UNIVERSIDADE D
COIMBRA

Cláudia Filipa Gaspar da Costa

# EXPLORAÇÃO DE DADOS EM FALTA: UMA ABORDAGEM VISUAL

Dissertação de Mestrado Integrado de Engenharia Biomédica,
orientada pelo Professor Doutor Pedro Henriques Abreu
e apresentada ao Departamento de Física da Faculdade de Ciências e tecnologia
da Universidade d Coimbra.

Setembro de 2018

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Cláudia Filipa Gaspar da Costa

# Exploração de dados em falta: Uma abordagem visual

Tese submetida à
Universidade de Coimbra para o grau de
Mestre em Engenharia Biomédica

Orientadores:
Pedro Henriques Abreu (PhD)
Miriam Seoane Santos (MSc)

**Coimbra, 2018**

This work was developped in collaboration with:

**Centro de Informática e Sistemas da Universidade de Coimbra**

# Agradecimentos

Na realização da presente dissertação, e porque contei com o apoio de múltiplas pessoas, gostaria de deixar aqui registados os meus mais sinceros agradecimentos.

Em primeiro lugar, quero expressar o meu sincero agradecimento e gratidão ao meu orientador, o Professor Doutor Pedro Henrique Abreu, e à Miriam Seoane Santos por todo o apoio e disponibilidade que demonstraram durante a minha tese.

Ao Professor Doutor Pedro Henrique Abreu, pela partilha de conhecimento e experiência científica, bem como pelo seu incentivo constante e pela sua honestidade.

À Miriam Seoane Santos, por toda a ajuda, paciência e disponibilidade durante este trabalho e pela sua humildade e carinho. A ela desejo a maior sorte e força para esta jornada do doutoramento.

Exprimo a minha gratidão aos meus colegas e família que me acompanharam durante o meu percurso académico e um especial agradecimento ao meu grupo de amigos do curso, por todos os momentos que passámos, que por terem sido tão diferentes e tão genuínos levarei comigo para a vida.

Finalmente, um agradecimento especial, à minha mãe. O maior pilar na minha vida, pois sem ela não teria a possibilidade de realizar os meus sonhos. Por ser o meu modelo a seguir e pelo apoio incondicional em todos os momentos da minha vida.

# Resumo

Os dados em falta são um problema comum na análise de dados e podem ocorrer devido a múltiplos fatores, tais como falha de sensores, resultados de análises perdidos ou amostras contaminadas.

Ao longo dos anos, foram utilizadas diversas estratégias para resolver este problema, sendo as mais comuns a eliminação dos casos com valores em falta e sua substituição por valores estimados de acordo com os restantes dados - imputação. Todavia, para se escolher a técnica mais adequada para lidar com os dados em falta é fundamental compreender as suas caraterísticas intrínsecas, tais como a sua distribuição e o tipo de mecanismo em causa.

Estudos recentes indicam que a exploração de dados em falta pode ser realizada através de métodos de visualização, afirmando que esta técnica permite uma análise aprofundada, orientando assim a escolha apropriada de um método de imputação.

O objetivo principal desta tese prende-se com o desenvolvimento de uma ferramenta intuitiva e transparente que permita a exploração dos dados em falta através de métodos de visualização, auxiliando o utilizador na escolha do método de imputação mais apropriado. A ferramenta foi desenhada tendo em mente dois perfis distintos de utilizador (iniciantes e especialistas da área) e a capacidade para lidar com conjuntos de dados de grande dimensionalidade. As diversas funcionalidades da ferramenta foram validades através da realização de um caso de estudo.

**Palavras chave:** Dados em falta, imputação de dados, visualização de dados, gráficos interativos, exploração de dados.

# Abstract

Missing data is a common problem in data analysis and can occur due to many factors, such as sensor failure, lost analysis results or contaminated samples.

Through the years, different strategies have been used to solve this problem, the most common being the elimination of cases with missing values and their substitution according to the remaining data - imputation. However, to choose the most adequate technique to handle missing data it is fundamental to understand its intrinsic characteristics, such as data distribution and missing mechanism.

Recent studies indicate that the missing data exploration can be performed through visualization methods, positively reinforcing that this technique allows a profound analysis, supporting the choice of an appropriate imputation method.

The main focus of this thesis is to develop an intuitive and transparent tool that allows missing data exploration through visualization methods and helps the user to choose the most appropriate imputation method . The tool was designed bearing in mind two distinct user profiles (beginners and experts) and ability to handle high-dimensional data. The several features included in tool were validated through a case study.

**Keywords:** Missing data, data imputation, missing values, data visualization, interactive graphics, exploratory analysis.

# Lista de Abreviaturas

**AWT** *Abstract Window Toolkit*.

**DT** *Decision Tree*.

**GUI** *Graphical User Interface*.

**kNN** *k-Nearest Neighbors*.

**M-M** Média-Moda.
**MANET** *Missing Now Equally Treated*.
**MAR** *Missing At Random*.
**MCAR** *Missing Completely At Random*.
**MD** *Missing Data*.
**ML** *Machine Learning*.
**MLP** *Multiple Layer Perceptron*.
**MNAR** *Missing Not At Random*.

**NRMSE** *Normalized Root Mean Squared Error*.

**PCA** *Principal Component Analysis*.
**PFC** *Proportion of Falsely Classified Entries*.

**RMSE** *Root Mean Squared Error*.

**SVM** *Support Vector Machine*.

**VIM** *Visualization and Imputation of Missing Values*.

# Lista de Figuras

# Lista de Tabelas

# Conteúdo

# 1

# Introdução

No contexto da análise de dados, os dados em falta, em inglês, *Missing Data* (MD), referem-se a observações que, por algum motivo, estão ausentes. Os MD podem ocorrer em diversos domínios: por exemplo, na área da saúde, onde podem ocorrer devido a falha de sensores, a resultados de análises perdidos ou até mesmo ao facto de um doente não preencher todas as opções de um questionário médico [1].

O problema da falta de dados pode ter consequências prejudiciais [2, 3]. Em primeiro lugar, pode influenciar significativamente o resultado de estudos de investigação [4]. Em segundo lugar, realizar uma análise estatística usando apenas casos completos e ignorando os casos com valores em falta pode reduzir significativamente o tamanho da amostra, prejudicando substancialmente a precisão das estimativas retiradas do estudo [3]. Finalmente, muitos dos algoritmos e técnicas estatísticas são geralmente adaptados para extrair inferências de conjuntos de dados (*datasets*) completos [5]. Pode ser difícil ou mesmo inapropriado aplicar esses algoritmos e técnicas estatísticas em *datasets* incompletos [6].

Ao longo dos anos foram utilizadas diversas estratégias para lidar com MD [7]. Dos diversos tipos de abordagens para lidar com MD, as mais comuns são a eliminação dos casos e a imputação [5]. A eliminação dos casos é a estratégia mais simples onde são removidas quaisquer observações onde existam dados em falta [5]. No entanto, esta estratégia pode levar a elevadas perdas de informação relevante. Diversos estudos indicam que os métodos de imputação, que substituem os valores em falta por valores estimados de acordo com os dados completos, têm um grande contributo no aumento da qualidade da classificação comparativamente aos métodos de eliminação de casos [8]. Geralmente, os métodos de imputação são divididos em imputação baseada em técnicas de aprendizagem automática (*Machine Learning* (ML)) ou em análise estatística [8, 9], como será detalhado na Secção 2.2. Acresce salientar que quase todos estes métodos são baseados em suposições sobre determinadas caraterísticas dos dados, como a sua distribuição, tipo de variáveis, padrão

e mecanismo de MD, entre outras [10, 11, 12]. Deste modo, a escolha da técnica mais apropriada para lidar com os dados em falta num contexto específico deve ser ajustada às suas características [13]. Escolher o melhor método para contornar o problema de MD é fundamental em todas as áreas, mas pode ser crucial em certas áreas, nomeadamente na área da saúde, onde os valores em falta podem ter grande impacto, considerando o que está em causa, um tratamento errado dos dados em falta pode ter graves consequências na decisão clínica [14].

Recentemente, a exploração dos dados em falta através de métodos visuais tem ganho a atenção da comunidade científica como uma estratégia de análise das suas propriedades e estrutura, orientando assim a escolha apropriada de um método de imputação [15, 16, 17, 18].

## 1.1 Contexto e Motivação

Atualmente, o uso de testes estatísticos é a principal estratégia utilizada no estudo das características intrínsecas aos dados em falta. Todavia a exploração visual dos dados em falta tem a vantagem, relativamente aos testes estatísticos, de não depender de suposições de distribuições dos dados, o que permite a análise de uma maior diversidade de *datasets* [19]. Além disso, a visualização facilita a análise de MD por pessoas não especializadas na área. Todavia, a maioria do software de análise estatística (e.g. SAS, SPSS, STATA e WEKA) não inclui módulos especializados para a visualização de MD [17]. Tendo em vista que um dos objetivos finais da exploração dos valores em falta é a escolha da melhor técnica para lidar com os mesmos, é fundamental que a ferramenta inclua um módulo de imputação de dados em falta. Deste modo, a ferramenta permitirá a exploração visual dos dados em falta, a sua imputação e uma avaliação exploratória do seu desempenho.

## 1.2 Objetivos

O principal objetivo é desenvolver uma ferramenta que permita, de forma transparente, intuitiva e acessível tanto a pessoas especializadas na área como a iniciantes, a exploração visual das características dos dados em falta. Os iniciantes podem explorar as aspetos básicos dos dados, tais como a sua distribuição e a relação entre as variáveis e os dados em falta, enquanto que os especialistas podem fazer análises mais complexas, como procurar padrões de falta de dados. A ferramenta deverá

ainda incluir uma componente de imputação, apesar de não ser o foco principal do trabalho. Resumidamente, a ferramenta deverá conter funcionalidades que permitam ao utilizador:

- Visualizar os dados incompletos através de gráficos interativos, tabelas e sumário das variáveis;
- Utilizar métodos de imputação baseados em análise estatística e técnicas ML;
- Visualizar os dados imputados através de: gráficos interativos;
- Aceder a componentes avançadas onde poderá criar automaticamente um relatório com um sumário estatístico e gráfico dos *datasets* originais e imputados, alterar os parâmetros dos métodos de imputação, guardar gráficos gerados e armazenar *datasets* imputados.

## 1.3 Estrutura do Documento

Este tese está organizada em 6 capítulos: no Capítulo 2 são descritos alguns fundamentos teóricos essenciais à fundamentação do trabalho desenvolvido, enquanto que no Capítulo 3 é feita uma revisão dos trabalhos relacionados que exploram i) abordagens visuais para a caracterização de dados em falta e ii) estratégias de imputação de dados em falta. No Capítulo 4 é explicada a arquitetura e implementação da ferramenta, enquanto que no Capítulo 5 são ilustradas as principais funcionalidades da ferramenta. Por fim, no Capítulo 6 são discutidas as principais conclusões e trabalho futuro a ser desenvolvido.

# 2

# Fundamentos Teóricos

Neste capítulo são descritos alguns fundamentos teóricos essenciais à fundamentação do trabalho desenvolvido.

## 2.1 Teoria de Dados em Falta

O correto conhecimento da origem dos dados em falta nos *datasets* é essencial para uma escolha acertada da estratégia para lidar com eles, de forma a evitar distorção dos dados. Rubin et al. [20] classificaram formalmente três mecanismos para explicar os dados em falta: *Missing Completely At Random* (MCAR), *Missing At Random* (MAR) e *Missing Not At Random* (MNAR).

Para ilustrá-los, considere o pequeno *dataset* na Tabela 2.1, adaptado de [21], que representa a avaliação dos alunos num exame (0 a 100%) e sua nota final obtida no curso no final do semestre (0 a 20).

#### *Missing Completely At Random*

Uma observação é dita MCAR se a probabilidade de falta de dados numa variável Y não está relacionada com nenhuma variável, seja observada ou não observada. Em MCAR, os dados observados são apenas uma amostra aleatória de todos os dados. É possível observar na Tabela 2.1, que os valores das notas finais estão aleatoriamente ausentes em todo o *dataset*.

#### *Missing At Random*

Os dados são classificados como MAR quando a probabilidade de haver falta de valores numa variável Y depende de outros dados observados no *dataset*, mas não depende de nenhum dos valores Y propriamente ditos. Ao analisar os dados presentes

**Tabela 2.1:** Conjunto de dados simulado da avaliação do exame e o valor da nota final no curso com a ocorrência dos mecanismos MAR, MNAR e MCAR na segunda variável.

| | Nota Final (0-20) | | | |
|---|---|---|---|---|
| Exame (%) | Complete | MCAR | MAR | MNAR |
| 20 | 5 | - | - | - |
| 30 | 5 | 5 | - | - |
| 35 | 10 | 10 | - | 10 |
| 40 | 8 | - | - | - |
| 56 | 10 | 10 | 10 | 10 |
| 70 | 12 | 12 | 12 | 12 |
| 80 | 15 | - | 15 | 15 |
| 81 | 18 | 18 | 18 | 18 |
| 81 | 16 | 16 | 16 | 16 |
| 90 | 8 | 8 | 8 | - |
| 95 | 19 | 19 | 19 | 19 |
| 95 | 19 | - | 19 | 19 |
| 99 | 7 | 7 | 7 | - |

na Tabela 2.1, é possível ver que os valores que estão em falta na variável de nota final correspondem às pontuações no exame abaixo de 50% e que não estão relacionados com a própria variável.

#### *Missing Not At Random*

Finalmente, o mecanismo MNAR ocorre quando a probabilidade de falta de dados numa variável X depende dos valores da variável em si. Para exemplificá-lo, reconsidere os dados das notas finais na Tabela 2.1. Suponha que o professor apenas define uma nota de avaliação em alunos com mais de 9. Como se pode visualizar, os valores que estão em falta na variável de nota final correspondem aos valores abaixo de 10 da mesma variável.

## 2.2 Imputação

A imputação de dados é o processo de substituir os valores em falta por valores plausíveis nos conjuntos de dados incompletos (Figura 2.1). Geralmente, os métodos de imputação são divididos em imputação baseada em técnicas de ML ou em análise estatística.

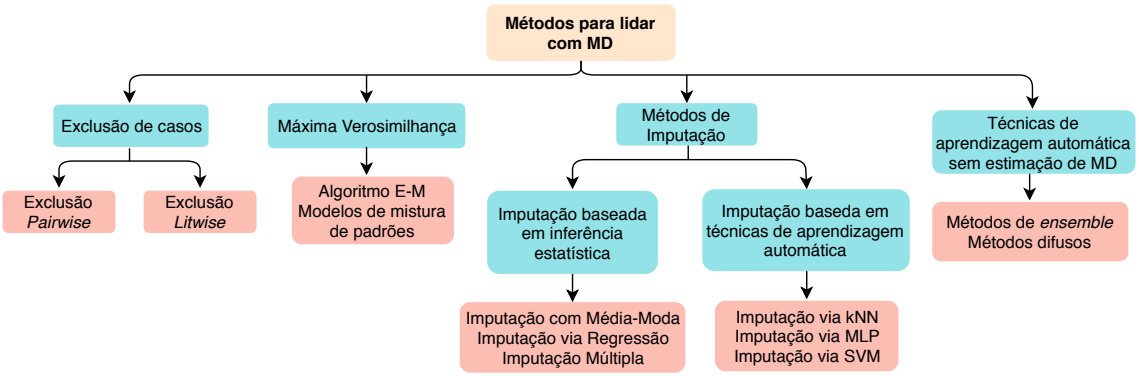



**Figura 2.1:** Métodos para lidar com dados em falta, adaptado de García-Laencina et al. [8].

Os métodos de imputação baseados na análise estatística consistem em preencher os valores em falta com uma estimativa plausível. A imputação baseada na Média-Moda (M-M) e a imputação baseada na regressão são dois exemplos deste tipo de imputação. Os métodos de imputação baseados em ML são procedimentos que geralmente recorrem aos valores disponíveis e aprendem esses valores, permitindo depois dar como entrada os dados para as variáveis completas de uma determinada observação e estimar os valores para as variáveis onde existem MD. Alguns dos algoritmos mais comuns para lidar com a falta de dados, nomeadamente, *Support Vector Machine* (SVM) [22, 23, 24], *k-Nearest Neighbors* (kNN) [25], *Decision Tree* (DT) [26, 27] and *Multiple Layer Perceptron* (MLP) [28, 29, 30].

## Imputação Baseada na M-M

A imputação com Média-Moda (M-M) consiste em substituir cada valor ausente pela média dos valores observados para essa variável, ou pela moda, caso seja um variável categórica. Esta técnica tem a vantagem de ser fácil de implementar e ter um baixo custo computacional. No entanto, acarreta consigo alguma redução da variabilidade natural dos dados, já que o número de observações completas disponíveis aumenta sem aumentar a informação estatisticamente diferente, uma vez que a média/moda de valores na variáveis onde existe MD permanecerá inalterada [31].

## Imputação Baseada em Modelos de Regressão

A imputação por regressão é uma método que recorre à modelação das variáveis observáveis para estimar os valores para substituir os valores em falta. Esta abordagem envolve o desenvolvimento de uma equação de regressão com base nos dados

completos para uma determinada variável, tratando-a como um *"output"* e usando todas as outras variáveis relevantes como preditoras. Como vantagem, este método preserva a correlação entre as variáveis [32]. Ainda assim, embora este método possa ser considerado um avanço em relação ao método previamente descrito, pode levar a uma sobre-estimação da correlação entre as variáveis.

## Imputação baseada em kNN

O *k-Nearest Neighbors* (kNN) é um algoritmo de classificação em que os k vizinhos mais próximos são escolhidos do conjunto completo de casos, encontrados pelos valores mais baixos da medida de similaridade. Depois de encontrar os k vizinhos mais próximos, o valor em falta é determinado de acordo com o tipo de dados. Para dados contínuos o valor a substituir será a média dos vizinhos próximos e a moda, no caso de serem categóricos. A principal desvantagem de usar este método é que ele tem um elevado custo computacional para grandes conjuntos de dados, uma vez que o kNN, ao contrário dos outros métodos de ML que usam os dados de treino para fazer generalizações, pesquisa por instâncias semelhantes em todo o *dataset*, sempre que é necessária a imputação de um novo caso.

## Imputação baseada em SVM

O algoritmo *Support Vector Machine* (SVM) é uma abordagem muito usada em tarefas de regressão e classificação, dado o seu desempenho a lidar com conjuntos de dados de elevadas dimensões e diversos tipos de dados [33]. Os SVMs pertencem à categoria geral de métodos do *kernel* [34]. Esta caraterística permite, por exemplo, mapear os dados para outro espaço onde eles sejam linearmente separáveis, através do ajuste do kernel. O algoritmo SVM durante o treino ajusta a complexidade do modelo e qualidade da tarefa para que foi desenhado, permitindo a generalização destes modelos a novos dados.

Em 2005, Hogain et al. [24], propuseram a aplicação do modelo SVM na imputação dos MD. Na imputação com SVM o modelo é treinado usando todos os exemplos que não possuem valores ausentes. Depois de atingir os parâmetros de SVM ideais, o modelo é usado para atribuir valores ausentes. Os atributos ausentes são tratados como "variáveis-alvo", usando os atributos completos restantes como entradas, semelhante ao que acontece na imputação por regressão e kNN.

### Imputação baseada em DT

As *Decision Tree* (DT) são árvores de decisão construídas a partir de um *dataset* de treino usando o conceito de entropia. Em cada nó da árvore, o algoritmo escolhe o atributo que melhor divide o conjunto de amostras em subconjuntos que tendem para uma categoria. O critério de divisão é o ganho de informação normalizado (diferença de entropia). Este método é talvez o método mais simples de entender e interpretar dos métodos de ML utilizados.

### Imputação baseada em MLP

*Multiple Layer Perceptron* (MLP) é a rede neuronal artificial mais comum em tarefas de imputação ou classificação, devido a ser um aproximador universal [35]. Na imputação com MLP, à semelhança da imputação com SVM e regressão, recorre-se aos casos completos para treinar o modelo que será utilizado para estimar os MD. Na fase de treino são estimados os melhores parâmetros da rede, desde os pesos que são atribuídos às entradas o número de neurónios na camada escondida.

## 2.3 Métricas de avaliação da imputação

A qualidade da imputação é a avaliação do quão próximo os valores imputados estão dos originais. As métricas para avaliar a qualidade da imputação são diferentes para variáveis discretas e contínuas.

***Root Mean Squared Error* (RMSE)** é uma das métricas mais usadas para avaliar a precisão das variáveis contínuas. RMSE é definida como raíz quadrada da média das diferenças quadráticas entre os valores originais e os valores imputados.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i}^{n}(x_i^{true} - x_i^{imp})^2} \tag{2.1}$$

***Normalized Root Mean Squared Error* (NRMSE)**, como o nome indica, é a normalização da RMSE. A vantagem da normalização de RMSE é que facilita a comparação entre conjuntos de dados ou modelos com diferentes escalas. Esta

métrica pode variar entre 0 e $\infty$.

$$NRMSE = \frac{RMSE}{\sigma_{x^{true}}} \tag{2.2}$$

***Proportion of Falsely Classified Entries* (PFC)** é a métrica mais comum na literatura para avaliar as variáveis categóricas. PFC é definida como a razão entre a quantidade de valores imputados diferentes dos originais e a quantidade total dos valores imputados (representada por N). Esta métrica pode assumir valores entre 0 e 1.

$$PFC = \frac{\sum_{i,j \ \epsilon \ missing} X_{i,j}^{true} \neq X_{i,j}^{imp}}{N} \tag{2.3}$$

Em ambas as métricas, quanto menor for o valor, melhor é a qualidade da imputação.

## 2.4 Otimização dos hiperparâmetros

Todos os algoritmos de ML possuem parâmetros que podem ser alterados para aumentar o desempenho do modelo. Portanto, encontrar o conjunto correto de parâmetros faz parte do trabalho de construir um bom estimador. No entanto, existem diferentes parâmetros que podemos definir ao treinar um modelo.

Umas das técnicas mais usadas para a otimização dos hiperparâmetros, para além da otimização manual, é o ***grid search*** [36]. Quando um *grid search* é executado, uma lista de valores possíveis é definida para os hiperparâmetros, e todas as combinações possíveis são testadas. Eles são testados treinando o estimador com esses hiperparâmetros e avaliando o seu desempenho fazendo uma validação cruzada. Por exemplo, se tivermos dois hiperparâmetros para os quais queremos encontrar a melhor combinação, uma pesquisa em grade seria a que se pode observar na Figura 2.2, em que cada ponto verde representa um teste feito. Neste caso, pode-se verificar que não foi encontrado o melhor valor para o parâmetro mais importante. Uma alternativa simples, mas recente, para a formalização da otimização de hiperparâmetros é o uso do ***random search*** [37]. A diferença entre o anterior é que este método procura valores aleatórios dentro de um intervalo de valores, como se pode observar na Figura 2.2. Alguns estudos mostram que por vezes a busca aleatória é muito mais eficiente do que a pesquisa em grade para otimizar os parâmetros de classificadores ou regressores e que tende a ser muito mais rápida [36].

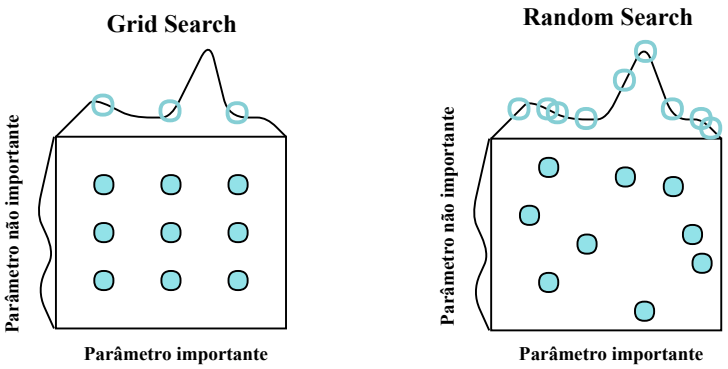



**Figura 2.2:** Exemplo de dois métodos de otimização de dois hiperparâmetros, adaptado de Bergstra et al. [36].

Os dois métodos para otimização de hiperparâmetros foram implementados na interface para os todos os métodos de imputação baseados em técnicas ML.

## 2.5 Pré-Processamento de dados

### Tratamento de dados contínuos

Muitas vezes, as variáveis numéricas nos conjuntos de dados têm escalas muito diferentes, isto é, assumem diferentes intervalos de valores. Geralmente é uma boa prática normalizar as variáveis para garantir que nenhuma assume um peso maior por estar representada numa escala de maior grandeza. Normalmente, a normalização ou padronização pode ser feita.

A normalização escala todas as variáveis numéricas no intervalo [0,1]. Uma fórmula possível é dada na equação 2.4. Nesta equação $x_{min}$ e $x_{max}$ representam o valor mínimo e máximo da variável, respetivamente.

$$x'_i = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{2.4}$$

Por outro lado, a padronização significa redimensionar os dados de modo que eles tenham as propriedades de uma distribuição normal padrão. Isso pode ser feito com uma média de 0 e um desvio padrão de 1 (equação 2.5).

$$z_i = \frac{x_i - \mu}{\sigma} \tag{2.5}$$

Na presença de alguns *outliers*, o uso das técnicas de normalização e padronização acima leva à compactação de *inliers*, já que os *outliers* têm uma influência no mínimo, no máximo, na média e no desvio padrão. Outra alternativa é uma padronização robusta que escala os recursos de maneira robusta a *outliers*: usando a mediana e o intervalo interquartil (2.6).

$$z_i = \frac{x_i - \tilde{x}}{IQR} \tag{2.6}$$

É importante observar que estas transformações alteram os dados em si, mas não a distribuição.

## Tratamento de dados categóricos

A maioria dos algoritmos requerem matrizes numéricas, portanto, é necessário converter uma variável categórica (por exemplo, codificada em texto) numa numérica antes da imputação. Existem algumas alternativas na literatura para codificar variáveis categóricas: Codificação Númerica e Codificação "*Dummy*".

A **codificação numérica** é uma codificação direta: atribuiu um número arbitrário a cada categoria, como se pode observar na Tabela 2.2. Porém, quando as categorias não apresentam uma relação ordinal entre elas, a codificação numérica não é suficiente. De facto, usar esse tipo de codificação que pressupõe um ordenamento natural entre as categorias pode resultar num péssimo desempenho.

**Tabela 2.2:** Exemplo de uma codificação numérica para uma variável categórica arbitrário com 4 categorias.

| **Variável** | **Codificação Numérica** |
|---|---|
| vermelho | 1 |
| azul | 2 |
| verde | 3 |
| azul | 2 |
| vermelho | 1 |
| amarelo | 4 |

A **codificação "*Dummy*"** é o método mais usado, convertendo uma variável categórica com N níveis de categorias em N variáveis binárias em que a presença de um nível é representada por 1 e a ausência é representada por 0. Na Tabela 2.3

encontra-se um exemplo de uma variável categórica com 4 níveis. Neste exemplo é possível observar que cada nível presente corresponde a uma variável.

**Tabela 2.3:** Exemplo de uma codificação "*dummy*" para uma variável categórica arbitrário com 4 categorias.

| **Variável** | **Variáveis *"Dummy"*** | | | |
|---|---|---|---|---|
| | **vermelho** | **azul** | **verde** | **amarelo** |
| vermelho | 1 | 0 | 0 | 0 |
| azul | 0 | 1 | 0 | 0 |
| verde | 0 | 0 | 1 | 0 |
| azul | 0 | 1 | 0 | 0 |
| vermelho | 1 | 0 | 0 | 0 |
| amarelo | 0 | 0 | 0 | 1 |

## Redução de Dimensionalidade

A elevada dimensionalidade dos conjuntos de dados aumenta a complexidade das técnicas de classificação e degrada o desempenho dos algoritmos de mineração de dados [38]. Para diminuir esses efeitos, as técnicas de redução de dimensão têm por objetivo representar um *dataset*, com uma certa dimensão, noutro espaço de dimensão menor que o original, procurando manter as características do conjunto.

O **Análise de Componentes Principais** (*Principal Component Analysis* (PCA)) é um método muito usado para fins de redução de dados. O PCA é um algoritmo de aprendizagem não supervisionado muito usado para fins de redução de dados. Ele transforma ortogonalmente as coordenadas n originais de um *dataset* num novo conjunto de coordenadas n chamadas componentes principais. Como resultado da transformação, a primeira componente principal tem a maior variância possível; cada componente sucessora tem a variação mais alta possível sob a restrição de ser ortogonal ( não correlacionada) às componentes precedentes [39].

Nesse sentido o método PCA foi implementado na interface e testado para *dataset* de grande dimensionalidade.

## 2.6 Visualização dos Dados em Falta

A visualização de *datasets* incompletos pode ser complexa, visto que os dados em falta não têm média ou distribuição e não é possível aplicar estatísticas-padrão [40]. Na literatura, existem algumas abordagens para visualizar *datasets* incompletos:

- Substituir os dados em falta, normalmente por zero;
- Omitir os dados em falta;
- Indicar a existência dos dados em falta.

Eaton et al. [41] realizaram um estudo com 30 pessoas com o intuito de avaliar as três abordagens para exibir dados em falta, concluindo que uma má indicação de ausência tem um claro efeito negativo na interpretação e sugere que as representações visuais devem ser aprimoradas por atributos que indicam a existência de valores em falta. A importância de exibir os dados em falta também é salientada no recente trabalho de Kirk [42]. A existência dos dados em falta é indicada através da sua localização e quantidade relativamente aos outros dados.

Na literatura, a visualização dos dados em falta é feita através da adaptação dos métodos usados na análise visual de dados completos, tais como gráficos de barras, gráficos de dispersão (*scatterplot*), gráficos de diagramas (*boxplots*), entre outros [18, 43]. Na Figura 2.3 está representado um *scatterplot* de duas variáveis onde é possível observar a localização dos dados em falta numa variável relativamente à outra (representados por linhas a tracejado). Neste exemplo também é possível observar a quantidade de dados em falta em cada variável.

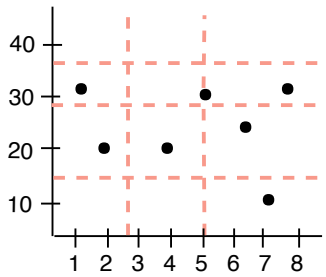



**Figura 2.3:** Exemplo de um *scatterplot* com indicação da falta de dados nas variáveis presentes. Adaptado de Eaton et al. [41]

# 3

# Revisão da literatura

Visto que o objetivo principal deste projeto é desenvolver uma ferramenta que permita a exploração visual e a imputação de dados incompletos é importante estudar as ferramentas visuais (e com componentes de imputação) existentes na literatura. O estudo do estado da arte focou-se maioritariamente na componente visual porque é o objetivo primordial.

## 3.1 MANET

Unwin et al. [15], em 1996, desenvolveram o ***Missing Now Equally Treated*** **(MANET)** - a primeira ferramenta de visualização especialmente desenvolvida para lidar com MD. MANET é uma ferramenta que só pode ser utilizada por utilizadores com o sistema operativo *Macintosh* e que permite que os MD sejam imputados e apresentados de várias maneiras diferentes (Tabela 3.1). As técnicas implementadas para imputar os valores nestas interfaces são: imputar com um valor fixo dado pelo utilizador, imputar com um valor aleatório, imputar com a média ou com múltipla imputação. No entanto, é importante notar que esta ferramenta já não se encontra ativa.

## 3.2 XGobi/GGobi

Em 1998, Swayne et al. [44] desenvolveram uma interface chamada **XGobi** restrito aos utilizadores com sistema operativo *UNIX*. XGobi é uma ferramenta de visualização desenhada para explorar dados de grande dimensão e oferece amplo suporte a gráficos dinâmicos e interativos para explorar dados. Apesar de esta interface não ser focada na análise dos MD, tem a capacidade de lidar com eles nos seus métodos

de visualização e imputação. Uma década depois, XGobi foi reimplementada num novo software chamado **GGobi** [45].

GGobi é uma interface com funcionalidades idênticas à XGobi, com uma interface melhorada e compatível com mais sistemas operativos. Nestas interfaces os gráficos estão todos ligados entre si, o que significa que se estiverem dois gráficos abertos e for selecionada uma variável ou valores de uma variável num gráfico, essa variável ou os valores irão aparecer realçados no outro gráfico. Esta ferramenta substitui os valores em falta nas variáveis por valores menores que 10% do valor mínimo naquela variável, à semelhança da interface MANET. Como é possível observar na Figura 3.1 (à direita), as linhas nos cantos direito e inferior do *scatterplot* representam os valores em falta de cada uma das variáveis separadamente e os pontos no canto inferior esquerdo representam os valores em falta comuns às duas variáveis em questão. Já os pontos no centro do gráfico representam os valores observados relativamente às duas variáveis. As técnicas implementadas para imputar os valores nestas interfaces são as mesmas que as existentes na interface MANET.

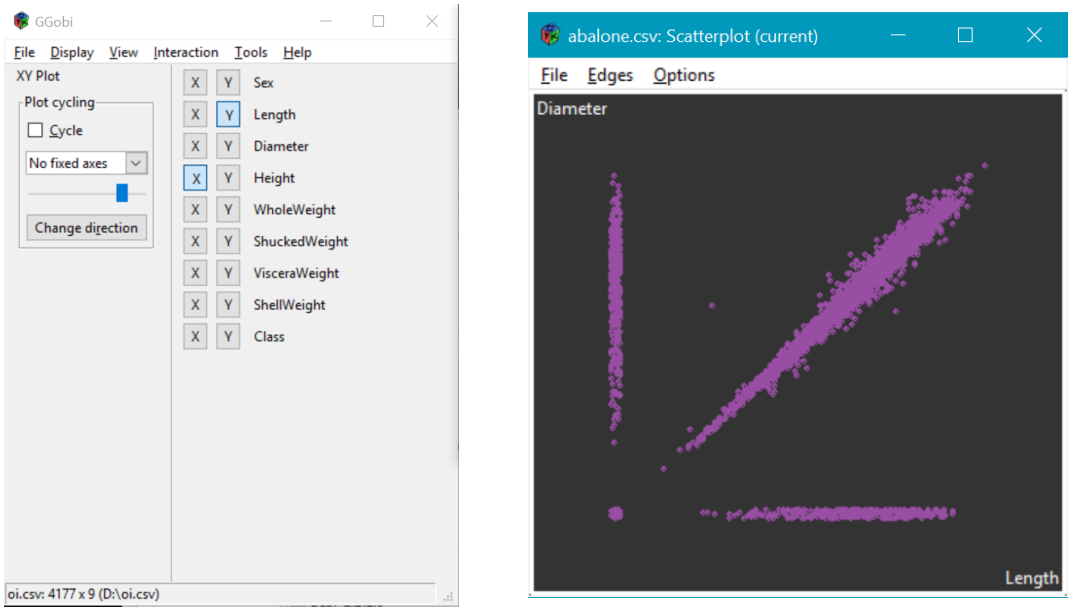



**Figura 3.1:** A interface GGobi (à direita) e um exemplo de um *scatterplot* (à esquerda) de duas variáveis. As linhas nos cantos direito e inferior do gráfico representam os valores observados (a violeta) e imputados (amarelo) de cada uma das variáveis separadamente. No centro do gráfico representam os valores observados e os valores imputados relativamente às duas variáveis.

## 3.3 Mondrian

A interface **Mondrian** foi desenhada com o propósito de ser um sistema estatístico de visualização de dados [46]. A Mondrian é parecida com GGobi e XGobi, no sentido em que todos os seus gráficos também estão ligados entre si. A Figura 3.2 mostra o gráfico de variáveis com MD no qual cada barra representa uma variável e está divida em duas partes: a parte da barra cinzenta é proporção de valores observados e a branca a de valores em falta. A cor vermelha corresponde à proporção de valores selecionada, neste caso, corresponde aos valores em falta na variável *Sex*. Do lado esquerdo é possível visualizar um *parallel boxplots* em que cada *boxplot* corresponde a cada uma das variáveis realçado com os valores correspondentes aos valores em falta na variável *Sex*. Esta interface não possui componente de imputação.

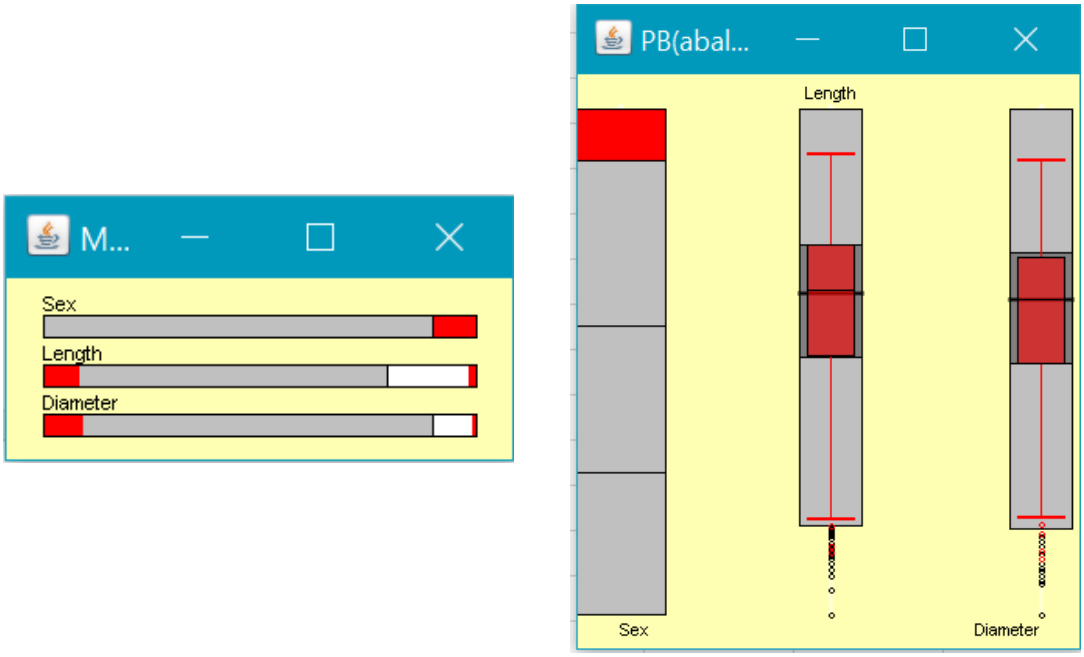



**Figura 3.2:** Dois exemplos de representações gráficas na interface de Modrian. À direita podemos ver o gráfico de falta de dados e à esquerda um *parallel boxplots*.

## 3.4 VIMGUI

Em 2008, Templ et. al [17] criaram uma biblioteca em *R*, ***Visualization and Imputation of Missing Values*** **(VIM)** que proporciona uma interface gráfica (*Graphical User Interface* (GUI)) através da função **VIMGUI**. Esta interface permite a exploração da estrutura de MD e a qualidade da imputação usando gráficos

estáticos. Todavia, só lida com conjuntos de dados importados a partir do ambiente *R*. Comparativamente às outras, esta interface fornece uma maior diversidade de métodos de visualização (Tabela 3.1). Ainda assim, não é uma ferramenta muito prática porque não permite mudar as variáveis apresentadas nos gráficos dinamicamente.

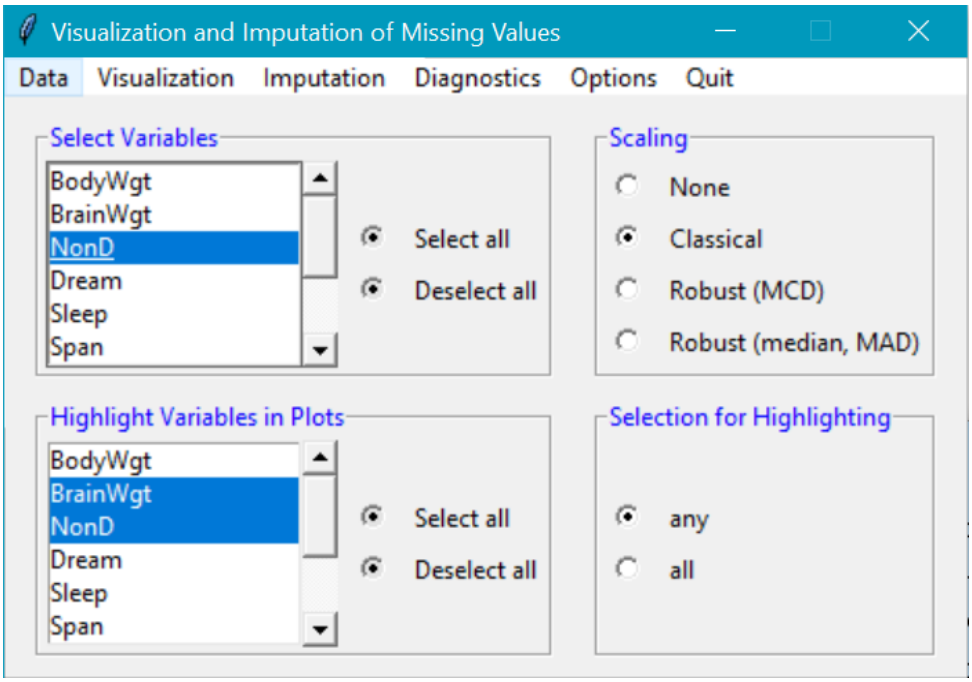



**Figura 3.3:** Interface VIMGUI.

## 3.5 MissingDataGUI

Cheng et al. [18] desenvolveram, em 2016, uma biblioteca para o *R*, a **MissingDataGUI**. Esta biblioteca à semelhança do VIM possuí também um GUI através de um função. Esta interface ajuda a explorar a estrutura de MD através de diferentes tipos de gráficos e métodos de imputação (Tabela 3.1 e Tabela 3.2, respetivamente). Esta interface assemelha-se à GGobi e MANET no sentido que para visualizar a estrutura dos dados substitui os valores em falta por valores 10% menores que o valores mínimos.

A Figura 3.4 permite visualizar um histograma da variável *length*, em que as barras azuis são os valores observados e as barras laranjas são os valores substituídos por valores menores que 10% dos valores mínimos. De todas as interfaces analisadas esta é a única que permite um sumário estatístico das variáveis.

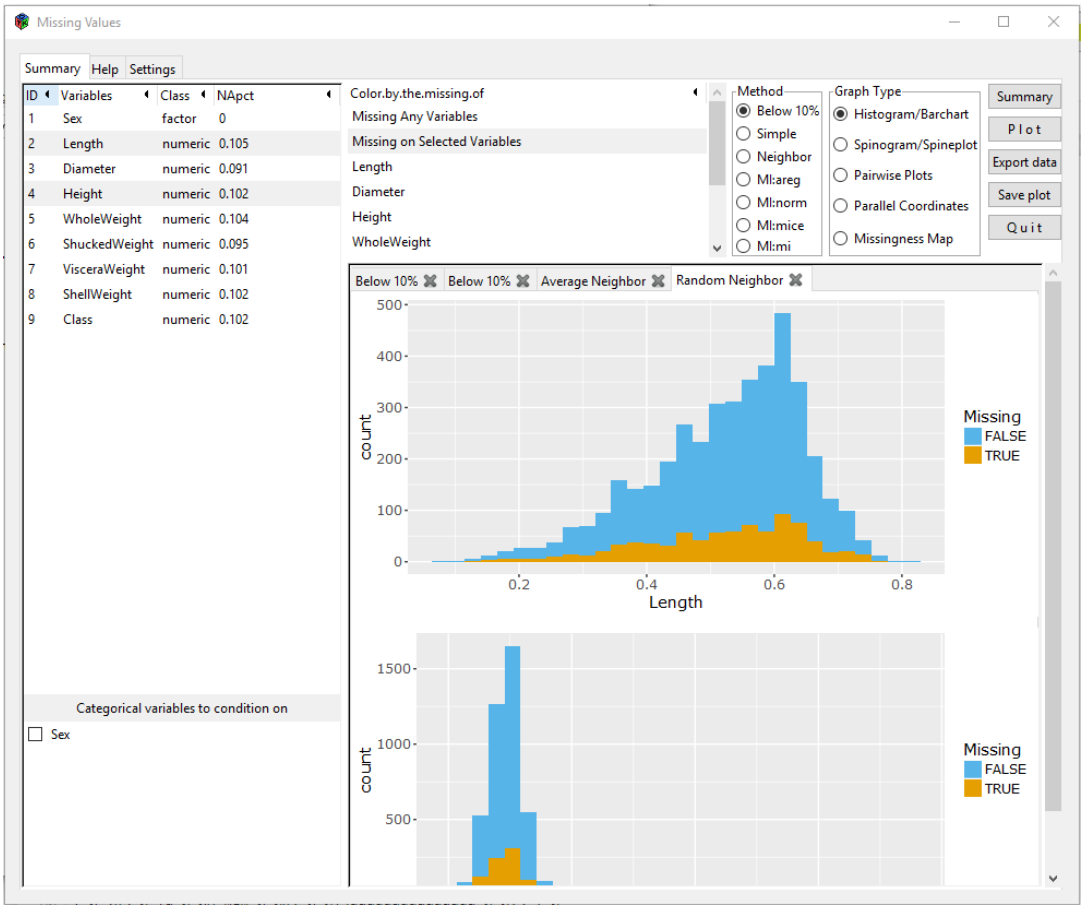



**Figura 3.4:** Interface MissingDataGUI.

## 3.6 Bibliotecas

Para além dos GUI já referidos, ao longos dos anos, foram criadas algumas bibliotecas que também permitem a exploração visual dos MD. No ambiente estatístico *R*, para além da biblioteca VIM, já descrita em cima, existe o **iPlots**, uma biblioteca do R escrita em Java introduzida em 2003 [47]. Esta ferramenta é considerada por muitos a ferramenta Mondrian para o ambiente estatístico *R*, dado que a diversidade e os aspeto dos gráficos é muito semelhante. Ainda no *R*, surgiu em 2018, a biblioteca **Naniar**, uma biblioteca mais completa que a anterior, que permite a exploração dos dados incompletos e imputados. O *Python*, muito recentemente, começou a dar os primeiros passos na exploração visual dos MD. Ainda este ano, 2018, Bilogur [48] desenvolveu a primeira biblioteca para o *Python* que proporciona algumas funções para a visualização dos MD, mas só de dados contínuos, denominada como **Missingno**. As últimas duas bibliotecas, Naniar e Missingno, apresentam alguns novos métodos de representação gráfica não usuais nas interfaces analisadas, representados na Tabela 3.1.

## 3.7 Análise Comparativa e Conclusões

Nas Tabelas 3.1 e 3.2 encontram-se, de uma forma resumida, as técnicas de visualização e imputação, respetivamente, implementadas nas diferentes interfaces e bibliotecas analisadas. As principais diferenças entre as ferramentas focam-se em diversos aspetos, tais como:

**Sistema Operativo** Duas das ferramentas são dependente do sistema operativo: MANET (*Macintosh*) e XGobi (*UNIX*), o que releva uma desvantagem comparativamente às outras.

**Disponibilidade** Das ferramentas revistas, duas delas já não se encontram ativas: MANET e XGobi.

**Bibliotecas versus Interface Gráfica** Quatro das ferramentas são bibliotecas o que é uma grande desvantagem comparativamente às interfaces já que é preciso entender as linguagens e as funções das bibliotecas para se poder criar os gráficos.

**Linguagens** As interfaces estudadas foram desenvolvidas em Java (GGobi e Modrian) ou R (MissingDataGUI e VIM). As bibliotecas foram desenvolvidas maioritariamente em R, à exceção da Missingno que foi criada em *Python*.

**Dinâmica versus Estática** As interfaces disponíveis GGobi e Modrian apresentam gráficos dinâmicos e interativos o que releva uma vantagem em relação às interfaces MissingDataGUI e VIM.

**Métodos de Visualização** De todas as ferramentas a VIMGUI é a ferramenta que apresenta uma maior diversidade de gráficos implementados (Tabela 3.1). Ainda assim, não contempla todos os gráficos existentes.

**Imputação de Dados** Apenas algumas das interfaces apresentam métodos de imputação (Tabela 3.2). Ainda assim, nessas interfaces há uma falta de diversidade de métodos de imputação, sendo que os que existem só praticamente todos baseados em análise estatística. Apenas a interface VIMGUI e MissignDataGUI fornecem um método baseado em ML, o kNN.

**Sumário estatístico** A ferramenta MissignDataGUI é única de todas as ferramentas que disponibiliza um sumário estatístico das variáveis.

Da literatura revista as principais conclusões que se podem retirar é que as interfaces desenvolvidas na linguagem *Java* (GGobi e Modrian) são melhores no sentido de terem gráficos dinâmicos e interativos mas pecam pela pouca numerosidade e

**Tabela 3.1:** Métodos de Visualização disponíveis nas interfaces VisTA, MANET, RGobi, GGobi, Modrian, VIMGUI e MissingDataGUI e bibliotecas VIM, iPlots, Missingno e Naniar.

| Métodos de Visualização | MANET | XGobi/GGobi | Modrian | VIM/VIMGUI | iPlots | MissingDataGUI | Missingno | Naniar |
|---|---|---|---|---|---|---|---|---|
| *Histogram/Barchart* | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| *Spinogram/Spineplot* | ✓ | | | ✓ | | ✓ | | |
| Parallel Coordinates | | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| *Scatterplots* | ✓ | ✓ | | ✓ | ✓ | | | ✓ |
| *Scatterplot matrix* | | ✓ | | ✓ | | ✓ | | |
| *Boxplot* | ✓ | | ✓ | ✓ | ✓ | | | |
| *Parallel Boxplots* | | | ✓ | ✓ | ✓ | | | |
| *Mosaic Plot* | ✓ | | ✓ | ✓ | ✓ | | | |
| *Spinogram/Spineplot* | ✓ | | | ✓ | ✓ | ✓ | | |
| *Aglomeration Plot* | | | | ✓ | | | | |
| *Marginplot* | | | | ✓ | | | | |
| *Bivariate Jitter Plot* | | | | ✓ | | | | |
| Marginplot matrix | | | | ✓ | | | | |
| *Density Plot* | | | | | | | | ✓ |
| Dendogram | | | | | ✓ | | ✓ | |
| *Violinplot* | ✓ | | | | | | | |
| *Correlation Heatmap* | | | | | | | ✓ | ✓ |
| *Missmap* | | | | | | ✓ | ✓ | ✓ |
| *Decision Tree Plot* | | | | | | | | ✓ |

**Tabela 3.2:** Número de métodos de imputação disponíveis nas interfaces MANET, XGobi/GGobi, VIM e MissingDataGUI.

| Tipos de Imputação | MANET | XGobi/Ggobi | VIM | MissingDataGUI |
|---|---|---|---|---|
| Imputação com a média | | ✓ | | ✓ |
| Imputação com a mediana | ✓ | ✓ | | ✓ |
| Imputação com um valor fixo | ✓ | ✓ | | ✓ |
| Imputação com um valor aleatório | ✓ | ✓ | | |
| *Iterative robust model-based imputation* | | | ✓ | |
| Múltipla imputação | ✓ | | | |
| Imputação com kNN | | | ✓ | ✓ |
| *Hot deck* | | | ✓ | |

diversidade dos gráficos de visualização. As interfaces desenhadas em ambiente *R* (MissingDataGUI e VIM) perdem por serem interfaces com gráficos estáticos e de não serem interfaces dinâmicas, mas tem a vantagem de possuírem um maior número de gráficos de visualização. Todas estas interfaces têm poucos recursos para imputar e os métodos de imputação disponíveis são muito simples e pouco referenciados atualmente para imputação na literatura [8].

Em suma, este trabalho tem como objetivo criar uma ferramenta com uma componente visual interativa contendo todos os métodos de visualização e alguns dos métodos de imputação do estado da arte. Ainda deverá conter um relatório estatístico, à semelhança da interface MissingDataGUI, mas mais completo, incluindo uma análise estatística, por medidas e gráficos, dos dados originais e imputados.

# 4

# Ferramenta Gráfica: Arquitectura e Implementação

Neste capítulo será descrita a arquitetura da ferramenta gráfica desenvolvida e os principais detalhes relativos à sua implementação.

A arquitetura da ferramenta é constituída por módulos, representados na Figura 4.1, onde as componentes a laranja foram implementadas na linguagem *R*, a amarelo em *Python* e as restante em *Java*. A interface em si é executada no ambiente *Java*. Nas seguintes secções serão descritos com mais detalhes esses módulos (entrada, visualização, imputação e interface), as suas componentes e a justificação das linguagens escolhidas. O módulo que implementa as funcionalidades extras será discutido no Capítulo 5, onde se apresentam todas as funcionalidades da interface.

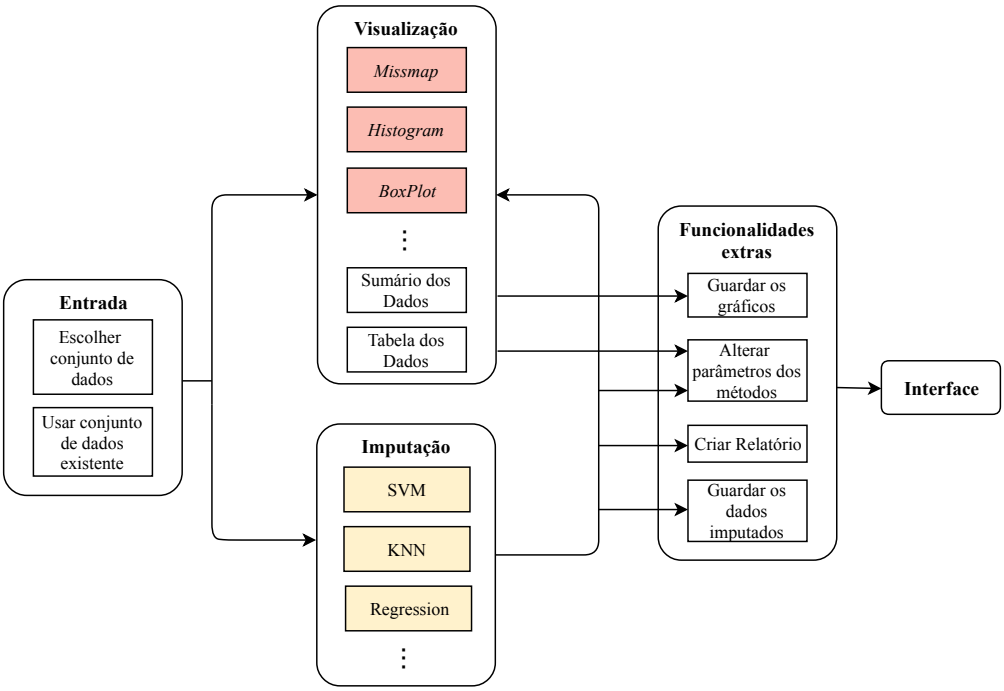



**Figura 4.1:** Sumário dos principais módulos da ferramenta.

## 4.1 Módulo de Entrada

O módulo de entrada tem como objetivo o pré-processamento dos dados introduzidos na interface. O utilizador tem duas opções para inicializar o *Graphical User Interface* (GUI): importar um conjunto de dados do seu computador ou utilizar um conjunto de dados-exemplo fornecido pela interface (Figura 4.2). Os formatos de ficheiros permitidos pela a interface são: *csv*, *arff* e *txt*.

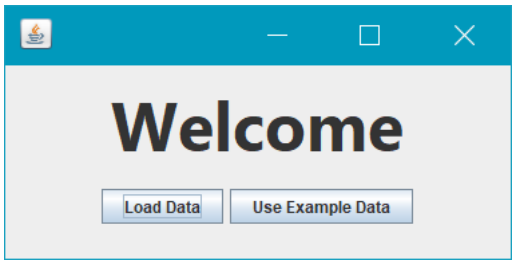



**Figura 4.2:** Painel inicial da interface.

Depois de escolhido o conjunto de dados é iniciada a fase de pré-processamento dos dados. Na primeira fase são detetadas as variáveis com valores em falta. Seguidamente, são determinados os diferentes tipos de variáveis existentes. Cada variável pode ser de um de três tipos: identificador (*id*), categórica ou numérica.

O esquema na Figura 4.3 mostra resumidamente como foi implementada a seleção do tipo de variáveis para os ficheiros do tipo *csv* e *txt*. Esta deteção é importante na fase de visualização e de imputação. No caso da visualização, por exemplo, alguns gráficos só funcionam para variáveis numéricas e outros para variáveis categóricas. Na imputação a deteção é importante pois diferentes tratamentos durante o pré-processamento são realizados para os diferentes tipos de variáveis, referidos na Secção 2.5. Relativamente aos ficheiros do tipo *arff* este processo de seleção não foi necessário, porque a informação do tipo das variáveis já vem inserida no próprio ficheiro.

A seleção dos tipos de variáveis não é imutável, podendo ser alterada pelo utilizador a qualquer momento (Figura 4.4).

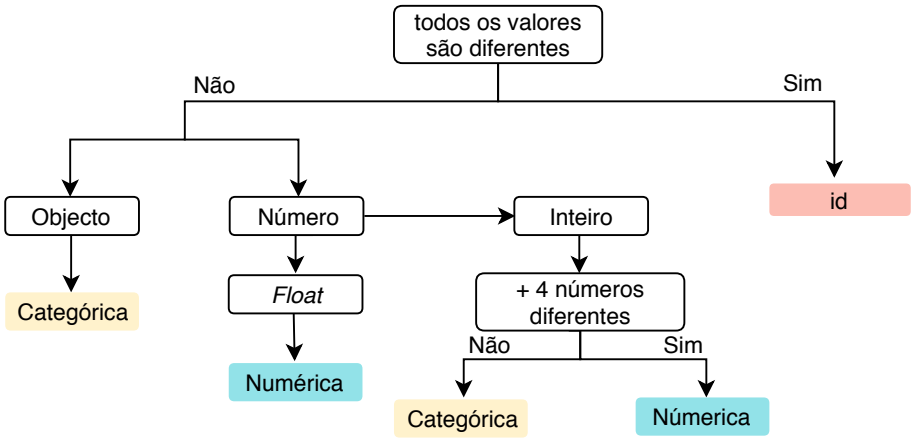



**Figura 4.3:** Esquema do processo de seleção do tipo de variável.

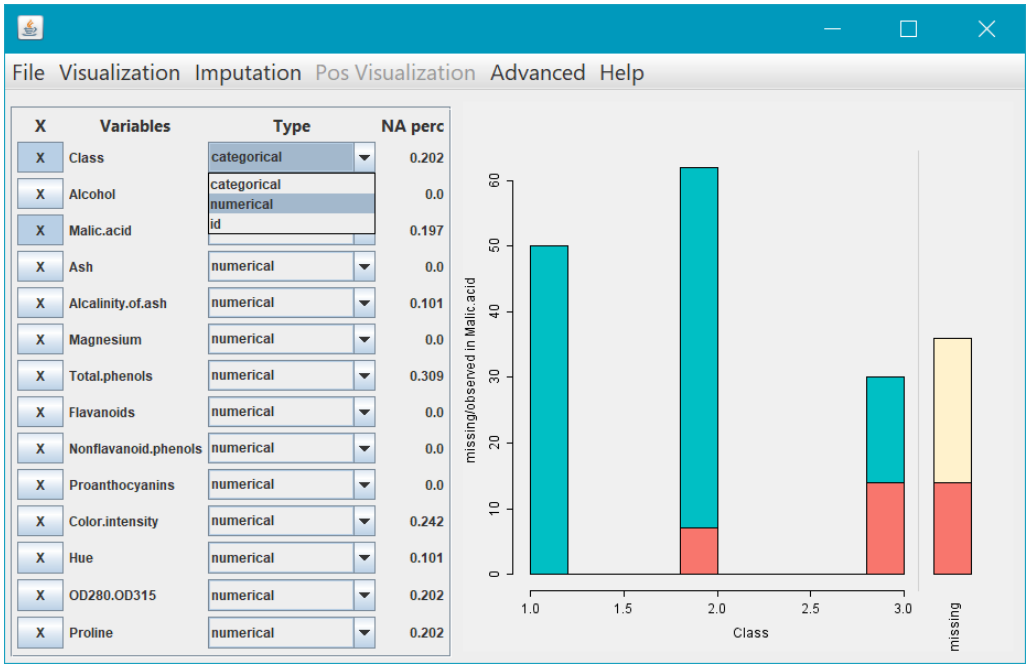


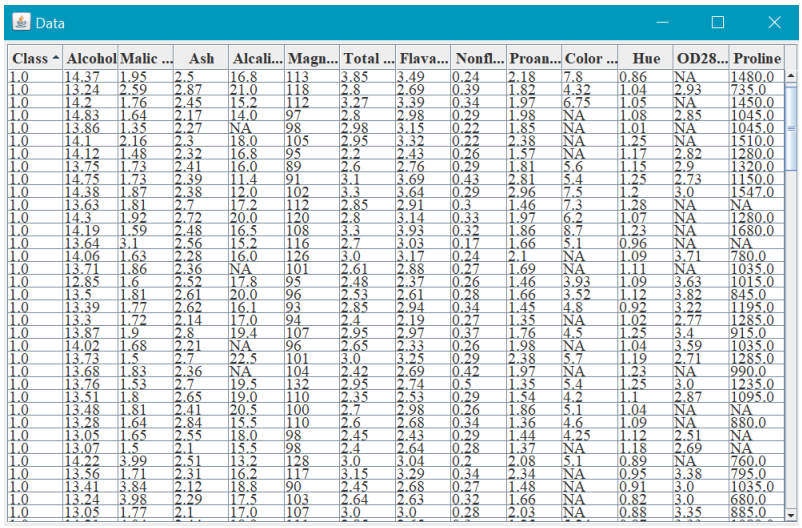

**Figura 4.4:** Exemplificação da alteração do tipo de uma variável do conjunto de dados.

## 4.2 Módulo de Visualização

O módulo da visualização destina-se à exploração visual dos *datasets* incompletos e posteriormente imputados. Este módulo é interessante, por exemplo, para visualizar o impacto que tem cada tipo de imputação. A visualização pode ser feita através diversas maneiras:

- Visualização dos dados numa tabela e sumário das variáveis;
- Representações Gráficas (como será demonstrado na Secção 5.1).

Na visualização dos dados numa tabela (Figura 4.5) o utilizador pode ordenar os valores do *dataset*. Por exemplo, o *dataset* apresentado na Figura 4.5 está ordenado, em ordem crescente, pela primeira coluna. O sumário das variáveis (painel à esquerda no exemplo da interface na Figura 4.4) inclui o nome, o tipo e a percentagem de dados em falta de cada variável. Por último, as representações gráficas referem-se aos métodos de visualização referidos no Capítulo 3 que são demonstrados na Secção 5.1. Estas representações gráficas foram maioritariamente implementadas em *R*. A preferência da linguagem *R* nesta tarefa foi devido à diversidade de bibliotecas capazes de criar métodos de visualização para os dados em falta e imputados, referidas no Capítulo 3.




| Class | Alcohol | Malic ... | Ash | Alcali... | Magn... | Total ... | Flava... | Nonfl... | Proan... | Color ... | Hue | OD28... | Proline |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.0 | 14.37 | 1.95 | 2.5 | 16.8 | 113 | 3.85 | 3.49 | 0.24 | 2.18 | 7.8 | 0.86 | NA | 1480.0 |
| 1.0 | 13.24 | 2.59 | 2.87 | 21.0 | 118 | 2.8 | 2.69 | 0.39 | 1.82 | 4.32 | 1.04 | 2.93 | 735.0 |
| 1.0 | 14.2 | 1.76 | 2.45 | 15.2 | 112 | 3.27 | 3.39 | 0.34 | 1.97 | 6.75 | 1.05 | NA | 1450.0 |
| 1.0 | 14.83 | 1.64 | 2.17 | 14.0 | 97 | 2.8 | 2.98 | 0.29 | 1.98 | NA | 1.08 | 2.85 | 1045.0 |
| 1.0 | 13.86 | 1.35 | 2.27 | NA | 98 | 2.98 | 3.15 | 0.22 | 1.85 | NA | 1.01 | NA | 1045.0 |
| 1.0 | 14.1 | 2.16 | 2.3 | 18.0 | 105 | 2.95 | 3.32 | 0.22 | 2.38 | NA | 1.25 | NA | 1510.0 |
| 1.0 | 14.12 | 1.48 | 2.32 | 16.8 | 95 | 2.2 | 2.43 | 0.26 | 1.57 | NA | 1.17 | 2.82 | 1280.0 |
| 1.0 | 13.75 | 1.73 | 2.41 | 16.0 | 89 | 2.6 | 2.76 | 0.29 | 1.81 | 5.6 | 1.15 | 2.9 | 1320.0 |
| 1.0 | 14.75 | 1.73 | 2.39 | 11.4 | 91 | 3.1 | 3.69 | 0.43 | 2.81 | 5.4 | 1.25 | 2.73 | 1150.0 |
| 1.0 | 14.38 | 1.87 | 2.38 | 12.0 | 102 | 3.3 | 3.64 | 0.29 | 2.96 | 7.5 | 1.2 | 3.0 | 1547.0 |
| 1.0 | 13.63 | 1.81 | 2.7 | 17.2 | 112 | 2.85 | 2.91 | 0.3 | 1.46 | 7.3 | 1.28 | NA | NA |
| 1.0 | 14.3 | 1.92 | 2.72 | 20.0 | 120 | 2.8 | 3.14 | 0.33 | 1.97 | 6.2 | 1.07 | NA | 1280.0 |
| 1.0 | 14.19 | 1.59 | 2.48 | 16.5 | 108 | 3.3 | 3.93 | 0.32 | 1.86 | 8.7 | 1.23 | NA | 1680.0 |
| 1.0 | 13.64 | 3.1 | 2.56 | 15.2 | 116 | 2.7 | 3.03 | 0.17 | 1.66 | 5.1 | 0.96 | NA | NA |
| 1.0 | 14.06 | 1.63 | 2.28 | 16.0 | 126 | 3.0 | 3.17 | 0.24 | 2.1 | NA | 1.09 | 3.71 | 780.0 |
| 1.0 | 13.71 | 1.86 | 2.36 | NA | 101 | 2.61 | 2.88 | 0.27 | 1.69 | NA | 1.11 | NA | 1035.0 |
| 1.0 | 12.85 | 1.6 | 2.52 | 17.8 | 95 | 2.48 | 2.37 | 0.26 | 1.46 | 3.93 | 1.09 | 3.63 | 1015.0 |
| 1.0 | 13.5 | 1.81 | 2.61 | 20.0 | 96 | 2.53 | 2.61 | 0.28 | 1.66 | 3.52 | 1.12 | 3.82 | 845.0 |
| 1.0 | 13.39 | 1.77 | 2.62 | 16.1 | 93 | 2.85 | 2.94 | 0.34 | 1.45 | 4.8 | 0.92 | 3.22 | 1195.0 |
| 1.0 | 13.3 | 1.72 | 2.14 | 17.0 | 94 | 2.4 | 2.19 | 0.27 | 1.35 | NA | 1.02 | 2.77 | 1285.0 |
| 1.0 | 13.87 | 1.9 | 2.8 | 19.4 | 107 | 2.95 | 2.97 | 0.37 | 1.76 | 4.5 | 1.25 | 3.4 | 915.0 |
| 1.0 | 14.02 | 1.68 | 2.21 | NA | 96 | 2.65 | 2.33 | 0.26 | 1.98 | NA | 1.04 | 3.59 | 1035.0 |
| 1.0 | 13.73 | 1.5 | 2.7 | 22.5 | 101 | 3.0 | 3.25 | 0.29 | 2.38 | 5.7 | 1.19 | 2.71 | 1285.0 |
| 1.0 | 13.68 | 1.83 | 2.36 | NA | 104 | 2.42 | 2.69 | 0.42 | 1.97 | NA | 1.23 | NA | 990.0 |
| 1.0 | 13.76 | 1.53 | 2.7 | 19.5 | 132 | 2.95 | 2.74 | 0.5 | 1.35 | 5.4 | 1.25 | 3.0 | 1235.0 |
| 1.0 | 13.51 | 1.8 | 2.65 | 19.0 | 110 | 2.35 | 2.53 | 0.29 | 1.54 | 4.2 | 1.1 | 2.87 | 1095.0 |
| 1.0 | 13.48 | 1.81 | 2.41 | 20.5 | 100 | 2.7 | 2.98 | 0.26 | 1.86 | 5.1 | 1.04 | NA | NA |
| 1.0 | 13.28 | 1.64 | 2.84 | 15.5 | 110 | 2.6 | 2.68 | 0.34 | 1.36 | 4.6 | 1.09 | NA | 880.0 |
| 1.0 | 13.05 | 1.65 | 2.55 | 18.0 | 98 | 2.45 | 2.43 | 0.29 | 1.44 | 4.25 | 1.12 | 2.51 | NA |
| 1.0 | 13.07 | 1.5 | 2.1 | 15.5 | 98 | 2.4 | 2.64 | 0.28 | 1.37 | NA | 1.18 | 2.69 | NA |
| 1.0 | 14.22 | 3.99 | 2.51 | 13.2 | 128 | 3.0 | 3.04 | 0.2 | 2.08 | 5.1 | 0.89 | NA | 760.0 |
| 1.0 | 13.56 | 1.71 | 2.31 | 16.2 | 117 | 3.15 | 3.29 | 0.34 | 2.34 | NA | 0.95 | 3.38 | 795.0 |
| 1.0 | 13.41 | 3.84 | 2.12 | 18.8 | 90 | 2.45 | 2.68 | 0.27 | 1.48 | NA | 0.91 | 3.0 | 1035.0 |
| 1.0 | 13.24 | 3.98 | 2.29 | 17.5 | 103 | 2.64 | 2.63 | 0.32 | 1.66 | NA | 0.82 | 3.0 | 680.0 |
| 1.0 | 13.05 | 1.77 | 2.1 | 17.0 | 107 | 3.0 | 3.0 | 0.28 | 2.03 | NA | 0.88 | 3.35 | 885.0 |

**Figura 4.5:** Exemplo da visualização de um *dataset* numa tabela.

Visto que o objetivo da ferramenta é a exploração visual dos dados incompletos e imputados, as representações gráficas são as componentes mais importantes. Deste modo, na criação desta componente tiveram-se em atenção 3 aspetos importantes:

- Tornar as representações gráficas interativas;
- Criação de mensagens de erro;
- Criação de mensagens de alerta.

A visualização interativa dos diferentes métodos foi feita através da criação de diferentes tipos de botões, dependendo do tipo de gráfico. Para representações gráficas que só permitem a visualização de duas variáveis foram criados duas colunas de botões, a "X" e a "Y", em que apenas é permitida a seleção de um botão por coluna e não é permitido a seleção da mesma variável no "X" e "Y" (exemplo na Figura 4.6).

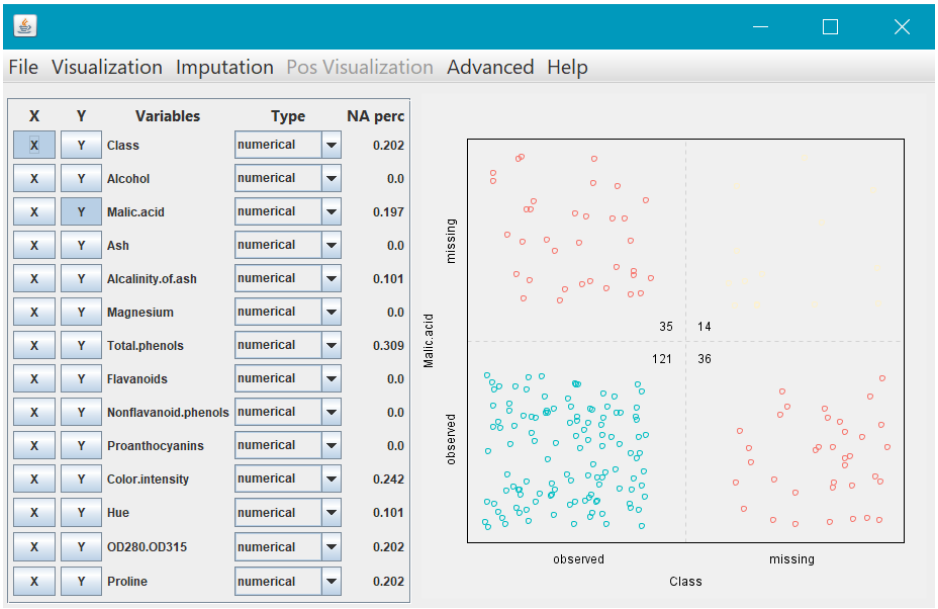



**Figura 4.6:** Exemplo da visualização do método *Bivariate Jitter Plot*.

Para representações gráficas que permitem a visualização de mais que duas variáveis ao mesmo tempo foi criada apenas uma coluna de botões "X" que permitem a seleção de vários botões ao mesmo tempo (Figura 4.7).

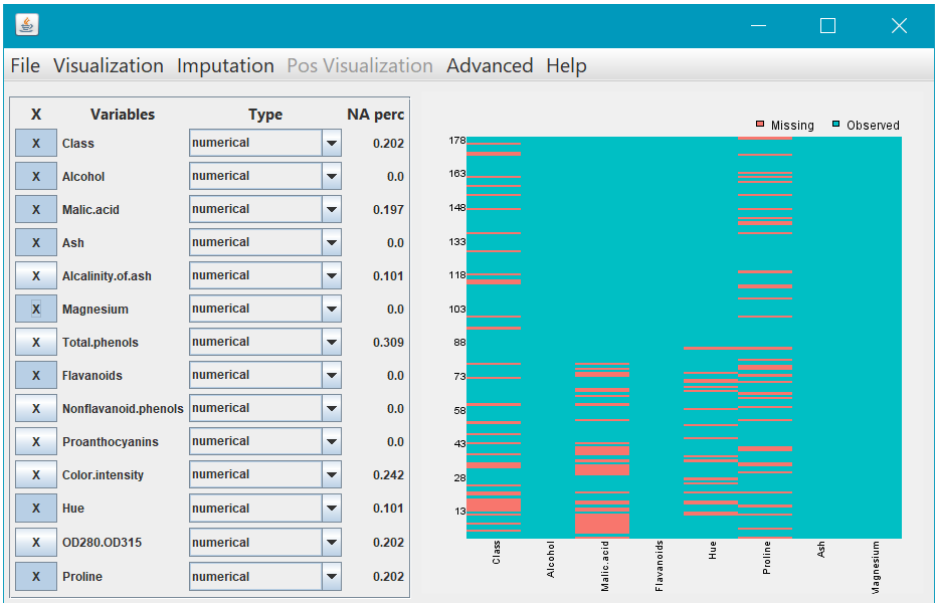



**Figura 4.7:** Exemplo da visualização do método *Missmap*.

Para este tipo de seleção de variáveis foi necessária a criação de mensagens de erro para alertar o utilizador quando selecionava um número de variáveis não permitido para aquele tipo de visualização. É possível observar um exemplo deste tipo de mensagem na Figura 4.8. Neste exemplo, o utilizador tentou visualizar um *parallel*

*coordinate plot* com apenas uma variável, no entanto este método necessita de pelo menos duas variáveis.

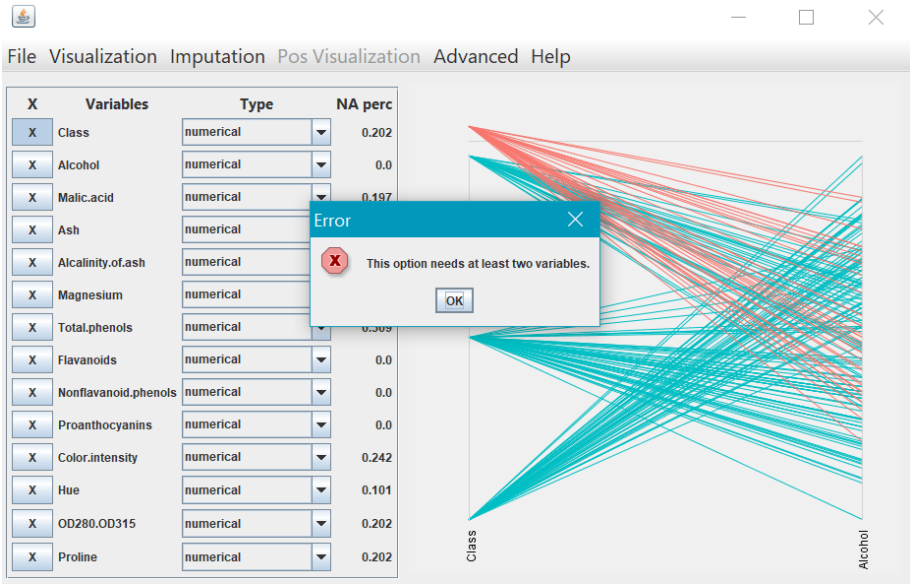



**Figura 4.8:** Exemplo de uma mensagem de erro.

Além disso, também foram criadas mensagens de alerta, pois apesar dos métodos de visualização permitirem a seleção de um número indefinido de variáveis, tal seleção pode tornar a sua visualização impercetível. Por exemplo, o *scatterplot matrices* é dos métodos que se torna impercetível a visualização dos diagramas de dispersão aquando da seleção de muitas variáveis, como se pode observar na Figura 4.9.

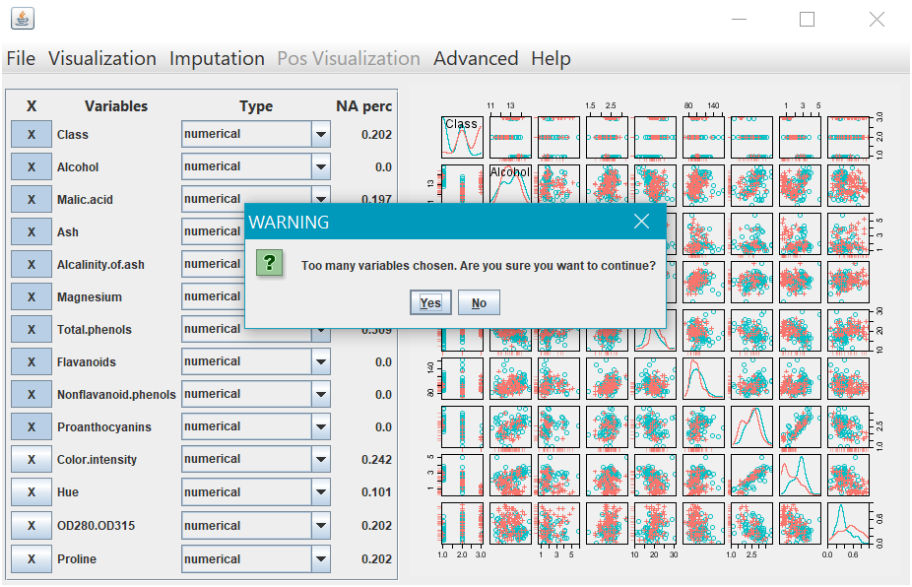



**Figura 4.9:** Exemplo de uma mensagem de alerta.

## 4.3 Módulo de Imputação

Este módulo tem como objetivo a imputação de dados incompletos. Os métodos de imputação foram implementados em *Python*. De acordo com a pesquisa recente da KDNuggets [49], o *Python* é o líder indiscutível no uso para ciência dos dados (*data science*) e ML. Algumas das razões frequentemente apontadas nessa pesquisa para essa preferência são a grande variedade de bibliotecas e o facto de ser considerada uma linguagem fácil de se trabalhar.

Os métodos de imputação implementados são: M-M, Regressão, SVM, kNN, MLP e DT. Todos os métodos foram implementados usando funções da biblioteca *scikit-learn* [50] do *Python*.

No processo de imputação baseada em técnicas de ML, o processo de treino é realizado com os dados observados e passo a passo para cada variável individual. Isto é, se considerarmos uma coluna com dados em falta como nossa variável alvo e as colunas existentes como nossas variáveis preditoras, então podemos construir um modelo de ML usando as observações para as quais o valor alvo não está ausente, e assim o método treinado é aplicado para imputar valores em falta. Este processo é repetido para todas as variáveis com valores em falta. Como a maioria dos algoritmos de ML não aceita valores em falta esses valores são normalmente imputados com um método simples (por exemplo, M-M). Na Figura 4.10 está representado um exemplo de imputação a uma das variáveis com valores em falta ($Y_3$). É possível ver neste exemplo que os dados de treino ($X_{train}$) correspondem aos valores onde a variável $Y_3$ é observada ($y_{train}$) e os dados de teste ($X_{test}$) correspondem aos valores onde a variável $Y_3$ está em falta ($y_{test}$).

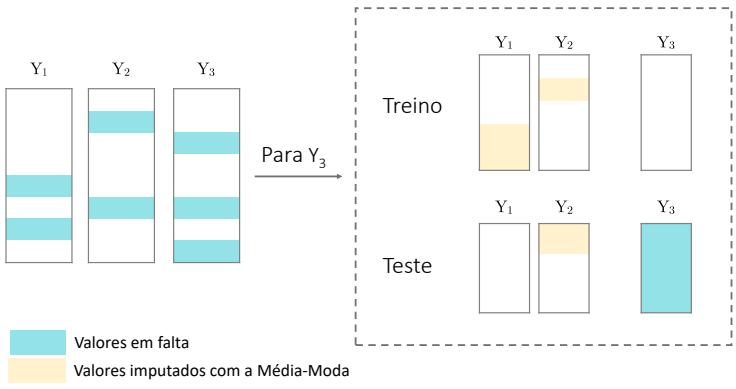



**Figura 4.10:** Exemplo de uma imputação na variável $Y_3$ num *dataset* incompleto arbitrário usando um algoritmo de ML.

Após separar o *dataset* em $X_{train}$, $X_{test}$, $y_{train}$ e $y_{test}$ é necessário pré-processar o *dataset* antes da imputação. Assim, a primeira etapa do pré-processamento baseia-se numa imputação dos valores em falta em $X_{train}$ com o método M-M. De seguida, dois processos são implementados: transformação dos dados e redução de dimensionalidade. Finalmente, é aplicado um modelo para a imputação de $y_{test}$: um classificador se a variável a imputar é categórica ou um regressor se for contínua.

### Transformação dos dados

Nesta etapa é realizada a transformação dos dados numéricos e categóricos. Enquanto que a transformação das variáveis categóricas foi realizada através da codificação *"dummy"*, para as variáveis numéricas foram testadas três técnicas, descritas na Secção 2.5: normalização, padronização normal e padronização robusta.

Para analisar as diferentes técnicas, foram selecionados a partir de repositórios de dados *online datasets* típicos do mundo real com várias dimensões, diversos tamanhos de amostra e diferentes tipos de variáveis. O menor número de variáveis foi de 3 e o maior de 152. Em relação ao número de observações, o valor mais baixo foi de 22 e o maior foi de 20000. Nos 78 *datasets* obtidos, 5 possuem apenas variáveis numéricas, 9 apenas categóricas e 64 ambas.

Todos os *datasets* foram obtidos do *UCI Machine Learning Repository* [51], *Knowledge Extraction based on Evolutionary Learing* (KEEL) [52] e do repositório *OpenML*. As características básicas desses *datasets* são resumidas na Tabela 4.1, estando no apêndice B na Tabela B.1 a sua descrição completa.

Foi desenvolvido um caso de estudo para testar diferentes métodos para o pré-processamento das variáveis numéricas durante a imputação. Para isso, foi gerado o mecanismo MCAR em todas as variáveis dos 78 *datasets* com diferentes percentagens de valores em falta: 10% e 20%. Os dados incompletos foram imputados com DT. As métricas usadas para avaliar as três hipóteses foram NRMSE, PFC e o tempo de execução.

Os resultados obtidos mostram que as três técnicas apresentam uma performance idêntica (valores de NRMSE e PFC) para os diferentes *datasets* testados, sendo que os resultados do PFC são ligeiramente melhores aquando da utilização normalização. No entanto, o tempo de execução dos métodos de imputação foi geralmente menor aquando da utilização da normalização, sendo que esta diferença se torna mais notável para *datasets* com uma maior dimensionalidade.

A Figura 4.11 mostra os valores do tempo de execução do método de imputação

**Tabela 4.1:** Uma breve descrição dos *datasets*.

| ***datasets*** | **Variáveis** | **Observações** | ***datasets*** | **Variáveis** | **Observações** |
|---|---|---|---|---|---|
| diabetes | 3 | 43 | glass_0_6_vs_5 | 10 | 108 |
| vertebral-2c | 3 | 310 | glass_0_1_5_vs_2 | 10 | 172 |
| ele-1 | 3 | 495 | glass_0_1_6_vs_5 | 10 | 184 |
| haberman | 4 | 306 | glass_0_1_6_vs_2 | 10 | 192 |
| quake | 4 | 2178 | glass_0_1_4_6_vs_2 | 10 | 205 |
| adult+strech | 5 | 22 | glass | 10 | 214 |
| iris0 | 5 | 150 | shuttle_6_vs_2_3 | 10 | 230 |
| balance_scaleBvsL | 5 | 337 | wisconsin | 10 | 683 |
| balance_scaleBvsR | 5 | 337 | breast-cancer-wisconsin | 10 | 699 |
| balance-scale | 5 | 625 | dataset_50_tic-tac-toe | 10 | 958 |
| transfusion | 5 | 748 | tic-tac-toe | 10 | 958 |
| laser | 5 | 993 | cmc1vs2 | 10 | 962 |
| newthyroid-v3 | 6 | 180 | shuttle_c0_vs_c4 | 10 | 1829 |
| newthyroid-v1 | 6 | 185 | poker_9_vs_7 | 11 | 244 |
| biomed | 6 | 194 | page_blocks_1_3_vs_4 | 11 | 472 |
| new-thyroid | 6 | 215 | pageblocks_1vs4_5 | 11 | 5116 |
| mammographic | 6 | 830 | pageblocks_1vs3_4_5 | 11 | 5144 |
| kala-azar | 7 | 68 | page_blocks0 | 11 | 5472 |
| bankruptcy | 7 | 250 | pageblocks_1_2vs_3_4_5 | 11 | 5473 |
| bupa | 7 | 345 | winequality_red_8_vs_6 | 12 | 656 |
| monk-2 | 7 | 432 | winequality_red_4 | 12 | 1599 |
| monks-problems-2 | 7 | 601 | wine | 13 | 178 |
| car_vgood | 7 | 1728 | relax | 13 | 182 |
| kr_vs_k_one_vs_fifteen | 7 | 2244 | cleveland_0_vs_4 | 14 | 173 |
| kr_vs_k_zero_one_vs_draw | 7 | 2901 | parkinson | 14 | 195 |
| kr_vs_k_three_vs_eleven | 7 | 2935 | vowel0 | 14 | 988 |
| appendicitis | 8 | 106 | eeg | 15 | 14980 |
| led7digit | 8 | 443 | seismic-bumps | 16 | 2584 |
| abalone_3_vs_11 | 9 | 502 | letter_Z | 17 | 20000 |
| abalone_21_vs_8 | 9 | 581 | vehicle0 | 19 | 846 |
| abalone9_18 | 9 | 731 | segment0 | 19 | 2308 |
| pima | 9 | 768 | auto | 25 | 159 |
| yeast | 9 | 1484 | wpbc | 33 | 198 |
| abalone_17_vs_7_8_9_10 | 9 | 2338 | ionosphere | 34 | 351 |
| abalone | 9 | 4177 | satimage | 37 | 4435 |
| nursery | 9 | 12960 | spectf | 45 | 267 |
| glass_0_4_vs_5 | 10 | 92 | sonar | 60 | 207 |
| fertility-diagnosis | 10 | 100 | R_data_frame | 101 | 1212 |
| breast-tissue-2c | 10 | 106 | grastroenterology | 152 | 700 |

com DT para todos os 78 *datasets* (ordenados por número de variáveis) com 10% de dados em falta. É possível observar nesta figura que para os *datasets* com maior número de variáveis a diferença entre os tempos de execução é maior, sendo que o que usa a normalização apresenta quase sempre uma valor menor.

Perante os resultados obtidos, decidiu-se optar pela normalização no pré-processamento dos *datasets* durante a imputação. Os restantes resultados estão no anexo C.

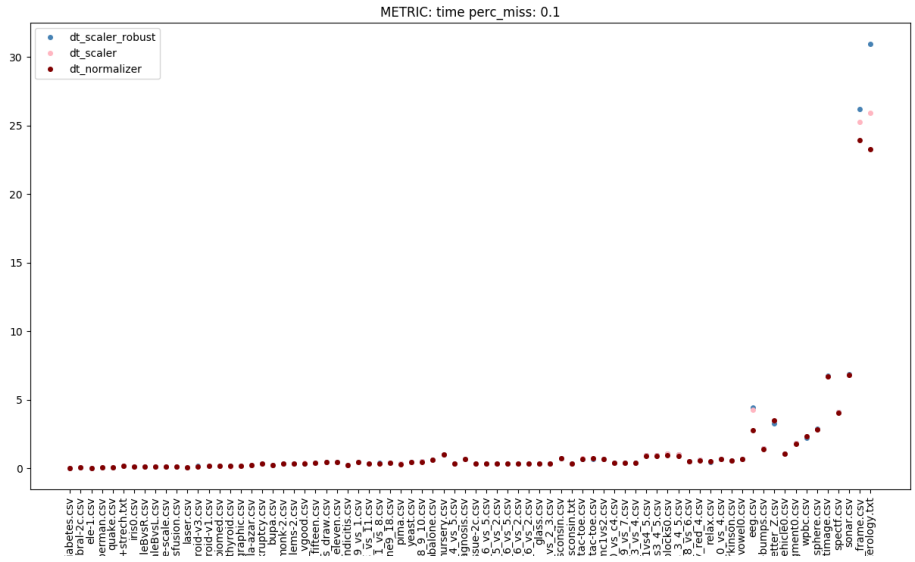



**Figura 4.11:** Tempo de execução da imputação com DT para os *datasets* nos quais contêm uma variável com 10% de dados em falta.

### Redução da dimensionalidade

Nesta etapa foi testada uma técnica de redução de dimensionalidade: o PCA. O caso de estudo é idêntico ao anterior mas com um método de imputação diferente: imputação com MLP. Foram utilizadas as mesmas métricas na avaliação do PCA. O objetivo foi determinar se o PCA melhora a performance das imputações para *datasets* a partir de uma certa dimensionalidade.

Os resultados obtidos podem ser visualizados no anexo D. Estes resultados revelam que o PCA não tem grande impacto na performance (NRMSE e PFC) para os diferentes *datasets* testados. Todavia, apresenta uma certa melhoria em alguns dos *datasets* com maiores quantidade de variáveis (Figura 4.12). Relativamente aos resultados do tempo de execução (Figura 4.13) os valores obtidos revelam que este método de redução de dimensionalidade melhora os resultados para *datasets* com um número de variáveis superior a 15. Concluindo, o PCA só foi implementado para *datasets* com mais de 15 variáveis.

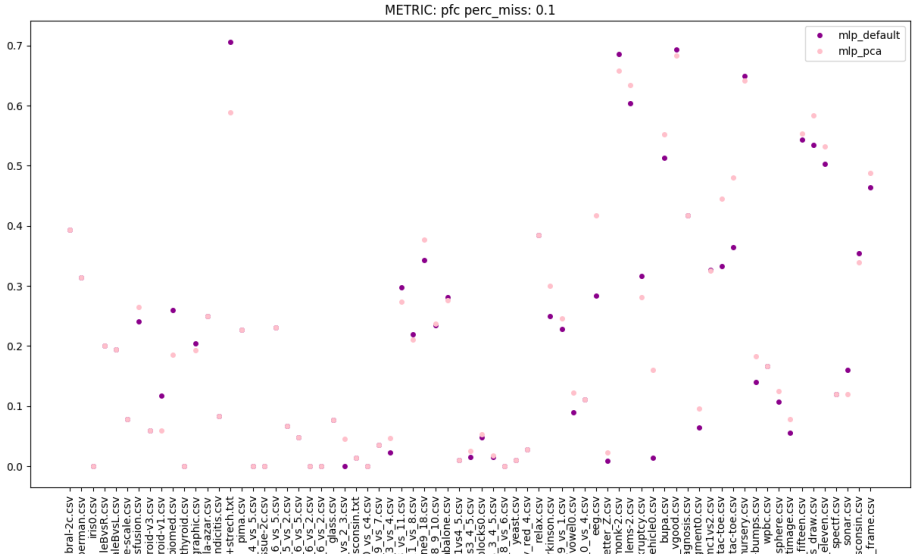



**Figura 4.12:** PFC da imputação com MLP para os *datasets* nos quais contêm uma variável com 10% de dados em falta.

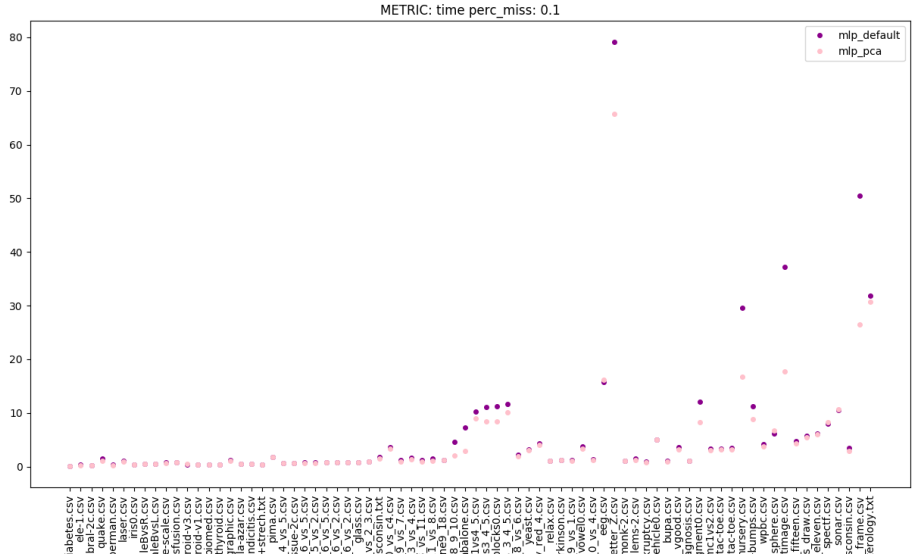



**Figura 4.13:** Tempo de execução da imputação com MLP para os *datasets* com 10% de dados em falta.

O esquema do *pipeline* pode ser visualizado na Figura 4.14. Este processo é repetido para cada variável a imputar. Na Figura 4.14, os blocos laranja representam funções implementadas e os blocos a verde representam as funções usadas diretamente de bibliotecas do *Python*.

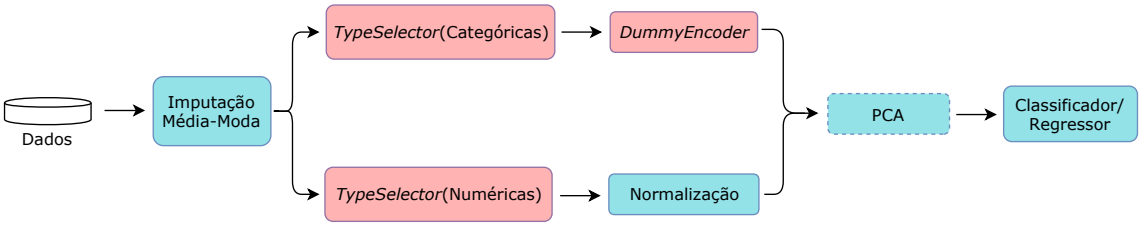



**Figura 4.14:** Representação da *pipeline* usada para imputação.

Ainda no módulo de imputação foram implementadas mensagens de texto sobre o estado da imputação. Estas mensagens podem ser de um de três tipos: mensagem de sucesso, erro ou de informação. Na Figura 4.15 é possível visualizar uma mensagem de sucesso. Já na Figura 4.16 está representado um exemplo de uma mensagem que a imputação ainda está em progresso. Estas mensagens são mensagens informativas, por isso diferem no símbolo das mensagens mostradas nas Figuras 4.8 e 4.9.

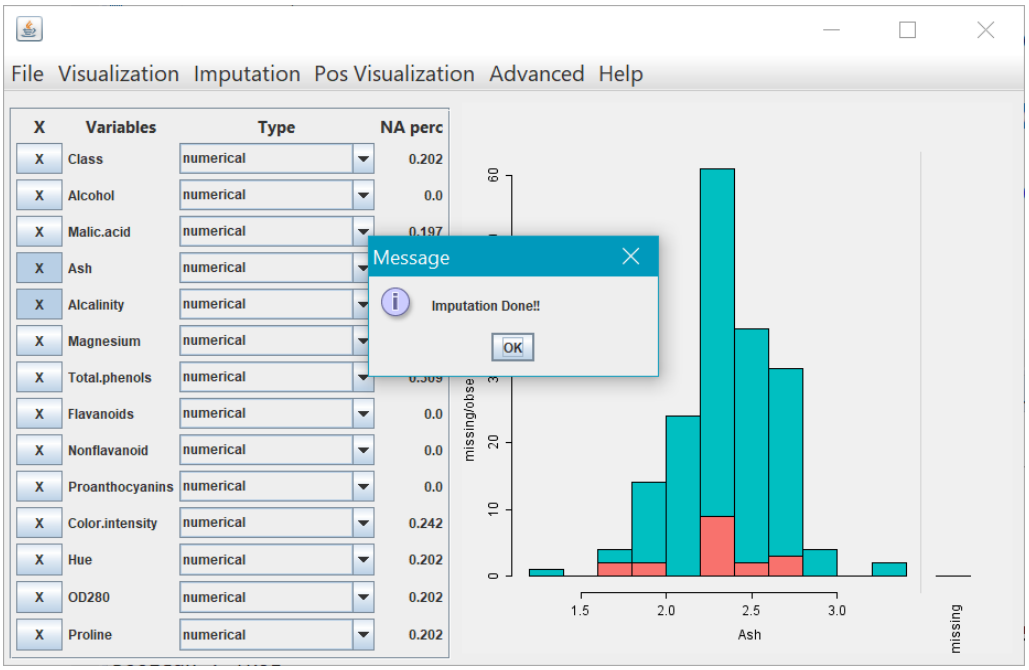



**Figura 4.15:** Exemplo de uma mensagem de sucesso da imputação.

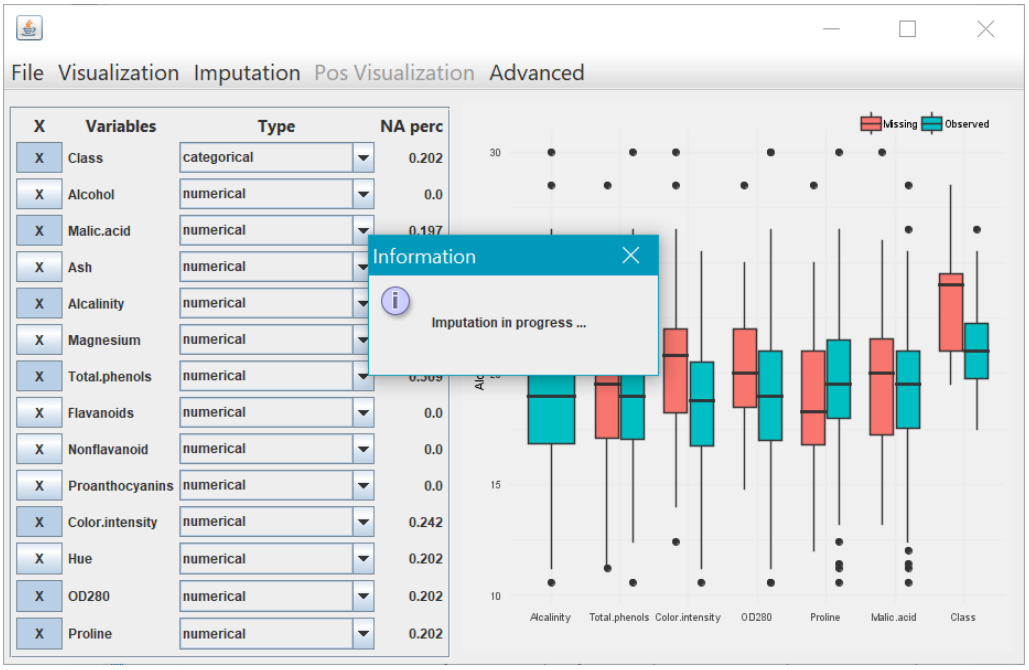



**Figura 4.16:** Exemplo de uma mensagem de informação quando imputação está em progresso.

## 4.4 Módulo da Interface

Este é o módulo onde todas as componentes e as linguagens se interligam.

A ligação entre a linguagem Java e R é feita através do *Renjin*. *Renjin* é um intérprete para a linguagem de programação R para computação estatística escrita em Java. Este intérprete permite executar o código em *R* dentro de aplicações Java. A maior vantagem do *Renjin* é que o próprio intérprete R é um módulo Java que pode ser perfeitamente integrado em qualquer aplicativo Java.

A comunicação entre a linguagem *Python* e *R* foi feita através da *package reticulate*. *Reticulate* é uma interface que permite importar módulos, correr funções e converter objetos de *Python* em *R* e vice-versa [53].

O GUI foi construído em Java com ajuda de duas bibliotecas gráficas: *Abstract Window Toolkit* (AWT) e Swing. Estas bibliotecas foram escolhidas por serem bastante simples de perceber e aplicar e por terem uma grande diversidade de componentes. Outra vantagem destas bibliotecas é a sua portabilidade, isto é, o aspeto das interfaces (cores, tamanhos) em diferentes sistemas operativos é exatamente o mesmo.

Os módulos de visualização e imputação foram implementados em *threads* distintas.

Este tipo de implementação teve como objetivo não deixar a interface bloqueada, isto é, o utilizador pode visualizar os diferentes métodos de visualização enquanto um método de imputação está em progresso. O facto dos métodos destes módulos estarem inseridos em *threads* diferentes também permite a interrupção de uma ação que esteja a decorrer para iniciar outra.

# 5

# Ferramenta Gráfica: Funcionalidades

Neste capítulo são detalhadas as funcionalidades da ferramenta gráfica desenvolvida: visualização, imputação e funcionalidades extra.

## 5.1 Funcionalidades de Visualização

A principal funcionalidade da ferramenta são os métodos de visualização. Estes métodos podem ser selecionados a partir do menu da visualização e pós-visualização (Figura 5.1).

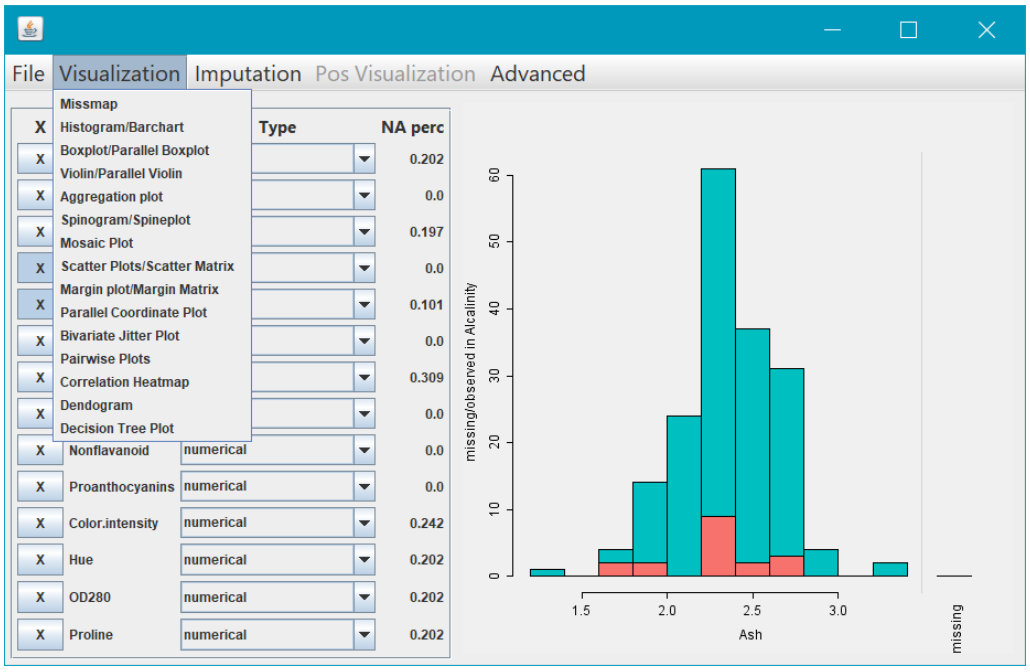



**Figura 5.1:** Menu dos gráficos de visualização.

Para ilustrar estas funcionalidades, utilizou-se um *dataset* real, recolhido do repositório UCI, onde foram gerados os três tipos de mecanismos em algumas das variáveis. Este *dataset* contém 14 variáveis resultantes da análise química de 176 vinhos cultivados (observações) numa região na Itália.

Para introduzir a falta de dados no *dataset* completo tiveram-se em consideração dois aspetos: percentagem e o mecanismo dos dados em falta. Os três tipos de mecanismos, MCAR, MAR e MNAR, foram implementados baseados na metodologia usada por Ali et al. [54].

Nesta subsecção decidiu-se não traduzir nenhum dos nomes dos gráficos para manter uma coerência, visto que alguns não tem tradução e porque serão os nomes usados na interface.

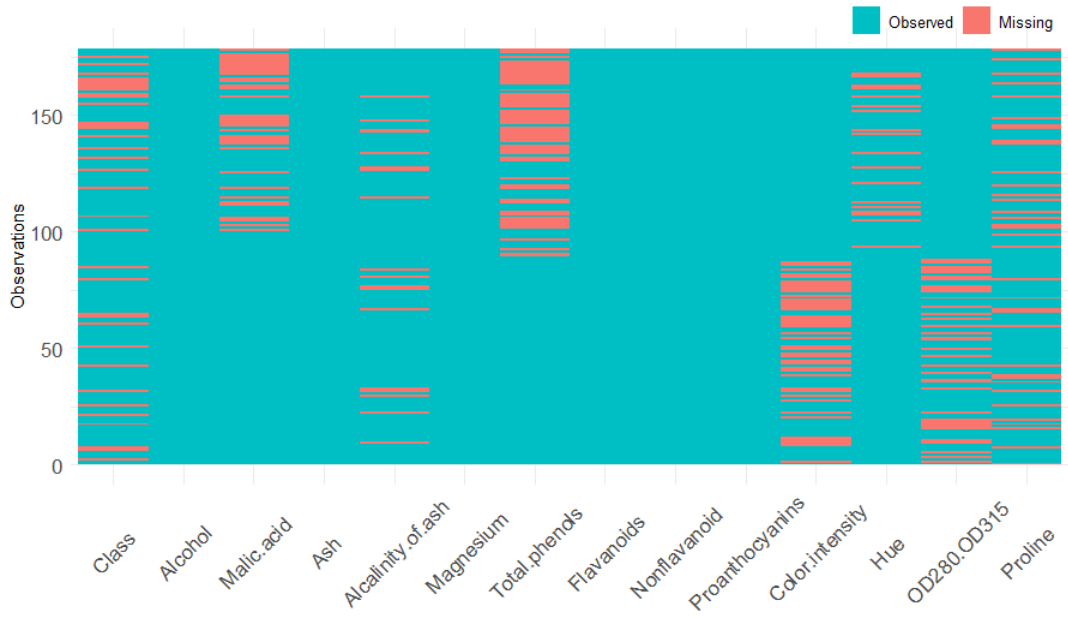



**Figura 5.2:** *MissMap* do *dataset wine*.

### *Missmap*

O *Missmap* providencia um sumário gráfico dos padrões de MD presente no *dataset*. Este tipo de gráfico mostra a posição dos dados em falta relativamente às variáveis e observações. A estrutura dos MD pode revelar padrões inesperados, como por exemplo, duas variáveis terem dados em falta para as mesmas observações. No entanto, os padrões de MD, por si só, não são o suficiente para detetar os mecanismos de MD que estão subjacentes aos conjuntos de dados. Na Figura 5.2 está representado um *missmap* do *dataset* em estudo, onde os dados observados são representados

por uma cor esverdeada e os dados em falta estão assinalados com cor alaranjada. Podem analisar-se alguns padrões de valores em falta. Por exemplo, os valores em falta na variável *Hue* ocorrem sempre para valores observáveis das variáveis *Color intensity* e *Proanthocyanins*. Este método de visualização foi implementado de raíz no ambiente *R* com a ajuda das bibliotecas *ggplot2* e *maggitr*.

### *Aggregation Plot*

Geralmente é interessante saber quantos valores ausentes contém cada variável. Ainda mais interessante, saber se há certas combinações de variáveis com um alto número de valores ausentes. A Figura 5.3 mostra essa informação. O *Aggregation Plot* é dividido em dois tipos de gráficos. O gráfico do lado esquerdo, é um *barchart* que mostra a percentagem de valores em falta por variável. Por exemplo, a variável *Class* contém, aproximadamente, 20% de valores em falta.

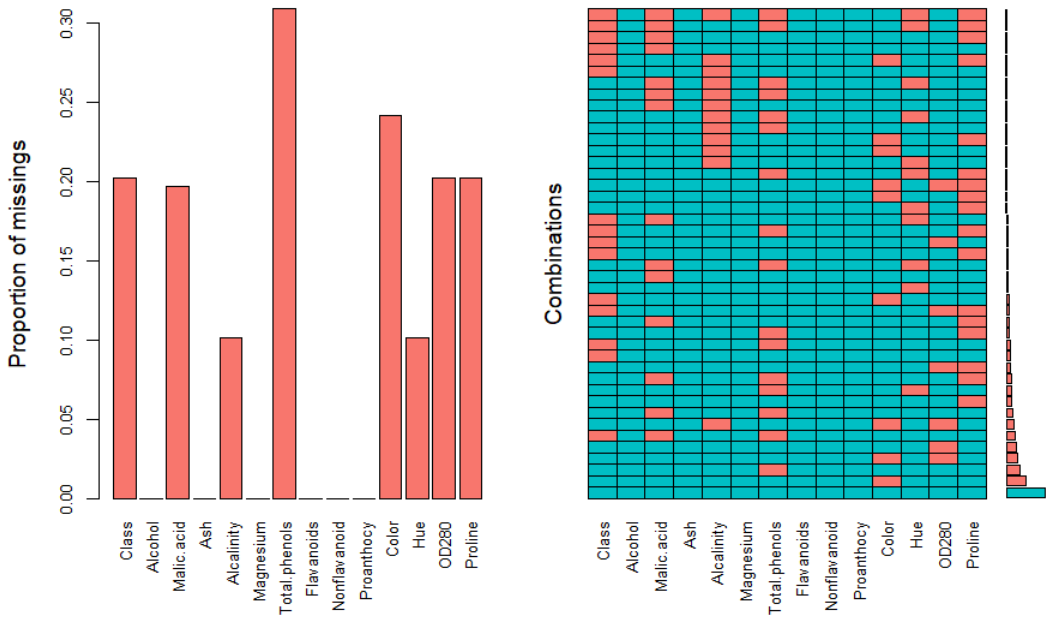



**Figura 5.3:** *Aggregation Plot* do *dataset wine*.

O gráfico do lado direito mostra para as mesmas variáveis (eixo horizontal) todas as combinações diferentes que estão presentes nas observações com valores ausentes e não ausentes (eixo vertical). A cor laranjada indica ausência, a cor esverdeada representa a presença de dados. As barras à direita representam as frequências de observações para as combinações correspondentes. Por exemplo, a terceira linha representa uma combinação em que só as variáveis *Class* e *Malic.acid* têm valores em falta. Outro exemplo: a última linha reflete as observações que não têm valores

ausentes em nenhuma variável, que é uma combinação que aparece bastante vezes (a maior barra).

### *Histogram/Barchart*

O *histogram* e o *barchart* são representações gráficas (de barras) que podem ser adaptadas para representar a quantidade de falta de dados numa ou mais variáveis relativamente aos valores de outra variável. Podem ainda ser adaptados para visualizar a quantidade de valores em falta na variável em análise através de uma barra extra à direita separada das restantes parcelas por um pequeno espaço (barra mais à direita na Figura 5.4). A diferença entre as duas representações é que o *histogram* é utilizado quando a variável de interesse é contínua e o *barchart* quando ela é categórica.

No lado direito da Figura 5.4 é possível observar um *histogram* da variável *Proanthocyanins* em que as barras estão separadas de acordo com o número de dados em falta (alaranjado) e valores observados (esverdeado) na variável *Malic.acid*. Já no lado direito, é possível observar um *barchart* da variável *Class*, uma variável categórica, relativamente à presença/ausência de valores nas variáveis *Hue* e *Malic.acid*. Além disso, é possível visualizar, na barra extra à direita, a quantidade de observações em que as variáveis estão simultaneamente em falta (laranja) em relação à quantidade de observações em que a variável em análise está em falta (amarelo).

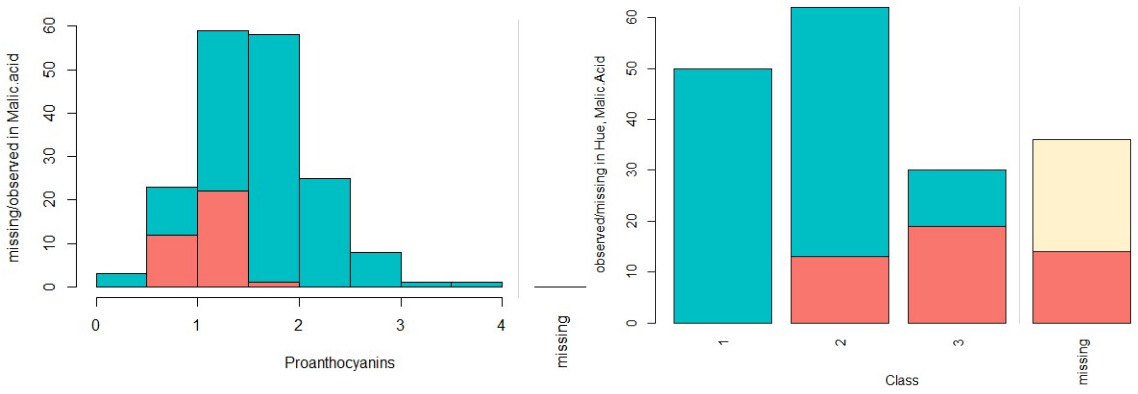



**Figura 5.4:** *Histogram* da variável *Proanthocyanins* (lado esquerdo) e *Barchart* da variável *Class* (lado direito).

### *Spinogram / Spineplot*

O *spinogram* e *spineplot* são uma extensão do *histogram* e *barchart*, respetivamente. A diferença entre as representações, é que neste tipo de representações gráficas, o eixo vertical representa a proporção de valores ausentes e observados em vez do número. Na Figura 5.5 mostra-nos um *Spinogram* e um *Spineplot* para as mesmas variáveis usadas na Figura 5.4.

É possível observar na Figura 5.4 e na Figura 5.5 que a ausência de valores na variável *Malic.acid* ocorre maioritariamente para valores mais baixos da variável *Proanthocyanins*. Isto significa que há uma certa dependência entre a presença/ausência dos valores na variável *Malic.acid* relativamente aos valores de *Proanthocyanins*, o que releva a presença do mecanismo MAR.

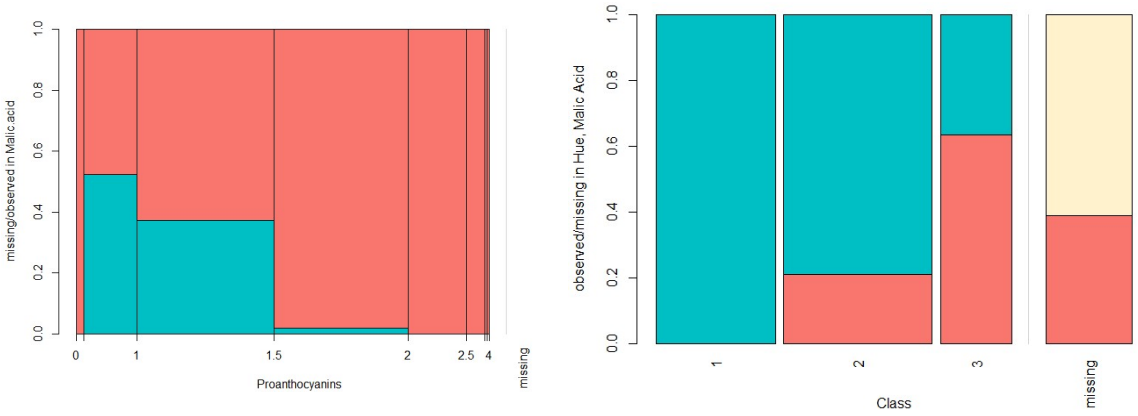



**Figura 5.5:** *Spinogram* (lado esquerdo) e *Spineplot* (lado direito).

### *Parallel Coordinate Plot*

Os *Parallel Coordinate Plot* são representações usadas para comparar os valores de dados de tipos ou magnitude completamente diferentes numa única visualização. Num *parallel coordinate plot*, cada observação do *dataset* normalizado é representado por uma linha, onde as variáveis são representadas por eixos paralelos. Este tipo de representação gráfica é ideal para comparar muitas variáveis ao mesmo tempo e ver as relações entre elas. Na Figura 5.6 é possível observar um subconjunto do *dataset wine* em que as linhas alaranjadas representam as observações que contem valores em falta na variável *Total.phenols*. Neste gráfico, é possível verificar que os valores em falta tendem a aparecer para valores específicos de *Flavanoids* (situação MAR) e que estão dispersos pelo resto das variáveis.

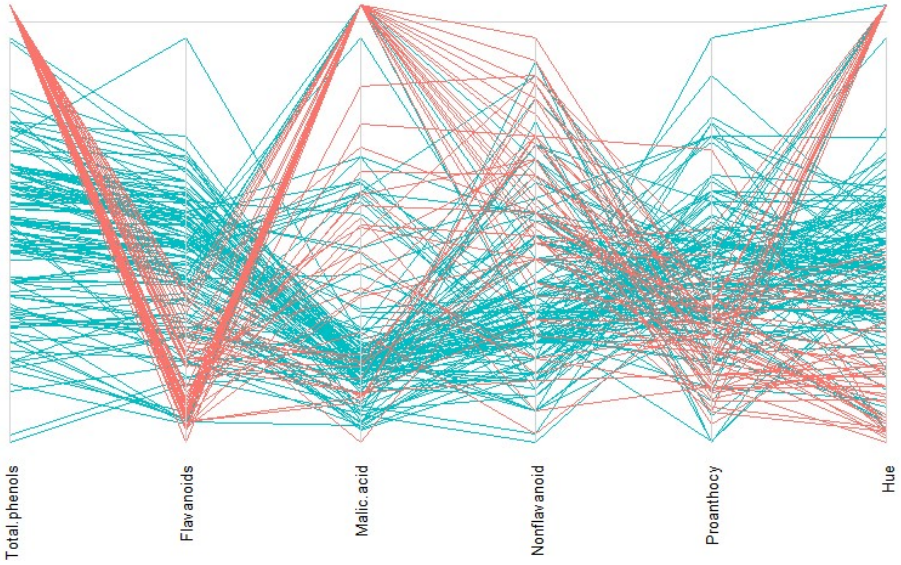



**Figura 5.6:** *Parallel Coordinate Plot* de um subconjunto do *dataset wine*.

### *Mosaic Plot*

Os *Mosaic Plot* é uma a extensão multidimensional dos *spineplots*, que exibem graficamente a mesma informação para apenas uma variável. Quando uma variável contínua é analisada os valores são separados em intervalos (como é feito nos *histograms* e *spineplots*). É possível observar na Figura 5.7 um *mosaic plot* das variáveis *Color.Intensity* (contínua) e *Class* (categórica). Neste gráfico é possível visualizar a relação entre a presença/ausência dos valores da variável *Class* (com cores esverdeadas e alaranjadas, respetivamente) relativamente aos valores da outra variável.

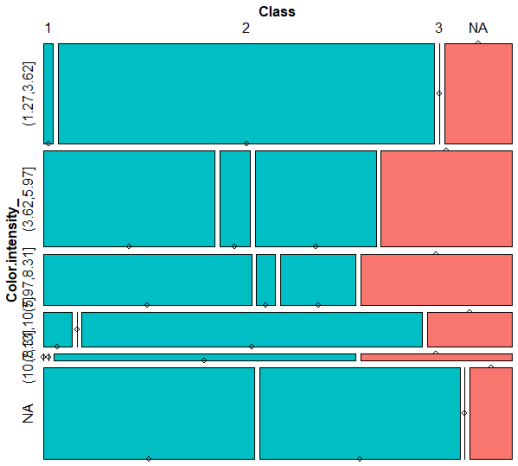



**Figura 5.7:** *Mosaic Plot* das variáveis *Class* e *Color.Intensity*.

### *Scatterplot/Scatterplot Matrices*

O *Scatterplot* é um tipo de gráfico de dispersão que usa coordenadas cartesianas para exibir valores de duas variáveis de um *dataset*. A essa representação pode ser adicionada informação sobre valores ausentes. A representação dos valores em falta é feita através de linhas tracejadas. Esta situação é verificada na parte direita da Figura 5.8, onde estão representada as posições dos valores em falta da variável *Hue* relativamente aos valores da variável *Malic.acid* através das linhas alaranjadas. Este tipo de representação é útil para verificar a distribuição dos dados e dos dados em falta. Os *Scatterplot Matrices* (Figura 5.8) são uma generalização do *scatterplot* para os casos multivariados.

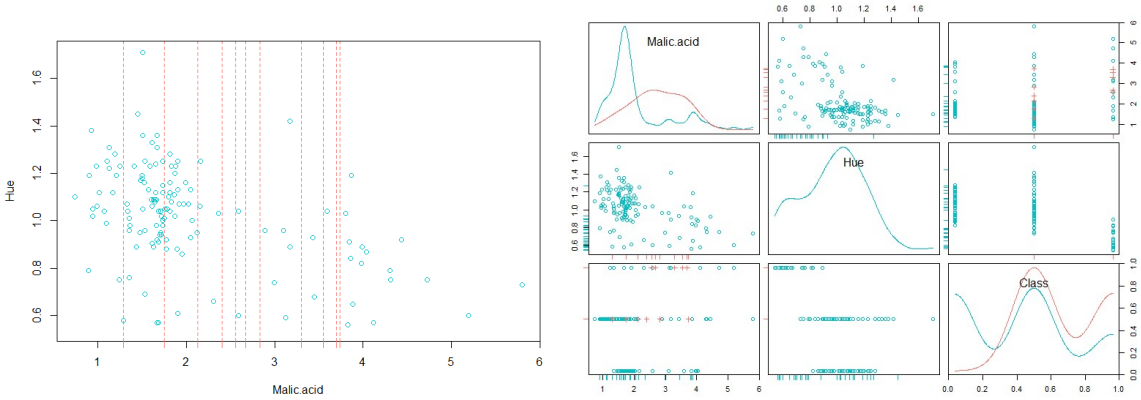



**Figura 5.8:** *Scatterplot* (lado esquerdo) das variáveis *Hue* e *Malic.acid* e *Scatterplot matrices* (lado direito) das variáveis *Hue*, *Malic.acid* e *Class*.

### Marginplot/Marginplot matrices

O *Marginplot* é um tipo de gráfico de dispersão semelhante ao *scatterplot*, sendo que este apresenta mais informação nas margens sobre os dados em falta. Considere o gráfico à esquerda da Figura 5.9, que mostra uma representação deste gráfico para as mesmas variáveis que o *scatterplot*. A diferença desta representação relativamente ao *scatterplot* é que margens contêm *boxplots* que mostram a distribuição dos valores em falta e observados para cada variável, e a localização dos valores em falta são representados por pontos em vez de linhas a tracejado. Este gráfico também contém na margem informação sobre o número de valores em falta de cada variável. Por exemplo, a variável *Hue* contêm 18 valores em falta. À semelhança do *scatterplot*, também o *marginplot* pode ser generalizado para mais que duas variáveis através de *marginplot matrices* (Figura 5.9 à direita).

Todos os métodos de visualização descritos acima, com a exceção do *Missmap*, foram implementados através da biblioteca *VIM* em *R*.

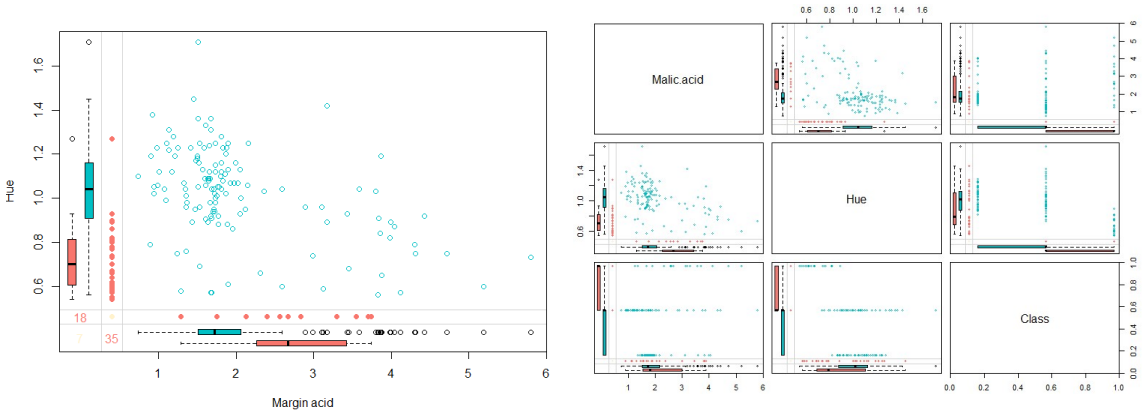



**Figura 5.9:** *Marginplot* (lado esquerdo) das variáveis *Hue* e *Malic.acid* e *Marginplot matrices* (lado direito) das variáveis *Hue*, *Malic.acid* e *Class*.

### *Densityplot*

O *Densityplot* é um gráfico de densidade que permite visualizar a distribuição dos dados de uma variável contínua. Este gráfico é uma variação do histograma. Os picos de um gráfico de densidade ajudam a exibir onde os valores são concentrados no intervalo. Na Figura 5.10 está representado o gráfico de densidade da variável *Proanthocyanins* dividida de acordo com os valores ausentes (a laranja) e presentes (a verde) na variável *Malic.acid*.

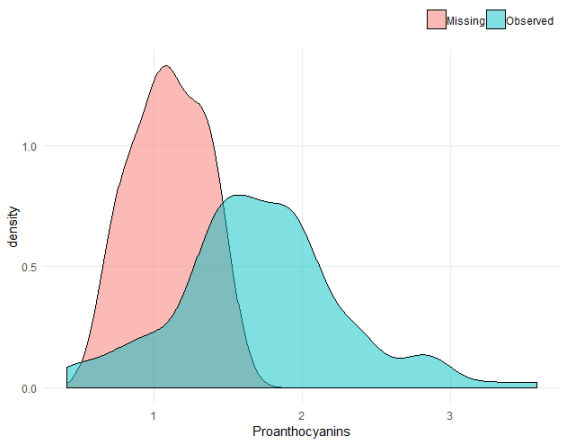



**Figura 5.10:** *Densityplot* da variável *Proanthocyanins* agrupada de acordo com os dados em falta (a laranja) e presentes (a verde) na variável *Malic.acid*

### *Boxplot / Paralell Boxplots*

O *Boxplot* é um diagrama de extremos e quartis usado para representar a variação de dados observados de uma variável numérica por meio de quartis. Este tipo de representação pode ser usada para visualizar se existe uma relação entre ausência/presença de dados numa variável e os dados observados noutra variável. Isto pode ser estendido para uma comparação das informações de falta de várias variáveis (*parallel boxplots*).

A Figura 5.11 mostra os valores da variável *Ash* na forma de um *boxplot* (à esquerda). Os outros *boxplots* mostrados na figura também se referem aos valores da variável *Ash*, mas agrupados de acordo com os dados em falta (alaranjados) ou observados (esverdeado) de cada variável presente. Alguns dos *boxplots* mostram uma clara dependência entre a magnitude dos valores de *Ash* e a presença de dados em falta. Por exemplo, valores ausentes na variável *Color intensity* aparecem especialmente para valores mais baixos de *Ash*. Isto indica uma situação clara do MAR presente na variável *Color intensity*.

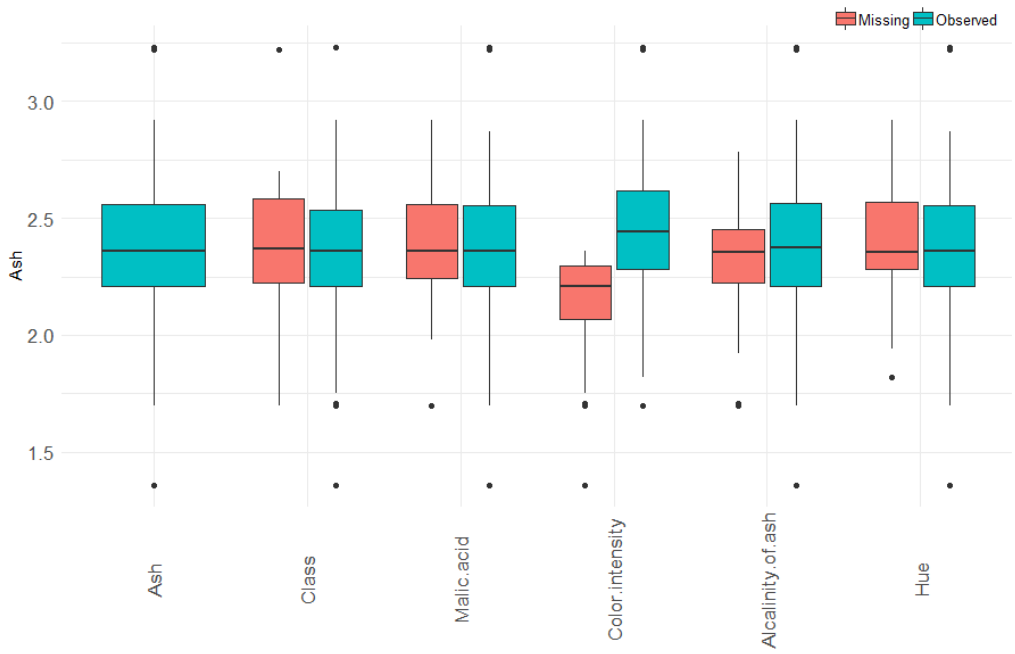



**Figura 5.11:** *Parallel Boxplot* de um subcojunto do *dataset wine* relativamente as valores da variável *Ash*.

### *Violinplot / Paralell Violinplot*

O *Violinplot* é um diagrama em forma de violino usado para visualizar a distribuição dos dados e sua densidade de probabilidade. Este tipo de representação gráfica é

uma combinação entre o *boxplot* e o gráfico de densidades, isto é, ao contrário de um *boxplot*, no qual todos os componentes do gráfico correspondem a pontos de dados reais, o gráfico de violino apresenta uma estimativa da densidade do núcleo da distribuição subjacente. À semelhança do *boxplot*, este também pode ser utilizado na comparação de informação em várias variáveis. A Figura 5.12 mostra-nos um *Parallel Violinplot* para as mesmas variáveis usadas na Figura 5.4. Neste exemplo, observam-se as mesmas dependências descritas anteriormente. Este gráfico e os dois anteriores foram implementados de raíz em ambiente *R* com a ajuda das mesmas bibliotecas que o método *Missmap*.

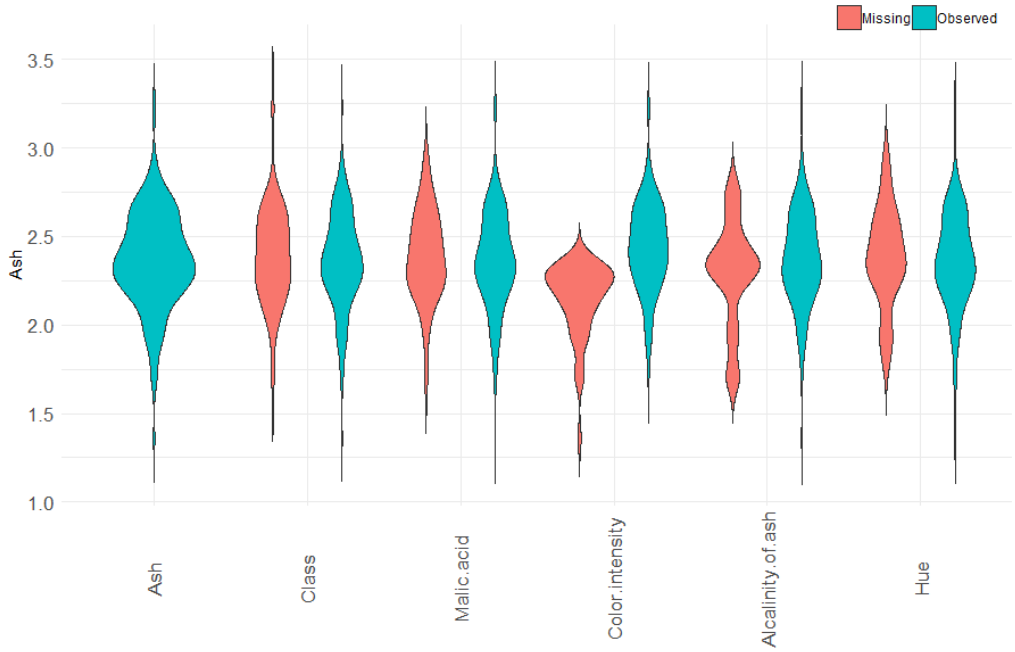



**Figura 5.12:** *Parallel Violin* de um subconjunto do *dataset wine* relativamente as valores da variável *Ash*.

### *Correlation Heatmap*

A Figura 5.13 mostra um *correlation heatmap*, implementado pela biblioteca *missingno* em *Python*. Este tipo de representação gráfica mede a correlação da falta de dados, isto é, quão fortemente a presença ou ausência de uma variável afeta a presença de outra. A correlação da falta de dados varia de -1 (se uma variável aparece a outra definitivamente não) a 0 (variáveis aparecendo ou não aparecendo não têm efeito uma na outra), e de 0 a 1 (se uma variável aparecer, a outra definitivamente também). Este método de visualização requer que o *dataset* a analisar contenha pelo menos duas variáveis com dados em falta. A Figura 5.13 mostra a correlação entre as variáveis em falta do *dataset wine*.

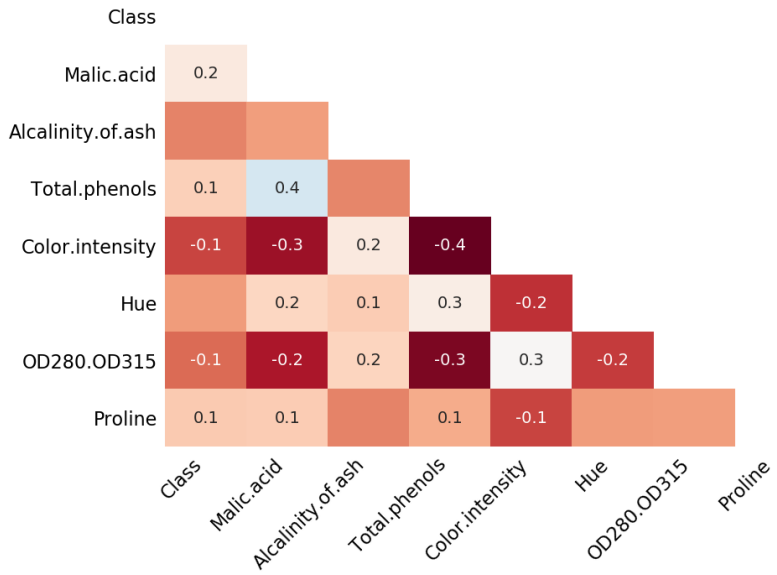



**Figura 5.13:** *Correlation Heatmap* das variáveis com dados em falta do *dataset Wine*.

### *Dendogram*

O *Dendogram* é um diagrama de árvore frequentemente usado para ilustrar o arranjo dos aglomerados produzidos pelo agrupamento hierárquico. A Figura 5.14 mostra um diagrama de árvore disponível na biblioteca *missingno* em *Python*. Este gráfico utiliza as correlações dos dados em falta (Figura 5.13) para fazer os agrupamentos das variáveis.

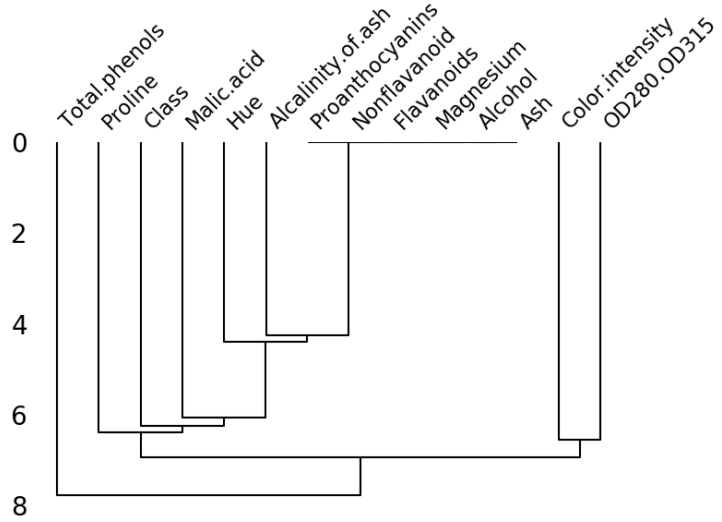



**Figura 5.14:** *Dendogram* do *dataset Wine*.

***Decision Tree Plot***

As Árvores de decisão (*Decision Tree* (DT)) são uma técnica não-paramétrica muito usada para a deteção da estrutura dos dados (descrita mais detalhadamente na Secção 2.2) [55]. Tierney et al., em 2015, propôs o uso de DT para estudar a estrutura dos valores em falta nos conjuntos de dados [56]. Esta metodologia foi, em 2018, implementada e encontra-se disponível numa biblioteca do *R*: *Naniar*. A Figura 5.15 mostra o exemplo deste método de visualização para um subconjunto do *dataset wine*, onde a "Prop.Miss" indica a percentagem de valores em falta nas variáveis em análise e "n" representa o número de observações.

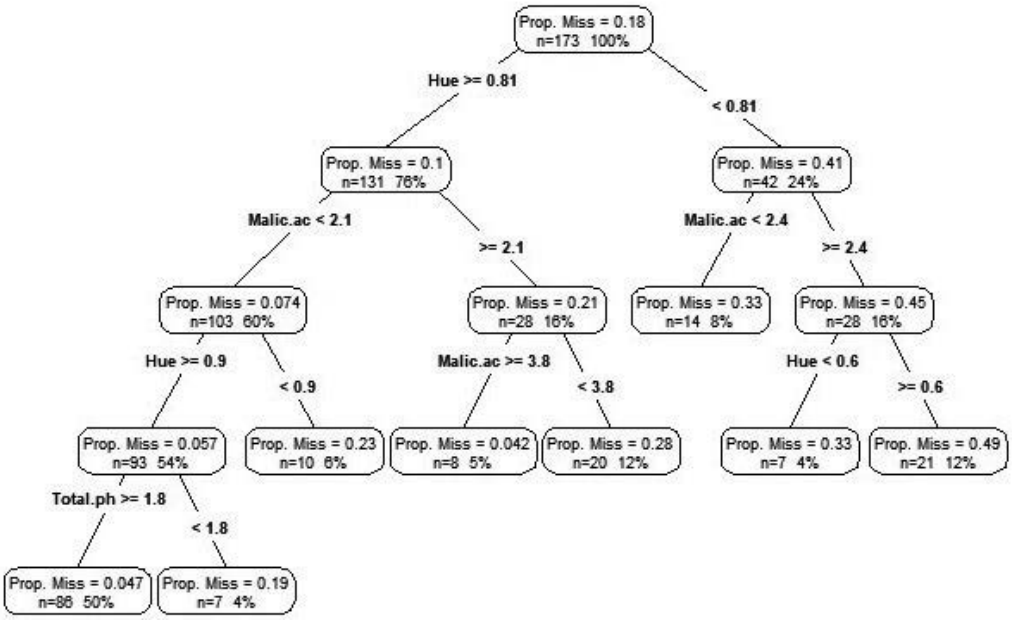



**Figura 5.15:** *Decision Tree Plot* das variáveis *Hue*, *Malic.acid* e *Total.phenols*.

## 5.2 Funcionalidades de Imputação

As funcionalidades básicas de imputação podem ser acedidas através do menu de imputação na interface (Figura 5.16). Os métodos de imputação (M-M, Regressão, DT, SVM, MLP e kNN) foram implementados com parâmetros *default* das funções usadas. Contudo, alguns desses parâmetros podem ser alterados pelo utilizador nas opções avançadas, descritas na Secção 5.3.

Estas funcionalidades podem ser ilustradas através das funcionalidades de visualização. Nesse sentido, imputou-se o *dataset wine* com o método de imputação SVM e M-M para ilustrar o impacto dos métodos no *datasets*.

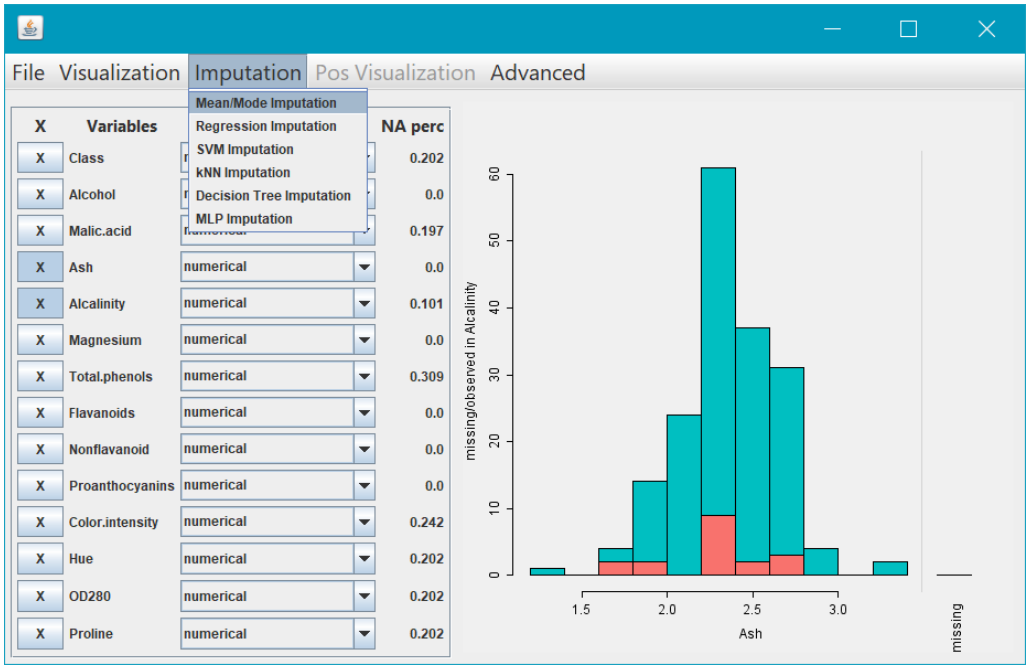



**Figura 5.16:** Menu dos métodos de imputação.

A Figura 5.17 demonstra dois *histogram* da variável *Malic.acid* após a imputação com SVM (à esquerda) e M-M (à direita). A laranja é possível observar a distribuição dos valores imputados nessa variável.

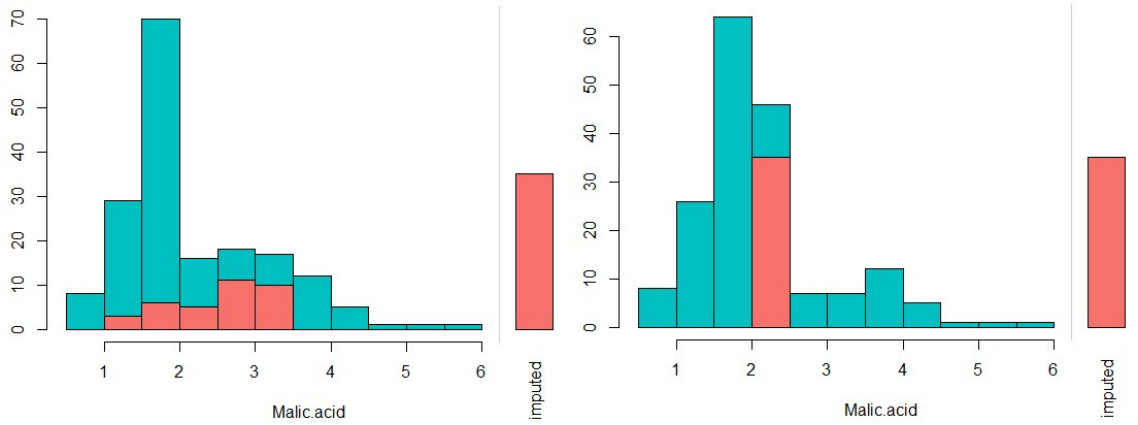



**Figura 5.17:** *Histogram* da variável *Malic.acid* posteriormente à imputação com SVM (à esquerda) e M-M (à direita).

Outro exemplo é visível na Figura 5.18 onde estão representadas dois *marginplots*, referentes à imputação com SVM e M-M, das variáveis *Malic.acid* e *Alcalinity*. Neste exemplo, a cor laranja representa os valores imputados de cada variável e a cor preta os valores imputados coincidentes em ambas as variáveis.

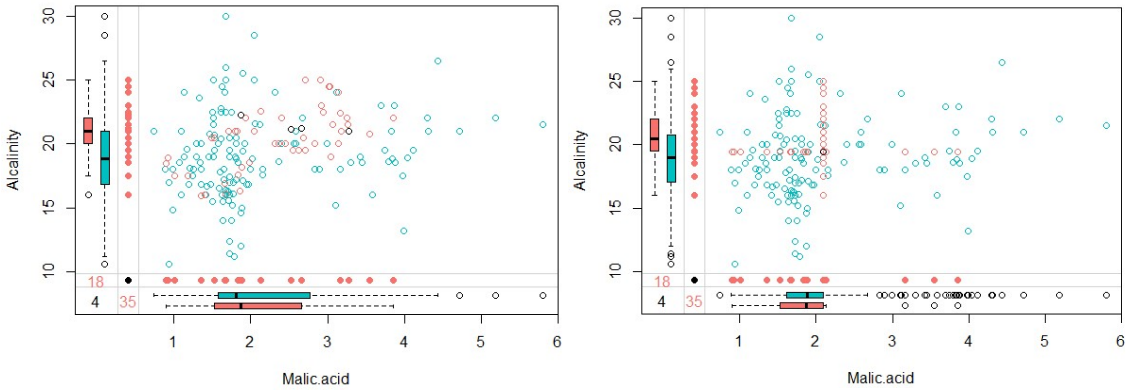



**Figura 5.18:** *Marginplot* das variáveis *Alcalinity* e *Mali.acid* imputadas com SVM (à esquerda) e M-M (à direita).

Para além das funcionalidades mencionadas, o módulo de imputação contém a possibilidade de guardar o *dataset* criado após as imputações no ficheiro de formato *csv*. Essa funcionalidade posso ser acedida através do menu "*File*", como se pode observar na Figura 5.19.

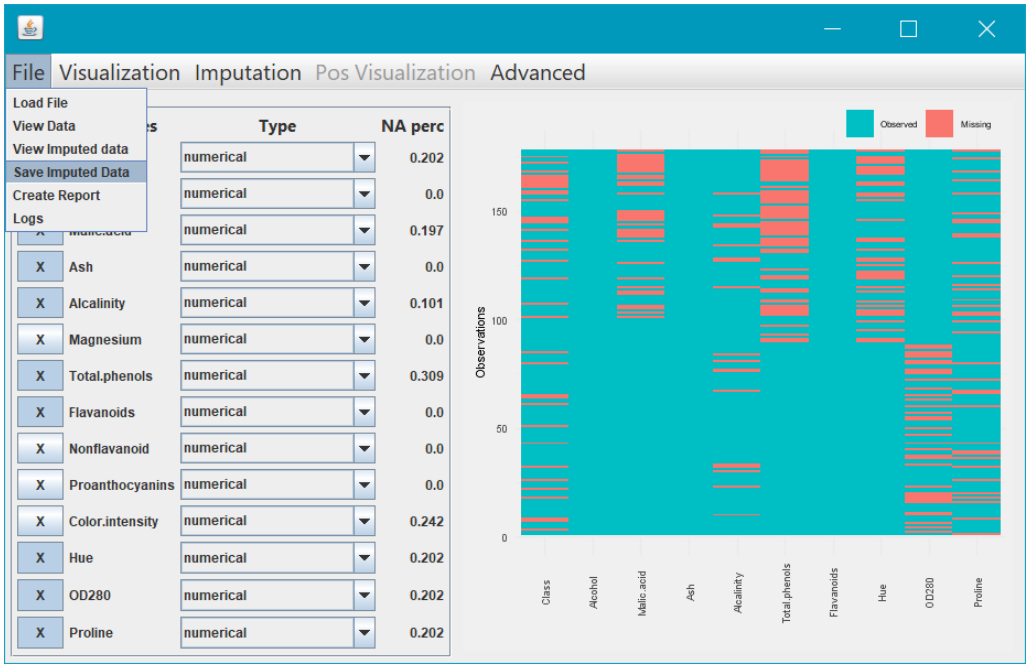



**Figura 5.19:** Menu da Entrada.

## 5.3 Funcionalidades Extra

Para além das funcionalidades básicas requeridas, a interface inclui opções avançadas que permitem um domínio maior por parte de um utilizador experiente. Nesta subsecção estão descritas essas funcionalidades opcionais dos módulos de visualização e imputação, denominadas por funcionalidades extra (FE).

### FE1: Guardar gráficos gerados

Ao módulo de visualização foi adicionada a funcionalidade do utilizador poder guardar os gráficos gerados pelos métodos de visualização. Para efetuar esta opção o utilizador apenas precisa de clicar com o botão do lado esquerdo no gráfico. O gráfico é guardado como uma imagem e é permitida a escolha de uma de duas extensões: *png* ou *jpeg*.

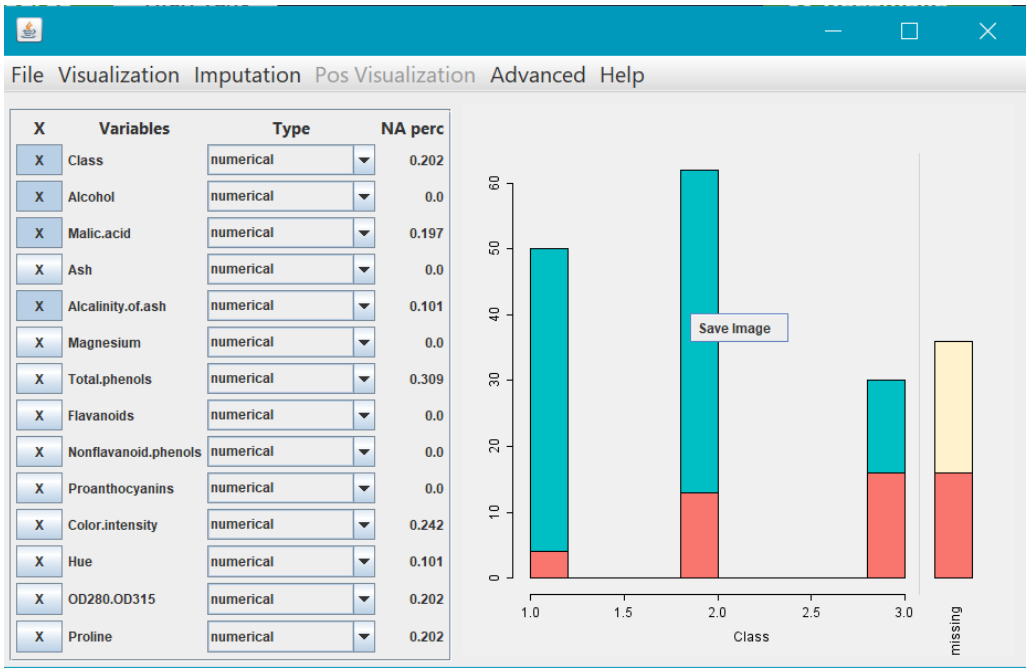



**Figura 5.20:** Exemplificação do modo de guardar uma representação gráfica numa imagem.

### FE2: Relatório Automático

Esta funcionalidade permite ao utilizador criar e guardar um relatório com informações estatísticas dos conjuntos de dados. Esse relatório inclui informação

sobre o *dataset* original (se o utilizador o quiser incluir) e sobre os conjuntos de dados imputados selecionados pelo utilizador (Figura 5.21).

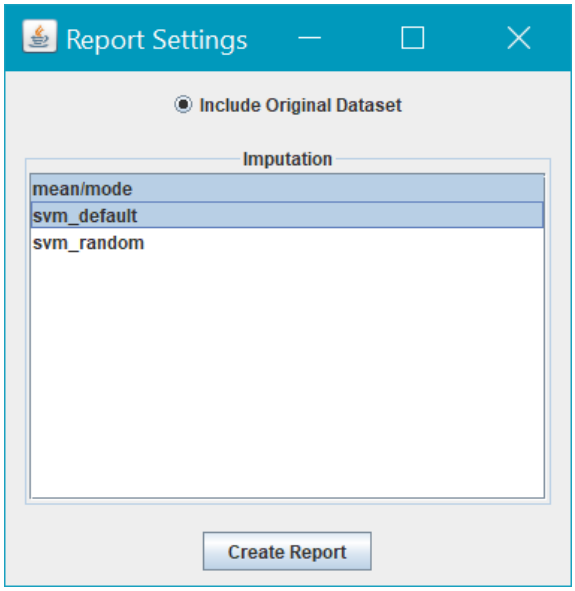



**Figura 5.21:** Painel para personalizar o relatório estatístico.

O relatório pode ainda ser dividido em três tipos de informação disponibilizada:

- Descrição geral do *dataset* original;
- Estatística descritiva das variáveis (medidas e representações gráficas);
- Informações consideradas importantes ou anómalas.

A informação geral é exclusiva do *dataset* original e inclui a quantidade de variáveis e observações; número e percentagem de valores em falta; número e percentagem de observações com valores em falta e quantidade de cada tipo de variável presentes.

Seguidamente é apresentada uma estatística descritiva para cada variável separadamente. O conteúdo dessa análise estatística varia consoante o tipo de variável. Se a variável for numérica inclui medidas de localização de tendência central e não-central (média, 1ºQuartil, mediana e 3ºQuartil), medidas de dispersão (mínimo, máximo, desvio-padrão, variância, intervalo de variação e intervalo interquartis), medidas de caracterização de distribuições (*skewness* e *kurtosis*) e ainda representações gráficas (histograma e *boxplot*). No caso da variável ser categórica a informação dada inclui uma medida de localização de tendência central (moda), medidas de quantidade e frequência de cada categoria, o número de categorias diferentes e ainda uma representação gráfica (histograma). Ambas incluem a quantidade e percentagem de valores em falta.

No *dataset* original esta estatística descritiva é feita para todas as variáveis presentes no *dataset* à exceção das variáveis do tipo *id*. Nos *datasets* imputados a análise é feita para as variáveis imputadas (que continham valores em falta). Para além da informação geral e da análise estatística, o relatório também inclui informação considerada importante para o utilizador. No *dataset* original são dadas informações das variáveis que contêm uma correlação entre elas superior a 80%, o nome das variáveis que foram rejeitadas da análise por serem do tipo *id* e a informação das variáveis que têm valores em falta. Na imputação são geradas mensagens de alerta caso os valores das medidas de média se distanciem muito dos valores originais (antes da imputação).

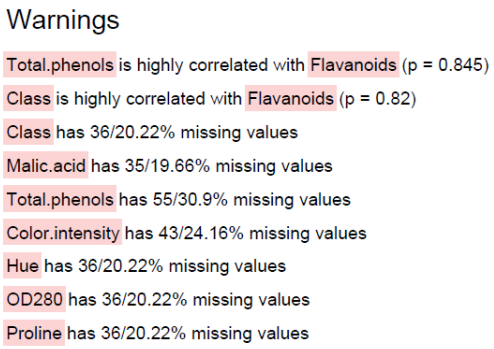



**Figura 5.22:** Exemplo de algumas mensagens de aviso apresentadas no relatório estatístico.

No Anexo A é possível encontrar um exemplo de um relatório criado pela interface. Esse relatório inclui a informação do *dataset* wine e dos *datasets* imputados com a M-M e SVM com os parâmetros *default*. Este relatório permite observar as diferenças estatísticas dos dois métodos de imputação relativamente ao *dataset* original.

## FE3: Configurações avançadas da imputação

Os métodos de imputação baseados em técnicas ML possuem hiperparâmetros que podem ser alterados para melhorar a sua performance. A alteração dos hiperparâmetros pode ser feita por três técnicas distintas, descritas na Secção 2.4: otimização manual, *grid search* e *random search*. Para a otimização *grid search* e *random search*, um número definido de parâmetros é fornecido e o melhor classificador é escolhido com base nas combinações de diferentes parâmetros e em função da pontuação. Na Tabela 5.1 são apresentados os valores dos parâmetros usados nas otimizações *grid search* e *random search*. Na Figura 5.23 está representado o painel das configurações dos métodos de imputação, que pode ser acedido através do menu "*Advanced*". A

figura mostra as opções disponíveis para parametrizar o método SVM.

**Tabela 5.1:** Valores do parâmetros usados na otimização dos diferentes métodos de imputação.

| Métodos de Imputação | Parâmetros | Valores |
|---|---|---|
| SVM | kernel | ['rfc', 'linear', 'poly'] |
| | *C* | [1, 10, 50, 100] |
| | *gamma* | [25,50,75,'auto'] |
| DT | *max_depth* | [1, 3, 5, 10, 15] |
| | *max_features* | [0.1, 0.2, 0.3, 0.5, 0.6] |
| MLP | *hidden_layer_sizes* | [(7,), (128,),(7, 7)] |
| | *tol* | [1e-2, 1e-3, 1e-5, 1e-6] |
| kNN | *n_neighbors* | [1,3,5,10,15,20] |

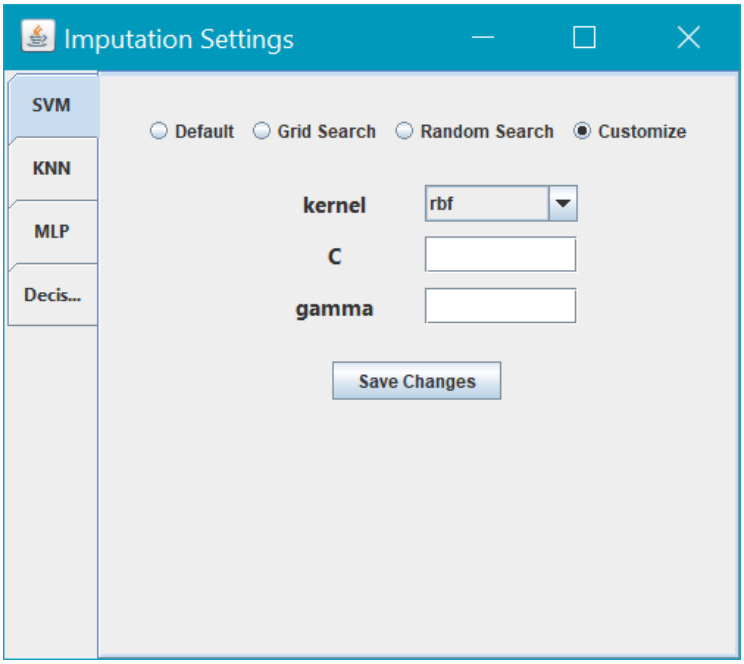



**Figura 5.23:** Painel para modificar os parâmetros dos métodos de imputação.

# 6

# Conclusão

Os dados em falta são um problema comum na análise de dados que pode ter consequências prejudicais, tais como a redução do número de observações para análise e a distorção dos modelos que se pretendem criar.

Diversas técnicas tem sido utilizadas para tentar lidar com este problema, sendo a imputação uma das abordagens mais comuns. Contudo, o desempenho dos diversos métodos de imputação está dependente das caraterísticas e propriedades dos dados. A exploração visual dos dados em falta permite a análise das suas caraterísticas e estrutura, orientando deste modo a escolha do método de imputação mais adequado.

O principal objetivo desta tese é a criação de uma ferramenta simples e intuitiva que integre todos os métodos de visualização existentes em interfaces de visualização do estado da arte e alguns dos principais métodos de imputação usados atualmente.

A ferramenta foi desenvolvida tendo em vista dois perfis distintos de utilizadores: iniciantes ou utilizadores de outros domínios (não-especialistas, por exemplo, médicos) e especialistas da área. Os iniciantes podem explorar as caraterísticas básicas dos dados em falta, tais como a sua distribuição e percentagem, e utilizar os diferentes métodos de imputação com os parâmetros *default*. Os especialistas podem procurar detetar os diferentes mecanismos e padrões dos dados em falta, imputar os dados com os diversos métodos implementados (podendo otimizar os seus parâmetros através das opções avançadas), visualizar o impacto das imputações nos *datasets* e avaliar as diferentes imputações através da análise estatística e mensagens de alerta proporcionada pelo relatório automático.

Além disso, a ferramenta foi desenhada para conseguir suportar a análise de dados de grande dimensionalidade. Este objetivo foi conseguido através da aplicação do método de redução PCA em *datasets* com grandes dimensões. Os resultados obtidos mostram que este método reduz o tempo computacional dos métodos em *datastes* com mais de 15 variáveis.

Foi ainda realizado um caso de estudo com o uso do *dataset wine* com o intuito de validar as diversas funcionalidades da ferramenta. O estudo efetuado permitiu demonstrar a deteção de padrões e possíveis tipos de mecanismos presentes nos dados, também como visualizar o impacto de dois métodos de imputação nos dados através dos diferente métodos de visualização e da análise estatística proporcionada no relatório automático.

Como trabalho futuro, a ferramenta poderá incluir testes estatísticos de modo enriquecer a exploração visual, podendo as duas técnicas complementar-se. Por exemplo, a visualização dos dados em falta pode confirmar os resultados obtidos pelos testes estatísticos ou servir para verificar os pressupostos necessários para a aplicação de determinados testes estatísticos.

# Bibliografia


[1] G. Fitzmaurice, "Missing data: implications for analysis," *Nutrition*, vol. 24, no. 2, pp. 200–202, 2008.

[2] D. B. Rubin, *Multiple imputation for nonresponse in surveys*, vol. 81. John Wiley & Sons, 2004.

[3] C.-Y. J. Peng, M. Harwell, S.-M. Liou, L. H. Ehman, *et al.*, "Advances in missing data methods and implications for educational research," *Real data analysis*, vol. 3178, 2006.

[4] J. L. Schafer, *Analysis of incomplete multivariate data*. Chapman and Hall/CRC, 1997.

[5] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*, vol. 333. John Wiley & Sons, 2014.

[6] K. Mohan, J. Pearl, and J. Tian, "Graphical models for inference with missing data," in *Advances in neural information processing systems*, pp. 1277–1285, 2013.

[7] M. S. Santos, P. H. Abreu, P. J. García-Laencina, A. Simão, and A. Carvalho, "A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients," *Journal of biomedical informatics*, vol. 58, pp. 49–59, 2015.

[8] P. J. García-Laencina, J.-L. Sancho-Gómez, and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review," *Neural Computing and Applications*, vol. 19, no. 2, pp. 263–282, 2010.

[9] A. F. Costa, M. S. Santos, J. P. Soares, and P. H. Abreu, "Missing data imputation via denoising autoencoders: the untold story," *Symposium on Intelligent Data Analysis 2018*.

[10] D. A. Bennett, "How can i deal with missing data in my study?," *Australian and New Zealand journal of public health*, vol. 25, no. 5, pp. 464–469, 2001.

[11] M. S. Santos, J. P. Soares, P. H. Abreu, H. Araújo, and J. Santos, "Influence of data distribution in missing data imputation," in *Conference on Artificial Intelligence in Medicine in Europe*, pp. 285–294, Springer, 2017.

[12] J. P. Soares, M. S. Santos, P. H. Abreu, H. Araújo, and J. Santos, "Exploring the effects of data distribution in missing data imputation," *Symposium on Intelligent Data Analysis 2018*.

[13] B. van Stein, W. Kowalczyk, and T. Bäck, "Analysis and visualization of missing value patterns," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp. 187–198, Springer, 2016.

[14] P. H. Abreu, M. S. Santos, M. H. Abreu, B. Andrade, and D. C. Silva, "Predicting breast cancer recurrence using machine learning techniques: a systematic review," *ACM Computing Surveys (CSUR)*, vol. 49, no. 3, p. 52, 2016.

[15] A. Unwin, G. Hawkins, H. Hofmann, and B. Siegl, "Interactive graphics for data sets with missing values—manet," *Journal of Computational and Graphical Statistics*, vol. 5, no. 2, pp. 113–122, 1996.

[16] D. F. Swayne and A. Buja, "Missing data in interactive high-dimensional data visualization," *Computational Statistics*, vol. 13, no. 1, pp. 15–26, 1998.

[17] M. Templ and P. Filzmoser, "Visualization of missing values using the r-package vim," *Reserach report cs-2008-1, Department of Statistics and Probability Therory, Vienna University of Technology*, 2008.

[18] X. Cheng, D. Cook, and H. Hofmann, "Missingdatagui: A gui for missing data exploration," 2016.

[19] M. Templ, A. Alfons, and P. Filzmoser, "Exploring incomplete data using visualization techniques," *Advances in Data Analysis and Classification*, vol. 6, no. 1, pp. 29–47, 2012.

[20] D. B. Rubin, "Inference and missing data," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976.

[21] C. K. Enders, *Applied missing data analysis*. Guilford Press, 2010.

[22] T. Sivapriya, A. N. B. Kamal, and V. Thavavel, "Imputation and classification of missing data using least square support vector machines–a new approach in dementia diagnosis," *International Journal of Advanced Research in Artificial Intelligence*, vol. 1, no. 4, pp. 29–33, 2012.

[23] K. Pelckmans, J. De Brabanter, J. A. Suykens, and B. De Moor, "Handling missing values in support vector machine classifiers," *Neural Networks*, vol. 18, no. 5-6, pp. 684–692, 2005.

[24] F. Honghai, C. Guoshun, Y. Cheng, Y. Bingru, and C. Yumei, "A svm regression based approach to filling in missing values," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pp. 581–587, Springer, 2005.

[25] L. Nanni, A. Lumini, and S. Brahnam, "A classifier ensemble approach for the missing feature problem," *Artificial intelligence in medicine*, vol. 55, no. 1, pp. 37–50, 2012.

[26] L. Rokach and O. Z. Maimon, *Data mining with decision trees: theory and applications*, vol. 69. World scientific, 2008.

[27] K. Lakshminarayan, S. A. Harp, R. P. Goldman, T. Samad, *et al.*, "Imputation of missing data using machine learning techniques.," in *KDD*, pp. 140–145, 1996.

[28] J. M. Jerez, I. Molina, P. J. García-Laencina, E. Alba, N. Ribelles, M. Martín, and L. Franco, "Missing data imputation using statistical and machine learning methods in a real breast cancer problem," *Artificial intelligence in medicine*, vol. 50, no. 2, pp. 105–115, 2010.

[29] M. M. Rahman and D. N. Davis, "Fuzzy unordered rules induction algorithm used as missing value imputation methods for k-mean clustering on real cardiovascular data," *Lect Notes Eng Comput Sci*, vol. 2197, no. 1, pp. 391–4, 2012.

[30] S. Azim and S. Aggarwal, "Hybrid model for data imputation: using fuzzy c means and multi layer perceptron," in *Advance Computing Conference (IACC), 2014 IEEE International*, pp. 1281–1285, IEEE, 2014.

[31] M. Humphries, "Missing data & how to deal: An overview of missing data," *Population Research Center. University of Texas. Recuperado de: http://www. google. com/url*, pp. 39–41, 2013.

[32] S. Van Buuren, *Flexible imputation of missing data*. Chapman and Hall/CRC, 2018.

[33] B. Schölkopf, K. Tsuda, J.-P. Vert, D. S. Istrail, P. A. Pevzner, M. S. Waterman, *et al.*, *Kernel methods in computational biology*. MIT press, 2004.

[34] N. Cristianini, J. Shawe-Taylor, *et al.*, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[35] C. Bishop, C. M. Bishop, *et al.*, *Neural networks for pattern recognition*. Oxford university press, 1995.

[36] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.

[37] N. Pinto, D. Doukhan, J. J. DiCarlo, and D. D. Cox, "A high-throughput screening approach to discovering good forms of biologically inspired visual representation," *PLoS computational biology*, vol. 5, no. 11, p. e1000579, 2009.

[38] H. Mallinson and A. Gammerman, "Imputation using support vector machines," 2003.

[39] A. M. Martínez and A. C. Kak, "Pca versus lda," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 2, pp. 228–233, 2001.

[40] S. J. Fernstad and R. C. Glen, "Visual analysis of missing data—to see what isn't there," in *Visual Analytics Science and Technology (VAST), 2014 IEEE Conference on*, pp. 249–250, IEEE, 2014.

[41] C. Eaton, C. Plaisant, and T. Drizd, "Visualizing missing data: Graph interpretation user study," in *IFIP Conference on Human-Computer Interaction*, pp. 861–872, Springer, 2005.

[42] A. Kirk, "Visualizing zero: How to show something with nothing," 2014.

[43] M. Templ, A. Alfons, A. Kowarik, and B. Prantner, "Vim: Visualization and imputation of missing values. cran," 2015.

[44] D. F. Swayne, D. Cook, and A. Buja, "Xgobi: Interactive dynamic data visualization in the x window system," *Journal of computational and Graphical Statistics*, vol. 7, no. 1, pp. 113–130, 1998.

[45] D. F. Swayne, D. T. Lang, A. Buja, and D. Cook, "Ggobi: evolving from xgobi into an extensible framework for interactive data visualization," *Computational Statistics & Data Analysis*, vol. 43, no. 4, pp. 423–444, 2003.

[46] M. Theus *et al.*, "Interactive data visualization using mondrian," *Journal of Statistical Software*, vol. 7, no. 11, pp. 1–9, 2002.

[47] S. Urbanek and M. Theus, "iplots: high interaction graphics for r," in *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*, Citeseer, 2003.

[48] A. Bilogur, "Missingno: a missing data visualization suite," *The Journal of Open Source Software*, no. 547, 2018.

[49] G. Piatetsky, "Python eats away at r: Top software for analytics, data science, machine learning in 2018: Trends and analysis," *URl: https://www.kdnuggets.com/2018/05/poll-tools-analytics-data-science-machine-learning-results.html*, 2018.

[50] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[51] D. Dheeru and E. Karra Taniskidou, "UCI machine learning repository," 2017.

[52] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework.," *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.

[53] J. Allaire and K. Ushey, "Package "reticulate"," 2018.

[54] N. A. Ali and Z. M. Omer, "Improving accuracy of missing data imputation in data mining," *Kurdistan Journal of Applied Research*, vol. 2, no. 3, pp. 66–73, 2017.

[55] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*, vol. 112. Springer, 2013.

[56] N. J. Tierney, F. A. Harden, M. J. Harden, and K. L. Mengersen, "Using decision trees to understand structure in missing data," *BMJ open*, vol. 5, no. 6, p. e007450, 2015.

# Anexos

# A

# Exemplo Relatório Estatístico

# Report

## Original Data set

### Dataset Info

| | |
|---|---|
| **Number of features** | 14 |
| **Number of instances** | 178 |
| **Number of Missing Values** | 295 |
| **Percentage of Missing Values** | 0.12 |
| **Number of instances with Missing Values** | 144 |
| **Percentage of instances with Missing Values** | 0.06 |

### Variables Type

| | |
|---|---|
| **Categoric** | 0 |
| **Numeric** | 14 |

### Warnings

Total.phenols is highly correlated with Flavanoids (p = 0.845)

Class is highly correlated with Flavanoids (p = 0.82)

Class has 36/20.22% missing values

Malic.acid has 35/19.66% missing values

Total.phenols has 55/30.9% missing values

Color.intensity has 43/24.16% missing values

Hue has 36/20.22% missing values

OD280 has 36/20.22% missing values

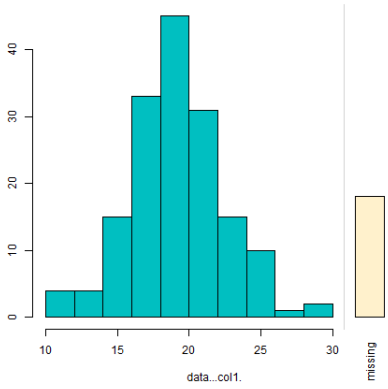
Proline has 36/20.22% missing values

**Variables**

### Alcalinity

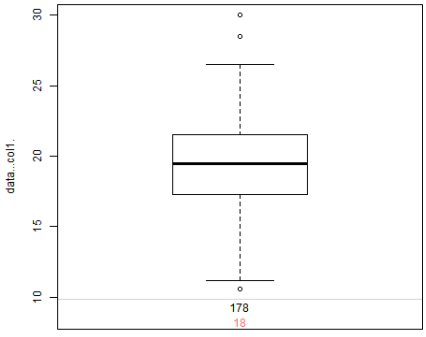
#### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 18 |
| **Missing (%)** | 10.11 |
| **Standard deviation** | 3.276 |
| **Kurtiosis** | 0.577 |
| **Mean** | 19.44 |
| **Skewness** | 0.12 |
| **Variance** | 10.734 |




#### Quantile statistics

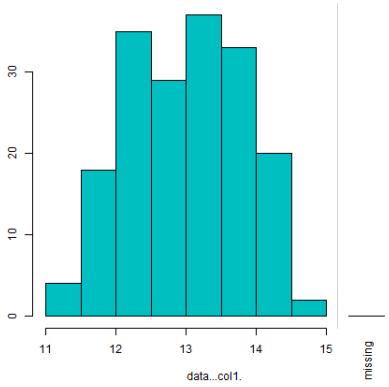
| | |
|---|---|
| **Min** | 10.6 |
| **Q1 (25%)** | 17.35 |
| **Median (50%)** | 19.5 |
| **Q3 (75%)** | 21.5 |
| **Max** | 30.0 |
| **Range** | 19.4 |



# Alcohol

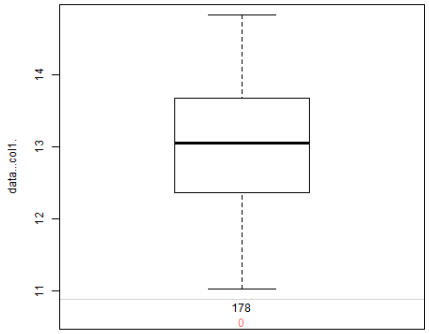
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.812 |
| **Kurtiosis** | -0.852 |
| **Mean** | 13.0 |
| **Skewness** | -0.051 |
| **Variance** | 0.659 |




## Quantile statistics

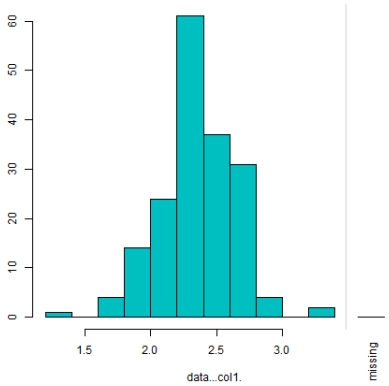
| | |
|---|---|
| **Min** | 11.03 |
| **Q1 (25%)** | 12.362 |
| **Median (50%)** | 13.05 |
| **Q3 (75%)** | 13.678 |
| **Max** | 14.83 |
| **Range** | 3.8 |




# Ash

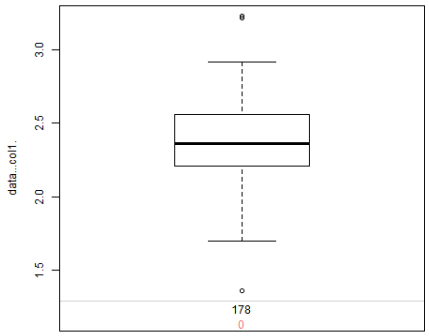
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.274 |
| **Kurtiosis** | 1.144 |
| **Mean** | 2.37 |
| **Skewness** | -0.177 |
| **Variance** | 0.075 |




## Quantile statistics

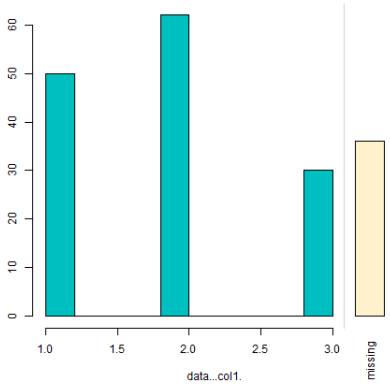
| | |
|---|---|
| **Min** | 1.36 |
| **Q1 (25%)** | 2.21 |
| **Median (50%)** | 2.36 |
| **Q3 (75%)** | 2.558 |
| **Max** | 3.23 |
| **Range** | 1.87 |



# Class

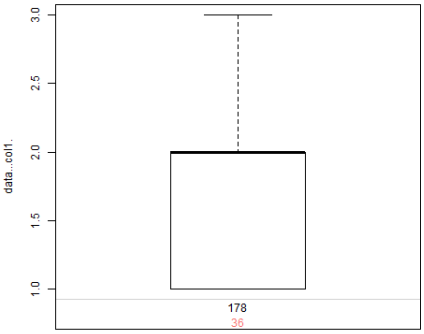
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 36 |
| **Missing (%)** | 20.22 |
| **Standard deviation** | 0.74 |
| **Kurtiosis** | -1.136 |
| **Mean** | 1.86 |
| **Skewness** | 0.231 |
| **Variance** | 0.547 |




## Quantile statistics

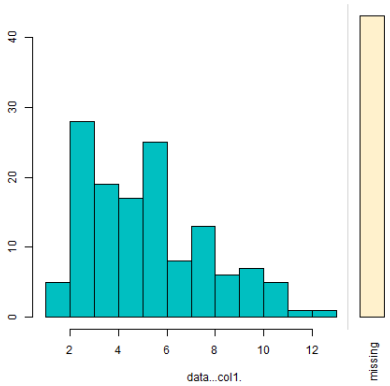
| | |
|---|---|
| **Min** | 1.0 |
| **Q1 (25%)** | 1.0 |
| **Median (50%)** | 2.0 |
| **Q3 (75%)** | 2.0 |
| **Max** | 3.0 |
| **Range** | 2.0 |




# Color.intensity

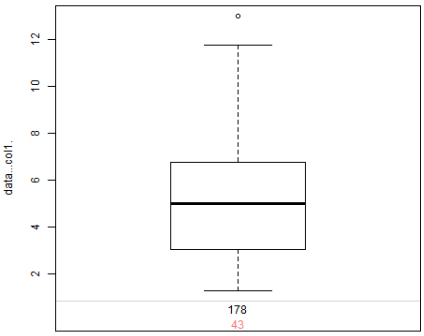
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 43 |
| **Missing (%)** | 24.16 |
| **Standard deviation** | 2.508 |
| **Kurtiosis** | -0.073 |
| **Mean** | 5.22 |
| **Skewness** | 0.739 |
| **Variance** | 6.289 |




## Quantile statistics

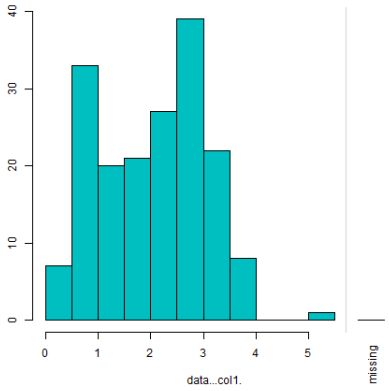
| | |
|---|---|
| **Min** | 1.28 |
| **Q1 (25%)** | 3.05 |
| **Median (50%)** | 5.0 |
| **Q3 (75%)** | 6.775 |
| **Max** | 13.0 |
| **Range** | 11.72 |



## Flavanoids

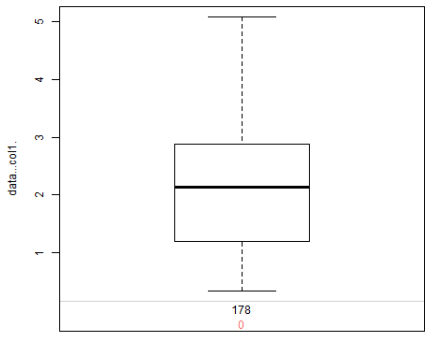
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.999 |
| **Kurtiosis** | -0.88 |
| **Mean** | 2.03 |
| **Skewness** | 0.025 |
| **Variance** | 0.998 |




### Quantile statistics

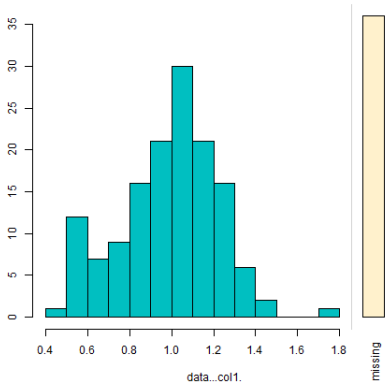
| | |
|---|---|
| **Min** | 0.34 |
| **Q1 (25%)** | 1.205 |
| **Median (50%)** | 2.135 |
| **Q3 (75%)** | 2.875 |
| **Max** | 5.08 |
| **Range** | 4.74 |




## Hue

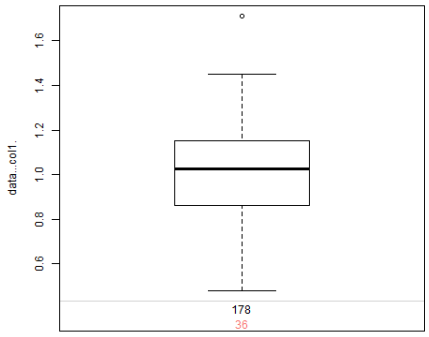
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 36 |
| **Missing (%)** | 20.22 |
| **Standard deviation** | 0.227 |
| **Kurtiosis** | -0.057 |
| **Mean** | 0.99 |
| **Skewness** | -0.167 |
| **Variance** | 0.051 |




### Quantile statistics

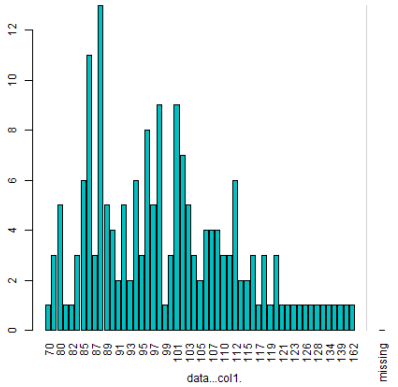
| | |
|---|---|
| **Min** | 0.48 |
| **Q1 (25%)** | 0.862 |
| **Median (50%)** | 1.025 |
| **Q3 (75%)** | 1.145 |
| **Max** | 1.71 |
| **Range** | 1.23 |



# Magnesium

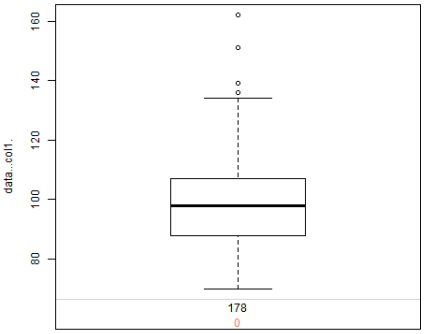
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 14.282 |
| **Kurtiosis** | 2.105 |
| **Mean** | 99.74 |
| **Skewness** | 1.098 |
| **Variance** | 203.989 |




## Quantile statistics

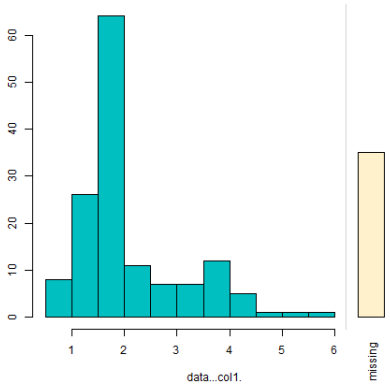
| | |
|---|---|
| **Min** | 70.0 |
| **Q1 (25%)** | 88.0 |
| **Median (50%)** | 98.0 |
| **Q3 (75%)** | 107.0 |
| **Max** | 162.0 |
| **Range** | 92.0 |




# Malic.acid

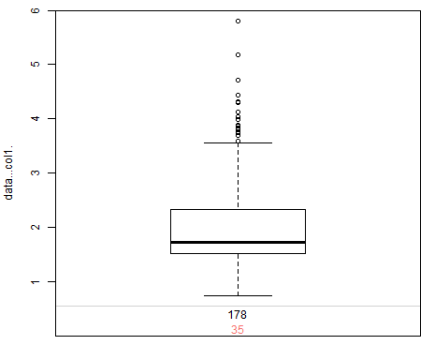
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 35 |
| **Missing (%)** | 19.66 |
| **Standard deviation** | 0.992 |
| **Kurtiosis** | 1.413 |
| **Mean** | 2.09 |
| **Skewness** | 1.399 |
| **Variance** | 0.984 |




## Quantile statistics

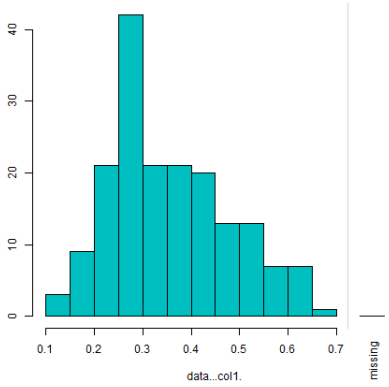
| | |
|---|---|
| **Min** | 0.74 |
| **Q1 (25%)** | 1.51 |
| **Median (50%)** | 1.73 |
| **Q3 (75%)** | 2.335 |
| **Max** | 5.8 |
| **Range** | 5.06 |



# Nonflavanoid

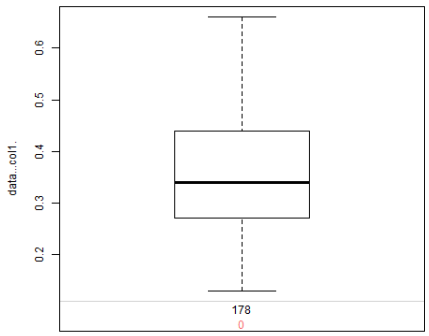
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.124 |
| **Kurtiosis** | -0.637 |
| **Mean** | 0.36 |
| **Skewness** | 0.45 |
| **Variance** | 0.015 |




## Quantile statistics

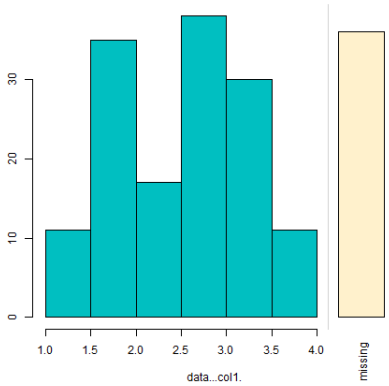
| | |
|---|---|
| **Min** | 0.13 |
| **Q1 (25%)** | 0.27 |
| **Median (50%)** | 0.34 |
| **Q3 (75%)** | 0.438 |
| **Max** | 0.66 |
| **Range** | 0.53 |




# OD280

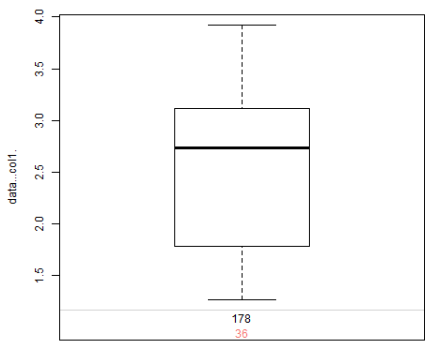
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 36 |
| **Missing (%)** | 20.22 |
| **Standard deviation** | 0.726 |
| **Kurtiosis** | -1.251 |
| **Mean** | 2.52 |
| **Skewness** | -0.143 |
| **Variance** | 0.528 |




## Quantile statistics

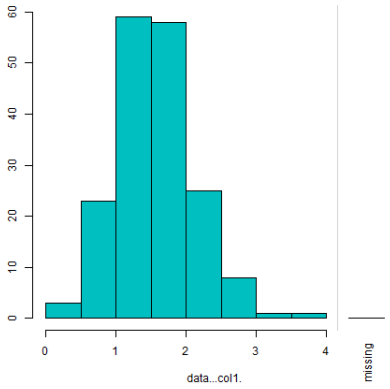
| | |
|---|---|
| **Min** | 1.27 |
| **Q1 (25%)** | 1.785 |
| **Median (50%)** | 2.735 |
| **Q3 (75%)** | 3.115 |
| **Max** | 3.92 |
| **Range** | 2.65 |



# Proanthocyanins

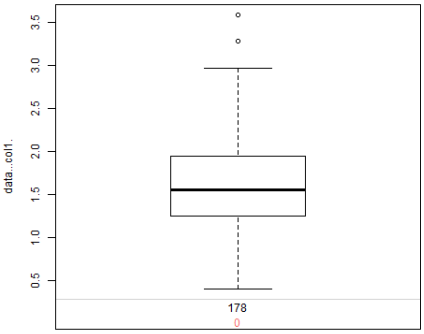
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.572 |
| **Kurtiosis** | 0.555 |
| **Mean** | 1.59 |
| **Skewness** | 0.517 |
| **Variance** | 0.328 |




## Quantile statistics

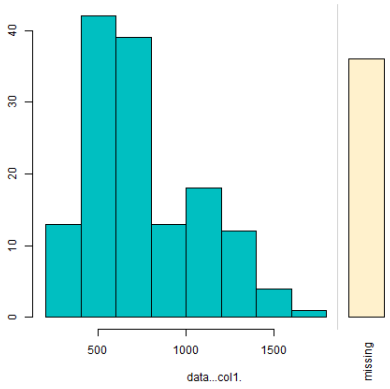
| | |
|---|---|
| **Min** | 0.41 |
| **Q1 (25%)** | 1.25 |
| **Median (50%)** | 1.555 |
| **Q3 (75%)** | 1.95 |
| **Max** | 3.58 |
| **Range** | 3.17 |




# Proline

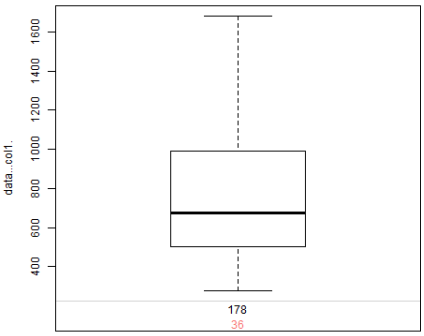
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 36 |
| **Missing (%)** | 20.22 |
| **Standard deviation** | 316.623 |
| **Kurtiosis** | -0.25 |
| **Mean** | 752.92 |
| **Skewness** | 0.787 |
| **Variance** | 100250.149 |




## Quantile statistics

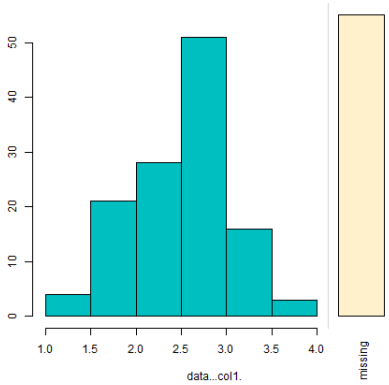
| | |
|---|---|
| **Min** | 278.0 |
| **Q1 (25%)** | 502.5 |
| **Median (50%)** | 673.5 |
| **Q3 (75%)** | 988.75 |
| **Max** | 1680.0 |
| **Range** | 1402.0 |



# Total.phenols

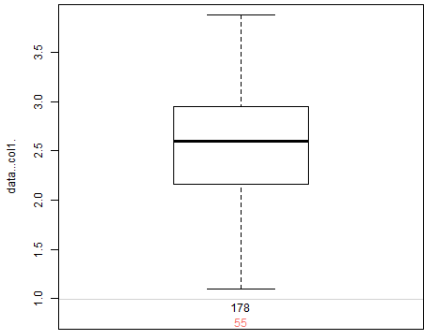
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 55 |
| **Missing (%)** | 30.9 |
| **Standard deviation** | 0.547 |
| **Kurtiosis** | -0.169 |
| **Mean** | 2.54 |
| **Skewness** | -0.247 |
| **Variance** | 0.299 |




## Quantile statistics

| | |
|---|---|
| **Min** | 1.1 |
| **Q1 (25%)** | 2.165 |
| **Median (50%)** | 2.6 |
| **Q3 (75%)** | 2.95 |
| **Max** | 3.88 |
| **Range** | 2.78 |

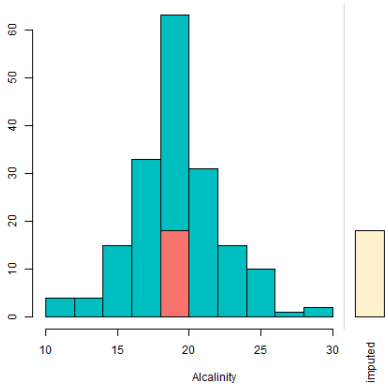


# Imputation mean/mode

## Alcalinity

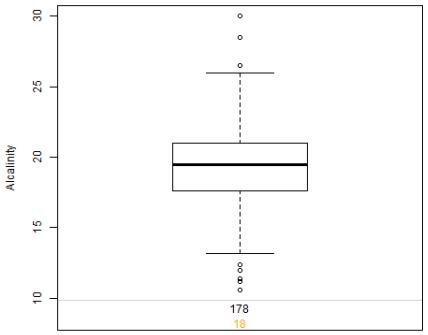
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 3.105 |
| **Kurtiosis** | 0.979 |
| **Mean** | 19.44 |
| **Skewness** | 0.126 |
| **Variance** | 9.642 |




### Quantile statistics

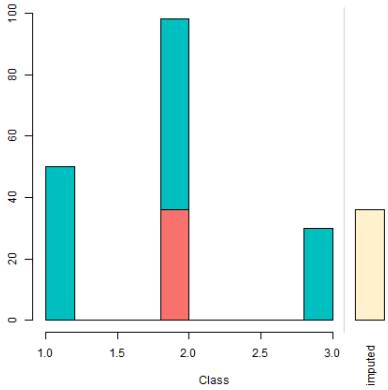
| | |
|---|---|
| **Min** | 10.6 |
| **Q1 (25%)** | 17.65 |
| **Median (50%)** | 19.442 |
| **Q3 (75%)** | 21.0 |
| **Max** | 30.0 |
| **Range** | 19.4 |




## Class

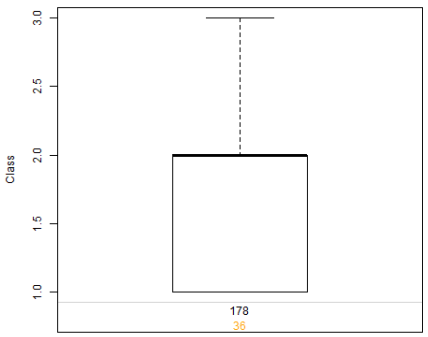
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.66 |
| **Kurtiosis** | -0.651 |
| **Mean** | 1.86 |
| **Skewness** | 0.258 |
| **Variance** | 0.436 |




### Quantile statistics

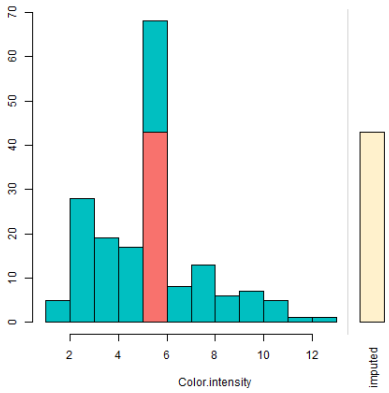
| | |
|---|---|
| **Min** | 1.0 |
| **Q1 (25%)** | 1.0 |
| **Median (50%)** | 2.0 |
| **Q3 (75%)** | 2.0 |
| **Max** | 3.0 |
| **Range** | 2.0 |



# Color.intensity

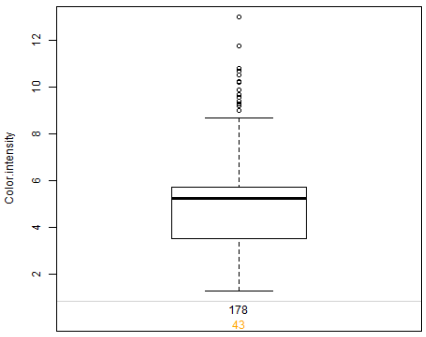
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 2.182 |
| **Kurtiosis** | 0.862 |
| **Mean** | 5.22 |
| **Skewness** | 0.846 |
| **Variance** | 4.761 |




## Quantile statistics

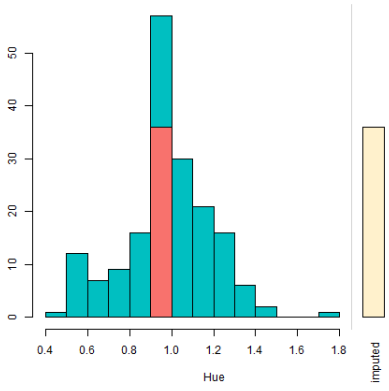
| | |
|---|---|
| **Min** | 1.28 |
| **Q1 (25%)** | 3.535 |
| **Median (50%)** | 5.223 |
| **Q3 (75%)** | 5.7 |
| **Max** | 13.0 |
| **Range** | 11.72 |




# Hue

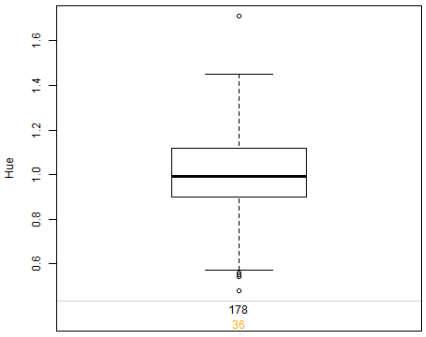
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.203 |
| **Kurtiosis** | 0.692 |
| **Mean** | 0.99 |
| **Skewness** | -0.186 |
| **Variance** | 0.041 |




## Quantile statistics

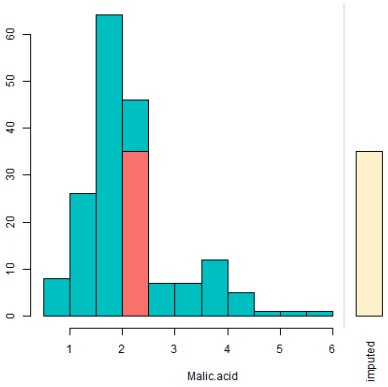
| | |
|---|---|
| **Min** | 0.48 |
| **Q1 (25%)** | 0.902 |
| **Median (50%)** | 0.993 |
| **Q3 (75%)** | 1.118 |
| **Max** | 1.71 |
| **Range** | 1.23 |



# Malic.acid

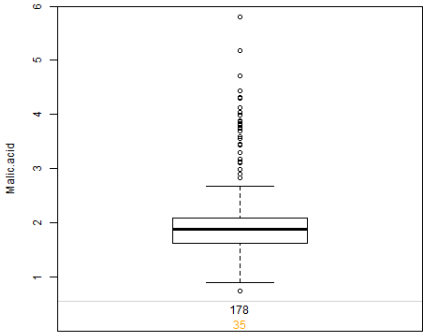
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.888 |
| **Kurtiosis** | 2.483 |
| **Mean** | 2.09 |
| **Skewness** | 1.557 |
| **Variance** | 0.789 |




## Quantile statistics

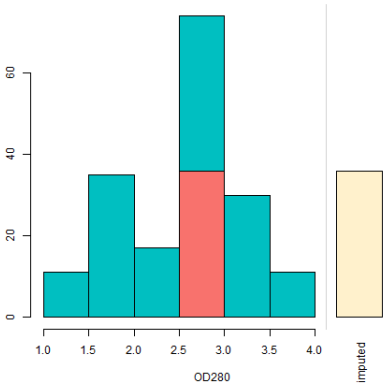
| | |
|---|---|
| **Min** | 0.74 |
| **Q1 (25%)** | 1.61 |
| **Median (50%)** | 1.875 |
| **Q3 (75%)** | 2.093 |
| **Max** | 5.8 |
| **Range** | 5.06 |




# OD280

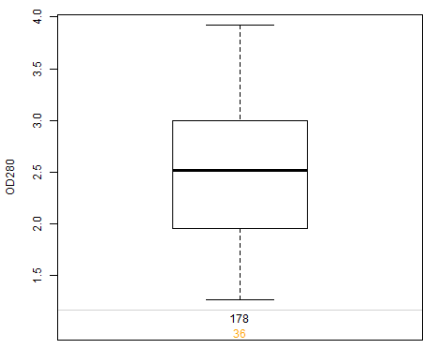
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.648 |
| **Kurtiosis** | -0.794 |
| **Mean** | 2.52 |
| **Skewness** | -0.159 |
| **Variance** | 0.42 |




## Quantile statistics

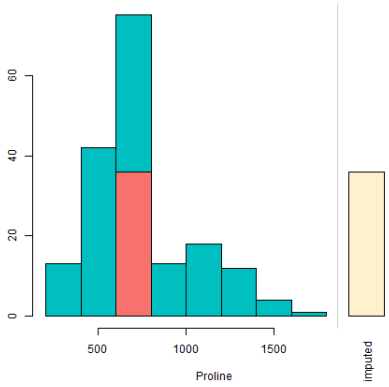
| | |
|---|---|
| **Min** | 1.27 |
| **Q1 (25%)** | 1.97 |
| **Median (50%)** | 2.518 |
| **Q3 (75%)** | 2.99 |
| **Max** | 3.92 |
| **Range** | 2.65 |



## Proline

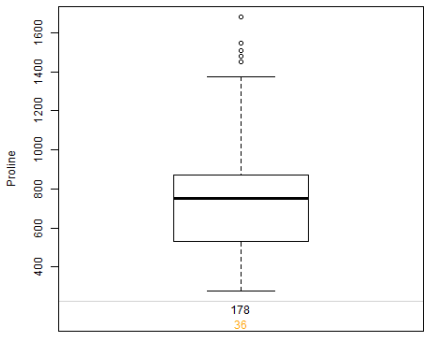
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 282.596 |
| **Kurtiosis** | 0.452 |
| **Mean** | 752.92 |
| **Skewness** | 0.879 |
| **Variance** | 79860.288 |




### Quantile statistics

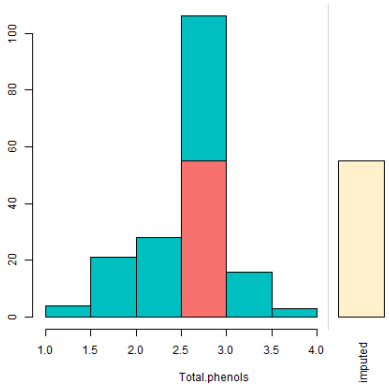
| | |
|---|---|
| **Min** | 278.0 |
| **Q1 (25%)** | 535.0 |
| **Median (50%)** | 751.458 |
| **Q3 (75%)** | 863.75 |
| **Max** | 1680.0 |
| **Range** | 1402.0 |




## Total.phenols

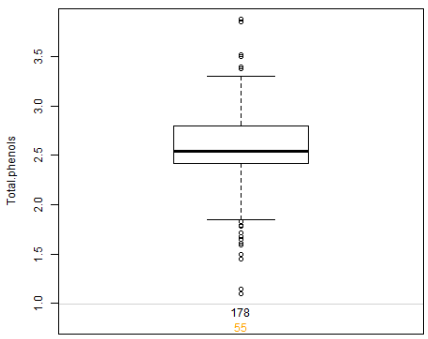
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.454 |
| **Kurtiosis** | 1.101 |
| **Mean** | 2.54 |
| **Skewness** | -0.296 |
| **Variance** | 0.206 |




### Quantile statistics

| | |
|---|---|
| **Min** | 1.1 |
| **Q1 (25%)** | 2.428 |
| **Median (50%)** | 2.544 |
| **Q3 (75%)** | 2.8 |
| **Max** | 3.88 |
| **Range** | 2.78 |

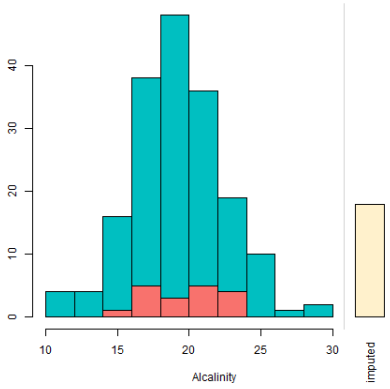


# Imputation svm_default

## Alcalinity

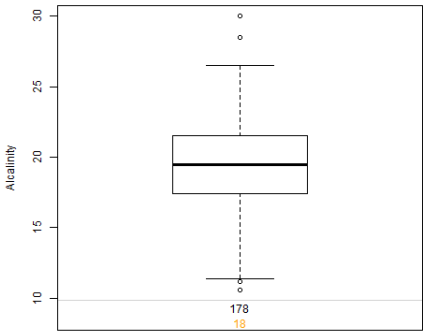
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 3.19 |
| **Kurtiosis** | 0.61 |
| **Mean** | 19.44 |
| **Skewness** | 0.113 |
| **Variance** | 10.176 |




### Quantile statistics

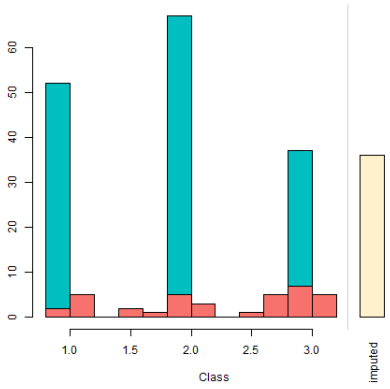
| | |
|---|---|
| **Min** | 10.6 |
| **Q1 (25%)** | 17.417 |
| **Median (50%)** | 19.5 |
| **Q3 (75%)** | 21.432 |
| **Max** | 30.0 |
| **Range** | 19.4 |




## Class

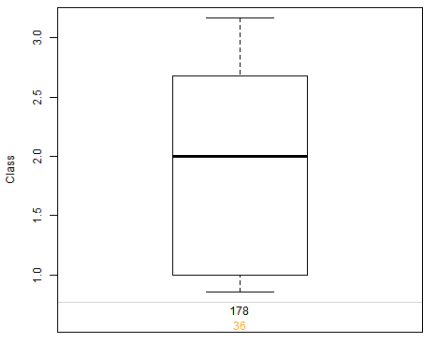
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.754 |
| **Kurtiosis** | -1.261 |
| **Mean** | 1.93 |
| **Skewness** | 0.107 |
| **Variance** | 0.568 |




### Quantile statistics

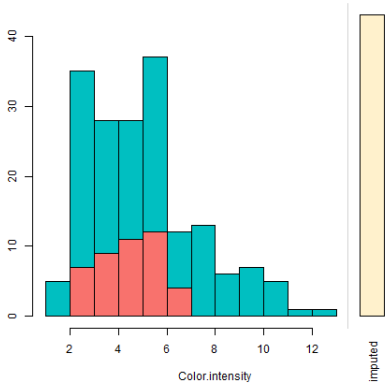
| | |
|---|---|
| **Min** | 0.8559833807055011 |
| **Q1 (25%)** | 1.0 |
| **Median (50%)** | 2.0 |
| **Q3 (75%)** | 2.666 |
| **Max** | 3.16651747121307 |
| **Range** | 2.311 |



# Color.intensity

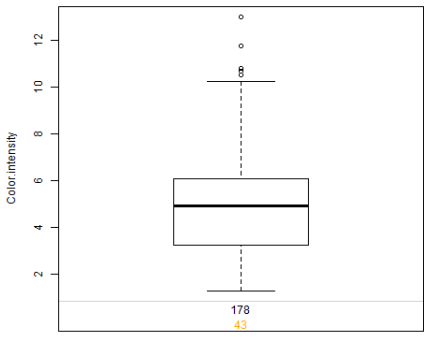
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 2.28 |
| **Kurtiosis** | 0.547 |
| **Mean** | 5.06 |
| **Skewness** | 0.896 |
| **Variance** | 5.2 |




## Quantile statistics

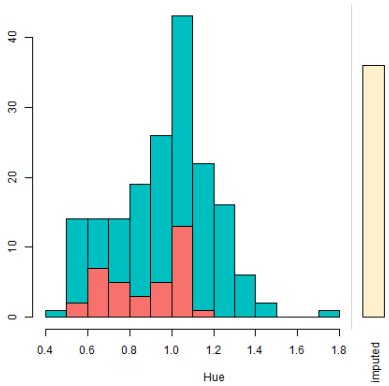
| | |
|---|---|
| **Min** | 1.2800000000000002 |
| **Q1 (25%)** | 3.25 |
| **Median (50%)** | 4.91 |
| **Q3 (75%)** | 6.062 |
| **Max** | 13.0 |
| **Range** | 11.72 |




# Hue

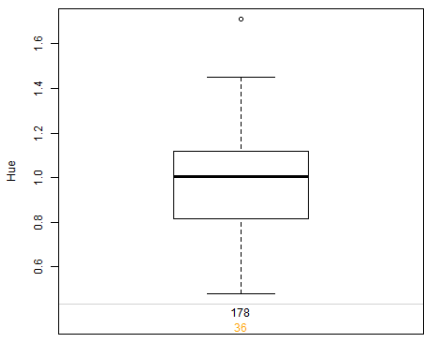
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.223 |
| **Kurtiosis** | -0.155 |
| **Mean** | 0.97 |
| **Skewness** | -0.086 |
| **Variance** | 0.05 |




## Quantile statistics

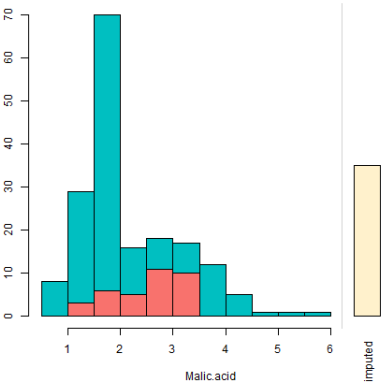
| | |
|---|---|
| **Min** | 0.48 |
| **Q1 (25%)** | 0.818 |
| **Median (50%)** | 1.004 |
| **Q3 (75%)** | 1.12 |
| **Max** | 1.71 |
| **Range** | 1.23 |



# Malic.acid

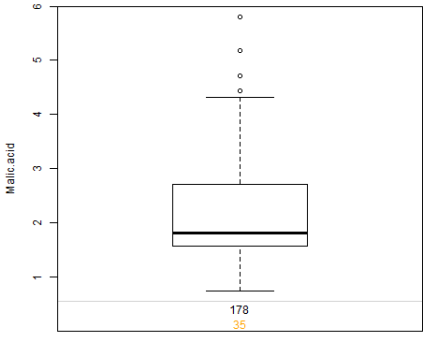
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.941 |
| **Kurtiosis** | 1.006 |
| **Mean** | 2.18 |
| **Skewness** | 1.134 |
| **Variance** | 0.885 |




## Quantile statistics

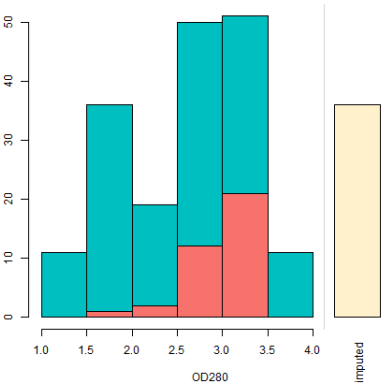
| | |
|---|---|
| **Min** | 0.74 |
| **Q1 (25%)** | 1.575 |
| **Median (50%)** | 1.817 |
| **Q3 (75%)** | 2.711 |
| **Max** | 5.8 |
| **Range** | 5.06 |




# OD280

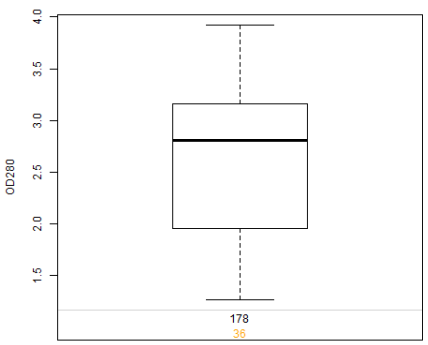
## Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.69 |
| **Kurtiosis** | -1.026 |
| **Mean** | 2.61 |
| **Skewness** | -0.418 |
| **Variance** | 0.476 |




## Quantile statistics

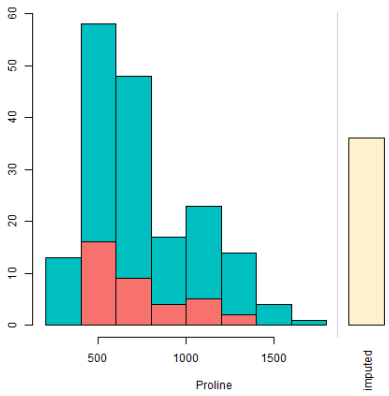
| | |
|---|---|
| **Min** | 1.27 |
| **Q1 (25%)** | 1.961 |
| **Median (50%)** | 2.806 |
| **Q3 (75%)** | 3.159 |
| **Max** | 3.92 |
| **Range** | 2.65 |



## Proline

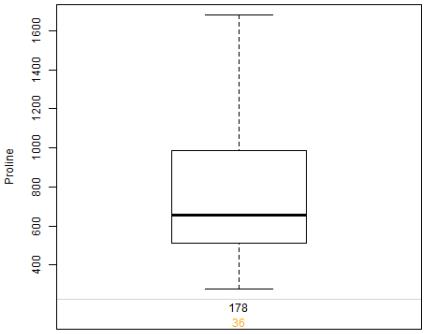
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 304.135 |
| **Kurtiosis** | -0.135 |
| **Mean** | 744.14 |
| **Skewness** | 0.834 |
| **Variance** | 92497.918 |




### Quantile statistics

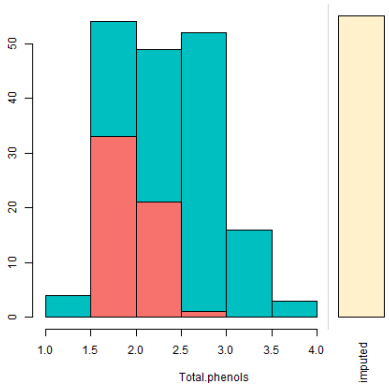
| | |
|---|---|
| **Min** | 278.0 |
| **Q1 (25%)** | 510.0 |
| **Median (50%)** | 655.0 |
| **Q3 (75%)** | 985.0 |
| **Max** | 1680.0 |
| **Range** | 1402.0 |




## Total.phenols

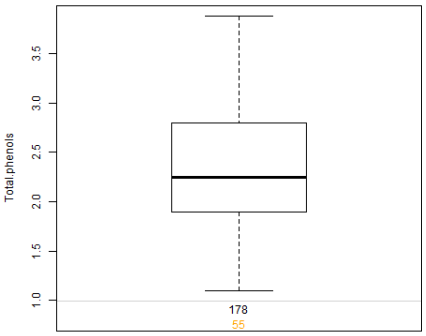
### Descriptive statistics

| | |
|---|---|
| **Missing (n)** | 0 |
| **Missing (%)** | 0.0 |
| **Standard deviation** | 0.543 |
| **Kurtiosis** | -0.505 |
| **Mean** | 2.36 |
| **Skewness** | 0.329 |
| **Variance** | 0.295 |




### Quantile statistics

| | |
|---|---|
| **Min** | 1.1 |
| **Q1 (25%)** | 1.905 |
| **Median (50%)** | 2.25 |
| **Q3 (75%)** | 2.8 |
| **Max** | 3.88 |
| **Range** | 2.78 |



# B

# Recolha de dados

**Tabela B.1:** Descrição do número e tipo de variáveis e do número de observações de cada conjunto de dados adquirido.

| Dataset | Variables | | | Instances | Dataset | Variables | | | Instances |
|---|---|---|---|---|---|---|---|---|---|
| | Numerical | Categorical | Total | | | Numerical | Categorical | Total | |
| diabetes | 3 | 0 | 3 | 43 | glass_0.6_vs.5 | 9 | 1 | 10 | 108 |
| vertebral-2c | 2 | 1 | 3 | 310 | glass_0.1.5_vs_2 | 9 | 1 | 10 | 172 |
| ele-1 | 3 | 0 | 3 | 495 | glass_0.1.6_vs_5 | 9 | 1 | 10 | 184 |
| haberman | 3 | 1 | 4 | 306 | glass_0.1.6_vs_2 | 9 | 1 | 10 | 192 |
| quake | 4 | 0 | 4 | 2178 | glass_0.1.4.6_vs_2 | 9 | 1 | 10 | 205 |
| adult+stretch | 0 | 5 | 5 | 22 | glass | 9 | 1 | 10 | 214 |
| iris0 | 4 | 1 | 5 | 150 | shuttle_6_vs_2_3 | 9 | 1 | 10 | 230 |
| balance_scaleBvsL | 4 | 1 | 5 | 337 | wisconsin | 9 | 1 | 10 | 683 |
| balance_scaleBvsR | 4 | 1 | 5 | 337 | breast-cancer-wisconsin | 1 | 9 | 10 | 699 |
| balance-scale | 4 | 1 | 5 | 625 | dataset_50_tic-tac-toe | 0 | 10 | 10 | 958 |
| transfusion | 4 | 1 | 5 | 748 | tic-tac-toe | 0 | 10 | 10 | 958 |
| laser | 5 | 0 | 5 | 993 | cmc1vs2 | 2 | 8 | 10 | 962 |
| newthyroid-v3 | 5 | 1 | 6 | 180 | shuttle_c0_vs_c4 | 9 | 1 | 10 | 1829 |
| newthyroid-v1 | 5 | 1 | 6 | 185 | poker_9_vs_7 | 10 | 1 | 11 | 244 |
| biomed | 5 | 1 | 6 | 194 | page_blocks_1_3_vs_4 | 10 | 1 | 11 | 472 |
| new-thyroid | 5 | 1 | 6 | 215 | pageblocks_1vs4_5 | 10 | 1 | 11 | 5116 |
| mammographic | 5 | 1 | 6 | 830 | pageblocks_1vs3_4_5 | 10 | 1 | 11 | 5144 |
| kala-azar | 5 | 2 | 7 | 68 | page_blocks0 | 10 | 1 | 11 | 5472 |
| bankruptcy | 0 | 7 | 7 | 250 | pageblocks_1_2vs_3_4_5 | 10 | 1 | 11 | 5473 |
| bupa | 5 | 2 | 7 | 345 | winequality_red_8_vs_6 | 11 | 1 | 12 | 656 |
| monk-2 | 0 | 7 | 7 | 432 | winequality_red_4 | 11 | 1 | 12 | 1599 |
| monks-problems-2 | 0 | 7 | 7 | 601 | wine | 12 | 1 | 13 | 178 |
| car_vgood | 0 | 7 | 7 | 1728 | relax | 12 | 1 | 13 | 182 |
| kr_vs_k_one_vs_fifteen | 0 | 7 | 7 | 2244 | cleveland_0_vs_4 | 13 | 1 | 14 | 173 |
| kr_vs_k_zero_one_vs_draw | 0 | 7 | 7 | 2901 | parkinson | 13 | 1 | 14 | 195 |
| kr_vs_k_three_vs_eleven | 0 | 7 | 7 | 2935 | vowel0 | 13 | 1 | 14 | 988 |
| appendicitis | 7 | 1 | 8 | 106 | eeg | 14 | 1 | 15 | 14980 |
| led7digit | 7 | 1 | 8 | 443 | seismic-bumps | 11 | 5 | 16 | 2584 |
| abalone_3_vs_11 | 7 | 2 | 9 | 502 | letter_Z | 16 | 1 | 17 | 20000 |
| abalone_21_vs_8 | 7 | 2 | 9 | 581 | vehicle0 | 18 | 1 | 19 | 846 |
| abalone9_18 | 7 | 2 | 9 | 731 | segment0 | 18 | 1 | 19 | 2308 |
| pima | 8 | 1 | 9 | 768 | auto | 12 | 13 | 25 | 159 |
| yeast | 8 | 1 | 9 | 1484 | wpbc | 32 | 1 | 33 | 198 |
| abalone_17_vs_7_8_9_10 | 7 | 2 | 9 | 2338 | ionosphere | 33 | 1 | 34 | 351 |
| abalone | 7 | 2 | 9 | 4177 | satimage | 36 | 1 | 37 | 4435 |
| nursery | 0 | 9 | 9 | 12960 | spectf | 44 | 1 | 45 | 267 |
| glass_0_4_vs_5 | 9 | 1 | 10 | 92 | sonar | 59 | 1 | 60 | 207 |
| fertility-diagnosis | 2 | 8 | 10 | 100 | R_data_frame | 100 | 1 | 101 | 1212 |
| breast-tissue-2c | 9 | 1 | 10 | 106 | grastroenterology | 152 | 0 | 152 | 700 |

# C

# Primeira Experiência

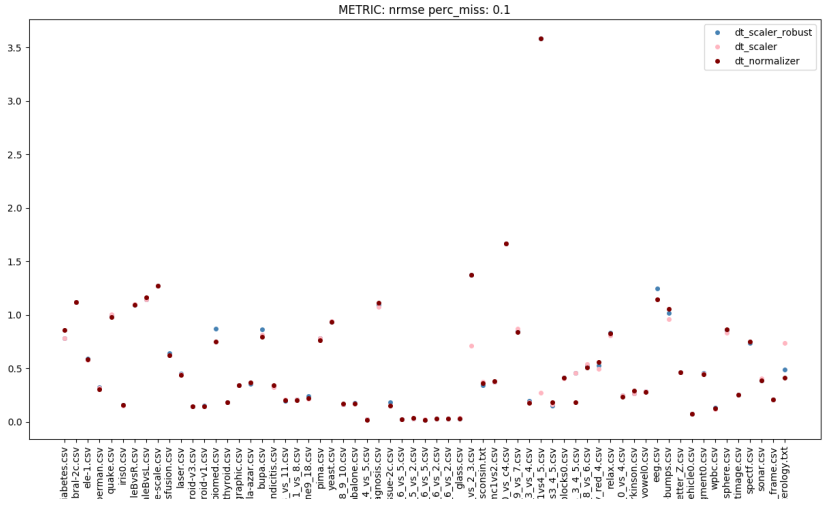



**Figura C.1:** Valores da métrica NRMSE nos *datasets* com 10% de dados em falta imputados com DT.

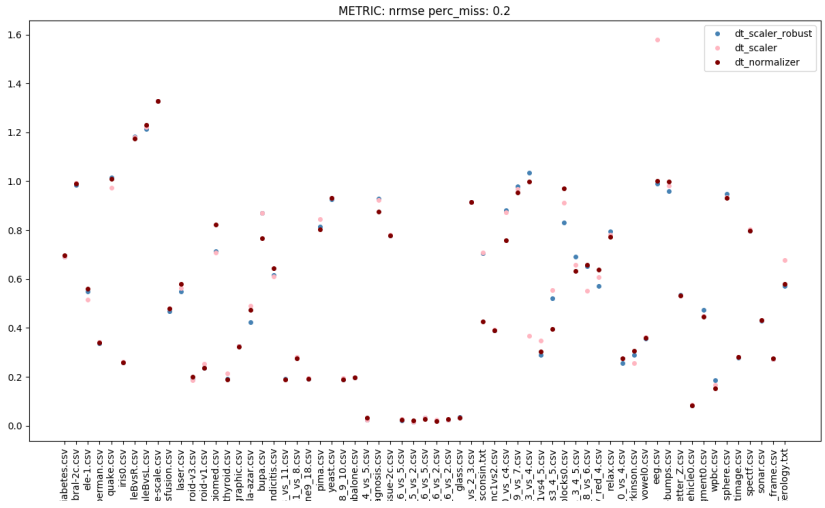



**Figura C.2:** Valores da métrica NRMSE nos *datasets* com 20% de dados em falta imputados com DT.

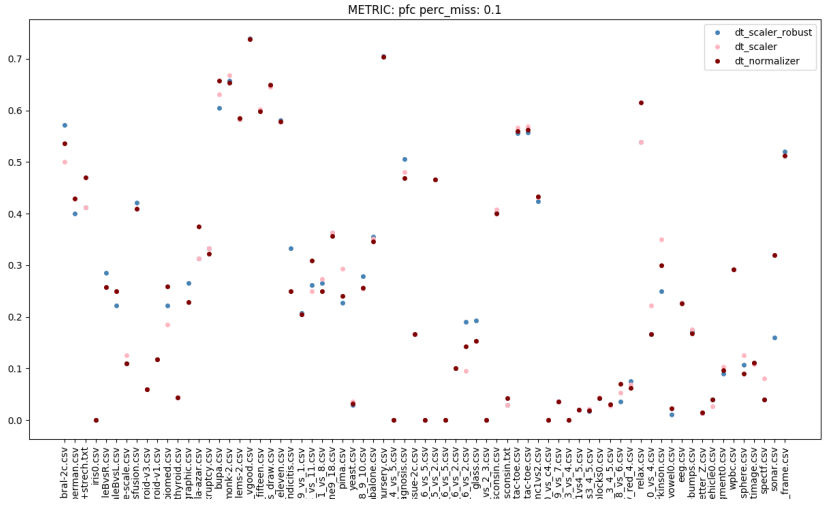



**Figura C.3:** Valores da métrica PFC nos conjunto de dados com 10% de dados em falta imputados com DT.

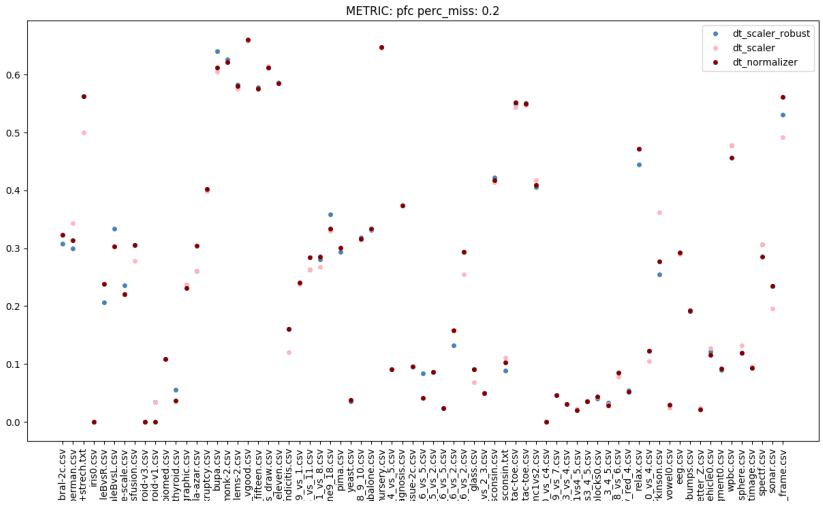



**Figura C.4:** Valores da métrica PFC nos *datasets* com 20% de dados em falta imputados com DT.

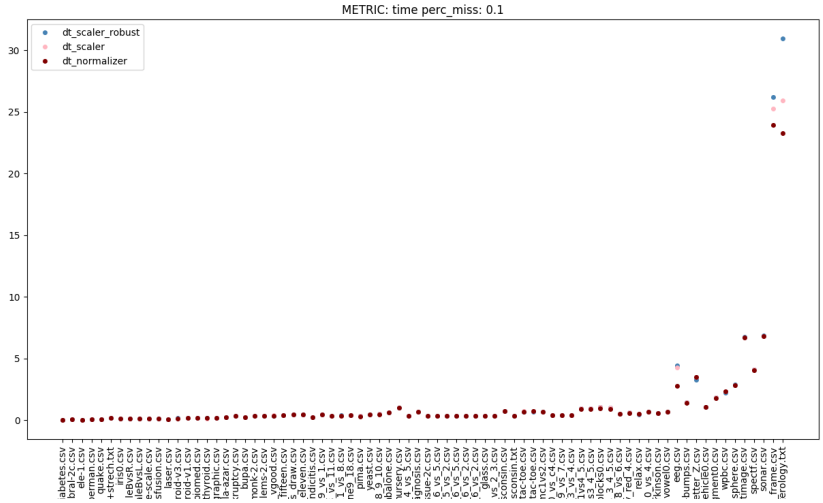



**Figura C.5:** Valores do tempo de execução nos *datasets* com 10% de dados em falta imputados com DT.

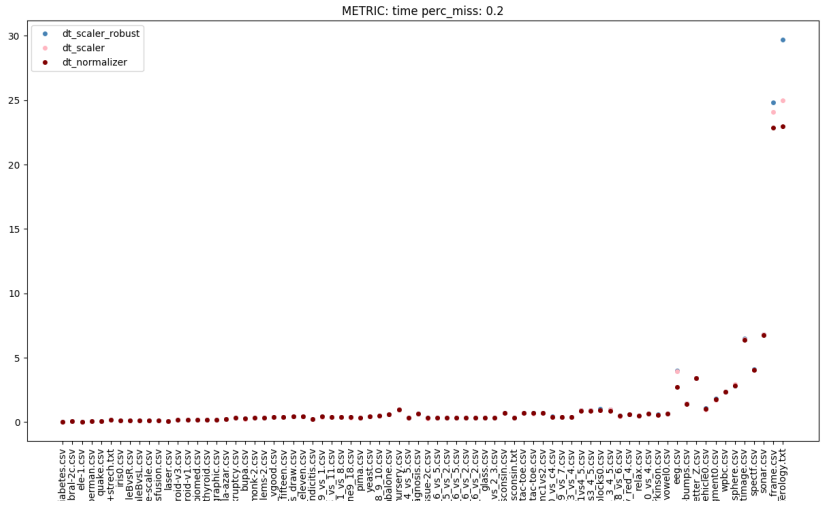



**Figura C.6:** Valores do tempo de execução nos *datasets* com 20% de dados em falta imputados com DT.

# D

# Segunda Experiência

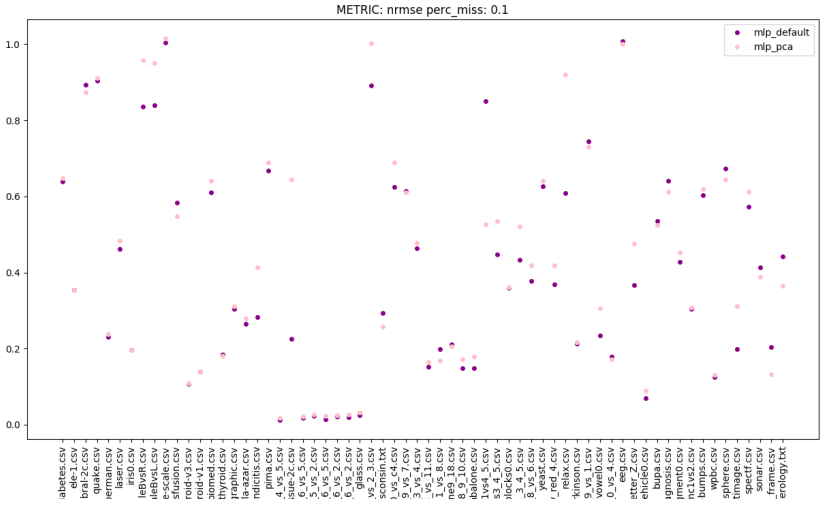



**Figura D.1:** Valores da métrica NRMSE nos *datasets* com 10% de dados em falta imputados com MLP.

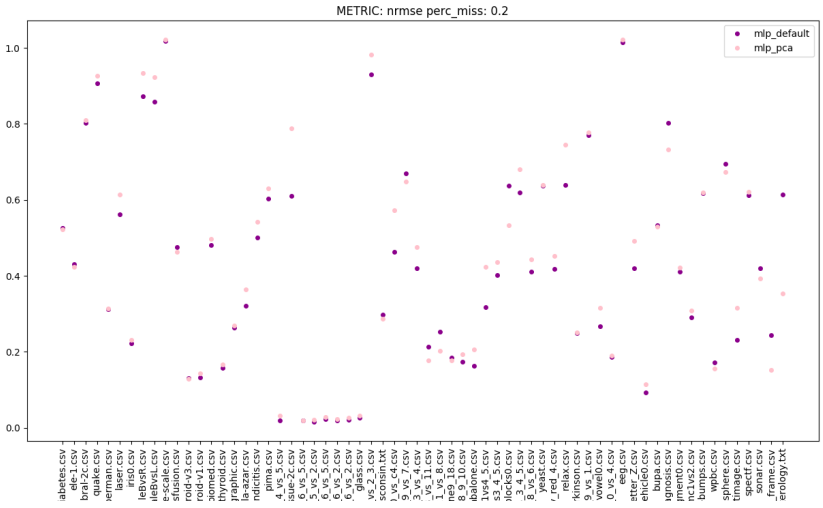



**Figura D.2:** Valores da métrica NRMSE nos *datasets* com 20% de dados em falta imputados com MLP.

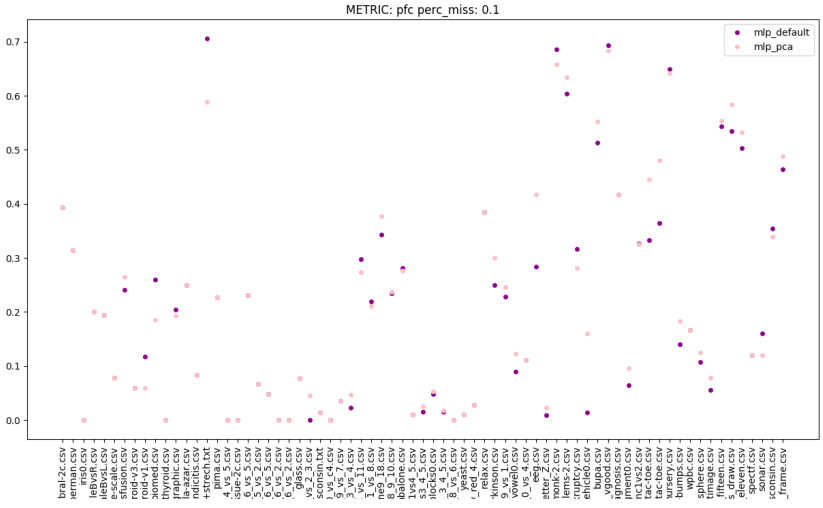



**Figura D.3:** Valores da métrica PFC nos *datasets* com 10% de dados em falta imputados com MLP.

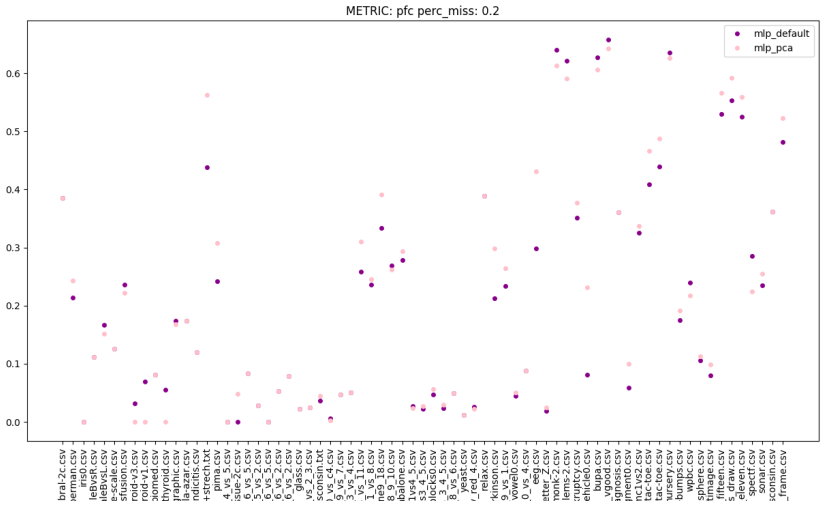



**Figura D.4:** Valores da métrica PFC nos *datasets* com 20% de dados em falta imputados com MLP.

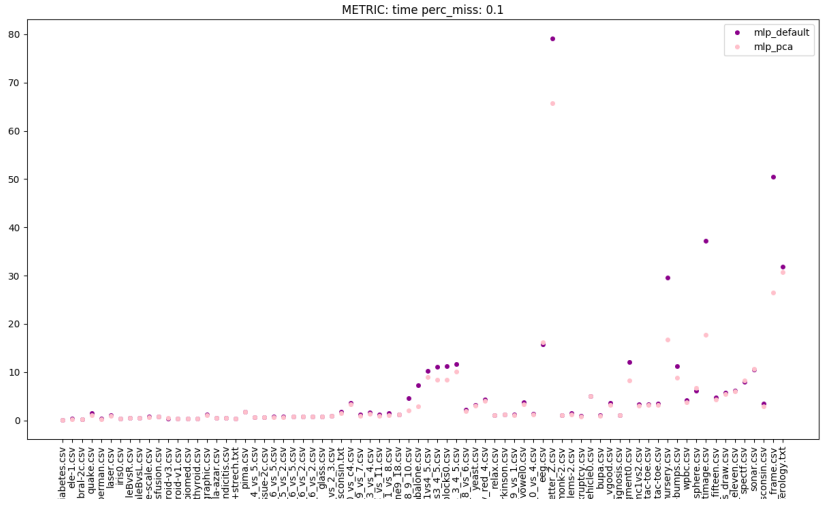



**Figura D.5:** Valores do tempo de execução nos *datasets* com 10% de dados em falta imputados com MLP.

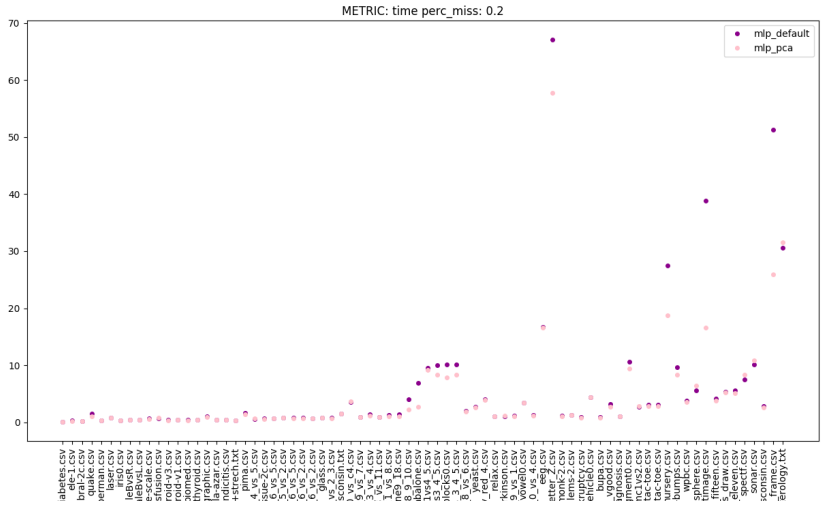



**Figura D.6:** Valores do tempo de execução nos *datasets* com 20% de dados em falta imputados com MLP.