Relatório Trabalho Prático - ADI

Grupo 20:
António Luís de Macedo Fernandes (a93312)
José Diogo Martins Vieira (a93251)
João Silva Torres (a93231)
Ricardo Lopes Santos Silva (a93195)

Maio 2022

Contents

1 Introdução

No âmbito do desenvolvimento do trabalho prático da unidade curricular de Aprendizagem e Decisão Inteligentes foi-nos proposto desenvolver um projeto utilizando os modelos de aprendizagem abordados ao longo do semestre.

Dito isto, trabalhamos com dois datasets. O primeiro foi disponibilizado pela equipa docente - USA Housing Regression. O objetivo é estimar o valor da habitação numa região dos Estados Unidos da América. Trata-se de um dataset de regressão. Relativamente ao segundo, foi-nos dada a liberdade de escolhermos um. Após diversas consultas e análises de vários no Google Dataset Search, no Kaggle e no UCI Machine Learning Repository decidimos trabalhar com o Stroke Prediction Dataset. Optamos por selecionar o mesmo, visto que se trata de um dataset de classificação.

Numa fase inicial deste relatório iremos abordar ambos os datasets e explicar em que consistem e qual o objetivo de estudo em cada um. De seguida, vamos expor qual a metodologia utilizada pelo grupo e, posteriormente, irá estar presente a descrição e exploração detalhada de ambos os datasets e do tratamento de dados efetuado. Vamos também esclarecer quais os modelos desenvolvidos e as suas características, bem como os parâmetros de treino. Por fim, vamos falar acerca dos resultados obtidos e faremos uma análise crítica dos mesmos e, depois, uma conclusão sobre todo o trabalho.

2 Casos de Estudo

A resolução deste trabalho prático consiste, maioritariamente, na análise e preparação de dois datasets. De seguida, iremos abordá-los e referir os seus features constituintes.

2.1 Dataset Stroke

De acordo com a Organização Mundial da Saúde (OMS) o AVC é a 2ª causa de morte no mundo, responsável por aproximadamente 11% do total de óbitos. Esse conjunto de dados é usado para prever se um paciente provavelmente sofrerá AVC com base nos parâmetros de entrada, como sexo, idade, várias doenças e tabagismo. Cada linha nos dados fornece informações relevantes sobre o paciente.

Assim sendo, escolhemos este dataset pois achamos interessante tentar prever se um paciente provavelmente sofrerá um AVC através de aspetos da sua vida. Para além disto, não só se trata de um dataset com dados simples e de fácil interpretação, como também é um dataset de classificação. Como nos foi atribuído pela equipa docente um de regressão, o grupo chegou à conclusão de que seria uma boa escolha optar por trabalhar, simultaneamente, com um de classificação.

O dataset é composto pelos seguintes parâmetros:

- id - identicador único
- gender - "Male", "Female" ou "Other"
- age - idade do paciente
- hypertension - 0 se o paciente não tiver hipertensão, 1 caso tenha
- heart_disease - 1 se o paciente sofrer de alguma doença de coração, 0 caso contrário
- ever_married - "No" ou "Yes"
- work_type - "children", "Got_jov", "Never_worked", "Private" ou "Self-employed"
- Residence_type - "Rural" or "Urban"
- avg_glucose_level - nível médio de glucose no sangue
- bmi - índice de massa corporal
- smoking_status - "formerly smoked", "never smoked", "smokes" ou "Unknown"*
- stroke - 1 se o paciente já teve um AVC, 0 caso contrário

2.2 USA housing regression dataset

Para o dataset fornecido pela equipa docente, como somos o grupo 20, iremos trabalhar com o dataset (USA_housing_regression), que tem como objetivo estimar o valor da habitação numa região dos Estados Unidos da América. Este dataset possui 6 atributos e um objetivo, sendo este o preço da habitação (target).

O dataset é composto pelos seguintes parâmetros:

- avg. Area Income - número médio de residentes.
- avg. Area House Age - número médio da idade das casas.
- avg. Number of Rooms - número médio das salas das casas.
- avg. area number of bedrooms - número médio dos quartos das casas.
- area population - número da população onde se localiza a casa.
- adress - morada da casa.

3 Metodologia

Um dos tópicos de estudo lecionado numa das primeiras aulas teóricas foi o das metodologias. Por definição, uma Metodologia para Análise de Dados descreve e cria um conjunto de passos pelos quais deverá passar o desenvolvimento de um Projeto de Aprendizagem Automática (Machine Learning) para a resolução de problemas.

Com efeito, escolhemos trabalhar com o CRISP-DM - CRoss Industry Standard Process for Data Mining.

O CRISP-DM é um modelo de processos com vista a definir um "guião" para o desenvolvimento de projetos de AD, que se desenrola em 6 etapas:

- Estudo do negócio;

  Numa fase inicial tentamos perceber quais eram os objetivos deste projeto. Para isso o grupo teve uma reunião onde "estudou" o enunciado que nos foi entregue e trocámos ideias sobre o que seria ou não vantajoso para o desenvolvimento do trabalho. Foi feita uma avaliação de todo o problema e, posteriormente, determinamos e definimos objetivos.

- Estudo dos dados;

  Assim, uma evidência notória que detetamos foi a de que estavamos perante um caso de aprendizagem com supervisão, pois os casos que estão a ser usados para aprender contêm informação acerca dos resultados pretendidos.

  De seguida, ao analisar o USA housing regression, que nos foi atribuído, concluímos que se tratava de um dataset de regressão, porque os resultados são contínuos (estimativa do valor da habitação numa região dos EUA).

Isto levou-nos a tomar a decisão de que o dataset a ser escolhido tinha de ser de classificação. Após algumas pesquisas escolhemos o Dataset Stroke. Este é usado para prever se um paciente provavelmente sofrerá AVC.

- Preparação dos dados;

  Após todo o processo de recolha, descrição, exploração e qualidade aprovado de dados, feito na fase anterior, passamos então para a preparação dos mesmos.

  Nesta etapa focamo-nos essencialmente na limpeza e formatação dos dados.

- Modelação;

  Na modelação começamos a trabalhar com as ferramentas de AD, com o objetivo de construir o modelo.

- Avaliação;

  Nesta penúltima etapa realizamos a comparação dos resultados obtidos com os objetivos do projeto e chegamos a algumas conclusões acerca do trabalho desenvolvido.

- Desenvolvimento.

  Por fim, colocamos o modelo em produção. Realizamos o relatório final e, por último, uma revisão geral a todo o projeto.

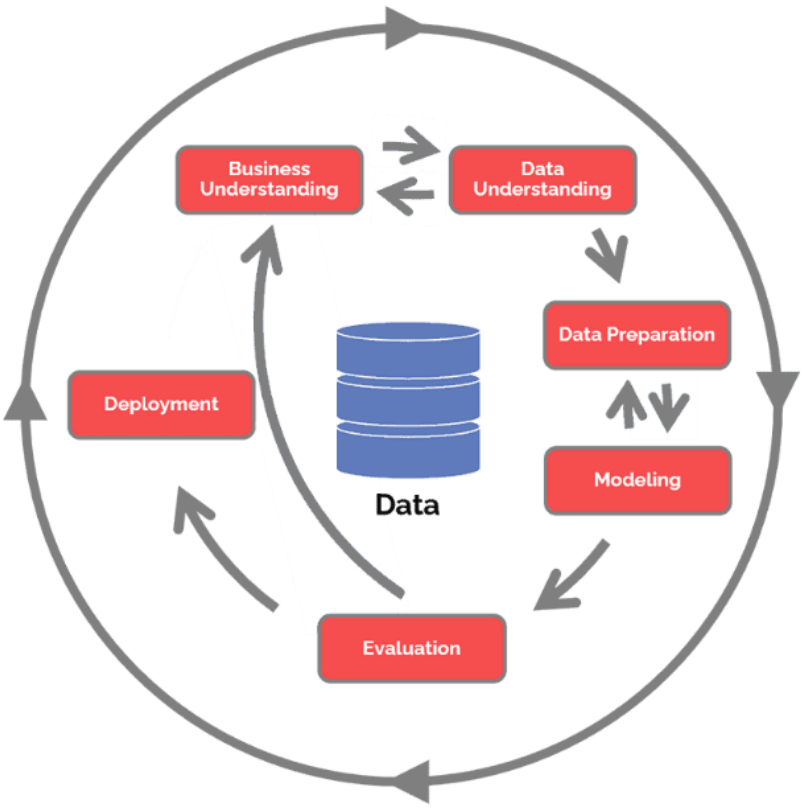


Figure 1: Diagrama CRISP-DM

4 Análise e Tratamento de Dados

4.1 USA housing regression dataset

Este dataset contém apenas 6 colunas e 10000 linhas, por isso, cada informação que se retirar pode ser prejudicial para a aprendizagem, visto que a quantidade de informação já é reduzida.

Inicialmente, começamos por aplicar um Column Filter à coluna "Address", por este ser diferente para cada linha e assim não serviria de grande ajuda para os nossos modelos. Porém, decidimos repensar esta decisão e optámos por, em vez de remover completamente, tentar de alguma forma obter informação que possa ser útil. Analisámos vários campos da coluna "Address" e observamos, que continha o indicativo do Estado. Tentamos assim, fazer o parsing dessa informação através do nodo String Replacer, que através duma expressão regular obtém a informação relativa ao estado. Passamos assim de conter uma coluna em que cada valor era um valor único, ou seja 10000 valores diferentes, para após o tratamento, conter apenas 573 valores únicos, podendo assim ser informação relevante para o modelo.

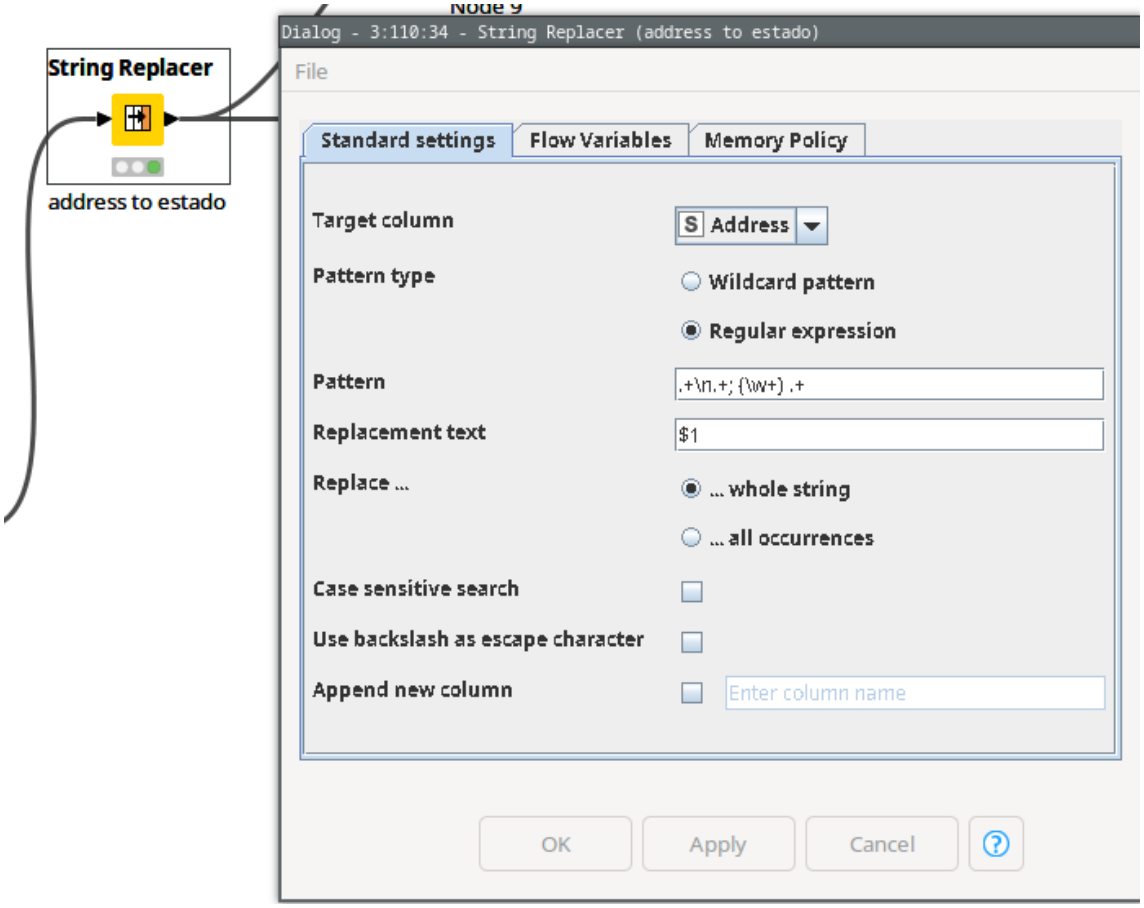


Figure 2: Configuração nodo String Replacer

Pela análise do nodo Box Plot, conseguimos verificar que também existem outliers. De modo, a que estes não distorçam muito a aprendizagem, vamos fazer o tratamento destes.

Para isso, utilizamos o nodo Numeric Outliers, que irá tratar de todos os outliers, fazendo a substituição pelo valor mais próximo permitido.

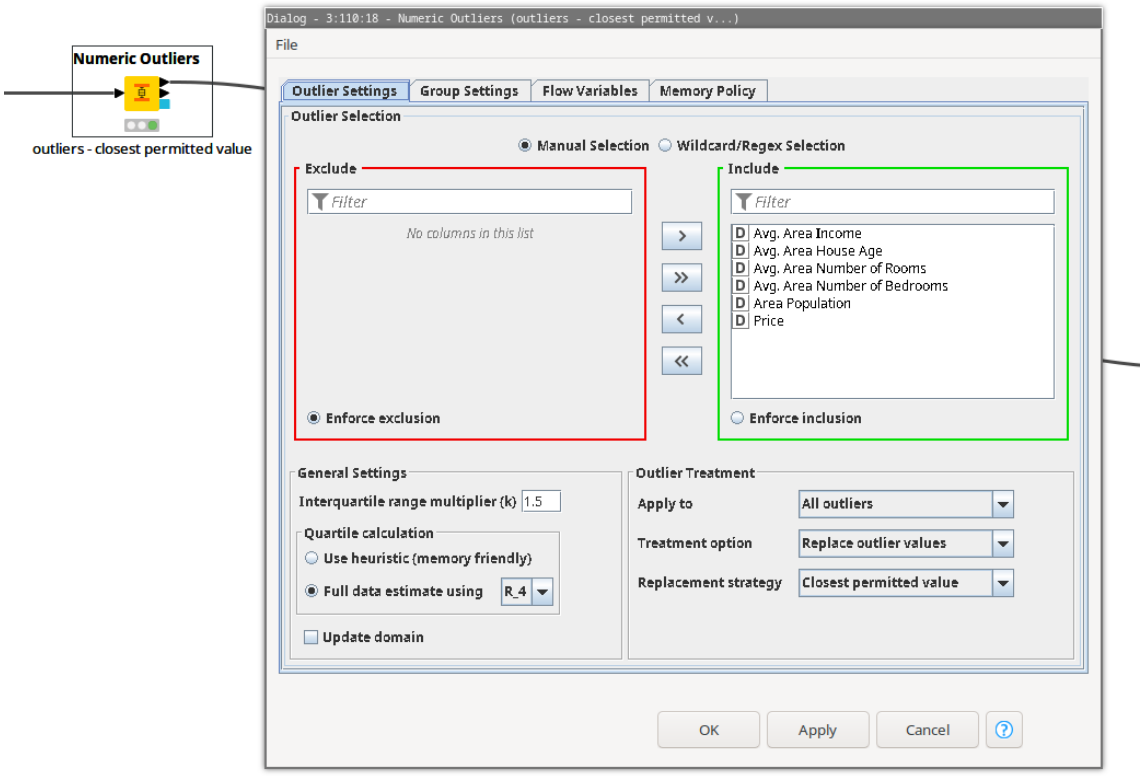


Figure 3: Configuração Nodo numeric Outlier

Foi também feita normalização dos dados de modo a que todos possam dar uma contribuição igual para o resultado.

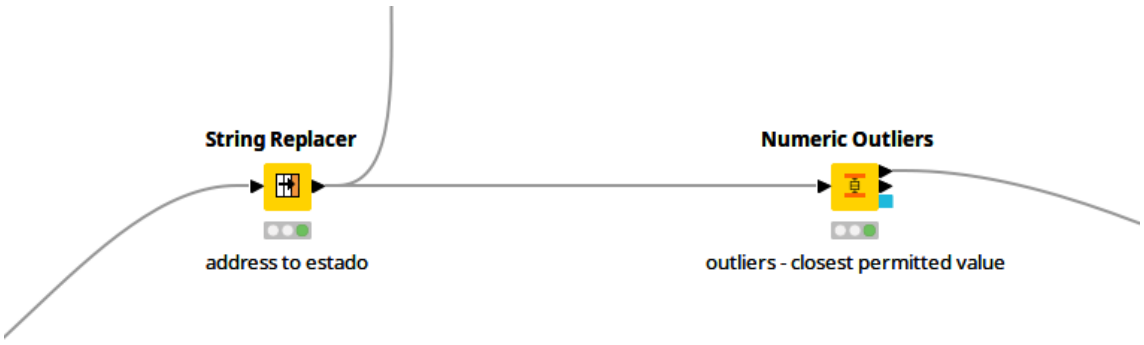


Figure 4: Ligação entre ambos os nodos

4.2 Dataset Stroke

Este dataset contém 12 colunas e 5110 linhas, logo tem de se ter cuidado com a remoção de informação devido a este ser também um dataset pequeno.

Começamos por aplicar um Column Filter para remover o id, visto que este é diferente para cada linha e por isso, não irá conter nenhuma informação que seja útil para os modelos de

aprendizagem.

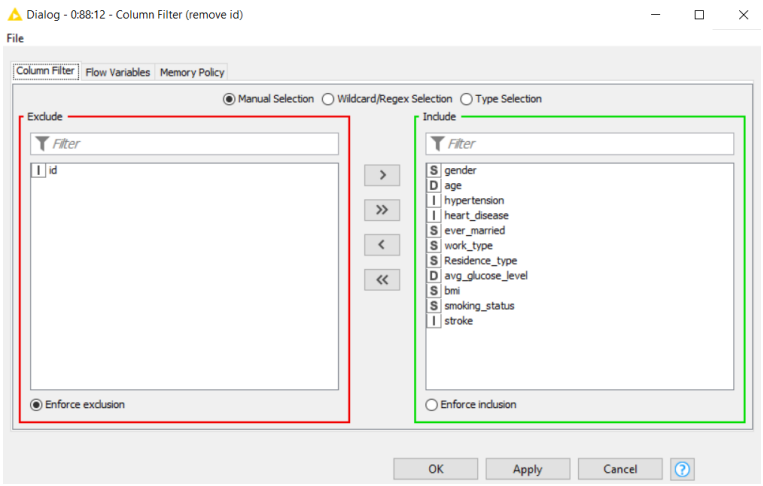


Figure 5: Configuração column Filter

De seguida, convertemos a coluna "Age" de double para inteiro, usando para esse efeito o nodo, Double to Int.

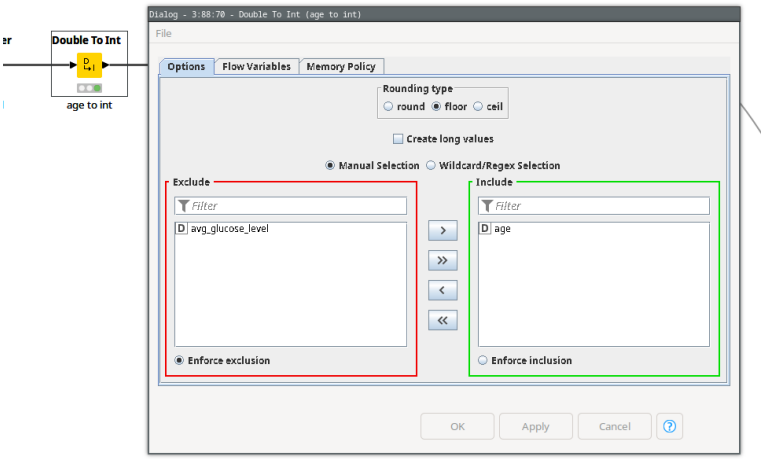


Figure 6: Configuração double to Int

Alteramos a coluna Stroke, de modo a que em vez de inteiro passa-se a String para usarmos depois nos modelos.

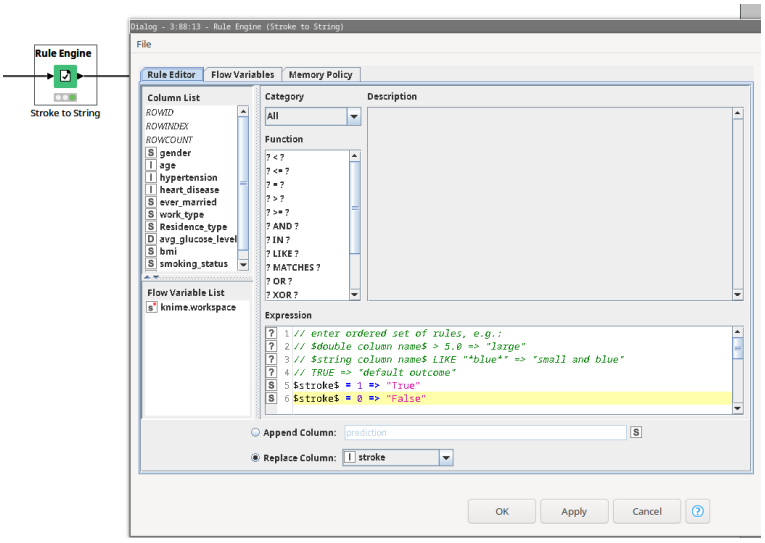


Figure 7: Configuração Rule Engine

Alteramos também a coluna "Smoking Status" de String para valor numérico, de modo a que quanto mais alto o valor, pior. Ou seja, 0 se nunca fumou, 1 se já fumou e 2 se atualmente fuma. Foi feita esta conversão de modo, a que caso o campo fosse Unknown passe para missing value, para posteriormente esse valor passa-se a ser a média. Tornando assim a informação mais útil.

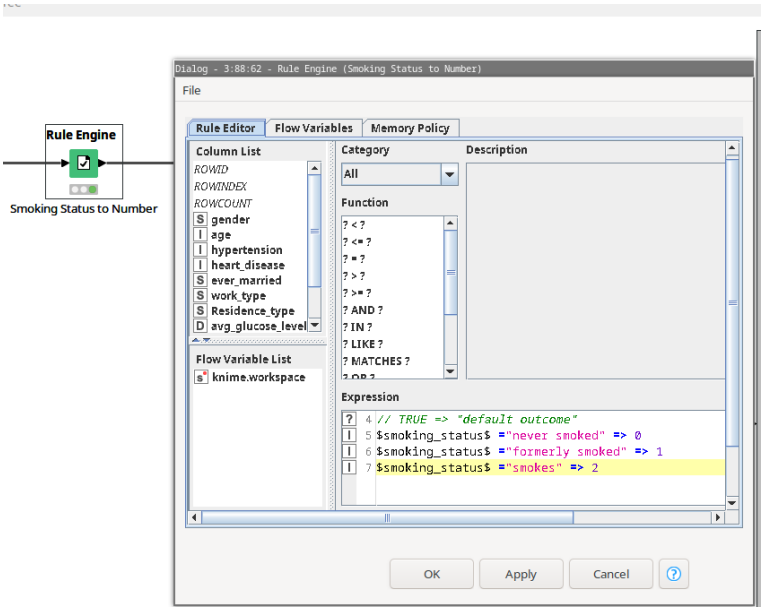


Figure 8: Configuração Rule Engine para a Smoking status

Fizemos a conversão também da coluna bmi, de string para valor numérico com recurso ao nodo String to Number. Esta coluna, inicialmente é uma string mas todos os valores nela presente representam um número, com exceção do valor "N/A" que indica que não existe informação indisponível sobre este valor. Ao fazer a conversão, este caso irá ficar como um valor em falta, este também será tratado posteriormente, passando para a média.

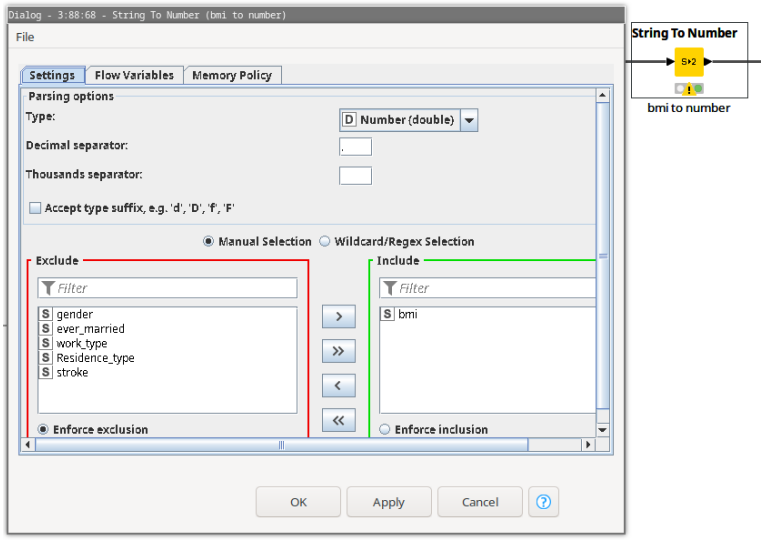


Figure 9: Configuração String to Number

Finalmente, fazemos o tratamento dos valores em falta. Inicialmente, não havia dados em falta propriamente, pois estes estavam colocados de outra forma, no caso da coluna "Smoking Status", seria na forma de Unknow, no caso da coluna de bmi seria na forma de "N/A", estes valores foram colocados devidamente como valores em falta, para agora serem tratados. Para os valores númericos, será atribuída a média.

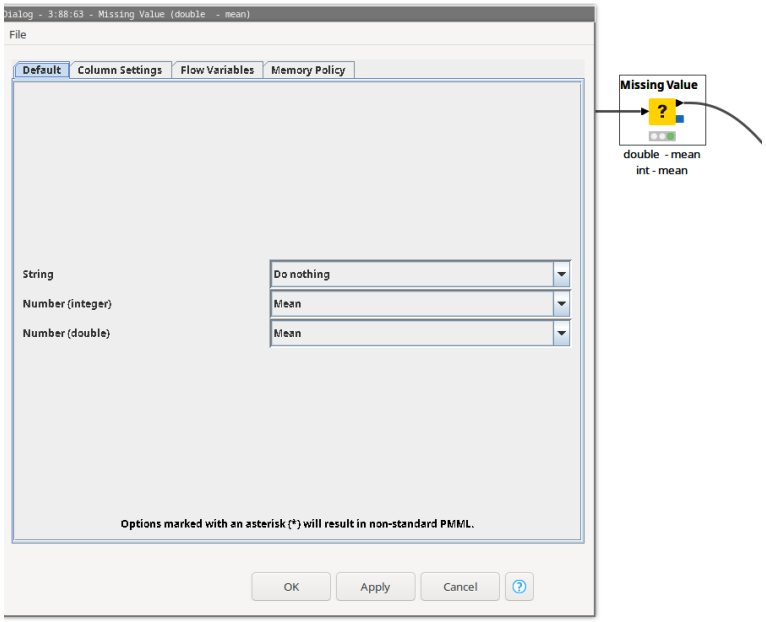


Figure 10: Configuração Missing Value

5 Modelos de Aprendizagem e Parâmetros de Treino

5.1 USA housing regression dataset

Como modelos de aprendizagem para este dataset foram utilizados três: - Modelo de Linear Regression - Modelo de Redes Neuronais - Modelo de Decision Tree

Inicialmente, começamos por fazer o modelo de regressão linear.

Para este, começamos por algo bastante simples, utilizamos o modelo treino/teste com uma razão de 80/20, 80% para treino e 20% para teste.

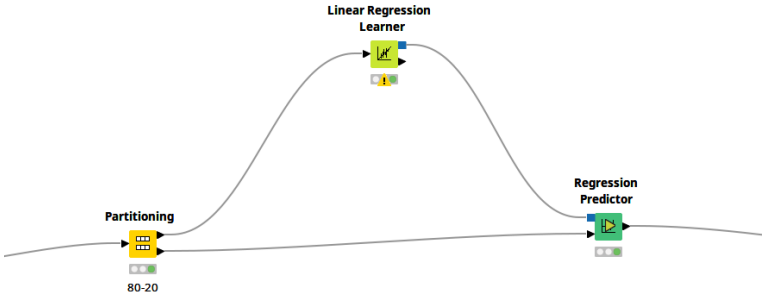


Figure 11: modelo de regressão

Ainda usando modelo de regressão linear, mas desta vez em vez de uma simples partição foi utilizado Cross Validation, com 10 folds.

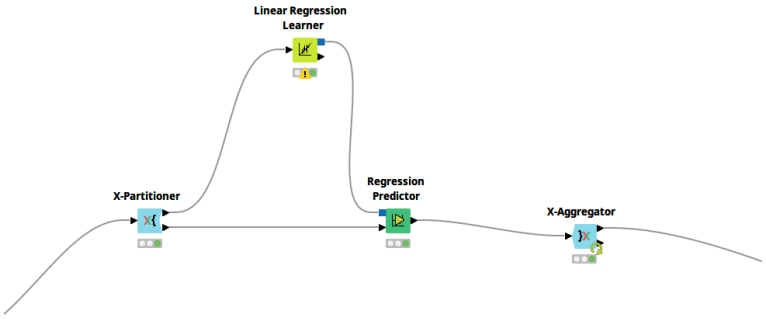


Figure 12: modelo de regressão com Cross Validation

Utilizamos também o modelo de Redes Neuronais.

Para tal, começamos por fazer remover a coluna dos Address, pois esta é do formato String, e só são aceites valores numéricos. De seguida, utilizamos o modelo treino/teste com uma razão de 80/20, 80% para treino e 20% para teste.

Os parâmetros utilizados para o RProp MLP Learner foram estes que se encontram na imagem abaixo.

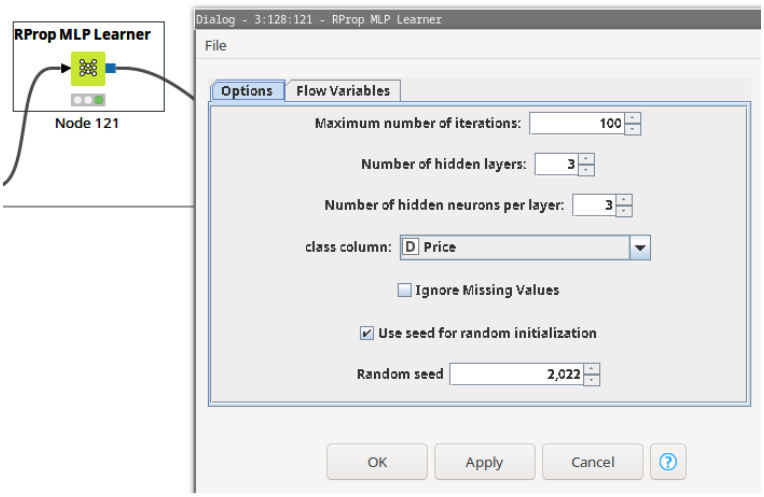


Figure 13: Configração nodo RProp

Por último, utilizamos o Modelo de Árvores de Decisão.

Aqui, tentamos repensar o problema para, em vez de ser resolvido com regressão, ser com classificação. Para tal, foram necessárias fazer algumas mudanças.

Utilizamos Bins para a coluna do Price, para que este passasse para três categorias: Low, Medium, High. Com o recurso do nodo Auto-Binner juntamente com um Rule Engine, fizemos essa substituição. Foi utilizado um modelo de Cross Validation com fold de 10.

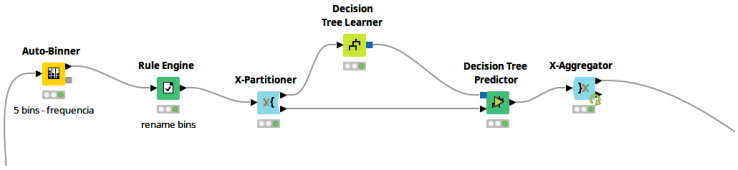


Figure 14: Modelo Árvore de Decisão

5.2 Dataset Stroke

Como modelos de aprendizagem para este dataset foram utilizados três: - Modelo de Decision Tree - Modelo de Random Forest - Modelo de Clustering

Foi utilizado o modelo de Decision Tree, de duas formas.

Primeiramente, com um modelo de Cross Validation com fold de 10.

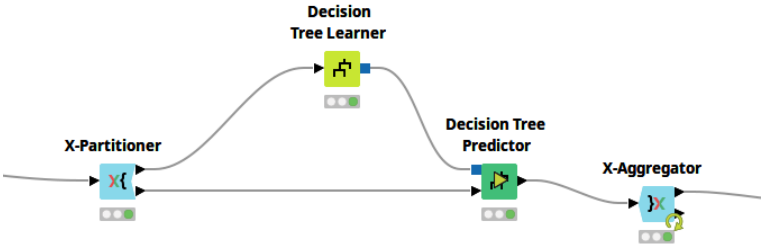


Figure 15: Modelo Cross Validation

Posteriormente, com um modelo de treino/teste com uma razão de 80/20 mas com a adição do nodo SMOTE para o dataste de teste, este nodo irá acrescentar linhas ao dataset para que haja uma igual percentagem de stroke a True e a False, como no dataset a percentagem de stroke a False é maior, o nodo SMOTE irá adicionar linhas artificiais com Stroke a True, com base em linhas reais. Deste modo, é possível haver uma igual razão, numa tentativa de uma melhor aprendizagem do modelo.

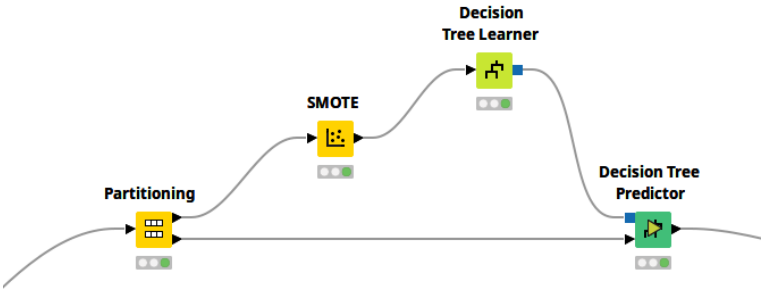


Figure 16: Adição nodo SMOTE ao modelo

Para estes modelos, foram utilizados os seguintes parâmetros para o nodo Decision Tree Learner:

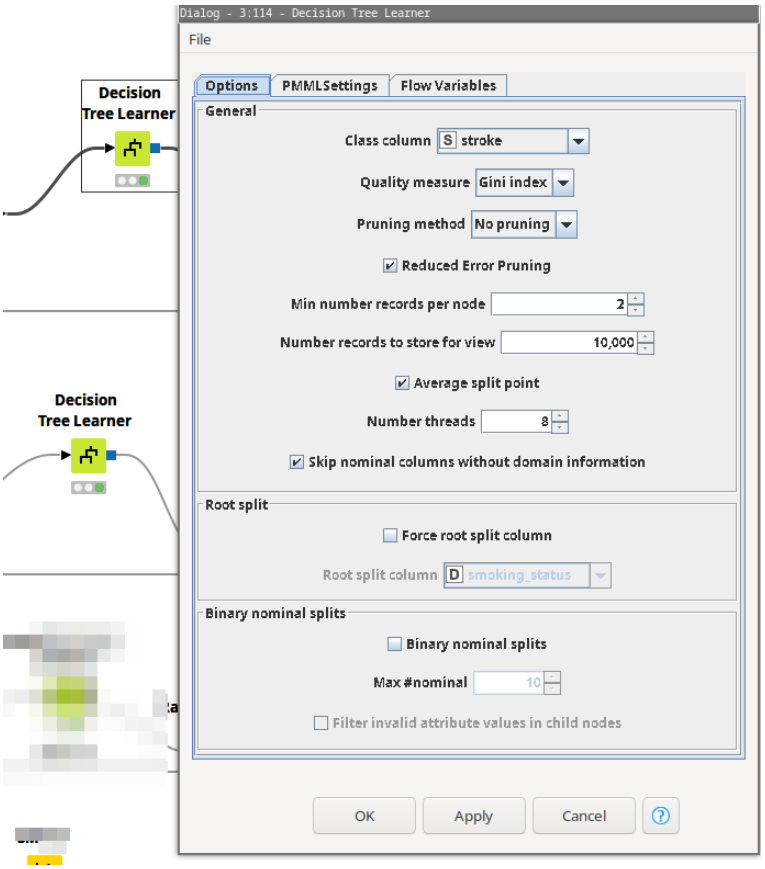


Figure 17: Configuração do nodo Decision Tree Learner

Foi também utilizado o modelo de Random Forest. Para este, utilizamos o modelo de Cross Validation com fold de 10.

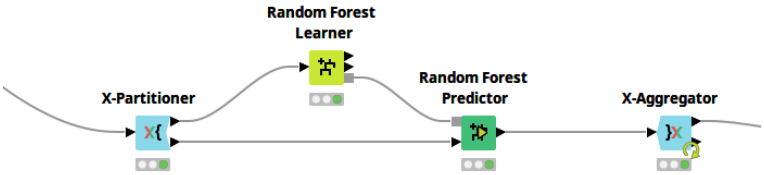


Figure 18: Modelo de Cross Validation

Como parâmetros para o nodo do Random Forest Learner, foram utilizados os seguintes:

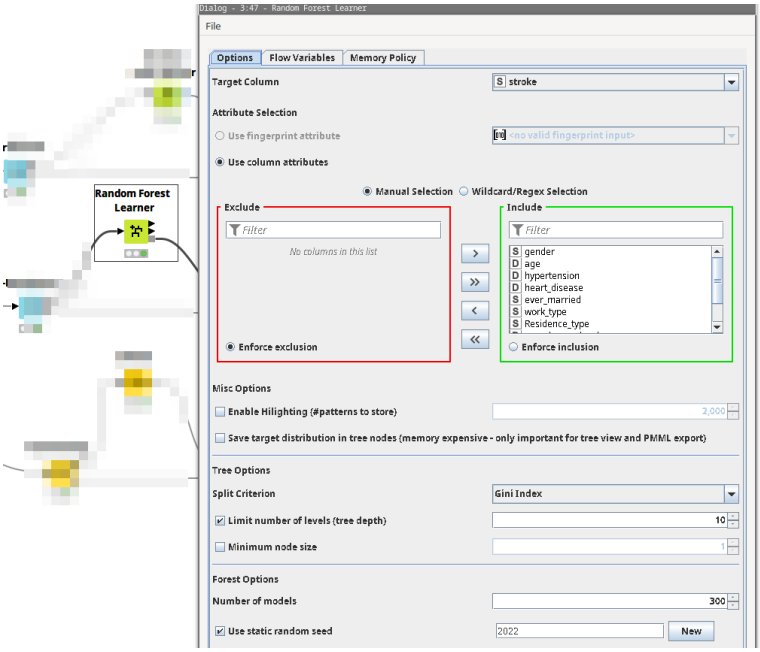


Figure 19: Configuração do nodo Random Forest Learner

Por fim, utilizamos o modelo de Clustering. Para este utilizamos um modelo de treino/teste com uma razão de 80/20. Neste foi também adicionado o nodo SMOTE,já explicado anteriormente, para o dataset de teste. Após o modelo, foi utilizado um Rule Engine para fazer a devida conversão.

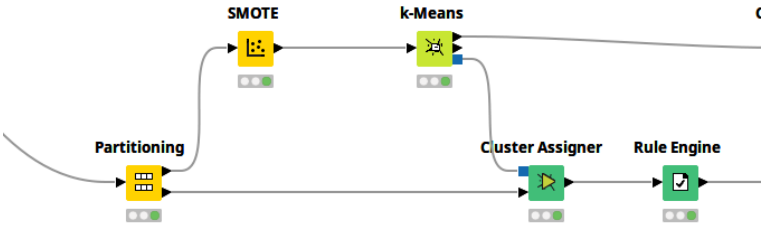


Figure 20: Modelo de Clustering

Como parâmetros para o nodo do k-Means, foram utilizados os seguintes:

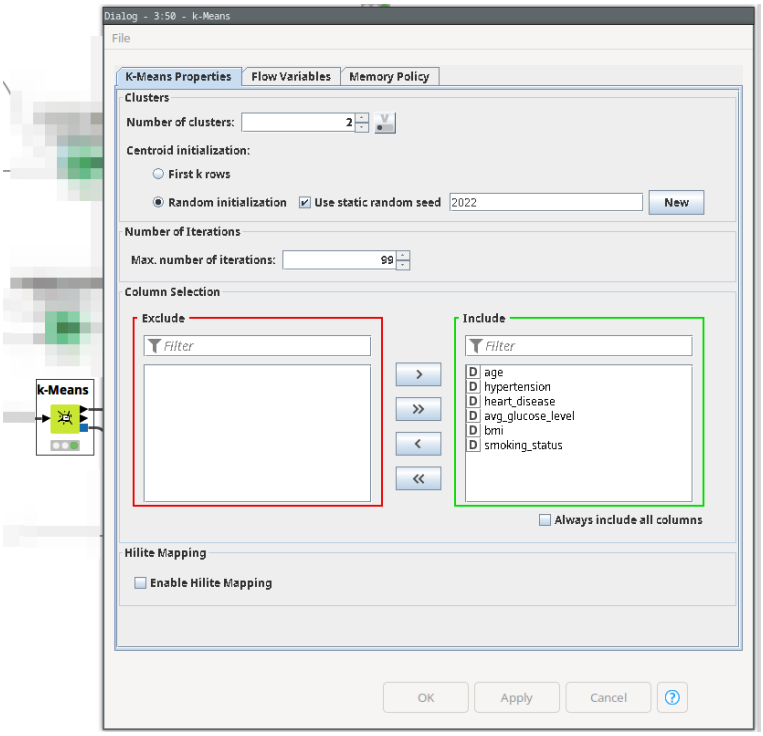


Figure 21: Configuração do nodo K-means

6 Resultados Obtidos

6.1 USA housing regression dataset

Para este dataset, os resultados obtidos foram bons para todos os modelos.

Para o modelo de Linear Regression com o modelo de Treino/Teste, os resultados obtidos foram:

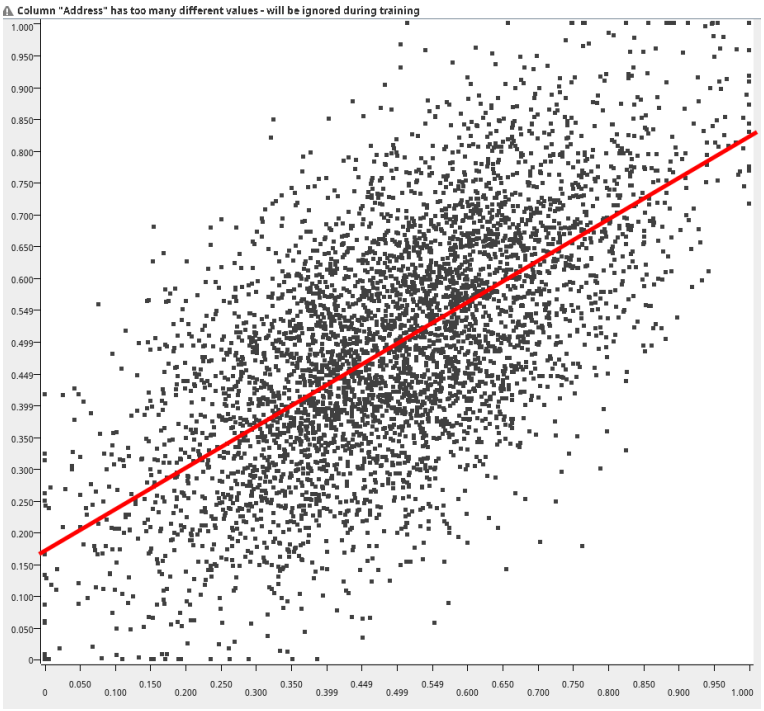


Figure 22: Linear Regression CV

R^2:	0.916
Mean absolute error:	0.043
Mean squared error:	0.003
Root mean squared error:	0.054
Mean signed difference:	0
Mean absolute percentage error:	NaN
Adjusted R^2:	0.916

Figure 23: Linear Regression CV Resultados

Para o modelo de Linear Regression com o modelo Cross Validation, os resultados obtidos foram:

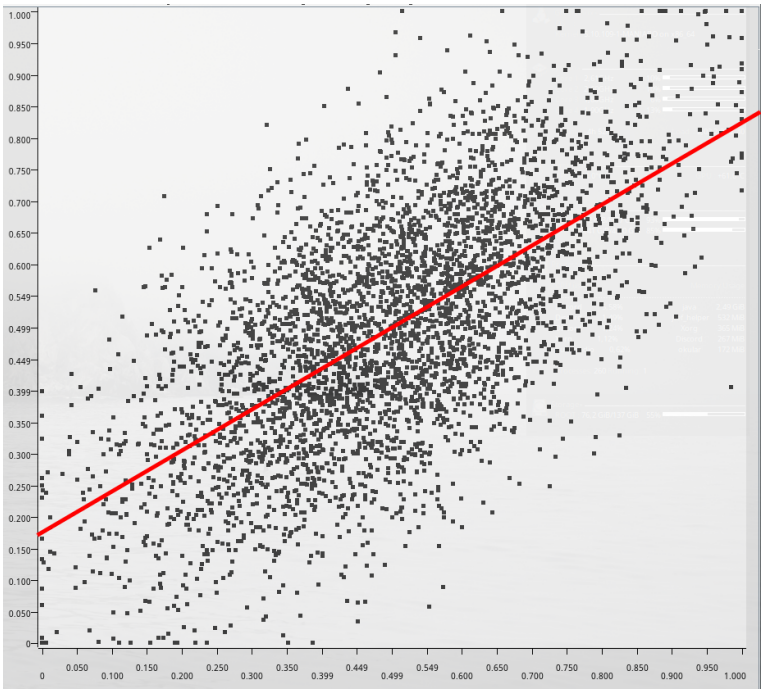
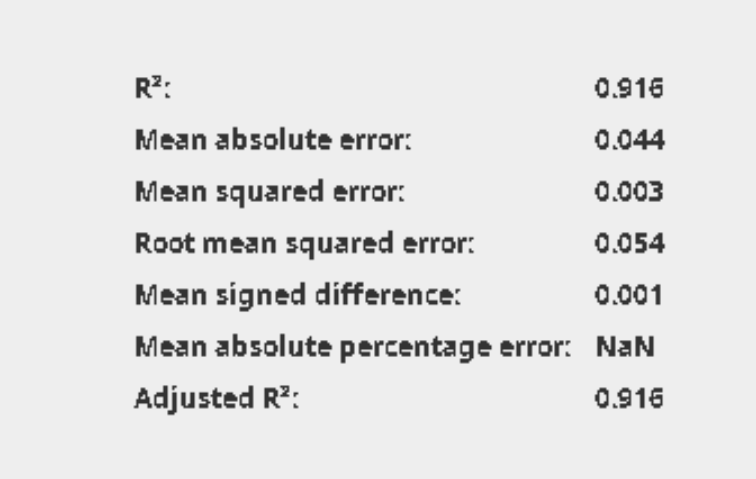


Figure 24: Linear Regression Simple

R^2:	0.925
Mean absolute error:	0.041
Mean squared error:	0.003
Root mean squared error:	0.051
Mean signed difference:	0.002
Mean absolute percentage error:	NaN
Adjusted R^2:	0.925

Figure 25: Linear Regression Simple Resultados

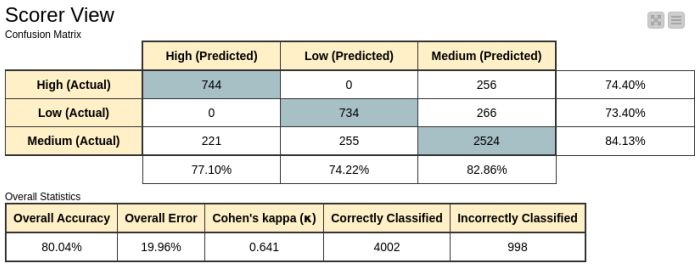
Para o modelo de Redes Neuronais com o modelo Cross Validation, os resultados obtidos foram:



R^2:	0.916
Mean absolute error:	0.044
Mean squared error:	0.003
Root mean squared error:	0.054
Mean signed difference:	0.001
Mean absolute percentage error:	NaN
Adjusted R^2:	0.916

Figure 26: Redes Neuronais Resultado

Para o modelo de Decision Tree com o modelo Cross Validation, os resultados obtidos foram:



Scorer View

Confusion Matrix

	High (Predicted)	Low (Predicted)	Medium (Predicted)	
High (Actual)	744	0	256	74.40%
Low (Actual)	0	734	266	73.40%
Medium (Actual)	221	255	2524	84.13%
	77.10%	74.22%	82.86%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
80.04%	19.96%	0.641	4002	998

Figure 27: Scorer com os resultados obtidos

Os resultados foram bastante similares para os modelos de Regressão, o modelo de Linear Regression com o modelo de Treino/Teste teve um R-Square um bocado superior, mas não chega a ser uma diferença muito significativa perante os outros dois.

Já o modelo de Decision Tree obteve um resultado relativamente bom consoante o contexto. Foram feitos ajustes relativamente ao número de bins a usar, consoante os resultados obtidos. Chegou-se à conclusão que o número ideal seria 5, mas com uma distribuição de 1 para Low, 3 para Medium e 1 para High. Deste modo, caso fosse Low obteve uma grande facilidade em reconhecer que não era High, mas a situação já era mais difícil para distinguir se era Medium. O mesmo aconteceu para o High.

6.2 Stroke dataset

Para este dataset os resultados já não foram tão bons. Apesar da eficácia global ser relativamente boa em quase todos os modelos, a eficácia alcançada para prever se teve um Stroke, não foi muito boa. Devido ao nosso Dataset ser relativamente pequeno, e haver uma discrepância muito grande entre linhas com stroke True e linhas com stroke False, alguns dos modelos foram incapazes de conseguir prever com eficácias as linhas a True.

Para o modelo de Decision Tree com o modelo Treino/Teste e com o SMOTE, os resultados obtidos foram:

Scorer View

Confusion Matrix

	False (Predicted)	True (Predicted)	
False (Actual)	920	52	94.65%
True (Actual)	43	7	14.00%
	95.53%	11.86%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
90.70%	9.30%	0.080	927	95

Figure 28: Resultado para o modelo Tree Learner com SMOTE

Para o modelo de Decision Tree com o modelo Cross Validation, os resultados obtidos foram:

Scorer View

Confusion Matrix

	False (Predicted)	True (Predicted)	
False (Actual)	4669	192	96.05%
True (Actual)	219	29	11.69%
	95.52%	13.12%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
91.96%	8.04%	0.082	4698	411

Figure 29: Resultado para o modelo de Decision tree com Cross Validation

Para o modelo de Random Forest com o modelo Cross Validation, os resultados obtidos foram:

Scorer View

Confusion Matrix

	False (Predicted)	True (Predicted)	
False (Actual)	4857	4	99.92%
True (Actual)	248	1	0.40%
	95.14%	20.00%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
95.07%	4.93%	0.006	4858	252

Figure 30: Resultado para o modelo de Random Forest

Para o modelo de Clustering com o modelo Treino/Teste com SMOTE, os resultados obtidos foram:

Scorer View

Confusion Matrix

	False (Predicted)	True (Predicted)	
False (Actual)	839	133	86.32%
True (Actual)	29	21	42.00%
	96.66%	13.64%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
84.15%	15.85%	0.143	860	162

Figure 31: Resultado para o modelo de Clustering

O pior resultado foi obtido com o modelo de Random Forest, que considerou quase tudo como falso e, por isso, obteve uma eficácia muito reduzida a prever quem teve um Stroke.

Os modelos de Decision Tree, tiveram desempenhos parecidos, mas estes também não obtiveram uma boa eficácia a prever quem teve um Stroke. Por fim, o modelo de Clustering, que apesar de ter uma eficácia global menor do que as dos restantes, foi o que teve uma eficácia mais elevada a prever quem teve Stroke.

6.3 Sugestões e recomendações após análise dos resultados obtidos e dos modelos desenvolvidos

Para o Dataset do Stroke, seria benéfico para todos os modelos a utilização do SMOTE, mas este não é compatível com o modelo de Cross Validation, por isso foi apenas feito nos modelos que não utilizaram esse modelo.

7 Conclusão

Nesta fase final do projeto constatamos que conseguimos cumprir com tudo o que nos foi pedido.

Consideramos que foi um projeto bastante interessante e desafiante, visto que nos obrigou a interligar conceitos e matéria lecionados nas aulas teóricas com a componente mais prática que nos foi instruída ao longo das aulas PLs, nomeadamente, na conceção de modelos de aprendizagem.

Numa fase inicial, sentimos um pouco de dificuldade em colocar em prática o que fomos aprendendo ao longo do semestre, mas com o empenho e dedicação de todos os elementos conseguimos superar as nossas dificuldades.

Assim, enquanto grupo conseguimos distribuir bem o trabalho entre todos. Ajudamo-nos mutuamente e, de forma geral, tivemos um aproveitamento positivo.

Concluindo, este projeto ajudou-nos a desenvolver novas aptidões e a consolidar toda a matéria lecionada em aula.