

Universidade do Minho

Departamento de Informática

Mestrado em Engenharia Informática

Perfil de Machine Learning Fundamentos e Aplicações

Sistemas Baseados em Similaridade

1º Ano, 1º Semestre

Ano letivo 2020/2021

Enunciado Prático nº 5

12 de novembro de 2020

Joel Costa Carvalho

PG42837

Índice

[Tarefas - 4 -](#_Toc56263011)

[T1 – Carregar, no *Knime*, os dois primeiros *datasets*, juntá-los e explorar os dados utilizando vistas gráficas que permitam perceber a análise efetuada; - 4 -](#_Toc56263012)

[T2 – Tratar os dados: - 6 -](#_Toc56263013)

[**a)** **Fazer *label* *encoding* à *feature* *isHoliday* (1 deve corresponder ao valor *True*);** - 6 -](#_Toc56263014)

[**b)** **Adicionar, a cada registo, as *features* ano e mês;** - 7 -](#_Toc56263015)

[**c)** **Agrupar os registos por loja, tipo, tamanho, ano e mês, agregando de forma a obter o somatório das vendas semanais de cada loja e a indicação da existência de feriados nesse mês;** - 7 -](#_Toc56263016)

[**d)** **Normalizar o somatório das vendas semanais usando a transformação linear *Min*-*Max* entre 0-1;** - 7 -](#_Toc56263017)

[**e)** **Criar 4 *bins* de igual frequência sobre o valor normalizado no passo anterior (ligando a opção *replace target column(s));*** - 8 -](#_Toc56263018)

[***f)*** **Renomear cada bin de forma a que o primeiro corresponda a *Low*, o segundo a *Medium*, o terceiro a *High* e o quarto a *Very High.*** - 8 -](#_Toc56263019)

[T3 – Treinar: - 8 -](#_Toc56263020)

[**a)** **Uma árvore de decisão;** - 8 -](#_Toc56263021)

[**b)** **Carregar o *dataset* teste e prever o valor de vendas de cada mês para cada uma das 17 lojas;** - 8 -](#_Toc56263022)

[**c)** **Mostrar, graficamente, uma tabela com a matriz de confusão do modelo.** - 9 -](#_Toc56263023)

[T4 - Fazer o tuning do modelo criado no passo anterior, experimentando: - 9 -](#_Toc56263024)

[**a)** **Todos os valores, entre 2 e 10, para o número mínimo de registos por nodo;** - 9 -](#_Toc56263025)

[**b)** **Todas as possibilidades para a medida de qualidade;** - 10 -](#_Toc56263026)

[**c)** **Todas as possibilidades para o método de *pruning*;** - 10 -](#_Toc56263027)

[**d)** **Fazer o *tuning* dos parâmetros anteriores num único *workflow*. Guardar e analisar todos os resultados obtidos para cada combinação de hiper-parâmetros.** - 10 -](#_Toc56263028)

[T5 - Treinar e fazer o *tuning* de uma *Random Forest*. Guardar e analisar todos os resultados obtidos para cada combinação de hiper-parâmetros; - 11 -](#_Toc56263029)

[T6 - Analisar e comparar as performances dos modelos treinados em T4 e T5. Que conclusões se podem tirar? - 11 -](#_Toc56263030)

Índice de figuras

[Figura 1 - Carregar dataset e explorar dados - 4 -](#_Toc56331642)

[Figura 2 - Data Explorer - 4 -](#_Toc56331643)

[Figura 3 - Correlação entre as features - 5 -](#_Toc56331644)

[Figura 4 - Análise das vendas por loja - 5 -](#_Toc56331645)

[Figura 5 - Metanode T2 - 6 -](#_Toc56331646)

[Figura 6 – Transformar feature isHoliday em boleano - 6 -](#_Toc56331647)

[Figura 7 – Extrair ano e mês - 7 -](#_Toc56331648)

[Figura 8 – Agrupar por loja, tipo, tamanho, ano e mês - 7 -](#_Toc56331649)

[Figura 9 – Normalizar somatório das vendas semanais - 7 -](#_Toc56331650)

[Figura 10 - 4 bins de frequência para o valor normalizado - 8 -](#_Toc56331651)

[Figura 11 - Renomear bins - 8 -](#_Toc56331652)

[Figura 12 - Carregar dataset de teste e carregar modelo - 8 -](#_Toc56331653)

[Figura 13 - Matriz confusão - 9 -](#_Toc56331654)

[Figura 14 – Tuning do modelo criado em T3 - 9 -](#_Toc56331655)

[Figura 15 - Tuning número de registos - 9 -](#_Toc56331656)

[Figura 16 - Todas as possibilidades para a medida de qualidade e pruning - 10 -](file:////Users/joelcarvalho/Desktop/Mestrado/Cadeiras/SBS/TP5/TP5_JoelCarvalho_PG42837.docx#_Toc56331657)

[Figura 17 - Tuning dos parâmetros anteriores - 10 -](#_Toc56331658)

[Figura 18 - Tuning Random Forest - 11 -](#_Toc56331659)

**Tarefas**

**T1 – Carregar, no *Knime*, os dois primeiros *datasets*, juntá-los e explorar os dados utilizando vistas gráficas que permitam perceber a análise efetuada;**

Após o carregamento dos *datasets* e a união dos mesmos, os dados foram explorados com o auxílio do nodo *Data Explorer*, permitindo efetuar médias, medianas, modas, variâncias, desvios padrões e entre outros. No final, a Figura 4., apresenta graficamente as vendas médias semanais por loja.

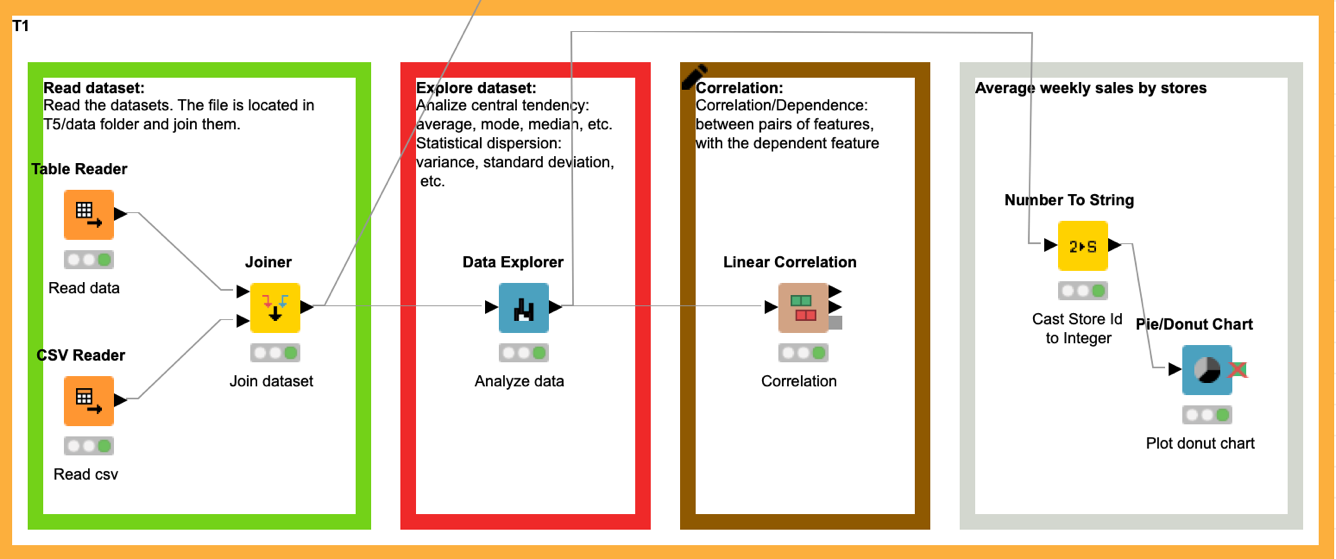


Figura 1 - Carregar dataset e explorar dados

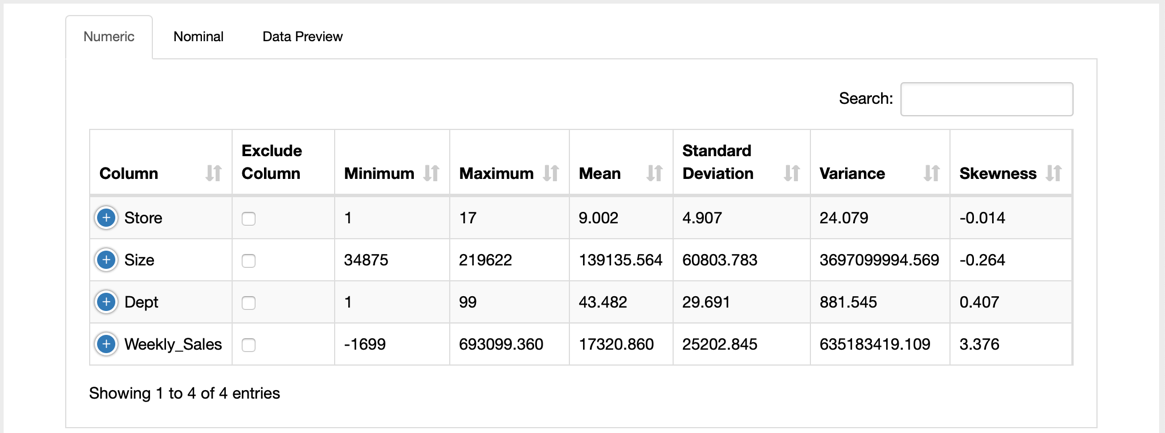


Figura 2 - Data Explorer

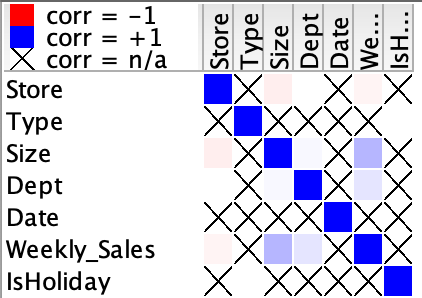


Figura 3 - Correlação entre as features

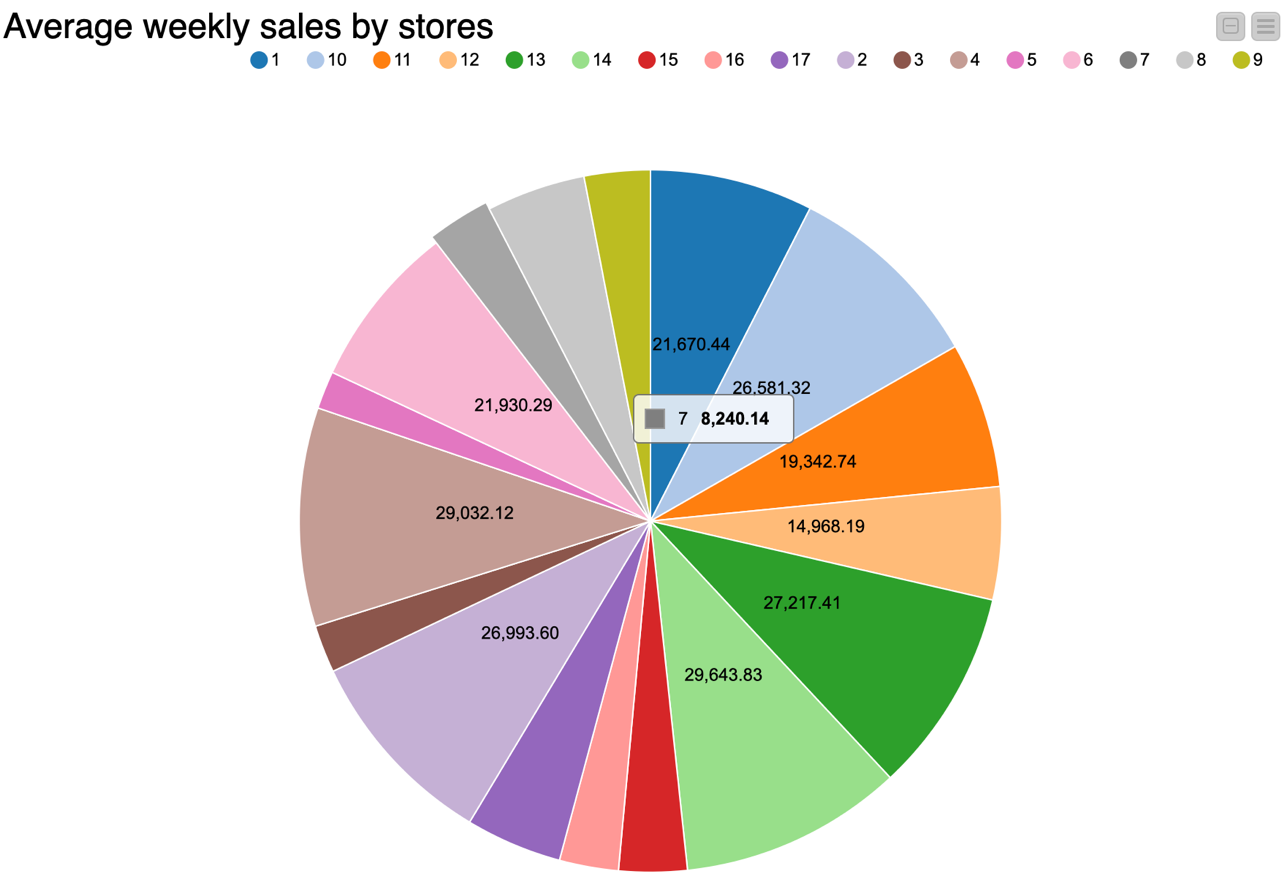


Figura 4 - Análise das vendas por loja

**T2 – Tratar os dados:**

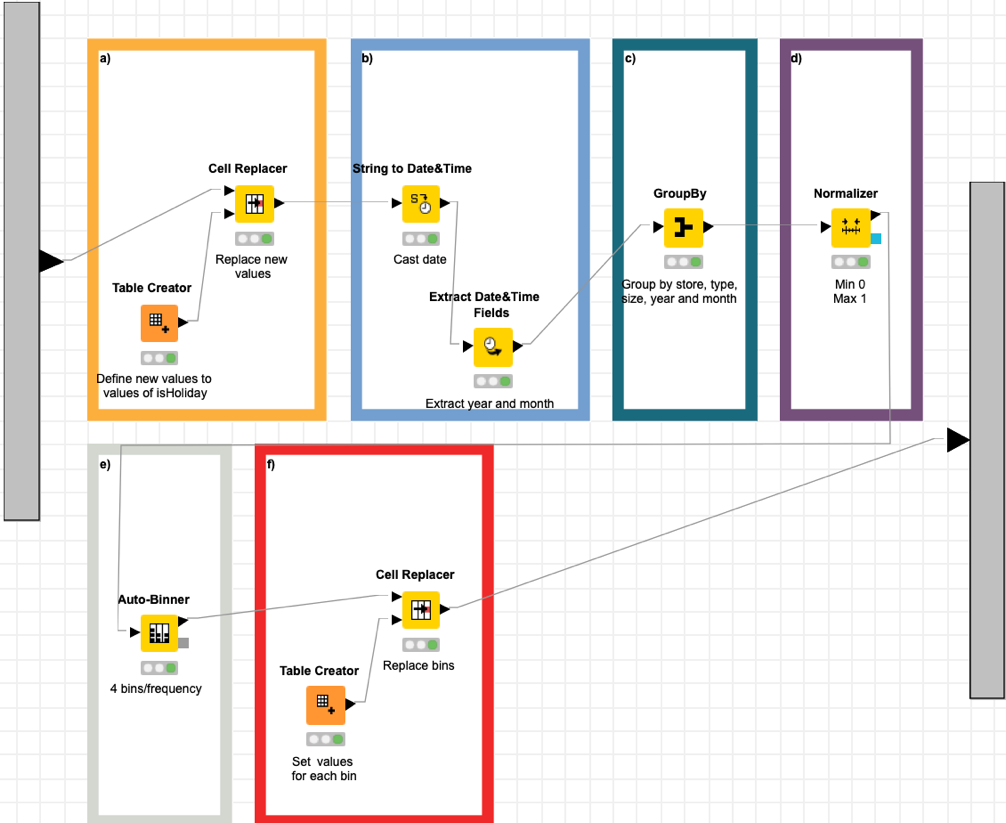
****

Figura 5 - Metanode T2

1. **Fazer *label* *encoding* à *feature* *isHoliday* (1 deve corresponder ao valor *True*);**

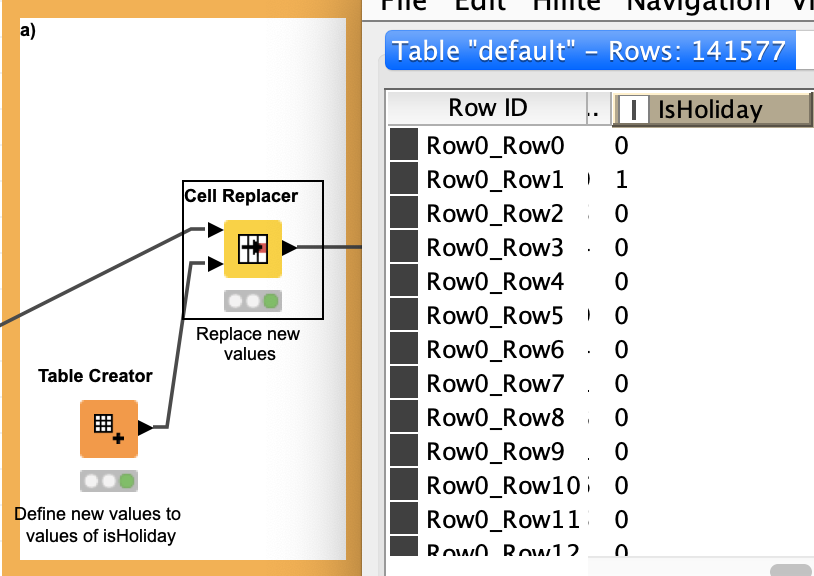
****

Figura 6 – Transformar feature isHoliday em boleano

1. **Adicionar, a cada registo, as *features* ano e mês;**

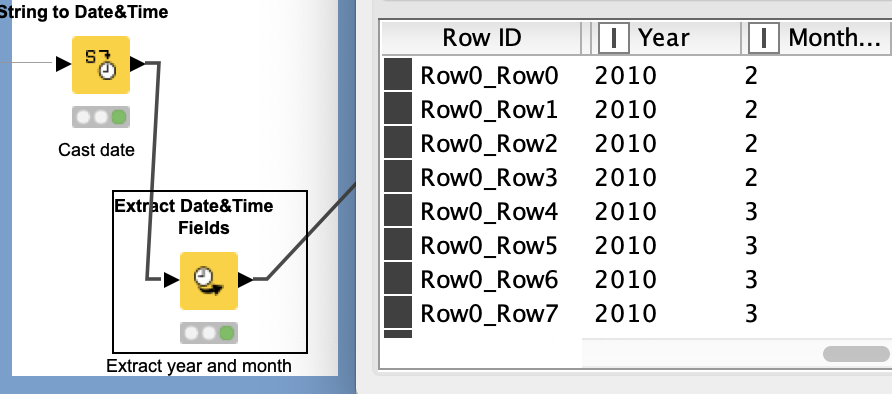
****

Figura 7 – Extrair ano e mês

1. **Agrupar os registos por loja, tipo, tamanho, ano e mês, agregando de forma a obter o somatório das vendas semanais de cada loja e a indicação da existência de feriados nesse mês;**

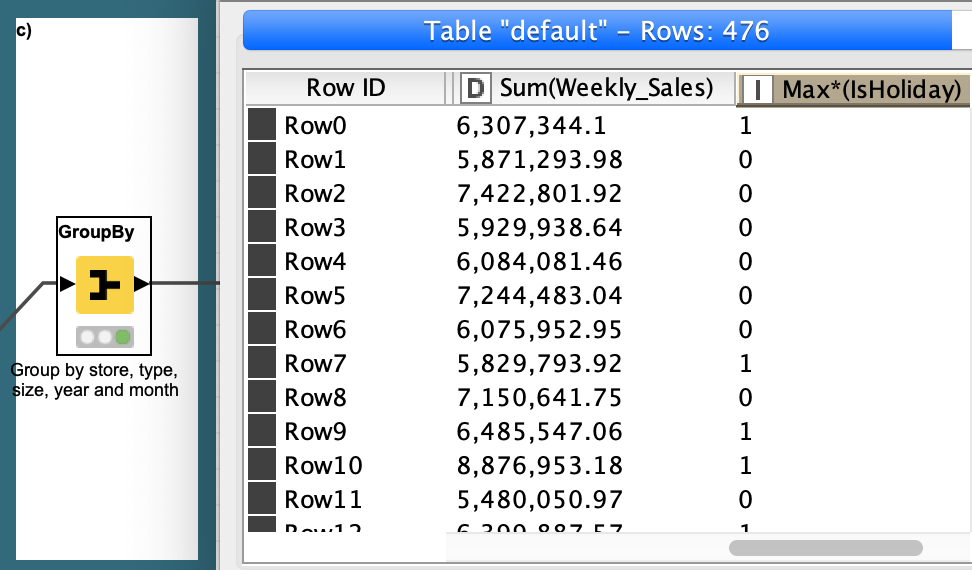
****

Figura 8 – Agrupar por loja, tipo, tamanho, ano e mês

1. **Normalizar o somatório das vendas semanais usando a transformação linear *Min*-*Max* entre 0-1;**

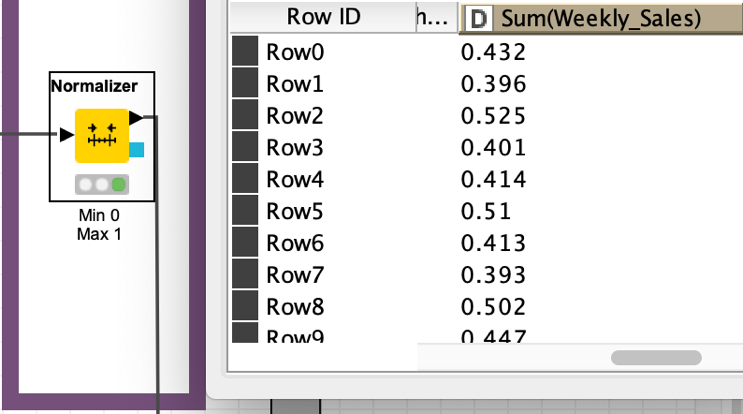


Figura 9 – Normalizar somatório das vendas semanais

1. **Criar 4 *bins* de igual frequência sobre o valor normalizado no passo anterior (ligando a opção *replace target column(s));***

**Uma imagem com mesa

Descrição gerada automaticamente**

Figura 10 - 4 bins de frequência para o valor normalizado

1. **Renomear cada bin de forma a que o primeiro corresponda a *Low*, o segundo a *Medium*, o terceiro a *High* e o quarto a *Very High.***

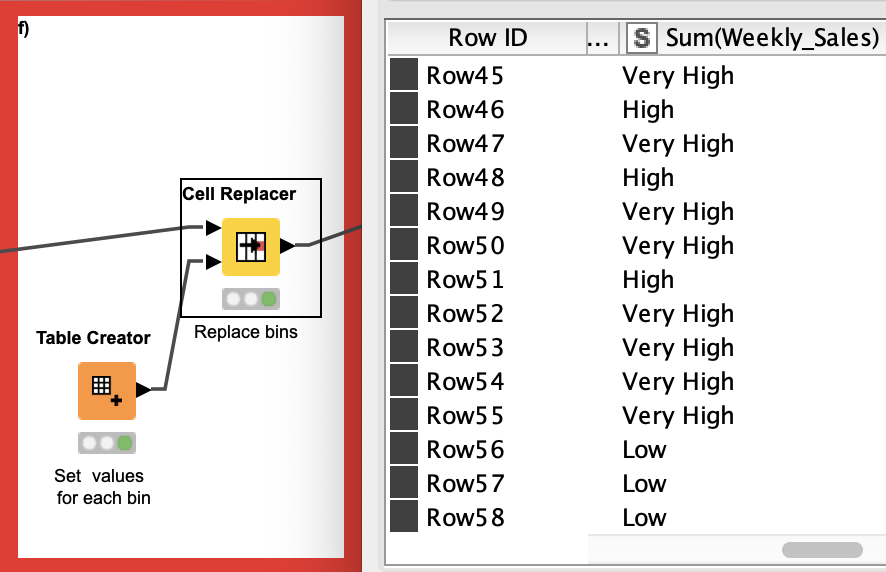
****

Figura 11 - Renomear bins

**T3 – Treinar:**

1. **Uma árvore de decisão;**
2. **Carregar o *dataset* teste e prever o valor de vendas de cada mês para cada uma das 17 lojas;**

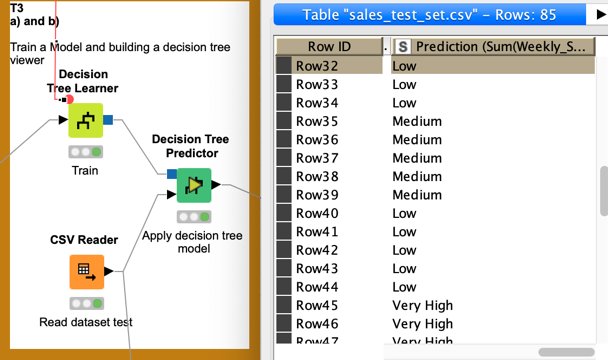
****

Figura 12 - Carregar dataset de teste e carregar modelo

1. **Mostrar, graficamente, uma tabela com a matriz de confusão do modelo.**

**Uma imagem com mesa

Descrição gerada automaticamente**

Figura 13 - Matriz confusão

**T4 - Fazer o tuning do modelo criado no passo anterior, experimentando:**

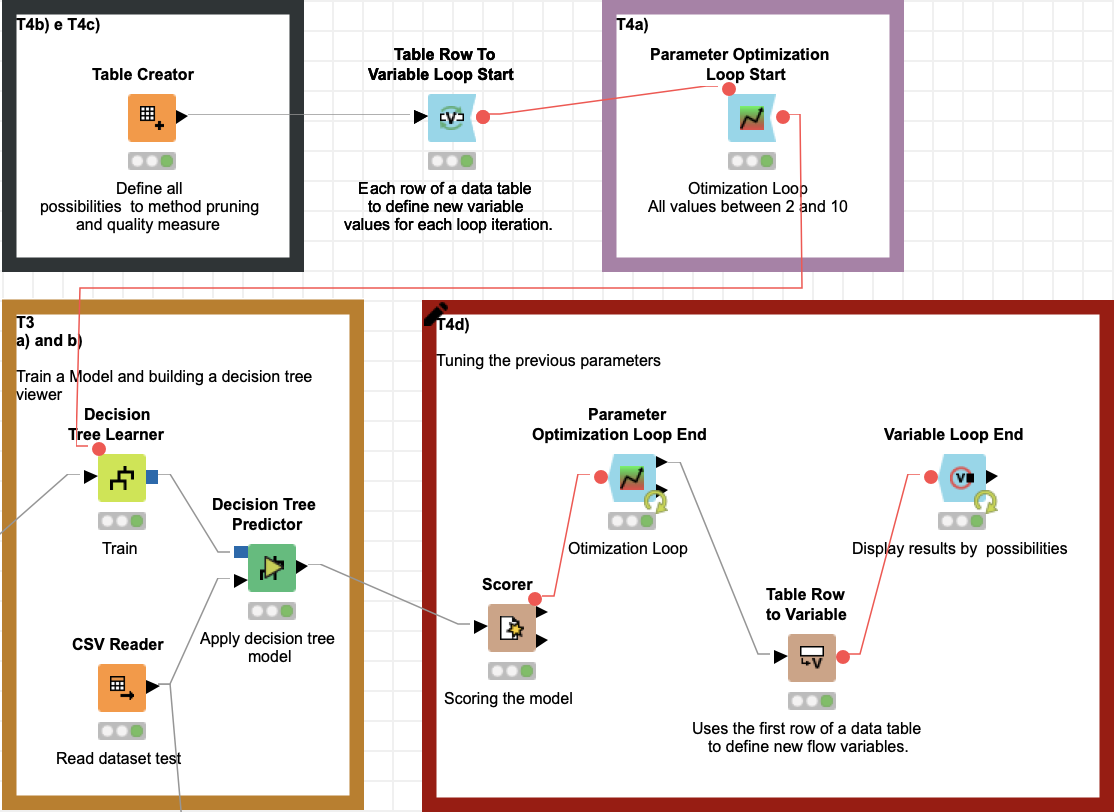


Figura 14 – Tuning do modelo criado em T3

1. **Todos os valores, entre 2 e 10, para o número mínimo de registos por nodo;**

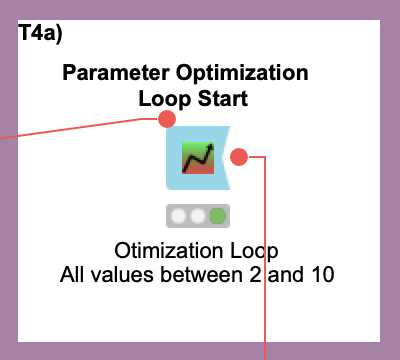


Figura 15 - Tuning número de registos

1. **Todas as possibilidades para a medida de qualidade;**
2. **Todas as possibilidades para o método de *pruning*;**

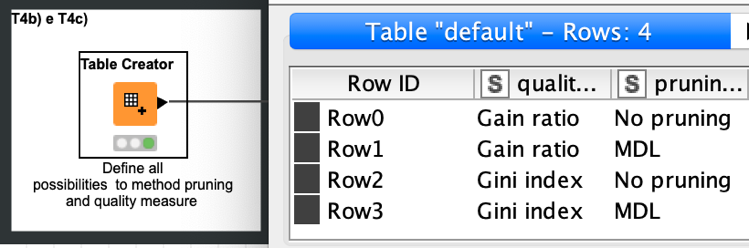
****

Figura 16 - Todas as possibilidades para a medida de qualidade e pruning

1. **Fazer o *tuning* dos parâmetros anteriores num único *workflow*. Guardar e analisar todos os resultados obtidos para cada combinação de hiper-parâmetros.**

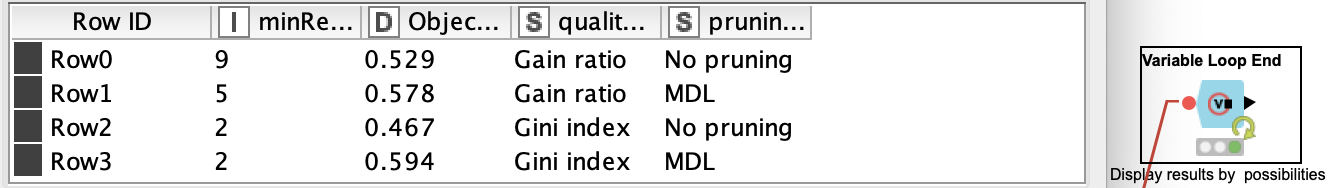
****

Figura 17 - Tuning dos parâmetros anteriores

**Qual a combinação que oferece melhor performance?**

Analisando os resultados obtidos podemos concluir que a melhor combinação seria colocar 2 nodos mínimos por registos, “*Gini* *index*” na *quality meausure* e o método *pruning* “*MDL*”, pois oferece uma *accuracy* de 0.594.

**Existem grandes discrepâncias?**

Não, a *accurary* é relativamente baixa em todas as combinações possíveis. O mínimo registo é de 0.467.

**T5 - Treinar e fazer o *tuning* de uma *Random Forest*. Guardar e analisar todos os resultados obtidos para cada combinação de hiper-parâmetros;**

Uma imagem com mesa

Descrição gerada automaticamente

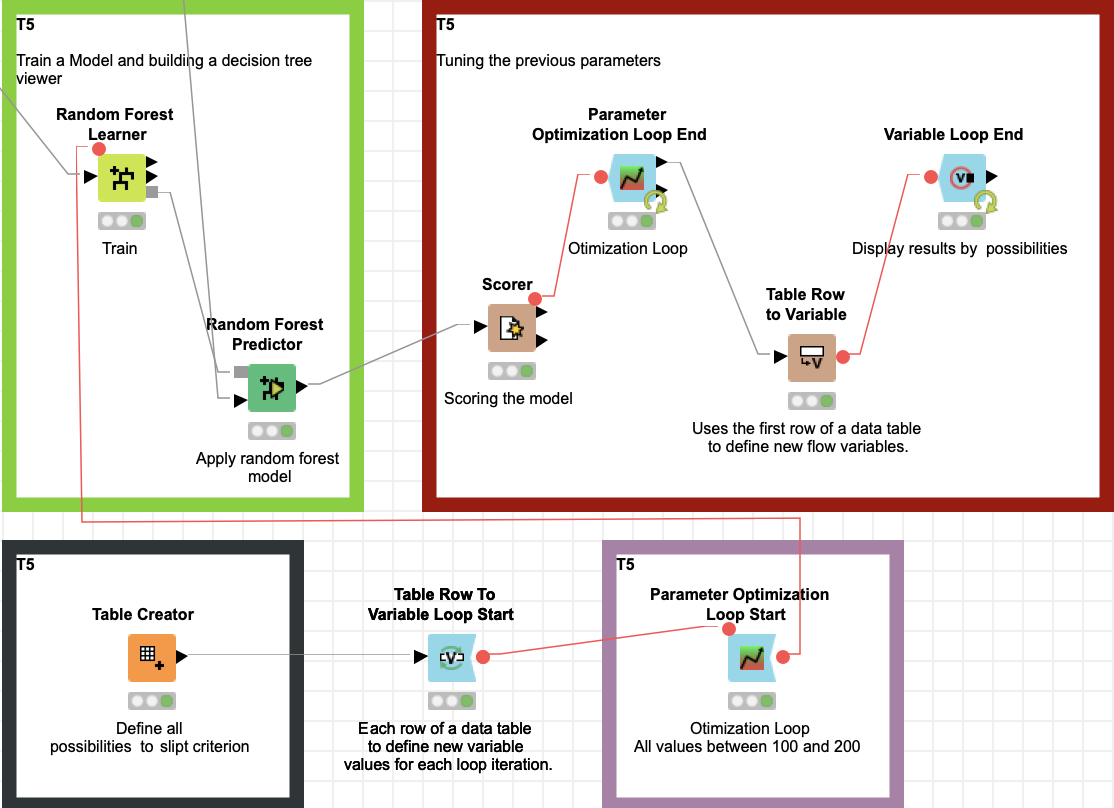


Figura 18 - Tuning Random Forest

**T6 - Analisar e comparar as performances dos modelos treinados em T4 e T5. Que conclusões se podem tirar?**

Analisando os resultados obtidos podemos concluir que com a *Decision Tree Learner* conseguimos uma ligeira superioridade na *accuracy* que o *Random Forest Learner*, contudo este último oferece uma menor discrepância que o primeiro modelo treinado.

Resumindo, o modelo treinado na alínea T4 é melhor que o da alínea T5.

|  |  |  |  |
| --- | --- | --- | --- |
| Modelo | Melhor combinação | Precisão máxima | Precisão mínima |
| *Decision Tree Learner* | 1. Mínimo de 2 nodos por registo 2. Medida de qualidade: *Gini Index* 3. Método de suprimir: MDL | 0.594 | 0.467  Diferença: 0.127 |
| *Random Forest Learner* | 1. 150 Modelos 2. Critério de divisão: *Information Gain Ratio* | 0.593 | 0.577  Diferença: 0.016 |