Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Aprendizagem e Decisão Inteligentes

Ano Letivo 2022/2023

Conceção de modelos de aprendizagem

Grupo 16

Ana Rita Santos Poças, A97284
Miguel Silva Pinto, A96106
Orlando José da Cunha Palmeira, A97755
Pedro Miguel Castilho Martins, A97613

12 de maio de 2023

Índice

Índice de figuras

1 Introdução

Este relatório surge no âmbito da Unidade Curricular de Aprendizagem e Decisão Inteligentes, em que nos foi proposto a conceção de modelos de aprendizagem. O trabalho a realizar engloba duas tarefas. A primeira tarefa, consiste na consulta, análise, exploração e preparação de um *dataset* selecionado pelos elementos do nosso grupo. Já a segunda tarefa, consiste na exploração análise e preparação de um *dataset* atribuído pelos docentes da Unidade Curricular.

2 Tarefa A: Car Price

A tarefa A consiste em o grupo de trabalho escolher um *dataset*, fazer a sua análise e recolher o conhecimento relevante ao contexto do problema.

Para esta fase o grupo explorou 3 *datasets* de forma a encontrar aquele que nos permitisse mostrar o conhecimento adquirido durante o semestre. Após inicializarmos a exploração de cada um deles, decidimos que o mais adequado para a entrega seria o *dataset Car Price*, visto que, apesar de não apresentar uma grande necessidade de tratamento de dados (que é compensada na tarefa B), o problema proposto permite que exploremos outros tipos de modelação para além dos que efetuamos para a tarefa B, e desta forma, conseguirmos evidenciar o esforço do grupo em demonstrar as aprendizagens consolidadas.

O *dataset* escolhido contém informação de carros de todos os tipos e os seus atributos. O objetivo deste problema é descobrir quais os atributos que afetam mais o preço final dos carros e o quão bem esses atributos estão relacionados com o preço do veículo.

A metodologia que será utilizada no processo da resolução do problema é o **CRISP-DM**. O modelo **CRISP-DM** define um guião para o desenvolvimento de projetos de análise de dados dividido em 6 etapas:

1. **Estudo do negócio**
2. **Estudo dos dados**
3. **Preparação dos dados**
4. **Modelação**
5. **Avaliação**
6. **Desenvolvimento**

Nesta tarefa iremos abordar apenas as 5 primeiras etapas, pois a etapa de desenvolvimento não será possível explorar por falta de dados extra para realizar a implementação e monitorização do modelo.

De seguida iremos abordar cada uma dessas etapas e o que foi feito em relação ao problema que abordamos.

2.1 Estudo do negócio

O objetivo deste problema é descobrir com o auxílio de modelos de aprendizagem os atributos que mais influenciam o preço de veículos no mundo automóvel. Para isso temos disponível um *dataset* com informação sobre vários tipo de carros e o auxílio da ferramenta de análise de dados e *machine learning* **KNIME** que nos irá permitir a criação de modelos de aprendizagem e de análise de dados para a resolução do problema.

Os objetivos a serem compridos neste problema são:

1. Analisar o *dataset* por completo.
2. Tratar de inconsistências no *dataset* e extrair conhecimento extra dos atributos se possível.
3. Criar gráficos, matrizes de correlação e outro tipos de visualização de dados.

4. Descobrir a partir de modelos de aprendizagem os atributos que mais afetam o preço dos veículos.

2.2 Estudo dos dados

Os dados para este problema foram retirados do site *kaggle* em Price Prediction e esse *dataset* contém 205 linhas e 26 atributos. Os atributos são os seguintes:

1. **ID**: Id do registo.
2. **Symboling**: Fator de risco associado ao seu preço (valores menores indicam riscos menores).
3. **name**: Nome do carro.
4. **fueltypes**: Tipo de combustível (gás/diesel).
5. **aspiration**: Tipo de entrada de ar no carro (std/turbo).
6. **doornumbers**: Número de portas do carro (two/four).
7. **carbody**: Tipo da carroçaria do carro.
8. **drivewheels**: Número de rodas motrizes (fwd, rwd, 4wd).
9. **enginelocation**: Localização do motor (front/rear).
10. **wheelbase**: Distância entre as rodas traseiras e dianteiras.
11. **carlength**: Comprimento do carro.
12. **carwidth**: Largura do carro.
13. **carheight**: Altura do carro.
14. **curbweight**: Peso do carro com o tanque de combustível cheio e sem passageiros.
15. **enginetype**: Tipo do motor.
16. **cylindernumber**: Número de cilindros no motor.
17. **enginesize**: Tamanho do motor em polegadas cúbicas (cubic inches).
18. **fuelsystem**: Tipo de sistema de combustível.
19. **boreratio**: Rácio entre o diâmetro do cilindro e a distância que o pistão percorre.
20. **stroke**: Distância que o pistão percorre no motor.
21. **compressionratio**: Rácio entre o volume de um cilindro e a câmara de combustão.
22. **horsepower**: Cavalos produzidos pelo motor.
23. **peakrpm**: Número máximo de rotações por minuto que o motor atinge.
24. **citympg**: Média de milhas por galão de combustível em condições de cidade .
25. **highwaympg**: Média de milhas por galão de combustível em condições de autoestrada.
26. **price**: Preço do veículo.

Os dados obtidos no *kaggle* não continham informação sobre os atributos presentes no *dataset* pelo que foi necessário a pesquisa do significado dos atributos.

O atributo **Price** é o atributo objetivo deste problema e é aquele que iremos antever com o desenvolvimento de modelos de previsão de regressão e também será utilizado para descobrir a qualidade dos outros atributos presentes no *dataset*. Para ter uma ideia inicial da qualidade dos outros atributos presentes no *dataset*

fizemos uma análise a cada atributo para descobrir o efeito que têm no preço final do carro.

2.2.1 Price

O preço dos veículos presentes no *dataset* variam entre 5118 e 45400 sendo a média dos preços 13276. No gráfico abaixo podemos ver que grande parte dos veículos têm um preço abaixo dos 25000 o que poderá ajudar a descobrir quais fatores tornam os carros mais caros.

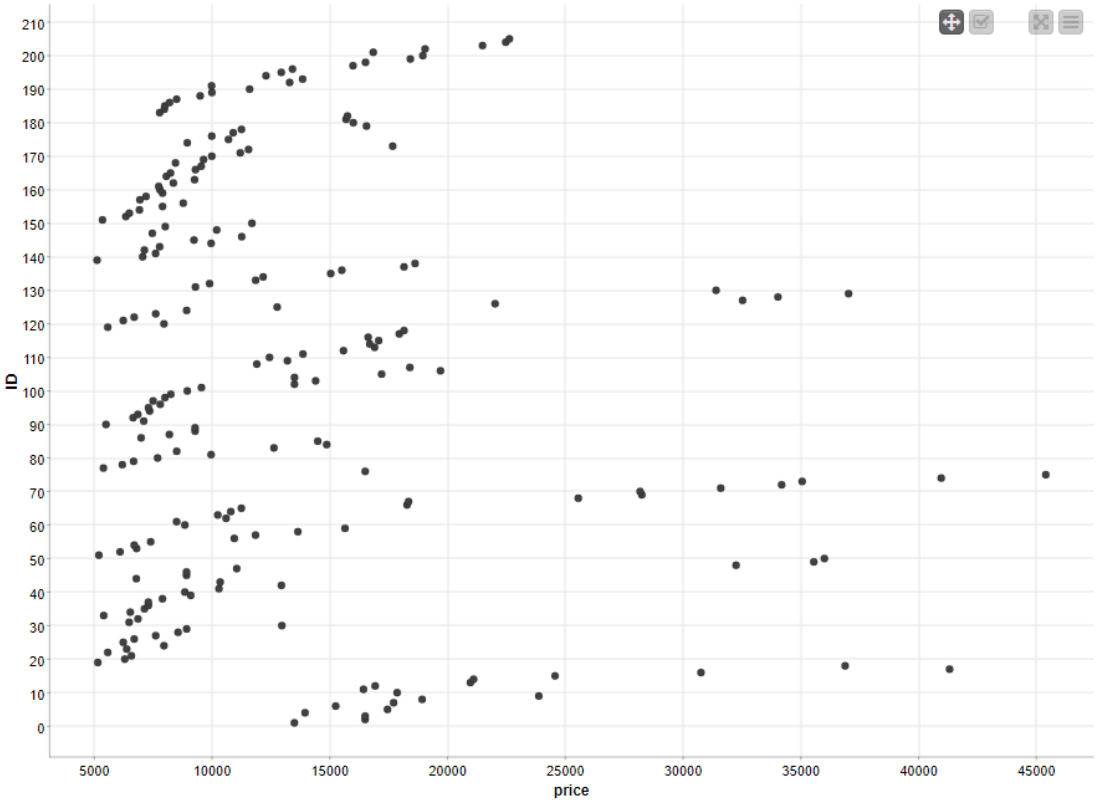

Figura 2.1: Distribuição dos preços dos carros

2.2.2 Performance

Em termos de performance do veículo temos no *dataset* os seguintes atributos:

- **Aspiration**
- **Cylindernumber**
- **Enginesize**
- **Boreratio**
- **Stroke**
- **Compressionratio**
- **Horsepower**
- **Peakrpm**

Todos estes atributos afetam a performance do veículo, e iremos analisar agora o efeito de cada um deles sobre o preço do veículo.

Aspiration: Os veículos com uma entrada de ar *standard* têm em média um preço de 12611 enquanto que os carros com *turbo* têm em média um preço de 16298, em termos de correlação os atributos *price* e *aspiration* têm um valor de 0,309, logo o tipo de aspiração tem um efeito sobre o preço do carro.

Cylindernumber: O número de cilindros no motor de um carro afetam o preço total do veículo. Apesar de cerca de 75% dos veículos presentes no *dataset* terem 4 cilindros, conseguimos agrupar este atributo em 3 grupos em relação ao preço médio do carro. Um carro com 2, 3 ou 4 cilindros no motor tem um valor em média de 10000, um carro com 5 ou 6 cilindros tem um valor em média de 22000, e um carro com 8 ou 12 cilindros tem um valor em média de 36000. Podemos então concluir que em geral quantos mais cilindros o motor possui maior será o preço do carro.

Enginesize: O tamanho do motor de um veículo também tem um efeito no preço total do veículo. No *dataset* conseguimos ver que para motores acima de 190 CI o preço dos carros está acima de 30000 e conseguimos ver também uma subida do preço com a subida do tamanho do motor.

Boreratio: Este atributo não parece ter um grande efeito sobre o preço final do carro quando analisamos o atributo através de um *Scatter Plot*. No entanto, esse atributo possui uma correlação de 0,6438 com o preço do veículo que indica que existe uma relação entre os 2 atributos.

Stroke: Este atributo parece não influenciar o preço dos carros pois apresenta uma correlação de apenas 0,1113.

Compressionratio: Semelhante ao atributo *Stroke* o *Compressionratio* do carro não influencia de forma significativa o preço do veículo.

Horsepower: A potência produzida pelo motor influencia diretamente o valor do carro. Carros com mais cavalos tendem a valer mais do que carros com menos cavalos. Este atributo apresenta uma correlação com o preço de 0,8546, o que mostra o efeito da potência do motor no preço do mesmo.

Peakrpm: O número máximo de rotações por minuto que o motor do carro atinge não tem uma grande influência no preço dos veículos presentes no *dataset*.

Após a análise feita sobre os atributos de performance do carro podemos concluir que a performance do carro influencia o preço final do veículo, principalmente os atributos **Aspiration**, **Cylindernumber**, **Enginesize** e **Horsepower**.

2.2.3 Eficiência

Em termos de eficiência do veículo temos no *dataset* os seguintes atributos:

- **Fueltypes**
- **Fuelsystem**
- **Citympg**
- **Highwaympg**

Estes atributos descrevem a autonomia do veículo. Vamos agora analisar cada um desses atributos para ver o seu efeito no preço final do veículo.

Fueltypes: Os veículos que usam o tipo de combustível *gas* têm em média um preço de aproximadamente 13000 (12999.798) enquanto que os carros que usam diesel *diesel* têm em média um preço de 15838. No entanto no *dataset* cerca de 90% dos carros utilizam o tipo de combustível ”*gas*” . Logo, não conseguimos tirar uma conclusão sobre o efeito real do tipo do combustível no preço.

Fuelsystem: Existem 8 tipos de sistemas de combustíveis no *dataset* analisado. Na tabela abaixo fizemos a análise deste atributo.

Tipo de sistema de combustível	Média do preço dos veículos	Carros com este tipo em %
mpfi	17754	45,85%
2bbl	7478	32,2%
mfi	12964	0,49%
1bbl	7555	5,37%
spfi	11048	0,49%
4bbl	12145	1,46%
idi	15838	9,76%
spdi	10990	4,39%

Como podemos ver na tabela, o tipo de sistema de combustível com maior ocorrências no *dataset* e com maior média de preço é o **mpfi**, o que indica que este tipo é o mais comum entre os carros com os preços acima da média (média de preços no *dataset* é de 13276). O tipo **2bbl** é o segundo mais comum e com a menor média de preços, o que indica que é o tipo mais comum entre os carros com os preços mais baixos. Sobre os outros tipos é mais difícil tirar uma conclusão sobre o seu efeito no preço devido à falta de dados exemplificativos presentes no *dataset*.

Citympg: Este atributo representa a média de milhas por galão de combustível em condições de cidade e pela análise que efetuamos foi possível verificar que quanto menor for o citympg maior tende a ser o preço do veículo. O veículo com menor preço (5118) tem um valor de citympg de 31 e o veículo com maior preço (45400) tem um valor de citympg de 14.

Highwaympg: Este atributo representa a média de milhas por galão de combustível em condições de autoestrada e pela análise efetuada podemos verificar que quanto menor for o valor do atributo highwaympg maior tende a ser o preço do veículo. O veículo com menor preço (5118) tem um valor de highwaympg de 36 e o veículo com maior preço (45400) tem um valor de highwaympg de 16.

2.2.4 Carroçaria

Em termos de carroçaria do veículo temos no *dataset* os seguintes atributos:

- **Doornumbers**
- **Carbody**
- **Enginelocation**
- **Wheelbase**
- **Carlength**
- **Carwidth**
- **Carheight**
- **Curbweight**

Doornumbers: Nos veículos estudados, existem dois tipos de portas: os que têm 2 portas, que têm um preço em média de aproximadamente 12990 (12989.92) e os que têm 4 portas, que têm um preço em média de 13501.

Carbody: Existem 5 tipos de carroçaria. Na tabela abaixo fizemos a análise deste atributo.

Tipo de carroçaria	Média do preço dos veículos	Carros com este tipo em %
convertible	21890	2.93%
hatchback	10376	34.15%
sedan	14344	46.83%
wagon	12371	12.2%
hardtop	22208	3.9%

Como é possível observar na tabela, o tipo de carroçaria com maior ocorrências no *dataset* é o *sedan* (sendo que é o terceiro com menor média de preço) e o tipo com maior média de preço é o **hardtop** (sendo que é o segundo com menos ocorrências no *dataset*).

Enginelocation: Nos veículos estudados, existem os que têm o motor localizado na frente, que têm um preço médio de 12961 e um nível de ocorrências de 98.54% e os que têm o motor localizado na traseira e que têm um preço médio de 34528 e um nível de ocorrências de 1.46%.

Wheelbase: Este atributo representa a distância entre os eixos no veículo, e pela análise que efetuamos foi possível verificar que quanto maior for a distância maior tende a ser o preço do veículo. Sendo que o veículo com menor preço (5118) tem um valor de wheelbase de 93.7 e o veículo com maior preço (45400) tem um valor de wheelbase de 112.

Carlength: Este atributo representa o comprimento do veículo, e a análise efetuada permitiu verificar que quanto maior for o comprimento do veículo, maior tende a ser o seu preço. O veículo com menor preço (5118) tem um valor de carlength de 156.9 e o veículo com maior preço (45400) tem um valor de carlength de 199.2.

Carwidth: Este atributo representa a largura do veículo, e a análise efetuada permitiu verificar que quanto maior for a largura do veículo, maior tende a ser o seu preço. O veículo com menor preço (5118) tem um valor de carwidth de 63.4 e o veículo com maior preço (45400) tem um valor de carwidth de 72.

Carheight: Este atributo representa a altura do veículo, e pela análise que fizemos não foi possível verificar nenhum padrão relativamente à relação preço $\Leftrightarrow$ altura do veículo.

Curbweight: Este atributo representa a tara do veículo, e a análise que fizemos permite-nos concluir que à medida que o atributo curbweight aumenta, o preço do veículo tende a aumentar também. O veículo com menor preço (5118) tem um valor de curbweight de 2050 e o veículo com maior preço (45400) tem um valor de curbweight de 3715.

2.2.5 Atributos extras

Os atributos que não se encaixam nas categorias acima são:

- **Symboling**
- **Name**
- **Drivewheels**
- **Enginetype**

Symboling: Este atributo indica o fator de risco associado ao preço do veículo. Pela análise feita ao atributo não foi possível chegar a uma conclusão concreta sobre o seu efeito no preço do veículo devido à falta de exemplos de carros com preços elevados.

Name: Este atributo indica o nome do carro. Analisando este atributo não conseguimos tirar uma conclusão sobre o seu efeito no preço do veículo devido à quantidade de diferentes tipos de nomes que existem no *dataset*. No entanto podemos extrair o fabricante do carro a partir do nome o que reduz o número de ocorrências deste atributo e assim facilitar a análise do mesmo. Fazendo a análise dos fabricantes dos

carros podemos notar que o preço de cada marca não varia muito de carro para carro. As marcas com os carros mais caros neste *dataset* são **Jaguar**, **Buick**, **Porsche** e **BMW** com um valor médio acima dos 26000. As outras marcas possuem um valor médio entre os 6000 e 18000.

Drivewheels: Este atributo representa o número de rodas motrizes do veículo. Existem 3 tipos deste atributo: fwd, rwd e 4wd.

Tipos de rodas motrizes	Média do preço dos veículos	Carros com este tipo em %
fwd	9239	58.54%
4wd	19919.8	4.39%
rwd	11087	37.07%

Como é possível observar, o tipo *fwd* é o que tem um maior número de ocorrências e o que tem um menor preço médio do veículo.

Enginetype: Este atributo representa o tipo de motor e existem 7 tipos deste atributo: ohc, l, dohcv, dohc, rotor, ohcv e ohcf.

Tipos de motor	Média do preço dos veículos	Carros com este tipo em %
ohc	11574	72.2%
l	14627	5.85%
dohcv	31400	0.49%
dohc	18116	5.85%
rotor	13020	1.95%
ohcv	25098	6.34%
ohcf	13738	7.32%

Como é possível verificar, o tipo de motor com maior número de ocorrências é o *ohc* e, coincidentemente, é também o atributo com menor média de preço dos veículos.

2.3 Preparação dos dados

Com o estudo dos dados feito, passamos para a preparação dos dados e a primeira coisa a fazer foi remover a coluna dos **IDs** dos veículos pois não será um atributo valioso para a parte da modelação por ser um atributo com um valor único para todos os veículos.

Como foi mencionado na análise dos dados, o atributo **nome** não caracterizava bem o preço do veículo devido ao número elevado de nomes presentes no *dataset* e, para tirar um melhor proveito desse atributo, resolvemos criar um novo atributo (**marca**) onde extraímos a primeira parte do nome com um **Regex Split** para obtermos o fabricante do carro. Alguns nomes das marcas continham erros de escrita e para resolver isso foi utilizado um **Rule Engine** para corrigir os valores errados. Com esse novo atributo criado conseguimos fazer uma melhor análise desse atributo e poderá ser mais valioso na parte da modelação.

Além disso, para preparar os dados para uma modelação de redes neuronais foi necessário transformar todos os atributos presentes no *dataset* em *Double* pois o nodo de aprendizagem utilizado apenas aceita atributos desse tipo. Utilizamos então os nodos **Category to Number** e **Math Formula** para transformar os valores discretos em inteiros e os valores inteiros para decimais, respetivamente.

Depois de todos os atributos terem sido convertidos, procedemos à normalização dos mesmos através do nodo **Normalizer** de forma a obter valores entre 0 e 1. A normalização dos atributos é necessária para o correto funcionamento do nodo de aprendizagem porém no **scorer** final gostaríamos de comparar os valores reais o que implicou 2 alterações à preparação:

1. Duplicação da coluna **price** e renomeada para **Prediction (price)** para que a função de normalização saiba como normalizar e desnormalizar o atributo que o **Predictor** irá criar.
2. Desnormalizar os atributos depois da aprendizagem para que o **Numeric scorer** calcule os valores na grandeza do problema original.

2.4 Modelação

2.4.1 Modelação com todos os atributos (Controlo)

A modelação do problema começou pelo uso de nodos de aprendizagem de regressão sobre os atributos presentes no *dataset*. Como foi referido na fase de **Preparação de dados**, o atributo **name** foi transformado no atributo **marca** e após serem feitos alguns testes notamos que esse novo atributo melhorava os resultados obtidos pelos modelos criados.

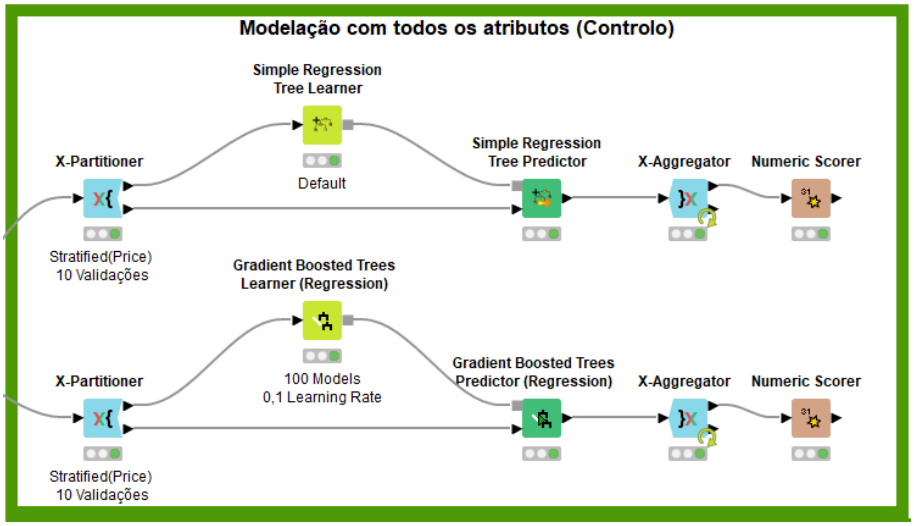

Figura 2.2: Modelação com todos os atributos (Controlo)

Para a criação dos modelos foi utilizado **Cross Validation** com o intuito de avaliar o modelo com diferentes grupos de treino e de teste e evitar bias na seleção dos grupos.

Os algoritmos de aprendizagem utilizados foram **Simple Regression Tree** e **Gradient Boosted Trees (Regression)**. Para estes algoritmos obtivemos os seguintes resultados:

R^2:	0,878	R^2:	0,914
Mean absolute error:	1 841,13	Mean absolute error:	1 542,518
Mean squared error:	7 761 891,383	Mean squared error:	5 448 641,101
Root mean squared error:	2 786,017	Root mean squared error:	2 334,232
Mean signed difference:	40,507	Mean signed difference:	65,325
Mean absolute percentage error:	0,138	Mean absolute percentage error:	0,117
Adjusted R^2:	0,878	Adjusted R^2:	0,914

Figura 2.3: Simple Regression Tree vs Gradient Boosted Trees

À esquerda temos o resultado obtido pelo algoritmos **Simple Regression Tree** e à direita os resultados obtidos pelo algoritmo **Gradient Boosted Trees**.

Como podemos ver, o algoritmo **Gradient Boosted Trees** obteve um melhor resultado em todas as métricas calculadas o que indica que é o melhor algoritmo para este problema. Iremos agora utilizar outras técnicas de modelação para tentar melhorar o resultado obtido.

2.4.2 Modelação com feature selection

Feature Selection é o processo de seleção de atributos mais relevantes para a construção de um modelo de previsão. No *KNIME* fomos capazes de usar esse processo através dos nodos *Feature Selection Loop Start*, *Feature Selection Loop End* e *Feature Selection Filter*. Esses nodos utilizam um modelo de previsão e fazem testes para descobrir quais são os melhores atributos a serem utilizados em modelos de previsão.

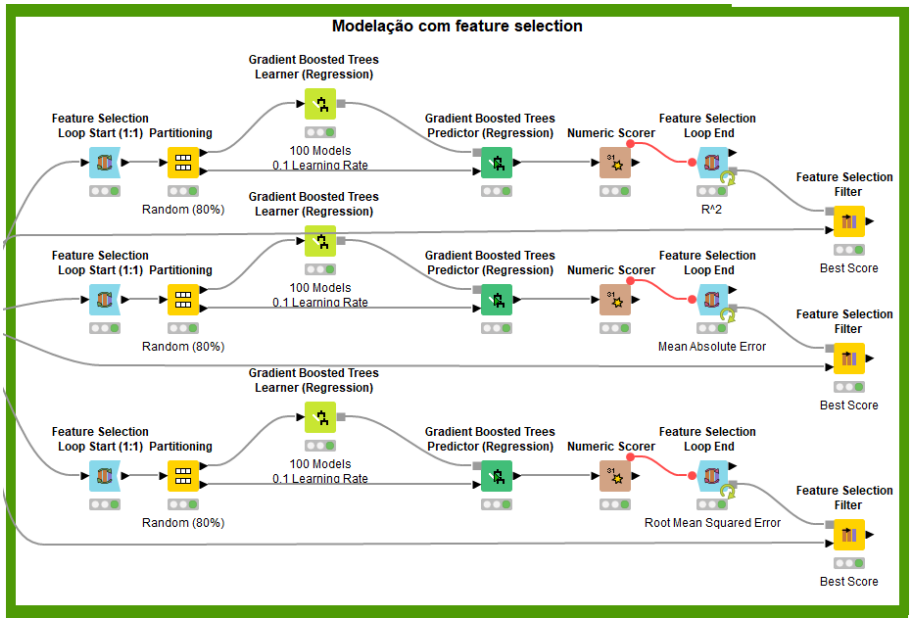

Figura 2.4: Modelação com Feature Selection

Como podemos ver na figura, foram realizados 3 processos de *feature selection* onde foram avaliadas 3 métricas diferentes, sendo elas o R^2, o *Mean Absolute Error* (**MAE**) e o *Root Mean Squared Error* (**RMSE**).

Nos processos onde as métricas de avaliação foram o R^2 e o *Root Mean Squared Error* (**RMSE**) os atributos que obtiveram um melhor resultado foram: **Aspiration**, **Carlength**, **Carwidth**, **EngineType**, **CylinderNumber**, **EngineSize** e **HorsePower** com um valor de 0,949 para a métrica R^2 e 2244,056 para a métrica **RMSE**, o que já são valores melhores relativamente aos da fase anterior onde eram utilizados todos os atributos. O resultado obtido com a métrica de avaliação *Mean Absolute Error* (**MAE**) foi de 1522,316 com os atributos **Fueltypes**, **Aspiration**, **Carlength**, **Carwidth**, **EngineType**, **CylinderNumber**, **EngineSize**, **Fuelsystem**, **Stroke**, **CompressionRatio** e **PeakRPM**.

Apesar de esses serem os melhores atributos obtidos através destes testes, é também importante denotar que outros atributos também obtiveram valores bastante próximos a estes o que demonstra que estes atributos não são definitivamente os melhores e em outras circunstâncias talvez outros atributos teriam um melhor desempenho.

2.4.3 Modelação com redes neuronais

As Redes Neuronais são um tipo de modelação inspirado no funcionamento de um cérebro humano. Elas são compostas por camadas de neurónios conectados através de sinapses que recebem informação, processam essa informação alterando o valor da sinapse por onde recebeu a informação e produzem uma saída que é transmitida para a próxima camada. Dessa forma as redes neuronais conseguem aprender a partir de exemplos onde identificam padrões e relações entre os atributos que recebem.

No nosso modelo foi utilizado o nodo **RProp MLP Learner** como algoritmo de rede neuronal. O algoritmo implementado por esse nodo é o **Multilayer Feedforward Networks** que é um algoritmo que não contém **loops** por ser *feedforward* e permite várias camadas de neurónios.

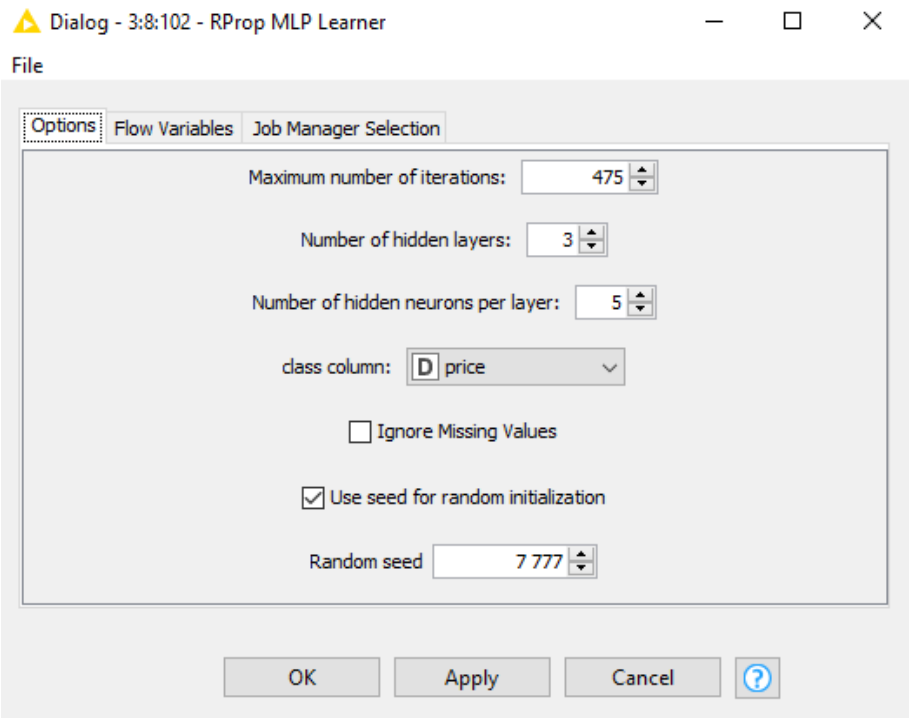

Figura 2.5: Configuração do nodo RProp MLP

Esta foi a configuração utilizada na modelação do problema onde temos um número máximo de iterações de 475, 3 camadas escondidas e 5 neurónios contidos em cada camada. Essa foi a melhor configuração encontrada e obtivemos os seguintes resultados.

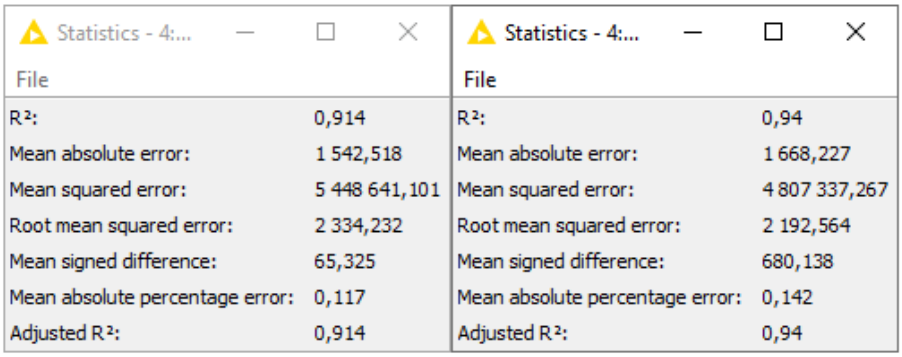

Figura 2.6: Gradient Boosted Trees vs Rede Neuronal

Os resultados obtidos pela rede neuronal demonstram algumas coisas interessantes quando comparadas aos resultados obtidos pelo algoritmo **Gradient Boosted Trees**. O valor de R^2 é maior na rede neuronal o que indica que os valores previstos pelo modelo estão mais próximos dos valores reais do que os valores obtidos pelo algoritmo **Gradient Boosted Trees**. No entanto quando analisámos os valores de **MAE** e **RMSE** conseguimos perceber que o algoritmo das redes neuronais possui menos erros de grande valor quando comparado com o algoritmo **Gradient Boosted Trees**, apesar de conter uma média de erros maior.

2.5 Avaliação

De acordo com o estudo que fizemos deste *dataset* somos capazes de concluir que, apesar de os atributos nele presentes necessitarem de pouco tratamento, uma vez que se encontravam relativamente ”limpos”, vimos que nem todos os atributos descrevem o preço final do veículo de forma adequada, como por exemplo os atributos **Stroke**, **CompressionRatio**, **CarHeight** e **Name**. Através da modelação fomos capazes de concluir que para este problema o melhor algoritmo para prever os preços dos carros é o algoritmo **Gradient Boosted Trees** pois é aquele que contém a menor média de erro no entanto em alguns casos este algoritmo faz umas previsões bastante longe do que se estaria à espera. Por isso devemos também ter em conta a modelação feita com redes neuronais pois essa modelação obteve valores mais próximos dos valores reais e com menos erros de grande escala.

3 Tarefa B: Classificação de Obesidade

Para a tarefa B foi-nos disponibilizado um *dataset* com a informação de várias pessoas, desde da sua idade, peso e altura até aos seus hábitos diários como o consumo de vegetais, a frequência de atividade física, entre outros.

A metodologia que será utilizada no processo da resolução do problema será novamente o **CRISP-DM** e seguiremos as mesmas etapas seguidas na tarefa A.

3.1 Estudo do negócio

O objetivo deste problema é criar um modelo para prever o nível de obesidade de uma pessoa a partir da informação obtida. Para isso temos disponível um *dataset* com informação sobre várias pessoas e o auxílio da ferramenta de análise de dados e *machine learning* **KNIME** que nos irá permitir a criação de modelos de aprendizagem para a resolução do problema.

Os objetivos a serem compridos neste problema são:

1. Analisar o *dataset* por completo.
2. Tratar de inconsistências no *dataset* e extrair conhecimento extra dos atributos se possível.
3. Criar gráficos, matrizes de correlação e outro tipos de visualização de dados.
4. Criar modelos de aprendizagem utilizando diferentes técnicas.

3.2 Estudo dos dados

O *dataset* contém 2111 linhas e 19 atributos. Os atributos são os seguintes:

1. **rowID**: Id do registo.
2. **Gender**: Sexo.
3. **Age**: Idade em Anos.
4. **Date_of_birth**: Data de nascimento (DD/MM/AAAA).
5. **Height**: Altura em metros (M).
6. **Weight**: Peso em kilos (Kg).
7. **family_history_with_overweight**: Histórico de obesidade na família.
8. **FAVC**: Frequente de consumo de comidas com muitas calorias (yes/no).
9. **FCVC**: Frequência de consumo de vegetais.
10. **NCP**: Número de refeições por dia.
11. **CAEC**: Consumo de comida entre as refeições.

12. **Smoke**: Se a pessoa fuma ou não.
13. **CH20**: Consumo de água durante o dia.
14. **SCC**: Monitorização do consumo de calorias.
15. **FAF**: Frequência de atividade física.
16. **TUE**: Tempo de uso de dispositivos eletrónicos.
17. **CALC**: Consumo de álcool.
18. **MTRANS**: Tipo de transporte usado.
19. **NObeyesdad**: Nível de obesidade.

O atributo **NObeyesdad** é aquele que iremos antever com o desenvolvimento de modelos de previsão através de várias técnicas de modelação.

De seguida vamos descrever a análise feita a cada atributo presente no *dataset* e algumas conclusões que tiramos com essa análise.

3.2.1 NObeyesdad

No *dataset* estudado os tipos de obesidade têm os seguintes valores: ***Insufficient_Weight***, ***Normal_Weight***, ***Overweight_Level_I***, ***Overweight_Level_II***, ***Obesity_Type_I***, ***Obesity_Type_II*** e ***Obesity_Type_III*** e como podemos ver pela imagem abaixo os valores estão razoavelmente bem distribuídos.

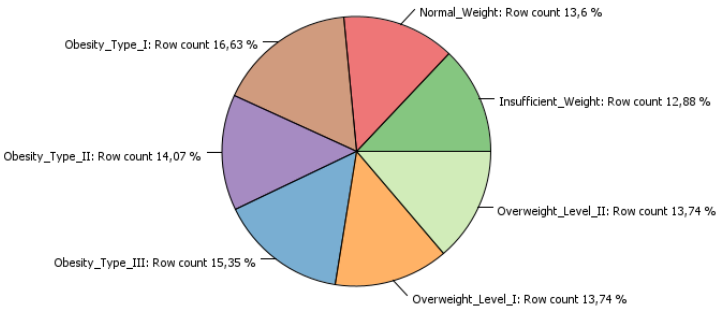

Figura 3.1: Distribuição dos tipos de obesidade

Para a análise dos atributos restantes foi utilizado o nodo **One to Many** para permitir relacionar cada atributo com o atributo objetivo, sendo este o ***NObeyesdad***.

3.2.2 Gender

No *dataset* existem 1068 Homens e 1043 Mulheres, ou seja, a quantidade de homens e mulheres está equilibrada. Na seguinte imagem somos capazes de reparar que os homens tendem estar mais acima do peso do que as mulheres.

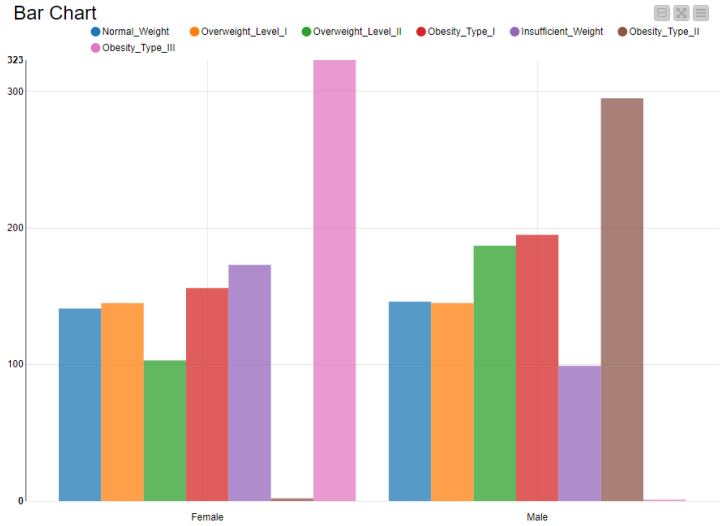

Figura 3.2: Distribuição dos sexos

3.2.3 Age

As idades no *dataset* variam entre 14 e 61 anos, porém existe uma maior distribuição entre os 18 e 26 anos. Através das cores no gráfico conseguimos ver o nível de obesidade para cada idade e com isso notamos que as pessoas com *Insufficient_Weight* tendem a ser as mais novas, entre as idades de 17 a 20 e a partir dos 26 anos grande parte das pessoas está acima do peso ideal.

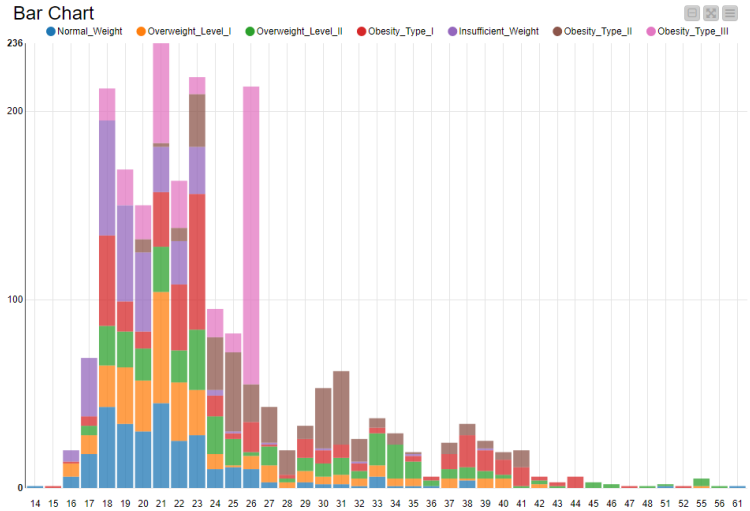

Figura 3.3: Distribuição das idades

3.2.4 Date_of_birth

Na data de nascimento das pessoas para facilitar a sua análise extraímos o ano que as pessoas nasceram e agrupamos de 10 em 10 anos. Com esse tratamento podemos ver que a maior parte das pessoas presentes no *dataset* nasceram entre 1986 e 2005. Podemos também ver que, como visto anteriormente na distribuição da idade, grande parte das pessoas com *Insufficient_Weight* são as pessoas mais novas nascidas entre 1996 e 2005.

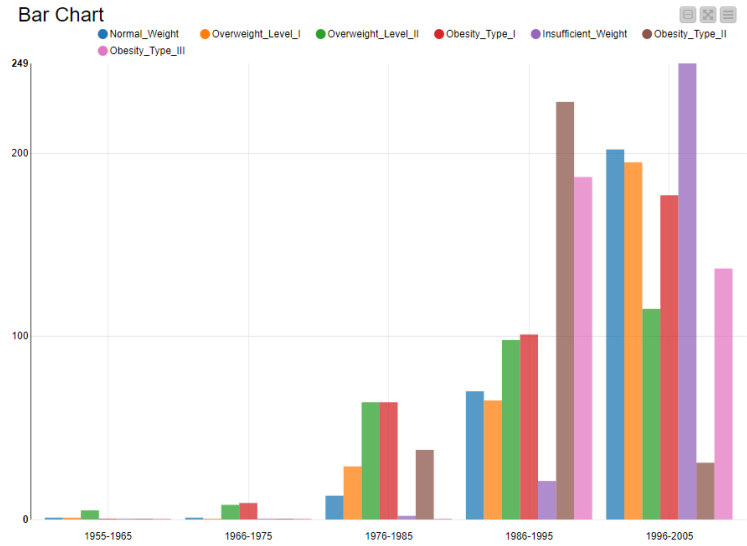

Figura 3.4: Anos de nascimento

3.2.5 Height & Weight

A altura mínima e máxima de uma pessoa neste *dataset* é de 1,45m e 1,98m respetivamente e a media é de 1,7017. Já o peso mínimo e máximo é de 39kg e 173kg respetivamente e a média é de 86,5861kg.

Column	Min	Mean	Median	Max	Std. Dev.	Skewness	Kurtosis	No. Missing	No. +∞	No. -∞
rowID	1	1 056	1 056	2 111	609,5375	0.0	-1,2	0	0	0
Height	1,45	1,7017	1,7005	1,98	0,0933	-0,0129	-0,5629	0	0	0
Weight	39	86,5861	83	173	26,1912	0,2554	-0,6999	0	0	0

Figura 3.5: Altura e peso

3.2.6 Family_history_with_overweight

No *dataset* 81,76% das pessoas têm um histórico de obesidade na sua família e as pessoas com um histórico de obesidade na sua família tendem a ter um nível de obesidade acima do normal.

3.2.7 FAVC

De todas as pessoas presentes do *dataset*, 88,39% consomem comidas com muitas calorias frequentemente, logo este atributo não possui uma boa distribuição.

3.2.8 FCVC

Neste *dataset* a frequência de consumo de vegetais têm os possíveis valores, *never* com 9,57% ocorrências, *sometimes* com 59,55% e *always* 30,89%. Algo um pouco inesperado é o facto que todas as pessoas com *Obesity_Type_III* consomem vegetais com muita regularidade.

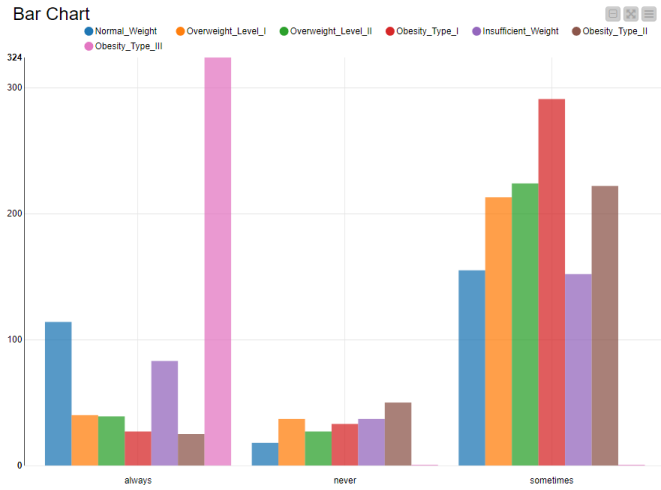

Figura 3.6: Frequência de consumo de vegetais

3.2.9 NPC

A distribuição do número de refeições feitas pelas pessoas é: 1 refeição 14,97%, 2 refeições 8,34%, 3 refeições 69,64% e 4 refeições 7,06%. Das pessoas que comem 4 refeições por dia, grande parte está na categoria *Insufficient_Weight*.

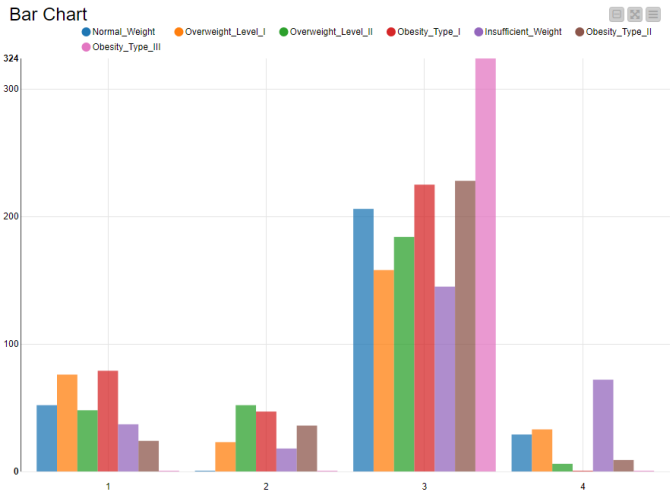

Figura 3.7: Número de refeições consumidas por dia

3.2.10 CAEC

A frequência de consumo de comida entre as refeições é: *No* 2,42%, *Sometimes* 83,61%, *Frequently* 11,46% e *Always* 2,51%. As pessoas que estão na categoria *Frequently* tendem a estar no seu peso normal (*Normal_Weight*) ou abaixo (*Insufficient_Weight*).

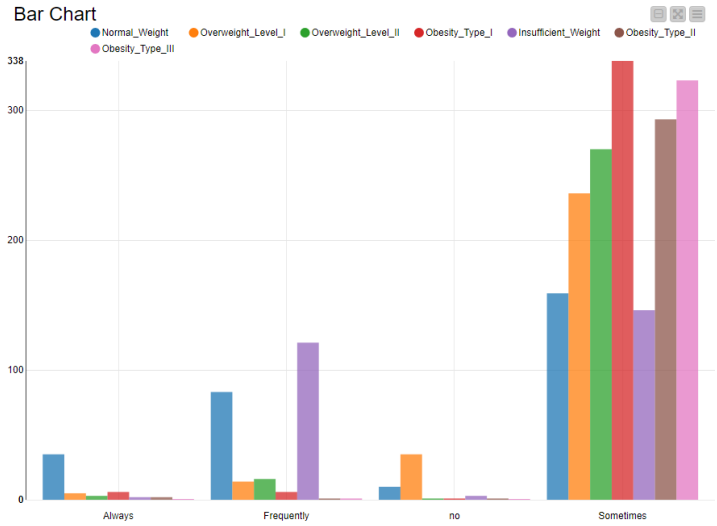

Figura 3.8: Consumo de comida entre as refeições

3.2.11 SMOKE

O número de pessoas que não fuma 97,92% é muito superior ao número de pessoas que fuma 2,08%, por isso não é possível tirar conclusões sobre o efeito de fumar com nível de obesidade da pessoa.

3.2.12 CH2O

A distribuição do consumo de água por dia é: 1 (<1L) 22,97%, 2 (1L - 2L) 52,58% e 3 (>2L) 24,44%. As pessoas que bebem mais água (>2L) estão mais propensas a ter um nível de obesidade elevado.

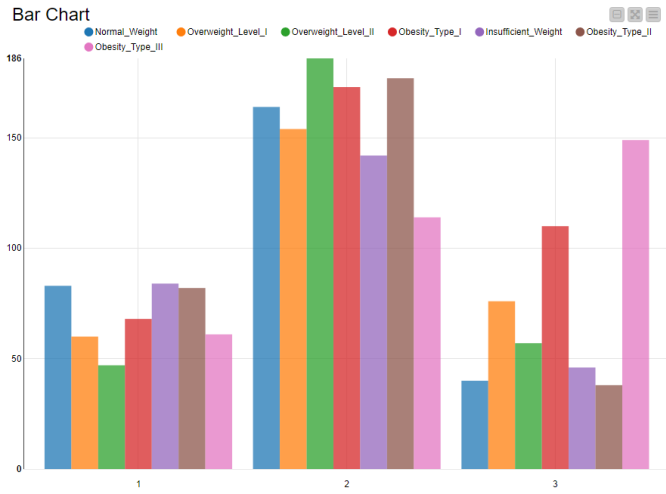

Figura 3.9: Consumo diário de água

3.2.13 SCC

Cerca de 95,45% das pessoas não monitoriza o seu consumo de calorias. Apenas 4,55% monitoriza as suas calorias e essas pessoas não exibem níveis de obesidade elevados, no entanto como o número de casos é baixo, não dá para tirar uma conclusão muito assertiva.

3.2.14 FAF

A distribuição de atividade física por semana é: 0 (nunca) 34,11%, 1 (1-2 dias) 36,76%, 2 (2-4 dias) 23,5% e 3 (4-5 dias) 5,64%. As pessoas que praticam menos exercício físico tendem a estar acima do seu peso ideal.

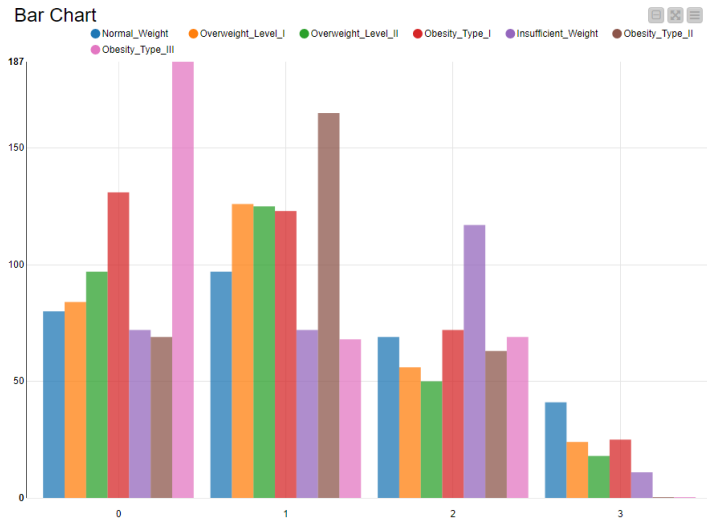

Figura 3.10: Frequência de atividade física

3.2.15 TUE

A distribuição de utilização de aparelhos tecnológicos por dia é: 0 (0-2 horas) 45,1%, 1 (3-5 horas) 43,34% e 2 (>5 horas) 11,56%. Através do gráfico de distribuições com os diferentes tipos de obesidade não conseguimos tirar conclusões sobre o efeito dos aparelhos tecnológicos nas pessoas.

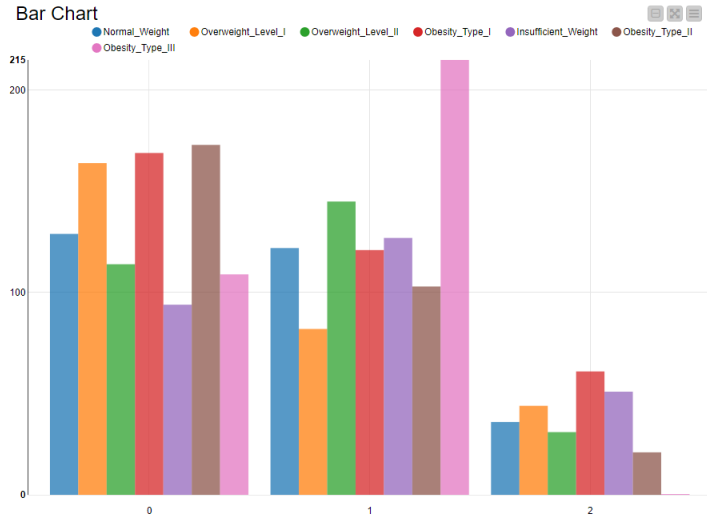

Figura 3.11: Tempo de uso de dispositivos tecnológicos

3.2.16 CALC

A frequência de consumo de álcool pelas pessoas é: *No* 30,27%, *Sometimes* 66,37%, *Frequently* 3,32% e *Always* 0,047%. Devido à fraca distribuição deste atributo no *dataset* não conseguimos tirar boas conclusões sobre o efeito do álcool no nível de obesidade das pessoas.

3.2.17 MTRANS

O método de transporte utilizado pelas pessoas são: *Public_Transprotation* 74,85%, *Automobile* 21,65%, *Walking* 2,65%, *Motorbike* 0,52%.e *Bike* 0,33%. A grande maioria utiliza transportes públicos ou automóveis individuais como modo de transporte o que dificulta a análise entre os diferentes modos de transporte.

3.3 Preparação dos dados

Na fase de preparação dos dados começamos por analisar os atributos do *dataset* e fazer as alterações que fossem necessárias para que o valor dos atributos estivesse de acordo com o que era esperado.

O tratamento de dados foi necessário para os seguintes atributos:

- **Age**: Em certas linhas do *dataset* o valor da idade tinha casas decimais, o que não faz sentido visto que o atributo em questão é a idade de uma pessoa, por isso, convertemos o valor de *string* para *double*, e de seguida para *int* arredondando o valor.
- **NCP, CH20, FAF, TUE**: São atributos que têm valores inteiros entre uma certa gama de valores, por exemplo o **NCP** só poderá ter os valores 1,2,3 ou 4, e em algumas linhas do *dataset* o valor para essas colunas apresentavam valores decimais, para isso fizemos o mesmo tratamento que foi feito para a idade por forma a obter valores válidos para esses campos.

Age	NCP	CH2O	FAF	TUE
17,486869	3,821168	2	2	0,556245
19,920629	3,897078	2,622638	2	1,894105
18,426619	3,092116	2,426465	2	1,839862
17,082867	3	2,911187	2,595128	1,380204
17,000433	3	2,310921	2,240714	1,59257
16,270434	3,286431	2,148146	2,458237	1,273333
17,908114	4	2	2	0,220029
17,120699	4	2	2	0,03838
19,329542	4	2,157395	2	1,679149
22,377998	3	2	0,139808	0,875464

Figura 3.12: Valores antes do tratamento

- **Height, Weight**: Os atributos de altura e peso foram convertidos de *string* para *double*.
- **Date_of_birth**: O atributo **Date_of_birth** que estava em formato *string* da seguinte forma (dd/MM/yyyy H:mm) foi convertido para *Date&Time*. Foi também criado um novo atributo **Year** que representa o ano de nascimento de cada pessoa. Esse atributo foi utilizado depois na modelação como substituto do atributo **Date_of_birth**.
- **FCVC**: Este atributo é uma *string* com os seguintes possíveis valores (never, sometimes, always), no entanto o *dataset* possuía linhas com valores incorretos mas com um formato uniformizado. Esse formato era ”(Valor válido),(valores aleatórios)”, por exemplo ”sometimes,9844sometimes5”. Depois

de uma análise completa ao *dataset* o grupo percebeu que tinha 2 escolhas, ou removia as linhas onde esse valor estava no formato errado, ou assumia o valor que estava antes da vírgula e removia o resto da *string*. No final decidimos assumir o valor antes da vírgula para não perder esses dados do *dataset*.

- **Gender, Smoke, CAEC, CALC**: Estes atributos possuíam alguns valores que não estavam de acordo com a norma do atributo. O atributo **Gender** possuía os valores (Male, Female, Man, Woman) e para normalizar o atributo convertemos os valores ”Man” e ”Woman” para ”Male” e ”Female” respetivamente através do nodo **Rule Engine**. Para os outros atributos foi feito um tratamento semelhante onde alteramos os valores para ficarem de acordo com o resto dos valores.

Para realizar uma modelação com todos os atributos agrupados, decidimos criar *Bins* em alguns atributos do *dataset*. Os atributos agrupados foram:

- **Age**: Idade dividida em 3 *Bins*: Jovem(14-22), Jovem Adulto(23-29) e Adulto(29-61)
- **Year**: Anos de nascimento agrupado de 10 em 10 anos, deste 1955 até 2005.
- **Height**: Altura dividida 5 *Bins*: Baixo(1,45-1,60), Médio(1,60-1,70), Alto(1,70-1,80), Muito-Alto(1,80-1,98)
- **Weight**: Peso das pessoas foram agrupadas de 20 em 20 kilos a começar em 30Kg e terminar em 180Kg.

Estes atributos agrupados irão ser utilizados na modelação com *bins* que será feita na próxima fase e aí veremos se compensa fazer esse agrupamento ou não.

Outra preparação de dados feita no *dataset* foi a conversão do atributo objetivo (**NObeyesdad**) para um valor numérico por forma a ser utilizado na modelação do problema com regressão. Para isso utilizados o nodo **Category to Number** para converter os tipos de obesidade em número e um **Rule engine** para ordenar os tipos de menos obeso até o mais obeso. A ordem escolhida foi a seguinte:

1. *Insufficient_Weight*.
2. *Normal_Weight*.
3. *Overweight_Level_I*.
4. *Overweight_Level_II*.
5. *Obesity_Type_I*.
6. *Obesity_Type_II*.
7. *Obesity_Type_III*.

A ultima preparação de dados feita no *dataset* foi a conversão dos atributos nominais em atributos numéricos mais uma vez utilizando o nodo **Category to Number**. A razão para esta conversão de todos os dados em números deve-se à modelação com *clustering* que será feita na fase de modelação. Com todos os atributos convertidos em valores numéricos os algoritmos de *clustering* terão mais informação sobre o problema pois esses algoritmos apenas conseguem interpretar esses valores.

3.4 Modelação

3.4.1 Modelação com dados normalizados

A criação dos modelos de *Machine Learning* começou com um modelo simples onde utilizamos apenas os dados tratados na fase de tratamento de dados.

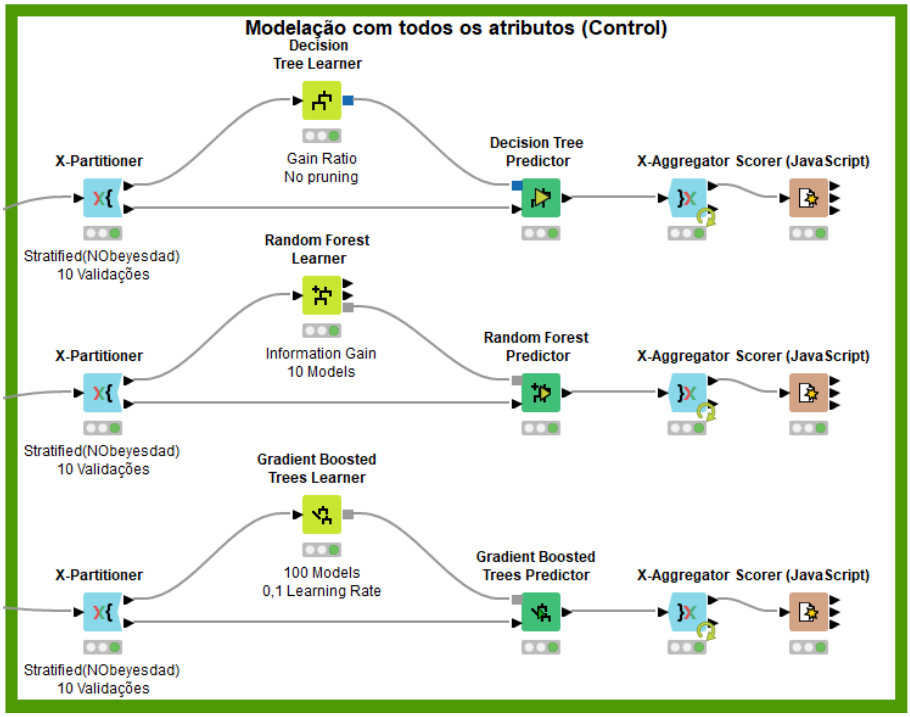

Figura 3.13: Modelação com dados normais

Como o *dataset* apresentava um problema de classificação, utilizamos nodos de árvores de decisão com os algoritmos de aprendizagem *Decision Tree*, *Random Forest* e *Gradient Boosting*. Foram utilizados também nodos de *Cross Validation* para correr os algoritmos de aprendizagem várias vezes, isto permite obter resultados mais fiéis do que conseguiríamos com um *Partitioning* básico. No nodo *X-Partitioner* foi utilizada a opção *Stratified sampling* na coluna objetivo *NObeyesdad* para garantir as mesmas proporções de cada tipo de obesidade.

Scorer View

Confusion Matrix

	Insufficie...	Normal_...	Obesity_T...	Obesity_T...	Obesity_T...	Overweig...	Overweig...	
Insufficie...	264	8	0	0	0	0	0	97.06%
Normal_...	4	271	0	0	0	12	0	94.43%
Obesity_T...	0	0	343	2	0	2	4	97.72%
Obesity_T...	0	0	7	290	0	0	0	97.64%
Obesity_T...	0	0	0	2	322	0	0	99.38%
Overweig...	0	14	0	0	0	268	8	92.41%
Overweig...	0	0	3	0	0	4	283	97.59%
	98.51%	92.49%	97.17%	98.64%	100.00%	93.71%	95.93%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
96.68%	3.32%	0.961	2041	70

Figura 3.14: Scorer Modelação Normal (Gradient Boosting)

Este foi o resultado obtido pelo algoritmo *Gradient Boosting*. Entre os 3 algoritmos este foi o que obteve uma *accuracy* melhor de 96,68%. O valor de *Cohen's kappa* está próximo do ideal com um valor de 0,961. Um dos fatores que causa esse valor é a distribuição dos tipos de obesidade estar relativamente equilibrada, porque se os dados estivessem enviesados para um tipo de obesidade era mais provável que esse valor fosse menor.

3.4.2 Modelação com dados *Binned*

Com estes novos atributos fizemos testes com os 3 algoritmos de aprendizagem utilizados anteriormente obtivemos os seguintes resultados:

Scorer View

Confusion Matrix

	Insufficie...	Normal_...	Obesity_T...	Obesity_T...	Obesity_T...	Overweig...	Overweig...	
Insufficie...	252	19	0	0	0	1	0	92.65%
Normal_...	24	226	0	0	0	21	16	78.75%
Obesity_T...	0	2	331	2	0	3	13	94.30%
Obesity_T...	0	1	4	291	1	0	0	97.98%
Obesity_T...	0	1	0	0	323	0	0	99.69%
Overweig...	0	35	5	0	0	231	19	79.66%
Overweig...	0	12	15	0	0	15	248	85.52%
	91.30%	76.35%	93.24%	99.32%	99.69%	85.24%	83.78%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
90.10%	9.90%	0.884	1902	209

Figura 3.15: Scorer da Modelação Binned (Random Forest)

Como podemos ver pela imagem acima, esta abordagem para o nosso problema resultou em uma *accuracy* pior do que a obtida anteriormente.

3.4.3 Modelação com Feature Selection

Para descobrir quais são os melhores atributos a serem usados no modelo de previsão recorremos à estratégia de **Feature Selection** que permite através de testes encontrar os atributos que devolvem a melhor *accuracy* no modelo.

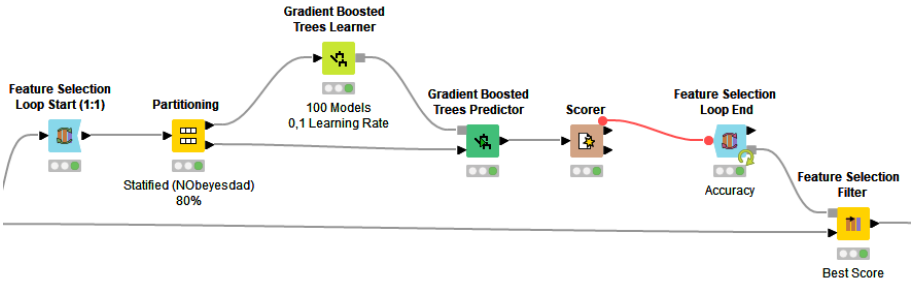

Figura 3.16: Nodos de Feature Selection

Os atributos que foram excluídos pelo processo de *Feature Selection* foram, **Age**, **FAVC**, **CH2O**, **FAF** e **MTRANS**. Ao correr os modelos com apenas os atributos selecionados pelo o processo anterior foi obtido o seguinte resultado pelo o algoritmo *Gradient Boosting*.

Scorer View

Confusion Matrix

	Insufficie...	Normal_...	Obesity_T...	Obesity_T...	Obesity_T...	Overweig...	Overweig...	
Insufficie...	264	8	0	0	0	0	0	97.06%
Normal_...	2	277	0	0	0	8	0	96.52%
Obesity_T...	0	0	344	3	0	1	3	98.01%
Obesity_T...	0	0	7	290	0	0	0	97.64%
Obesity_T...	0	0	0	2	322	0	0	99.38%
Overweig...	0	13	0	0	0	272	5	93.79%
Overweig...	0	0	3	0	0	6	281	96.90%
	99.25%	92.95%	97.18%	98.31%	100.00%	94.77%	97.23%	

Overall Statistics

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
97.11%	2.89%	0.966	2050	61

Figura 3.17: Scorer da Modelação com Feature Selection (Gradient Boosting)

A *accuracy* através deste método subiu para 97,11% em relação ao melhor modelo até agora explorado que tinha uma *accuracy* de 96,68%.

3.4.4 Modelação com o atributo objetivo contínuo (Regressão)

Para realizar a modelação deste problema como um problema de regressão recorremos à conversão do atributo objetivo para um valor numérico e à utilização de 2 algoritmos de resolução de problemas de regressão. Esses algoritmos foram o **Simple Regression Tree** e o **Gradient Boosted Trees (Regression)**.

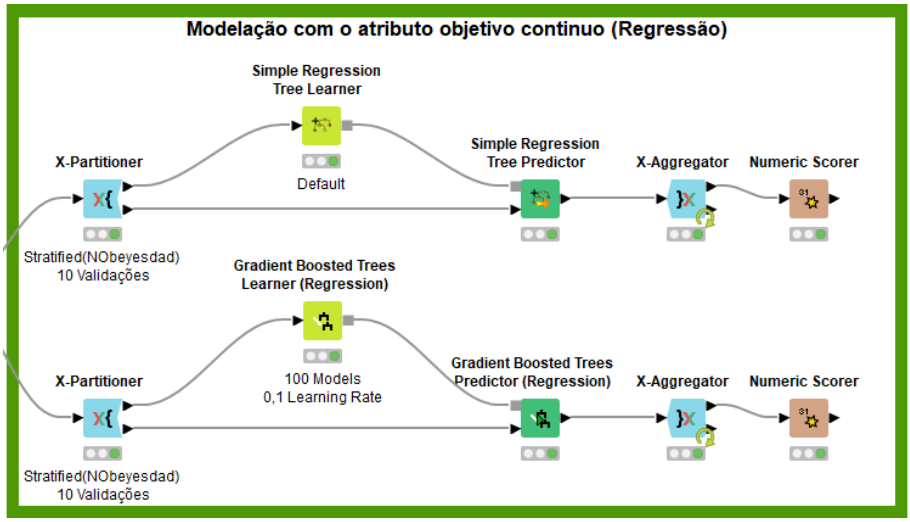

Figura 3.18: Modelação com o atributo objetivo continuo (Regressão)

Para esta modelação foi também utilizado *Cross Validation* com *Stratified Sampling* sobre a coluna objetivo por forma a garantir proporções equilibradas de níveis de obesidade.

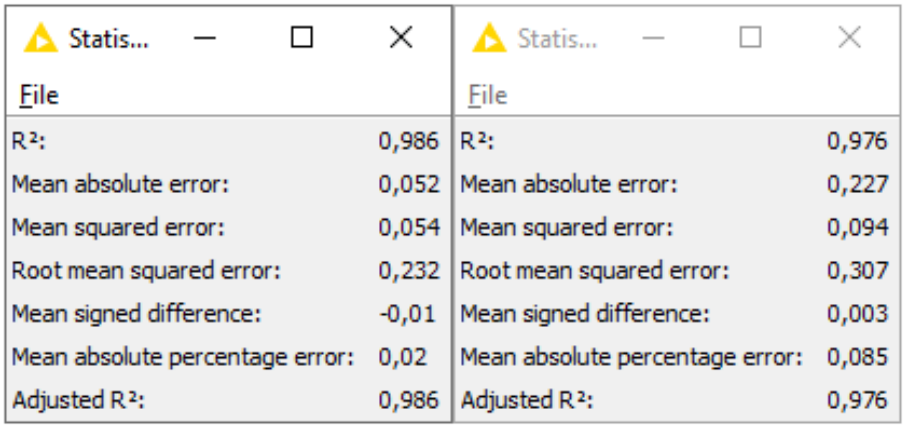

Figura 3.19: Simple Regression vs Gradient Boosted Trees (Regression)

Estes foram os resultados obtidos onde à esquerda podemos ver o algoritmo **Simple Regression Tree** e à direita o algoritmo **Gradient Boosted Trees (Regression)**. Está claro que com estes resultados o melhor algoritmo para este problema é o **Simple Regression Tree** pois obteve os melhores valores em todas as métricas calculadas pelo **Numeric Scorer**.

3.4.5 Modelação com *Clustering*

Para finalizar a modelação deste problema decidimos ver como uma estratégia de *clustering* se comportaria com este *dataset*.

O nodo utilizado neste modelo foi o **SOTA Learner** porque foi o nodo de *clustering* onde obtivemos os melhores resultados.

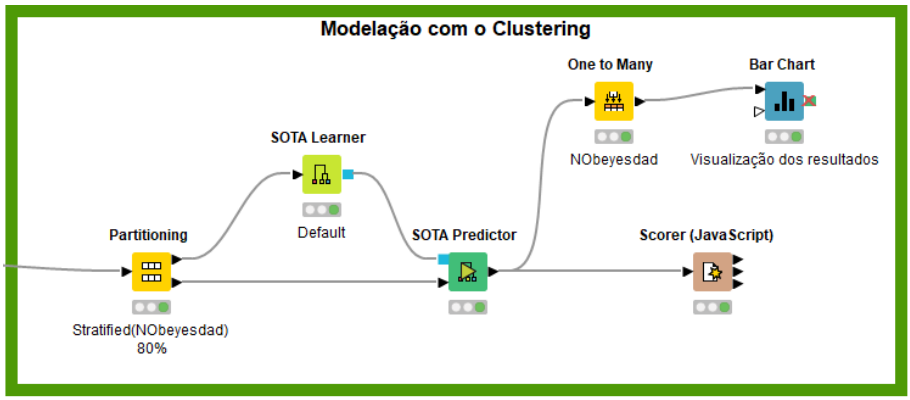

Figura 3.20: Modelação com *clustering* (SOTA)

Para este caso também analisamos os resultados obtidos de forma gráfica através dos nodos **One to Many** e **Bar Chart**. Este foi o gráfico gerado.

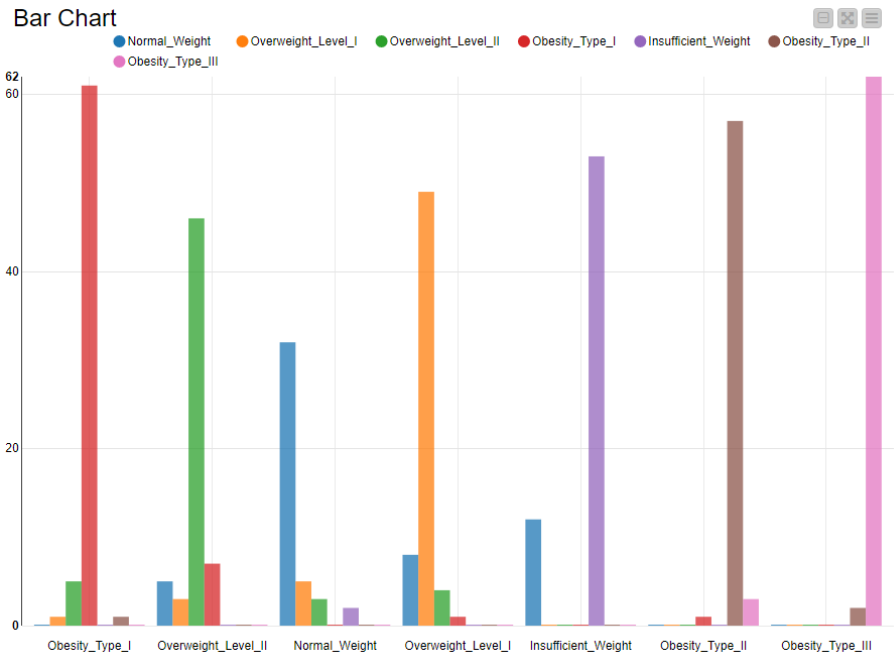

Figura 3.21: Gráfico dos resultados (Clustering)

Como podemos ver pelo gráfico os resultados obtidos pelo modelo não foram perfeitos tendo apenas obtido uma **accuracy** de **85,11%**. No entanto pelo gráfico conseguimos perceber que quando o modelo errou no tipo de obesidade a sua previsão esteve em um nível próximo do esperado o que indica que a precisão do modelo está razoavelmente boa.

3.5 Avaliação

Através do estudo que realizamos deste *dataset* somos capazes de concluir que os dados nele presentes, apesar de terem necessitado de um tratamento considerável, permitem que sejamos capazes de efetuar uma boa análise do problema.

Na fase da modelação utilizamos diferentes estratégias para descobrir qual seria a melhor para criar um modelo de previsão e chegamos à conclusão que a utilização de algoritmos de árvores de decisão sobre todos os atributos do *dataset* apresentava bons resultados. No entanto, através da técnica de **Feature Selection**, conseguimos encontrar os melhores atributos para obter um resultado ainda melhor. A estratégia de agrupamento dos atributos numéricos (*binning*) acabou por se revelar pior do que a original e com isso podemos ver que essa estratégia para este problema não foi bem sucedida. De seguida resolvemos modificar os dados para que o problema se tornasse em um problema de regressão. Depois procedemos ao uso de algoritmos de regressão e chegamos à conclusão que também é uma boa forma de resolver o problema pois obtemos resultados favoráveis com este tipo de modelação. E para finalizar a modelação deste problema resolvemos explorar a estratégia de *clustering* onde o objetivo é encontrar padrões nos dados por forma a agrupar-los. Neste problema os resultados não foram os melhores quando comparados as outras estratégias exploradas.

Para finalizar gostaríamos então dizer que o melhor algoritmo para resolver este problema foi o **Gradient Boosted Trees** pois foi aquele que obteve o melhor resultado e quase todos os atributos presentes no *dataset* eram bons o suficiente para serem utilizados como atributos de aprendizagem, com a exceção dos atributos **Age**, **FAVC**, **CH2O**, **FAF** e **MTRANS** que foram os atributos que o processo de **Feature Selection** excluiu.

3.6 Conclusão

Com a conclusão deste trabalho prático, conseguimos dizer que fomos capazes de aplicar vários conceitos associados ao desenvolvimento de modelos de aprendizagem abordados ao longo do semestre e outros conceitos não abordados como a estratégia de *Feature Selection*.

Nos *datasets* que exploramos durante o desenvolvimento deste projeto, para além da criação dos modelos de aprendizagem, efetuamos também a exploração dos dados e pré-processamento dos dados por forma a entender melhor o problema que estávamos a estudar e dessa forma arranjar as melhores ferramentas que dessem as respostas mais satisfatórias ao problema.

Apesar de no início haver alguns problemas relativamente à escolha do *dataset* para a Tarefa A, estamos satisfeitos com o projeto que realizámos, pois conseguimos desenvolver modelos de aprendizagem satisfatórios para os *datasets* estudados e por termos feito uma boa documentação de todo os processo do desenvolvimento do projeto.