DANIEL JOSÉ
LOBO PEREIRA
DE CASTRO

# AnalyzeR: Aplicação Web em Shiny para análise interativa de dados

# AnalyzeR: Shiny Web Application for interactive data analysis

Universidade de Aveiro
2021

**DANIEL JOSÉ LOBO PEREIRA DE CASTRO**

# AnalyzeR: Aplicação Web em Shiny para análise interativa de dados

# AnalyzeR: Shiny Web Application for interactive data analysis

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Estatística Médica, realizada sob a orientação científica da Porfessora Gabriela Maria Ferreira Ribeiro de Moura, Professora Auxiliar do Departamento de Ciências Médicas da Universidade de Aveiro e do Professor Doutor Luís Miguel Almeida da Silva, Professor Auxiliar Convidado do Departamento de Matemática da Universidade de Aveiro.

Para as minhas avós Paula e Teresa.

**o júri / the jury**

presidente / president

Prof. Doutora Vera Mónica Almeida Afreixo
Professora Auxiliar, Universidade de Aveiro

vogais / examiners committee

Professora Doutora Gabriela Maria Ferreira Ribeiro de Moura
Professora Auxiliar, Universidade de Aveiro

Doutor Miguel Monsanto Pinheiro
Equiparado a Investigador Auxiliar, Universidade de Aveiro

**Palavras Chave** Análise de dados, Estatística, R Shiny.

**Resumo**

A análise de dados ganhou relevância significativa no mundo atual. À medida que aumenta a quantidade de dados gerados, aumenta também a necessidade de compreender toda essa informação. Ferramentas clássicas, como folhas de cálculo, não permitem lidar com grandes quantidades de dados nem suportar procedimentos estatísticos complexos. Para realizar essa análise, é frequentemente necessário um software específico e aplicado. Os softwares com mais flexibilidade analítica requerem proficiência em programação. A linguagem R é muito popular para análise de dados, no entanto, pode ser difícil de aprender, especialmente por alguém sem nenhuma experiência de programação. O objetivo deste trabalho é construir uma aplicação web baseada em R Shiny que permita ao utilizador realizar análises estatísticas complexas sem requerer conhecimentos de programação.

**Keywords** Data analysis, statistics, R Shiny.

**Abstract** The analysis of data has gained significant relevance in the world of today. As the amount of data generated increases, so does the need for understanding and make sense of such information. Classical tools such as spreadsheets cannot handle large amounts of data nor support complex statistical procedures. In order to perform such analysis, often a specific and applied software is required. Most of those softwares require some level of programming proficiency. The R language has been very popular for data analysis, however, it can be hard to learn, especially if the learner lacks previous programming experience. The purpose of this work is to construct an R-based web application with Shiny that allows the user to perform complex statistical analysis without requiring any programming knowledge.

# Conteúdo

# Lista de figuras

# Lista de Tabelas

CAPÍTULO 1

# Introdução

## 1.1 Exposição do problema

A estatística desempenha um papel crucial na Ciência. Permite extrair informação relevante a partir de dados gerados no decorrer das investigações científicas e organizá-la de uma forma intelegível à luz do Método Científico. No entanto, a aplicação da mesma é frequentemente condicionada pelas limitações das aplicações de software que a suportam. De facto, as aplicações de uso "tradiconal", como é o caso das folhas de cálculo não suportam ou permitem as análises específicas (e muitas vezes complexas) requeridas pelas mais básicas investigações científicas da atualidade. Para além deste fator de complexidade, o incremento das possibilidades de colheita e armazenamento de dados requer ferramentas de análise mais especializadas e flexíveis. Neste contexto, surgiram alguns softwares comerciais que respondem a esta necessidade. Não obstante, as possibilidades de métodos e flexibilidade de análise muitas vezes são comprometidas ou limitadas. Assim, se for requerida uma análise mais avançada e adaptável, torna-se necessário recorrer à programação em linguagens especializadas. Para uma utilização eficaz das mesmas, é necessário conhecimentos e prática de programação. Este facto contitui uma importante dificuldade e embargo à procecussão e desenvolvimento de análises estatísticas no meio científico.

## 1.2 Programação em R

Uma das melhores linguagens para o uso da estaística é R. Esta está implementada e distribuída como software livre e aberto à comunidade, suportada pela R Foundation for Statistical Computing [4]. A linguagem R é amplamente utilizada para análises de qualidade e complexidade de nível superior. R está implementada essencialmente em C, C++ e Fortran e está disponível gratuitamente sob a GNU General Public License

[5]. Recentemente surgiram novos paradigmas de programação em R, nomeadamente através do IDE (Integrated Development Environment) RStudio. Uma destas novidades é o pacote e ambiente `Shiny` [6, 1]. Este permite implementar código R em aplicações Web com interatividade. Estas aplicações podem ser adaptadas recorrendo à sintaxe R em `Shiny`, e mesmo com HTML, CSS e JavaScript.

## 1.3 Estado da arte

Desde a sua criação (em 2012) a maior parte das aplicações em `Shiny` foram desenvolvidas por programadores inseridos no público em geral, sob a forma de projetos com utilidade lúdica e, muitas vezes, limitada. Uma vez que o desenvolvimento de uma aplicação em Shiny não requer forçosamente conhecimentos de linguagens de programação web apresenta-se como uma alternativa muito atrativa para iniciantes. Existem, de facto, competições globais que premeiam as melhores aplicações desenvolvidas pelo público (como é o caso do *Shiny Contest* em `https://www.rstudio.com/blog/shiny-contest-2020-is-here/`). Uma amostra da diversidade de alicações passíveis de serem desenvolvidas pode ser consultada na geleria online do Shiny em `https://shiny.rstudio.com/gallery/`. No entanto, apesar da sua crescente popularidade, a comunidade `Shiny` ainda é relativamente reduzida (em comparação com outras do espectro de desenvolvimento *web*) e são muito poucas as aplicações desenvolvidas no meio académico. Não obstante, começa a surgir, no meio empresarial, uma tendência crescente para a comercialização de aplicações com alto nível de qualidade visual e técnico, à medida do cliente. Estas últimas, integram as linguagens web (HTML, JavaScript, CSS, entre outras) de forma avançada, de forma a atingirem um patamar claramente superior e com valor comercial. De entre vários *players* que desenvolvem a sua atividaqde nesta área, destaca-se a empresa *Appsilon* (`https://appsilon.com/`), que dedica exclusivamente a sua atividade ao desenvolvimento e comercialização de aplicações `Shiny` de alto nível.

## 1.4 Objetivos

Esta dissertação tem como objetivo o desenvolvimento de uma aplicação em `Shiny` (nomeada de `AnalyzeR`) que permita e facilite a prossecução de análises estatísticas complexas, sem que o utilizador tenha necessidade de programar.

CAPÍTULO 2

# Métodos

## 2.1 Funcionamento

Uma aplicação em `Shiny` caracteriza-se pela sua *reatividade*, destacando-se assim do paradigma *imperativo* de um *script* padrão de `R`. Esta reatividade permite a atualização de outputs como gráficos e tabelas sempre que se verifica a existência ou alteração de *inputs* do utilizador. Para entender melhor a diferença entre o paradigma *reativo* e *imperativo*, é necessário compreender o conceito de *reatividade*. Numa estrutura *imperativa*, se codificarmos $Y = A + B$, sabemos que $Y$ é atribuída à soma dos termos $A$ e $B$ previamente definidos, portanto, $Y$ não mudará quando os valores de A e B forem alterados sem que $Y$ seja reavaliado. No contexto da programação *reativa*, o valor de $Y$ é atualizado instantaneamente, incluindo todas as outras variáveis e *outputs* dependentes de $Y$, sempre que $A$ ou $B$ forem alterados. A estrutura reativa permite que os inputs do utilizador sejam avaliados por através de uma UI (*user interface*) com uma série de elementos programáveis, como caixas de texto, botões e menus. Uma aplicação *Shiny* apresenta três componentes fundamentais: *Global*, $UI$ e *server*. O primeiro serve para inclusão de todos os pacotes e elementos constantes (variáveis, links, idem) necessários para o funcionamento da aplicação. A $UI$ contém todo o código que define o que é mostrado para o utilizador. A $UI$ também é responsável por permitir a recolha e saída de *inputs* e *outputs* processados pelo *server*. Este útlimo tem como função comunicar com o servidor para a execução das funcionalidades programadas no `Shiny`.

## 2.2 AnalyzeR

A aplicação `AnalyzeR` foi desenvolvida em R [2], com o IDE RStudio [3]. As dependências necessárias para o bom funcionamento da aplicação encontram-se sob a forma de

ficheiros markdown (MD e RMD), localizados no mesmo diretório que o ficheiro app.R (contém o código fonte). Adicionalmente, código LaTeX, HTML e CSS foi utilizado para personalizar texto matemático, *layouts* e *aparência*, respetivamente. Todos os ficheiros base de imagem (para logótipos, por exemplo) encontram-se no formato PNG ou JPEG, sendo localizados e chamados a partir do *path* relativo *./www*. Em baixo, encontram-se listados os pacotes R (ver Tabela 2.1) que a aplicação requer para a execução os métodos estatísticos implementados.

**Tabela 2.1:** Pacotes utilizados no *AnalyzeR*.

| | | |
|---|---|---|
| `broom` [50] | `BSDA` [46] | `corrr` [59] |
| `dashboardthemes` [26] | `DataExplorer` [62] | `DescTools` [20] |
| `dlookr` [34] | `DT` [28] | `epitools` [29] |
| `factoextra` [41] | `FactoMineR` [42] | `fontaweome` [52] |
| `fresh` [64] | `GGally` [40] | `ggeffects` [53] |
| `ggforce` [61] | `ggfortify` [37] | `ggplot2` [33] |
| `gt` [49] | `gtsummary` [51] | `heatmaply` [58] |
| `hrbrthemes` [47] | `knitr` [63] | `markdown` [35] |
| `mice` [27] | `naniar` [31] | `NbClust` [22] |
| `openxlsx` [45] | `plotly` [36] | `psych` [39] |
| `purrr` [44] | `see` [55] | `shiny` [24] |
| `shinyalert` [54] | `shinycustomloader` [56] | `shinydashboard` [25] |
| `sjPlot` [60] | `summarytools` [43] | `survival` [30] |
| `survminer` [38] | `tidyr` [21] | `viridis` [57] |
| `vov` [65] | | |

CAPÍTULO 3

# AnalyzeR

Como já foi elucidado no primeiro capítulo do presente documento, as aplicações desenvolvidas em *Shiny* funcionam segundo um paradigma *reactivo*. Cada *output* é *executado* (pelo *server*) e *atualizado* quase instantaneamente segundo o *input* (no $UI$) do utilizador. Isto implica uma ligação permanente entre o $UI$ e *server*. Nestes ambientes, todos e quaisquer objetos têm que ser do tipo *reactivo* ou têm que ser *encapsulados* em funções reativas para responderem às atualizações de *input* e *output*. Todo o espaço que se encontra fora das funções de $UI$ e *server* é considerado im ambiente *golbal*. Aqui poderão ser chamados e carregados todos os pacotes necessários (pelo menos o pacote `Shiny`). Este espaço também permite a definição de elementos constantes (não passíveis de alteração ou reatividade) como *links externos*, dados importados *localmente* por um *diretório* fixo ou mesmo código de CSS, por exemplo. A figura seguinte sumaria, de forma esquemática, o funcionamento e arquitetura do `AnalyzeR` (ver Figura 3.1):

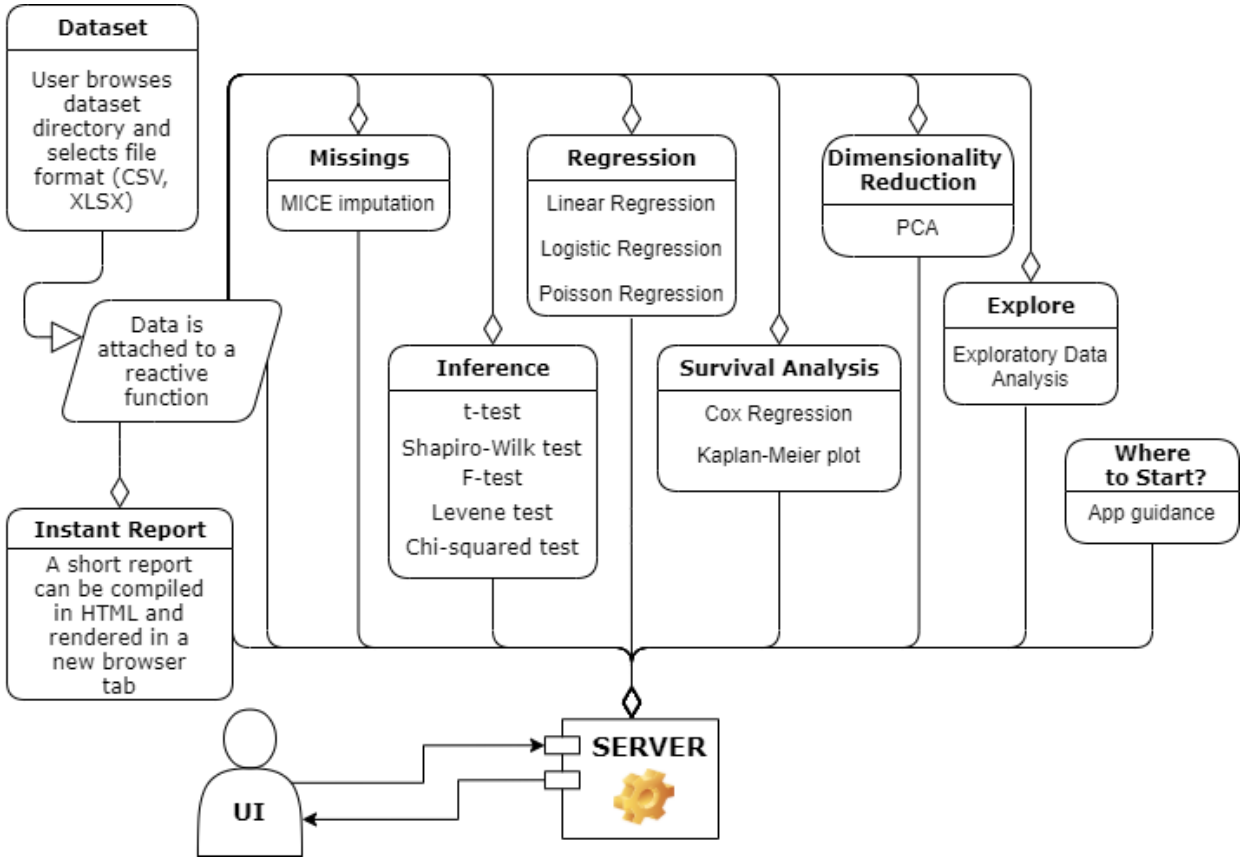


**Figura 3.1:** Arquitetura do AnalyzeR.

De seguida, srão apresentados os fundamentos teóricos, acompanhados de explicações demonstrativas das várias funcionalidades da aplicação, com utilização de dados exemplificativos, patentes em [71] e [72].

## 3.1 Página de Orientação

Esta página, designada de *"Where to Start?"* tem como objetivo orientar o utilizador para o método estatístico mais adequado ao seu objetivo e natureza dos seus dados (ver Figura 3.2). Adicionalmente, é disponibilizada uma pequena introdução com base na seleção de método no *"I want to check..."*. Estas foram desenvolvidas a partir de ficheiros Markdown em conjugação com elementos de HTML e LaTeX.

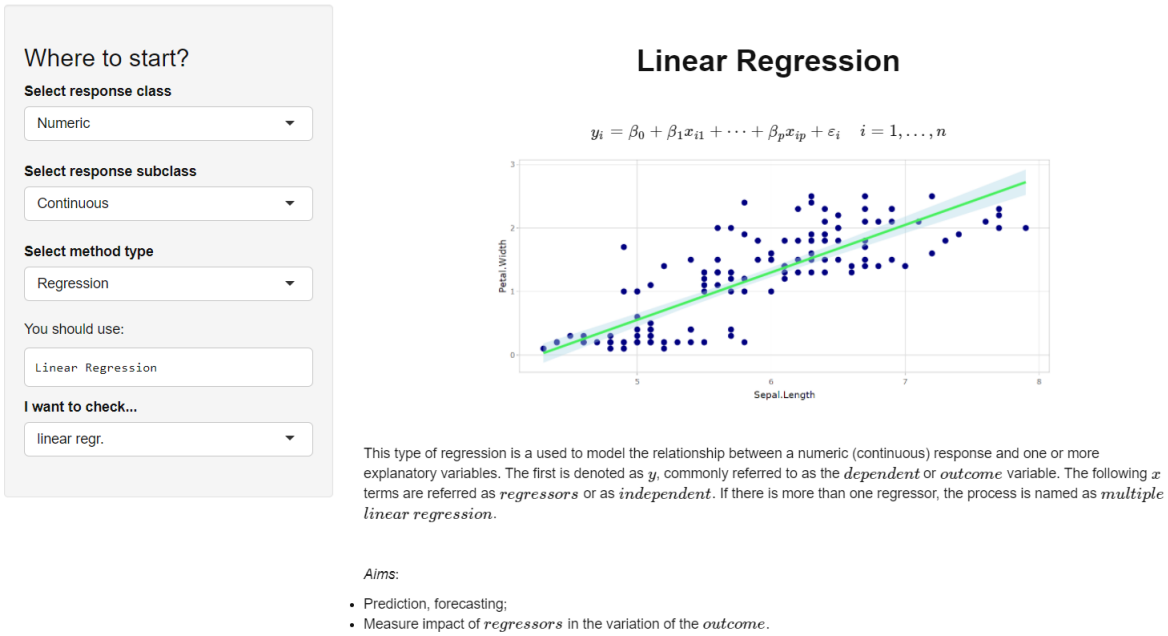


**Figura 3.2:** Página orientação *"Where to Start?"*.

## 3.2 Importação de dados

Nesta página, o utilizador poderá importar dados para o AnalyzeR sob o formato de `xlsx` e `csv`. Uma vez completo o processo de importação, estes são associados a uma *função reativa*, a partir da qual se poderá chamar os dados importados para aceder às respetivas variáveis e incluir os mesmos nos diferentes métodos implementados na aplicação. Não obstante, o utilizador só terá de atentar para a seleção correta dos inputs segundo a natureza dos dados e método a utilizar. Para uma correta importação de dados, o utilizador deve seguir os seguintes passos (ver também as Figuras 3.3 e 3.4):

1. Localizar-se na página *Dataset* no menu lateral.
2. Selecionar o formato do ficheiro a importar.
3. Clicar no botão *Browse* para selecionar o ficheiro.
4. Se as seleções anteriores estiverem corretas, aparecerá as primeiras 10 observações (*head*) dos dados numa tabela interativa.

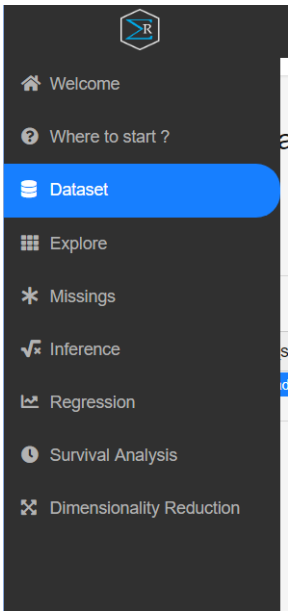


**Figura 3.3:** Seleção no menu lateral.

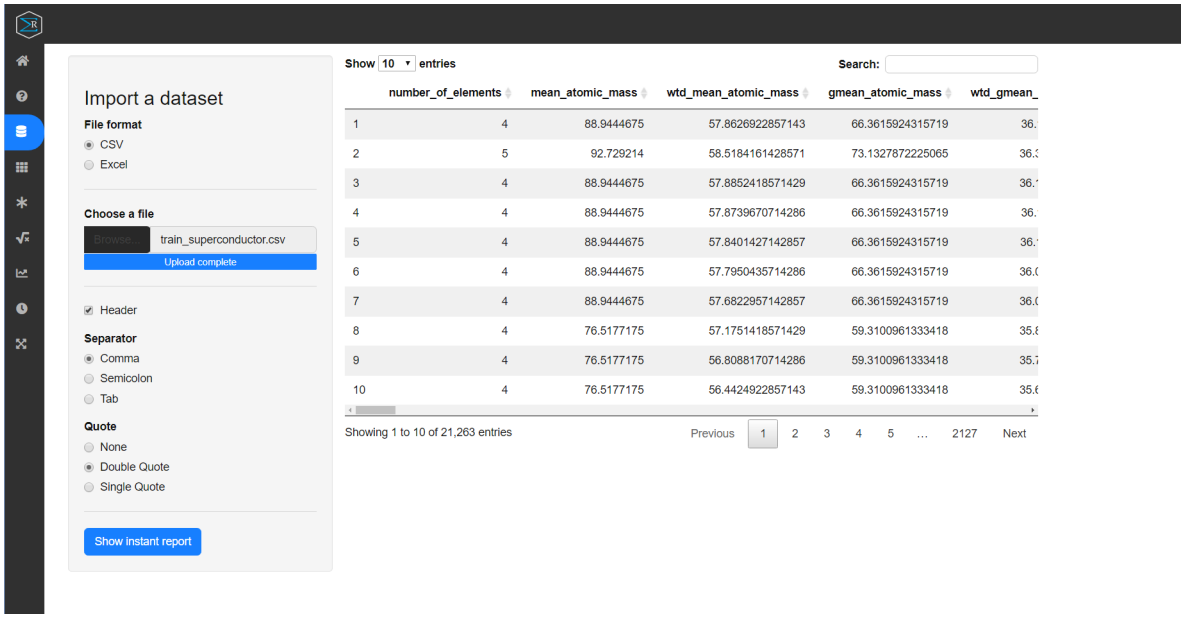


**Figura 3.4:** Página *Dataset*.

Adicionalmente, é possível gerar um relatório de análise exploratória, compilado em formato HTML recorrendo ao pacote *"Data Explorer"* [62]. Este aparecerá numa nova janela do browser. Este relatório contém vários sumários, visualizações como histogramas, gráficos de barras, informações sobre valores omissos, um *heatmap* de correlações e uma análise *PCA* para as variáveis numéricas. A utilização deste relatório não é recomendada para dados de grandes dimensões pois pode exponenciar o tempo de compilação. A Figura 3.5 exemplifica a porção superior do relatório, onde se encontram listados os conteúdos.

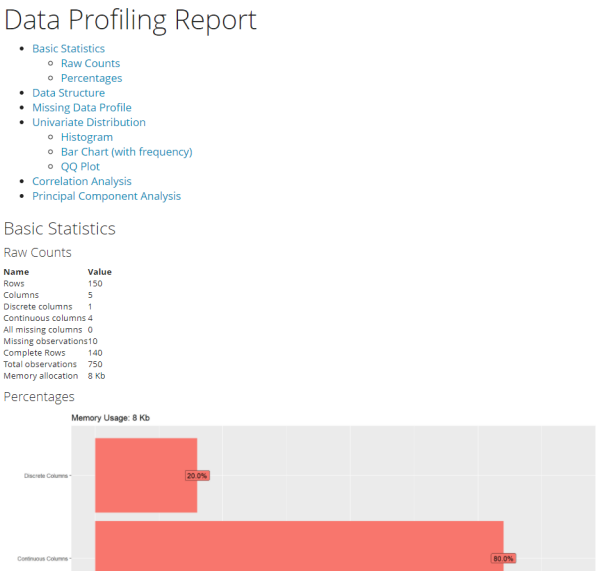


**Figura 3.5:** Porção superior do relatório de análise exploratória.

## 3.3 Análise Exploratória

### 3.3.1 Tabelas Sumário

Uma das primeiras secções da página *"Explore"* apresenta duas tabelas que sumariam os dados. A primeira apresenta alguns atributos e estatísticas básicas relativas a cada variável, bem como um gráfico indicativo da sua distribuição (variáveis numéricas) ou frequência de categorias (variáveis categóricas). Esta tabela (*General Summary*) é gerada automaticamente através do pacote *SummaryTools* (ver Figura 3.6) .

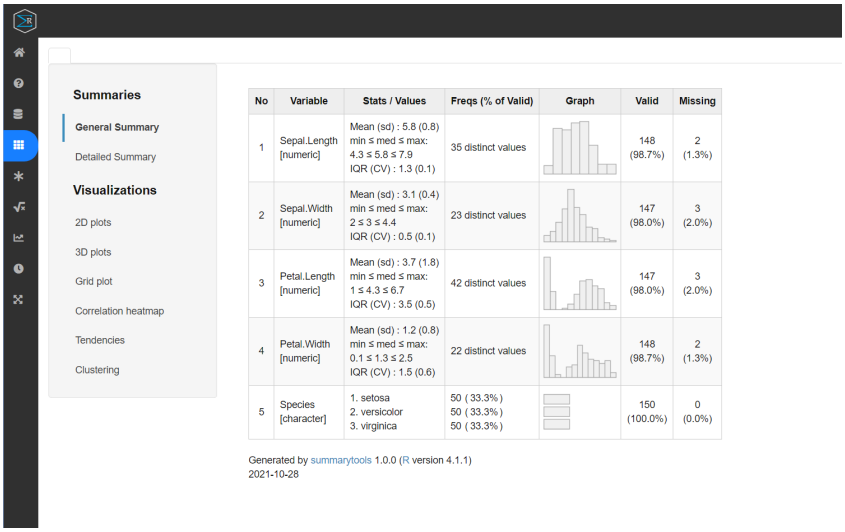


**Figura 3.6:** Tabela sumário.

Na segunda tabela de sumário (*Detailed Summary*), o utilizador poderá obter a mesma com destaque para uma variável categórica (ver Figura 3.7). Adicionalmente, este sumário é complementado com testes estatísticos.

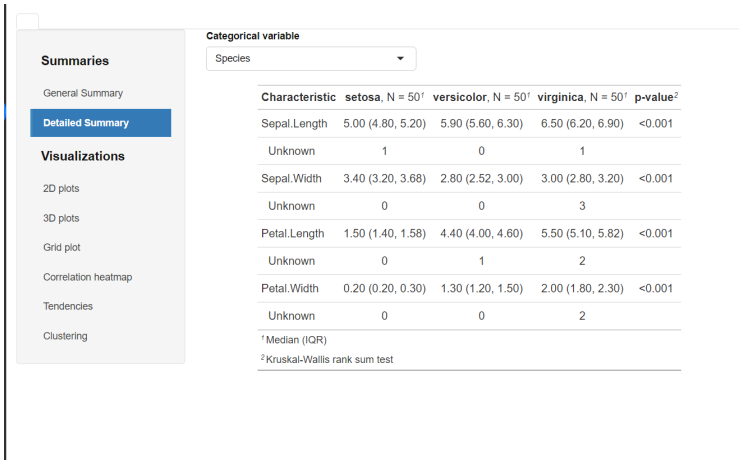


**Figura 3.7:** Tabela sumário face a um fator.

### 3.3.2 Visualização

#### Gráficos de 2 dimensões

Nesta secção, o utilizador terá de selecionar uma variável numérica para visualizar os histogramas, caixas de bigodes e gráficos de dispersão. Adicionalmente, o utilizador pode selecionar a opção de adicionar grupos. Se este for o caso, todos estes gráficos serão sub-divididos e/ou coloridos por cores de acordo com as categorias da variável categórica selecionada para o agrupamento (ver Figuras 3.8 e 3.9).

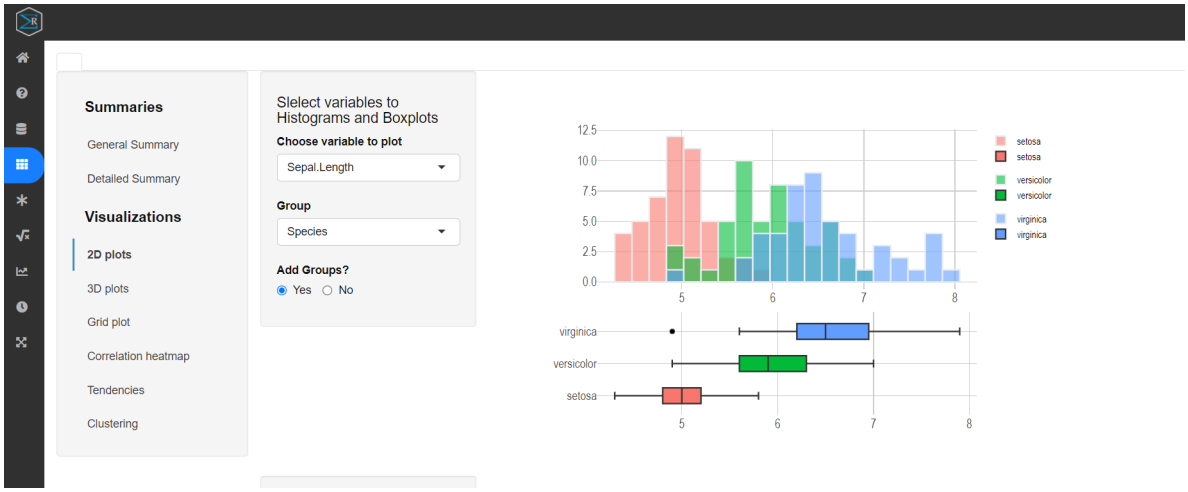


**Figura 3.8:** Histogramas e caixas de bigodes.

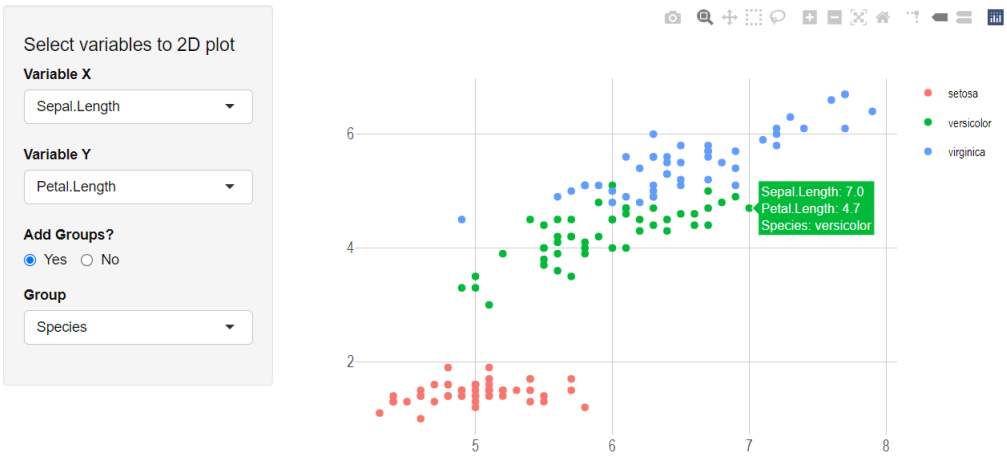


**Figura 3.9:** Gráfico de dispersão.

### Gráficos de 3 dimensões

No `AnalyzeR` também são disponibilizadas visualizações de dispersão com 3 dimensões. Nestas, o utilizador deve selecionar 3 variáveis numéricas distintas para cada um dos eixos e uma variável adicional para colorir os pontos. Se esta variável for do tipo numérico, será atribuído um gradiente, se esta for categórica, os pontos serão coloridos com cores distintas correspondentes às categorias da variável de agrupamento. As Figuras 3.10 e 3.11 exemplificam este procedimento.

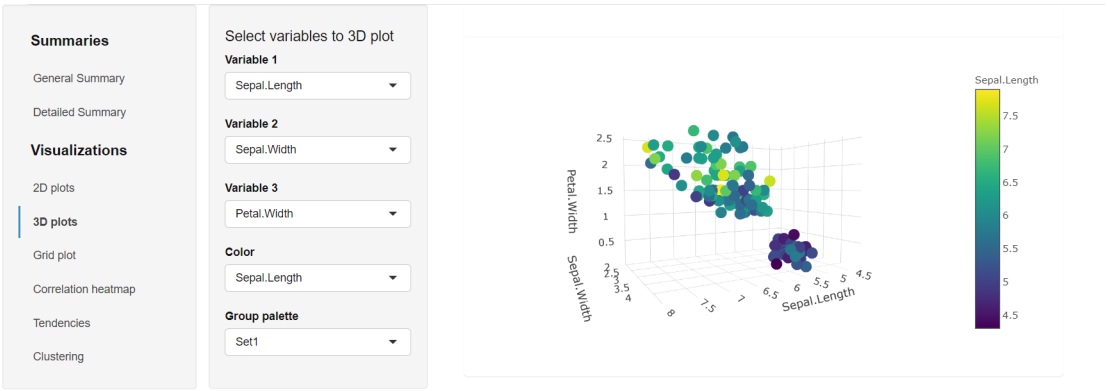


**Figura 3.10:** Gráfico de dispersão 3D com gradiente.

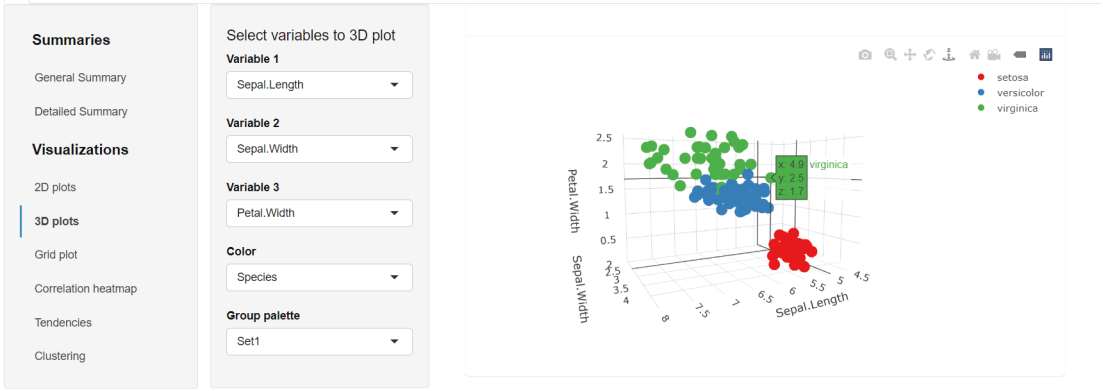


**Figura 3.11:** Gráfico de dispersão 3D com grupos.

### Visualização em Grelha

Na secção *"Grid plot"* é possível obter uma visualização que combina vários tipos de gráficos, cruzando das diferentes variáveis (ver Figura 3.12). Para gerar esta visualização, o utilizador terá de selecionar as variáveis a incluir na mesma. Os gráficos serão gerados automaticamente.


**Figura 3.12:** Gráficos em grelha.

### Correlação

Na secção de *Correlation Heatmap* é gerado automaticamente um *heatmap* interativo de correlação de Pearson com todas as variáveis numéricas (ver Figura 3.13). Os valores dos coeficientes de correlação são traduzidos em cores retiradas de um gradiente. A interatividade deste gráfico permite que o utilizador possa verificar o coeficiente e o par de variáveis correspondente ao levar o cursor às células. É ainda possível fazer *zoom in* para visualizar melhor certas porções, se a visualização ficar muito densa por haver um grande número de variáveis.

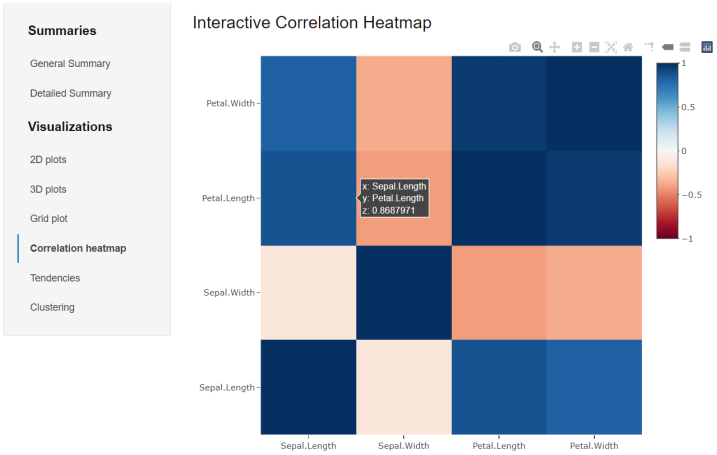


**Figura 3.13:** *Heatmap* de correlação interativo.

### Tendências

Nesta secção é possível visualizar tendências lineares entre duas variáveis numéricas. Estas linhas de tendência obtidas a partir da regressão linear (opção *lm*), logística binária (*glm*, família *binomial*) , Poisson (*glm*, família *poisson*) ou mesmo como regressão *loess* (*locally estimated scatterplot smoothing* [70]). Para este último caso, é possível variar o parâmetro de amaciamento (ver Figura 3.14).

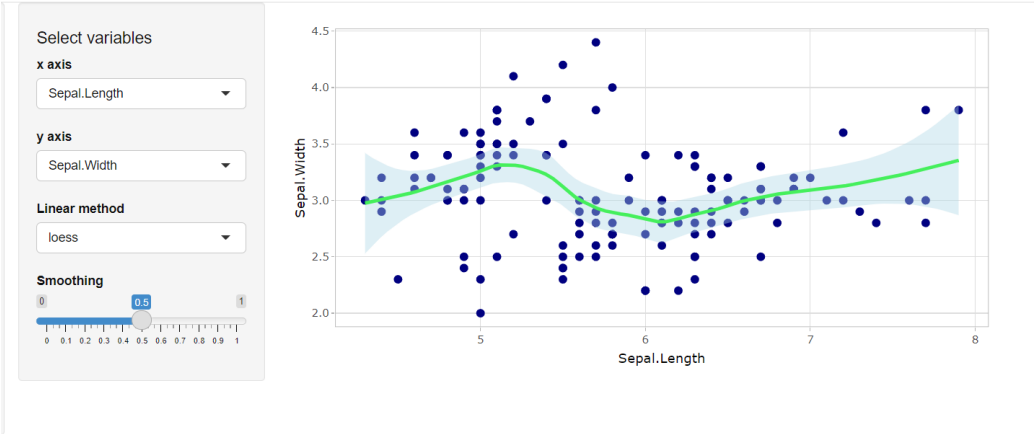


**Figura 3.14:** Tendência linear *loess*.

### Agrupamento em Clusters

Na secção de *Clustering* é possível visualizar a melhor partição estimada por vários índices, segundo o algoritmo selecionado. Este processo é efetuado com a distância euclidiana e método selecionado através do pacote *NbClust* [22]. O utilizador terá de selecionar duas variáveis numéricas distintas assim como o algoritmo de *clustering* (ver Figura 3.15).

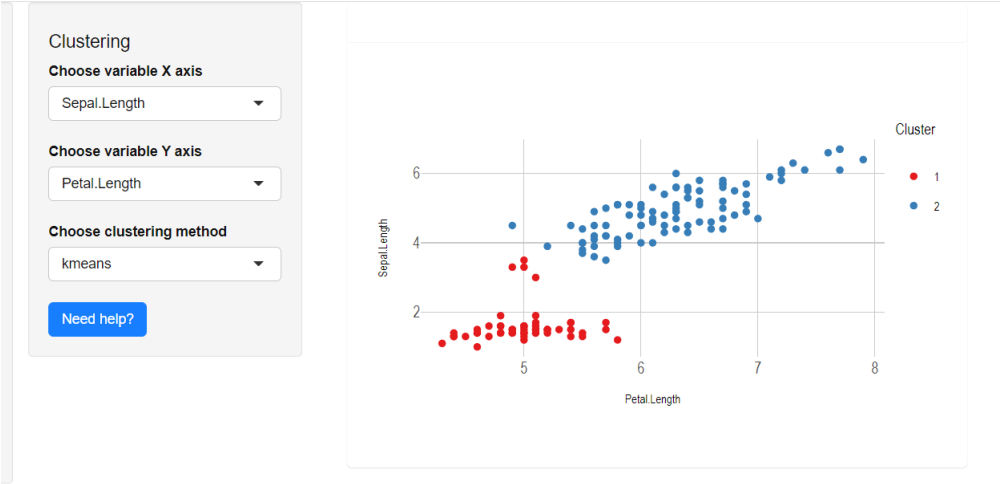


**Figura 3.15:** Visualização de *Clusters*.

Os métodos de clustering implementados no `Analyzer` para o *NbClust* estão listados na Tabela 3.1.

**Tabela 3.1:** Métodos de clustering.

| | |
|---|---|
| `ward.D` | `average` |
| `ward.D2` | `mcquitty` |
| `single` | `median` |
| `complete` | `centroid` |
| `kmeans` | |

Por defeito, o número ótimo de clusters é estimado por vários índices (alguns destes estão listados na Tabela 3.2). O número de *clusters* mais *votado* é então utilizado para estimar a partição ótima segundo o método (algoritmo de clustering) selecionado.

**Tabela 3.2:** Alguns dos índices utilizados pelo NbClust.

| | |
|---|---|
| `kl` | `hartigan` |
| `duda` | `beale` |
| `hubert` | `ptbiserial` |
| `mcclain` | `silhouette` |
| `dindex` | `friedman` |

## 3.4 Imputação de Valores Omissos

### Teoria

No `AnalyzeR`, a secção *"Missings"* permite a vizualização e imputação de valores em falta. O processo de imputação recorre ao algoritmo MICE (*Multiple Imputation Chained Equations*) [17]. Durante o processo de imputação, o MICE identifica as variáveis que contêm valores em falta e preenche-os (imputação) de acordo com estimativas obtidas através de modelos de regressão para todas as variáveis com com observações incompletas (os modelos de regressão são selecionado em concordância com a classe da varável [numérica: regr. Linear, categórica: regr. Logística, idem]). Cada variável com *missings* funciona como uma variável dependente e as restantes como independentes. A imputação pode ser autalizada por iterações, de forma a obter melhores estimativas para os valores imputados. Finalizada a imputação, o utilizador poderá exporar os dados completos (imputados) para um ficheiro `xlsx`. Os detalhes relacionados com o processamento matemático podem ser consultados em pormenor em [17] e [18].

### Aplicação

Na página *"Missings"*, o utilizador poderá vizualizar a distribuição dos valores em falta e imputá-los através do algoritmo MICE. É ainda possível alterar o número máximo de iterações, no entanto, em dados de grande dimensionalidade, o algoritmo poderá demorar consideravelmente mais tempo a terminar a imputação. O utlizador pode ainda vizualizar (no *"Diagnostics"*) as distribuições de densidade dos valores observados e imputados por variável. É ainda possível executar o *download* dos dados imputados sob o formato de um ficheiro `xlsx` (ver Figura 3.16 e 3.17).


**Figura 3.16:** Visualização de valores em falta.

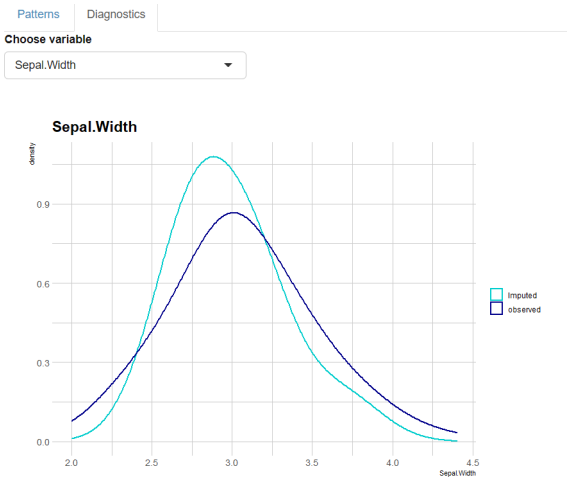


**Figura 3.17:** Densidade de valores imputados e observados.

## 3.5 Testes de Hipóteses

### Teoria

### 3.5.1 Teste-$t$

O teste-$t$ avalia a existência de uma diferença significativa entre as médias de duas amostras numéricas [11].

$$t_{student} = \frac{\bar{x_1} - \bar{x_2}}{\sqrt{\frac{s_p^2}{n_1} + \frac{s_p^2}{n_2}}}, \qquad for \quad n_1 = n_2, \quad \sigma_1 = \sigma_2 \tag{3.1}$$

onde $\bar{x_1}$ e $\bar{x_2}$ denotam a média da primeira e segunda amostra, respetivamente. As estatísticas de desvio padrão singular e emparelhado correspondem a $\sigma$ e $s_p$, respetivamente. As hipóteses em teste são as seguintes:

- $H_0$: A diferença entre as médias amostrais é igual a 0.
- $H_1$: A diferença entre as médias amostrais *não* é igual a 0.

Existem diferentes tipos de testes-$t$:

- Amostras *emparelhadas*: as amostras têm origem no mesmo grupo populacional.

- *Independentes*: as amostras não têm origem no mesmo grupo populacional.
- *Uni-amostral*: uma amostra comparada com um valor.

Para a realização de um teste-$t$ é necessário que as amostras:

- Sejam aleatórias e numéricas.
- Sigam a distribuição normal.

### 3.5.2 Teste do $\chi^2$ (Qui-quadrado)

A associação entre duas variáveis do tipo categórico é frequentemente avaliada utilizando o teste $\chi^2$ de Pearson [10]. Este envolve a avalia as frequências presentes nas células de uma tabela de contingência e avalia a diferença entre as observadas e esperadas. A estatística de teste $\chi^2$ é dada por

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \qquad i = 1, \ldots, n. \tag{3.2}$$

onde $O_i$ representa o número de observações (contagem) do grupo $i$, $E_i$ corresponde à frequência esperada do grupo $i$ e $n$ representa o número de células na tabela de contingência. As hipóteses em teste são as seguintes:

- $H_0$: a proporção respetiva a cada categoria é a mesma.
- $H_1$: a proporção respetiva a cada categoria *não* é a mesma.

Este teste tem como pressuposto a independência das variáveis avaliadas.

### 3.5.3 Teste à normalidade de Shapiro-Wilk

O teste de Shapiro-Wilk [13] é amplamente utilizado para testar se uma amostra ou variável de teor numérico segue a distribuição normal. A estatística de teste é dada por:

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \qquad i = 1, \ldots, n. \tag{3.3}$$

onde $x_{(i)}$ segue uma indexação ordenada $i$ (distinto do $i$ associado à obsevração) e $a$ representa um coeficiente utilizado na estimação de $W$. A hipóteses em teste são:

- $H_0$:A população é normalmente distribuída.
- $H_1$:A população *não* é normalmente distribuída.

### 3.5.4 Teste à Correlação

O termo *correlação* é geralmente utilizado para se referir a uma associação entre duas variáveis. O coeficiente de correlação entre duas variáveis aleatórias é definido como a razão da covariância entre essas duas variáveis e o produto de seus respectivos desvios-padrão (i.e, uma medida de associação linear entre duas variáveis e assume valores entre -1 e 1). Os coeficientes de correlação mais comumente utilizados, para variáveis contínuas, são o produto-momento de Pearson [12] e o coeficiente de correlação de Spearman. Este último caracteriza-se por ser um método não paramétrico e apresenta uma menor sensibilidade a *outliers* do que o coeficiente de Pearson. Este último é dado por:

$$\rho_{Pearson} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \qquad i = 1, \ldots, k. \tag{3.4}$$

enquanto que o coeficiente de Spearman é dado por:

$$\rho_{Spearman} = 1 - \frac{6 \sum d_i^2}{n - (n^2 - 1)} \qquad i = 1, \ldots, n. \tag{3.5}$$

onde $d_i = rank(X_i) - rank(Y_i)$ representa a diferença entre dois níveis (*ranks*) de cada observação e $n$ denota o número de observações. Para testar a diferença entre correlação, as hipóteses são as seguintes (generaliza-se o coeficiente de correlação por $c$):

- $H_0$: $\rho_1 = \rho_2$
- $H_1 : \rho_1 \neq ou > ou < \rho_2$

### 3.5.5 Teste à Homogeneidade de Variâncias

**Teste-F**

Este teste é utilizado para determinar se a variância de duas amostras (preferencialmente numéricas e normalmente distribuídas) é diferente [16]. A homogeneidade de variâncias constitui um pressuposto de várias técnicas estatísticas, em particular para análises de variância. A estatística de teste é dada por:

$$F = \frac{\sigma_1^2}{\sigma_2^2} \tag{3.6}$$

onde $\sigma_1^2$ e $\sigma_2^2$ denotam as variâncias da primeira e segunda amostra, respetivamente.

As hipóteses em teste são as seguintes:

- $H_0 : \sigma_1^2 = \sigma_2^2$
- $H_1 : \sigma_1^2 \neq ou > ou < \sigma_2^2$

**Teste de Levene**

O teste de Levene testa a homogeneidade de variâncias entre $k$ amostras [15]. Apresenta-se como uma alternativa ao teste-F e ao de Bartlett uma vez que é menos sensível a desvios da normalidade. A estatística do teste de Levene é dada por:

$$W = \frac{(N-k)}{(k-1)} \frac{\sum_{i=1}^{k} N_i (\bar{Z}_{i.} - \bar{Z}_{..})^2}{\sum_{i=1}^{k} \sum_{j=1}^{N_i} (Z_{ij} - \bar{Z}_{i.})^2} \qquad i = 1, \ldots, k. \tag{3.7}$$

onde $Z_{ij} = |Y_{ij} - \tilde{Y}_{i.}|$ e $\tilde{Y}_{i.}$ é a mediana do grupo $i$. É possível que $\tilde{Y}_{i.}$ represente [por opção] outras estatísticas como a *média*, no entanto a mediana produz uma estimativa mais robusta.

As hipóteses em teste são:

- $H_0 : \sigma_1^2 = \sigma_2^2 = \cdots = \sigma_k^2$
- $H_1$ : Existe, pelo menos, um valor de variância $\sigma_1^2$ $(i = 1, ..., k)$ significativamente diferente dos restantes.

### Aplicação

Na página de *"Inference"* é possível realizar testes de hipóteses, nomeadamente testes-t, teste de Shapiro-Wilk, teste-F, teste de Levene, teste do $\chi^2$ e um teste à correlação. Para cada um destes é possível selecionar as duas variáveis (uma variável numérica e outra de grupo, no caso do teste de Levene) bem como ajustar diferentes parâmetros como a natureza da hipótese alternativa, o nível de confiança $\alpha$, método de correlação e outros (ver Figura 3.18).

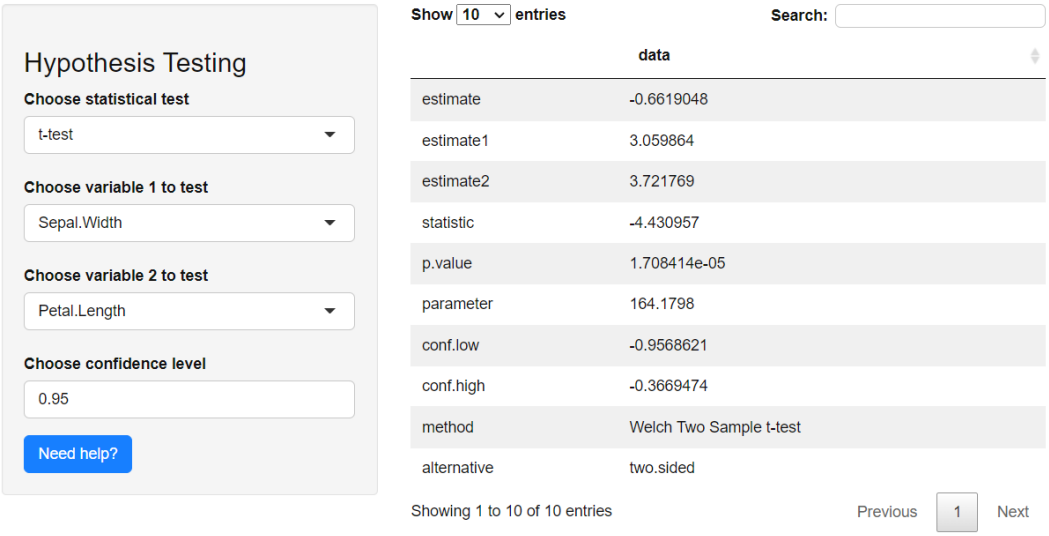


**Figura 3.18:** Página de *Inference*.

## 3.6 Regressões

### Teoria

### 3.6.1 Regressão Linear

Este tipo de regressão é utilizado para modelar variáveis do tipo numérico, permitindo estimar a influência de certas variáveis na varição de uma resposta numérica de interesse [67]. O modelo de regressão linear pode ser formulado na sua forma condensada da seguinte forma:

$$y = X\beta + \varepsilon \tag{3.8}$$

Onde $Y$, $\beta$, $X$ e $\varepsilon$ representam as observações da variável dependente, as estimativas que traduzem a influência de cada preditor em $Y$ e os termos de erro, respetivamente. Assim, para o modelo mais simples (só com um preditor) têm-se:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \qquad i = 1, \ldots, n. \tag{3.9}$$

As estimativas $\beta$ são calculadas por estimação de mínimos quadrados, de forma a otimizar o ajuste. Estes coeficientes podem ser interpretados como a influência do preditor correspondente no incremento de uma unidade no valor da variável dependente. A estimação de $\beta_0$ e $\beta_{1,\ldots,n}$ é efetuada da seguinte forma:

$$\hat{\beta}_0 = \left(\sum_i y_i - \hat{\beta}_1 \sum_i x_i\right)/n \qquad i = 1, \ldots, n. \tag{3.10}$$

$$\hat{\beta}_1 = \frac{n\sum_i x_i y_i - \sum_i x_i \sum_i y_i}{n\sum_i x_i^2 - (\sum_i x_i)^2}, \qquad i = 1, \ldots, n. \tag{3.11}$$

Se o modelo incluir mais do que uma variável, passa a designar-se por *regressão linear multivariada* e pode ser representado por:

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i \quad i = 1, \ldots, n \tag{3.12}$$

Ao realizar este tipo de regressão, o utilizador deverá ter o cuidado de diagnosticas a sua validade e a qualidade do ajuste. Para tal é necessário atentar para algumas estatísticas como o VIF (*Variance Inflation Factor*) que determina a existência de multicolinearidade entre preditores, a normalidade dos resíduos e também o valor do coeficiente de determinação $r^2$, que avalia a qualidade do ajuste. No `AnalyzeR`, estas medidas estão incluídas num diagnóstico gráfico, na página *"Regression"*.

### 3.6.2 Regressão Logística

A regressão logística [68] insere-se nos modelos lineares generalizados e permite modelar respostas categóricas. Para este efeito, é utilizada uma função canónica $g$. No caso da regressão logística binária a função canónica é a *logit*. Formalmente:

$$g(E[y]) = logit(\pi) = X\beta \tag{3.13}$$

portanto, para o modelo mais simples, têm-se:

$$logit(\pi_i) = \beta_0 + \beta_1 x_i \qquad i = 1, \ldots, n. \tag{3.14}$$

onde $\pi_i = Pr(y_i = 1|x_i)$ e $logit(\pi_i) = ln[\pi_i/(1-\pi_i)]$. Os parâmetros $\beta$ são estimados por máxima verosimilhança e podem ser exponenciados para permitir a sua interpretação como *Odds Ratio* (OR). No `AnalyzeR`, o diagnóstico da regressão logística é avaliado através da análise do desvio do modelo executado face ao modelo vazio, em relação ao *intercept*.

### 3.6.3 Regressão de Poisson

Este tipo de modelo generalizado é utilizado para modelar variáveis dependentes que representam contagens [66]. Na regressão de Poisson, os parâmetros são estimados pelo método da máxima verosimilhança. O modelo de Poisson pode ser formulado da seguinte forma:

$$\mathrm{P}(Y = y|\lambda) = \frac{e^{-\lambda}\lambda^y}{y!} \qquad y = 0, 1, 2, \ldots \tag{3.15}$$

Onde $\lambda$ é interpretado como a taxa média de ocorrência de um evento. Este parâmetro é obtido pela seguinte forma:

$$\lambda = \exp\{\mathbf{X}\beta\}. \tag{3.16}$$

onde $\mathbf{X} = (X_1, \ldots, X_{p-1})$. Pelo que a formulação pode ser expandida para:

$$\mathrm{P}(Y_i = y_i|\mathbf{X}_i, \beta) = \frac{e^{-\exp\{\mathbf{X}_i\beta\}} \exp\{\mathbf{X}_i\beta\}^{y_i}}{y_i!}. \tag{3.17}$$

No `AnalyzeR`, o diagnóstico da regressão de Poisson é avaliado através da análise do desvio do modelo executado face ao modelo vazio, em relação ao *intercept*.

### Aplicação

Na página de *"Regression"* é possível efetuar regressões do tipo linear, Logística Binária e Poisson. É gerada uma tabela que sumaria os principais resultados da regressão. Nesta estão incluídas as estimativas dos ceficientes, os respetivos intevralos de confiança, valores $p$ calculados a partir de testes $Wald$ [69]aos coeficientes e ainda uma estimativa do valor $r^2$ bem como a sua estimativa ajustada (ver Figuras 3.19, 3.20 e 3.21). Adicionalmente é gerado uma visualização dos coeficientes e um diagnóstico visual do modelo efetuado. Nos casos particulares das regressões Poisson e Logística Binária, os sumários conterão os coeficientes sob a forma de *Odds Ratios* e *Incidence Ratios* respetivamente. Nestes casos, o diagnóstico é efetuado por comparação de desvio do modelo efetuado face ao modelo vazio, em relação ao *intercept*.

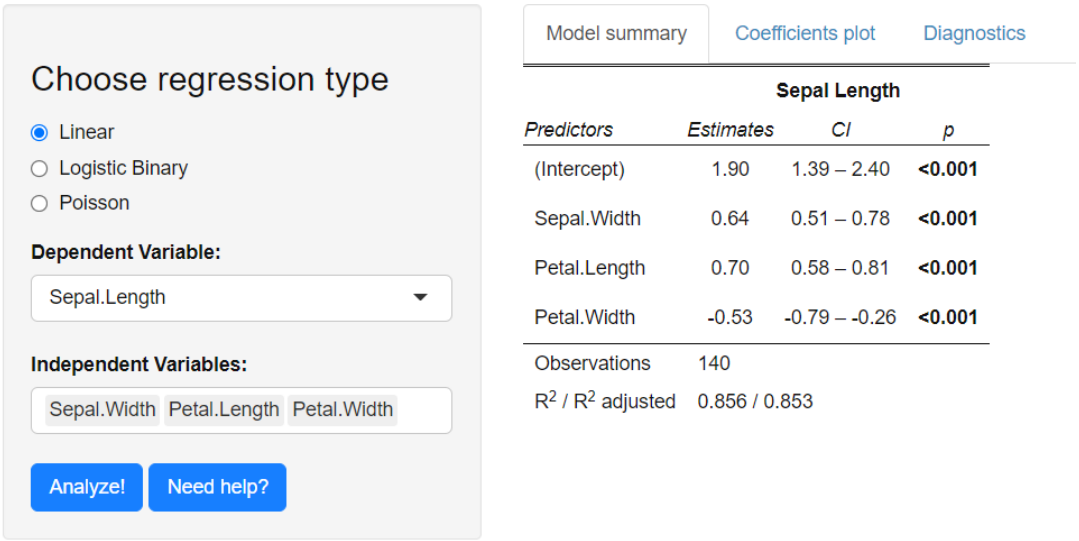


**Figura 3.19:** Página *Regression* com sumário da regressão.

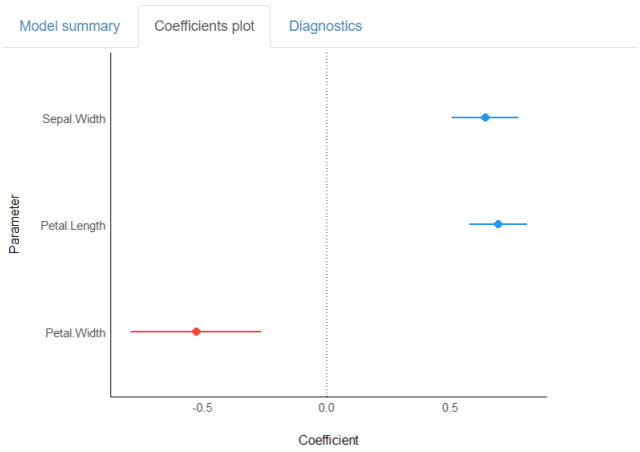


**Figura 3.20:** Gráfico dos coeficientes com o respetivo intervalo de confiança

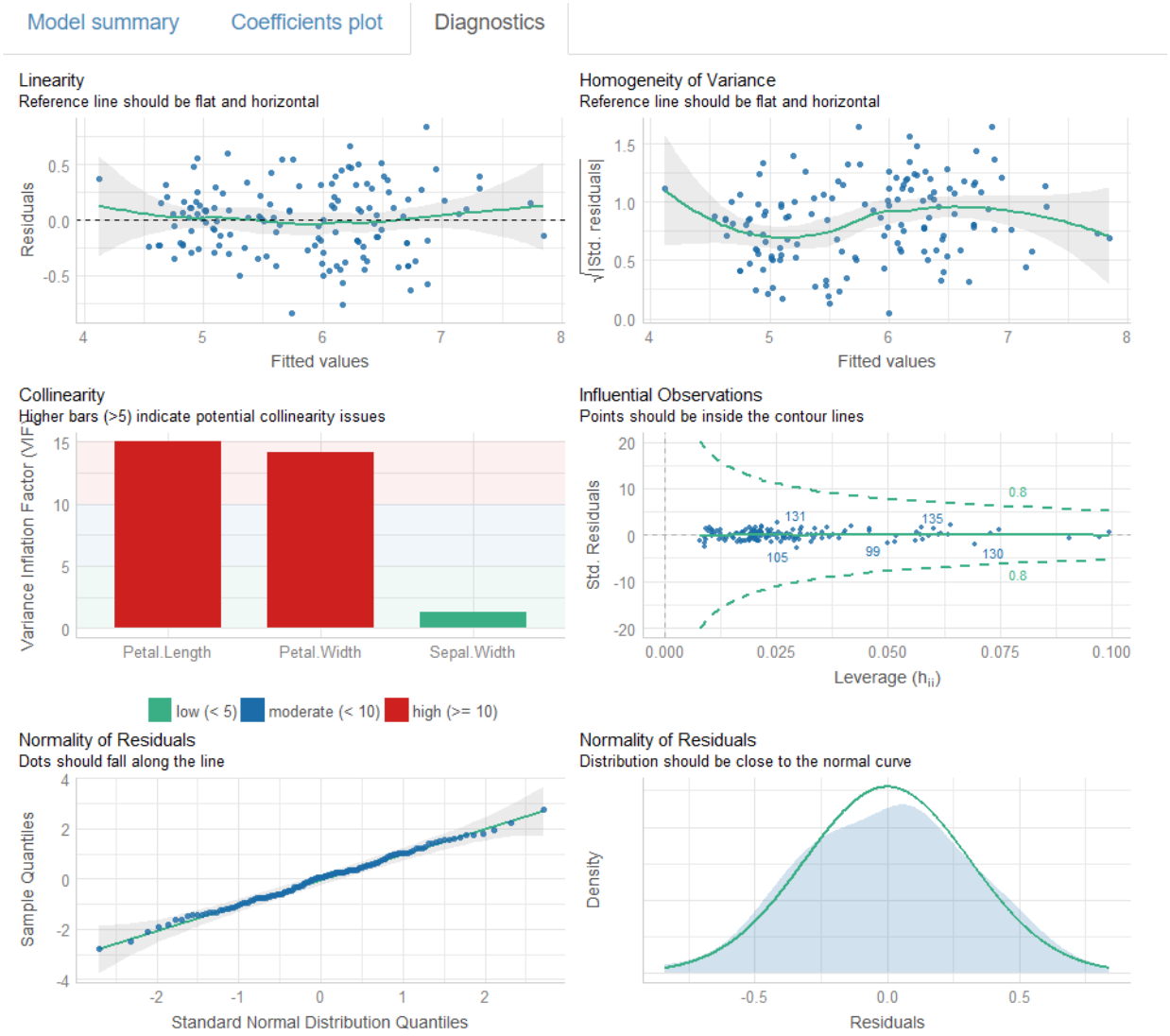


**Figura 3.21:** Diagnóstico visual da regressão linear.

**Teoria**

### 3.7.1 Regressão de Cox

A regressão de Cox [8] (riscos proporcionais) é amplamente utilizada em análises de sobrevivência. Este tipo de análise pretende explicar a relação entre um evento de interesse (*outcome*), em conjugação com o tempo decorrido até à ocorrência desse mesmo evento, e uma ou mais variáveis sobre as quais se pretende testar a sua influência neste processo. As análises de sobrevivência dependem intrinsecamente dos estudos e ensaios clínicos que as precedem. Ensaios estes que frequentemente sofrem de vicissitudes inerentes à investigação clínica como é o caso das desistências de participantes e mesmo a inocorrência de *outcomes* (censura). Este tipo de especificidades invalidam o uso de métodos estatísticos mais usuais como testes-t, análise de variância e outro tipo de regressões. A regressão de Cox para riscos proporcionais pode ser escrita sob forma:

$$h(t) = h_0(t) \times \exp(b_1X_1 + b_2X_2 + b_3X_3 + \cdots + b_kX_k) \tag{3.18}$$

onde $h(t)$ corresponde ao risco esperado no instante $t$, $h_0(t)$ representa o risco basal, i.e. o risco que se obtém quando todos os preditores $(X_1, \ldots, X_k)$ são iguais a 0. $h(t)$ resulta do produto do risco basal $h_0$ com a exponencial da combinação linear dos preditores. Resulta então que estes últimos intervêm proporcionalmente, no seu efeito, para o cálculo do risco estimado. Por vezes, este modelo pode ser reformulado em relação ao risco relativo (quociente da divisão $\frac{h(t)}{h_0(t)}$). Adicionalmente, é possível aplicar o logaritmo natural aos dois lados da expressão, esta operação facilita a interpretação dos coeficientes (em relação ao risco relativo).

$$ln(\frac{H(t)}{H_0(t)}) = b_1X_1 + b_2X_2 + b_3X_3 + \cdots + b_kX_k \tag{3.19}$$

Se um preditor for do tipo binário (1:presente, 0:ausente), $\exp(b_i)$ pode ser interpretado como o *risco relativo instantâneo* de ocorrência de *outcome*, a todo o momento, para um indivíduo com fatores de risco (preditores) presentes face a outro indivíduo com os mesmos fatores de risco, mas ausentes. Se o preditor for de natureza contínua,

$\exp(b_i)$ representa o risco relativo instantâneo, a todo o tempo, para um indivíduo com incremento de 1 no valor deste preditor em comparação com outro indivíduo com os mesmos fatores de risco. O uso apropriado de uma regressão-Cox (com riscos proporcionais), apesar de não enunciar pressupostos quanto à função do risco basal, requer:

- Independência dos tempos de sobrevivência entre indivíduos distintos.
- Risco relativo constante ao longo do tempo.
- Proporcionalidade dos diferentes riscos.
- Associação linear entre o logaritmo natural do risco relativo e os preditores.

### 3.7.2 O estimador de Kaplan-Meier

A função de *Kaplan-Meier* [9] permite estimar a probabilidade de sobrevivência até ao instante $t$. O método do produto limite de kaplan e Meier, é utilizado para estimar a função de sobrevivência $S$:

$$S_t = \prod_{t_i \leq t} \left(1 - \frac{d_i}{n_i}\right) \tag{3.20}$$

onde $t_i$ representa a duração do estudo até ao instante $i$, $d_i$ representa o número de ocorrências de *outcome* (ex: morte) até $i$ e $n_i$ representa o número de indivíduos em risco no instante anterior a $t_i$. Os pressupostos deste método são os seguintes:

- Indivíduos com censura apresentam as mesmas possibilidades de sobrevivência que indivíduos que permanecem no estudo.
- Possibilidade de sobrevivência mantêm-se igual para indivíduos recrutados no início ou já numa fase avançada do estudo.

A visualização gráfica do estimador de Kaplan-Meier é muito utilizada para facilitar a perceção da sobrevivência ao longo do tempo e até mesmo em relação a outros fatores como diferentes tratamentos, por exemplo.

#### Aplicação

Na página *Survival Analysis* é possível realizar análises de sobrevivência via regressões de Cox e visualizações das estimativas de sobrevivência de Kaplan-Meier (ver Figuras 3.22 e 3.23). Neste último caso, é calculado um valor $p$ que traduz a significância da

diferença das curvas de sobrevivência correspondentes a diferentes grupos (tratamentos, por exemplo). O utilizador terá de selecionar a variável que codifica o tempo até ao evento, *outcome*, variável de tratamento (se existir grupos para visualizar nas curvas de sobrevivência), e variáveis independentes para o modelo de Cox.

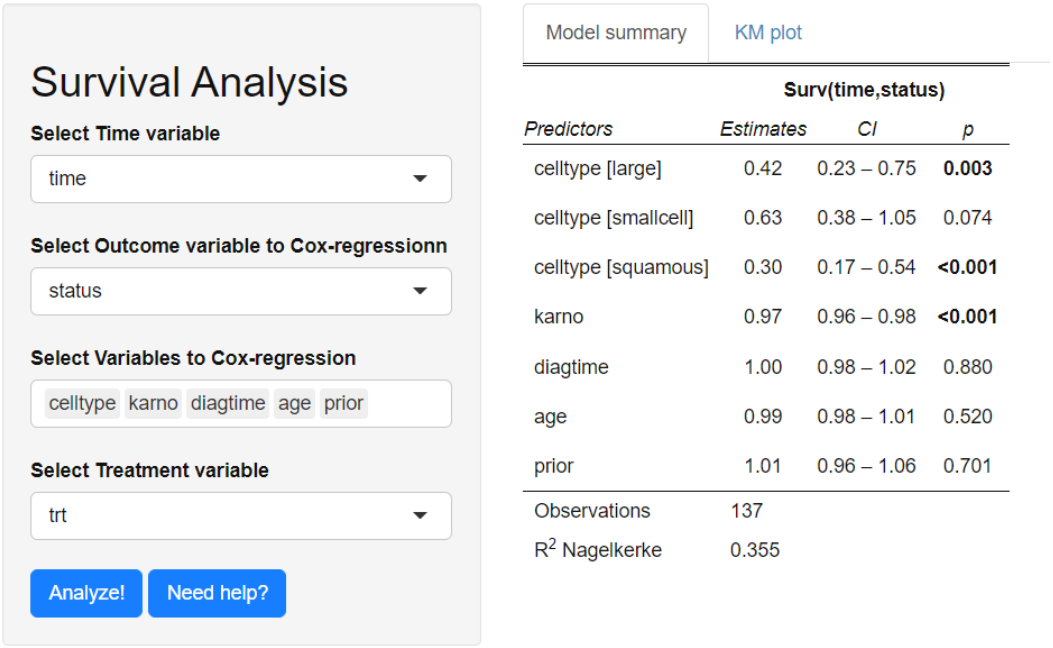


**Figura 3.22:** Página *Survival Analysis* com sumário da regressão de Cox.

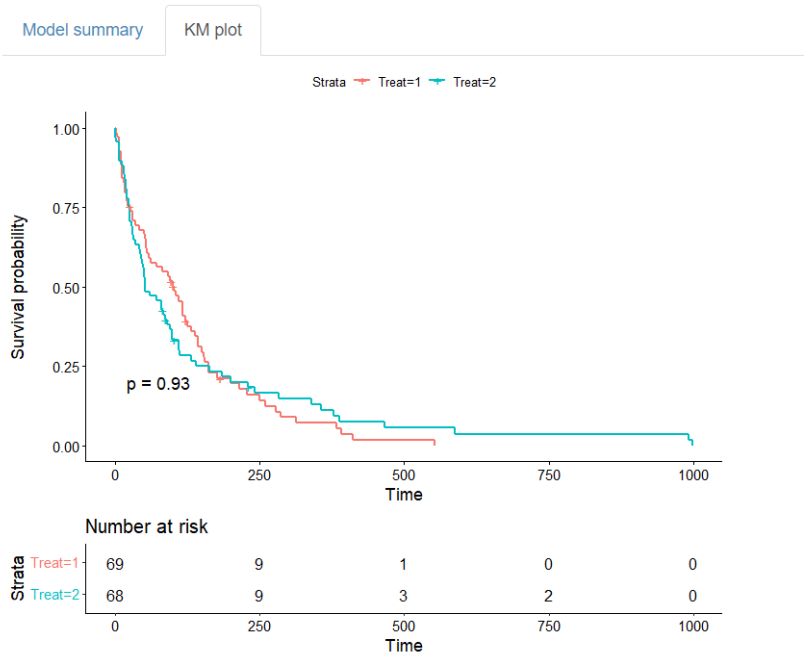


**Figura 3.23:** Curvas de Kaplan-Meier.

**Teoria**

Os métodos de redução de dimensionalidade são utilizados para facilitar a análise, exploração e interpretação de dados com um grande número de variáveis. Este tipo de métodos pretende reduzir o número de variáveis para um reduzido número de combinações lineares interpretáveis das mesmas (componentes). No caso das variáveis serem numéricas, deve-se utilizar o método de *Análise de Componentes Principais* [14].

### 3.8.1 Análise de Componentes Principais

Este método (em inglês, *PCA: Principal Component Anaysis*) pressupõe que todas as variáveis incluídas são do tipo numérico. Se a escalas destas variáveis forem distintas, é recomendável que se proceda previamente à padronização das mesmas. Para padronizar uma variável, subtrai-se a média e divide-se pelo desvio padrão da mesma:

$$Z_{ij} = \frac{X_{ij} - \bar{x}_j}{s_j} \tag{3.21}$$

onde $i$ e $j$ correspondem ao índice de observação e variável, respetivamente. Após a padronização, inserem-se as variáveis como input do $PCA$. Assim, considere-se as mesmas sob a forma de um vetor $Z$.

$$\mathbf{Z} = \begin{pmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{pmatrix} \tag{3.22}$$

onde a matriz de covariância apresenta-se como:

$$\text{var}(\mathbf{Z}) = \Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \dots & \sigma_{1p} \\ \sigma_{21} & \sigma_2^2 & \dots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \dots & \sigma_p^2 \end{pmatrix} \tag{3.23}$$

Assim, as combinações lineares tomam a forma:

$$\begin{aligned} Y_1 &= e_{11}Z_1 + e_{12}Z_2 + \cdots + e_{1p}Z_p \\ Y_2 &= e_{21}Z_1 + e_{22}Z_2 + \cdots + e_{2p}Z_p \\ &\vdots \\ Y_p &= e_{p1}Z_1 + e_{p2}Z_2 + \cdots + e_{pp}Z_p \end{aligned} \tag{3.24}$$

sendo que estas podem ser conceptualizadas como regressões lineares, que prevêm $Y_i$ a partir de $Z_p$. A maximização da informação captada é efetuada através da maximização da variância explicada:

$$\text{var}(Y_i) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{ik}e_{il}\sigma_{kl} = \mathbf{e}_i'\Sigma\mathbf{e}_i \tag{3.25}$$

Assim, serão escolhidos os parâmetros $e_{ip}$ que maximizem $Var\,(Y_i)$ (ou *eigenvalues*, $\lambda_i$), segundo a restrição de a soma dos quadrados dos coeficientes $e_{ip}$ (ou instâncias do *eigenvector* $e_p$) seja igual a 1:

$$\mathbf{e}_1'\mathbf{e}_1 = \sum_{j=1}^{p} e_{1j}^2 = 1 \tag{3.26}$$

A primeira componente é a que apresenta a maior estimativa de variância explicada. As componentes subsequentes explicarão sequencialmente menos do que as anteriores e ainda contam com a restrição adicional:

$$\text{cov}(Y_{i-1}, Y_i) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{i-1,k}e_{il}\sigma_{kl} = \mathbf{e}_{i-1}'\Sigma\mathbf{e}_i = 0 \tag{3.27}$$

Deste modo, todas as componentes não são correlacionadas umas com as outras. Assim, é possível reter as primeiras $k$ componentes principais de forma a reduzir a dimensionalidade, sem perder muita informação. Para evitar ou minimizar estas perdas, é necessário que a proporção de variância explicada por essas $k$ componentes seja mais próxima de 1 possível:

$$\frac{\hat{\lambda}_1 + \hat{\lambda}_2 + \cdots + \hat{\lambda}_k}{\hat{\lambda}_1 + \hat{\lambda}_2 + \cdots + \hat{\lambda}_p} \cong 1 \tag{3.28}$$

Outra particularidade dos métodos de redução de dimensionalidade é que permitem identificar quais a variáveis mais associadas a cada componente, no entanto, devido ao seu carácter exploratório, devem ser acompanhados de outros métodos (testes de hipóteses, por exemplo) que os corroborem.

### Aplicação

Na página de *"Dimensionality Reduction"* é possível realizar análise de componentes principais (PCA), obter informações sobre os componentes como a variância explicada, desvios padrão e importância das variáveis em cada componente. A visualização dos coeficientes (mediante a seleção de duas componentes principais) apresenta-se num *biplot* interativo (ver Figura 3.24). Neste, os pontos são coloridos segundo a melhor partição em clusters, através de métodos análogos aos utilizados na secção de *clustering* da página *Explore*.

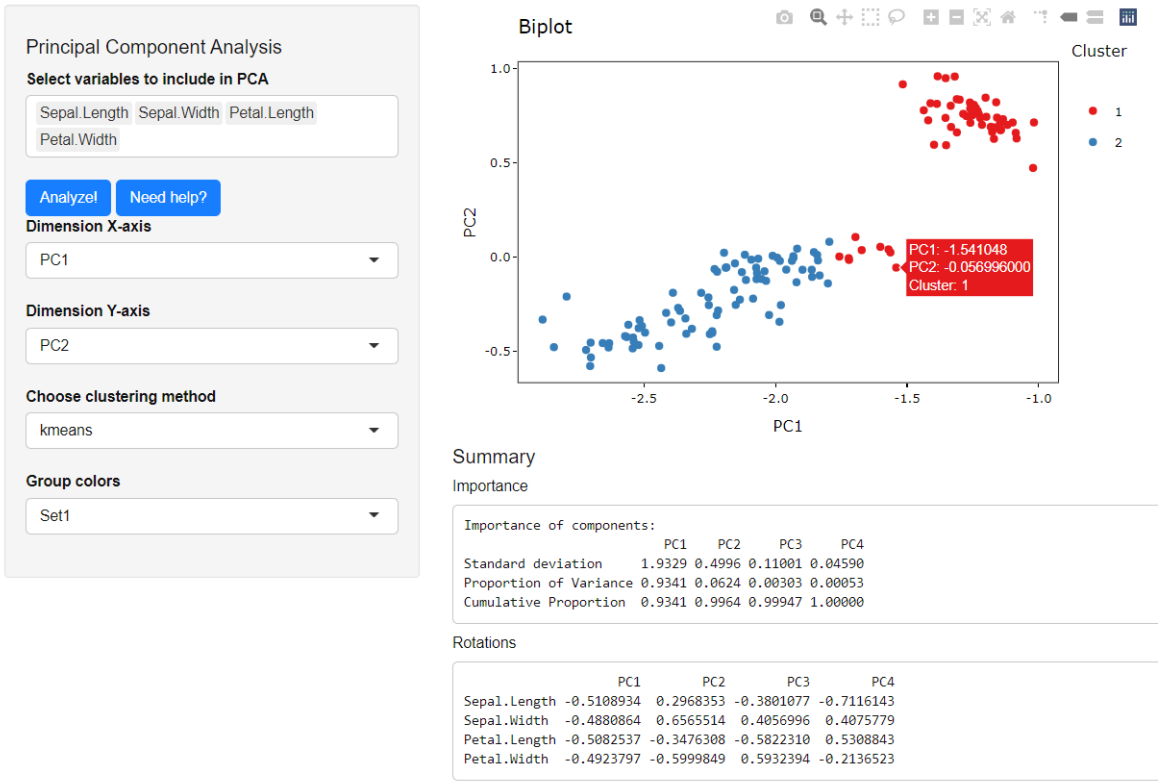


**Figura 3.24:** Página *Dimensionality Reduction*.

CAPÍTULO 4

# Conclusão

A aplicação `AnalyzeR` abrange vários métodos estatísticos amplamente utilizados tanto em meio académico, como também, crescentemente, no setor empresarial e industrial. O facto de o utilizador não necessitar de programar, aliado à intuitividade do UI, permite um maior foco nas análises a efetuar. A presente aplicação em Shiny, é passível de ser modificada e expandida no futuro, uma vez que as suas características de implementação permitem seccionar, modularizar e chamar scripts externos para a aplicação. No entanto, é considerado uma boa prática particionar grandes aplicações e/ou scripts de forma a reduzir a potencialidade de bugs ou erros bem como para facilitar futuras manutenções do código ou implementação em servidor (ou *cloud*). Não obstante, a aplicação que aqui se apresnta constitui, por si só, uma base sólida, sobre a qual poderão surgir melhoramentos e variantes, adaptadas às diferentes àreas que necessitam de análises de dados. Futuros desenvolvimentos poderão incluir novos tipos de regressão que aceitam variáveis dependentes do tipo categórico não binário como a Logística Ordinal e Multinomial. Outros métodos exploratórios também podem ser implemenmtados, nomeadamente na área da redução de dimensionalidade para variáveis categóricas (Análise de Correspondência (AC) e AC Múltipla). Uma versão demosntrativa do `AnalyzeR` pode ser experimentada em `http://danlobocastro.shinyapps.io/analyzer_final`.

# Bibliografia


[1] Winston Chang, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert and Barbara Borges (2021). *Shiny: Web Application Framework for R*. R package version 1.6.0. Available from `https://CRAN.R-project.org/package=shiny`.

[2] R version 4.1.1 (2021-08-10) – "Kick Things"Copyright (C) 2021 The R Foundation for Statistical Computing Platform: x86-64-w64-mingw32/x64 (64-bit).

[3] RStudio Team (2021). *RStudio: Integrated Development Environment for R*. RStudio, PBC, Boston, MA. Available from `http://www.rstudio.com/`.

[4] R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. Available from `http://www.R-project.org/`.

[5] "GNU R". Free Software Foundation (FSF) Free Software Directory. Retrieved 3 June 2021.

[6] A. Gunuganti. *Application Development Framework for R/Shiny* (2018). In PharmaSUG 2018 Conference Proceedings, volume AD-24(9), Seattle. PharmaSUG.

[7] Rubin DB. (1976).*Inference and missing data*. Biometrika. 63(3), 581–592. Available from `https://doi.org/10.1093/biomet/63.3.581`.

[8] Cox, D. R. and Oakes, D. (1984). *Analysis of Survival Data*. Chapman and Hall, London.

[9] Kaplan EL, Meier P. (1958) *Nonparametric Estimation from Incomplete Observations*. Journal of the American Statistical Association (53) 457-481.

[10] Pearson K. (1900) *On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling.*The London, Edinburgh and Dublin Phil Mag J Sci. Ser 5.

[11] Britannica, T. Editors of Encyclopaedia (2020, May 27). *Student's t-test*. Encyclopedia Britannica. Available from `https://www.britannica.com/science/Students-t-test`

[12] Kirch, Wilhelm.(2008) *Pearson's Correlation Coefficient*. Encyclopedia of Public Health 1090-1091, Springer Netherland. ISBN: 978-1-4020-5614-7. DOI: 10.1007/978-1-4020-5614-7_256. Available from `https://doi.org/10.1007/978-1-4020-5614-7_2569`.

[13] Thode, H.C. (2002). Testing For Normality (1st ed.). CRC Press. Available from `https://doi.org/10.1201/9780203910894`.

[14] Jolliffe Ian T, and Cadima Jorge. (2016) P*rincipal component analysis: a review and recent developments*. Phil. Trans. R. Soc. A.374. Available from `http://doi.org/10.1098/rsta.2015.0202t`

[15] Levene, H. (1960). *In Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, I. Olkin et al. eds., Stanford University Press, 278-292.

[16] Snedecor, George W. and Cochran, William G. (1989), *Statistical Methods*, Eighth Edition, Iowa State University Press.

[17] van Buuren, S., Groothuis-Oudshoorn, K. (2011). *mice: Multivariate Imputation by Chained Equations in R*. Journal of Statistical Software, 45(3), 1–67.

[18] Azur, M. J., Stuart, E. A., Frangakis, C., and Leaf, P. J. (2011). M*ultiple imputation by chained equations: what is it and how does it work?*. International journal of methods in psychiatric research, 20(1), 40–49. Available from `https://doi.org/10.1002/mpr.329`.

[19] Kuhn M (2021). *caret: Classification and Regression Training*. R package version 6.0-88. Available from `https://CRAN.R-project.org/package=caret`.

[20] Andri et mult. al. S (2021). *DescTools: Tools for Descriptive Statistics*. R package version 0.99.42, Available from `https://cran.r-project.org/package=DescTools`.

[21] Wickham H (2021). *tidyr: Tidy Messy Data*. R package version 1.1.4, Available from Available from `https://CRAN.R-project.org/package=tidyr`.

[22] Charrad M, Ghazzali N, Boiteau V, Niknafs A (2014). *NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set*. Journal of Statistical Software, *61*(6), 1-36. Available from `http://www.jstatsoft.org/v61/i06/`.

[23] Hadley Wickham and Jennifer Bryan (2019). *readxl: Read Excel File*s. R package version 1.3.1. Available from `https://CRAN.R-project.org/package=readxl`.

[24] Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B (2021). *shiny: Web Application Framework for R*. R package version 1.6.0, Available from `https://CRAN.R-project.org/package=shiny`.

[25] Chang W, Borges Ribeiro B (2018). *shinydashboard: Create Dashboards with 'Shiny'*. R package version 0,7,1. Available from `https://CRAN.R-project.org/package=shinydashboard`.

[26] Lilovski N (2021). *dashboardthemes: Customise the Appearance of 'shinydashboard' Applications using Themes*. R package version 1.1.5. Available from `https://CRAN.R-project.org/package=dashboardthemes`.

[27] van Buuren S, Groothuis-Oudshoorn K (2011). *mice: Multivariate Imputation by Chained Equations in R*. Journal of Statistical Software, 45(3), 1-67. Available from `https://www.jstatsoft.org/v45/i03/`.

[28] Xie Y, Cheng J, Tan X (2021). *DT: A Wrapper of the JavaScript Library 'DataTables'*. R package version 0.18. Available from `https://CRAN.R-project.org/package=DT`.

[29] Aragon T (2020). *epitools: Epidemiology Tools*. R package version 0.5-10.1, Available from `https://CRAN.R-project.org/package=epitools`.

[30] Therneau T (2021). *A Package for Survival Analysis in R*. R package version 3.2-11, Available from `https://CRAN.R-project.org/package=survival`.

[31] Tierney N, Cook D, McBain M, Fay C (2021). n*aniar: Data Structures, Summaries, and Visualisations for Missing Data*. R package version 0.6.1, Available from `https://CRAN.R-project.org/package=naniar`.

[32] Wickham H, François R, Henry L, Müller K (2021). d*plyr: A Grammar of Data Manipulation*. R package version 1.0.7. Available from `https://CRAN.R-project.org/package=dplyr`.

[33] Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4, `https://ggplot2.tidyverse.org`.

[34] Ryu C (2021). *dlookr: Tools for Data Diagnosis, Exploration*, Transformation. R package version 0.5.0. Available from `https://CRAN.R-project.org/package=dlookr`.

[35] Allaire J, Horner J, Xie Y, Marti V, Porte N (2019). *markdown: Render Markdown with the C Library 'Sundown'*. R package version 1.1. Available from `https://CRAN.R-project.org/package=markdown`.

[36] Sievert C (2020). *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC. ISBN 9781138331457. Available from `https://plotly-r.com`.

[37] Tang Y, Horikoshi M, Li W (2016). *ggfortify: Unified Interface to Visualize Statistical Result of Popular R Packages*. The R Journal, 8(2), 474-485. Available from `https://doi.org/10.32614/RJ-2016-060`.

[38] Kassambara A, Kosinski M, Biecek P (2021). *survminer: Drawing Survival Curves using 'ggplot2'*. R package version 0.4.9. Available from `https://CRAN.R-project.org/package=survminer`.

[39] Revelle W (2021). *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois. R package version 2.1.6. Available from `https://CRAN.R-project.org/package=psych`.

[40] Schloerke B, Cook D, Larmarange J, Briatte F, Marbach M, Thoen E, Elberg A, Crowley J (2021). *GGally: Extension to 'ggplot2'*. R package version 2.1.2. Available from `https://CRAN.R-project.org/package=GGally`.

[41] Kassambara A, Mundt F (2020). f*actoextra: Extract and Visualize the Results of Multivariate Data Analyses*. R package version 1.0.7. Available from `https://CRAN.R-project.org/package=factoextra`.

[42] Lê S, Josse J, Husson F (2008). F*actoMineR: A Package for Multivariate Analysis*. Journal of Statistical Software, 25(1), 1-18. Available from `https://doi.org/10.18637/jss.v025.i01`.

[43] Comtois D (2021). *summarytools: Tools to Quickly and Neatly Summarize Data*. R package version 1.0.0, URL: https://CRAN.R-project.org/package=summarytools.

[44] Henry L, Wickham H (2020). *purrr: Functional Programming Tools*. R package version 0.3.4. Available from `https://CRAN.R-project.org/package=purrr`.

[45] Schauberger P, Walker A (2021). *openxlsx: Read, Write and Edit xlsx Files*. R package version 4.2.4. Available from `https://CRAN.R-project.org/package=openxlsx`.

[46] Arnholt A, Evans B (2017). *BSDA: Basic Statistics and Data Analysis*. R package version 1.2.0. Available from `https://CRAN.R-project.org/package=BSDA`.

[47] Rudis B (2020). *hrbrthemes: Additional Themes, Theme Components and Utilities for 'ggplot2'*. R package version 0.8.0. Available from `https://CRAN.R-project.org/package=hrbrthemes`.

[48] Lüdecke D, Ben-Shachar M, Patil I, Waggoner P, Makowski D (2021). *performance: An R Package for Assessment, Comparison and Testing of Statistical Models*. Journal of Open Source Software, 6(60), 3139. Available from `10.21105/joss.03139`.

[49] Iannone R, Cheng J, Schloerke B (2021). *gt: Easily Create Presentation-Ready Display Tables*. R package version 0.3.1. Available from `https://CRAN.R-project.org/package=gt`.

[50] Robinson D, Hayes A, Couch S (2021). *broom: Convert Statistical Objects into Tidy Tibbles* R package version 0.7.9. Available from `https://CRAN.R-project.org/package=broom`.

[51] Sjoberg D, Curry M, Hannum M, Larmarange J, Whiting K, Zabor E (2021). *gtsummary: Presentation-Ready Data Summary and Analytic Result Tables*. R package version 1.4.2. Available from `https://CRAN.R-project.org/package=gtsummary`.

[52] Iannone R (2021). f*ontawesome: Easily Work with 'Font Awesome' Icons*. R package version 0.2.2. Available from `https://CRAN.R-project.org/package=fontawesome`.

[53] Lüdecke D (2018). g*geffects: Tidy Data Frames of Marginal Effects from Regression Models*. Journal of Open Source Software, 3(26), 772. Available from `10.21105/joss.00772`.

[54] Attali D, Edwards T (2020). s*hinyalert: Easily Create Pretty Popup Messages (Modals) in 'Shiny'*. R package version 2.0.0. Available from `https://CRAN.R-project.org/package=shinyalert`.

[55] Lüdecke D, Patil I, Ben-Shachar M, Wiernik B, Waggoner P, Makowski D (2021). *see: An R Package for Visualizing Statistical Models*. Journal of Open Source Software, 6(64), 3393. Available from `10.21105/joss.03393`.

[56] Tanaka E, Niichan (2018). *shinycustomloader: Custom Loader for Shiny Outputs*. R package version 0.9.0. Available from `https://CRAN.R-project.org/package=shinycustomloader`.

[57] Garnier, Simon, Ross, Noam, Rudis, Robert, Camargo, Pedro A, Sciaini, Marco, Scherer, Cédric (2021). *viridis - Colorblind-Friendly Color Maps for R*. R package version 0.6.1. Available from `https://sjmgarnier.github.io/viridis/`.

[58] Galili, Tal, O'Callaghan, Alan, Sidi, Jonathan, Sievert, Carson (2017). *heatmaply: an R package for creating interactive cluster heatmaps for online publishing*. Bioinformatics. Available from `10.1093/bioinformatics/btx657`.

[59] Kuhn M, Jackson S, Cimentada J (2020). *corrr: Correlations in* R. R package version 0.4.3. Available from `https://CRAN.R-project.org/package=corrr`.

[60] Lüdecke D (2021). *sjPlot: Data Visualization for Statistics in Social Science*. R package version 2.8.9. Available from `https://CRAN.R-project.org/package=sjPlot`.

[61] Pedersen T (2021). *ggforce: Accelerating 'ggplot2'*. R package version 0.3.3. Available from `https://CRAN.R-project.org/package=ggforce`.

[62] Cui B (2020). *DataExplorer: Automate Data Exploration and Treatment*. R package version 0.8.2. Available from `https://CRAN.R-project.org/package=DataExplorer`.

[63] Xie Y (2021). *knitr: A General-Purpose Package for Dynamic Report Generation in R*. R package version 1.33. Available from `https://yihui.org/knitr/`.

[64] Perrier V, Meyer F (2020). f*resh: Create Custom 'Bootstrap' Themes to Use in 'Shiny'*. R package version 0.2.0. Available from `https://CRAN.R-project.org/package=fresh`.

[65] Littlefield T (2020). *vov: CSS Animations for 'shiny' Elements*. R package version 0.1.2. Available from `https://CRAN.R-project.org/package=vov`.

[66] Cameron, A. C. and Trivedi, P. K. (1998). *Regression Analysis of Count Data*. New York: Cambridge Press.

[67] Berry, W. D. & Feldman, S. (1985). *The multiple regression model: a review*. In Multiple regression in practice (pp. 10-18). SAGE Publications, Inc. Available from `https://www.doi.org/10.4135/9781412985208`.

[68] Bewick, V., Cheek, L., & Ball, J. (2005). S*tatistics review 14: Logistic regression*. Critical care (London, England), 9(1), 112–118. Available from `https://doi.org/10.1186/cc3045`.

[69] Fox, J. (1997) A*pplied regression analysis, linear models, and related methods*. Thousand Oaks, CA: Sage Publications.

[70] Austin PC, Steyerberg EW. (2014) *Graphical assessment of internal and external calibration of logistic regression models by using loess smoothers*. Statistical Medicine.

[71] R. A. Fisher (1936). *The use of multiple measurements in taxonomic problems*. Annals of Eugenics. 7:179–188. Available from `https://doi.org/10.1111/j.1469-1809.1936.tb02137.x`.

[72] Miriam Seoane Santos, Pedro Henriques Abreu, Pedro J Garcia-Laencina, Adelia Simao, Armando Carvalho (2015). *A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients*. Journal of biomedical informatics. 58, 49-59.