



Universidade do Minho

Licenciatura em Engenharia Informática

Ano Letivo 2022/2023

## **Aprendizagem e Decisão Inteligente**

### **Grupo 12**

Inês Ferreira (A97372)

Joana Branco (A96584)

João Braga (A97368)

Robert Szabo (A91682)

13 de maio de 2023

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Doenças Cardiovasculares</b>	<b>1</b>
2.1	Características do <i>dataset</i> . . . . .	1
2.2	Análise das <i>features</i> . . . . .	2
2.3	<i>KNIME workflow</i> . . . . .	7
2.3.1	Estudo dos dados . . . . .	8
2.3.2	Tratamento de dados . . . . .	11
2.3.3	Análise dos dados pós-tratamento . . . . .	12
2.3.4	Modelos de <i>Machine Learning</i> . . . . .	14
2.3.5	Análise de resultados . . . . .	17
<b>3</b>	<b>Obesidade</b>	<b>17</b>
3.1	Características do <i>dataset</i> . . . . .	17
3.2	<i>KNIME workflow</i> . . . . .	19
3.2.1	Estudo dos dados . . . . .	19
3.2.2	Tratamento de dados . . . . .	20
3.2.3	Análise dos dados pós-tratamento . . . . .	22
3.2.4	Modelos de <i>Machine Learning</i> . . . . .	22
3.2.5	Análise de resultados . . . . .	27
<b>4</b>	<b>Conclusão</b>	<b>27</b>
<b>5</b>	<b>Referências</b>	<b>27</b>

# Lista de Figuras

1	Idades representados por um <i>Pie Chart</i> . . . . .	3
2	Altura exibida por um <i>Pie Chart</i> . . . . .	3
3	Pesos representados por um <i>Pie Chart</i> . . . . .	4
4	Valores do Colesterol representados por um <i>Pie Chart</i> . . . . .	5
5	Níveis de Glicose representados por um <i>Pie Chart</i> . . . . .	5
6	Fumadores mostrados por um <i>Pie Chart</i> . . . . .	6
7	Consumo de álcool representados por um <i>Pie Chart</i> . . . . .	6
8	Indivíduos ativos fisicamente apresentados por um <i>Pie Chart</i> . . . . .	7
9	Doentes cardíacos exibidos por um <i>Pie Chart</i> . . . . .	7
10	Workflow completo sobre Doenças Cardiovasculares no KNIME . . . . .	8
11	Estudo dos dados (Statistics) no KNIME . . . . .	8
12	Detalhes do nodo Data Explorer . . . . .	9
13	Detalhes do nodo Scatter Plot . . . . .	9
14	Detalhes do nodo Scatter Plot . . . . .	10
15	Matriz de Correlação do nodo Linear Correlation . . . . .	10
16	Matriz de Correlação do nodo Rank Correlation . . . . .	11
17	Preparação dos dados no KNIME . . . . .	11
18	Detalhes da <i>height</i> antes da preparação de dados . . . . .	12
19	Dados após a preparação dos mesmos . . . . .	12
20	Detalhes do nodo Bar Chart . . . . .	13
21	Detalhes do nodo Bar Chart . . . . .	13
22	Detalhes do nodo Line Plot . . . . .	14
23	Detalhes do nodo Bar Chart . . . . .	14
24	Modelos de classificação no KNIME . . . . .	15
25	Modelos de Regressão no KNIME . . . . .	16
26	Workflow para estudo sobre obesidade KNIME . . . . .	19
27	Uso do Data Explorer para encontrar dados anómalos . . . . .	19
28	Metanodo para o pré-processamento dos dados . . . . .	20
29	Uso do Rank Correlation para verificar correlação das features . . . . .	21
30	Uso do Scatter Plot para analisar o tipo de obesidade em relação ao peso e altura . . . . .	22
31	Metanodo relativo aos modelos de classificação . . . . .	23
32	Cálculo do IMC e criação da variável IMC-Type a partir do IMC . . . . .	25
33	Scorer - NObeyesdad vs IMC-Type . . . . .	25
34	Metanodo relativo aos modelos de regressão . . . . .	26

# 1 Introdução

Neste trabalho prático foram considerados dois *datasets*: o primeiro, selecionado pelo grupo de trabalho, contém informação sobre doenças cardiovasculares; o segundo, atribuído de acordo com o número do grupo (número par), contém informação acerca da obesidade. Para a construção dos modelos e análise dos mesmos damos uso à ferramenta *KNIME*.

Utilizando os modelos de aprendizagem aprendidos, construíram-se modelos de *Machine Learning* com o objetivo de, respetivamente, prever doença cardiovascular através dos seus diferentes hábitos e características (o que constitui um problema de classificação) e o nível de obesidade (também constituindo um problema de classificação).

No presente relatório serão especificadas as diferentes etapas do processo de conceção de modelos de aprendizagem de ambos os *dataset*, como o estudo, tratamento e respetiva modelação.

## 2 Doenças Cardiovasculares

Para a realização da primeira tarefa houve a necessidade de procura de um *dataset* onde fôssemos capazes de explorar os dados. Tendo em conta que há cada vez mais doenças cardiovasculares acreditamos que a escolha do *dataset* pode ser benéfico para conhecimento geral e, com isto, temos o objetivo de saber qual a probabilidade de uma pessoa ter a doença e o quanto as suas rotinas e fisionomia influenciam o aparecimento da mesma.

### 2.1 Características do dataset

Este *dataset* é constituído por 70000 linhas e são enunciados de seguida as *features* que o constituem e uma breve descrição dos mesmos:

**Age** Idade - representado por int (dias);

**Height** Altura - representado por int (cm);

**Weight** Peso - representado por float (kg);

**Gender** Género - representado por int (1: género feminino, 2: género masculino);

**Systolic blood pressure - ap\_hi** Pressão arterial sistólica - representado por int;

**Diastolic blood pressure - ap\_lo** Pressão arterial diastólica - representado por int;

**Cholesterol** Colesterol - representado por int (1:normal, 2: acima do normal, 3: muito acima do normal);

**Glucose** Glicose - representado por int (1:normal, 2: acima do normal, 3: muito acima do normal);

**Smoking** Fumar - representado por boolean (0 ou 1);

**Alcohol intake - alco** Ingestão de álcool - representado por boolean (0 ou 1);

**Physical activity - active** Atividade física - representado por boolean (0 ou 1);

**Presence or absence of cardiovascular - cardio** Presença de doença cardíaca - representado por boolean (0 ou 1).

Também existe uma coluna *id* que é apenas um identificador de todas as linhas presentes mas, não sentimos necessidade de a considerar para o nosso estudo nem apresentar a sua especificação.

## 2.2 Análise das features

É possível relacionar as diferentes colunas entre si considerando, por exemplo, o nível de colesterol de uma pessoa qual a probabilidade de ter a doença cardiovascular.

De seguida vamos avançar para uma análise mais detalhada de cada uma das *features* presentes no nosso *dataset* com diversas ilustrações, lembrando que não consideramos a coluna *id* para a nossa análise.

Podemos também classificar este problema como um problema **classificação** pois o nosso objetivo é prever se uma pessoa tem doença cardiovascular ou não tendo em conta todas as suas características.

### Age

As idades do nosso *dataset* estão compreendidas entre os 10798 dias (30 anos) e os 23713 dias (65 anos). Foi possível saber esta informação através do nodo **Data Explorer** onde fomos capazes de consultar o valor mínimo e máximo para a coluna da idade. De modo a ser mais esclarecedor decidimos transformar a coluna da idade de dias para anos com a ajuda do nodo **Math Formula**. O *dataset* apresenta a seguinte distribuição para a coluna de idade.

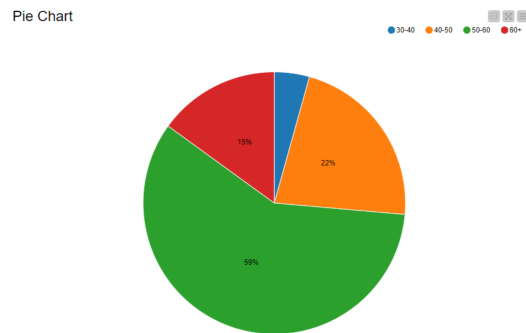


Figura 1: Idades representados por um *Pie Chart*

É possível reparar, segundo a imagem, que mais de 70% tem acima de 50 anos.

## Height

Esta feature especifica que a grande maioria das pessoas tem entre 150cm a 180cm.

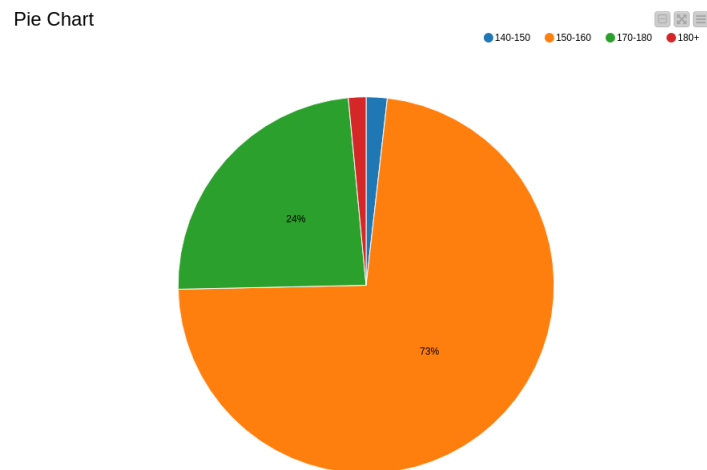


Figura 2: Altura exibida por um *Pie Chart*

## Weight

A partir do gráfico seguinte, compreende-se que existem mais dados relativos a pessoas entre os 60kg a 100kg.

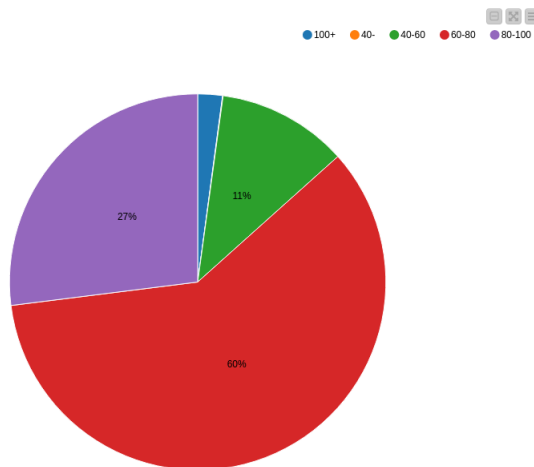
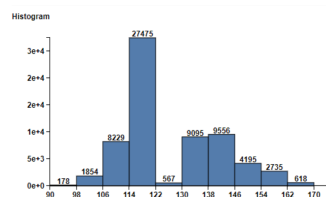


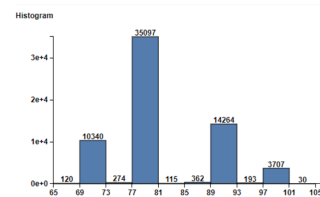
Figura 3: Pesos representados por um *Pie Chart*

### Ap\_hi-Systolic blood pressure e Ap\_lo-Diastolic blood pressure

A distribuição dos dados acerca da pressão arterial sistólica e diastólica pode ser evidenciada pelos seguintes histogramas, respetivamente.



(a) Valores de *ap\_hi* representados por um histograma



(b) Valores de *ap\_lo* representados por um histograma

### Cholesterol

Com base na visualização gráfica apresentada abaixo, pode-se inferir que a maior parte dos participantes examinados apresenta níveis de colesterol dentro da faixa considerada normal.

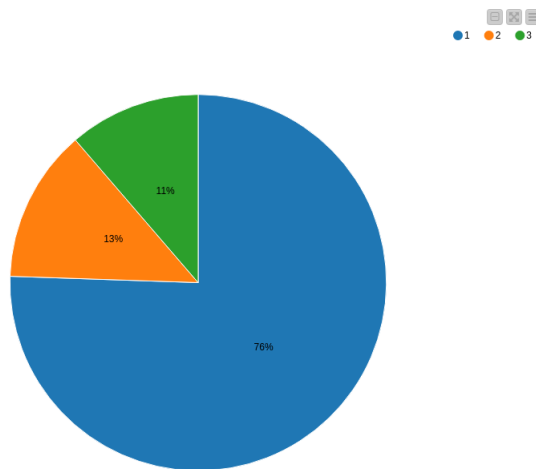


Figura 4: Valores do Colesterol representados por um *Pie Chart*

## Gluc

Tendo em conta a imagem seguinte, é possível constatar que a maioria dos indivíduos analisados possui níveis de glicose no sangue considerados normais .

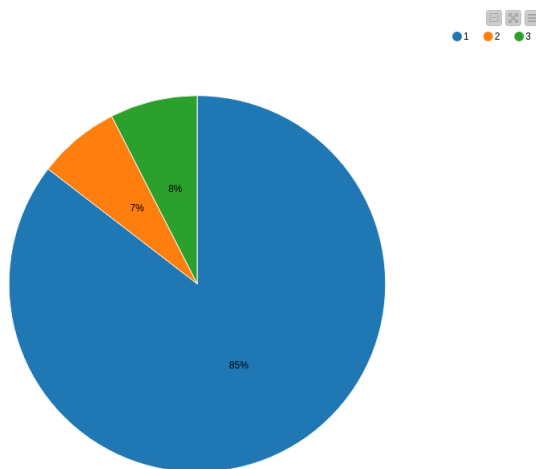


Figura 5: Níveis de Glicose representados por um *Pie Chart*

## Smoke

A partir do gráfico seguinte, compreende-se que existem mais dados relativos a pessoas não fumantes.



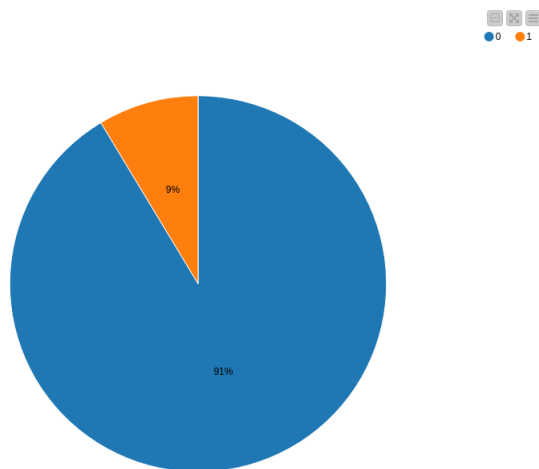


Figura 6: Fumadores mostrados por um *Pie Chart*

## Alcohol

Pode-se inferir a partir do gráfico que há uma maior quantidade de informações disponíveis sobre indivíduos que não fazem uso de bebidas alcoólicas.

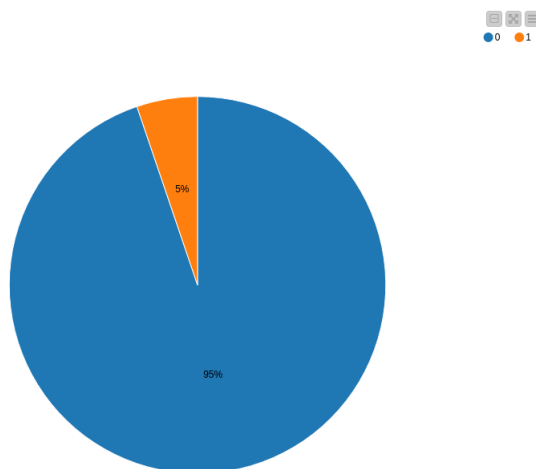


Figura 7: Consumo de alchool representados por um *Pie Chart*

## Active

Analisando o gráfico, pode-se concluir que há uma predominância de dados referentes a pessoas que fazem exercício regularmente em relação aquelas que não o fazem.

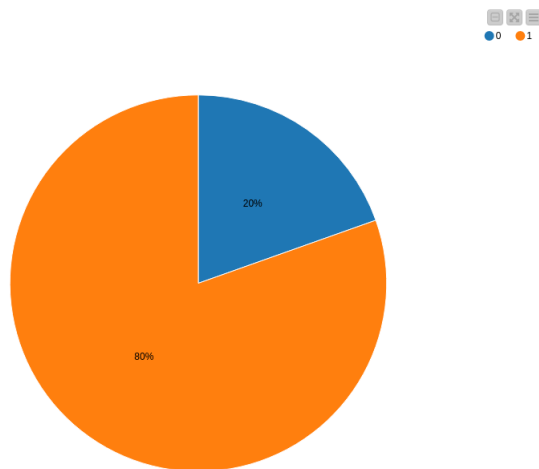


Figura 8: Indivíduos ativos fisicamente apresentados por um *Pie Chart*

## Cardio

Relativamente aos indivíduos que apresentam doença cardíaca, podemos notar uma distribuição relativamente uniforme.

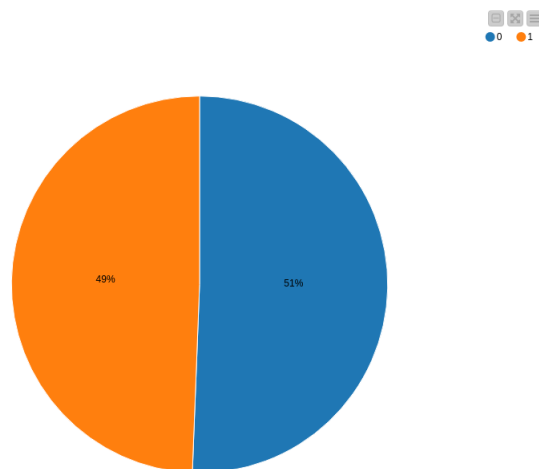


Figura 9: Doentes cardíacos exibidos por um *Pie Chart*

## 2.3 KNIME workflow

O *Workflow* do respetivo *dataset* foi subdivido em diversas fases que consistem em estudo, preparação e tratamento dos dados prosseguindo depois para a sua modelação.

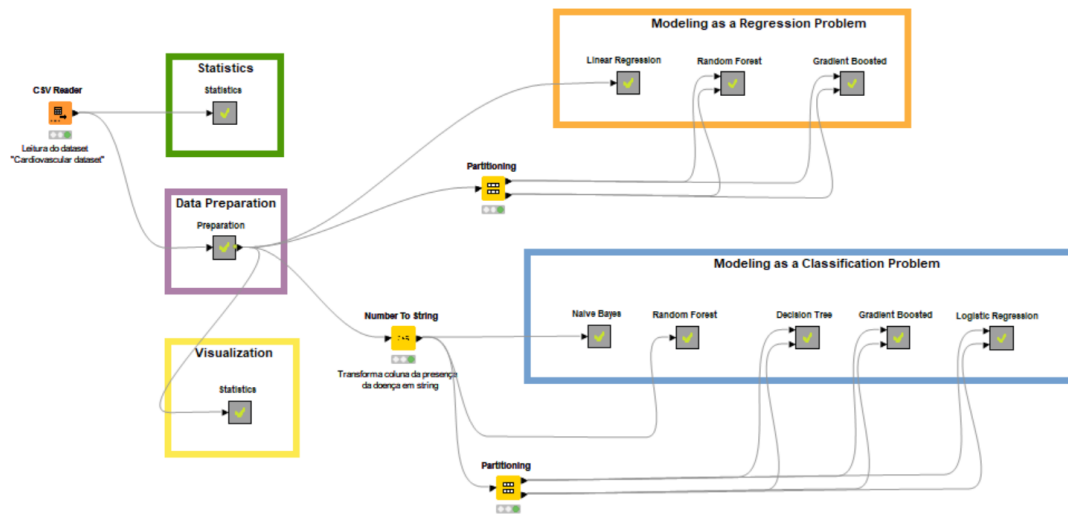


Figura 10: Workflow completo sobre Doenças Cardiovasculares no KNIME

### 2.3.1 Estudo dos dados

Nesta secção apresentamos diversas formas de estudo dos dados, analisando principalmente as suas estatísticas. Fizemos uso de nodos como **Data Explorer**, de modo a analisarmos as diferentes colunas do *dataset*, visualizar valores mínimos e máximos de cada uma delas e perceber se há a possibilidade de *outliers* ou não, ver se há presença ou não de *missing values*, entre outros. Também o nodo **Statistics** permitiu a observação dos valores presentes nas *features* e os seus respetivos histogramas.

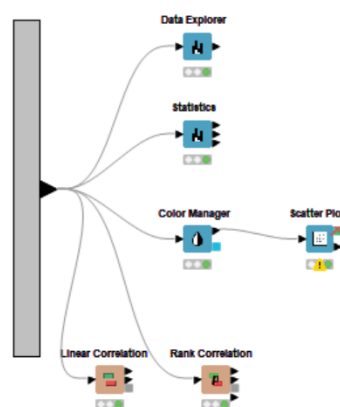


Figura 11: Estudo dos dados (Statistics) no KNIME

Com o **Data Explorer** conseguimos analisar que existem valores que não fazem sentido, como *outliers* nas *features* *ap\_hi*, *ap\_lo* e *weight*, por exemplo. Estes valores serão tratados na

secção seguinte. Há que ter em conta que este *dataset* não apresenta *nominal* nem *missing values*.

Column	Exclude Column	Minimum	Maximum	Mean	Standard Deviation	Variance	Skewness
id	<input type="checkbox"/>	0	99999	49972.420	28851.302	832397645.743	-0.001
age	<input type="checkbox"/>	10798	23713	19468.866	2467.252	6087330.790	-0.307
gender	<input type="checkbox"/>	1	2	1.350	0.477	0.227	0.631
height	<input type="checkbox"/>	55	250	164.359	8.210	67.406	-0.642
weight	<input type="checkbox"/>	10	200	74.206	14.396	207.238	1.012
ap_hi	<input type="checkbox"/>	-150	16020	128.817	154.011	23719.517	85.296
ap_lo	<input type="checkbox"/>	-70	11000	96.630	188.473	35521.895	32.114
cholesterol	<input type="checkbox"/>	1	3	1.367	0.680	0.463	1.587
gluc	<input type="checkbox"/>	1	3	1.226	0.572	0.327	2.397
smoke	<input type="checkbox"/>	0	1	0.088	0.283	0.080	2.906
alco	<input type="checkbox"/>	0	1	0.054	0.226	0.051	3.957
active	<input type="checkbox"/>	0	1	0.804	0.397	0.158	-1.529
cardio	<input type="checkbox"/>	0	1	0.500	0.500	0.250	0.001

Figura 12: Detalhes do nodo Data Explorer

Já com o nodo **Scatter Plot** é possível analisar a relação entre a altura e idade de uma pessoa, sabendo o seu género. A construção do gráfico é apresentada de tal forma que os valores a vermelho correspondem ao sexo feminino e a azul masculino. O que se pode concluir da imagem é que existem diversos *outliers* em relação a *height* e, maior parte dos casos estão bem condensados num intervalo de valores relativamente a *age*. Podemos dizer, por exemplo, que pessoas do sexo feminino tendem a ter menor altura que as outras.

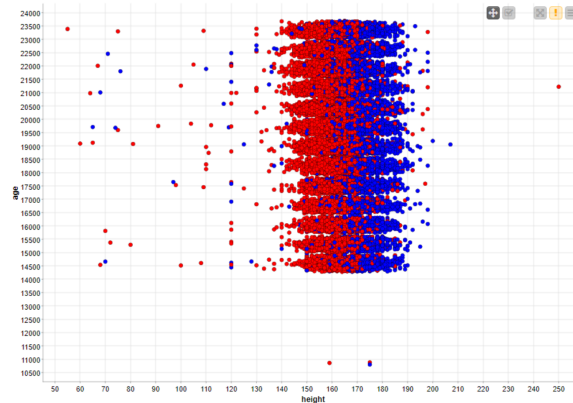


Figura 13: Detalhes do nodo Scatter Plot

Um outro exemplo que pode ser considerado é o apresentado de seguida. Os valores a azul representa que é fumador (1) e a vermelho que não fuma (0). Também é possível observar os *outliers* de *ap\_hi* e de *weight*. Denota-se uma predominância de pessoas não fumadoras,

tendo em conta a análise das primeiras 50000 linhas.

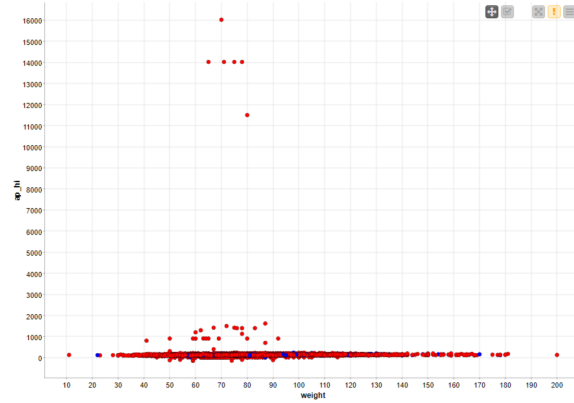


Figura 14: Detalhes do nodo Scatter Plot

Por outro lado, fizemos uso dos nodos **Linear Correlation** e **Rank Correlation** de modo a conseguirmos ter uma noção de quais são os valores que se podem relacionar entre si. O **Linear Correlation** permite comparar variáveis contínuas, observável na imagem seguinte.

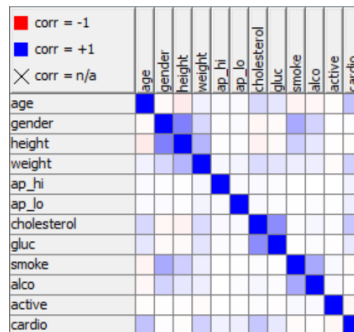


Figura 15: Matriz de Correlação do nodo Linear Correlation

Aqui podemos analisar que existem diversas colunas que se correlacionam positivamente, ou seja, relacionam-se na mesma direção. Com esta análise não podemos tirar conclusões precipitadas servindo apenas para termos um ponto de partida para comparar as diferentes *features*. Com a imagem é possível observar que, por exemplo, *active* não apresenta qualquer tipo de correlação. Já, *ap\_hi*, *ap\_lo* apenas apresentam correlação positiva com *cardio*, que é a nossa previsão final. Assim sendo, podemos verificar que as *features* mencionadas anteriormente não apresentam grande relevância para o problema em causa. Contudo, apenas vamos filtrar *active* no tratamento dos dados.

Já o **Rank Correlation** permite a comparação entre variáveis que não são numéricas mas apresentam uma escala definida. Neste caso, é mais relevante observar *features* que são considerados categóricos como, por exemplo, *cholesterol* e *smoke*.

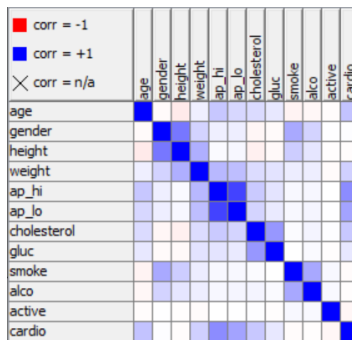


Figura 16: Matriz de Correlação do nodo Rank Correlation

Com a análise da figura anterior concluímos que existem bastante correlação entre *cholesterol* e *gluc*, por exemplo. E, de forma inversa, *smoke* e *age* apresentam correlação negativa.

### 2.3.2 Tratamento de dados

No pré-processamento dos dados passamos por filtrar colunas desnecessárias ao problema, mudar a forma de apresentação de algumas e houve remoção de *outliers*. É essencial uma boa preparação dos dados de modo a remover algumas informações que não irão contribuir para a aprendizagem do modelo.

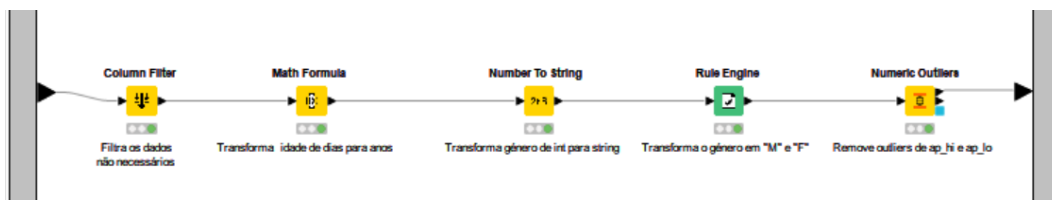


Figura 17: Preparação dos dados no KNIME

Inicialmente, usando o **Column Filter** decidimos remover a coluna *id* que não influencia em nada o problema e *active* que, como já visto no estudo dos dados não apresenta qualquer correlação com as suas demais. Com o **Math Formula** conseguimos converter a nossa coluna *age* de dias para anos através de uma fórmula matemática e, para melhor percepção dos dados, transformamos a representação da coluna *Gender* de 1 e 2 para 'W' e 'M', mulher e homem, respetivamente com o **Rule Engine**.

Tal como foi possível concluir na secção anterior de estudo dos dados, existem diversas *features* com *outliers*, como se pode ver na figura seguinte, em que decidimos proceder à eliminação das linhas onde estão os valores mínimos e máximos dos mesmos.

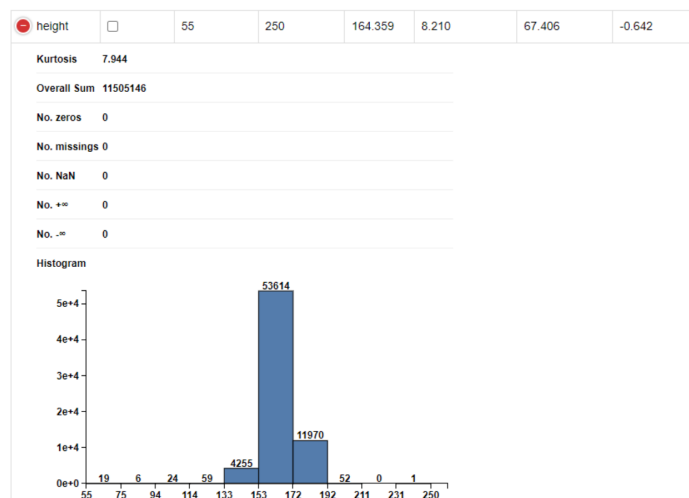


Figura 18: Detalhes da *height* antes da preparação de dados

Tentamos testar outras hipóteses para tratar os dados mas o que concluímos é que não melhoravam em nada os nossos resultados, fizemos, por exemplo, normalização dos valores das pressões arteriais de modo a terem uma escala semelhante ao colesterol mas sem sucesso.

Após a preparação de todos os dados é possível mostrá-los, pelo **Data Explorer**, tal como apresentado na figura seguinte.

Column	Exclude Column	Minimum	Maximum	Mean	Standard Deviation	Variance	Skewness
age	<input type="checkbox"/>	30	65	53.406	6.742	45.453	-0.316
height	<input type="checkbox"/>	143	186	164.406	7.531	56.721	0.069
weight	<input type="checkbox"/>	40	107	73.181	12.272	150.604	0.389
ap_hi	<input type="checkbox"/>	90	170	126.420	14.290	204.199	0.730
ap_lo	<input type="checkbox"/>	65	105	81.699	7.673	58.881	0.451
cholesterol	<input type="checkbox"/>	1	3	1.358	0.675	0.456	1.628
gluc	<input type="checkbox"/>	1	3	1.221	0.568	0.322	2.446
smoke	<input type="checkbox"/>	0	1	0.086	0.281	0.079	2.946
alco	<input type="checkbox"/>	0	1	0.052	0.222	0.049	4.028
active	<input type="checkbox"/>	0	1	0.804	0.397	0.157	-1.534
cardio	<input type="checkbox"/>	0	1	0.494	0.500	0.250	0.024

Figura 19: Dados após a preparação dos mesmos

### 2.3.3 Análise dos dados pós-tratamento

Após o tratamento dos dados houve a necessidade de visualizar os mesmos de modo a perceber o quão eficaz foi o todo processo até ao momento e de forma a termos uma base consistente para prosseguir para a conceção de modelos de aprendizagem.

Através do uso do **Group By** conseguimos agrupar a informação de, por exemplo, o

género, idade, altura e a presença da doença. De seguida são apresentados alguns exemplos conseguidos.

Inicialmente agrupamos as pessoas do estudo pelo seu género, idade e presença ou não da doença cardíaca. Posteriormente comparamos com a média dos valores relativos à altura e ao peso.

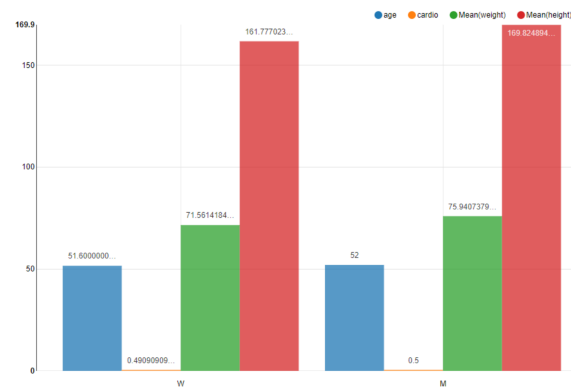


Figura 20: Detalhes do nodo Bar Chart

Seguindo o mesmo raciocínio mas agora não tendo em conta a *feature* idade e considerando os valores médios de *ap\_lo* e *ap\_hi*.



Figura 21: Detalhes do nodo Bar Chart

Por outro lado, decidimos comparar os valores médios de colesterol e glicose para as diferentes idades, pois ambas as *features* encontram-se na mesma escala (1, 2 e 3).



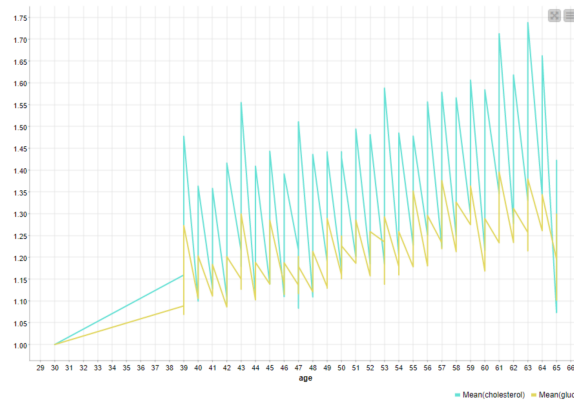


Figura 22: Detalhes do nodo Line Plot

Explorando o nodo **Numeric Binner** decidimos agrupar algumas *features* em intervalos de tempo, como por exemplo *weight*. Na figura seguinte compara-se o número de pessoas que tem a doença cardíaca (*feature cardio* é 1) com as escalas de peso das diferentes pessoas. É de notar que, apesar do valor mínimo presente de *weight* ser 40, o intervalo "40-" é inclusivo, ou seja, alberga todos os valores até 40 (sem exclusão do valor 40).

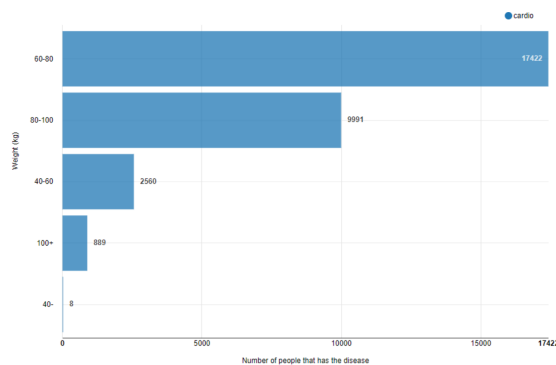


Figura 23: Detalhes do nodo Bar Chart

### 2.3.4 Modelos de Machine Learning

Já podemos concluir que o problema se trata de um problema de classificação mas, mesmo assim, decidimos tratar da modelação dos dados tanto como um modelo de regressão como classificação. De seguida iremos enumerar quais os algoritmos usados para cada um dos tipos de problema e respetivos resultados.

O objetivo dos modelos de aprendizagem é serem capazes de prever a presença ou ausência de uma doença cardiovascular.

## Modelos de classificação

Inicialmente como a *feature* da presença ou não da doença em causa está representada como *string* decidimos usar o nodo **Number To String** de modo a conseguirmos ter acesso à mesma de forma categórica em cada um dos algoritmos.

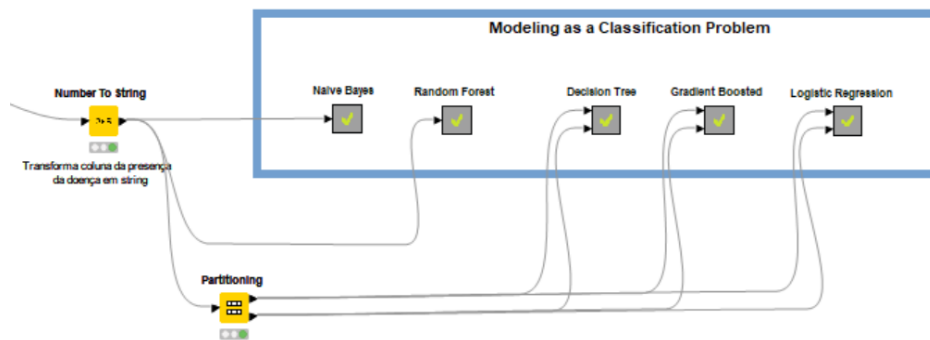


Figura 24: Modelos de classificação no KNIME

De maneira a aplicar os conhecimentos dados tanto nas aulas práticas como investigados pelos membros dos grupos, decidimos para alguns algoritmos usar o nodo **Partitioning** e para outros o **X-Partitioner**.

Para termos a certeza que a partição dos dados é feita de igual forma, aplicamos apenas um nodo **Partitioning** que redirecionava inicialmente 70% dos dados do *dataset* para treino e 30% para teste. Experimentando com outros valores reparamos que com 80% para treino é como se obtém melhores resultados.

Em ambas as formas de partição, **Partitioning** e **X-Partitioner**, é usada a mesma *random seed* para todos estes casos garantindo assim a aplicação dos algoritmos de forma consistente ao mesmo conjunto de dados.

Através do nodo **Scorer (JavaScript)** é possível saber a qualidade do modelo, ou seja, é possível consultar a *accuracy* através das imagens seguintes.

Overall Statistics				
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified
71.10%	28.90%	0.420	44438	18067

(a) Scorer View de Nave Bayes

Overall Statistics				
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified
72.50%	27.50%	0.449	45316	17189

(b) Scorer View de Random Forest

Overall Statistics				
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified
65.50%	34.50%	0.310	8188	4313

(c) Scorer View de Decision Tree

Overall Statistics				
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified
73.07%	26.93%	0.461	9135	3366

(d) Scorer View de Gradient Boosted

Overall Statistics				
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified
66.06%	33.94%	0.324	8258	4243

(e) Scorer View de Logistic Regression

Após a apresentação dos resultados somos capazes de observar que os algoritmos que apresentam melhores soluções são *Gradient Boosted* e *Random Forest*.

## Modelos de regressão

O nodo **Partitioning** e **X-Partitioner** foi aplicado da mesma forma e pela mesma razão como mencionado na secção "Modelos de classificação".

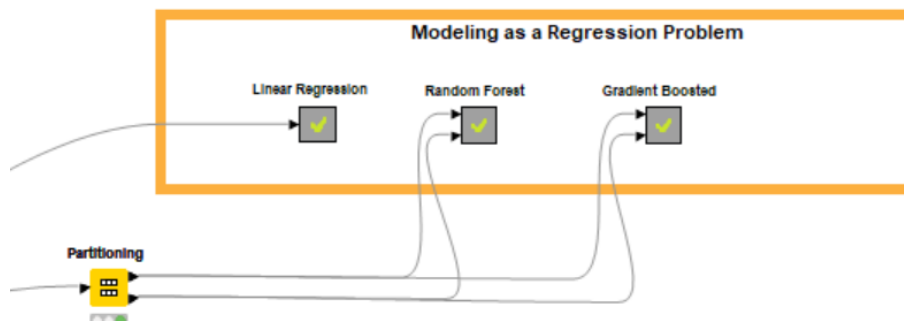
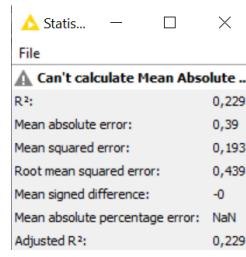
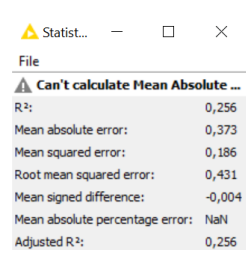
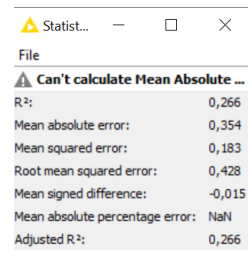


Figura 25: Modelos de Regressão no KNIME

Com base nos algoritmos usados e na seleção das diferentes *features* vamos apresentar o resultado de cada um deles. Com o nodo **Numeric Scorer** é possível saber a qualidade do modelo para isso, vamos ter em atenção ao valor de  $R^2$ .

 <p>Statist... — □ ×</p> <p>File</p> <p>Can't calculate Mean Absolute ...</p> <p>R<sup>2</sup>: 0,229</p> <p>Mean absolute error: 0,39</p> <p>Mean squared error: 0,193</p> <p>Root mean squared error: 0,439</p> <p>Mean signed difference: -0</p> <p>Mean absolute percentage error: NaN</p> <p>Adjusted R<sup>2</sup>: 0,229</p>	 <p>Statist... — □ ×</p> <p>File</p> <p>Can't calculate Mean Absolute ...</p> <p>R<sup>2</sup>: 0,256</p> <p>Mean absolute error: 0,373</p> <p>Mean squared error: 0,186</p> <p>Root mean squared error: 0,431</p> <p>Mean signed difference: -0,004</p> <p>Mean absolute percentage error: NaN</p> <p>Adjusted R<sup>2</sup>: 0,256</p>	 <p>Statist... — □ ×</p> <p>File</p> <p>Can't calculate Mean Absolute ...</p> <p>R<sup>2</sup>: 0,266</p> <p>Mean absolute error: 0,354</p> <p>Mean squared error: 0,183</p> <p>Root mean squared error: 0,428</p> <p>Mean signed difference: -0,015</p> <p>Mean absolute percentage error: NaN</p> <p>Adjusted R<sup>2</sup>: 0,266</p>
(a) Scorer de Linear Regression	(b) Scorer de Random Forest	(c) Scorer de Gradient Boosted

Posto os valores de qualidade podemos concluir que os algoritmos que apresentam melhores resultados são *Gradient Boosted* e *Random Forest*.

Para além de todos os algoritmos aplicados, houve tentativas de aplicação de algoritmos na categoria de *clustering*, como o *k-Means* e *k-Medoids* e, também, da categoria de redes neurais mas, por causa de dificuldades intermédias e maus resultados obtidos, não apresentamos os mesmos no presente relatório.

### 2.3.5 Análise de resultados

Nesta secção vamos comparar entre os diferentes algoritmos e, também, equipar os resultados obtidos tanto na modelação de classificação como de regressão. A divisão dos dados é feita de forma semelhante para todos os nodos em questão, isto é garantido pelo mesmo valor atribuído à *random seed*.

Se tratarmos o problema tendo em conta que se trata de um problema de regressão, o algoritmo que revelou melhores resultados foi o *Gradient Boosted Trees*, não tendo em conta todas as características do conjunto de dados, isto é, não foram consideradas as *features smoke* e *alcohol*. Ou seja, ao incorporar as *features* mencionadas, os resultados eram piores.

Porém, quando este problema é elaborado como um problema de classificação, verificamos que os algoritmos *Gradient Boosted* e *Random Forest* que revelam uma *accuracy* de 73.7% e 72.50%, respetivamente, tendo em conta todas as características do *dataset* selecionado.

## 3 Obesidade

### 3.1 Características do dataset

O conjunto de dados fornecido ao grupo de trabalho pela equipa docente foi disponibilizado com base na paridade do número de grupo (neste caso, 12) e consiste no *dataset obesidade.csv* com 17 *features* e 2111 registos. Este *dataset* tem como *target* o nível de

obesidade, ou seja, os dados podem ser utilizados para a previsão do nível de obesidade de um indivíduo.

Na seguinte listagem são apresentadas as *features* presentes, descrevendo sucintamente o seu significado, assim como o tipo de dados utilizado para a sua representação:

- **Gender:** Sexo do indivíduo (representado sobre a forma de uma String)
- **Age:** Idade do indivíduo (representado sobre a forma de um Double)
- **Date\_of\_birth:** Data de nascimento do indivíduo (representado sobre a forma de uma String)
- **Height:** Altura do indivíduo (representado sobre a forma de um Double)
- **Weight:** Peso do indivíduo (representado sobre a forma de um Double)
- **family\_history\_with\_overweight:** Histórico familiar em obesidade, isto é, a existência de familiares que sofrem/sofreram de obesidade (representado sobre a forma de uma String)
- **FAVC:** Consumo frequente de alimentos calóricos (representado sobre a forma de uma String)
- **FAVC:** Consumo frequente de alimentos calóricos (representado sobre a forma de uma String)
- **FCVC:** Frequência de consumo de vegetais (representado sobre a forma de uma String)
- **NCP:** Número de refeições principais ingeridas ao dia (representado sobre a forma de um Double)
- **CAEC:** Consumo de comida entre refeições (representado sobre a forma de uma String)
- **SMOKE:** Fumador (Se o indivíduo fuma ou não) (representado sobre a forma de uma String)
- **CH2O:** Consumo de água diariamente (representado sobre a forma de um Double)
- **SCC:** Monitorização das calorias consumidas (se o indivíduo monitoriza as calorias ingeridas) (representado sobre a forma de uma String)
- **FAF:** Frequência de atividade física (representado sobre a forma de um Double)
- **TUE:** Tempo utilizado em dispositivos tecnológicos (representado sobre a forma de um Double)
- **CALC:** Consumo de álcool (representado sobre a forma de uma String)
- **MTRANS:** Transporte utilizado (representado sobre a forma de um)

## 3.2 KNIME workflow

A estrutura do *workflow* desenvolvido para o *dataset* obesidade.csv encontra-se representado na Figura 26.

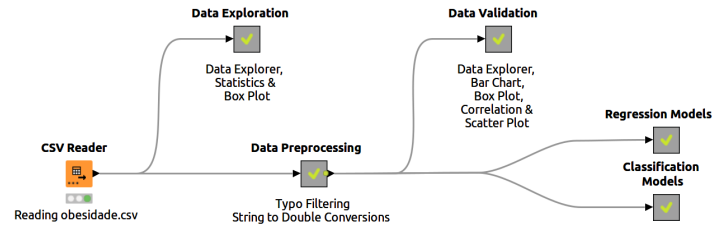


Figura 26: Workflow para estudo sobre obesidade KNIME

### 3.2.1 Estudo dos dados

Na primeira análise realizada sobre o *dataset*, com intuito de identificar valores anómalos que não estavam de acordo com os valores esperados para cada *feature*, foram utilizados os nodos **Data Explorer**, **Statistics** e **Box Plot**.

O nodo **Data Explorer** permitiu uma visualização rápida dos dados, exibindo informações como a distribuição das variáveis e a presença de *missing values*. Foi a ferramenta principal para encontrar erros de ortográficos e valores irregulares assim como sinónimos de certos valores.

Column	Exclude Column	No. missings	Unique values	All nominal values
Gender	<input type="checkbox"/>	0	4	Male, Female, Woman, Man
FCVC	<input type="checkbox"/>	0	810	always, sometimes, never, sometimes,56906always, sometimes,4078never7, [...], sometimes,57neversometimes74, never,9always6479, sometimes,49445never, sometimes,5always0sometimesalwaysalways, never,neversometimesometimesneversometimes7
CAEC	<input type="checkbox"/>	0	5	Sometimes, Frequently, Always, no, Sometyes
CALC	<input type="checkbox"/>	0	5	Sometimes, no, Frequently, Frequently, Always

Figura 27: Uso do Data Explorer para encontrar dados anómalos

Além disso, o nodo **Statistics** forneceu medidas estatísticas descritivas, como média, mediana, desvio padrão, mínimo e máximo, para cada variável. Essas estatísticas ajudaram a ter uma visão geral das características e variabilidade dos dados, apesar de não ser a melhor fase para analisar dados estatísticos devido aos pontos referidos anteriormente.

Por fim, o nodo **Box Plot** foi utilizado com o intuito de visualizar as métricas estatísticas de forma mais gráfica, principalmente para se ter uma melhor ideia sobre a distribuição dos valores e *outliers*.

### 3.2.2 Tratamento de dados

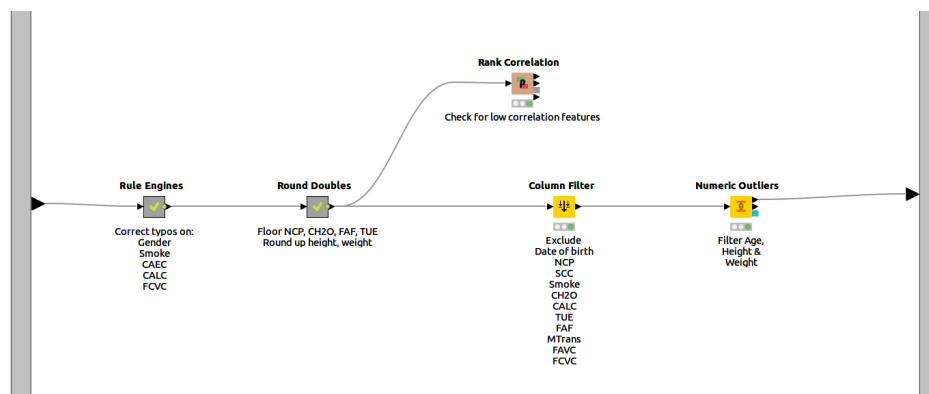


Figura 28: Metanodo para o pré-processamento dos dados

#### Erros ortográficos e valores residuais

Com a informação obtida da exploração de dados, foi realizado o tratamento de dados de forma a uniformizar os valores e corrigir erros de ortográficos. Alguns exemplos (Fig. 32) são valores como "Man" e "Woman" passaram para "Male" e "Female", "Sometymes" que foram corrigidos para "Sometimes", "Frequently" que foram corrigidos para "Frequently", nas colunas **Gender**, **CAEC** e **CALC**, respectivamente. Na coluna **FCVC** havia uma grande quantidade de dados inutilizáveis, porém foi detectado um padrão: todas as linhas adulteradas continham valores aceites, seguidos de uma vírgula e valores residuais. Utilizando o nodo **Java Snippet** em conjunto com a biblioteca *java.util.regex* para o uso de expressões regulares, foi extraída a primeira palavra até à vírgula nos casos dos dados alterados.

#### Arredondamentos

Foi também removida a parte decimal de todos os valores das colunas em que os valores supostos estão numa escala inteira, nomeadamente as features **NCP**, **CH2O**, **FAF** e **TUE**. Os valores das colunas **Height** e **Weight** foram arredondados para cima usando no máximo duas casas decimais no caso da altura e uma casa decimal no caso do peso.

## Correlação de features

Ainda no tratamento de dados foi utilizado o nodo **Rank Correlation** para se verificar a correlação entre as diferentes *features*, no sentido de se eliminar aquelas com baixas correlações em relação às restantes.

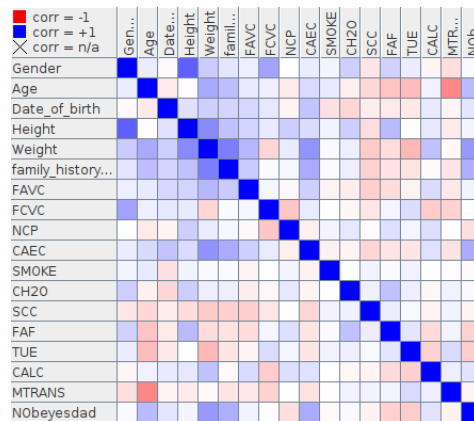


Figura 29: Uso do Rank Correlation para verificar correlação das features

Desta forma foi decidido pela equipa manter apenas as *features* que tivessem correlações abaixo de -0.4 e acima de 0.4.

Desta forma, foram eliminadas as seguintes colunas:

- Date\_of\_birth
- NCP
- SCC
- Smoke
- CH2O
- CALC
- TUE
- TUE
- FAF
- MTRANS
- FAVC
- FCVC



### Remoção de outliers

Finalmente, foi utilizado o nodo **Numeric Outliers** para remover os valores discrepantes, de forma a não afetar a tendência geral dos dados. Assim, removendo os outliers das *features* **Age**, **Height** e **Weight** foram removidas 148 linhas da tabela, resultando assim na tabela final para ser realizada a validação dos dados.

### 3.2.3 Análise dos dados pós-tratamento

Uma vez realizado o processamento dos dados, a fase seguinte consiste na validação dos dados processados para a fase de modelação. Nesta fase foram utilizados os nodos **Data Explorer**, **Bar Chart**, **Box Plot** e **Scatter Plot** para verificar se as alterações realizadas na fase de tratamento forem bem sucedidas. Foram analisados alguns dados estatísticos, como por exemplo, a dispersão dos diferentes níveis de obesidade, que foram distinguidos graficamente com o auxílio do nodo **Color Manager**, atribuindo uma cor diferente a cada tipo, de acordo com a altura e peso utilizado o **Scatter Plot** para verificar a relação. Também foi analisada a frequência absoluta do número de indivíduos por cada nível de obesidade, podendo-se concluir que não há nenhum nível com valores discrepantes ou extremos.

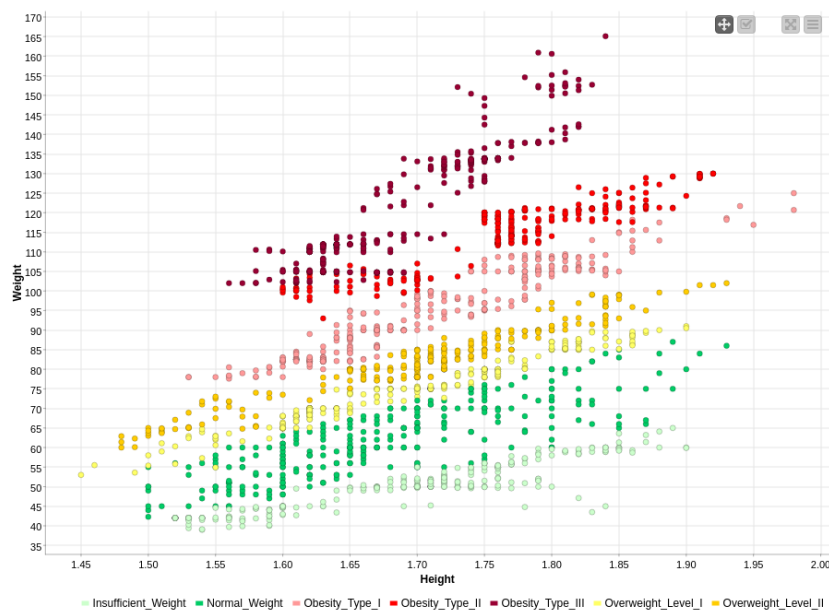


Figura 30: Uso do Scatter Plot para analisar o tipo de obesidade em relação ao peso e altura

### 3.2.4 Modelos de Machine Learning

Após a etapa de pré-processamento dos dados, o dataset em questão encontra-se pronto para ser abordado pelo uso de técnicas de *Machine Learning*.

Foi realizada uma preparação dos dados para conseguirmos aplicar tanto modelos de classificação como de modelos de regressão, a fim de obter resultados valiosos e tomar decisões informadas.

## Modelos de Classificação

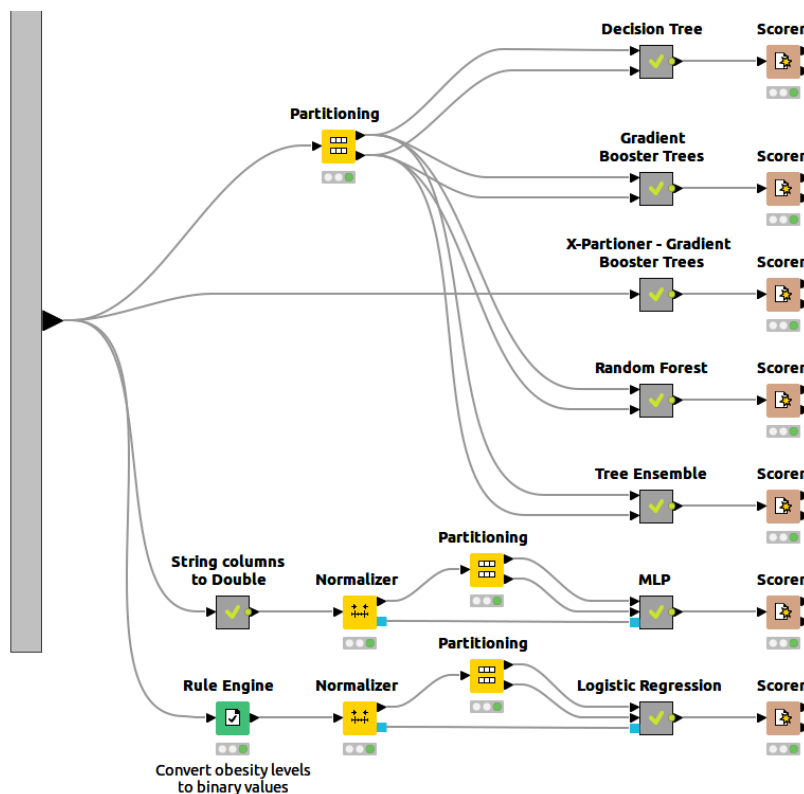


Figura 31: Metanodo relativo aos modelos de classificação

No que toca ao uso de modelos de classificação, a *feature* que queremos prever, **NObesidad**, é uma variável categórica que representa a categoria de obesidade de um indivíduo, não sendo necessário realizar nenhum tratamento ou processamento adicional (exceto para os modelos **MLP** e **Logistic Regression**). Foram utilizados seis modelos diferentes, nomeadamente **Decision Tree**, **Gradient Boosted Trees**, **Random Forest**, **Tree Ensemble**, **MLP** e **Logistic Regression**, no intuito de avaliar o desempenho destes na previsão do tipo de obesidade.

Desta forma, podemos aplicar o nodo **Partitioning** para dividir a tabela completa em duas tabelas, uma que servirá de *input* para os nodos de aprendizagem dos modelos, e a outra que irá ser *input* para os nodos de previsão de dados para teste do modelo gerado. A partição dos dados foi realizada com a técnica *Stratified Sampling* usando a seed 841738, em que 70% dos dados são utilizados para treino e aprendizagem dos modelos e os outros 30% serão utilizados para previsão do tipo de obesidade de um indivíduo.

Os dados também foram normalizados e desnormalizados (usando os nodos **Normalizer** e **Denormalizer**), respetivamente, no caso dos modelos *Multilayer Perceptron* (MLP) e *Logistic Regression*, uma vez que estes modelos são sensíveis à escala e a normalização dos dados

permite uma melhor precisão na previsão de resultados. Os restantes modelos não receberam dados normalizados, uma vez que não são sensíveis à escala.

É de notar que para estes últimos modelos, **MLP** e **Logistic Regression**, foi realizado um pré-processamento adicional dos dados antes da normalização. O modelo **MLP** necessita que os valores de *input* sejam numéricos, posto isto foi utilizado *Label Encoding* para as variáveis **Gender**, **family\_history\_with\_overweight** e **CAEC**. Já para o *Logistic Regression*, sendo este um modelo de classificação binária, foi necessária a conversão da variável dos tipos de obesidade para uma variável binária. Desta forma, a nova variável define se um indivíduo é obeso ou não de acordo com a categoria de obesidade a que pertence, os níveis de obesidade I, II e III, tomam valor *True* e os restantes *False*.

No que diz respeito à avaliação dos modelos, estes obtiveram resultados muito satisfatórios por parte do nodo **Scorer**, obtendo valores de precisão da previsão entre os 93% e os 96%. Podemos considerar que a escolha de um modelo para a previsão do tipo da obesidade dependerá do tempo de treino e previsão e da escalabilidade da quantidade dos dados.

Não poderá ser considerado válido para comparação o modelo **Logistic Regression**, uma vez que prevê dados para um problema de classificação binária e o domínio da classe prevista não é o mesmo. Este foi o modelo que demonstrou ser o mais preciso, obtendo uma precisão de 100%, mas, sendo o domínio da classe de previsão binário, é normal que seja um problema mais inclinado a obter maior precisão.

Por outro lado, também foram usados os nodos **X-Partitioner** e **X-Aggregator** em conjunto com o modelo **Gradient Boosted Trees**, que foi o modelo que alcançou maior precisão (95,925%) com o nodo **Partitioning**, para testar se uma partição dos dados usando *cross-validation* aumentaria o desempenho do modelo. Isto foi verificado, obtendo uma precisão de 97,453%, mas, embora a precisão tenha aumentado ligeiramente, a diferença não foi estatisticamente significativa para justificar a utilização adicional do **X-Partitioner** e **X-Aggregator** em conjunto com o modelo **Gradient Boosted Trees**. Portanto, a abordagem com o nodo **Partitioning** foi considerada suficiente para a obtenção de resultados precisos.

Em suma, é possível confirmar que o pré-processamento dos dados ajudou a alcançar bons resultados e houve uma seleção de modelos adequados. Isso resultou em resultados promissores, com altos níveis de precisão na previsão do tipo de obesidade, podendo apoiar a tomada de decisões informadas no contexto do problema em questão.

## Modelos de Regressão

Já no caso dos modelos de regressão, utilizamos as outras *features* do *dataset*, nomeadamente o peso e altura, para prever um valor contínuo ou numérico relacionado à obesidade, como o índice de massa corporal, **IMC**. Para obter este valor, foi utilizado o nodo **Java Snippet** para adicionar 2 colunas de dados:

- **IMC**, que representa o valor do índice de massa corporal, calculado segundo a fórmula:

$$IMC = Weight/Height^2$$

- **IMC-Type**, que representa a classificação correspondente ao valor de IMC calculado anteriormente.

```

out_IMC = c_Weight / Math.pow(c_Height, 2);

if ( out_IMC < 18.5 ){
    out_IMCType = "Insufficient_Weight";
}
else if ( out_IMC >= 18.5 && out_IMC < 25.0 ){
    out_IMCType = "Normal_Weight";
}
else if ( out_IMC >= 25.0 && out_IMC < 27.5 ){
    out_IMCType = "Overweight_Level_I";
}
else if ( out_IMC >= 27.5 && out_IMC < 30.0 ){
    out_IMCType = "Overweight_Level_II";
}
else if ( out_IMC >= 30.0 && out_IMC < 35.0 ){
    out_IMCType = "Obesity_Type_I";
}
else if ( out_IMC >= 35.0 && out_IMC < 40.0 ){
    out_IMCType = "Obesity_Type_II";
}
else if ( out_IMC >= 40.0 ){
    out_IMCType = "Obesity_Type_III";
}

```

Figura 32: Cálculo do IMC e criação da variável IMC-Type a partir do IMC

De forma a comparar as classificações originais do *dataset* (**NObeyesdad**) e as classificações resultantes do cálculo (**IMC-Type**), foi utilizado o nodo **Scorer** para comparar ambas as colunas de dados e observar a variação de resultados.

NObeyes...	Normal_W...	Overweig...	Overweig...	Obesity_T...	Insufficie...	Obesity_T...	Obesity_T...
Normal ...	278	0	0	0	3	0	0
Overweig...	12	258	3	0	0	0	0
Overweig...	0	51	203	3	0	0	0
Obesity_T...	0	0	3	283	0	2	0
Insufficie...	5	0	0	0	266	0	0
Obesity_T...	0	0	0	20	0	250	0
Obesity_T...	0	0	0	0	0	60	263
Correct classified: 1 801				Wrong classified: 162			
Accuracy: 91,747%				Error: 8,253%			
Cohen's kappa (κ): 0,904%							

Figura 33: Scorer - NObeyesdad vs IMC-Type

Desta forma podemos concluir que existem linhas cuja atribuição do tipo de obesidade não corresponde ao tipo de obesidade obtido através do cálculo do IMC.

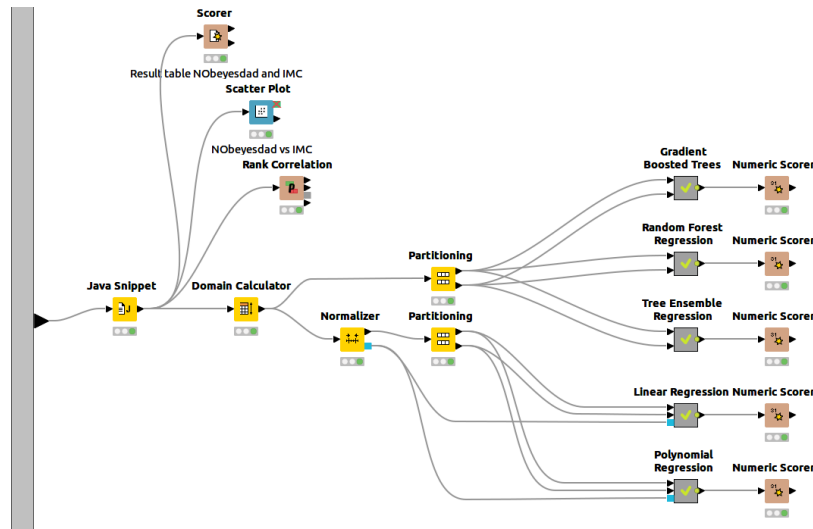


Figura 34: Metanodo relativo aos modelos de regressão

Foram utilizados 5 modelos diferentes de regressão, nomeadamente **Gradient Boosted Trees**, **Random Forest**, **Tree Ensemble**, **Linear Regression** e **Polynomial Regression**, com o intuito de avaliar o seu desempenho na previsão do índice de massa corporal de um indivíduo.

Com isto, podemos aplicar o nodo **Partitioning** de forma a efetuar a partição dos dados resultando em dois conjuntos, um que servirá de input para os nodos de aprendizagem dos modelos e outro que servirá de input para os nodos de previsão de dados, para teste do modelo gerado. Foi utilizada a técnica de *Stratified Sampling* segundo a coluna **IMC-Type** usando a seed 841738, em que 70% dos dados são utilizados para treino e aprendizagem dos modelos e os outros 30% serão utilizados para previsão do valor do IMC.

Além disso, os dados foram normalizados (com recurso aos nodos **Normalizer** e **Denormalizer**) para os modelos **Linear Regression** e **Polynomial Regression** uma vez que estes modelos são sensíveis à escala e a normalização dos dados permite uma melhor precisão na previsão de resultados. Os restantes modelos não receberam dados normalizados, uma vez não são sensíveis à escala.

### 3.2.5 Análise de resultados

Na Tabela 3.1 podemos comparar os resultados de todos os modelos de regressão, todos apresentando bons resultados com valores de  $R^2$  superiores a 0,985, destacando-se o modelo de regressão **Gradient Boosted Trees** com  $R^2$  0,996 juntamente com um MAE (Mean Absolute Error) de 0,38. Estes valores indicam que este modelo foi capaz de explicar 99,6% das variações dos dados, com um erro absoluto médio de 0,38 unidades.

Modelo	$R^2$	MAE
Gradient Boosted Trees	0,996	0,38
Random Forest	0,986	0,695
Tree Ensemble	0,987	0,684
Linear Regression	0,995	0,466
Polynomial Regression	0,993	0,519

Tabela 3.1: Resultados obtidos dos modelos de regressão

Já a especificação dos resultados dos modelos de classificação encontram-se na secção "Modelos de classificação" onde são apresentados os valores obtidos em cada um dos algoritmos mas podemos concluir que o que obteve melhor resultado foi o *Gradient Boosted Trees*.

## 4 Conclusão

Durante o processo de análise, foram aplicados diversos conceitos aprendidos nas aulas, que incluíram desde a aplicação de diferentes técnicas de pré-processamento dos dados até a seleção de algoritmos adequados aos conjuntos de dados em questão.

Apesar de não terem sido identificadas muitas necessidades de tratamento de dados no conjunto de dados sobre doenças cardiovasculares, consideramos que realizamos uma boa exploração das diversas características disponíveis e apresentamos resultados relevantes e variados.

## 5 Referências

- *Dataset selecionado (Cardiovascular Disease dataset)*  
<https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset>
- Valores de referência para IMC (Obesidade *dataset*)  
<https://www.who.int/europe/news-room/fact-sheets/item/>