos *Area Code* e *Phone*.

Na Figura 2 temos a configuração do *Numeric Binner*, utilizado para transformar o atributo *Churn* em nominal.

Por fim, na Figura 3 temos o circuito completo, com a leitura dos respetivos dados.

2.2 Tarefa 2

Enunciado: Aplicar nodos para exploração de dados, i.e., analisar os dados em relação às suas características e padrões, procurando extrair informação relevante dos dados.

Statistics

Rows: 3 | Columns: 5

Name	Type	Mean Absolute Deviation	Standard Deviation	10 most common values
Area Code	Number (integer)	36.704	42.371	415 (1655; 49.65%), 510 (840; 25.2%), 408 (838; 25.14%)
Account Length	Number (integer)	31.821	39.822	105 (43; 1.29%), 87 (42; 1.26%), 93 (40; 1.2%), 101 (40; 1.2%), 90
Churn	String			Permaneceu (2850; 85.51%), Não Permaneceu (483; 14.49%)

Figura 4: Estatísticas de Churn

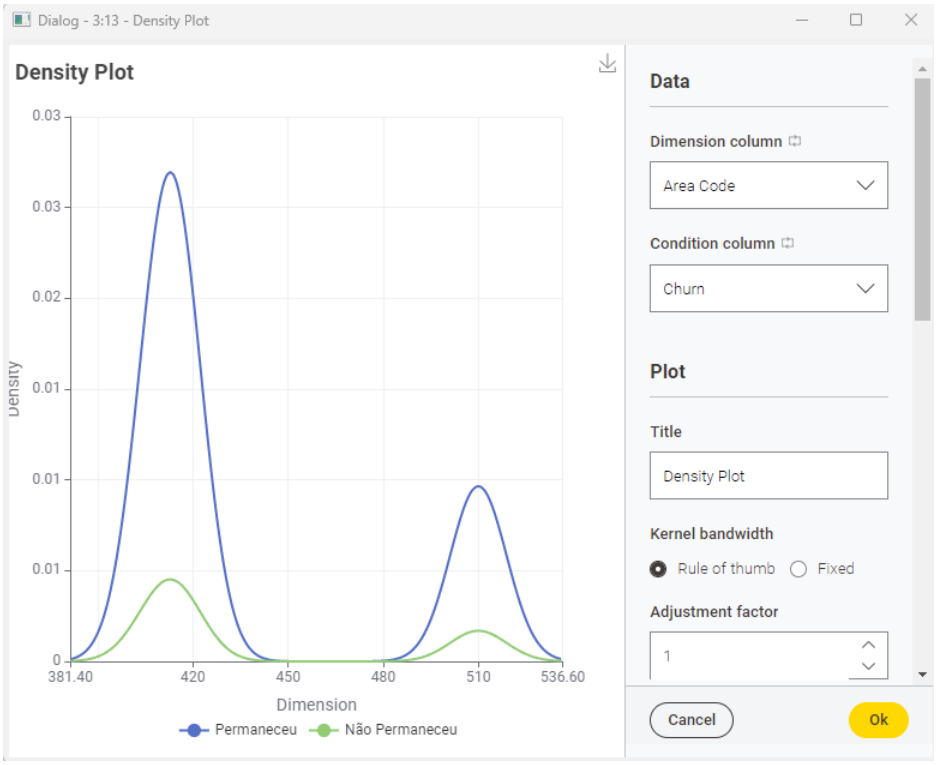

Figura 5: Densidade de Churn

Nesta tarefa analisei alguns dados resultantes de aplicar os nodos anteriormente demonstrados. Com efeito, na Figura 4 vemos algumas estatísticas de Churn, após remoção de algumas colunas. Repare-se que 85.51% das pessoas permaneceram na companhia.

Na figura 5 apliquei um nodo de densidade e relacionei a Area Code na **Dimension Column** com o Churn na **Condition Column** e observa-se que há maior frequência de clientes a viver nas zonas 380 - 450, comparativamente às zonas 480 - 540 mas a frequência de Churn não varia muito.

2.3 Tarefa 3

Enunciado: Particionar os dados de forma estratificada (pela feature "Churn"), utilizando 70% para aprendizagem e 30% para teste. Aplicar um Decision Tree Learner e um Decision Tree Predictor. Avaliar a precisão (accuracy) do modelo e a respetiva matriz de confusão;

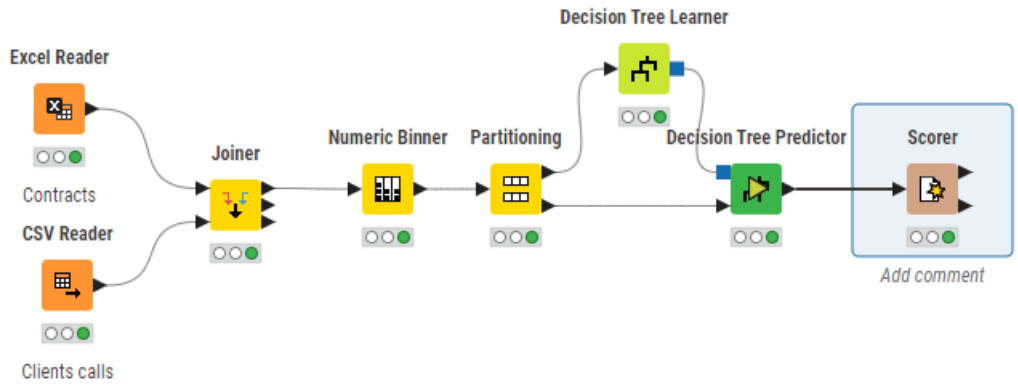

Figura 6: Circuito Completo

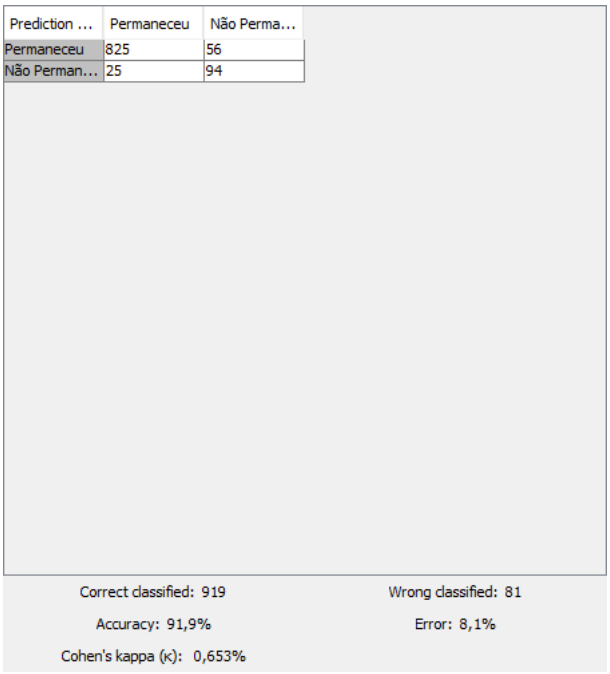

Figura 7: Matriz de Confusão

Nesta tarefa, inicialmente utilizei o nodo **Partitioning** para realizar a partição pedida, aplicando de seguida a Decision Tree. Como resultado obtemos 919(825+94) casos positivos e 81(25+56) casos negativos, resultando numa accuracy de 91.9%.

Consideremos o coeficiente **Cohen's kappa** com k=0.653%, o que significa que em 65.3% dos casos dois observadores distintos concordariam com as decisões tomadas, tendo em conta casos reais e esperados.

2.4 Tarefa 4

Enunciado: Remover, iterativamente, features do dataset e reavaliar a performance dos modelos candidatos. Descrever os resultados obtidos;

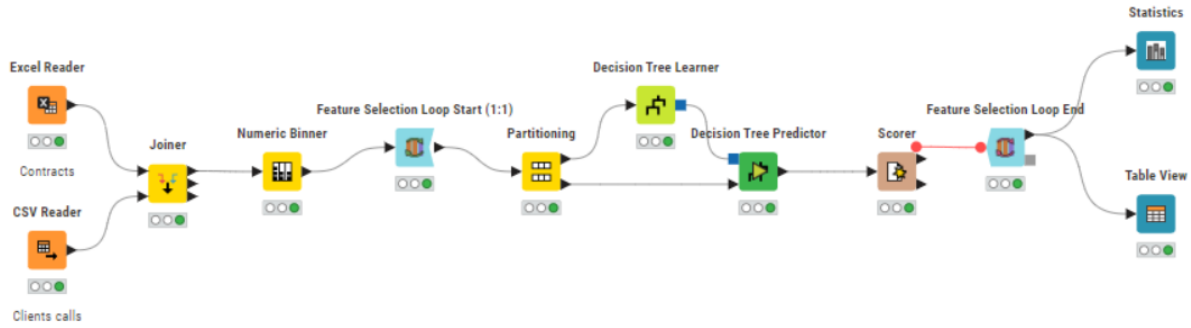

Figura 8: Circuito Completo

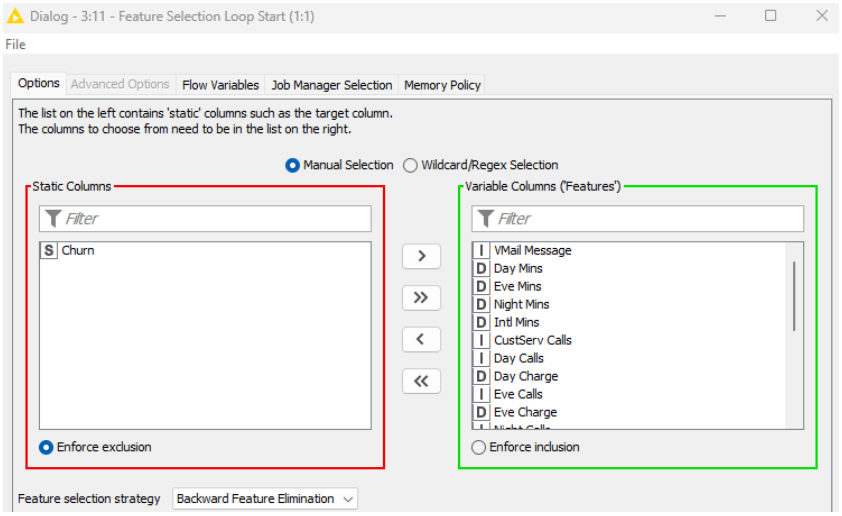

Figura 9: Configuração do Loop

Interactive View: Table View

Table View

Rows: 22 | Columns: 3

Row...	Nr. of features *Number (integer)*	Accuracy *Number (double)*	Removed feature *String*
All	22	0.905	
21	21	0.932	Night Calls
20	20	0.918	Night Charge
19	19	0.922	Night Mins
18	18	0.918	State
17	17	0.952	Phone
16	16	0.934	Day Charge
15	15	0.933	VMail Plan
14	14	0.942	Eve Calls
13	13	0.939	Day Calls
12	12	0.937	Account Length
11	11	0.943	Eve Charge
10	10	0.938	Area Code (right)

Figura 10: Table View

Inicialmente, na Figura 8 vemos o circuito usado para a resolução do problema, dando especial destaque ao nodo Feature Selection Loop Start, que foi usado para iterativamente remover features do dataset.

Por sua vez, na Figura 9 é mostrada a configuração do loop, em que o atributo *Churn* foi colocado na coluna static devido ao facto de ser um atributo nominal e em que foi usada a estratégia **Backward Feature Elimination** para, iterativamente, remover features do dataset.

Por fim, na Figura 10 vemos a accuracy do modelo com diferentes números de features. Se observarmos a tabela percebemos que até certo ponto a accuracy do nosso modelo aumenta à medida que retiramos features. Isto não é necessariamente boa notícia visto que uma maior quantidade de dados geralmente contribui para um modelo mais sólido.

Concluo, portanto, que nem sempre a accuracy do modelo é equivalente à qualidade do mesmo.

2.5 Tarefa 5

Enunciado: Seguir as práticas de bons-hábitos na construção de workflows.

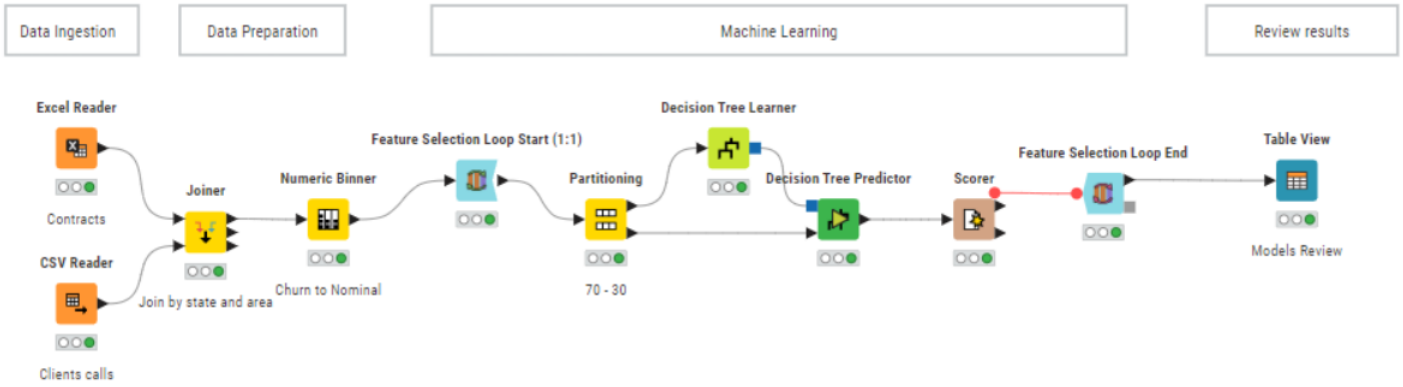

Figura 11: Circuito anotado

Na tarefa 5 procurei seguir boas práticas na construção de workflows através da utilização de notas, que marcam as diferentes fases do meu circuito. Para além disso também nomeei alguns nodos dando a entender de forma simples e sucinta as suas funcionalidades.

Podia ainda ter utilizado metanodos para compactar o workflow, no entanto, não achei necessário neste caso, visto que o circuito não demonstra grande complexidade.

2.6 Tarefa 6

Enunciado: Utilizar o output de um nodo Decision Tree Learner para criar uma imagem de uma Árvore de Decisão e guardar essa imagem no ambiente de trabalho.

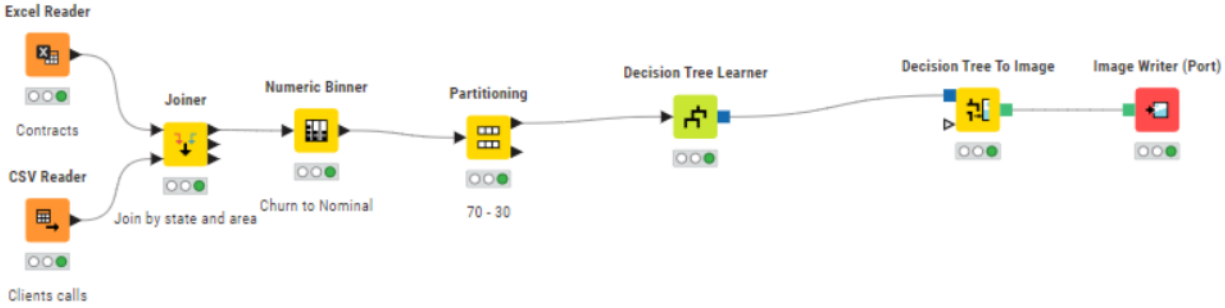

Figura 12: Workflow usado para representação em imagem

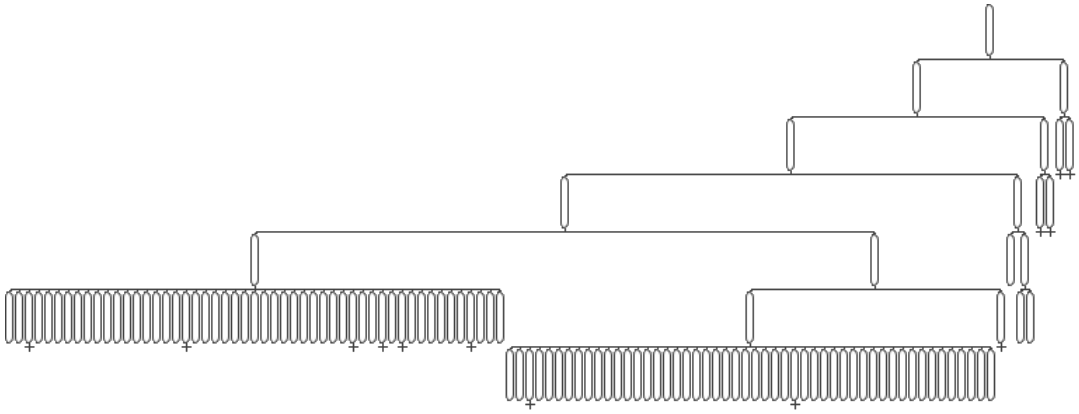

Figura 13: Decision Tree em formato PNG

Para finalizar, utilizei o nodo **Decision Tree To Image** para converter o modelo para imagem, e de seguida utilizei o nodo **Image Writer** para guardar a imagem resultante. O resultado final está expresso na Figura 13.

3 Conclusão

Para concluir, esta tarefa foi importante, principalmente, para o conhecimento de novos nodos do knime e para desenvolver a minha capacidade na análises de dados.