UNIVERSIDADE FEDERAL DE MATO GROSSO
INSTITUTO DE CIÊNCIAS EXATAS E DA TERRA
BACHARELADO EM ESTATÍSTICA

MATHEUS NUNES PAIVA

AUTOMATIZANDO A ANÁLISE DE REGRESSÃO LINEAR NO RSTUDIO: UM APLICATIVO INTERATIVO EM *SHINY*

CUIABÁ - MT
2024

MATHEUS NUNES PAIVA

AUTOMATIZANDO A ANÁLISE DE REGRESSÃO LINEAR NO RSTUDIO: UM APLICATIVO INTERATIVO EM *SHINY*

Trabalho de conclusão de curso apresentada à banca examinadora do Curso de Bacharelado em Estatístisca da Universidade Federal de Mato Grosso, como requisito parcial, para obtenção do título de Bacharel em Estatística

Orientadora: Profa. Dra. Eveliny Barroso da Silva

CUIABÁ - MT
2024

Dados Internacionais de Catalogação na Fonte.

P149a Paiva, Matheus Nunes.
Automatizando a análise de regressão linear no RStudio: um aplicativo interativo em Shiny [recurso eletrônico] / Matheus Nunes Paiva. -- Dados eletrônicos (1 arquivo : 45 f., il. color., pdf). -- 2024.

Orientadora: Eveliny Barroso Silva.
TCC (graduação em Estatística) - Universidade Federal de Mato Grosso, Instituto de Ciências Exatas e da Terra, Cuiabá, 2024.
Modo de acesso: World Wide Web: https://bdm.ufmt.br.
Inclui bibliografia.

1. análise de regressão linear. 2. seleção de modelos. 3. linguagem R. 4. RStudio. 5. pacote Shiny. I. Silva, Eveliny Barroso, *orientador*. II. Título.

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

MATHEUS NUNES PAIVA

AUTOMATIZANDO A ANÁLISE DE REGRESSÃO LINEAR NO RSTUDIO: UM APLICATIVO INTERATIVO EM *SHINY*

Trabalho de conclusão de curso apresentada à banca examinadora do Curso de Bacharelado em Estatístisca da Universidade Federal de Mato Grosso, como requisito parcial, para obtenção do título de Bacharel em Estatística

BANCA EXAMINADORA

Profa. Dra. Eveliny Barroso da Silva _____
Orientadora
Prof. Dr. José Nilton da Cruz _____
Examinador
Prof. Dr. Marcelino Alves Rosa de Pascoa _____
Examinador

CUIABÁ - MT
2024

RESUMO

Este trabalho propõe o desenvolvimento de um aplicativo interativo em *Shiny* via linguagem R para automatizar a análise de regressão linear, uma técnica estatística amplamente utilizada em diversas áreas do conhecimento. A aplicação da análise de regressão linear pode ser complexa e requer conhecimento em estatística e programação, o que pode ser um obstáculo para muitos usuários. Além disso, pode ser uma tarefa demorada, especialmente ao lidar com grandes conjuntos de dados e múltiplas variáveis. O aplicativo proposto visa superar esses desafios, permitindo que os usuários realizem análises de regressão linear de maneira intuitiva e sem a necessidade de codificação. Com este aplicativo, os usuários poderão inserir seu próprio banco de dados, selecionar as variáveis independentes e a variável dependente, e realizar uma análise de regressão linear. Isso torna a técnica estatística mais acessível e fácil de utilizar, economizando tempo e esforço dos usuários e permitindo que eles se concentrem na interpretação dos resultados em vez da implementação do modelo. Para o teste do aplicativo, será utilizado um banco de dados de consumo de energia da empresa *DAEWOO Steel Co*., Ltd, empresa Sul Coreana (UCI, 2023). A variável resposta de interesse é o consumo de energia da indústria e as covariáveis são dióxido de carbono, potência relativa da corrente e dias da semana. O aplicativo construído permite a inserção de base de dados em Excel, apresentação dos dados em formato de tabela, análise descritiva dos dados, análise de regressão linear simples e múltipla, seleção de modelos e análise de diagnósticos.
Palavras-chave: análise de regressão linear; seleção de modelos; linguagem R; RStudio; pacote *Shiny*.

ABSTRACT

This work proposes the development of an interactive application in R language to automate linear regression analysis, a statistical technique widely used in various areas of knowledge. The application of linear regression analysis can be complex and requires knowledge in statistics and programming, which can be an obstacle for many users. In addition, it can be a time-consuming task, especially when dealing with large data sets and multiple variables. The proposed application aims to overcome these challenges, allowing users to perform linear regression analyses in an intuitive way and without the need for coding. With this application, users will be able to enter their own database, select the independent variables and the dependent variable, and perform a linear regression analysis. This makes the statistical technique more accessible and easy to use, saving users' time and effort and allowing them to focus on interpreting the results instead of implementing the model. For the application test, a database of energy consumption from the company DAEWOO Steel Co., Ltd, a South Korean company (UCI, 2023), will be used. The response variable of interest is the industry's energy consumption and the covariates are carbon dioxide, relative current power and days of the week. The built application allows the insertion of a database in Excel, presentation of the data in table format, descriptive analysis of the data, simple and multiple linear regression analysis, graphical analysis of the residuals and model selection.
Keywords: linear regression analysis; model Selection; R language; RStudio; *Shiny* package.

LISTA DE ABREVIATURAS

ANOVA - *Analysis of Variance*, em português, Análise de Variância.

kVarh - quilovolt-ampere reativo-hora.

kWh - quilowatt-hora.

MELNV - Melhores Estimadores Lineares Não-viesados.

MQO - Minímos Quadrados Ordinários.

RLM - Regressão Linear Múltipla.

RLS - Regressão Linear Simples.

VIF - *Variance Inflation Factor*, em português, Fator de Inflação da Variância.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO

A análise de regressão linear é uma ferramenta de análise estatística fundamental que permite aos pesquisadores e profissionais de diversas áreas do conhecimento explorar e quantificar as relações entre as variáveis. Utilizada em diversas áreas, tais como economia, biologia, engenharia, ciências sociais, etc, essa técnica é essencial para a modelagem de fenômenos complexos e a tomada de decisões baseada em dados. No entanto, a sua aplicação prática necessita de conhecimento estatístico e de programação. O domínio de conceitos estatísticos, juntamente com a habilidade de implementar esses métodos em um *software* de programação, como por exemplo o R (R *Core Team*, 2024), pode ser um desafio para muitos (Alooba, 2024).

Na tentativa de viabilizar o uso da ferramenta de análise de regressão no R (R *Core Team*, 2024), que é um *software* livre, pensou-se em desenvolver um aplicativo de fácil acesso a essa ferramenta estatística. Dentre a infinidade de funcionalidades do R, há o *R Shiny*. O *R Shiny* é uma ferramenta de desenvolvimento *web* que permite a criação de aplicativos interativos diretamente em R (R *Core Team*, 2024). Com o *R Shiny*, os usuários podem transformar análises de dados complexas em aplicativos *web* interativos sem a necessidade de conhecimento em linguagens de programação *web*, como HTML, CSS e *JavaScript* (Escola, 2024). O *R Shiny* funciona por meio de uma combinação de código R e código HTML. O código R é utilizado para realizar as análises de dados e gerar gráficos, tabelas e outros elementos interativos, enquanto o código HTML é utilizado para criar a interface do aplicativo *web* (Escola, 2024). Alguns dos principais recursos do *R Shiny* incluem a capacidade de criar *dashboards* interativos, a integração com bibliotecas de gráficos como *ggplot2* (Wickham, 2016) e *plotly* (Sievert, 2020), e a capacidade de compartilhar aplicativos *web* por meio da nuvem.

Encontrou-se na literatura diversos trabalhos que tiveram motivação similar tais como: Saavedra e Lobos (2018), que construiu um aplicativo em *Shiny* para o ajuste de modelos lineares generalizados. Konrath et al. (2019) que desenvolveu um aplicativo *Shiny* para a construção de modelos de previsão com séries temporais. Os autores Konrath et al. (2018) abordaram as principais funcionalidades do pacote *Shiny* para o ensino de estatística. Em Miranda (2018) desenvolveu-se a construção de um aplicativo em *Shiny* para análise exploratória de dados. Na linha de análise de regressão linear múltipla, Figueiredo (2019) desenvolveu um aplicativo em *Shiny* para a análise de regressão múltipla.

Assim, este trabalho de conclusão de curso visa atender estudantes, pesquisadores e entusiastas que, apesar do interesse no assunto, não detêm um conhecimento avançado em programação. Apesar de Figueiredo (2019) ter desenvolvido um aplicativo em *Shiny* para a análise de regressão múltipla, o presente trabalho desenvolveu um aplicativo interativo utilizando o pacote *Shiny* (Chang et al., 2024) no ambiente R Studio, que facilita o acesso à análise de regressão linear simples e múltipla. Este aplicativo oferece aos usuários uma interface intuitiva que permite a realização de análises complexas sem a necessidade de habilidades técnicas especializadas, oferecendo as análises descritivas dos dados bem como as análises de regressão linear simples ou múltipla, de forma intuitiva e automatizada. Além das análises de regressão, também está implementado técnicas de seleção de modelos e análise de diagnósticos do modelo.

Embora a análise de regressão linear seja amplamente reconhecida por sua utilidade, sua adoção é muitas vezes limitada pela complexidade das ferramentas estatísticas e de programação. Ao automatizar o processo de ajuste de modelos de regressão linear, o aplicativo proposto não apenas torna a técnica mais acessível, como também economiza tempo do usuário facilitando a seleção de variáveis de grandes conjuntos de dados.

1.1 Justificativa

O ajuste de um modelo de regressão pode ser complexo e exigir um certo conhecimento de estatística e de programação. Estes fatores podem ser obstáculos para muitos usuários que poderiam se beneficiar da análise de regressão linear mas não têm as habilidades necessárias para implementá-la. Por esta razão, o aplicativo proposto neste trabalho, visa facilitar a análise de regressão linear iniciando com a estatística descritiva das variáveis, ajuste de modelos, seleção automática de variáveis e validação do modelo selecionado. Esta validação ocorrerá por meio de técnicas de análise de diagnósticos dos modelos de regressão.

2 REFERENCIAL TEÓRICO

2.1 Correlação linear

É comum na prática, o interesse em se analisar o comportamento conjunto de duas ou mais variáveis quantitativas. Suponha que seja de interesse obter uma medida estatística que indique se existe ou não relação entre duas variáveis e se existe relação qual a sua magnitude e sinal. O coeficiente estatístico que mede o grau dessa correlação, ou ausência dela, é o Coeficiente de Correlação linear de Pearson (Charnet et al., 2008).

Para o cálculo do coeficiente de correlação é necessário calcular a covariância entre as duas variáveis quantitativas. A covariância é uma estatística que mede o grau de associação linear entre duas variáveis quantitativas (Charnet et al., 2008). O coeficiente de correlação amostral pode ser calculado pela expressão abaixo:

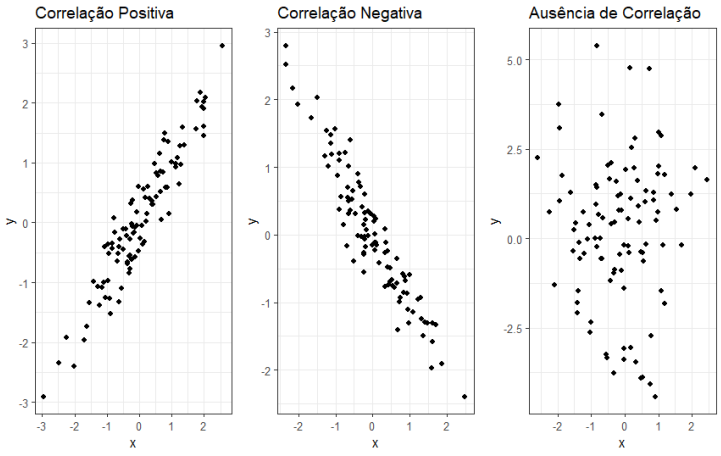
$$r_{xy} = \frac{\mathrm{Cov}(x,y)}{S_x S_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

em que:

- $\mathrm{Cov}(x,y)$ é a covariância entre as variáveis X e Y;
- S_x e S_y, são os desvios-padrão para as variáveis X e Y, respectivamente;
- $x_i = x_1, x_2, \ldots, x_n$, são os valores observados da variável X;
- $y_i = y_1, y_2, \ldots, y_n$, são valores observados da variável Y;
- $\bar{x}$ e $\bar{y}$, são as médias de X e Y, respectivamente.

O cálculo do coeficiente de correlação linear resulta em um valor entre -1 e 1 no qual valores próximos de -1 representam uma correlação linear fortemente negativa e valores próximo a 1 representam uma correlação linear fortemente positiva. Para valores próximos a 0 tem-se a ausência de correlação linear. A Figura 1 apresenta um exemplo para cada tipo de correlação.

Figura 1 – Exemplos de correlação.



Fonte: Dados Simulados no R.

2.2 Regressão linear

A palavra "Regressão" foi ultilizada pela primeira vez por Galton (1886) em seus estudos sobre hereditariedade. Galton notou que pais de estatura alta tem filhos de alta estatura, mas não tão alta, em média, como os pais, e pais de baixa estatura tem filhos de baixa estatura, mas não tão baixa, em média como os pais. Esta tendência da média da geração seguinte tender à média na geração anterior ele denominou de regressão (Dachs, 1978).

Em resumo, a análise de regressão tem como objetivo principal estudar a influência que determinadas variáveis exercem sobre outras, estabelecendo uma relação linear entre elas. Quando existe apenas uma variável explicativa influenciando uma variável dependente, tem-se a regressão linear simples. Quando mais de uma variável explicativa influencia a variável dependente, tem-se a regressão linear múltipla.

2.3 Regressão Linear Simples

Seja Y a variável dependente (de interesse) e X a variável independente. A média ou a resposta média de Y varia com a variável X da seguinte forma:

$$I\!E(Y|X) = f_X(x).$$

Supondo que Y se relaciona linearmente com X, tem-se:

$$I\!E(Y|X) = \beta_0 + \beta_1 X, \tag{2.1}$$

em que β_0 e β_1 são parâmetros a serem estimados. A equação (2.1) é denominada de Função de Regressão Populacional (Gujarati, 2011).

Na prática dificilmente obtém-se dados populacionais para as variáveis X e Y mas se tivermos um amostra de n pares de valores para as duas variáveis, X e Y, sob a suposição de que Y é uma função linear de X, a equação de regressão linear simples é definida por :

$$\mathrm{y_i} = \beta_0 + \beta_1 \mathrm{x_i} + \epsilon_i, \quad i = 1, \ldots, n, \tag{2.2}$$

em que:

- β_0 e β_1 são parâmetros desconhecidos a serem estimados;
- $x_i = x_1, x_2, \ldots, x_n$ são valores observados da variável independente X;
- $y_i = y_1, y_2, \ldots, y_n$ são valores observados da variável dependente Y;
- ϵ_i é o erro aleatório não observável.

2.3.1 Estimação dos parâmetros da RLS

Para estimar os parâmetros do modelo de regressão linear simples (RLS), equação (2.2), é utilizado o método dos mínimos quadrados. Este método consiste em adotar os estimadores que minimizam a soma dos quadrados dos desvios entre valores estimados e valores observados na amostra (Hoffmann, 2016).

O método dos mínimos quadrados consiste em minimizar a soma de quadrados dos erros definidos por:

$$\sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_i)]^2,$$

em que $\epsilon_i = y_i - (\beta_0 + \beta_1 x_i), \quad i = 1, \ldots, n,$. Para minimizar a soma de quadrados dos erros é necessário obter as seguintes derivadas parciais:

$$\frac{\partial}{\partial \beta_0} \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_i) \right]^2, \tag{2.3}$$

$$\frac{\partial}{\partial \beta_1} \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_i) \right]^2, \tag{2.4}$$

a partir das equações (2.3) e (2.4) tem-se as seguintes equações normais:

$$\begin{cases} \sum_{i=1}^{n} y_i - n\hat{\beta}_0 - \hat{\beta}_1 \sum_{i=1}^{n} x_i = 0, \\ \sum_{i=1}^{n} y_i - \hat{\beta}_0 \sum_{i=1}^{n} x_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 = 0, \end{cases} \tag{2.5}$$

Resolvendo o sistema de equações normais (2.5) tem-se:

$$\hat{\beta}_0 = \bar{y} - \beta_1 \bar{x},$$

$$\hat{\beta}_1 = \frac{\sum y_i x_i - n\bar{y}\bar{x}}{\sum x_i^2 - n\bar{x}^2}.$$

Em que:

- $\hat{\beta}_0$: intercepto do modelo ajustado;
- $\hat{\beta}_1$: coeficiente angular do modelo ajustado.

Dado um valor x da covariável X, o modelo de regressão linear ajustado, a reta ajustada ou a Função de Regressão Estimada é dada por:

$$I\!E(\hat{Y}|X = x) = \hat{\beta}_0 + \hat{\beta}_1 x.$$

Ou seja, o valor predito da variável resposta Y para a i-ésima observação dado o valor $x_i, i = 1, 2, \ldots, n$, da variável explicativa X é dado por:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i, \quad i = 1, 2, \ldots, n.$$

A interpretação das estimativas dos parâmetros do modelo ajustado é como segue:

- Intercepto ($\hat{\beta}_0$): é o valor esperado para Y quando $X = 0$.
- Coeficiente angular ($\hat{\beta}_1$): representa o quanto varia a média de Y para cada aumento de uma unidade da variável X.

Em muitas situações, usualmente tem-se interesse em verificar se mais de uma variável explicativa influencia a variável dependente. Este cenário configura uma análise de regressão linear múltipla que será apresentado a seguir.

2.4 Regressão Linear Múltipla

Sob a suposição de que a variável dependente é uma função linear de várias variáveis independentes, tem-se uma regressão linear múltipla. O modelo de regressão linear múltipla (RLM) com k variáveis independentes pode ser definido por (Hoffmann, 2016):

$$\mathbf{Y} = \boldsymbol{\beta}\mathbf{X} + \boldsymbol{\epsilon},$$

em que:

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix},$$

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_j \end{bmatrix}, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

Em que:

- $\mathbf{Y}$ é um vetor coluna de observações da variável resposta de tamanho $n \times 1$;
- $\mathbf{X}$ é a matriz de variáveis explanatórias X_{nk}, $k = 1, \cdots, j$ e uma coluna de uns;
- o número de colunas de $\mathbf{X}$ é igual ao número de elementos em $\boldsymbol{\beta}$ e o número de linhas de $\mathbf{X}$ é o tamanho da amostra;
- $\boldsymbol{\beta}$ é o vetor coluna $(j+1) \times 1$ de coeficientes a serem estimados;
- $\boldsymbol{\epsilon}$ é o vetor coluna $n \times 1$ de erros aleatórios.

O método dos mínimos quadrados é utilizado para estimar o vetor de parâmetros $\boldsymbol{\beta}$, minimizando a soma dos quadrados dos erros $\boldsymbol{\epsilon}$ ($SQ(\boldsymbol{\epsilon})$), que em forma matricial é dada por (Hoffmann, 2016):

$$SQ(\boldsymbol{\epsilon}) = \boldsymbol{\epsilon}'\boldsymbol{\epsilon} = \left(\mathbf{Y}' - \boldsymbol{\beta}'\mathbf{X}'\right)(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{Y}'\mathbf{Y} - \mathbf{Y}'\mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}'\mathbf{X}'\mathbf{Y} + \boldsymbol{\beta}'\mathbf{X}'\mathbf{X}\boldsymbol{\beta},$$

Como as matrizes $\mathbf{Y}'\mathbf{X}\boldsymbol{\beta}$ e $\boldsymbol{\beta}'\mathbf{X}'\mathbf{Y}$ são iguais, simplifica-se para:

$$SQ(\boldsymbol{\epsilon}) = \mathbf{Y}'\mathbf{Y} - 2\boldsymbol{\beta}'\mathbf{X}'\mathbf{Y} + \boldsymbol{\beta}'\mathbf{X}'\mathbf{X}\boldsymbol{\beta}, \tag{2.6}$$

Derivando a equação (2.6) em relação ao parâmetro $\boldsymbol{\beta}$ temos:

$$\frac{\partial SQ(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -2\boldsymbol{X}'\boldsymbol{Y} + 2\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta}, \tag{2.7}$$

Igualando a derivada da expressão (2.7) a 0, obtém-se as seguintes equações normais de mínimos quadrados na forma matricial:

$$(\boldsymbol{X}'\boldsymbol{X})\hat{\boldsymbol{\beta}} = \boldsymbol{X}'\boldsymbol{Y},$$
$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}.$$

Realizado o ajuste do modelo é necessário verificar, dentro de um conjunto de k covariáveis, quais que de fato constituem um melhor cenário para predizer a variável resposta Y. As técnicas utilizadas são as de seleção de modelos. Na literatura é possível encontrar diversas técnicas que auxiliem nessa tomada de decisão. Charnet et al. (2008) apresentam algumas delas, por exemplo:

- Método “passo a frente” (*forward*);
- Método “passo a atrás” (*backward*);
- Método “passo a passo” (*stepwise*);

A técnica utilizada neste estudo é de *stepwise* que será apresentada brevemente na próxima seção.

2.5 Seleção de variáveis via *Stepwise*

Dentro do grupo de variáveis independentes, algumas podem ter uma influência mínima na variável resposta Y. O método *stepwise* é aplicado para identificar as variáveis que têm o maior impacto na predição de Y. Este método pode reduzir o número de variáveis necessárias para formar a equação de regressão. O método *stepwise* é uma generalização do método "passo a frente" (*forward*), em que após cada etapa de inclusão de uma variável, tem-se uma etapa onde uma das variáveis já selecionadas pode ser retirada "passo atrás"(*backward*) (Charnet et al., 2008). A inserção e a exclusão de uma variável é feita segundo algum critério, o critério utilizado neste estudo é o Critério de Informação *Akaike* (*Akaike information criterion* ou AIC). O AIC pode ser obtido pela seguinte expressão:

$$AIC = -2\ln(\hat{L}) + 2p,$$

em que:

- $\hat{L}$ é o valor máximo estimado da função de log-verossimilhança para o modelo.
- p é o número de parâmetros do modelo.

Akaike (1974) propôs o uso da informação ou distância de Kullback–Leibler (K-L) como base fundamental para a seleção de modelos. Akaike (1974) encontrou uma maneira de estimar a informação de K-L baseado na função log-verossimilhança em seu ponto de máximo. "O AIC é uma medida da qualidade de ajuste de um modelo estatístico estimado" (Emiliano, 2009). Na prática selecionamos o modelo com o menor valor de AIC (Burnham e Anderson, 2002).

2.6 Diagnóstico do modelo e qualidade do ajuste

A qualidade do ajuste em um modelo de regressão refere-se ao nível de acoplamento entre as observações reais e os valores previstos pelo modelo. Em outras palavras, quanto melhor o ajuste do modelo, mais bem ele explica os dados estudados.

2.6.1 O teorema de Gauss-Markov

O teorema de Gauss-Markov postula que, quando a distribuição de probabilidade do erro é desconhecida em um modelo linear, o estimador obtido usando o método dos mínimos quadrados é aquele que minimiza a variância entre todos os estimadores lineares não tendenciosos para os parâmetros do modelo linear. Em resumo, sob certas codições os estimadores de mínimos quadrados são os melhores estimadores lineares não viesados. Gauss (1823) provou seu teorema em seu trabalho chamado "*Theoria Combinationis Observationum Erroribus Minimis Obnoxiae*", Markov generalizou seu teorema em 1900 e Graybill (1976) fez uma notação moderna para o teorema (Dodge, 2008).

A seguir estão apresentadas as 5 condições para um bom ajuste de mínimos quadrados segundo o Teorema de Gauss-Markov, (Wooldridge, 2015):

1. **Linearidade dos parâmetros**

   Se o modelo pode ser escrito como: $\mathbf{Y} = \boldsymbol{\beta}\mathbf{X}+\boldsymbol{\epsilon}$, sendo $\boldsymbol{\beta} = (\beta_0, \beta_1, ..., \beta_j)$ um vetor de parâmetros e $\boldsymbol{\epsilon}$ um vetor de erros aleatórios, então podemos dizer que o modelo é linear nos parâmetros.

2. **Ausência de colinearidade entre os parâmetros**

   Na amostra aleatória nenhuma variável independente deve ser constante e, portanto, não deverá existir relação linear entre as variáveis independentes.

3. **Esperança dos erro igual a 0**

   A esperança de $\boldsymbol{\epsilon}$ deve ser igual a 0. De maneira simples: $\mathbb{E}[\boldsymbol{\epsilon}|\mathbf{X}] = 0$

4. **Homocedasticidade**

   A variância de $\boldsymbol{\epsilon}$ é constante dados quaisquer variável independente. De maneira simples:

   $$Var[\boldsymbol{\epsilon}|\mathbf{X}] = \sigma^2$$

5. **Erros não correlacionados**

   Dados quaisquer par de valores de $\mathbf{X}$ a correlação entre esses dois valores deverá ser igual a 0. De maneira simples:

   $$cov(\epsilon_i, \epsilon_m|X_i, X_m) = 0,$$

   i e m são duas observações diferentes.

2.6.2 Coeficiente de determinação R^2

O coeficiente de determinação R^2 mede a proporção da variância dos dados que é explicada por um modelo estatístico. A soma de quadrados total de $\mathbf{Y}$, (SQT), mensura a variabilidade das observações em torno de sua própria média, enquanto que a soma de quadrados dos resíduos, $(SQRes)$, mensura a qualidade do ajuste do modelo ajustado, quanto menor for a soma de quadrados dos resíduos melhor é o ajuste do modelo. O coeficiente de determinação é definido como:

$$R^2 = \frac{SQReg}{SQT}.$$

O coeficiente de variação assume valores no intervalo [0,1], valores próximos a 1 indicam um melhor ajuste do modelo e valores próximos a 0 indicam que o modelo é ruim, de modo resumido o R^2 é a proporção da variabilidade dos valores de Y considerando o modelo ajustado. O cálculo das somas de quadrado estão apresentadas na Tabela 1 (Charnet et al., 2008).

2.6.3 Análise de variância e estatística F

A estatística F indica se o modelo fornece um bom ajuste aos dados, ela é calculada através da decomposição da soma de quadrados do modelo de regressão e dos resíduos Tabela 1. Em resumo ela testa se o modelo ajustado é melhor do que o modelo nulo (modelo sem variáveis independentes), se a valor de probabilidade da estatística F for inferior ao nível de significância α se assume que há evidências o suficiente para rejeitar a hipótese nula, caso contrário a estatística F irá indicar que modelo não se ajusta bem aos dados (Sureiman e Mangera, 2020). As hipóteses para a realização do teste F são definidas por:

$$H_0 : \mathbf{Y} = \beta_0 + \boldsymbol{\epsilon};$$
$$H_1 : \mathbf{Y} = \beta_0 + \beta_1 x_1 + \ldots + \beta_j x_j + \boldsymbol{\epsilon}.$$

A análise de variância (ANOVA) baseia-se na decomposição da variabilidade total, o cálculo da estatística F está apresentada na Tabela 1. As demonstrações da decomposição da variância não estão no escopo deste trabalho; portanto, apenas o resultado será apresentado

Tabela 1 – Análise de Variância.

Fonte de variação	GL	SQ	QM	F
Regressão	$p-1$	$\boldsymbol{\beta}'\mathbf{X}'\mathbf{Y} - \frac{(\sum y_i)^2}{n}$	$\frac{SQReg}{p-1}$	$\frac{QMReg}{QMRes}$
Resíduo	$n-p$	$\mathbf{Y}'\mathbf{Y} - \boldsymbol{\beta}'\mathbf{X}'\mathbf{Y}$	$\frac{SQRes}{n-p}$	
Total	$n-1$	$\mathbf{Y}'\mathbf{Y} - \frac{(\sum y_i)^2}{n}$		

GL : Graus de liberdade.
SQ : Soma de quadrados.
QM : Quadrado médio.
p : Número de parâmetros do modelo.
F : Valor-F calculado.
$i = 1, 2, 3, ..., n$.

Apesar de não fazer parte das hipóteses básicas descritas no Teorema de Gauss-Markov, uma hipótese importante a ser investigada em RLM é a existência de multicolinearidade entre as variáveis. Este tópico é desenvolvido na subseção a seguir.

2.6.4 Multicolinearidade

Multicolinearidade em um modelo de regressão múltipla é quando duas ou mais variáveis independentes são fortemente relacionadas linearmente entre si. Segundo Gujarati (2011), o termo multicolinearidade deve-se a Ragnar Frisch. Originalmente, significava a existência de uma relação linear "perfeita" ou exata entre algumas ou todas as variáveis explanatórias do modelo de regressão. Na prática,

a colinearidade exata raramente ocorre e se ocorre geralmente é por falha na especificação do modelo (Maia, 2017).

Para calcular o grau de colinearidade que existe entre as variáveis explanatórias utiliza-se o fator de inflação da variância (*VIF*) que é uma medida que indica o quanto a variância de um estimador aumenta devido à multicolinearidade entre as variáveis independentes em um modelo de regressão (Gujarati, 2011). O *VIF* pode ser definido como:

$$VIF = \frac{1}{(1 - R^2)}, \tag{2.8}$$

em que R^2 é o coeficiente de determinação do modelo. Segundo Chatterjee e Simonoff (2012), o *VIF* não é um problema desde que satisfaça:

$$VIF < max\left(10, \frac{1}{1 - R^2_{\text{modelo}}}\right),$$

em que R^2_{modelo} é o coeficiente de determinação do modelo ajustado. De acordo com Chatterjee e Simonoff (2012), valores de VIF abaixo de 10 não se tornam um problema de colinearidade. A presença da multicolinearidade não afeta as propriedades dos estimadores de MQO mas afeta a identificação da contribuição isolada das covariáveis (Maia, 2017).

Apesar de não fazer parte das 5 hipóteses básicas descritas no Teorema de Gauss-Markov, uma hipótese importante a ser investigada em RLM é se os resíduos do modelo de regressão são normalmente distribuídos. Esta hipóteses é importante para a realização de inferências a respeito dos parâmetros do modelo tais como Intervalos de Confiança e Testes de Hipóteses.

2.6.5 Normalidade

A normalidade dos resíduos pode ser investigada de forma gráfica e via testes estatísticos. De forma gráfica, pode-se usar o Histograma, o Box-Plot e o QQ-Plot, por exemplo.

- **Histograma**

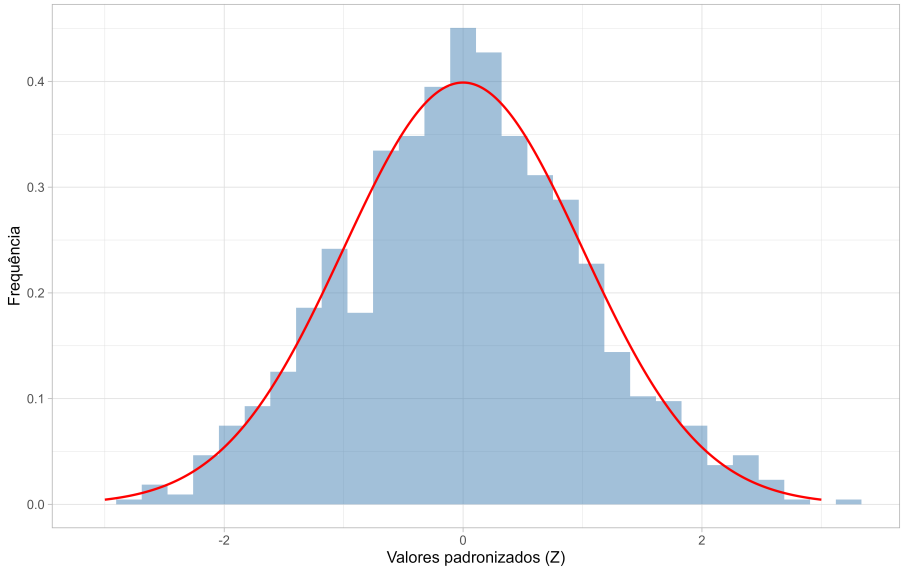
Uma maneira simples e fácil de se ter um indicativo sobre normalidade é através do histograma, onde a distribuição das frequências deve se aproximar da curva da densidade da distribuição Normal. A Figura 2 apresenta um histograma gerado através de uma amostra fictícia de uma distribuição Normal padrão, média zero e variância um. A curva em vermelho representa a curva teórica da distribuição Normal padrão.

- **QQ-Plot**

Uma outra maneira gráfica de indentificar a distribuição dos dados é o gráfico de QQ-Plot (Quantil quantil plot). O gráfico de QQ-Plot é construído através de uma amostra $x_1, x_2, \ldots, x_n$, de maneira que os quantis da amostra é plotado contra os quantis teóricos da distribuição de interesse, caso a distribuição da amostra se aproxime da distribuição teórica os pontos ficarão próximos a reta identidade $(x = y)$ (Loy, Follett e Hofmann, 2015).

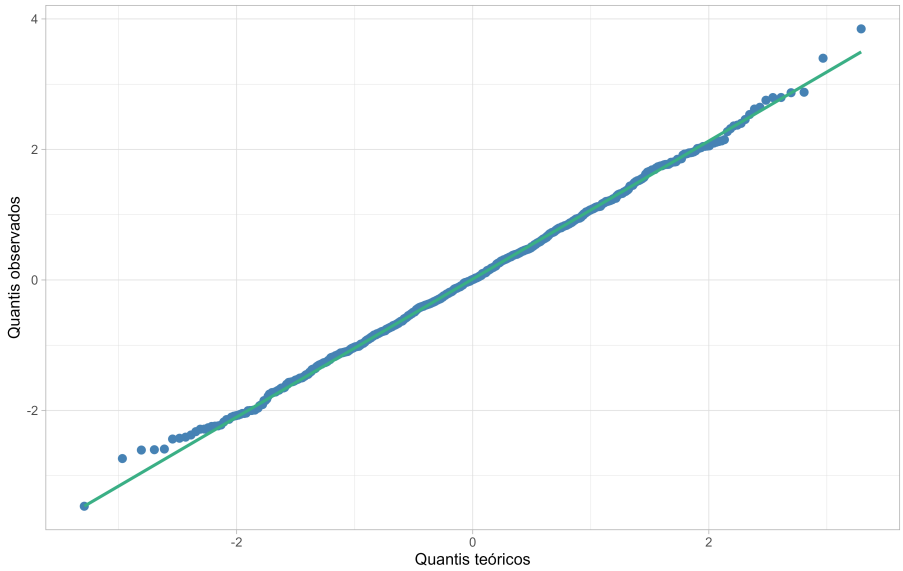
A Figura 3 apresenta um gráfico QQ-Plot para uma amostra aleatória de uma distribuição Normal padrão, os quantis teóricos são da distribuição Normal padrão.

Figura 2 – Histograma de dados gerados artificialmente de uma distribuição Normal padrão.



Fonte: Dados Simulados no R.

Figura 3 – QQ-Plot de dados gerados artificialmente de uma distribuição Normal padrão



Fonte: Dados Simulados no R.

- **Teste de Shapiro-Wilk**

Suponha que desejamos testar a hipótese:

$$H_0 : X \sim N(\mu, \sigma^2),$$

contra a hipótese:

$$H_1 : X \neq N(\mu, \sigma^2).$$

Para isso é necessário calcular a estatística W, obtida através da seguinte expressão:

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n}(x_i - \bar{X})^2},$$

em que $x_{(i)}$ é o i-ésimo menor valor da amostra, $x_i = x_1, x_2, ..., x_n$ são os valores observados da variável X ordenados e a_i são coeficientes tabulados (Hanusz, Tarasinska e Zieliński, 2016).

- **Teste de Anderson-Darling**

O Teste de Anderson-Darling (ou A-D) foi introduzido por Theodore Anderson e Donald Darling em 1952. Ele compara a distribuição empírica dos dados (ou seja, os dados reais que temos) com uma distribuição teórica (como a distribuição Normal ou outra distribuição específica).

A estatística do teste é dada por:

$$AD = -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\left[\log u_{(i)} + \log\left(1 - u_{(n-i+1)}\right)\right],$$

em que $u_{(i)} = F(u_{(i)})$ são os valores ordenados da distribuição teórica da Normal.

- **Teste de Kolmogorov-Smirnov**

O teste de Kolmogorov-Smirnov compara a distribuição empírica dos dados (ou seja, os dados reais coletados) com uma distribuição teórica (geralmente a distribuição Normal). Ele calcula a maior diferença entre essas duas distribuições, avaliando se os dados observados se ajustam bem à distribuição de referência.

A estatística do teste é definida por:

$$D_n = sup_x[|F(X) - S(X)|],$$

em que $F(X)$ é a função de distribuição acumulada da amostra, $S(X)$ é a função de distribuição acumulada da distribuição teórica a ser testada e D_n é o limite superior das diferenças pontuais entre as distribuições acumuladas téorica e observada.

2.6.6 Homocedasticidade

Dentre as suposições do teorema de Gauss-Markov, encontra-se a hipótese de homocedasticidade, que consiste em verificar se a variância do erro é constante na população condicional às variáveis independentes. Segundo Maia (2017), algumas das principais causas da heterocedasticidade são:

1. Natureza das Variáveis: algumas variáveis usualmente apresentam característica heterocedástica. Por exemplo, renda e poupança. Pessoas pobres são limitadas pela renda ao poupar e possuem pouca dispersão em relação aos valores médios de seu grupo econômico. Já entre os ricos, o comportamento é mais disperso: há os que poupam boa parcela de seus rendimentos, até aqueles que nada poupam.

2. Valores Extremos (*Outliers*): a presença de valores extremos na amostra pode aumentar a variabilidade dos erros em determinados pontos do modelo.

3. Omissão de Variáveis Importantes: quando variáveis relevantes são omitidas do modelo, a variabilidade dos erros pode aumentar ou diminuir em diferentes níveis das variáveis preditoras.

4. Transformações Inadequadas dos Dados: relações não lineares entre as variáveis independentes e dependentes podem causar heterocedasticidade. Transformações inadequadas dos dados também podem contribuir para esse problema

5. Especificação Incorreta do Modelo: um modelo mal especificado, que não captura corretamente a relação entre as variáveis, pode levar à heterocedasticidade.

A heterocedasticidade pode ter várias consequências negativas em um modelo de regressão linear, afetando a validade e a interpretação dos resultados. Maia (2017) enumera algumas das principais consequências:

1. Estimativas Ineficientes: Os coeficientes estimados ainda serão não viesados, mas não serão eficientes. Isso significa que os erros padrão dos coeficientes podem ser subestimados ou superestimados, levando a inferências estatísticas incorretas.

2. Testes de Hipóteses Comprometidos: A heterocedasticidade pode distorcer os testes de hipóteses, como o teste t e o teste F, resultando em valores p incorretos. Isso pode levar a conclusões erradas sobre a significância das variáveis independentes.

3. Intervalos de Confiança Inadequados: Os intervalos de confiança para os coeficientes podem ser mais estreitos ou mais largos do que deveriam ser, o que afeta a precisão das previsões e a confiança nas estimativas.

4. Previsões Menos Confiáveis: A heterocedasticidade pode reduzir a precisão das previsões do modelo, especialmente para valores fora da faixa dos dados observados.

5. Problemas de Diagnóstico: Pode ser mais difícil identificar a verdadeira relação entre as variáveis independentes e a variável dependente, complicando a interpretação dos resultados do modelo.

Podemos ter indícios da heterocedasticidade por meio de gráficos, e também pode ser confirmada via testes estatísticos.

- **De forma gráfica:** um gráfico de $\hat{\epsilon}^2 \times X_j$ ou $\hat{\epsilon}^2 \times \hat{Y}_j$; Os gráficos devem mostrar que não há padrão sistemático entre as duas variáveis, o que sugere a homocedasticidade dos dados.

- **Testes estatísticos:**

    1. Teste Goldfeld-Quandt
    2. Teste de Breusch-Pagan
    3. Teste de White

Neste trabalho, utilizou-se os Testes de White e Breusch-Pagan. Estes testes estatísticos já estão implementados no R *Core Team* (2024). Mais detalhes sobre o funcionamento dos testes podem ser encontrados em Gujarati (2011).

2.6.7 Autocorrelação

Uma das 5 hipóteses definidas pelo Teorema de Gauss-Markov é a ausência de autocorrelação nos erros do modelo de regressão.

Segundo Gujarati (2011) a autocorrelação pode ser definida como: "correlação entre integrantes de séries de observações ordenadas no tempo [como as séries temporais] ou no espaço [como nos dados de corte transversal]". No caso da regressão linear, se pressupõe que não haja esta autocorrelação nos erros. Gujarati (2011) e Maia (2017) elencam as principais causas da autocorrelação:

i) Inércia: uma característica que se destaca na maioria das séries temporais econômicas é a inércia. Séries temporais como PNB, índices de preço, produção, emprego e desemprego são cíclicas.

ii) Viés de especificação: esse viés pode ser resultado de variáveis que foram excluídas do modelo ou forma funcional incorreta.

iii) Defasagens: decisões econômicas em um período podem depender de informações de períodos anteriores. Ignorar essas relações pode resultar em correlação serial nos erros.

A autocorrelação pode ter várias consequências negativas. Gujarati (2011) e Maia (2017) elencam as principais:

i) Estimativas ineficientes: quando os erros forem autocorrelacionados, os estimadores MQO continuam sendo não viesados e consistentes, mas deixam de ser eficientes (em termos relativos). Isso significa que os estimadores não têm a menor variância possível, o que pode prejudicar a precisão das previsões.

ii) Erros-Padrão subestimados: a autocorrelação pode fazer com que os erros-padrão dos coeficientes sejam subestimados. Por esta razão, as estatísticas de teste t e F deixam de ser válidas, pois dependem da variância do estimador. Ou seja, os testes de significância podem indicar que um coeficiente é estatisticamente significativo quando, na verdade, não é.

iii) Viés nos testes de hipóteses: a autocorrelação pode levar a conclusões incorretas em testes de hipóteses. Por exemplo, pode aumentar a probabilidade de rejeitar a hipótese nula quando ela é verdadeira (erro tipo I).

iv) Previsões enganosas: modelos com autocorrelação podem gerar previsões que parecem precisas no curto prazo, mas que são enganosas no longo prazo, devido à dependência temporal não modelada adequadamente.

Para lidar com a autocorrelação, é comum utilizar métodos como a inclusão de defasagens das variáveis independentes, a transformação dos dados ou o uso de modelos específicos como o ARIMA (AutoRegressive Integrated Moving Average). Mais detalhes em Gujarati (2011). As formas de detecção podem ser por meio de gráficos e/ou testes estatísticos. Nesta monografia utilizou-se os Teste de Durbin-Watson e Breusch-Godfrey. Ambos estão implementados no software R *Core Team* (2024). Mais detalhes podem ser encontrados em diversos livros de econometria, por exemplo: Gujarati (2011), Maia (2017), Wooldridge (2015) e entre outros.

2.6.8 Pontos influentes

Pontos influentes em um modelo de regressão são observações que têm um impacto significativo nos resultados do modelo. Eles podem distorcer as estimativas dos coeficientes e afetar a precisão das previsões.

Um método tradicional para verificar os pontos de influência em um modelo de regressão é mediante de uma medida denominada Distância de Cook que é definido pela seguinte expressão:

$$D_i = \frac{\epsilon_i^2}{k+1} \times \frac{h_{ii}}{1-h_{ii}}, \quad i = 1, \ldots, n. \tag{2.9}$$

Em que ϵ_i é o i-ésimo resíduo, k é o número de parâmetros do modelo e h_i é o i-ésimo termo da matriz de projeção $\mathbf{H}$. A Matriz de projeção $\mathbf{H}$ é utilizada para detectar pontos de alavanca e é definida por:

$$\mathbf{H}_{(n\times n)} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}',$$

em que:

- $h_{ii} = \left(\frac{1}{n} + \frac{(x_i-\bar{x})^2}{s_{xx}}\right)$;
- $0 < h_{ii} < 1$;
- $\sum_{i=1}^{n} h_{ii} = k$; ($k$ é o número de colunas da matriz $\mathbf{X}$.)
- $h_{ij} = \left(\frac{1}{n} + \frac{(x_i-\bar{x})(x_j-\bar{x})}{s_{xx}}\right)$.

Não existe um consenso referente ao valor da Distância de Cook, Cook e Weisberg (1982) propuseram 1 como um valor de tolerância para a distância e Bollen e Jackman (1990) propuseram $4/n - k$, em que n é o número de observação e k o número de parâmetros do modelo.

2.7 Linguagem R e R Studio

O *software* (R *Core Team*, 2024) é uma linguagem de programação e ambiente de *software* para computação estatística e gráficos. Desenvolvido inicialmente nos Laboratórios *Bell*, o *software* (R *Core Team*, 2024) é semelhante à linguagem e ambiente do *software* S. O *software* (R *Core Team*, 2024) é considerado uma implementação diferente do *software* S, com algumas diferenças importantes, mas a maioria do código escrito para o *software* S pode ser executado no *software* (R *Core Team*, 2024) sem alterações. O *software* (R *Core Team*, 2024) é amplamente utilizado em diversas áreas de pesquisa e análise de dados devido à sua extensa gama de funções estatísticas e análises gráficas. Além disso, o (R *Core Team*, 2024) é um *software* de código aberto, o que significa que é livre para usar e distribuir (R *Core Team*, 2024).

O R Studio é uma interface gráfica do usuário (GUI) para a linguagem de programação do *software* (R *Core Team*, 2024). Ele fornece uma maneira amigável e eficiente de interagir com o (R *Core Team*, 2024), tornando a programação em (R *Core Team*, 2024) mais acessível para usuários iniciantes e mais eficiente para usuários avançados. O R Studio oferece uma série de recursos úteis, como edição de *scripts*, histórico de comandos e um ambiente de trabalho organizado. Esses recursos tornam o R Studio uma escolha popular entre os usuários do R para o desenvolvimento e análise de dados (Coursera, 2024). Além disso, o R Studio também suporta o desenvolvimento de aplicativos *web* interativos usando o pacote *Shiny* (Chang et al., 2024).

2.8 Pacote *Shiny*

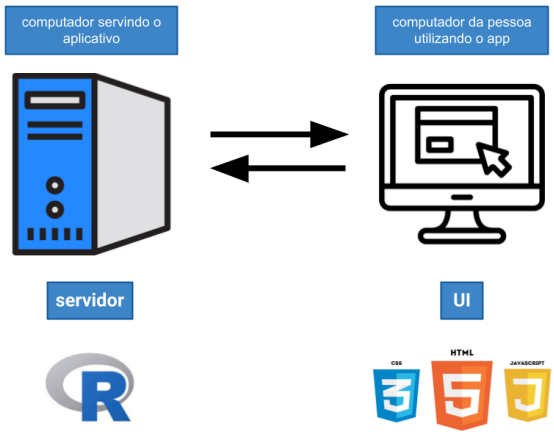
O aplicativo *Shiny* (Chang et al., 2024) é basicamente um site interativo construído com a linguagem de programação (R *Core Team*, 2024). Ele tem um URL próprio e é composto por HTML, CSS e *JavaScript*. Quando se acessa o URL, o site exibe informações em forma de texto e imagens (Amorim, 2024b).

O que torna o *Shiny* único é que ele não é apenas um site estático mas uma aplicação *web* interativa (Chang et al., 2024). Isso significa que é possível enviar dados para o site que serão processados e usados para gerar novas informações. Esta interação é realizada graças a um servidor onde o aplicativo *Shiny* (Chang et al., 2024) é hospedado, que mantém uma sessão de R *Core Team* (2024) em execução (Amorim, 2024b).

O código do aplicativo *Shiny* (Chang et al., 2024) é escrito em R *Core Team* (2024), usando o pacote *shiny* (Chang et al., 2024), juntamente com muitos outros pacotes disponíveis na comunidade R *Core Team* (2024) e criados pela R Studio. Estes pacotes permitem criar a estrutura e o d*esign* do site, estabelecer a comunicação entre o usuário e o servidor, e construir a lógica interna do aplicativo. Isso inclui a criação de visualizações de dados que são exibidas no site, tudo feito com código R *Core Team* (2024) puro. Desta forma é possível aproveitar todas as ferramentas de manipulação, visualização e modelagem de dados disponíveis em R *Core Team* (2024) para criar seus aplicativos Shiny (Amorim, 2024b).

A estrutura de um aplicativo *shiny* (Chang et al., 2024) consiste em dois componentes principais, a interface de usuário (UI ou *user interface*) e o servidor (*server*). O primeiro componente básico é a tela do aplicativo, basicamente é o que vemos quando estamos usando o aplicativo. O segundo componente do aplicativo é o que não vemos, é a lógica do aplicativo que é responsável por receber as entradas do usuário pela UI, processar as entradas e retornar o resultado de volta para a interface (Shiny, 2024). A Figura 4 faz uma exemplificação gráfica dos componentes principais de um aplicativo *shiny* (Chang et al., 2024) .

Figura 4 – UI/servidor de um aplicativo Shiny.



Fonte: Amorim (2024b).

2.8.1 Aplicativo com *Shiny*

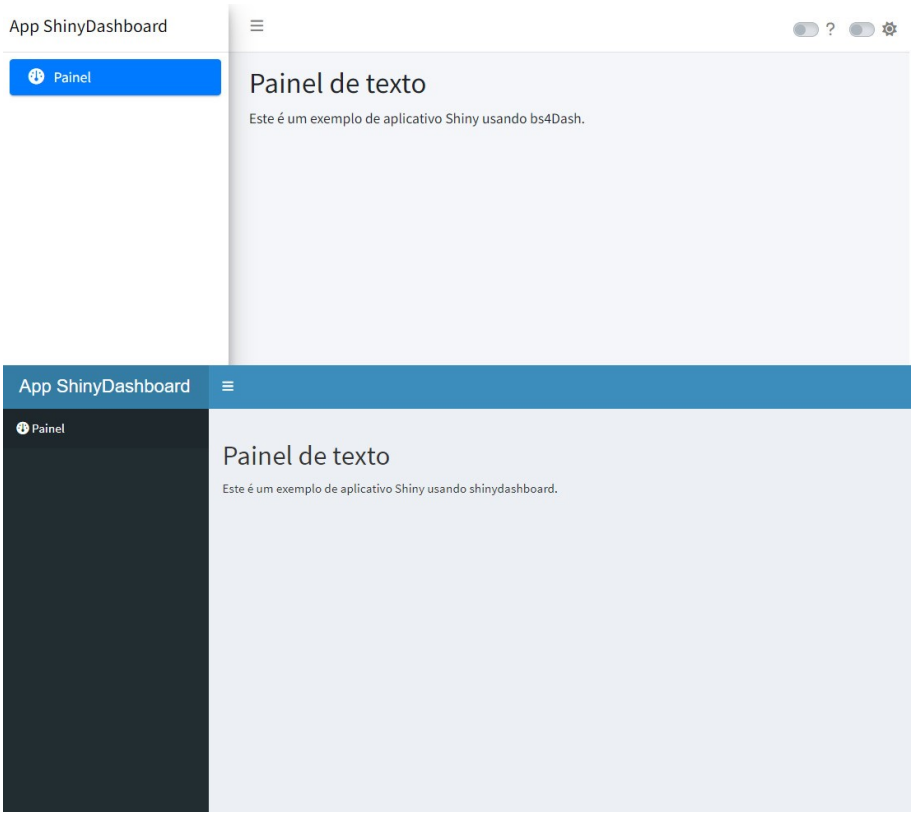
O pacote *shiny* (Chang et al., 2024) possui diversos *layouts* nativos, tais como : *sidebarLayout()*, *navbarPage()* e *navlistPanel()*. Além dos *layouts* nativos do *Shiny* (Chang et al., 2024), existem pacotes

complementares que oferecem ainda mais opções de layout. Por exemplo, o pacote *shinydashboard* permite a criação de painéis interativos com uma aparência profissional (Amorim, 2024a). Ele se baseia no modelo HTML AdminLTE (2024) oferencendo um layout de painel com uma barra lateral e um corpo principal.

Outro pacote útil é o *bs4Dash*, que também permite a criação de painéis interativos, mas com uma aparência mais moderna. Ele usa o *Bootstrap 4* e oferece suporte a recursos adicionais, como *popovers*, *tooltips*, alternância de tela cheia, troca de *skin* claro/escuro, barra lateral direita (*controlbar*), caixas fecháveis, barra lateral da caixa, e outras opções adicionais, no entanto os recursos adicionais do pacote *bs4Dash* não estão no escopo deste trabalho. A transição do *shinydashboard* para o *bs4Dash* pode ser realizada simplesmente mudando a biblioteca (Granjon, 2024).

Ambos os pacotes, *shinydashboard* e *bs4Dash*, podem ser usados em conjunto com o *Shiny* (Chang et al., 2024) para criar aplicativos *web* interativos, a Figura 5 apresenta a diferença visual entre esses dois pacotes.

Figura 5 – Exemplo de APP shiny usando os pacotes *shinydashboard* e *bs4Dash*



Fonte: R *Core Team* (2024).

A criação de um aplicativo *Shiny* (Chang et al., 2024) em R *Core Team* (2024) pode ser comparada, em muitos aspectos, à elaboração de um relatório na mesma linguagem. Ambos os processos envolvem a manipulação de dados, a implementação de funções e a apresentação de resultados. No entanto, existem algumas pequenas diferenças que diferenciam um aplicativo e um relatório (Amorim, 2024b). O Quadro 1 apresenta as principais funções utilizadas na criação de um aplicativo *Shiny* (Chang et al., 2024).

Quadro 1 – Descrição das funções

Função do servidor	Função da UI	Tipo de saída
renderDataTable()	dataTableOutput()	uma tabela interativa
renderImage()	imageOutput()	uma imagem salva
renderPlot()	plotOutput()	um gráfico R
renderPrint()	verbatimTextOutput()	uma saída de tipo console R
renderTable()	tableOutput()	uma tabela estática
renderText()	textOutput()	uma string de texto
renderUI()	uiOutput()	um elemento de tipo UI

Fonte: Adaptado de (Castro, 2023).

O *script* a seguir apresenta a estrutura da UI de um *dashboard* em *Shiny* (Chang et al., 2024).

```
library(shiny)
library(shinydashboard)

dashboardPage(
 dashboardHeader(),
 dashboardSidebar(),
 dashboardBody()
)
```

em que:

- *dashboardPage()*: É usada para criar a página do dashboard, que tem três componentes.
- *dashboardHeader()*: Define o cabeçalho do *dashboard*.
- *dashboardSidebar()*: Define a barra lateral do *dashboard*.
- *dashboardBody()*: Define o corpo do *dashboard*, que é onde os resultados são exibidos.

O *script* abaixo apresenta a função básica do *server* de um aplicativo *Shiny* (Chang et al., 2024):

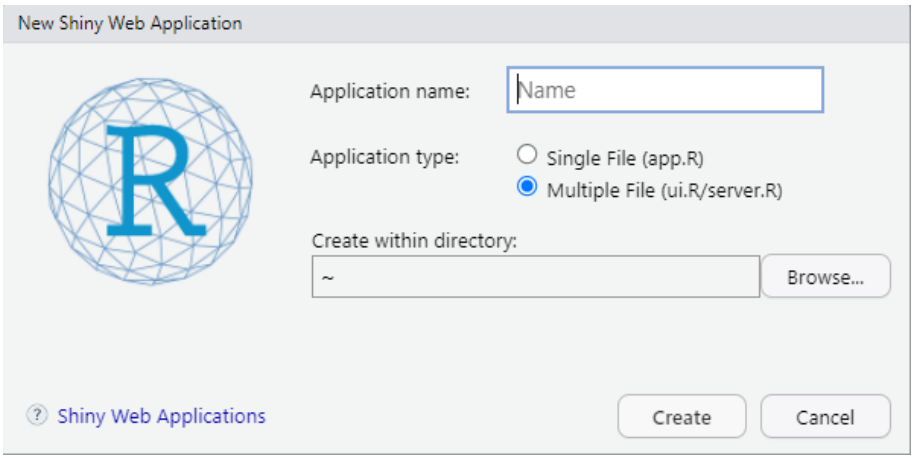
```
library(shiny)

function(input, output, session) {


}
```

O *input* é um objeto que contém todos os valores dos controles de entrada do aplicativo *Shiny*. O *output* é o objeto usado para enviar as saídas para a interface de usuário (UI), *session* é responsável por conter as informações da sessão atual do usuário (Posit, 2024).

Figura 6 – Criando um aplicativo em *Shiny*.



Fonte: R *Core Team* (2024).

Para criar um aplicativo *Shiny* (Chang et al., 2024) em R *Core Team* (2024) basta abrir o *software*, na barra superior clicar em "*File*", depois em "*New File*" e por fim clicar em "*Shiny Web App...*", após isso a seguinte imagem irá aparecer:

Em seguida, nomeie o aplicativo e em "*Application type*" é recomendado que se use o "*Multiple File*" para aplicativos maiores, essa opção fará com que o aplicativo seja dividido em dois código, um contendo a interface de usuário (UI) e outro contendo o *server*.

3 MATERIAL E MÉTODOS

3.1 Descrição dos dados

Os dados utilizados para a aplicação do aplicativo em *Shiny* (Chang et al., 2024) foram retirados do repositório da *UC Irvine Machine Learning Repository* referente a indústria *DAEWOO Steel* Co., Ltd, em Gwangyang, Coreia do Sul (UCI, 2023). Esta indústria produz diversos tipos de bobinas, chapas de aço e chapas de ferro. As informações sobre o consumo de energia elétrica são mantidas em um sistema baseado em nuvem. As informações sobre o consumo de energia da indústria são armazenadas no site da *Korea Electric Power Corporation* (pccs.kepco.go.kr), e as perspectivas sobre dados diários, mensais e anuais são calculadas e mostradas (UCI, 2023).

Originalmente os dados foram coletados de 15 em 15 minutos, iniciando em 01/01/2018 às 00:15 e terminando no dia 31/12/2018 às 00:00, devido a grande extensão do banco de dados eles foram agrupados por dia, iniciando no dia 01/01/2018 até o dia 31/12/2018. O Quadro 2 faz uma breve descrição dos dados.

Quadro 2 – Descrição dos Dados.

Variável	Descrição
Usage kWh	Consumo de energia diário em kWh(quilowatt-hora)
Lagging Current Reactive Power kVarh	Potência Reativa de Corrente Atrasada em kVarh(quilovolt-ampere reativo-hora)
Leading Current Reactive Power kVarh	Potência Reativa de Corrente Adiantada em kVarh(quilovolt-ampere reativo-hora)
CO2(tCO2)	Dióxido de carbono (Toneladas)
Lagging Current Power Factor	Fator de potência atrasado medido em %
Leading Current Power Factor	Fator de potência adiantado medido em %

Fonte: Adaptado de (UCI, 2023) .

Para o ajuste do modelo de regressão linear múltipla as variáveis *Lagging Current Reactive Power kVarh* e *Leading Current Reactive Power kVarh* foram agrupadas pela média diária, a variável CO2 foi considerado apenas o total diário e para as variáveis *Lagging Current Power Factor* e *Leading Current Power Factor* foi considerado a proporção média de ambas.

3.2 O aplicativo em *Shiny* para regressão linear

O aplicativo desenvolvido neste trabalho está em formato *dashboard* (painel visual com um conjunto de informações), que está organizado em 4 abas principais e 9 sub-abas, que estão dispostas da seguinte maneira:

- **Entrada do banco de dados**
- **Estatística descritiva**
  - Gráfico de dispersão;
  - Histograma;
  - Boxplot;

  - Medidas de resumo.

- **Regressão linear simples**
  - Ajuste de modelo;
  - Análise de resíduos;
  - Predição.

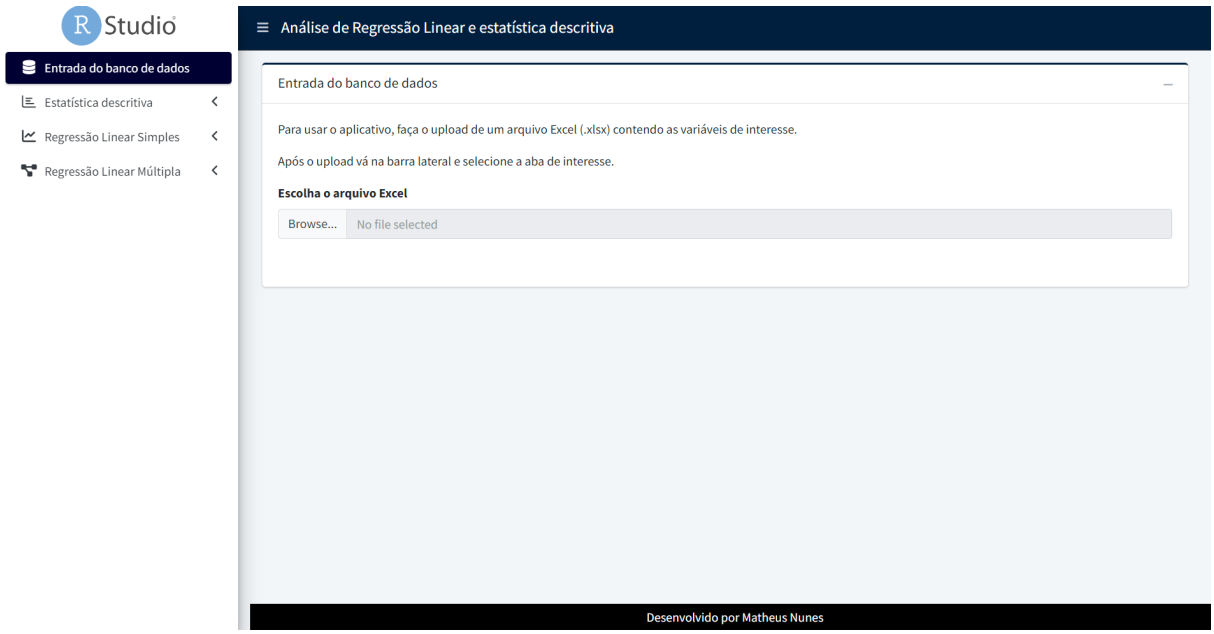
- **Regressão linear múltipla**
  - Ajuste de modelo;
  - Análise de Diagnóstico.

A interação do usuário com o aplicativo será fazendo uso do *mouse*, onde ele deverá arrastar e clicar o *mouse* sobre a aba de interesse e assim então prosseguir com suas análises, um breve tutorial será apresentado a seguir.

3.3 Entrada do banco de dados

Esta será a tela inicial do aplicativo onde o usuário deverá selecionar o banco de dados de interesse para seguir com as demais análises no aplicativo, para selecionar o conjunto de dados basta que o usuário selecione a opção *Browse* e carregue o arquivo de interesse. O conjunto de dados deve estar em formato **xlsx**.

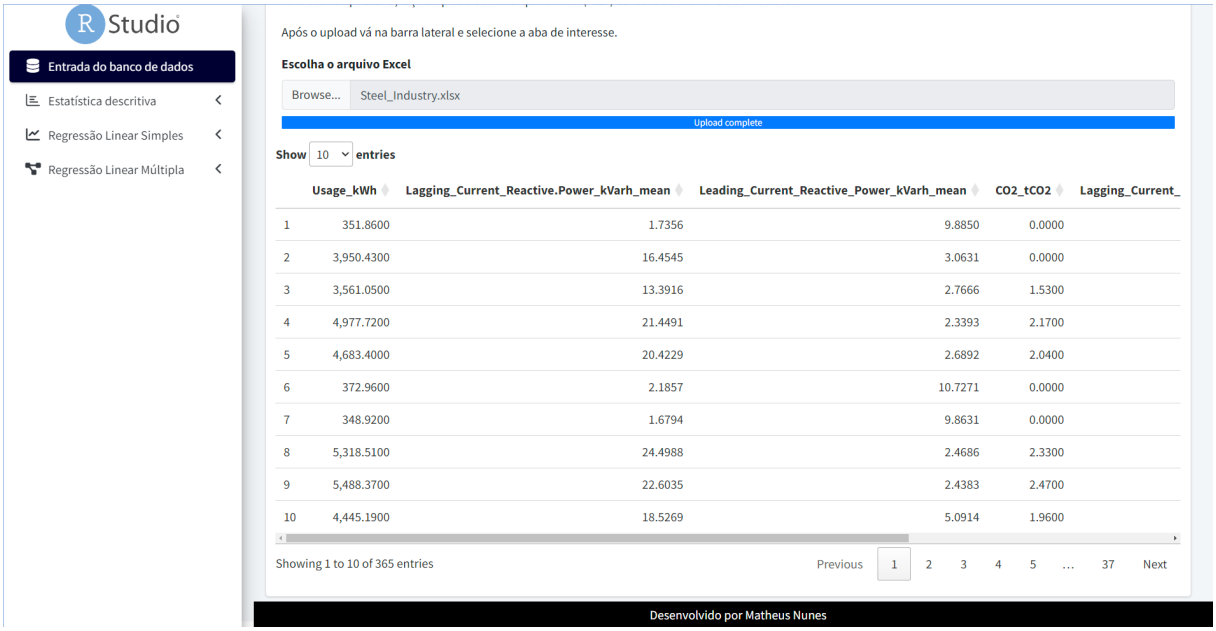
Figura 7 – Tela inicial do aplicativo.



Fonte:Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024).

Após a seleção do conjunto de dados, será exibido uma tabela apresentando as 10 primeiras linhas do conjunto de dados, para que o usuário possa ver o que ele inseriu no aplicativo.

Figura 8 – Tela inicial do aplicativo após carregar o conjunto de dados.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024).

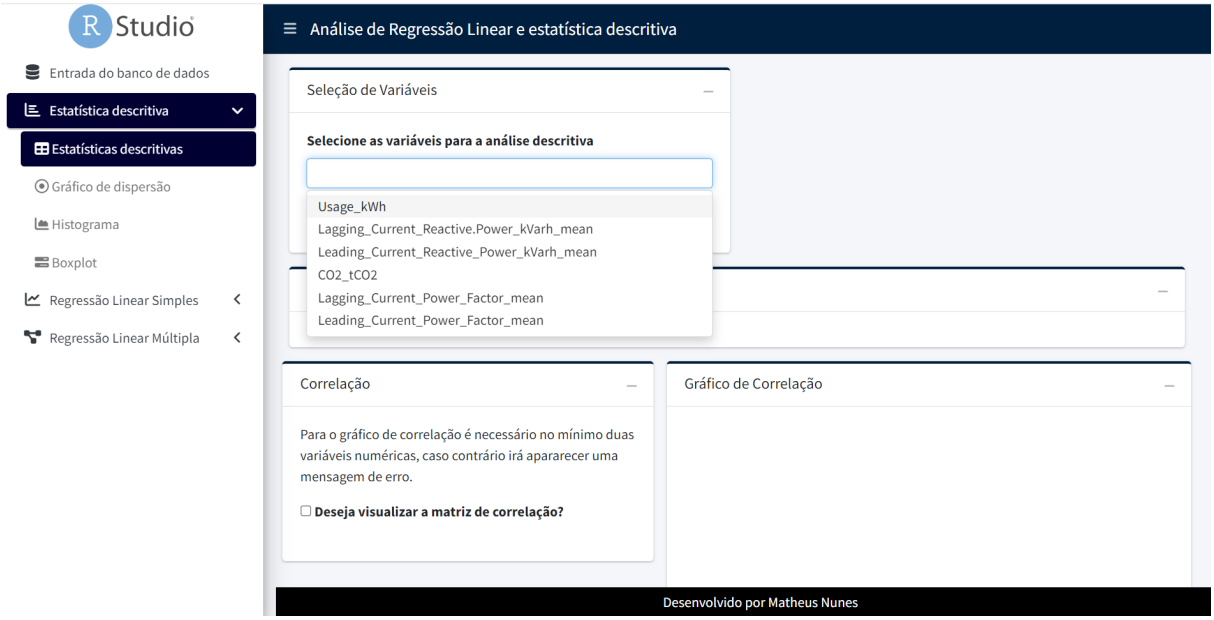
Seguindo adiante no aplicativo, após a inserção do conjunto de dados o usuário deverá ir para a aba de interesse, sendo elas: Estatística descritiva, Regressão linear simples e Regressão linear múltipla.

3.4 Estatística descritiva

Nesta seção, o usuário deve selecionar as variáveis numéricas para realizar as estatísticas descritivas e gerar gráficos. Embora seja possível carregar dados não numéricos, o aplicativo não foi adaptado para esse tipo de variável. Portanto, é importante que o usuário não tente aplicar a análise após selecionar qualquer variável que não seja numérica.

No primeiro quadrante superior do aplicativo na aba de estatística descritiva (Figura 9) tem a opção para a seleção de variáveis numéricas. Abaixo da caixa de seleção tem a caixa que aparecerá uma tabela contendo as medidas de resumo das variáveis, tais como: média, desvio padrão, erro padrão, mínimo e máximo. Abaixo da tabela de resumo existem duas caixas: à esquerda o usuário pode marcar a opção de plotar o gráfico da matriz de correlação das variáveis anteriormente selecionadas e à direita aparecerá o gráfico gerado.

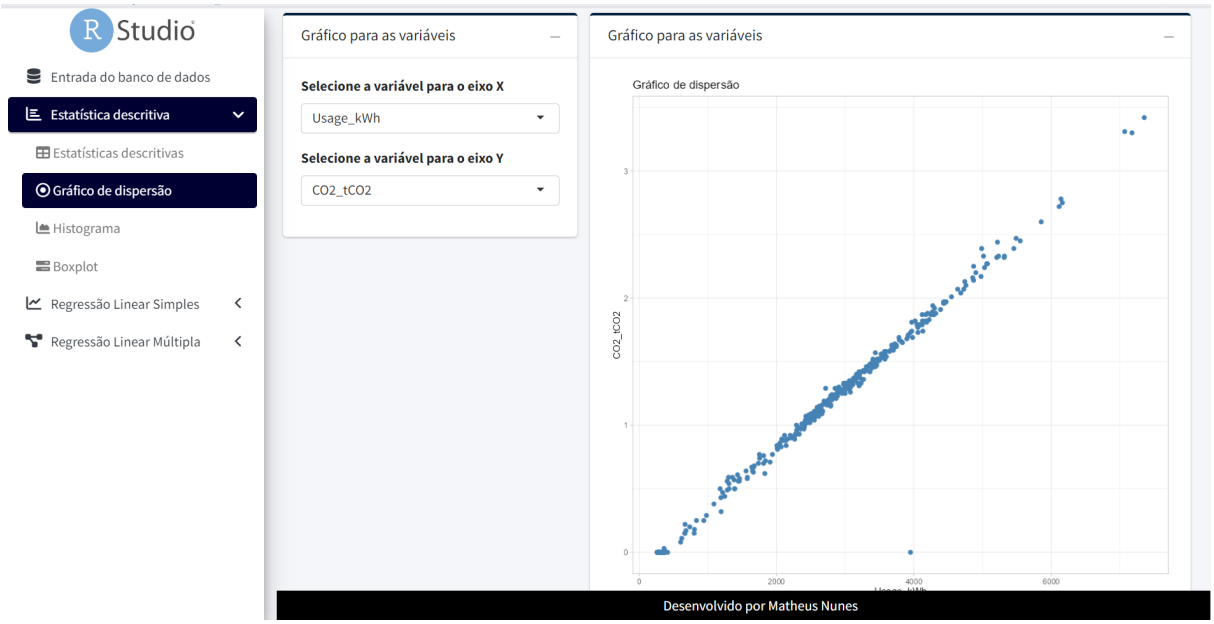
Figura 9 – Tela do aplicativo: aba de estatísticas descritivas.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024).

Na parte gráfica do aplicativo, todos os gráficos funcionam da mesma maneira, onde o usuário deverá apenas selecionar as variáveis do gráfico e então será plotado o gráfico referente as variáveis selecionadas (Figura 10). Para este exemplo foi mostrado o gráfico de dispersão para duas variáveis do conjunto de dados inserido na aba "Entrada do banco de dados".

Figura 10 – Tela do aplicativo: aba de estatística descritiva e sub-aba gráfico de dipersão

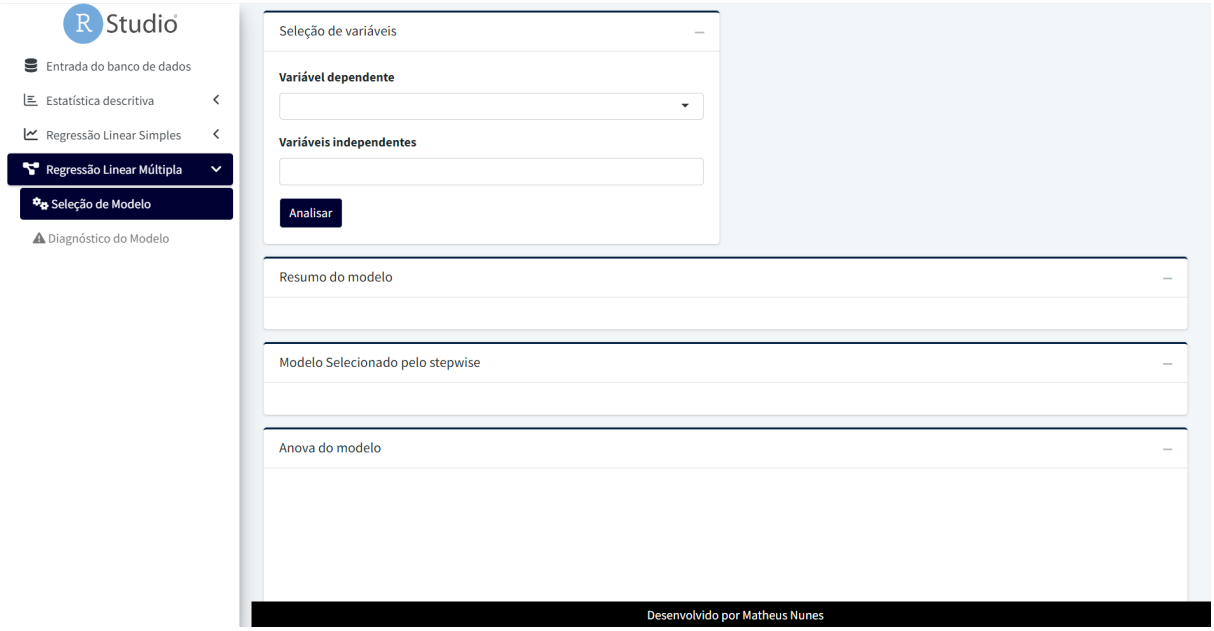


Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024).

3.5 Regressão linear simples e múltipla

Sendo a parte principal do aplicativo, as abas "Regressão linear simples"e "Regressão linear múltipla"funcionam basicamente da mesma maneira, por isso serão abordadas dentro de um único tópico. Para ajustar o modelo de regressão o usuário deverá selecionar a variável dependente Y e as variáveis independentes $X_j, \quad j = 1, 2, \ldots, k$. Em regressão linear simples só é possível selecionar uma variável. Após o ajuste do modelo, o usuário pode realizar uma predição onde insere-se um valor para a variável independente e estima-se o valor de Y. Na sub-aba de verificação da qualidade do ajuste em "Regressão linear múltipla" é possível calcular o VIF (equação 2.8) para verificar a existencia de multicolinearidade entre as variáveis.

Figura 11 – Tela do aplicativo: aba de regressão linear múltipla.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024).

Nesta aba do aplicativo, no primeiro quadrante aparecerá novamente o conjunto de dados que o usuário inseriu na entrada do banco de dados, logo abaixo estará uma caixa contendo as opções de seleção de variáveis do modelo, e em seguida a opção de aplicar.

Após aplicar, 3 tabelas serão exibidas ao usuário:

- **Resumo do modelo**: aparecerá os coeficientes do modelo completo com todas as variáveis selecionadas pelo usuário.
- **Modelo selecionado pelo *stepwise***: uma tabela similar a anterior, porém com as variáveis selecionadas pelo *stepwise*, esta tabela está presente apenas da aba de regressão linear múltipla.
- **Anova do modelo**: tabela de análise de variância do modelo selecionado via *stepwise*.

3.6 Qualidade do ajuste

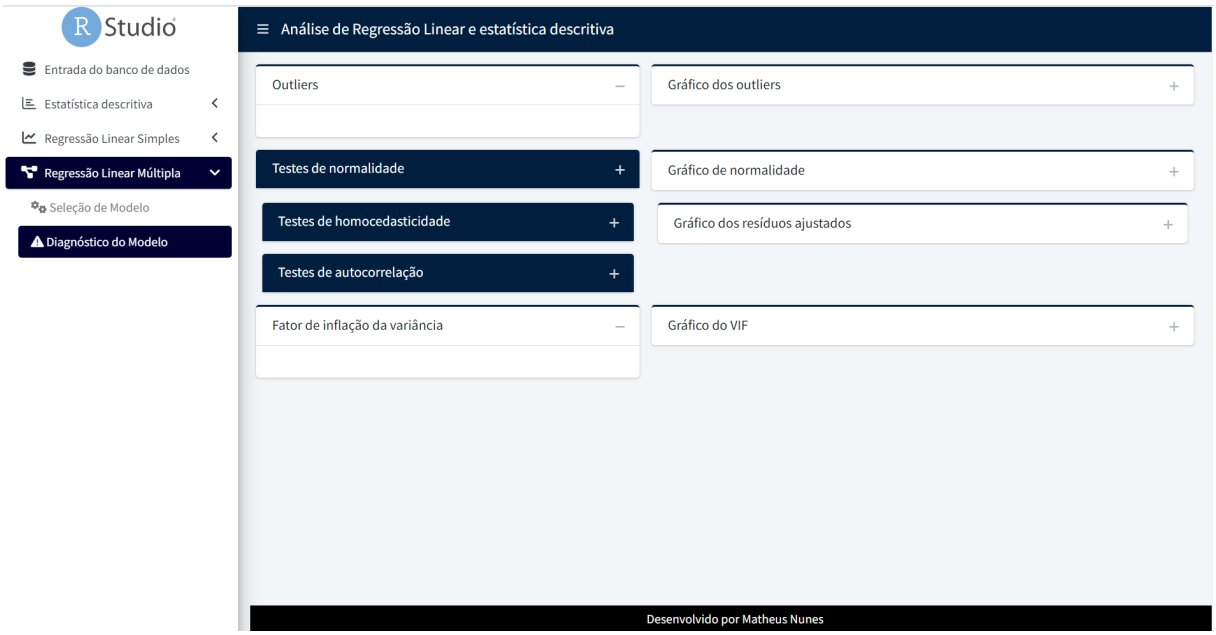
Esta sub-aba contém algumas medidas para a verificação da qualidade do ajuste apresentadas no Referencial Teórico:

- Gráfico dos *outliers*;
- Testes de normalidade e o gráfico "QQ-plot";
- Testes de homocedasticidade e gráfico dos resíduos padronizados;
- Teste de autocorrelação;
- Fator de inflação da variância (VIF) e gráfico.

A (Figura 12) apresenta a disposição das caixas com os testes estatísticos e gráficos apresentados neste tópico. No caso da regressão linear simples as caixas fator de inflação da variância e Gráfico do VIF não estão presentes, já que são medidas utilizadas em regressão linear múltipla.

Na caixa de *outliers* é apresentada uma tabela contendo os *outliers* detectados pela distância de *cook* (equação 2.9). Se o conjunto de dados não tiver *outliers* a tabela estará vazia. São apresentados os testes de normalidade: Teste de Shapiro-Wilk, Teste Anderson-Darling e Teste de Kolmogorov-Smirnov. Para testar a homocedasticidade dos resíduos: Testes de White e Teste de Breusch-Pagan e para verificar se os erros são autocorrelacionados: Teste de Durbin-Watson e Teste Breusch-Godfrey.

Figura 12 – Tela do aplicativo: aba de diagnóstico do modelo.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024).

4 RESULTADOS E DISCUSSÃO

Neste capítulo são apresentados os resultados das análises feitas para os dados que foram retirados do repositório da *UC Irvine Machine Learning Repository* referente a indústria *DAEWOO Steel* Co., Ltd, em Gwangyang, Coreia do Sul (UCI, 2023). Primeiramente são apresentadas as estatísticas descritivas para todas as variáveis e em seguida um modelo de regressão linear múltiplo será ajustado.

4.1 Estatísticas descritivas

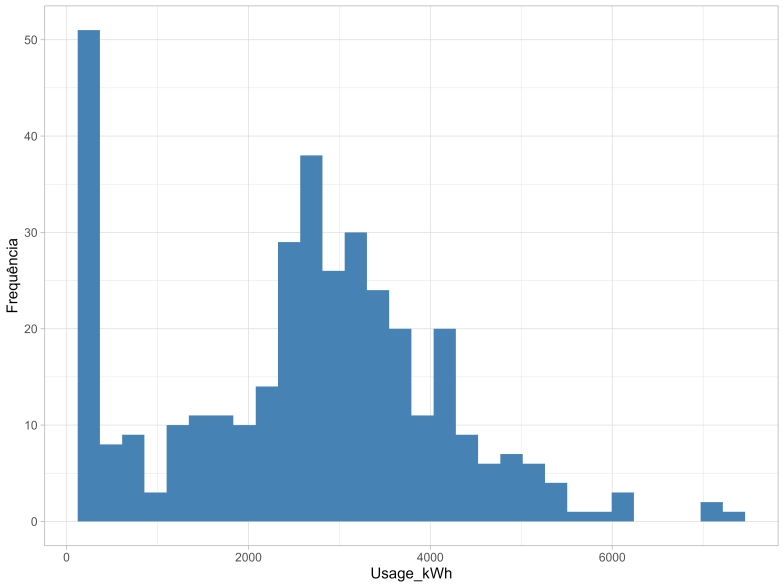
Tabela 2 – Estatísticas descritivas das variáveis apresentadas no Quadro 2.

Variáveis	Média	Desvio Padrão	Erro padrão	Mínimo	Máximo
Usage kWh	2.629,1417	1.480,1734	77,4758	259,4600	7.353,8400
Lagging C. R. P. kVarh	13,0354	7,4339	0,3891	1,1890	36,3566
CO2_tCO2	1,1063	0,7074	0,0370	0,0000	3,4200
Lagging C. P. F.	80,5781	5,5341	0,2897	69,3727	92,8384
Leading C. R. P. kVarh	3,8709	2,8162	0,1474	0,0330	12,9205
Leading C. P. F.	84,3679	13,9681	0,7311	51,6533	99,9992

Fonte: UCI (2023).

Dentre as variáveis descritas, a variável de interesse é o Consumo de energia diário (kWh) denominada de *Usage kWh*. Foi construído um histograma com os dados observados desta variável para observarmos o comportamento dos dados. O resultado é apresentado na Figura 13.

Figura 13 – Histograma da variável *Usage kWh*



Fonte: UCI (2023).

4.2 Ajuste do modelo

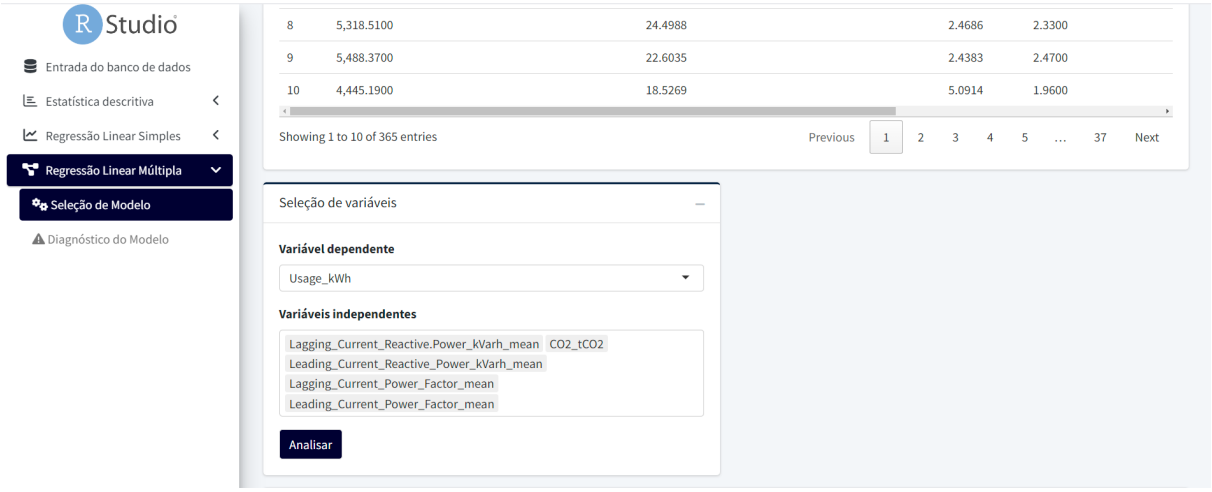
Nesta seção são apresentados os resultados do ajuste realizado para o seguinte modelo de regressão linear múltiplo:

$$Usage_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \beta_3 X_{i3} + \beta_4 X_{i4} + \beta_5 X_{i5} + \epsilon_i, \quad i = 1, \ldots, n.$$

Na equação acima, $Usage_i$ é a variável resposta de interesse, $X_{i1}, \ldots, X_{i5}$ representam genericamente as variáveis independentes e ϵ_i é o erro aleatório.

Para o ajuste do modelo no aplicativo, foram selecionadas as variáveis independentes e a variável dependente. Após a seleção, marcou-se a opção "Analisar", como mostra a Figura 14.

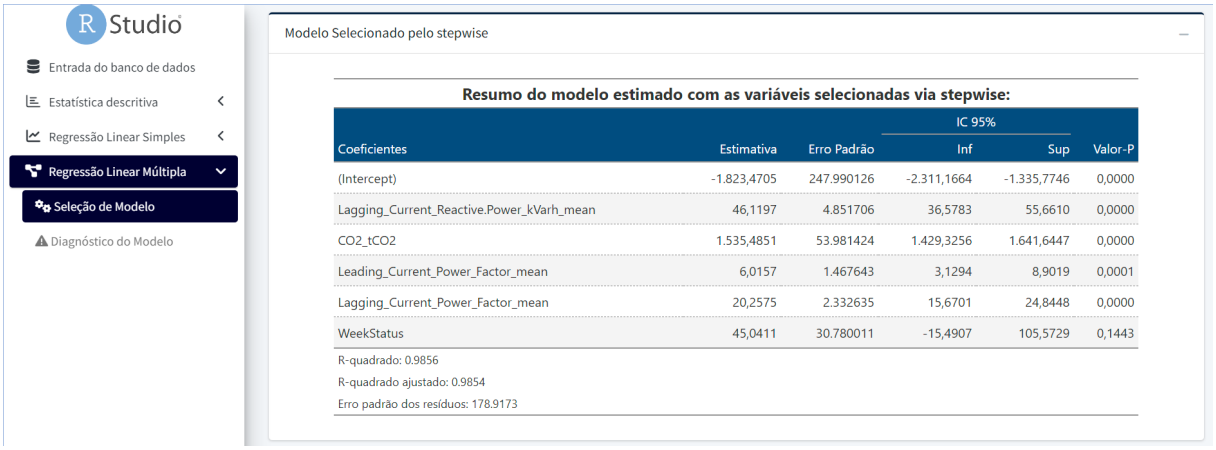
Figura 14 – Seleção de variáveis.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

Após marcar a opção "Analisar", o aplicativo irá retornar 2 tabelas: uma contendo os coeficientes do modelo completo e outra contendo os coeficientes apenas das variáveis selecionadas via *stepwise*. A Figura 15 apresenta as estimativas dos coeficientes do modelo selecionado via *stepwise*.

Figura 15 – Variáveis do modelo selecionadas via *stepwise*.

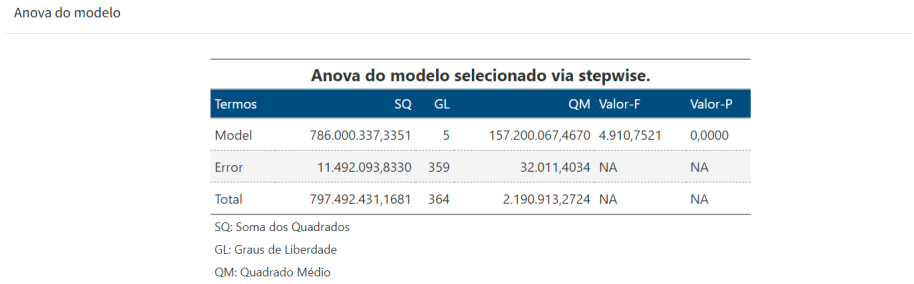


Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

Analisando a Figura 15 verifica-se que o método *stepwise* removeu a variável *Leading Currente Reactive Power kVarh* resultando em um modelo com quatro variáveis independentes. No rodapé da tabela contida na Figura 15, tem-se o R-quadrado e o R-quadrado ajustado do modelo. O R^2 de aproximadamente 0,9855 indica que essas 4 variáveis são suficientes para explicar aproximadamente 98,55% da variabilidade da variável dependente.

Após o ajuste do modelo de regressão, foi realizada uma análise de variância (anova) apresentada na Figura 16. Analisando os resultados vemos que valor-p associado ao teste F é inferior a um nível de significância de 5% indicando que o modelo é significativo para explicar a variável dependente.

Figura 16 – Anova do modelo selecionado via *stepwise*.


Anova do modelo

Anova do modelo selecionado via stepwise.

Termos	SQ	GL	QM	Valor-F	Valor-P
Model	786.000.337,3351	5	157.200.067,4670	4.910,7521	0,0000
Error	11.492.093,8330	359	32.011,4034	NA	NA
Total	797.492.431,1681	364	2.190.913,2724	NA	NA

SQ: Soma dos Quadrados
GL: Graus de Liberdade
QM: Quadrado Médio

Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

4.3 Qualidade do ajuste

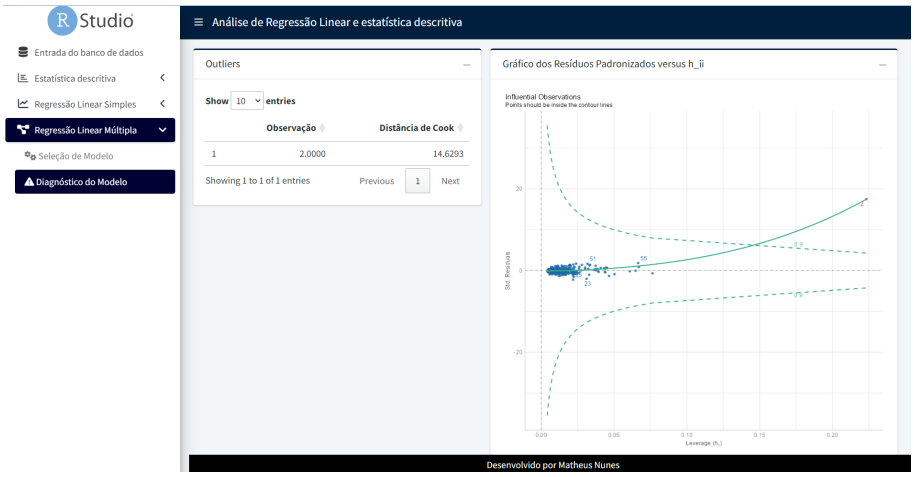
Na seção 2.6 apresentou-se diversas hipóteses e medidas necessárias para garantir uma boa qualidade no ajuste do modelo de regressão. Estas medidas foram implementadas no aplicativo e são apresentadas nesta seção utilizando o modelo ajustado e selecionado via *stepwise* apresentado na seção anterior. As medidas de qualidade de ajuste implementadas no aplicativo são:

1. Gráfico dos Resíduos Padronizados *versus* h_{ii};
2. Distância de Cook;
3. Gráfico QQ-plot;
4. Testes de Normalidade;
5. Teste de homocedásticidade;
6. Teste de Autocorrelação;
7. Fator de Inflação da Variância (VIF).

Analisando a (Figura 17) vemos que a 2ª observação foi detectada como um *outlier*. Tanto na distância de *Cook* quanto na representação gráfica.

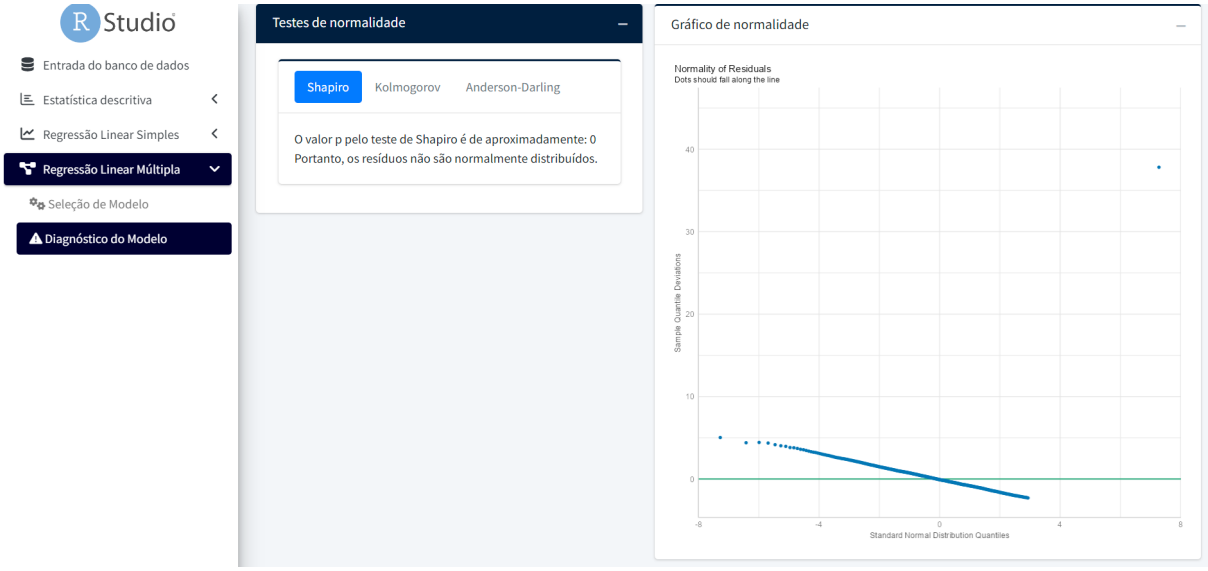
A Figura 18 apresenta o gráfico de QQ-plot para o modelo de regressão selecionado via *stepwise*. Analisando a figura vemos indícios de que resíduos não são normalmente distribuídos, com os testes de Shapiro-Wilk, Anderson Darling e Kolmogorov-Smirnov apresentaram Valor-p menor que 5% confirmando a suspeita apresentada no gráfico.

Figura 17 – Distância de Cook e Gráfico dos Resíduos Padronizados *versus* h_{ii}.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

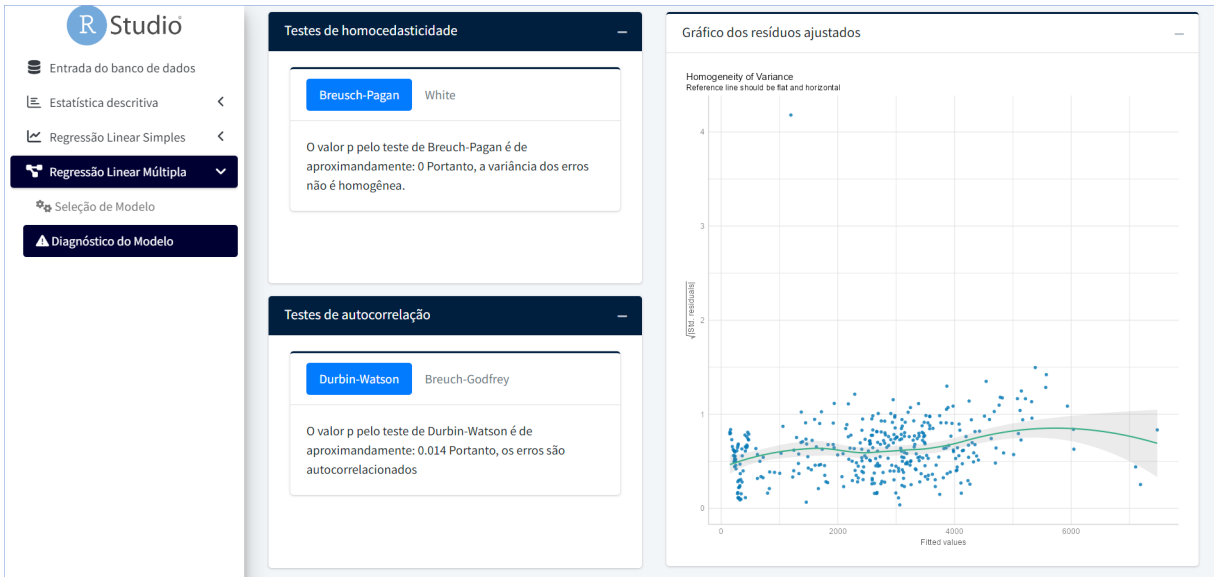
Figura 18 – QQ-plot para o modelo selecionado via *stepwise*.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

Analisando a Figura 19, vemos que os testes de homocedasticidade e de autocorrelação, rejeitaram H_0, pois o valor-p para ambos os testes foram abaixo do nível de significância de 5%. Ou seja, a variância dos erros não é homogênea e é autocorrelacionada.

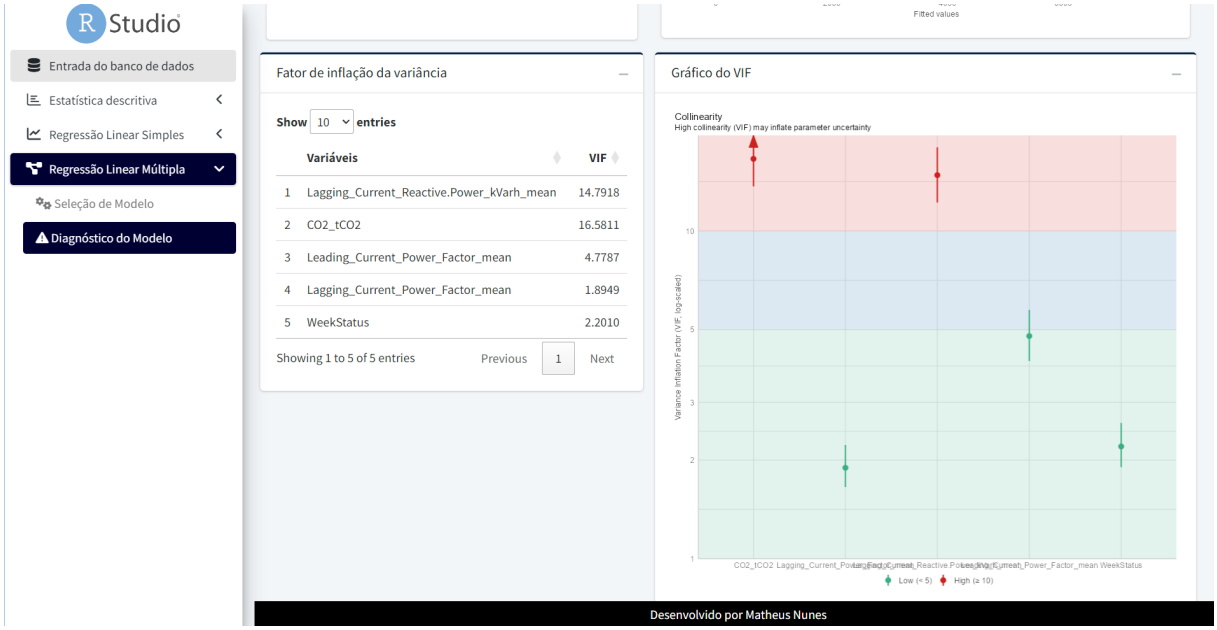
Figura 19 – Testes de Homocedasticidade e Autocorrelação.



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

Para verificar a presença de multicolinearidade, analisamos a Figura 20. Analisando os valores do Fator de Inflação da Variância -*VIF*, vemos que duas variáveis do modelo selecionado estão fortemente correlacionadas, pois o *VIF* para ambas é superior a 10.

Figura 20 – Fator de Inflação da Variância (*VIF*).



Fonte: Imagem da tela do aplicativo desenvolvido no R *Core Team* (2024) para os dados provenientes de UCI (2023).

5 CONCLUSÃO

A proposta deste trabalho consistiu no desenvolvimento de um aplicativo utilizando *Shiny* - R Studio, que permite ao usuário realizar uma análise descritiva das variáveis e uma análise de regressão simples ou múltipla após o *upload* de uma base de dados em Excel. Além da análise de regressão, o aplicativo dispõe de ferramentas de seleção de variáveis e análise de diagnósticos do modelo. Todas essas ferramentas estatísticas são apresentadas de forma intuitiva no aplicativo, tornando-o acessível a usuários de diversas áreas do conhecimento.

Para testar o funcionamento do aplicativo, foram realizadas análises descritivas e de regressão linear múltipla utilizando o banco de dados da *DAEWOO Steel Co*., Ltd. Os resultados demonstraram a eficiência do aplicativo em lidar com dados reais, através de análises descritivas e regressões lineares simples e múltiplas, além da seleção de modelos e análise de diagnósticos. A capacidade de inserir bases de dados em Excel torna o aplicativo prático e funcional para uma ampla gama de usuários.

Com a publicação do aplicativo na *web*, espera-se que ele se torne uma ferramenta valiosa não apenas para pesquisadores e profissionais, mas também para professores e alunos, facilitando o ensino e o aprendizado da regressão linear. A disponibilização gratuita desta ferramenta pode contribuir significativamente para a disseminação do conhecimento estatístico e para a melhoria da qualidade das análises realizadas em diversas áreas, promovendo uma tomada de decisões mais informada e precisa. O aplicativo pode ser encontrado nesse seguinte endereço *web*:

https://2bj2vw-matheus-paiva.shinyapps.io/Analise_regressao/

REFERÊNCIAS

ADMINLTE. *AdminLTE - Bootstrap 5 Admin Dashboard*. 2024. Acesso em: 05/04/2024. Disponível em: <https://github.com/ColorlibHQ/AdminLTE>.

AKAIKE, H. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, v. 19, n. 6, p. 716–723, 1974.

ALOOBA. *Linear Model Analysis: Everything You Need to Know When Assessing*. 2024. Accesso em: 05/04/2024. Disponível em: <https://www.alooba.com/skills/concepts/statistics/linear-model-analysis/>.

AMORIM, W. *Layouts*. 2024. [Online; acesso em 14-03-2024]. Disponível em: <https://programando-em-shiny.curso-r.com/layouts.html>.

AMORIM, W. *Seu primeiro aplicativo Shiny*. 2024. [Online; acesso em 14-03-2024]. Disponível em: <https://programando-em-shiny.curso-r.com/primeiro-app.html>.

BOLLEN, K. A.; JACKMAN, R. W. Regression diagnostics: An expository treatment of outliers and influential cases. In: FOX, J.; LONG, J. S. (Ed.). *Modern Methods of Data Analysis*. [S.l.]: SAGE, 1990.

BURNHAM, K. P.; ANDERSON, D. R. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. 2nd. ed. [S.l.]: Springer, 2002. 60-64 p.

CASTRO, H. C. . J. S. . J. G. . P. . C. *Tutorial R Shiny*. Portugal: Repositório Universidade do Minho, 2023. 23 p.

CHANG, W. et al. *shiny: Web Application Framework for R*. [S.l.], 2024. R package version 1.8.1.9001, https://github.com/rstudio/shiny. Disponível em: <https://shiny.posit.co/>.

CHARNET, R. et al. *Análise de modelos de regressão linear: com aplicações*. [S.l.]: Editora da UNICAMP, 2008.

CHATTERJEE, S.; SIMONOFF, J. S. *Handbook of Regression Analysis*. Wiley, 2012. Disponível em: <https://api.semanticscholar.org/CorpusID:60060694>.

COOK, R. D.; WEISBERG, S. *Residuals and Influence in Regression*. New York: Chapman and Hall, 1982. v. 5.

COURSERA. *Data Analysis with RStudio: Understanding the Basics*. 2024. Accesso em 05/04/2024. Disponível em: <https://www.coursera.org/blog/data-analysis-with-rstudio>.

DACHS, J. N. W. *Análise de Dados e Regressão*. Campinas: IMECC UNICAMP, 1978. 44 p.

DODGE, Y. *The Concise Encyclopedia of Statistics*. New York, NY: Springer, 2008. ISBN 978-0-387-31745-5.

EMILIANO, P. C. *Fundamentos e Aplicações dos Critérios de Informação: AKAIKE E BAYESIANO*. 40 p. Dissertação (Mestrado) — Universidade Federal de Labras, Minas-Gerais, 2009.

ESCOLA, S. *O que é: R Shiny*. 2024. [Online; acesso em 14-08-2024]. Disponível em: <https://www.soescola.com/glossario/o-que-e-r-shiny>.

FIGUEIREDO, A. M. R. *Econometria básica: Shiny code para regressão múltipla em R*. Campo Grande-MS, Brasil: [s.n.], 2019. RStudio/Rpubs. Disponível em: <http://rpubs.com/amrofi/Econometrics_shiny_regressao_af>.

GALTON, F. Regression towards mediocrity in hereditary stature. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 1886.

GAUSS, C. F. Theoria combinationis observationum erroribus minimis obnoxiae. *Annalen der Physik*, Wiley Online Library, v. 69, n. 1, p. 29–48, 1823.

GRANJON, D. *bs4Dash: A 'Bootstrap 4' Version of 'shinydashboard'*. [S.l.], 2024. R package version 2.3.3, https://github.com/RinteRface/bs4Dash. Disponível em: <https://rinterface.github.io/bs4Dash/index.html>.

GRAYBILL, F. A. *Theory and Application of the Linear Model*. North Scituate, Mass.: Duxbury Press, 1976. 704 p. Includes index. Bibliography: p. 687-698. ISBN 0878721088.

GUJARATI, D. N. *Econometria Básica*. 5. ed. [S.l.]: Makron Books, 2011.

HANUSZ, Z.; TARASINSKA, J.; ZIELIńSKI, W. Shapiro–wilk test with known mean. *Revstat Statistical Journal*, v. 14, p. 89–100, 02 2016.

HOFFMANN, R. *ANÁLISE DE REGRESSÃO Uma Introdução à Econometria*. 5. ed. Piracicaba: HUCITEC, 2016.

KONRATH, A. et al. Aplicativo shiny como suporte de ensino de métodos de previsão. *Abakós*, v. 7, p. 35–50, 11 2019.

KONRATH, A. et al. Desenvolvimento de aplicativos web com r e shiny: inovações no ensino de estatística. *Abakós*, v. 6, p. 55–71, 05 2018.

LOY, A.; FOLLETT, L.; HOFMANN, H. *Variations of Q-Q Plots – The Power of our Eyes!* 2015. Disponível em: <https://arxiv.org/abs/1503.02098>.

MAIA, A. G. *Econometria: conceitos e aplicações.* [S.l.]: Editora Saint Paul., 2017. 384 p.

MIRANDA, S. A. da Silva; Andréa Cristina Konrath; Luiz Ricardo Nakamura;Rodrigo Gabriel de. ConstruÇÃo de um aplicativo em shiny para anÁlise exploratÓria de dados. In: *III Seminário Internacional de Estatística com R.* Niterói-RJ: [s.n.], 2018. [Online; acesso em 14-08-2024]. Disponível em: <https://silvio.shinyapps.io/ingressoifsc/>.

POSIT. *Understanding Shiny: Session Object*. 2024. Acesso em: 05/04/2024. Disponível em: <https://www.posit.com/blog/understanding-shiny-session-object>.

R *Core Team. R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2024. [Online; acesso em 14-03-2024]. Disponível em: <https://www.R-project.org/>.

SAAVEDRA, C. A. P. B.; LOBOS, C. M. V. Um aplicativo shiny para modelos lineares generalizados. In: *Reunião Anual da Região Brasileira da Sociedade Internacional de Biometria (RBras)*. [S.l.]: Universidade Federal do Paraná, 2018.

Shiny. *Welcome to Shiny*. 2024. Acesso em: 05/04/2024. Disponível em: <https://shiny.posit.co/r/getstarted/shiny-basics/lesson1/index.html>.

SIEVERT, C. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020. ISBN 9781138331457. Disponível em: <https://plotly-r.com>.

SUREIMAN, O.; MANGERA, C. F-test of overall significance in regression analysis simplified. *Journal of the Practice of Cardiovascular Sciences*, v. 6, p. 116, 01 2020.

UCI, U. I. M. L. R. *Steel Industry Energy Consumption*. 2023. [Online; acessado em 14-03-2024]. Disponível em: <https://archive.ics.uci.edu/dataset/851/steel+industry+energy+consumption>.

WICKHAM, H. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016. ISBN 978-3-319-24277-4. Disponível em: <https://ggplot2.tidyverse.org>.

WOOLDRIDGE, J. M. *Introductory Econometrics: A Modern Approach*. 6. ed. [S.l.]: Cengage Learning, 2015. 789 p. ISBN 1305446380.