Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Aprendizagem e Decisão Inteligentes

Ano Letivo de 2023/2024

Conceção de modelos de aprendizagem e decisão

António Filipe Castro Silva(a100533) Diogo Rafael dos Santos Barros(a100600)
Duarte Machado Leitão(a100550) Pedro Emanuel Organista Silva(a100745)

3 de maio de 2024

ADI

Índice

Lista de Figuras

1 Introdução

Este relatório foi realizado no âmbito do trabalho prático da Unidade Curricular de Aprendizagem e Decisões Inteligentes, onde nos foi proposta a conceção de modelos de aprendizagem e decisão.

Este trabalho prático divide-se em duas tarefas separadas, mas com o mesmo pretexto. A primeira tarefa consiste na consulta, exploração, análise e preparação de um *dataset* escolhido pelo grupo. A segunda tarefa consiste na mesma base, mas desta vez sobre um *dataset* escolhido pelos professores.

Como o *dataset* escolhido pelos professores demonstrava normalmente um problema de **Classificação**, o *dataset* que resolvemos escolher demonstra normalmente um problema de **Regressão**.

2 Tarefa: Dataset Grupo

Esta tarefa consiste na escolha de um determinado *dataset*, que o nosso grupo de trabalho analisou e recolheu dados sobre o mesmo, para o sucesso deste trabalho prático.

Este *dataset* contém várias características relacionadas a carros e os seus atributos. Estes atributos fornecem informações valiosas sobre os fatores que influenciam os preços dos carros e podem ser usados para desenvolver modelos de previsão para estimar o preço de venda dos carros. Este *dataset* é considerado como um **Problema de Regressão.**

A metodologia que foi usada no processo da resolução do problema é o **CRISP-DM**, que é constituído pelas seguintes 6 etapas:

- **Estudo do negócio**
- **Estudo dos dados**
- **Preparação dos dados**
- **Modelação**
- **Avaliação**
- **Desenvolvimento**

NOTA: neste *dataset* iremos apenas usar as 5 primeiras etapas.

2.1 Estudo do negócio

Os objetivos a cumprir com este *dataset* e com o uso de técnicas de modelação existentes na aplicação **KNIME** são:

- Analisar/Avaliar a situação e os dados existentes no *dataset*;
- Tratar dos erros, valores inexistentes e outras irregularidades de dados, que possam existir no *dataset*;
- Se necessário, devo usar gráficos, matrizes e outro tipos de visualização de dados para retirar informações e analisar aspetos para melhorar o projeto.
- Descobrir a partir de modelos de aprendizagem, os atributos que mais afetam **o preço dos veículos atribuídos pelos seus fabricantes** e através destes conseguir prever o preço dos mesmos.

2.2 Estudo dos dados

O *dataset* para esta tarefa foi retirado do site **Kaggle** e este *dataset* contém **428 linhas e 15 colunas.** Os atributos/colunas deste respetivo *dataset* são os seguintes:

- 1. **Make:** representa a marca ou o fabricante do carro;
- 2. **Model:** representa o modelo do carro em questão;
- 3. **Type:** representa o tipo de veículo que o carro em questão representa;
- 4. **Origin:** representa o local de origem de fabrico de um determinado veículo;
- 5. **DriveTrain:** representa o tipo de sistema de transmissão de um veículo, que é responsável por transmitir a potência do motor para as rodas e, consequentemente, movimentar o veículo **(All/Front/Rear)**;
- 6. **MSRP:** representa o preço que o fabricante recomenda ao vendedor para atribuir ao carro para o vender **(USD)**;
- 7. **Invoice :** representa o valor de compra do veículo diretamente da fábrica **(USD)**;
- 8. **EngineSize:** representa ao volume total de deslocamento dos cilindros num motor de um carro **L - em litros**;
- 9. **Cylinders:** representa o número de cilindros que cada carro tem;
- 10. **Horsepower:** representa a medida de potência do motor de um carro **(cv)**;
- 11. **MPG_City:** representa as milhas que um carro pode percorrer com um *galon* de combustível enquanto é dirigido pela cidade **(gal)**;
- 12. **MPG_Highway:** representa as milhas que um carro pode percorrer com um *galon* de combustível enquanto é dirigido pela auto-estrada **(gal)**;
- 13. **Weight:** representa o peso total do veículo em **pounds (lbs);**
- 14. **Wheelbase:** representa a distância entre os eixos dianteiro e traseiro do veículo, em **inches (in);**
- 15. **Length:** representa o comprimento total do veículo, em **inches (in).**

O atributo **"MSRP"** deste problema é aquele que iremos prever com o desenvolvimento de modelos de regressão e também vai ser utilizado para descobrir a qualidade dos outros atributos do *dataset*. Este estudo de dados foi feito depois do tratamento de dados que vamos explicar na próxima secção.

2.2.1 MSRP

Na seguinte tabela, o **MSRP** como o atributo principal a analisar tem uma variedade de valores entre os **10280 e os 192465 USD**, sendo que a média calculada está à volta dos **32804,5 USD.**

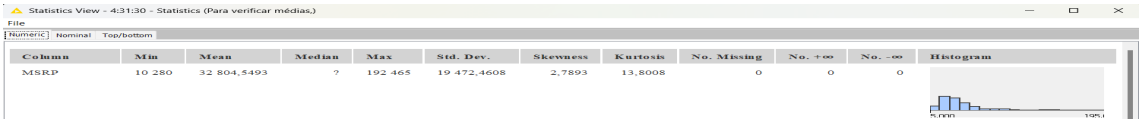

Figura 2.1: Dados do MSRP no Statistics

2.2.2 Make

Este atributo indica o **nome do fabricante do veículo.** Através da análise da correlação, verificamos que a **Make** de um veículo não tem um efeito suficiente para influenciar a previsão do preço, **MSRP**. No entanto, verificamos que a marca mais cara é a **Porsche**, com um valor médio de **83565 USD** e a mais barata é a **Scion**, com um valor médio de **13565 USD**.

2.2.3 Model

Este atributo indica o **nome do modelo do veículo.** Através da análise da correlação, verificamos que o **Model** de um veículo não tem um efeito nenhum para influenciar a previsão do preço, **MSRP**. No entanto, verificamos que o modelo mais caro é o **911 GT2 2dr**, com um valor médio de **192465 USD** e o mais barato é o **Rio 4dr manual**, com um valor médio de **10280 USD**.

2.2.4 Type

Este atributo identifica todo o tipo de veículo que uma marca pode corresponder. Através da análise da correlação, verificamos que o **Type**, tem uma **correlation** de valor de **0.0148** com o **MSRP**. Como podemos ver na tabela em baixo, o **Type** de veículo mais referenciado neste *dataset* é o **Sedan**, com um preço médio de **29773.6 USD**:

TYPES	QUANTIDADE DE CADA TYPE EM PERCENTAGEM	MSRP MÉDIO DE CADA TYPE
SUV	14.08%	34790.3 USD
SEDAN	61.5%	29773.6 USD
SPORTS	11.03%	54533.3 USD
WAGON	7.04%	28840.5 USD
TRUCK	5.63%	24941.4 USD
HYBRID	0.7%	19920 USD

Figura 2.2: Quantidade e MSRP associados ao Type.

2.2.5 Origin

Este atributo identifica o local de origem de um veículo. Ao analisarmos a correlação, verificamos que a **Origin** possui uma correlação com o **MSRP**, valorizada em 0.1515, o que sugere uma relação positiva, que simboliza uma pequena influência positiva na previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, a **Origin** de veículo mais referenciada neste *dataset* é a **Asia**, com o preço médio mais baixo de **24719.4 USD**:

ORIGIN	QUANTIDADE DE CADA ORIGIN EM PERCENTAGEM	MSRP MÉDIO DE CADA ORIGIN
EUROPE	28.87%	48349.8 USD
USA	34.51%	28377.4 USD
ASIA	36.62%	24719.4 USD

Figura 2.3: Quantidade e MSRP associados à Origin.

2.2.6 DriveTrain

Este atributo identifica o tipo de transmissão de um veículo. Ao analisarmos a correlação, verificamos que a **DriveTrain** possui uma correlação com o **MSRP**, valorizada em 0.1245, o que sugere uma relação positiva, que simboliza uma pequena influência na previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o **DriveTrain** de veículo mais referenciado neste *dataset* é o **Front**, com o preço médio mais baixo de **24782.6 USD**:

DRIVETRAIN	QUANTIDADE DE CADA DRIVETRAIN EM PERCENTAGEM	MSRP MÉDIO DE CADA DRIVETRAIN
ALL	21.6%	36483.5 USD
FRONT	53.05%	24782.6 USD
REAR	25.35%	46457.4 USD

Figura 2.4: Quantidade e MSRP associados à DriveTrain.

2.2.7 Invoice

Este atributo simboliza o preço que o vendedor pagou ao fabricante do carro para o puder obter e vender pelo preço de **MSRP**. Ao analisarmos a correlação, verificamos que o **Invoice** possui uma correlação com o **MSRP**, valorizada em 0.9991, o que sugere uma relação positiva muito forte, simbolizando até uma grande influência positiva na previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o **Invoice** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	INVOICE MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	15215.6 USD
[20324.75 ; 27807.5]	22160.4 USD
[27807.5 ; 39225]	30182.4 USD
[39225 ; 192465]	52531.9 USD

Figura 2.5: MSRP associados à Invoice.

2.2.8 EngineSize

Este atributo representa o volume de deslocamento dos cilindros num motor de um carro em **litros (L)**. Ao analisarmos a correlação, verificamos que o **EngineSize** possui uma correlação com o **MSRP**, valorizada em 0.6731, o que sugere uma relação positiva forte, podendo até influenciar positivamente a previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o **EngineSize** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	ENGINESIZE MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	2.2 L
[20324.75 ; 27807.5]	3.07 L
[27807.5 ; 39225]	3.39 L
[39225 ; 192465]	4.16 L

Figura 2.6: MSRP associados ao EngineSize.

2.2.9 Cylinders

Este atributo representa o número de cilindros que cada carro tem. Ao analisarmos a correlação, verificamos que o **Cylinders** possui uma correlação com o **MSRP**, valorizada em 0.73, o que sugere uma relação positiva forte, podendo até influenciar positivamente a previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o número de **Cylinders** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	CYLINDERS MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	4.27
[20324.75 ; 27807.5]	5.53
[27807.5 ; 39225]	6.10
[39225 ; 192465]	7.33

Figura 2.7: MSRP associados aos Cylinders.

2.2.10 Horsepower

Este atributo representa a potência de um motor de um carro **(cv)**. Ao analisarmos a correlação, verificamos que o **Horsepower** possui uma correlação com o **MSRP**, valorizada em 0.8666, o que sugere uma relação positiva forte, podendo até influenciar positivamente a previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o **Horsepower** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	HORSEPOWER MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	141.8 cv
[20324.75 ; 27807.5]	193.1 cv
[27807.5 ; 39225]	231.6 cv
[39225 ; 192465]	296.9 cv

Figura 2.8: MSRP associados ao Horsepower.

2.2.11 MPG_City e MPG_Highway

Estes atributo representam o consumo de um carro dentro de uma cidade ou numa auto-estrada **(gal)**. Ao analisarmos as correlações, verificamos que o **MPG_City e MPG_Highway** possuem correlações com o **MSRP**, valorizadas em -0.7051 e -0.6169, o que sugerem uma relação negativa forte, podendo até influenciar negativamente a previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o **MPG_City e o MPG_Highway** vai diminuindo à medida que o **MSRP** vai aumentando:

INTERVALOS QUANTIS DE MSRP	MPG_CITY MÉDIO DE CADA INTERVALO	MPG_HIGHWAY MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	25.2 gal	31.9 gal
[20324.75 ; 27807.5]	20.3 gal	27.2 gal
[27807.5 ; 39225]	18.1 gal	24.9 gal
[39225 ; 192465]	16.7 gal	23.4 gal

Figura 2.9: MSRP associados ao MPG_City e MPG_Highway.

2.2.12 Weight

Este atributo representa o peso total de um veículo **(lbs)**. Ao analisarmos a correlação, verificamos que o **Weight** possui uma correlação com o **MSRP**, valorizada em 0.6703, o que sugere uma relação positiva forte, podendo até influenciar positivamente a previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, o **Weight** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	WEIGHT MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	2865.7 lbs
[20324.75 ; 27807.5]	3509.7 lbs
[27807.5 ; 39225]	3835.6 lbs
[39225 ; 192465]	4112.6 lbs

Figura 2.10: MSRP associados ao Weight.

2.2.13 Wheelbase

Este atributo representa a distância entre o eixo dianteiro e traseiro de um veículo em **(in)**. Ao analisarmos a correlação, verificamos que o **Wheelbase** possui uma correlação com o **MSRP**, valorizada em 0.3977, o que sugere uma relação positiva, que pode ter alguma influência positiva na previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, a **Wheelbase** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	WHEELBASE MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	103.2 in
[20324.75 ; 27807.5]	108.6 in
[27807.5 ; 39225]	110.3 in
[39225 ; 192465]	110.6 in

Figura 2.11: MSRP associados à Wheelbase.

2.2.14 Length

Este atributo representa o comprimento total de um veículo em **(in)**. Ao analisarmos a correlação, verificamos que a **Length** possui uma correlação com o **MSRP**, valorizada em 0.3177, o que sugere uma relação positiva, que pode ter alguma influência positiva na previsão do preço de **MSRP**. Como podemos ver na tabela em baixo, a **Length** vai aumentando à medida que o **MSRP** também aumenta:

INTERVALOS QUANTIS DE MSRP	LENGTH MÉDIO DE CADA INTERVALO
[10280 ; 20324.75]	177.8 in
[20324.75 ; 27807.5]	188.7 in
[27807.5 ; 39225]	188.7 in
[39225 ; 192465]	190.5 in

Figura 2.12: MSRP associados à Length.

2.3 Preparação dos dados

Inicialmente, o nosso grupo reparou que os valores das colunas, **MSRP e Invoice**, tinham os valores em strings, com os símbolos do **dólar ($)** e com pontuação já feita para as casas dos milhares. Então para resolver essa situação, utilizámos um nodo do **KNIME**, chamado de **Java Snippet**, onde se aplicaram os seguintes "comandos":

- **1:** out_MSRP = c_MSRP.replace("$",).replace(",",);
- **2:** out_Invoice = c_Invoice.replace("$",).replace(",",);

Depois de removida a simbologia dos preços, usamos o nodo **String to Number**, para passar de *strings* para *doubles* os valores das colunas do **MSRP (coluna a prever), Invoice e EngineSize** para condizer com o resto do *dataset* e para podermos aplicar os nodos de regressão com sucesso.

Em seguida, foi aplicado o nodo **Missing Value**, para retirar 2 linhas do *dataset* que tinham **dois missing values** na coluna dos **Cylinders**, ficando com um *dataset* de **426 linhas**. Visto que são somente dois *missing values*, consideramos que não é necessário diferentes tipos de tratamentos para os *Missing Values*.

Apesar de não aparecer na **fig. 2.13**, foi usado um **Column Filter**, no metanodo de **Modelação dos Dados**, para retirar a coluna do **Model** de um veículo, visto que não apresentava qualquer correlação para ajudar a prever o **MSRP** e porque todos os modelos eram diferentes, logo serviam como uma espécie de **ID** para todos os veículos do *dataset*.

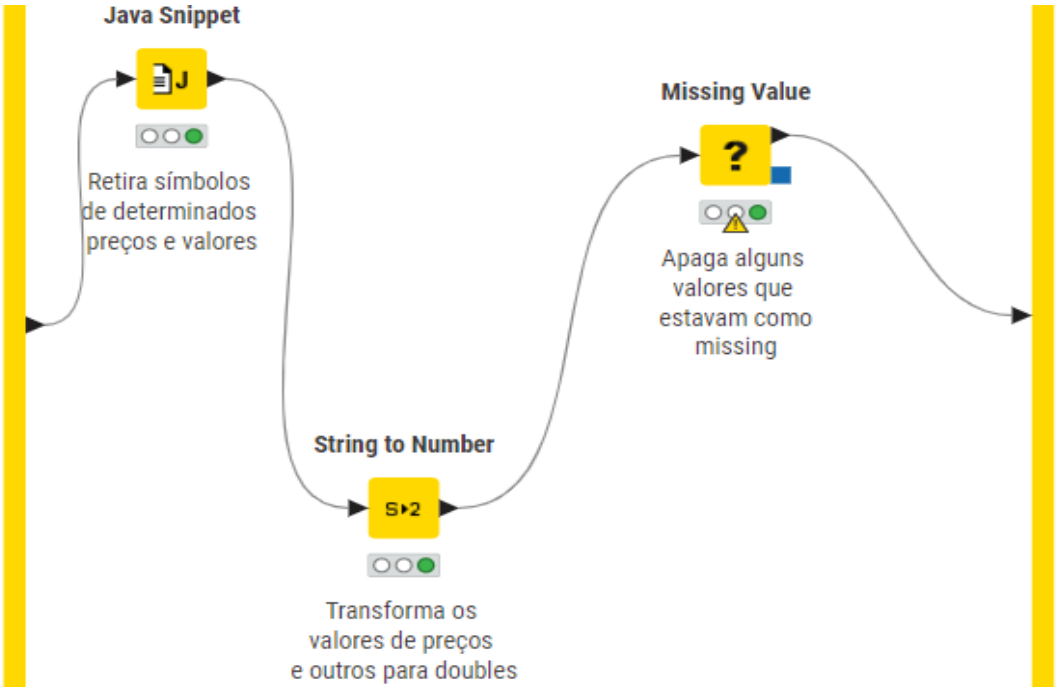

Figura 2.13: Nodos usados para preparar os dados para o KNIME.

2.4 Modelação

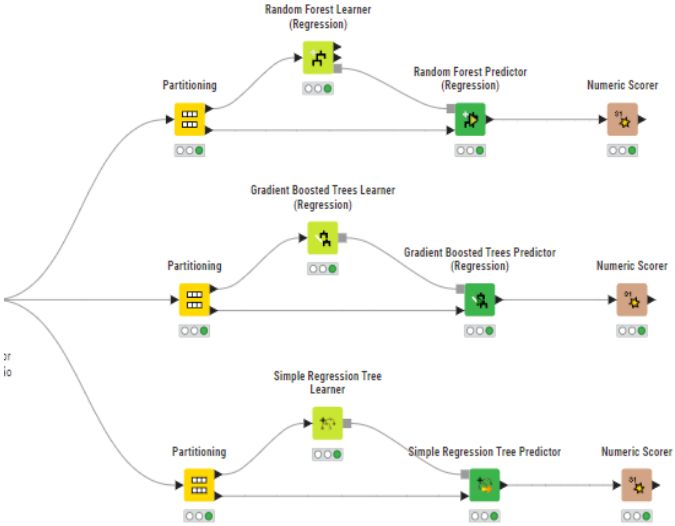

Figura 2.14: Modelação com 3 nodos de regressão.

Como se pode ver na imagem em cima, usámos o nodo **Partitioning**, com a divisão de partição feita de forma **Random** e com uma valorização relativa igual a **70%.** Em seguida, foram aplicados os 3 métodos de regressão seguintes e os seus nodos:

- **Random Forest Learner (RFL)**, onde usámos o **MSRP**, como a **target column** a prever nos diferentes 100 modelos que vão ser calculados pelo nodo;
- **Gradient Boosted Trees Learner (GBTL)**, onde usámos o **MSRP**, como a **target column** a prever nos diferentes 100 modelos que vão ser calculados e de onde e vai retirar uma aprendizagem no valor de **10%** de cada um deles;
- **Simple Regression Tree Learner (SRTL)**, onde usámos o **MSRP**, como a **target column** a prever.

Por fim, através do **Numeric Scorer**, obtivemos os resultados que aparecem na tabela da **Figura 2.15.**

	RFL (REGRESSION)	GBTL (REGRESSION)	SRT(REGRESSION)
R^2	0.905	0.992	0.993
MEAN ABS ERROR	3990.762	902.551	995.187
MEAN SQRT ERROR	38880768.459	2989920.5	2098685.875
ROOT MEAN SQRT ERROR	6235.445	1729.139	1448.684
MEAN SIGNED DIFF	1299.625	366.075	403.219
MEAN ABS % ERROR	0.127	0.025	0.029
ADJUSTED R^2	0.905	0.992	0.993

Figura 2.15: Resultados dos cálculos da regressão.

2.5 Avaliação

Com o estudo deste *dataset*, descobrimos que este se encontrava mais limpo e organizado do que aquilo que se esperava. Era um *dataset* em que praticamente todas as colunas, com a exceção da coluna **Model**, eram necessárias para se efetuar uma modelação bem sucedida e bem calculada.

Com os dados e resultados retirados da parte da modelação, verificamos que este *dataset* tem dois métodos de regressão, que foram muito bem sucedidos em diferentes aspetos da previsão de valores do **MSRP**:

- **Gradient Boosted Tree Learner:** este nodo teve melhores valores nos campos de **Mean Absolute Error, Mean Signed Difference e na Mean Absolute Percentage Error**, logo este método teve mais precisão nos valores de previsão dos dados;
- **Simple Regressiom Tree Learner:** este nodo teve melhores valores nos campos de R^2 **e Adjusted** R^2, logo este método teve mais sucesso no ajuste de modelo em relação aos dados observados.

Em resumo, o melhor método de regressão aplicado foi o Gradient Boosted Tree Learner, visto que foi o mais preciso a prever os dados do *dataset* **(MSRP)**.

2.6 Tratamento como um problema de Classificação

De forma a vermos este *dataset* de uma perspetiva diferente, resolvemos fazer um tratamento de dados que transformasse o problema atual num problema de Classificação. Nesse tratamento, determinamos intervalos com um Histograma, de forma a mantermos algum balanceamento entre cada tipo, nos quais em vez de ter o valor numérico de **MRSP** vai ter um de 5 diferentes tipos qualitativos: Muito barato, Barato, Normal, Caro, Muito Caro. Depois, procedemos à modelação para problemas de classificação.

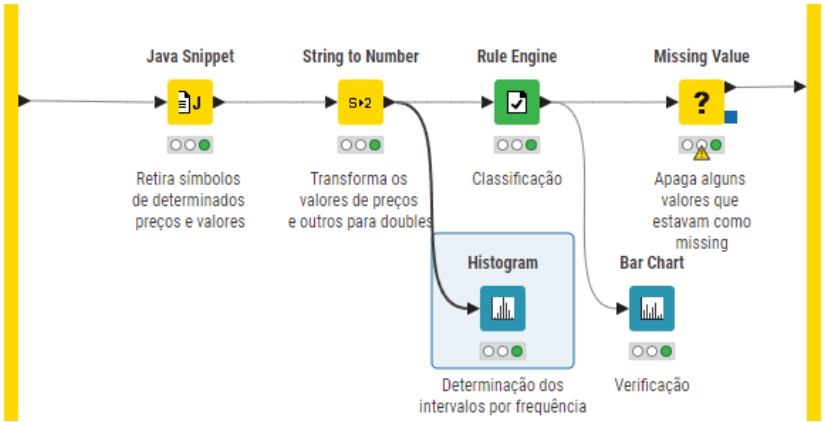

Figura 2.16: Tratamento dos dados

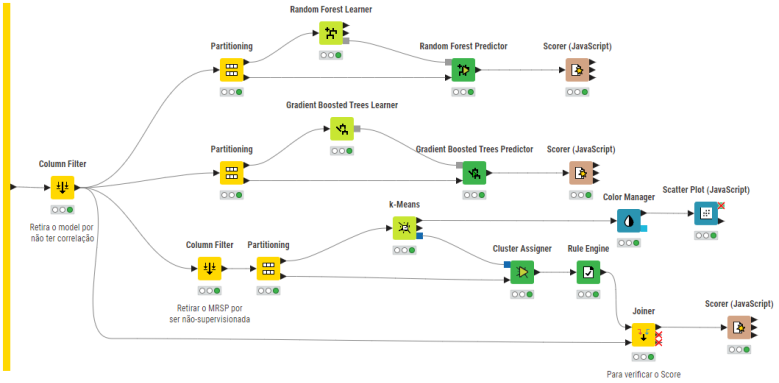

Figura 2.17: Modelação com 3 nodos de classificação

Tal como na regressão, removemos o *model* e depois fazemos **Partitioning** como para a Regressão, de forma **Random** e com uma valorização relativa de **70%**. De seguida, aplicamos 3 métodos de classificação e os seus nodos:

- **Random Forest Learner (RFL)**, onde usámos o **MRSP**, como a **target column** a prever, com um *split criterion* do tipo **Information Gain**, que resulta na **maior redução na entropia (ou aumento no ganho de informação)** em relação à **MRSP**;
- **Gradient Boosted Trees Learner (GBTL)**, onde usámos o **MRSP**, como a **target column** a prever nos diferentes 100 modelos que vão ser calculados e de onde e vai retirar uma aprendizagem no valor de **10%** de cada um deles, para prever os melhores resultados possíveis;
- **K-Means(K-M)**, onde temos de remover a tabela **MRSP** por ser modelo de aprendizagem não-supervisionado e selecionamos 5 *clusters* para os cinco tipos de **MRSP** existentes. Depois fazemos **Rule Engine e Joiner** para conseguirmos de algum modo ver quão bom correu o **K-Means**

Por fim, através do **Scorer**, obtivemos os resultados que aparecem na tabela **2.18**.

	RFL	GBTL	K-M
OVERALL ACCURACY	92.97%	96.09%	49.22%
OVERALL ERROR	7.03%	3.91%	50.78%
COHEN'S KAPPA (K)	0.912	0.951	0.373
CORRECTLY CLASSIFIED	119	123	63
INCORRECTLY CLASSIFIED	9	5	65

Figura 2.18: Resultados dos cálculos da classificação.

Conseguimos ver que o **GBTL** não dá valores tão precisos quanto a sua vertente na regressão, mas mesmo assim dá bons resultados e o **K-Means**, como previsto, por ser não-supervisionado, não dá os melhores valores, mas mesmo assim **49.22%** é muito respeitável. O **RFL** até deu melhores valores que na sua vertente de Regressão, o que é uma bocado surpreendente.

3 Tarefa: Dataset Atribuído

Esta tarefa consiste no uso de um *dataset* atribuído pelos professores, que o nosso grupo de trabalho analisou e recolheu dados sobre o mesmo, para o sucesso deste trabalho prático.

Este *dataset* contém valores laboratoriais de dadores de sangue e pacientes com hepatite C. Estes atributos fornecem informações valiosas sobre os fatores que influenciam a classificação de um paciente, como sendo dador de sangue, doente com Hepatite C, doente com **Fibrosis** e doente com **Cirrhosis**. Este *dataset* é considerado como um **Problema de Classificação.**

A metodologia que foi usada no processo da resolução do problema é o **CRISP-DM**, que é constituído pelas seguintes 6 etapas:

- **Estudo do negócio**
- **Estudo dos dados**
- **Preparação dos dados**
- **Modelação**
- **Avaliação**
- **Desenvolvimento**

NOTA: neste *dataset* iremos apenas usar as 5 primeiras etapas.

3.1 Estudo do negócio

Os objetivos a cumprir com este *dataset* e com o uso de técnicas de modelação existentes na aplicação **KNIME** são:

- Analisar/Avaliar a situação e os dados existentes no *dataset*;
- Tratar dos erros, valores inexistentes e outras irregularidades de dados, que possam existir no *dataset*;
- Se necessário, devo usar gráficos, matrizes e outro tipos de visualização de dados para retirar informações e analisar aspetos para melhorar o projeto.
- Descobrir a partir de modelos de aprendizagem, os atributos que classificam os respetivos **dadores de sangue com as corretas classificações.**

3.2 Estudo dos dados

O *dataset* para esta tarefa contém **615 linhas e 18 colunas.** Os atributos/colunas deste respetivo *dataset* são os seguintes:

- 1. **id:** o identificador de um respetivo dador de sangue;
- 2. **age:** a idade do dador de sangue;
- 3. **year_of_birth:** o ano do nascimento do dador de sangue;
- 4. **month_of_birth:** o mês do nascimento do dador de sangue;
- 5. **day_of_birth:** o dia do nascimento do dador de sangue;
- 6. **sex:** o género do dador de sangue;
- 7. **birth_location:** o local onde o dador de sangue nasceu;
- 8. **ALB:** o valor de **albumin** no sangue do dador **(g/L)**;
- 9. **ALP:** o valor de **alkaline phosphatase** no sangue do dador **(U/L)**;
- 10. **ALT:** o valor de **alanine transaminase** no sangue do dador **(U/L)**;
- 11. **AST:** o valor de **aspartate transaminase** no sangue do dador **(U/L)**;
- 12. **BIL:** o valor de **bilirubin** no sangue do dador **(μmol/L)**;
- 13. **CHE:** o valor de **acetylcholinesterase** no sangue do dador **(U/L)**;
- 14. **CHOL:** o valor de **cholesterol** no sangue do dador **(mmol/L)**;
- 15. **CREA:** o valor de **creatinine** no sangue do dador **(μmol/L)**;
- 16. **GGT:** o valor de **gamma-glutamyl transferase** no sangue do dador **(U/L)**;
- 17. **PROT:** o valor de **proteins** no sangue do dador **(g/L)**;
- 18. **Category:** o tipo de classificação que foi atribuída a cada dador de sangue, **('0=Blood Donor', '0s=suspect Blood Donor', '1=Hepatitis', '2=Fibrosis', '3=Cirrhosis').**

O atributo **"Category"** deste problema é aquele que iremos prever com o desenvolvimento de modelos de previsão de classificação e também vai ser utilizado para descobrir a qualidade dos outros atributos do *dataset*.

Fizemos o estudo dos dados depois de tratarmos dum pequeno erro que havia na coluna **Sex** que fazia existir "mm"em vez de "M"e substituímos todos os NA por **Missing Value** utilizando **String to Number** que troca as strings que são corretamente números para números e transforma os NA para **Missing Value**.

3.2.1 Category

Com a ajuda do **Pie Chart**, obtivemos esta figura com as percentagens correspondentes a cada tipo de dador de sangue **(Category)** existente no nosso *dataset*:

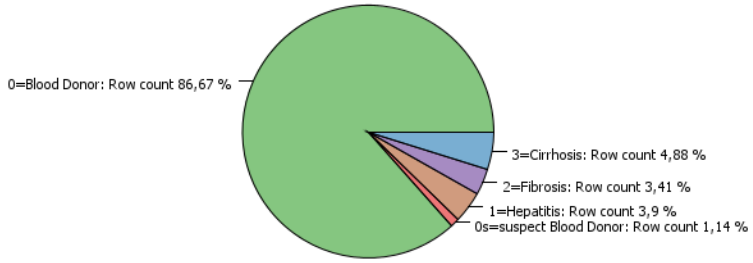

Figura 3.1: Tipos de dadores de sangue.

3.2.2 Age

Este atributo representa a idade de um dador presente no nosso *dataset*. Ao analisarmos a correlação, verificamos que a **Age** possui uma correlação com a **Category**, valorizada em 0.083, o que sugere uma relação positiva fraca, que poderá até não influenciar a previsão da **Category** de um dador. Como podemos ver na tabela em baixo, o valor mais alto de **Age** média vai para os casos suspeitos de dador, demonstrando que a idade não é um fator direto para se desenvolver uma doença grave:

CATEGORY	AGE MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	47.13
0s=SUSPECT BLOOD DONOR	57.57
1=HEPATITIS	38.71
2=FIBROSIS	52.33
3=CIRRHOSIS	53.47

Figura 3.2: Idade média para cada Category.

3.2.3 Sex

Este atributo representa a género de um dador de sangue presente no nosso *dataset*. Ao analisarmos a correlação, verificamos que o **Sex** possui uma correlação com a **Category**, valorizada em 0.0754, o que sugere uma relação positiva fraca, que poderá até não influenciar a previsão da **Category** de um dador. Na tabela em baixo, temos uma contagem de dadores de cada **Sex**, para cada tipo de **Category**:

CATEGORY	Nº DE DADORES DO SEX (M)	Nº DE DADORES DO SEX (F)
0=BLOOD DONOR	318	215
0s=SUSPECT BLOOD DONOR	6	1
1=HEPATITIS	20	4
2=FIBROSIS	13	8
3=CIRRHOSIS	20	10

Figura 3.3: Contagem de géneros presentes nos diferentes tipos de Category.

3.2.4 ALB

Este atributo representa o valor de **albumin** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **ALB** no sangue, para cada tipo de **Category**:

CATEGORY	ALB MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	42.24 g/L
0s=SUSPECT BLOOD DONOR	24.40 g/L
1=HEPATITIS	43.83 g/L
2=FIBROSIS	41.76 g/L
3=CIRRHOSIS	32.48 g/L

Figura 3.4: Quantidade média de ALB no sangue para cada Category.

3.2.5 ALP

Este atributo representa o valor de **alkaline phosphatase** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **ALP** no sangue, para cada tipo de **Category**:

CATEGORY	ALP MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	68.37 U/L
0s=SUSPECT BLOOD DONOR	107.30 U/L
1=HEPATITIS	42.11 U/L
2=FIBROSIS	37.84 U/L
3=CIRRHOSIS	93.22 U/L

Figura 3.5: Quantidade média de ALP no sangue para cada Category.

3.2.6 ALT

Este atributo representa o valor de **alanine transaminase** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **ALT** no sangue, para cada tipo de **Category**:

CATEGORY	ALT MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	26.63 U/L
0s=SUSPECT BLOOD DONOR	102.11 U/L
1=HEPATITIS	26.90 U/L
2=FIBROSIS	59.60 U/L
3=CIRRHOSIS	22.97 U/L

Figura 3.6: Quantidade média de ALT no sangue para cada Category.

3.2.7 AST

Este atributo representa o valor de **aspartate transaminase** presente no sangue de um dador do nosso *dataset*. Ao analisarmos a correlação, verificamos que o **AST** possui uma correlação com a **Category**, valorizada em 0.4484, o que sugere uma relação positiva forte, que poderá influenciar a previsão da **Category** de um dador. Na tabela em baixo, temos uma média de valores de **AST** no sangue, para cada tipo de **Category**:

CATEGORY	AST MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	26.55 U/L
0s=SUSPECT BLOOD DONOR	71.00 U/L
1=HEPATITIS	75.73 U/L
2=FIBROSIS	81.17 U/L
3=CIRRHOSIS	107.46 U/L

Figura 3.7: Quantidade média de AST no sangue para cada Category.

3.2.8 BIL

Este atributo representa o valor de **bilirubin** presente no sangue de um dador do nosso *dataset*. Ao analisarmos a correlação, verificamos que o **BIL** possui uma correlação com a **Category**, valorizada em 0.2851, o que sugere uma relação positiva, que poderá até influenciar a previsão da **Category** de um dador. Na tabela em baixo, temos uma média de valores de **BIL** no sangue, para cada tipo de **Category**:

CATEGORY	BIL MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	8.53 umol/L
0s=SUSPECT BLOOD DONOR	4.69 umol/L
1=HEPATITIS	15.63 umol/L
2=FIBROSIS	13.43 umol/L
3=CIRRHOSIS	59.13 umol/L

Figura 3.8: Quantidade média de BIL no sangue para cada Category.

3.2.9 CHE

Este atributo representa o valor de **acetylcholinesterase** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **CHE** no sangue, para cada tipo de **Category**:

CATEGORY	CHE MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	8.40 U/L
0s=SUSPECT BLOOD DONOR	7.48 U/L
1=HEPATITIS	9.28 U/L
2=FIBROSIS	8.33 U/L
3=CIRRHOSIS	3.82 U/L

Figura 3.9: Quantidade média de CHE no sangue para cada Category.

3.2.10 CHOL

Este atributo representa o valor de **cholesterol** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **CHOL** no sangue, para cada tipo de **Category**:

CATEGORY	CHOL MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	5.49 mmol/L
0s=SUSPECT BLOOD DONOR	4.45 mmol/L
1=HEPATITIS	5.10 mmol/L
2=FIBROSIS	4.60 mmol/L
3=CIRRHOSIS	4.01 mmol/L

Figura 3.10: Quantidade média de CHOL no sangue para cada Category.

3.2.11 CREA

Este atributo representa o valor de **creatinine** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **CREA** no sangue, para cada tipo de **Category**:

CATEGORY	CREA MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	78.98 umol/L
0s=SUSPECT BLOOD DONOR	61.71 umol/L
1=HEPATITIS	73.96 umol/L
2=FIBROSIS	73.49 umol/L
3=CIRRHOSIS	138.22 umol/L

Figura 3.11: Quantidade média de CREA no sangue para cada Category.

3.2.12 GGT

Este atributo representa o valor de **gamma-glutamyl transferase** presente no sangue de um dador do nosso *dataset*. Ao analisarmos a correlação, verificamos que o **GGT** possui uma correlação com a **Category**, valorizada em 0.367, o que sugere uma relação positiva, que poderá influenciar a previsão da **Category** de um dador. Na tabela em baixo, temos uma média de valores de **GGT** no sangue, para cada tipo de **Category**:

CATEGORY	GGT MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	29.04 U/L
0s=SUSPECT BLOOD DONOR	151.51 U/L
1=HEPATITIS	92.58 U/L
2=FIBROSIS	79.55 U/L
3=CIRRHOSIS	129.44 U/L

Figura 3.12: Quantidade média de GGT no sangue para cada Category.

3.2.13 PROT

Este atributo representa o valor de **proteins** presente no sangue de um dador do nosso *dataset*. Na tabela em baixo, temos uma média de valores de **PROT** no sangue, para cada tipo de **Category**:

CATEGORY	PROT MÉDIA PARA CADA CATEGORY
0=BLOOD DONOR	72.11 g/L
0s=SUSPECT BLOOD DONOR	53.91 g/L
1=HEPATITIS	74.70 g/L
2=FIBROSIS	76.10 g/L
3=CIRRHOSIS	70.05 g/L

Figura 3.13: Quantidade média de PROT no sangue para cada Category.

3.3 Preparação dos dados

Inicialmente para este *dataset* encontramos um erro na coluna **Sex**, relativamente às designações de género, onde encontramos **"mm"**, como designação para o sexo masculino, logo para resolver esse problema aplicamos o **Rule Engine** e modificámos essa designação para **"M"** para resolver a situação.

Em seguida, havia linhas no *dataset* sobre certas enzimas que tinham **Missing Values**, mas com uma designação de **NA**. Utilizando o String to Number e desativando o **Fail on error**, todos os valores corretos alteram para Double e todos os **NA** alteram para **Missing Values** para depois podermos fazer o tratamento deles a partir do nodo **Missing Value**.

Nós fizemos 3 tipos de tratamentos dos dados diferentes após estes dois nodos e em cada um desses 3 tratamentos, 3 diferentes tipos de tratamento para os **Missing Values**. O caso mais básico é não haver qualquer tratamento de **Outliers**, removendo o nodo de **Numeric Outliers** de todo e depois os outros 3 tratamentos para os **Missing Values**, que vão ser iguais para todos, são: remover linhas com *Missing Values*, tornar o *Missing Value* na média de todos os valores dessa coluna ou torná-lo na mediana de todos os valores dessa coluna. Depois temos os outros dois tratamentos de dados em que tratamos dos **Outliers**. Ou mudamos esse **Outliers para Missing Value** ou para o **Closest Permitted Value**, mantendo assim ainda um valor elevado ou baixo, mas não tendo de utilizar o tratamento dos *Missing Values* para esses valores.

Já na parte do metanodo de modelação, foi aplicado um **Column Filter**, que filtrou todas as colunas que não apresentavam grande importância para uma previsão bem sucedida para o atributo objetivo **(Category)** deste *dataset*, sendo elas o ID, pois apesar de ter correlação positiva alta, não faz sentido utilizar ID para este objetivo e este só tem correlação devida à forma que o *dataset* está formulado, o **year_of_birth, month_of_birth e day_of_birth** pois têm correlação de 1 com a **Age**, mostrando que basta utilizar a Age dessas 4 colunas e, por fim, a **birth_location**, porque como todos têm o mesmo valor ou *Missing Value*, resolvemos remover, pois consideramos que não é algo necessário para este objetivo em concreto.

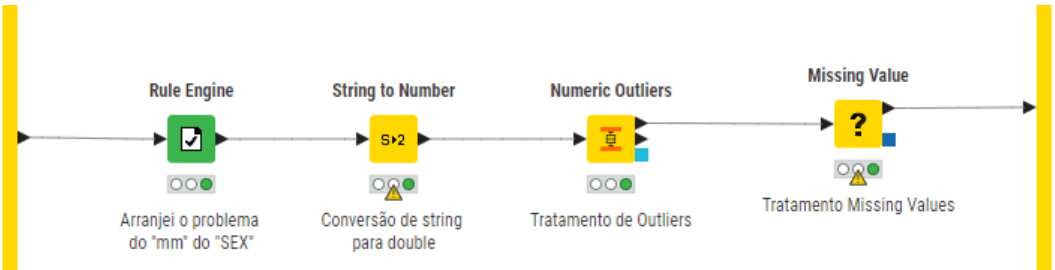

Figura 3.14: Preparação dos dados no KNIME para Classificação.

3.4 Modelação

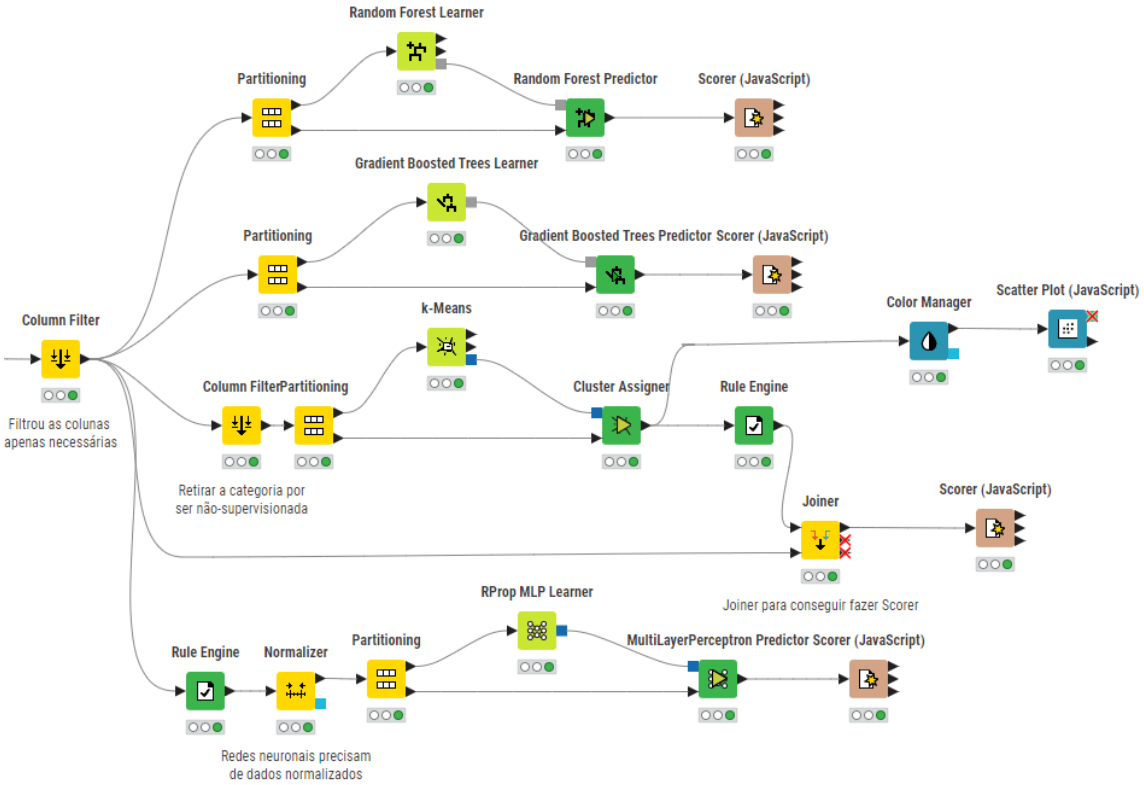

Figura 3.15: Modelação dos dados no KNIME para Classificação.

Como se pode ver na imagem em cima, usámos o nodo **Partitioning**, com a divisão de partição feita de forma **Stratified Sampling**, exceto no **K-Means** que fazemos **Draw Randomly** por removermos a **Category** para ser não-supervisionada, para pudermos garantir a mesma proporção de tipos de dadores, e com uma valorização relativa igual a **70%.** Em seguida, foram aplicados os métodos de classificação seguintes e os seus nodos:

- **Random Forest Learner (RFL)**, onde usámos a **Category**, como a **target column** a prever, com um *split criterion* do tipo **Information Gain**, que resulta na **maior redução na entropia (ou aumento no ganho de informação)** em relação à **Category**;
- **Gradient Boosted Trees Learner (GBTL)**, onde usámos a **Category**, como a **target column** a prever nos diferentes 100 modelos que vão ser calculados e de onde e vai retirar uma aprendizagem no valor de **10%** de cada um deles, para prever os melhores resultados possíveis;
- **K-Means (K-M)**, onde removemos a **Category**, porque é um algoritmo não-supervisionado, utilizamos **5 clusters**, definidos nas definições do nodo, onde cada *cluster* representa um tipo de **Category** de dador de sangue, depois definidos no **Rule Engine** para podermos depois ver no **Scorer**, depois de utilizar o **Joiner** para colocar a coluna da **Category** de novo.
- **RProp MPL Learner (RPROP)**, onde usámos a **Category**, como a **target column**, com 100 iterações no máximo, 1 camada escondida e 10 neurónios contidos por camada.

Como acabamos por ter no final, 9 tratamentos diferentes(3 de missing values dentro dos 3 em tratamento de outliers), vamos só colocar aqui as 3 tabelas dos Scorers dos melhores dos 3 tratamentos diferentes de **Outliers**, em que um certo tratamento de Missing Value teve o melhor resultado.

	RFL	GBTL	K-M	RPROP
OVERALL ACCURACY	95.48%	94.92%	40.68%	96.05%
OVERALL ERROR	4.52%	5.08%	59.32%	3.95%
COHEN'S KAPPA (K)	0.735	0.718	0.003	0.797
CORRECTLY CLASSIFIED	169	168	72	170
INCORRECTLY CLASSIFIED	8	9	105	7

Figura 3.16: Tabela dos Scorer da Classificação, sem tratamento de outliers, removendo linhas com Missing Values

	RFL	GBTL	K-M	RPROP
OVERALL ACCURACY	97.74%	96.99%	24.06%	98.5%
OVERALL ERROR	2.26%	3.01%	75.94%	1.5%
COHEN'S KAPPA (K)	0.000	0.489	0.009	0.594
CORRECTLY CLASSIFIED	130	129	32	131
INCORRECTLY CLASSIFIED	3	4	101	2

Figura 3.17: Tabela dos Scorer da Classificação, tratamento outliers para Missing Value, removendo linhas com Missing Values

	RFL	GBTL	K-M	RPROP
OVERALL ACCURACY	95.68%	94.05%	37.84%	94.59%
OVERALL ERROR	4.32%	5.95%	62.16%	5.41%
COHEN'S KAPPA (K)	0.821	0.745	0.108	0.763
CORRECTLY CLASSIFIED	177	174	70	175
INCORRECTLY CLASSIFIED	8	11	115	10

Figura 3.18: Tabela dos Scorer da Classificação, tratamento outliers para Closest Permitted Value, Missing Values para mediana

Todas as tabelas formuladas vão num Word chamado **Scores** que vem anexado junto com o relatório para os professores puderem ver de forma detalhada todas as tabelas que resultaram dos **Scorers**.

3.5 Avaliação

Com o estudo deste *dataset*, descobrimos que este continha alguns erros e falhas de dados, como os **NA**, que mostraram ser inesperados e demorados a encontrar e resolver. Era um *dataset* que continha várias colunas desnecessárias, para se efetuar uma modelação bem sucedida e bem classificada.

Em termos dos **Scorers**, a primeira óbvia observação é que o **K-Means** não tem os melhores resultados devido à sua natureza de não-supervisionada. Aliás, durante a execução do trabalho, durante várias iterações a sua precisão poderia ir de menos de 2% para chegar a 80% de forma completa aleatória. Tentamos meter uns resultados minimamente decentes, que dê para ver de alguma forma que sem tratamento de **Outliers** este funciona melhor e se trocando os **Outliers por Missing Values**, resulta numa descida da precisão deste. Mas convêm realçar novamente que este é o mais instável dos 4 algoritmos utilizados.

Em termos globais, o que teve melhores resultados foi o **RProp MPL Learner**, apesar de em algumas situações o **Random Forest Learner** ter conseguido melhores resultados em termos de prever a **Category** dos dadores. Em termos de **Cohen's Kappa**, tivemos uma situação inesperada que o **RFL** deu 0.000 e descobrimos que isso aconteceu devido ao facto deste ter acertado todos os **Blood Donor**, mas não ter previsto de forma correta qualquer outra das **Category** existentes.

Por fim, em termos de tratamento global do que deu os melhores resultados, conseguimos ver que foi o de transformar os **Outliers em Missing Value** e remoção de todos os *Missing Value*s, mas isso aconteceu devido ao facto que, como removemos todos os *Missing Values*, sendo alguns provenientes dos *Outliers*, como em termos médicos, *Outliers* costumam ser os doentes, aconteceu que se deve ter removido todos os que tinham doenças na **Category** e este previu quase todos como **Blood Donor**, o que fez com que, apesar de ter previsto de melhor forma, tenha tido um *Cohen's Kappa* mais baixo em todos os casos, tal como já mencionamos. Por isso em termos globais de Precisão e *Cohen's Kappa* o tratamento que realmente esteve melhor foi o de não tratamento de *Outliers* e remoção de *Missing Values*, apesar de tratamento de *Outliers* para *Closest Permitted Value*, colocando os *Missing Values* existentes em Mediana, não esteve muito longe.

Em termos de tratamento mais específico de *Outliers*, o que deu melhores resultados globais, foi o que tratava os **Outliers transformando-os em Closest Permitted Value** e nós achamos que isso aconteceu devido ao facto que, em termos médicos, certos valores médicos fazem sentido ser *Outliers*, mas há outros que podem prejudicar a Precisão caso não façam tanto sentido, logo colocá-los num valor ainda que se diferencie do resto, mas que de forma a que não prejudique totalmente a previsão, melhorou de certa forma como os algoritmos interagiam com estes valores.

3.6 Tratamento como um problema de Regressão

De forma a vermos este *dataset* de uma perspetiva diferente, resolvemos fazer um tratamento de dados que transformasse o problema atual num **problema de Regressão**. Nesse tratamento, transformamos as diferentes categorias em números utilizando **Rule Engine**: "0=Blood Donor" passou a ser **0**, "1=Hepatitis" passou a ser **1**, "2=Fibrosis" passou a ser **2**, "3=Cirrhosis" passou a ser **3** e "0s=suspect Blood Donor" passou a ser **4**. Além disso continuamos a ter todos os **tratamentos de Outliers e Missing Values** que tínhamos para o problema de Classificação.

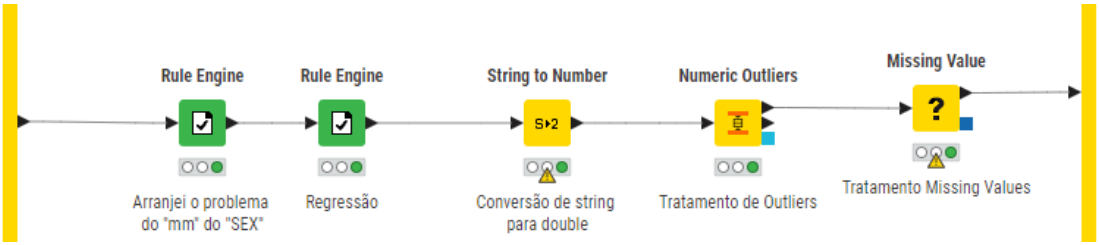

Figura 3.19: Preparação dos dados no KNIME para Regressão.

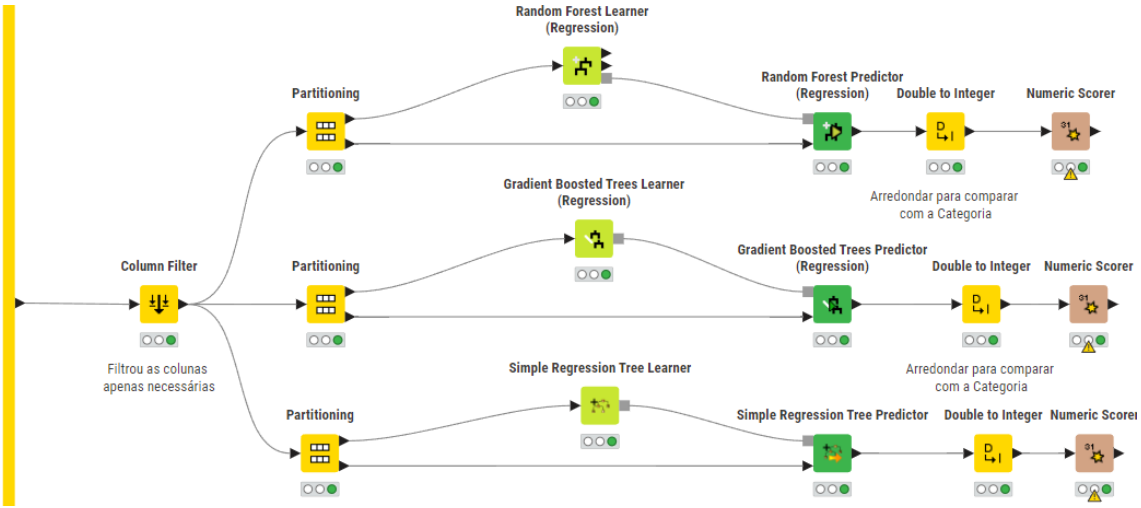

Figura 3.20: Modelação dos dados no KNIME para Regressão.

Tal como na Classificação, começamos por filtrar colunas que não serviam para o algoritmo (exemplo:ID) e começamos por fazer o **Partitioning** desta vez usando **Draw Randomly** e com valorização de 70%, pois como a **Category** agora é numérica já não dá para fazer **Stratified Sampling**, algo que pode vir a prejudicar os resultados.

Utilizamos os mesmo algoritmos que usamos na **Tarefa do grupo** em termos de Regressão, usando como **target column** a **Category**, mas mantendo os restos dos parâmetros iguais.

Por fim, em termos de modelação, tivemos de utilizar o nodo **Double to Integer**, pois a regressão dá **doubles**, não inteiros, logo se fossemos fazer Scorer com esses valores, ia dar valores horríveis. Para isso, arredondamos os valores para depois podermos ver quão boa a

regressão foi para este problema utilizando o **Scorer** para podermos comparar ao problema em Classificação.

Tal como no problema de Classificação, devido à existência de tantos tratamentos, vamos somente colocar as tabelas dos melhores **Scorers** de cada Outlier de um certo tratamento de Missing Value:

	RFL (REGRESSION)	GBTL (REGRESSION)	SRT(REGRESSION)
OVERALL ACCURACY	94.35%	95.48%	95.48%
OVERALL ERROR	5.65%	4.52%	4.52%
COHEN'S KAPPA (K)	0.648	0.708	0.778
CORRECTLY CLASSIFIED	167	169	169
INCORRECTLY CLASSIFIED	10	8	8

Figura 3.21: Tabela dos Scorer da Regressão, sem tratamento de outliers, removendo linhas com Missing Values

	RFL (REGRESSION)	GBTL (REGRESSION)	SRT(REGRESSION)
OVERALL ACCURACY	97.44%	96.241%	99.248%
OVERALL ERROR	2.256%	3.759%	0.752%
COHEN'S KAPPA (K)	0.393	0	0.853
CORRECTLY CLASSIFIED	130	128	132
INCORRECTLY CLASSIFIED	3	5	1

Figura 3.22: Tabela dos Scorer da Classificação, tratamento outliers para Missing Value, removendo linhas com Missing Values

	RFL (REGRESSION)	GBTL (REGRESSION)	SRT(REGRESSION)
OVERALL ACCURACY	90.96%	94.35%	90.96%
OVERALL ERROR	9.04%	5.65%	9.04%
COHEN'S KAPPA (K)	0.655	0.763	0.578
CORRECTLY CLASSIFIED	161	167	161
INCORRECTLY CLASSIFIED	16	10	16

Figura 3.23: Tabela dos Scorer da Classificação, tratamento outliers para Closest Permitted Value, removendo linhas com Missing Values

Em termos de Avaliação, de todos os tratamentos globais, o melhor foi ao tratar dos *Outliers* transformá-los em *Missing Values* e depois remover todas as linhas com *Missing Values*, mas isso ocorreu devido à existência de raros casos de doenças, pois removemos quase todos ao retirar os *Outliers*, fazendo com que na Regressão quase todos os valores se aproximassem de 0 e quase todos os casos eram realmente de **Blood Donor** logo estavam quase todos corretos. Por isso é que estes, exceto o **SRT**, têm *Cohen's Kappa* baixíssimos, chegando até a ser 0 no **GBTL** pois somente acertou os **Blood Donor**.

Em comparação com os melhores resultados de cada tratamento global com os do problema de Classificação, conseguimos reparar numa semelhança de resultados na vertente do **RFL** e

GBTL de problemas de regressão, exceto quando se trata de tornar **Outliers para Closest Permitted Value** onde se repara em melhores resultados no problema de Classificação em que o melhor resultado se encontra em transformar os *Missing Values* em mediana e não em remover linhas como se fez na regressão, no melhor resultado.

Tal como no caso das tabelas da Classificação, todas as tabelas formuladas neste problemas também estão no mesmo Word de forma que os professores consigam verificá-las todas com mais detalhe.

4 Conclusão

Com a resolução deste trabalho prático, achamos que conseguimos com sucesso fazer a conceção de variados modelos de aprendizagem e decisão que foram abordados durante as aulas práticas e teóricas do semestre.

Tendo em conta os *datasets* utilizados neste projeto, além da conceção dos modelos, também fizemos estudos dos dados e preparação dos mesmos para conseguir de alguma forma percebermos com que tipo de problema nos deparávamos, fazendo com que soubesse que tipo de ferramentas tínhamos de utilizar concretamente para cada *dataset* em si.

Como o *dataset* atribuído tinha mais parecenças com um problema de Classificação, tivemos de pesquisar por um *dataset* que fosse mais caracterizado com um problema de Regressão, de forma a trabalharmos este dois tipos de problemas. Depois de muita procura e experimentação com variados *datasets*, consideramos que fizemos um bom trabalho na conceção dos modelos para o *dataset* que escolhemos. Apesar de sabermos o tipo de problema de cada *dataset*, também os tratamos com o tipo oposto para vermos de alguma forma como estes se comportavam com ferramentas que imaginávamos que não fossem tão indicadas para eles.

Por fim, consideramos que temos um trabalho bem conseguido para cada um dos *datasets* que exploramos e concebemos modelos de aprendizagem e decisão de forma a ver como estes funcionavam/previam.