

CADERNO Nº 1

ELTON NUNES BRITTO  
DAMIÃO PEDRO MEIRA FILHO  
MARIA RAIMUNDA SILVA REGO

---

# ESTATÍSTICA APLICADA À BIOLOGIA

USANDO O



ESTATÍSTICA DESCRITIVA E  
INFERENCIAL

---

 **Atena**  
Editora

Ano 2021

CADERNO Nº 1

ELTON NUNES BRITTO  
DAMIÃO PEDRO MEIRA FILHO  
MARIA RAIMUNDA SILVA REGO

---

# ESTATÍSTICA APLICADA À BIOLOGIA

USANDO O



ESTATÍSTICA DESCRITIVA E  
INFERENCIAL

---

 **Atena**  
Editora

Ano 2021

**Editora Chefe**

Profª Drª Antonella Carvalho de Oliveira

**Assistentes Editoriais**

Natalia Oliveira

Bruno Oliveira

Flávia Roberta Barão

**Bibliotecária**

Janaina Ramos

**Projeto Gráfico e Diagramação**

Natália Sandrini de Azevedo

Camila Alves de Cremona

Luiza Alves Batista

Maria Alice Pinheiro

**Imagens da Capa**

Shutterstock

**Edição de Arte**

Luiza Alves Batista

**Revisão**

Os autores

2021 by Atena Editora

Copyright © Atena Editora

Copyright do Texto © 2021 Os autores

Copyright da Edição © 2021 Atena Editora

Direitos para esta edição cedidos à Atena Editora pelos autores.

Open access publication by Atena Editora



Todo o conteúdo deste livro está licenciado sob uma Licença de Atribuição *Creative Commons*. Atribuição-Não-Comercial-NãoDerivativos 4.0 Internacional (CC BY-NC-ND 4.0).

O conteúdo dos artigos e seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva dos autores, inclusive não representam necessariamente a posição oficial da Atena Editora. Permitido o *download* da obra e o compartilhamento desde que sejam atribuídos créditos aos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Todos os manuscritos foram previamente submetidos à avaliação cega pelos pares, membros do Conselho Editorial desta Editora, tendo sido aprovados para a publicação com base em critérios de neutralidade e imparcialidade acadêmica.

A Atena Editora é comprometida em garantir a integridade editorial em todas as etapas do processo de publicação, evitando plágio, dados ou resultados fraudulentos e impedindo que interesses financeiros comprometam os padrões éticos da publicação. Situações suspeitas de má conduta científica serão investigadas sob o mais alto padrão de rigor acadêmico e ético.

**Conselho Editorial**

**Ciências Humanas e Sociais Aplicadas**

Prof. Dr. Alexandre Jose Schumacher – Instituto Federal de Educação, Ciência e Tecnologia do Paraná

Prof. Dr. Américo Junior Nunes da Silva – Universidade do Estado da Bahia

Profª Drª Andréa Cristina Marques de Araújo – Universidade Fernando Pessoa

Prof. Dr. Antonio Carlos Frasson – Universidade Tecnológica Federal do Paraná

Prof. Dr. Antonio Gasparetto Júnior – Instituto Federal do Sudeste de Minas Gerais

Prof. Dr. Antonio Isidro-Filho – Universidade de Brasília  
Prof. Dr. Arnaldo Oliveira Souza Júnior – Universidade Federal do Piauí  
Prof. Dr. Carlos Antonio de Souza Moraes – Universidade Federal Fluminense  
Prof. Dr. Crisóstomo Lima do Nascimento – Universidade Federal Fluminense  
Prof<sup>a</sup> Dr<sup>a</sup> Cristina Gaió – Universidade de Lisboa  
Prof. Dr. Daniel Richard Sant’Ana – Universidade de Brasília  
Prof. Dr. Deyvison de Lima Oliveira – Universidade Federal de Rondônia  
Prof<sup>a</sup> Dr<sup>a</sup> Dilma Antunes Silva – Universidade Federal de São Paulo  
Prof. Dr. Edvaldo Antunes de Farias – Universidade Estácio de Sá  
Prof. Dr. Elson Ferreira Costa – Universidade do Estado do Pará  
Prof. Dr. Eloi Martins Senhora – Universidade Federal de Roraima  
Prof. Dr. Gustavo Henrique Cepolini Ferreira – Universidade Estadual de Montes Claros  
Prof. Dr. Humberto Costa – Universidade Federal do Paraná  
Prof<sup>a</sup> Dr<sup>a</sup> Ivone Goulart Lopes – Istituto Internazionele delle Figlie de Maria Ausiliatrice  
Prof. Dr. Jadson Correia de Oliveira – Universidade Católica do Salvador  
Prof. Dr. José Luis Montesillo-Cedillo – Universidad Autónoma del Estado de México  
Prof. Dr. Julio Candido de Meirelles Junior – Universidade Federal Fluminense  
Prof<sup>a</sup> Dr<sup>a</sup> Lina Maria Gonçalves – Universidade Federal do Tocantins  
Prof. Dr. Luis Ricardo Fernandes da Costa – Universidade Estadual de Montes Claros  
Prof<sup>a</sup> Dr<sup>a</sup> Natiéli Piovesan – Instituto Federal do Rio Grande do Norte  
Prof. Dr. Marcelo Pereira da Silva – Pontifícia Universidade Católica de Campinas  
Prof<sup>a</sup> Dr<sup>a</sup> Maria Luzia da Silva Santana – Universidade Federal de Mato Grosso do Sul  
Prof. Dr. Miguel Rodrigues Netto – Universidade do Estado de Mato Grosso  
Prof. Dr. Pablo Ricardo de Lima Falcão – Universidade de Pernambuco  
Prof<sup>a</sup> Dr<sup>a</sup> Paola Andressa Scortegagna – Universidade Estadual de Ponta Grossa  
Prof<sup>a</sup> Dr<sup>a</sup> Rita de Cássia da Silva Oliveira – Universidade Estadual de Ponta Grossa  
Prof. Dr. Rui Maia Diamantino – Universidade Salvador  
Prof. Dr. Saulo Cerqueira de Aguiar Soares – Universidade Federal do Piauí  
Prof. Dr. Urandi João Rodrigues Junior – Universidade Federal do Oeste do Pará  
Prof<sup>a</sup> Dr<sup>a</sup> Vanessa Bordin Viera – Universidade Federal de Campina Grande  
Prof<sup>a</sup> Dr<sup>a</sup> Vanessa Ribeiro Simon Cavalcanti – Universidade Católica do Salvador  
Prof. Dr. William Cleber Domingues Silva – Universidade Federal Rural do Rio de Janeiro  
Prof. Dr. Willian Douglas Guilherme – Universidade Federal do Tocantins

#### **Ciências Agrárias e Multidisciplinar**

Prof. Dr. Alexandre Igor Azevedo Pereira – Instituto Federal Goiano  
Prof. Dr. Arinaldo Pereira da Silva – Universidade Federal do Sul e Sudeste do Pará  
Prof. Dr. Antonio Pasqualetto – Pontifícia Universidade Católica de Goiás  
Prof<sup>a</sup> Dr<sup>a</sup> Carla Cristina Bauermann Brasil – Universidade Federal de Santa Maria  
Prof. Dr. Cleberton Correia Santos – Universidade Federal da Grande Dourados  
Prof<sup>a</sup> Dr<sup>a</sup> Diocléa Almeida Seabra Silva – Universidade Federal Rural da Amazônia  
Prof. Dr. Écio Souza Diniz – Universidade Federal de Viçosa  
Prof. Dr. Fábio Steiner – Universidade Estadual de Mato Grosso do Sul  
Prof. Dr. Fágner Cavalcante Patrocínio dos Santos – Universidade Federal do Ceará  
Prof<sup>a</sup> Dr<sup>a</sup> Gírlene Santos de Souza – Universidade Federal do Recôncavo da Bahia  
Prof. Dr. Jael Soares Batista – Universidade Federal Rural do Semi-Árido  
Prof. Dr. Jayme Augusto Peres – Universidade Estadual do Centro-Oeste  
Prof. Dr. Júlio César Ribeiro – Universidade Federal Rural do Rio de Janeiro  
Prof<sup>a</sup> Dr<sup>a</sup> Lina Raquel Santos Araújo – Universidade Estadual do Ceará  
Prof. Dr. Pedro Manuel Villa – Universidade Federal de Viçosa  
Prof<sup>a</sup> Dr<sup>a</sup> Raissa Rachel Salustriano da Silva Matos – Universidade Federal do Maranhão  
Prof. Dr. Ronilson Freitas de Souza – Universidade do Estado do Pará  
Prof<sup>a</sup> Dr<sup>a</sup> Talita de Santos Matos – Universidade Federal Rural do Rio de Janeiro

Prof. Dr. Tiago da Silva Teófilo – Universidade Federal Rural do Semi-Árido  
Prof. Dr. Valdemar Antonio Paffaro Junior – Universidade Federal de Alfenas

### **Ciências Biológicas e da Saúde**

Prof. Dr. André Ribeiro da Silva – Universidade de Brasília  
Profª Drª Anelise Levay Murari – Universidade Federal de Pelotas  
Prof. Dr. Benedito Rodrigues da Silva Neto – Universidade Federal de Goiás  
Profª Drª Daniela Reis Joaquim de Freitas – Universidade Federal do Piauí  
Profª Drª Débora Luana Ribeiro Pessoa – Universidade Federal do Maranhão  
Prof. Dr. Douglas Siqueira de Almeida Chaves – Universidade Federal Rural do Rio de Janeiro  
Prof. Dr. Edson da Silva – Universidade Federal dos Vales do Jequitinhonha e Mucuri  
Profª Drª Elizabeth Cordeiro Fernandes – Faculdade Integrada Medicina  
Profª Drª Eleuza Rodrigues Machado – Faculdade Anhanguera de Brasília  
Profª Drª Elane Schwinden Prudêncio – Universidade Federal de Santa Catarina  
Profª Drª Eysler Gonçalves Maia Brasil – Universidade da Integração Internacional da Lusofonia Afro-Brasileira  
Prof. Dr. Ferlando Lima Santos – Universidade Federal do Recôncavo da Bahia  
Profª Drª Fernanda Miguel de Andrade – Universidade Federal de Pernambuco  
Prof. Dr. Fernando Mendes – Instituto Politécnico de Coimbra – Escola Superior de Saúde de Coimbra  
Profª Drª Gabriela Vieira do Amaral – Universidade de Vassouras  
Prof. Dr. Gianfábio Pimentel Franco – Universidade Federal de Santa Maria  
Prof. Dr. Helio Franklin Rodrigues de Almeida – Universidade Federal de Rondônia  
Profª Drª Iara Lúcia Tescarollo – Universidade São Francisco  
Prof. Dr. Igor Luiz Vieira de Lima Santos – Universidade Federal de Campina Grande  
Prof. Dr. Jefferson Thiago Souza – Universidade Estadual do Ceará  
Prof. Dr. Jesus Rodrigues Lemos – Universidade Federal do Piauí  
Prof. Dr. Jônatas de França Barros – Universidade Federal do Rio Grande do Norte  
Prof. Dr. José Max Barbosa de Oliveira Junior – Universidade Federal do Oeste do Pará  
Prof. Dr. Luís Paulo Souza e Souza – Universidade Federal do Amazonas  
Profª Drª Magnólia de Araújo Campos – Universidade Federal de Campina Grande  
Prof. Dr. Marcus Fernando da Silva Praxedes – Universidade Federal do Recôncavo da Bahia  
Profª Drª Maria Tatiane Gonçalves Sá – Universidade do Estado do Pará  
Profª Drª Mylena Andréa Oliveira Torres – Universidade Ceuma  
Profª Drª Natiéli Piovesan – Instituto Federacl do Rio Grande do Norte  
Prof. Dr. Paulo Inada – Universidade Estadual de Maringá  
Prof. Dr. Rafael Henrique Silva – Hospital Universitário da Universidade Federal da Grande Dourados  
Profª Drª Regiane Luz Carvalho – Centro Universitário das Faculdades Associadas de Ensino  
Profª Drª Renata Mendes de Freitas – Universidade Federal de Juiz de Fora  
Profª Drª Vanessa da Fontoura Custódio Monteiro – Universidade do Vale do Sapucaí  
Profª Drª Vanessa Lima Gonçalves – Universidade Estadual de Ponta Grossa  
Profª Drª Vanessa Bordin Viera – Universidade Federal de Campina Grande  
Profª Drª Welma Emidio da Silva – Universidade Federal Rural de Pernambuco

### **Ciências Exatas e da Terra e Engenharias**

Prof. Dr. Adélio Alcino Sampaio Castro Machado – Universidade do Porto  
Profª Drª Ana Grasielle Dionísio Corrêa – Universidade Presbiteriana Mackenzie  
Prof. Dr. Carlos Eduardo Sanches de Andrade – Universidade Federal de Goiás  
Profª Drª Carmen Lúcia Voigt – Universidade Norte do Paraná  
Prof. Dr. Cleiseano Emanuel da Silva Paniagua – Instituto Federal de Educação, Ciência e Tecnologia de Goiás  
Prof. Dr. Douglas Gonçalves da Silva – Universidade Estadual do Sudoeste da Bahia  
Prof. Dr. Eloi Rufato Junior – Universidade Tecnológica Federal do Paraná  
Profª Drª Érica de Melo Azevedo – Instituto Federal do Rio de Janeiro

Prof. Dr. Fabrício Menezes Ramos – Instituto Federal do Pará  
Profª Dra. Jéssica Verger Nardeli – Universidade Estadual Paulista Júlio de Mesquita Filho  
Prof. Dr. Juliano Carlo Rufino de Freitas – Universidade Federal de Campina Grande  
Profª Drª Luciana do Nascimento Mendes – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte  
Prof. Dr. Marcelo Marques – Universidade Estadual de Maringá  
Prof. Dr. Marco Aurélio Kistemann Junior – Universidade Federal de Juiz de Fora  
Profª Drª Neiva Maria de Almeida – Universidade Federal da Paraíba  
Profª Drª Natiéli Piovesan – Instituto Federal do Rio Grande do Norte  
Profª Drª Priscila Tessmer Scaglioni – Universidade Federal de Pelotas  
Prof. Dr. Sidney Gonçalo de Lima – Universidade Federal do Piauí  
Prof. Dr. Takeshy Tachizawa – Faculdade de Campo Limpo Paulista

#### **Linguística, Letras e Artes**

Profª Drª Adriana Demite Stephani – Universidade Federal do Tocantins  
Profª Drª Angeli Rose do Nascimento – Universidade Federal do Estado do Rio de Janeiro  
Profª Drª Carolina Fernandes da Silva Mandaji – Universidade Tecnológica Federal do Paraná  
Profª Drª Denise Rocha – Universidade Federal do Ceará  
Profª Drª Edna Alencar da Silva Rivera – Instituto Federal de São Paulo  
Profª Drª Fernanda Tonelli – Instituto Federal de São Paulo,  
Prof. Dr. Fabiano Tadeu Grazioli – Universidade Regional Integrada do Alto Uruguai e das Missões  
Prof. Dr. Gilmei Fleck – Universidade Estadual do Oeste do Paraná  
Profª Drª Keyla Christina Almeida Portela – Instituto Federal de Educação, Ciência e Tecnologia do Paraná  
Profª Drª Miranilde Oliveira Neves – Instituto de Educação, Ciência e Tecnologia do Pará  
Profª Drª Sandra Regina Gardacho Pietrobon – Universidade Estadual do Centro-Oeste  
Profª Drª Sheila Marta Carregosa Rocha – Universidade do Estado da Bahia

#### **Conselho Técnico científico**

Prof. Me. Abrãao Carvalho Nogueira – Universidade Federal do Espírito Santo  
Prof. Me. Adalberto Zorzo – Centro Estadual de Educação Tecnológica Paula Souza  
Prof. Dr. Adaylson Wagner Sousa de Vasconcelos – Ordem dos Advogados do Brasil/Seccional Paraíba  
Prof. Dr. Adilson Tadeu Basquerote Silva – Universidade para o Desenvolvimento do Alto Vale do Itajaí  
Profª Ma. Adriana Regina Vettorazzi Schmitt – Instituto Federal de Santa Catarina  
Prof. Dr. Alex Luis dos Santos – Universidade Federal de Minas Gerais  
Prof. Me. Alexsandro Teixeira Ribeiro – Centro Universitário Internacional  
Profª Ma. Aline Ferreira Antunes – Universidade Federal de Goiás  
Profª Drª Amanda Vasconcelos Guimarães – Universidade Federal de Lavras  
Prof. Me. André Flávio Gonçalves Silva – Universidade Federal do Maranhão  
Profª Drª Andrezza Lopes – Instituto de Pesquisa e Desenvolvimento Acadêmico  
Profª Drª Andrezza Miguel da Silva – Faculdade da Amazônia  
Profª Ma. Anelisa Mota Gregoleti – Universidade Estadual de Maringá  
Profª Ma. Anne Karynne da Silva Barbosa – Universidade Federal do Maranhão  
Prof. Dr. Antonio Hot Pereira de Faria – Polícia Militar de Minas Gerais  
Prof. Me. Armando Dias Duarte – Universidade Federal de Pernambuco  
Profª Ma. Bianca Camargo Martins – UniCesumar  
Profª Ma. Carolina Shimomura Nanya – Universidade Federal de São Carlos  
Prof. Me. Carlos Antônio dos Santos – Universidade Federal Rural do Rio de Janeiro  
Prof. Me. Carlos Augusto Zilli – Instituto Federal de Santa Catarina  
Prof. Me. Christopher Smith Bignardi Neves – Universidade Federal do Paraná  
Profª Drª Cláudia de Araújo Marques – Faculdade de Música do Espírito Santo  
Profª Drª Cláudia Taís Siqueira Cagliari – Centro Universitário Dinâmica das Cataratas  
Prof. Me. Clécio Danilo Dias da Silva – Universidade Federal do Rio Grande do Norte  
Prof. Me. Daniel da Silva Miranda – Universidade Federal do Pará

Profª Ma. Daniela da Silva Rodrigues – Universidade de Brasília  
Profª Ma. Daniela Remião de Macedo – Universidade de Lisboa  
Profª Ma. Dayane de Melo Barros – Universidade Federal de Pernambuco  
Prof. Me. Douglas Santos Mezacas – Universidade Estadual de Goiás  
Prof. Me. Edevaldo de Castro Monteiro – Embrapa Agrobiologia  
Prof. Me. Edson Ribeiro de Britto de Almeida Junior – Universidade Estadual de Maringá  
Prof. Me. Eduardo Gomes de Oliveira – Faculdades Unificadas Doctum de Cataguases  
Prof. Me. Eduardo Henrique Ferreira – Faculdade Pitágoras de Londrina  
Prof. Dr. Edwaldo Costa – Marinha do Brasil  
Prof. Me. Eliel Constantino da Silva – Universidade Estadual Paulista Júlio de Mesquita  
Prof. Me. Ernane Rosa Martins – Instituto Federal de Educação, Ciência e Tecnologia de Goiás  
Prof. Me. Euvaldo de Sousa Costa Junior – Prefeitura Municipal de São João do Piauí  
Prof. Dr. Everaldo dos Santos Mendes – Instituto Edith Theresa Hedwing Stein  
Prof. Me. Ezequiel Martins Ferreira – Universidade Federal de Goiás  
Profª Ma. Fabiana Coelho Couto Rocha Corrêa – Centro Universitário Estácio Juiz de Fora  
Prof. Me. Fabiano Eloy Atilio Batista – Universidade Federal de Viçosa  
Prof. Me. Felipe da Costa Negrão – Universidade Federal do Amazonas  
Prof. Me. Francisco Odécio Sales – Instituto Federal do Ceará  
Prof. Me. Francisco Sérgio Lopes Vasconcelos Filho – Universidade Federal do Cariri  
Profª Drª Germana Ponce de Leon Ramírez – Centro Universitário Adventista de São Paulo  
Prof. Me. Gevair Campos – Instituto Mineiro de Agropecuária  
Prof. Me. Givanildo de Oliveira Santos – Secretaria da Educação de Goiás  
Prof. Dr. Guilherme Renato Gomes – Universidade Norte do Paraná  
Prof. Me. Gustavo Krahl – Universidade do Oeste de Santa Catarina  
Prof. Me. Helton Rangel Coutinho Junior – Tribunal de Justiça do Estado do Rio de Janeiro  
Profª Ma. Isabelle Cerqueira Sousa – Universidade de Fortaleza  
Profª Ma. Jaqueline Oliveira Rezende – Universidade Federal de Uberlândia  
Prof. Me. Javier Antonio Albornoz – University of Miami and Miami Dade College  
Prof. Me. Jhonatan da Silva Lima – Universidade Federal do Pará  
Prof. Dr. José Carlos da Silva Mendes – Instituto de Psicologia Cognitiva, Desenvolvimento Humano e Social  
Prof. Me. Jose Elyton Batista dos Santos – Universidade Federal de Sergipe  
Prof. Me. José Luiz Leonardo de Araujo Pimenta – Instituto Nacional de Investigación Agropecuaria Uruguay  
Prof. Me. José Messias Ribeiro Júnior – Instituto Federal de Educação Tecnológica de Pernambuco  
Profª Drª Juliana Santana de Curcio – Universidade Federal de Goiás  
Profª Ma. Juliana Thaisa Rodrigues Pacheco – Universidade Estadual de Ponta Grossa  
Profª Drª Kamilly Souza do Vale – Núcleo de Pesquisas Fenomenológicas/UFGA  
Prof. Dr. Kárpio Márcio de Siqueira – Universidade do Estado da Bahia  
Profª Drª Karina de Araújo Dias – Prefeitura Municipal de Florianópolis  
Prof. Dr. Lázaro Castro Silva Nascimento – Laboratório de Fenomenología & Subjetividade/UFPR  
Prof. Me. Leonardo Tullio – Universidade Estadual de Ponta Grossa  
Profª Ma. Lilian Coelho de Freitas – Instituto Federal do Pará  
Profª Ma. Lilian de Souza – Faculdade de Tecnologia de Itu  
Profª Ma. Liliani Aparecida Sereno Fontes de Medeiros – Consórcio CEDERJ  
Profª Drª Lúvia do Carmo Silva – Universidade Federal de Goiás  
Prof. Dr. Lucio Marques Vieira Souza – Secretaria de Estado da Educação, do Esporte e da Cultura de Sergipe  
Prof. Dr. Luan Vinicius Bernardelli – Universidade Estadual do Paraná  
Profª Ma. Luana Ferreira dos Santos – Universidade Estadual de Santa Cruz  
Profª Ma. Luana Vieira Toledo – Universidade Federal de Viçosa  
Prof. Me. Luis Henrique Almeida Castro – Universidade Federal da Grande Dourados  
Prof. Me. Luiz Renato da Silva Rocha – Faculdade de Música do Espírito Santo  
Profª Ma. Luma Sarai de Oliveira – Universidade Estadual de Campinas  
Prof. Dr. Michel da Costa – Universidade Metropolitana de Santos

Prof. Me. Marcelo da Fonseca Ferreira da Silva – Governo do Estado do Espírito Santo  
Prof. Dr. Marcelo Máximo Purificação – Fundação Integrada Municipal de Ensino Superior  
Prof. Me. Marcos Aurelio Alves e Silva – Instituto Federal de Educação, Ciência e Tecnologia de São Paulo  
Prof. Me. Marcos Roberto Gregolin – Agência de Desenvolvimento Regional do Extremo Oeste do Paraná  
Prof<sup>a</sup> Ma. Maria Elanny Damasceno Silva – Universidade Federal do Ceará  
Prof<sup>a</sup> Ma. Marileila Marques Toledo – Universidade Federal dos Vales do Jequitinhonha e Mucuri  
Prof. Dr. Pedro Henrique Abreu Moura – Empresa de Pesquisa Agropecuária de Minas Gerais  
Prof. Me. Pedro Panhoca da Silva – Universidade Presbiteriana Mackenzie  
Prof<sup>a</sup> Dr<sup>a</sup> Poliana Arruda Fajardo – Universidade Federal de São Carlos  
Prof. Me. Rafael Cunha Ferro – Universidade Anhembi Morumbi  
Prof. Me. Ricardo Sérgio da Silva – Universidade Federal de Pernambuco  
Prof. Me. Renan Monteiro do Nascimento – Universidade de Brasília  
Prof. Me. Renato Faria da Gama – Instituto Gama – Medicina Personalizada e Integrativa  
Prof<sup>a</sup> Ma. Renata Luciane Polsaque Young Blood – UniSecal  
Prof. Me. Robson Lucas Soares da Silva – Universidade Federal da Paraíba  
Prof. Me. Sebastião André Barbosa Junior – Universidade Federal Rural de Pernambuco  
Prof<sup>a</sup> Ma. Silene Ribeiro Miranda Barbosa – Consultoria Brasileira de Ensino, Pesquisa e Extensão  
Prof<sup>a</sup> Ma. Solange Aparecida de Souza Monteiro – Instituto Federal de São Paulo  
Prof. Dr. Sullivan Pereira Dantas – Prefeitura Municipal de Fortaleza  
Prof<sup>a</sup> Ma. Taiane Aparecida Ribeiro Nepomoceno – Universidade Estadual do Oeste do Paraná  
Prof. Me. Tallys Newton Fernandes de Matos – Universidade Estadual do Ceará  
Prof<sup>a</sup> Ma. Thatianny Jasmine Castro Martins de Carvalho – Universidade Federal do Piauí  
Prof. Me. Tiago Silvio Dedoné – Colégio ECEL Positivo  
Prof. Dr. Welleson Feitosa Gazel – Universidade Paulista



# Caderno nº1. Estatística aplicada à biologia usando o R: estatística descritiva e inferencial

**Bibliotecária:** Janaina Ramos  
**Diagramação:** Camila Alves de Cremo  
**Correção:** Mariane Aparecida Freitas  
**Edição de Arte:** Luiza Alves Batista  
**Revisão:** Os autores  
**Autores:** Elton Nunes Britto  
Damião Pedro Meira Filho  
Maria Raimunda Silva Rego

## Dados Internacionais de Catalogação na Publicação (CIP)

B862 Britto, Elton Nunes  
Caderno nº1. Estatística aplicada à biologia usando o R:  
estatística descritiva e inferencial / Elton Nunes Britto,  
Damião Pedro Meira Filho, Maria Raimunda Silva Rego.  
– Ponta Grossa - PR: Atena, 2021.

Formato: PDF

Requisitos de sistema: Adobe Acrobat Reader

Modo de acesso: World Wide Web

Inclui bibliografia

ISBN 978-65-5983-248-4

DOI: <https://doi.org/10.22533/at.ed.484210507>

1. Software R. 2. Bioestatística. 3. Biologia. 4.  
Descritiva. 5. Inferencial. I. Britto, Elton Nunes. II. Meira  
Filho, Damião Pedro. III. Rego, Maria Raimunda Silva. IV.  
Título.

CDD 001.422

Elaborado por Bibliotecária Janaina Ramos – CRB-8/9166

**Atena Editora**

Ponta Grossa – Paraná – Brasil

Telefone: +55 (42) 3323-5493

[www.atenaeditora.com.br](http://www.atenaeditora.com.br)

contato@atenaeditora.com.br

## DECLARAÇÃO DOS AUTORES

Os autores desta obra: 1. Atestam não possuir qualquer interesse comercial que constitua um conflito de interesses em relação ao artigo científico publicado; 2. Declaram que participaram ativamente da construção dos respectivos manuscritos, preferencialmente na: a) Concepção do estudo, e/ou aquisição de dados, e/ou análise e interpretação de dados; b) Elaboração do artigo ou revisão com vistas a tornar o material intelectualmente relevante; c) Aprovação final do manuscrito para submissão.; 3. Certificam que os artigos científicos publicados estão completamente isentos de dados e/ou resultados fraudulentos; 4. Confirmam a citação e a referência correta de todos os dados e de interpretações de dados de outras pesquisas; 5. Reconhecem terem informado todas as fontes de financiamento recebidas para a consecução da pesquisa; 6. Autorizam a edição da obra, que incluem os registros de ficha catalográfica, ISBN, DOI e demais indexadores, projeto visual e criação de capa, diagramação de miolo, assim como lançamento e divulgação da mesma conforme critérios da Atena Editora.

## DECLARAÇÃO DA EDITORA

A Atena Editora declara, para os devidos fins de direito, que: 1. A presente publicação constitui apenas transferência temporária dos direitos autorais, direito sobre a publicação, inclusive não constitui responsabilidade solidária na criação dos manuscritos publicados, nos termos previstos na Lei sobre direitos autorais (Lei 9610/98), no art. 184 do Código penal e no art. 927 do Código Civil; 2. Autoriza e incentiva os autores a assinarem contratos com repositórios institucionais, com fins exclusivos de divulgação da obra, desde que com o devido reconhecimento de autoria e edição e sem qualquer finalidade comercial; 3. Todos os e-book são *open access*, desta forma não os comercializa em seu site, sites parceiros, plataformas de *e-commerce*, ou qualquer outro meio virtual ou físico, portanto, está isenta de repasses de direitos autorais aos autores; 4. Todos os membros do conselho editorial são doutores e vinculados a instituições de ensino superior públicas, conforme recomendação da CAPES para obtenção do Qualis livro; 5. Não cede, comercializa ou permite a utilização dos nomes e e-mails dos autores, bem como nenhum outro dado dos mesmos, para qualquer finalidade que não o escopo da divulgação desta obra.

## SUMÁRIO

<b>O QUE É O SOFTWARE R?</b> .....	<b>1</b>
Como instalar o R? .....	1
Como criar um arquivo para ser lido no R?.....	2
Como abrir um arquivo R no console com as funções <code>setwd("...")</code> e <code>dir()</code> ? .....	2
O que são objetos e vetores? E como criar uma amostra no R? .....	3
Função <code>seq(sequence)</code> .....	5
Como criar um script no R? .....	7
Importando um arquivo em Excel usando o pacote <code>readxl</code> para ser aberto no R .....	8
Como ver uma tabela no R usando a função <code>read.table</code> e os comandos <code>print</code> e <code>view</code> no R console. ....	10
Usando o comando <code>print</code> (nome do objeto) .....	11
Como construir uma tabela no R usando a função <code>cbind</code> e a função <code>data.frame</code> ? .....	12
Como extrair valores das tabelas abertas e criadas no R? .....	13
Como extrair valores maiores e menores que determinados valores de conjuntos no R?.....	14
Extrair valores entre dois conjuntos no R?.....	15
Calculando a estatística descritiva a partir de tabelas abertas e criadas no R .....	16
Calculando as medidas de posição e dispersão com a função <code>tapply</code> .....	18
Como instalar pacotes no R com a função <code>install.packages</code> e carregar os pacotes com as funções <code>library</code> ou <code>require</code> .....	19
Usando o pacote “ <code>psych</code> ” para calcular a estatística descritiva de variáveis em tabelas e conjuntos e a função <code>describe</code> .....	20
Fazendo gráficos no R.....	24
Histograma com a função <code>hist()</code> .....	30
Como construir uma tabela de dados agrupados no R e calcular as frequências (absolutas, relativa, frequências acumuladas, absolutas e relativas)? .....	31
Testes de hipóteses para distribuição paramétrica no R .....	33
Teste de Hipótese segundo Crawley (2013) .....	34
Como visualizar a distribuição normal por histograma, uma opção visual .....	34
Como calcular a homogeneidade das variâncias e construir as hipóteses? .....	35
Transformação dos dados com as funções <code>log ()</code> e <code>sqrt ()</code> no R.....	37

Teste estatístico de distribuição paramétrica no R.....	38
Teste de hipóteses segundo Triola (2008).....	38
Teste estatístico equivalente não paramétrico no R do teste de T de Student.....	42
Teste estatístico de distribuição paramétrica no R.....	44
Como visualizar os resultados da ANOVA com a função summary .....	46
Calculando os testes <i>a posteriori</i> no R com a função TukeyHSD .....	47
Utilizando o pacote “psych” e a função describeBy() para calcular a média e o erro padrão de grupos .....	47
Teste estatístico equivalente não paramétrico da ANOVA one way no R.....	49
Realizando a comparação de medias <i>a posteriori</i> com a função dunnTest no pacote FSA.....	51
Teste estatístico de distribuição paramétrica no R.....	52
Teste estatístico equivalente não paramétrico da correlação de Pearson no R.....	56
Abordagem EPR .....	59
<b>REFERÊNCIAS .....</b>	<b>60</b>
<b>SOBRE O AUTOR.....</b>	<b>62</b>

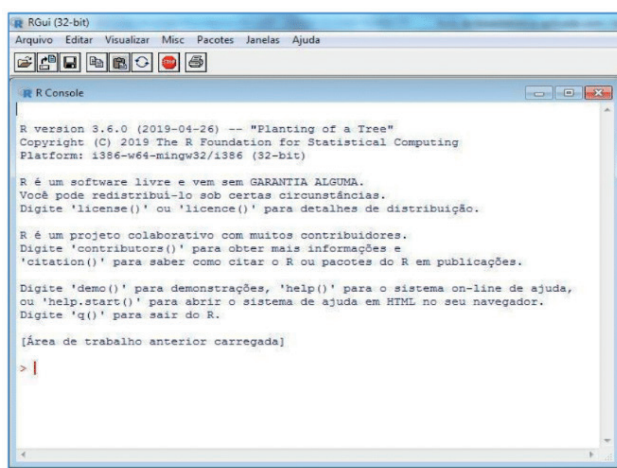
## O QUE É O SOFTWARE R?

De acordo com Da Silva et al. (2009), o R é um projeto de linguagem e ambiente computacional para estatística e está disponível como um software livre funcionando nos sistemas operacionais Linux, Windows e Macintosh. Segundo Alcoforado (2021) o software R criado por Development Core Team R (2008) compõe-se de uma coleção de quase 20 mil pacotes para a análise estatística e gráfica disponíveis através da família CRAN que podem ser baixados do site <http://www.r-project.org/web/packages/index.html>.

## COMO INSTALAR O R?

Segundo Ribeiro et al. (2009), a instalação do R é realizada pelo site [www.r-project.org](http://www.r-project.org) clicando-se em (download R) que direcionará o usuário para um espelho composto por (sites de universidades), no qual terá que ser feita uma escolha clicando-se no site. Esta escolha o levará para uma nova página onde terá que selecionar o sistema operacional (Linux, Macintosh e Windows) clicando naquele sistema instalado no seu computador. Posteriormente o usuário será redirecionado a uma nova janela e clicará em (base) que novamente o direcionará para outra janela onde estará a versão para ser instalada Download R (versão do R) (número de megabytes e bytes) ou clicando-se nela. Um detalhe importante durante à instalação é que devemos saber se nosso computador é 32-bit Windows ou 64-bit Windows. Podemos instalar em 64-bit Windows as versões 32-bit Windows e 64-bit Windows. Na versão 32-bit Windows só 32-bit. Depois se deve salvar em uma pasta ou diretório do seu computador e executá-lo, criando-se um ícone de atalha para a sua utilização.

Segundo Da Silva et al. (2009), o R abre-se diretamente em uma janela chamada R console que está pronta para execução de comandos conforme ilustra a figura 1.



```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

R version 3.6.0 (2019-04-26) -- "Planting of a Tree"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> |
```

Figura 1. R console para execução de comandos

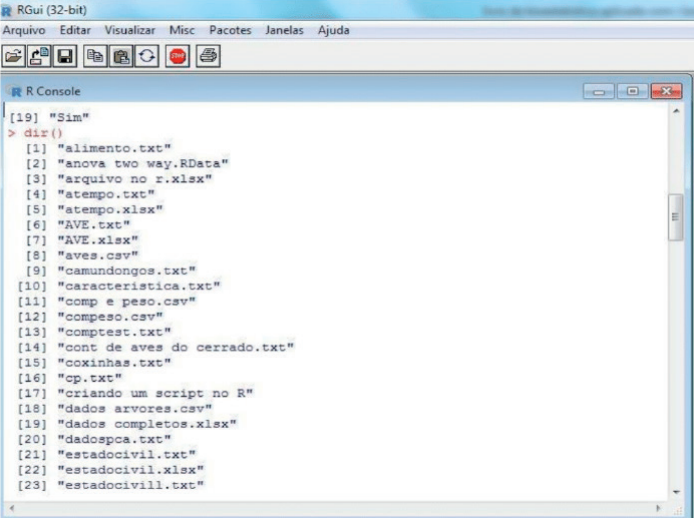
Fonte. Autores (2021)

## COMO CRIAR UM ARQUIVO PARA SER LIDO NO R?

Segundo Vital (2015) para criamos um diretório na pasta C do nosso computador e colocamos um nome, por exemplo, Pasta Elton. A pasta Elton será utilizada para salvarmos todos os arquivos para serem abertos no R. Então se tivermos uma tabela em Excel e queremos abri-la no R copiaremos a tabela e salvaremos em um arquivo do (bloco de notas) com a extensão.txt, por exemplo, alimento.txt na pasta Elton. Desta forma temos o arquivo alimento.txt para ser visualizado e aberto no R.

Para abrir este o arquivo alimento.txt a partir do R console clicaremos em (Arquivo) e depois (Mudar dir...) e será aberta uma janela pequena com o nome (procurar pasta). Clicamos em pasta C e depois em (Pasta Elton). Observação: apenas selecionaremos a Pasta Elton e clicaremos em Ok. Depois retornamos ao R console e digitamos a função `dir()` no R console e visualizaremos todos os arquivos contidos na pasta Elton (VITAL, 2015).

```
> dir ()
```



```
[19] "Sim"  
> dir()  
[1] "alimento.txt"  
[2] "anova two way.RData"  
[3] "arquivo no r.xlsx"  
[4] "atempo.txt"  
[5] "atempo.xlsx"  
[6] "AVE.txt"  
[7] "AVE.xlsx"  
[8] "aves.csv"  
[9] "camundongos.txt"  
[10] "caracteristica.txt"  
[11] "comp e peso.csv"  
[12] "compeso.csv"  
[13] "comptest.txt"  
[14] "cont de aves do cerrado.txt"  
[15] "coxinhas.txt"  
[16] "cp.txt"  
[17] "criando um script no R"  
[18] "dados arvores.csv"  
[19] "dados completos.xlsx"  
[20] "dadospca.txt"  
[21] "estadocivil.txt"  
[22] "estadocivil.xlsx"  
[23] "estadocivil.txt"
```

Figura 2. R console para execução de comandos

Fonte. Autores (2021)

## COMO ABRIR UM ARQUIVO R NO CONSOLE COM AS FUNÇÕES SETWD(“”) E DIR()?

Apartir do nosso computador, apenas abrimos a pasta Elton e na barra de ferramentas clicamos com o botão esquerdo do mouse e selecionamos os caminhos e com o botão direito copiamos o caminho até a pasta Elton (VITAL, 2015). No R console digitaremos segundo Souza et al. (2018) a função `setwd(“ ”)` e salvamos dentro dos parênteses o caminho e entre aspas. No nosso caso, o caminho copiado do computador foi: C:\2021 o que estou usando\Projeto de extensão com R\Pasta Elton. Atenção, para que funcione a

função `setwd(" ")` devemos salvar o caminho acima e mudarmos as posições das barras para a direita assim como nas figuras 3 e 4.

```
> setwd("C:/2021 o que estou usando/Projeto de extensão com R/Pasta Elton")
```

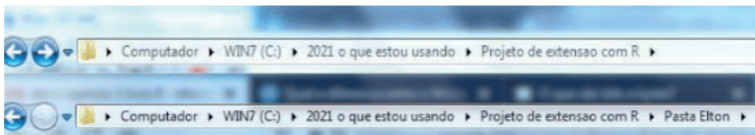


Figura 3. Visualizando a pasta R (não selecionada); selecionando a pasta R (cor azul) na barra de ferramentas.

Fonte. Autores (2021)

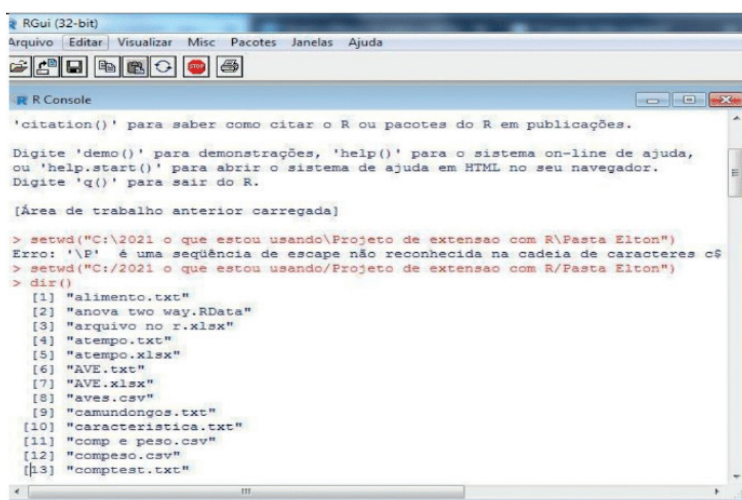


Figura 4. A função `setwd()` no R console.

Fonte. Autores (2021)

Depois de clicamos enter e após digitamos `dir()` (o comando no R console para a execução das funções é o enter (SOUZA ET AL., 2018).

## O QUE SÃO OBJETOS E VETORES? E COMO CRIAR UMA AMOSTRA NO R?

Segundo Souza et al. (2018) objetos no R são criados para o armazenamento de dados. Os dados que podem ser guardados em objetos variam desde numéricos, alfanuméricos (palavras), lógicos, e também podem ser armazenadas tabelas como `data.frames` e matrizes e funções. Aos objetos que memorizam informações numéricas são chamados de vetores. De acordo com estes autores os nomes de objetos podem ser letras ou palavras, preferencialmente opta-se por letras, que podem ser minúsculas ou maiúsculas. O nome de um objeto não pode ter espaços entre ele se for uma palavra,



porém pode ter símbolos como (=, . e\_) e a denominação de um objeto não pode ser um símbolo matemático (+, -, ±, \*, /). Devemos atentar para não errar no R memorizando sempre o nome dos objetos criados, visto que, se o objeto foi criado em letra minúscula e for digitado em letra maiúscula é gerado um erro no R. Em concordância com os autores supracitados o R é “case sensitive”, isto é, difere letras maiúsculas de minúsculas.

A criação de um objeto envolve a utilização de um símbolo (<-) a direita do objeto que significa (cópia e salva). Para criarmos um conjunto necessitaremos da função *c* de (concatenate) além de darmos um nome a ele. Por exemplo, queremos criar o conjunto com *a* com os valores (2, 3 e 4) devemos digitar no R console *a <- c(2,3,4)* (SOUZA ET AL.,2018).

```
a<-c(2, 3, 4)
[1] 2 3 4
```

O resultado acima indica que os valores do conjunto *c* (2,3,4) foram copiados e salvados no objeto *a*. A resposta no R console após o enter é gerada com o colchete e o número 1 e à direita, os valores. Este conjunto criado pode ser, desta forma, a nossa amostra ou população.

## Vetores

Em consonância com os autores Souza et al. (2018) os vetores são considerados são formas elementares de memorizar dados possuindo uma única dimensão e pertencendo ao único tipo ou classe. Uma forma de reconhecer que classe pertence um vetor podemos digitar a função *class* (nome do objeto) clicando enter.

```
> a<-c(2, 3, 4)
> class(a)
```

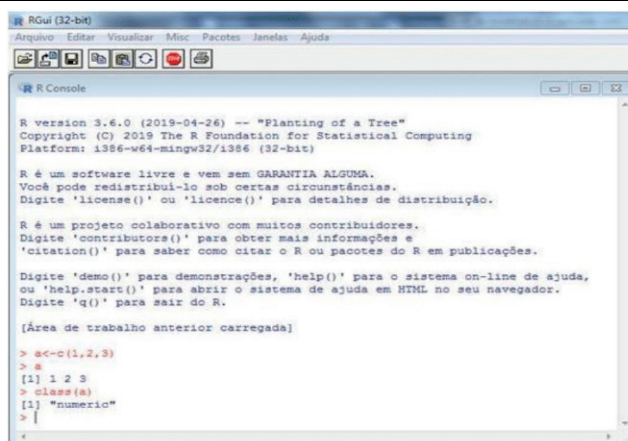


Figura 5. Verificamos o objeto *a* e a função *class(a)* denominando que ele é formado por números

Fonte. Autores (2021)

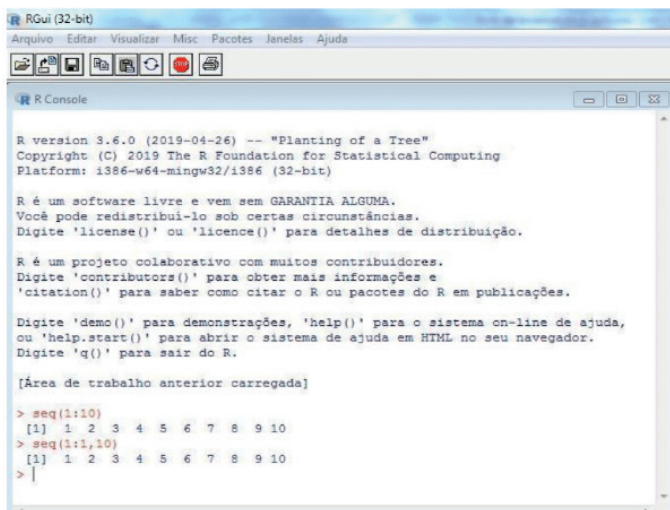
Existem outros tipos de funções para criarmos conjuntos ou vetores no R. Temos objetos criados com as funções *c*(concatenate) escritos como *c* (); *seq* (sequence), *seq* () e *rep* (replicate), *rep* (). A função *c* () tem o objetivo de criarmos conjuntos numéricos ou alfanuméricos (SOUZA ET AL., 2018).

```
> a<-c(2, 3, 4)
[1] 2 3 4
> b<-("Manaus", "Pará", "Rio Grande do Sul")
[1] "Manaus" "Pará" "Rio Grande do Sul"
```

## FUNÇÃO SEQ(SEQUENCE)

A função *seq* () tem o objetivo de criar uma sequência de números contidos em intervalos conhecidos variando de um limite inferior até um limite superior (SOUZA ET AL., 2018).

```
> seq(from=1, to= 10, by= 1)
> seq(1:10)
> seq(1:1,10)
```



The screenshot shows the RGui (32-bit) interface with the R Console window open. The console displays the following text:

```
R version 3.6.0 (2019-04-26) -- "Planting of a Tree"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> seq(1:10)
[1] 1 2 3 4 5 6 7 8 9 10
> seq(1:1,10)
[1] 1 2 3 4 5 6 7 8 9 10
> |
```

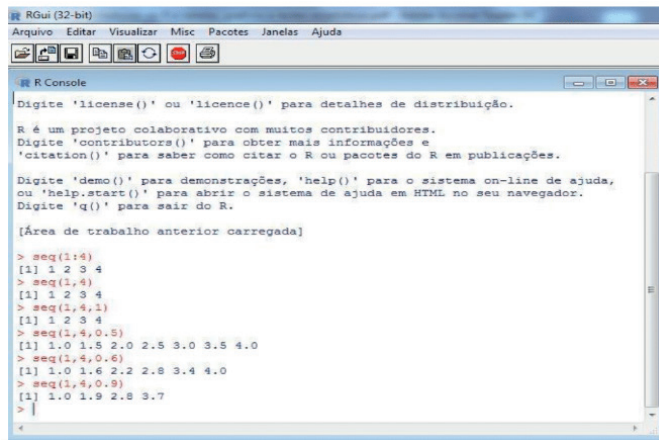
Figura 6: Função sequência

Fonte. Autores (2021)

Nossa explicação para a sequência *seq* (1:10) é 1 1+1 2+1 3+1 4+1 5+1 6+1 7+1 8+1 9+1. Por sua vez, na sequência *seq* (1,4,2) é 1 1+2.

Em geral, as sequências geram números acumulados de um limite inferior até um superior não ultrapassando este, ou dizendo de outra forma INCLUI o limite inferior e EXCLUI o último valor se ele ultrapassar o limite superior ou INCLUI o último valor se ele for o limite superior (MARTINS & DONAIRE, 1990). Observe as saídas das sequências *seq*

(1,4,0.5), *seq*(1,4,0.6) e *seq*(1,4,0.9) ilustradas na figura 7.



```
RGUí (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console
Digite 'license()' ou 'licence()' para detalhes de distribuição.
R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.
Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> seq(1:4)
[1] 1 2 3 4
> seq(1,4)
[1] 1 2 3 4
> seq(1,4,1)
[1] 1 2 3 4
> seq(1,4,0.5)
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0
> seq(1,4,0.6)
[1] 1.0 1.6 2.2 2.8 3.4 4.0
> seq(1,4,0.9)
[1] 1.0 1.9 2.8 3.7
> |
```

Figura 7: Exemplos da função *seq* (sequence)

Fonte. Autores (2021)

Na prática, a explicação para a saída dada pelo R para a expressão do quadro abaixo é  $1 + 0.9$   $1.9 + 0.9$   $2.8 + 0.9$ . Observe que  $3.7 + 0.9$  não existe porque ultrapassa o limite superior 4.

```
> seq(1,4,0.9)
[1] 1 1.9 2.8 3.7
```

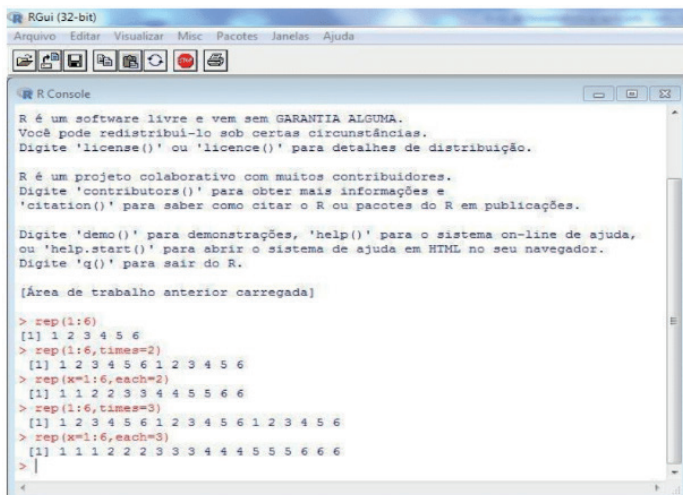
Novamente, segundo Souza et al. (2018) temos *rep*(1:4,2) que repete a sequência de 1 a 4 duas vezes. Nesse caso, a função é *rep*(1:6, times=2).

```
> rep(1:6, 2)
[1] 1 2 3 4 5 6 1 2 3 4 5 6
> rep(1:6, times = 2)
[1] 1 2 3 4 5 6 1 2 3 4 5 6
```

Também se pode ter a repetição de 1 a 6 repetindo cada valor duas vezes com a função *for rep*(1:6, each=2).

```
> rep(x=1:6, each= 2)
```

Verificamos na figura 4 as repetições: *rep*(1:6), *rep*(1:6, times= 2), *rep*(x=1:6, each=2) e *rep*(x=1:6, each=3) na figura 8.



```
RGui (32-bit)
Arquivo  Editor  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licenca()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> rep(1:6)
[1] 1 2 3 4 5 6
> rep(1:6,times=2)
[1] 1 2 3 4 5 6 1 2 3 4 5 6
> rep(x=1:6,each=2)
[1] 1 1 2 2 3 3 4 4 5 5 6 6
> rep(1:6,times=3)
[1] 1 2 3 4 5 6 1 2 3 4 5 6 1 2 3 4 5 6
> rep(x=1:6,each=3)
[1] 1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6
> |
```

Figura 8: Usando a função repetição no R console

Fonte. Autores (2021)

## COMO CRIAR UM SCRIPT NO R?

Para criar um script no R basta clicar em arquivo – novo script – e depois salvar. Os scripts são salvos na pasta onde são armazenados os arquivos e são arquivos que executam os cálculos no R console (VITAL, 2015). Segundo Ribeiro et al. (2009) o script é um arquivo tipo texto que pode ser executado no R console bastando para isto posicionar o cursor a direita do comando que se quer e clicar control+R. Fazendo isto a operação aparece em vermelho no R console e é executada conforme a figura 9. Na figura 10 executamos a operação de soma pela função *sum* (nome do vetor) segundo Souza et al. (2018) para fazer esta operação pode-se selecionar tudo de uma única vez. A seleção no script é feita com clicando-se com o botar esquerdo do mouse.

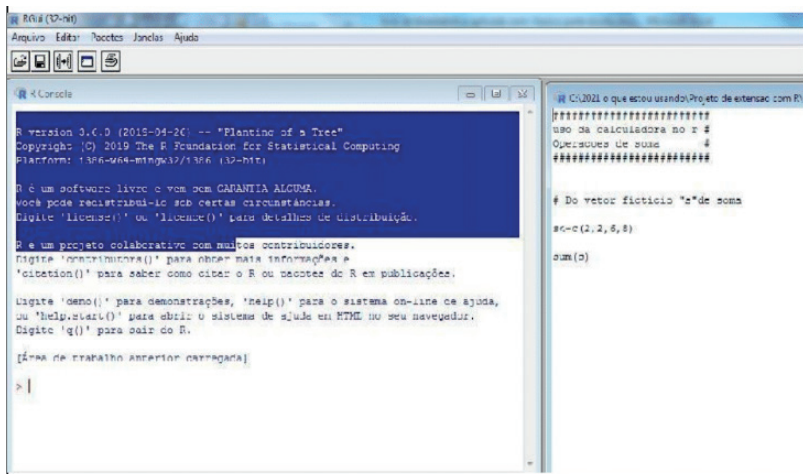


Figura 9: Usando a função repetição no R console

Fonte. Autores (2021)

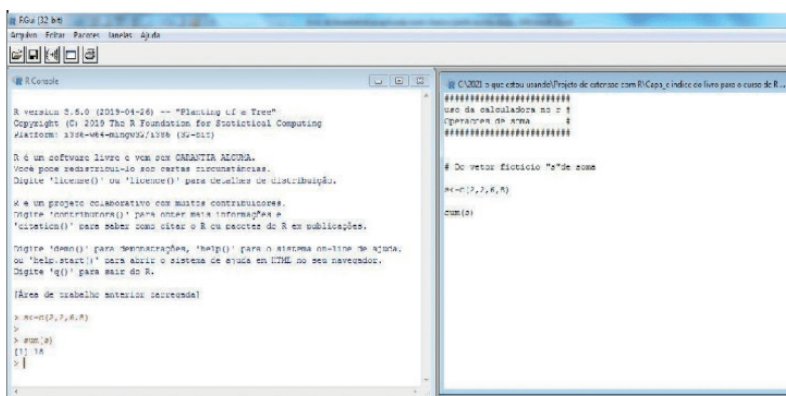


Figura 10: Operação do script criado sendo executada no R console

Fonte. Autores (2021)

## IMPORTANDO UM ARQUIVO EM EXCEL USANDO O PACOTE READXL PARA SER ABERTO NO R

Para abrir um arquivo em Excel usando o pacote readxl criado por Hadley & Jennifer (2019) é necessário instalar o pacote readxl conforme o procedimento citado por Vicentini & Perdiz (2020). A operação é feita no R console digitando o comando `install.packages("readxl")` e clicando enter. Realizada a instalação do pacote temos que carregá-lo no R. No R console usamos o comando `library("readxl")` ou `require("readxl")` clicando enter. Precisamos fazer duas operações antes de ler o arquivo em Excel no R que descreveremos por primeira e segunda operações (ver figura 11).

Operação "1ª": digitaremos no R console um objeto que iremos copiar e salvar o

nosso arquivo em Excel e teclaremos *read\_excel* (“nome do arquivo em Excel”) e enter.

```
> b<-read_excel("alturapeso.xlsx")
```

Operação “2ª”: digitaremos no R console outro nome de objeto para copiar e salvar a função *as.data.frame* (nome do objeto que contem o arquivo em Excel) e enter.

```
> c<-as.data.frame(b)
```



```
RGui (32-bit) - [R Console]
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda

> dir()
[1] "ABNT+NR+6023+2018+(1).pdf"
[2] "alturapeso.xlsx"
[3] "Introdução à Bioestatística-3ª edição-Sonia Vieira.pdf"
[4] "liv de bioestat aplic com r material raiz"
[5] "parte escrita por mim"
[6] "partes corrigidas pela Maria"
[7] "pesoidade.csv"
[8] "pesoidade.xls"
[9] "pesoidade1.xlsx"
[10] "referencia bibliograficas usadas"
[11] "script da soma.R"
> install.packages("readxl")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
tentando a URL 'https://vps.fmvz.usp.br/CRAN/bin/windows/contrib/3.6/readxl_1.3.1.zip'
Content type 'application/zip' length 1526130 bytes (1.5 MB)
downloaded 1.5 MB

package 'readxl' successfully unpacked and MD5 sums checked

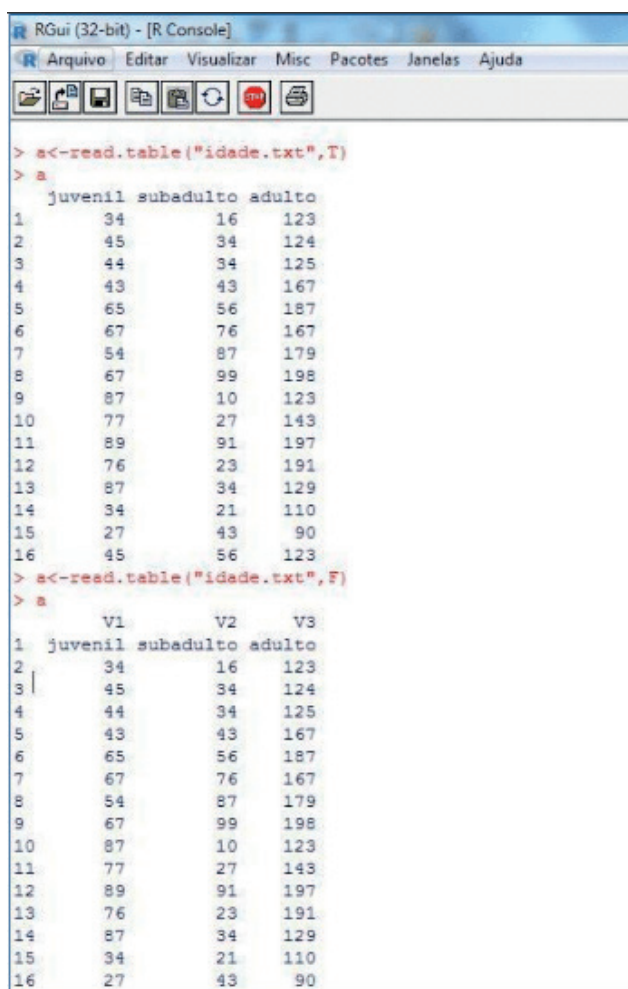
The downloaded binary packages are in
  C:\Users\ELTON\AppData\Local\Temp\RtmpSACjud\downloaded_packages
> require("readxl")
Carregando pacotes exigidos: readxl
Warning message:
package 'readxl' was built under R version 3.6.3
> x<-read_excel("alturapeso.xlsx")
> v<-as.data.frame(x)
> v
  altura peso
1   178   65
2   189   68
3   185   84
4   165   59
5   187   65
6   174   80
```

Figura 11: Instalando, carregando o pacote “readxl” para abrir um arquivo em Excel no R console.

Fonte. Autores (2021)

## COMO VER UMA TABELA NO R USANDO A FUNÇÃO READ.TABLE E OS COMANDOS PRINT E VIEW NO R CONSOLE

Após acessarmos o diretório que é a pasta onde salvamos os nossos arquivos no computador com o R console aberto, utilizaremos a função `read.table` (“nome do arquivo”.txt, T) onde temos o argumento “nome do arquivo” salvo em .txt (salvo no bloco de notas) e seguido da letra “T” que é a abreviatura de TRUE. Com esta função podemos abrir uma tabela no R que foi salva em .txt e é possível ver o nome das colunas. Se optarmos pelo FALSE, cada primeira linha será nomeada com a letra “V” sendo V1, V2 e etc. como variável 1, variável 2 respectivamente (Figura 12). Conforme Ribeiro et al. (2009) a função `read.table` ( ) serve para importarmos arquivos criados em formato de texto para ser aberto no R console.



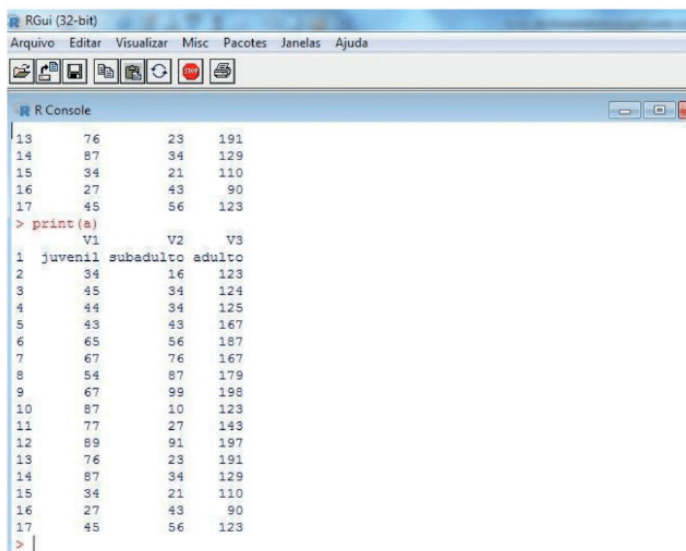
```
> a<-read.table("idade.txt",T)
> a
  juvenil subadulto adulto
1      34         16   123
2      45         34   124
3      44         34   125
4      43         43   167
5      65         56   187
6      67         76   167
7      54         87   179
8      67         99   198
9      87         10   123
10     77         27   143
11     89         91   197
12     76         23   191
13     87         34   129
14     34         21   110
15     27         43    90
16     45         56   123
> a<-read.table("idade.txt",F)
> a
      V1      V2      V3
1 juvenil subadulto adulto
2      34      16     123
3      45      34     124
4      44      34     125
5      43      43     167
6      65      56     187
7      67      76     167
8      54      87     179
9      67      99     198
10     87      10     123
11     77      27     143
12     89      91     197
13     76      23     191
14     87      34     129
15     34      21     110
16     27      43      90
```

Figura 12: Usando a função `read.table` no R console com o argumento T=TRUE e F=FALSE.

Fonte. Autores (2021)

## USANDO O COMANDO PRINT (NOME DO OBJETO)

A partir de um objeto que contém uma tabela com a função *print* podemos abrir a tabela no R console, sendo uma operação equivalente ao digitarmos o nome do objeto que contém a tabela e clicar “enter”. No exemplo com o mesmo objeto chamado de “a” usaremos `print(a) + enter` (vide figura 13) e para vermos a tabela com linhas e colunas, o comando `View(a) + enter`. Conforme Crawley (2013) o comando `print` exibe os resultados no R console.



```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

13      76      23      191
14      87      34      129
15      34      21      110
16      27      43      90
17      45      56      123
> print(a)
  V1      V2      V3
1 juvenil subadulto adulto
2      34      16      123
3      45      34      124
4      44      34      125
5      43      43      167
6      65      56      187
7      67      76      167
8      54      87      179
9      67      99      198
10     87      10      123
11     77      27      143
12     89      91      197
13     76      23      191
14     87      34      129
15     34      21      110
16     27      43      90
17     45      56      123
> |
```

Figura 13: Utilizando a função `print` para visualizar a tabela no R console

Fonte. Autores (2021)



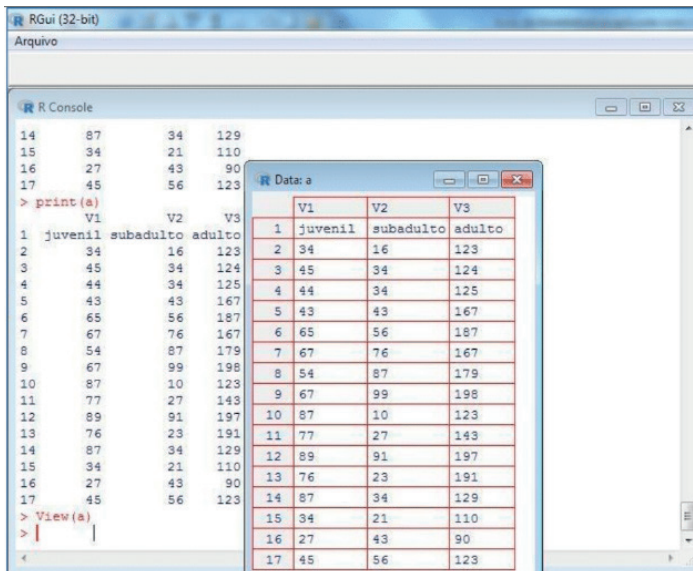


Figura 14: Visualizando a tabela no R console com a função View (nome do objeto)

Fonte. Autores (2021)

## COMO CONSTRUIR UMA TABELA NO R USANDO A FUNÇÃO CBIND E A FUNÇÃO DATA.FRAME?

Segundo Souza et al. (2018) com a função *cbind* (objeto1, objeto2 e etc...) construiremos uma matriz de dados combinando os objetos e conforme Ribeiro et al. (2009) com a função *data.frame* (objeto1, objeto2 e etc...) construiremos uma tabela de dados com os objetos que conterà linhas e colunas onde a informação da primeira linha e demais colunas equivalem a um indivíduo (vide figura 15). De acordo com Ribeiro et al. (2009), uma matriz é um conjunto de objetos numéricos de mesmo tamanho que são organizados lado a lado contendo linhas e colunas.

```

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> comprimento<-c(170,189,145,154)
> peso<-c(54,89,66,89)
> cbind(comprimento,peso)
  comprimento peso
[1,]         170  54
[2,]         189  89
[3,]         145  66
[4,]         154  89
> data.frame(comprimento,peso)
  comprimento peso
1           170  54
2           189  89
3           145  66
4           154  89

```

Figura 15: Usando as funções cbind e data.frame no R console

Fonte. Autores (2021)

## COMO EXTRAIR VALORES DAS TABELAS ABERTAS E CRIADAS NO R?

Utilizando a tabela “comptest.txt” da figura 16 iremos extrair linhas e colunas. Usaremos os colchetes os colchetes [ ] onde toda a informação dentro dos colchetes que está à esquerda da vírgula refere-se à linha e aquela à direita, refere-se à coluna (SOUZA ET AL., 2018). Desta forma após um objeto seguido dos colchetes as informações serão extraídas conforme sua posição em relação à vírgula. Neste caso copiamos e salvamos as variáveis em objetos para realizar a extração (RIBEIRO ET AL., 2009).

Na figura 16 estão os resultados dos comandos abaixo:

- A primeira coluna: a[1] ou a[,1]
- A segunda coluna: a[2] ou a[,2]
- O objeto da primeira linha e da primeira coluna: a[1,1]
- O objeto da primeira linha e da segunda coluna: a[1,2]
- O objeto da segunda linha e da primeira coluna: a[2,1]
- O objeto da segunda linha e da segunda coluna: a[2,2]
- As duas primeiras linhas da coluna 1: a[c(1,2),1]
- As duas primeiras linhas da coluna 2: a[c(1,2),2]

- As linhas 2 e 5 da coluna 1: `a[c(2,5),1]`
- As linhas de 2 a 5 da coluna 1: `a[c(2:5),1]`
- As linhas 1 e 3 da coluna 2: `a[c(1,3),2]`
- Os objetos das linhas 3 e 5 das primeira e segunda colunas: `a[c(3,5), c(1,2)]`

```

> a
  compr peso
1     45   74
2     42   59
3     56   85
4     52   89
5     62   96
> a[1,]
  compr
1     45
2     42
3     56
4     52
5     62
> a[,1]
[1] 45 42 56 52 62
> a[2,]
  peso
1     74
2     59
3     85
4     89
5     96
> a[,2]
[1] 74 59 85 89 96
> a[1,1]
[1] 45
> a[1,2]
[1] 74
> a[2,1]
[1] 42
> a[2,2]
[1] 59
> a[c(1,2),1]
[1] 45 42
> a[c(1,2),2]
[1] 74 59
> a[c(2,5),1]
[1] 42 62
> a[c(2:5),1]
[1] 42 56 52 62

```

Figura 16: Utilização dos colchetes para extrair informações de uma tabela importada no R console.

Fonte. Autores (2021)

## COMO EXTRAIR VALORES MAIORES E MENORES QUE DETERMINADOS VALORES DE CONJUNTOS NO R?

De acordo com Ribeiro et al. (2009) a extração de valores de conjuntos é possível através do uso dos colchetes. Transformamos os valores da tabela em conjuntos e usamos o comando `nome do objeto [nome do objeto > valor citado]` para extrair valores maiores que o citado. Para valores menores faz-se substituindo pelo sinal de menor (vide Figura 18).

```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de a:
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.

[Área de trabalho anterior carregada]

> c(145,154,156,178,189)
[1] 145 154 156 178 189
> altura<-c(145,154,156,178,189)
> altura
[1] 145 154 156 178 189
> sort(altura)
[1] 145 154 156 178 189
> altura[altura<160]
[1] 145 154 156
> altura[altura>170]
[1] 178 189
> |

> a[c(1,3),2]
[1] 74 85
> a[c(3,5),c(1,2)]
  compr peso
3     56  85
5     62  96
> a[c(1,2),2]
[1] 74 59
```

Figura 18: Utilização dos colchetes para extrair informações de uma tabela importada no R console.

Fonte. Autores (2021)

## EXTRAIR VALORES ENTRE DOIS CONJUNTOS NO R?

Para extrairmos valores entre dois conjuntos usamos o comando “&”, por exemplo: nome do objeto [variável < valor citado & variável < valor citado] (RIBEIRO ET AL., 2009). Na figura 19 queremos extrair valores de comprimentos menores que 170 cm e pesos menores que 80 Kg.

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

> read.table("tabela comp e peso.txt",T)
  comp. peso
1    123   45
2    145   67
3    156   54
4    167   87
5    145   87
6    178   87
7    178   87
8    178   87
9    189   98
10   190   76
11   190   65
12   190   65

> a<-read.table("tabela comp e peso.txt",T)
> a
  comp. peso
1    123   45
2    145   67
3    156   54
4    167   87
5    145   87
6    178   87
7    178   87
8    178   87
9    189   98
10   190   76
11   190   65
12   190   65

> comprimento<-a$comp.
> comprimento
 [1] 123 145 156 167 145 178 178 178 189 190 190 190

> peso<-a$peso
> peso
 [1] 45 67 54 87 87 87 87 87 98 76 65 65

> # extrair comprimentos menores que 170 e pesos menores que 80
> comprimento[comprimento<170 & peso<80]
 [1] 123 145 156

```

Figura 19: Extração de valores entre conjuntos no R.

Fonte. Autores (2021)

## CALCULANDO A ESTATÍSTICA DESCRITIVA A PARTIR DE TABELAS ABERTAS E CRIADAS NO R

De acordo com Da Silva et al. (2009) as medidas de posição e de dispersão e os cálculos dos valores mínimos, máximos, (n) amostral, raiz quadrada e coeficiente de variação são calculados conforme os comandos e formulações apresentadas na figura 20.

```

> a<-read.table("tabela comp e peso.txt",I)
> a
  comp. peso
1    123  45
2    145  67
3    156  54
4    167  87
5    145  87
6    178  87
7    178  87
8    178  87
9    189  98
10   190  76
11   190  65
12   190  65
> # queremos calcular as medidas de posicao e dispersao no x
> a$comp.
[1] 123 145 156 167 145 178 178 178 189 190 190 190
> comprimento<-a$comp.
> comprimento
[1] 123 145 156 167 145 178 178 178 189 190 190 190
> peso<-a$peso
> peso
[1] 45 67 54 87 87 87 87 87 98 76 65 65
> # calculo da media no x
> # media de comprimento
> mean(comprimento)
[1] 169.0833
> # variancia
> var(comprimento)
[1] 491.5379
> #desvio padrao
> sd(comprimento)
[1] 22.17065
> # n amostral
> length(comprimento)
[1] 12
> # rol em ordem crescente no x
> sort(comprimento)
[1] 123 145 145 156 167 178 178 178 189 190 190 190
> # mediana
> median(comprimento)

```

Figura 20: Cálculo das medidas de posição e dispersão no R rol, média, mediana, variância, desvio padrão, raiz quadrada, erro padrão, mínimo, máximo, n amostral.

Fonte. Autores (2021)

```

| > # valor maximo
| > max(comprimento)
| [1] 190
| > # valor minimo
| > min(comprimento)
| [1] 123
| > # erro padrao
| > # ep<-sd/sqrt(length)pela formula
| > dv<-sd(comprimento)
| > dv
| [1] 22.17065
| > raiz<-sqrt(comprimento)
| > raiz
| [1] 11.09054 12.04159 12.49000 12.92285 12.04159 13.34166 13.34166 13.34166
| [9] 13.74773 13.78405 13.78405 13.78405 13.78405
| > # corrigindo e raiz no n amostral
| > nr<-sqrt(comprimento)
| > # corrigindo novamente
| > n<-length(comprimento)
| > n
| [1] 12
| > # raiz de n abreviaremos como rn
| > rn<-sqrt(n)
| > rn
| [1] 3.464102
| > # agora o erro padrao sera aplicando-se a formula ep<-dv/rn
| > ep<-dv/rn
| > ep
| [1] 6.400116
| > # ep<-dv/rn para o alfa de 5% multiplicando-se por z=1,96
| > # ep*1.96
| > ep*1.96
| [1] 12.54423
| > # coeficiente de variacao
| > # usaremos o desvio padrao que criamos o dv
| > # a media sera copiada e salva no objeto m
| > m<-mean(comprimento)
| > m
| [1] 169.0833
| > # o coeficiente de variacao sera calculado pela formula
| > # cv<-100*dv/m
| > cv
| Erro: objeto 'cv' não encontrado
| > # corrigindo porque o cv foi criado apos a cerquilha, e tudo que esta apos a $
| > cv<-100*dv/m
| > cv
| [1] 13.11226

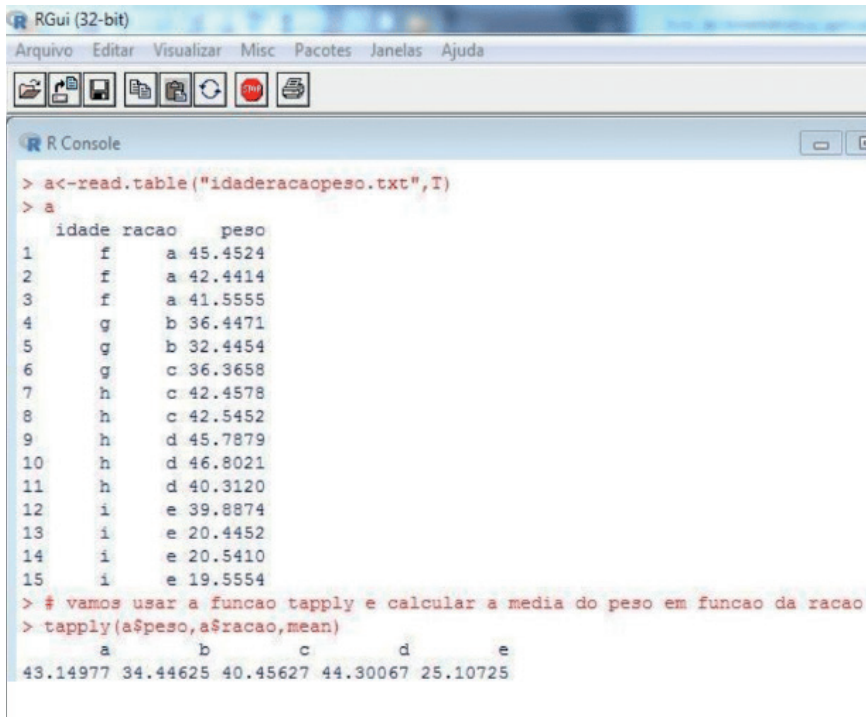
```

Figura 20: Cálculo das medidas de posição e dispersão no R rol, media, mediana, variância, desvio padrão, raiz quadrada, erro padrão, mínimo, máximo, n amostral.

Fonte. Autores (2021)

## CALCULANDO AS MEDIDAS DE POSIÇÃO E DISPERSÃO COM A FUNÇÃO TAPPLY

O cálculo as medidas de posição e de dispersão pode ser realizado com a função `tapply`. O comando para a função `tapply` é `tapply` (variável resposta, tratamento, função), clicando-se enter posteriormente (Figura 21) (CRAWLEY, 2013).



```
> a<-read.table("idaderacaopeso.txt",T)
> a
  idade racao  peso
1     f     a 45.4524
2     f     a 42.4414
3     f     a 41.5555
4     g     b 36.4471
5     g     b 32.4454
6     g     c 36.3658
7     h     c 42.4578
8     h     c 42.5452
9     h     d 45.7879
10    h     d 46.8021
11    h     d 40.3120
12    i     e 39.8874
13    i     e 20.4452
14    i     e 20.5410
15    i     e 19.5554

> # vamos usar a funcao tapply e calcular a media do peso em funcao da racao
> tapply(a$peso,a$racao,mean)
      a      b      c      d      e
43.14977 34.44625 40.45627 44.30067 25.10725
```

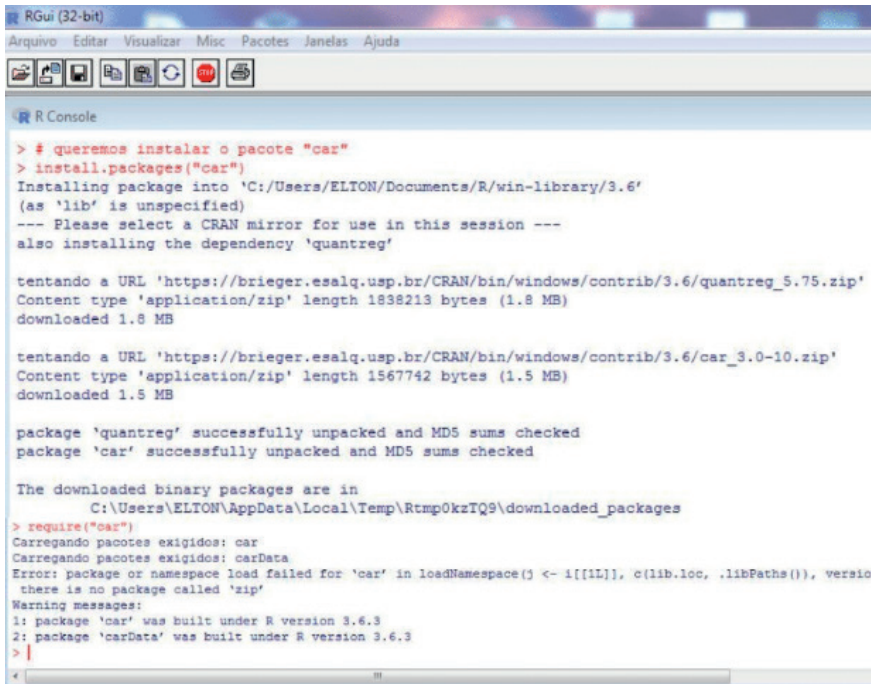
Figura 21: Cálculo das medidas de posição e de dispersão usando a função tapply

Fonte. Autores (2021)

## COMO INSTALAR PACOTES NO R COM A FUNÇÃO INSTALL.PACKAGES E CARREGAR OS PACOTES COM AS FUNÇÕES LIBRARY OU REQUIRE

Segundo Dalgaard (2008) para instalarmos um pacote no R (ver figura 22) utilizaremos o comando `install.packages` (“nome do pacote”) e para carregarmos o pacote, o comando `library` (“nome do pacote”) ou `require` (“nome do pacote”). Conforme Dalgaard (2008) é necessário instalar e carregar os pacotes no R cada vez que abrirmos um novo arquivo.





```
> # queremos instalar o pacote "car"
> install.packages("car")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
also installing the dependency 'quantreg'

tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/quantreg_5.75.zip'
Content type 'application/zip' length 1838213 bytes (1.8 MB)
downloaded 1.8 MB

tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/car_3.0-10.zip'
Content type 'application/zip' length 1567742 bytes (1.5 MB)
downloaded 1.5 MB

package 'quantreg' successfully unpacked and MD5 sums checked
package 'car' successfully unpacked and MD5 sums checked

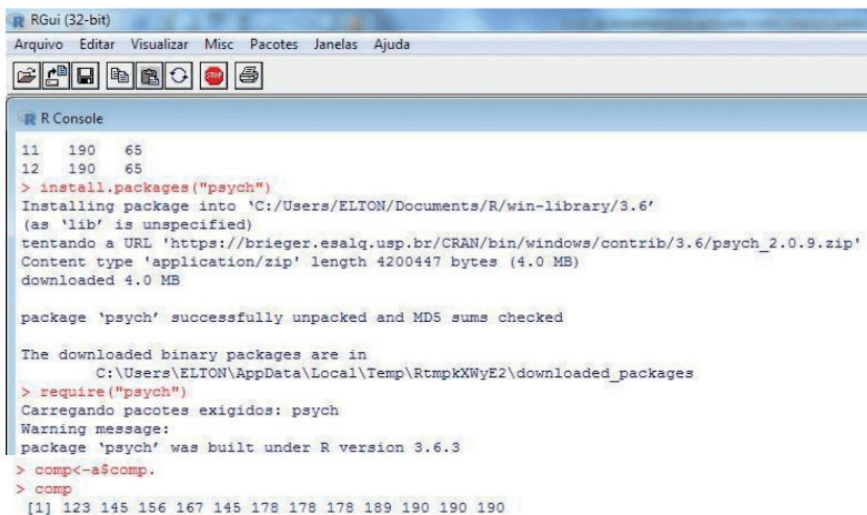
The downloaded binary packages are in
  C:\Users\ELTON\AppData\Local\Temp\Rtmp0kzTQ9\downloaded_packages
> require("car")
Carregando pacotes exigidos: car
Carregando pacotes exigidos: carData
Error: package or namespace load failed for 'car' in loadNamespace() c(lib.loc, .libPaths()), version
there is no package called 'zip'
Warning messages:
1: package 'car' was built under R version 3.6.3
2: package 'carData' was built under R version 3.6.3
> |
```

Figura 22: Como instalar e carregar pacotes no R

Fonte. Autores (2021)

## USANDO O PACOTE “PSYCH” PARA CALCULAR A ESTATÍSTICA DESCRITIVA DE VARIÁVEIS EM TABELAS E CONJUNTOS E A FUNÇÃO DESCRIBE

O pacote “psych” criado por Revelle (2020) permite o cálculo dos principais indicadores estatísticos, como “n” amostral ou “N” populacional, média aritmética, das medidas de dispersão (amplitude, desvio padrão e erro padrão) além dos valores mínimos e máximos. Depois de carregar o pacote no R utilizamos a função *describe* (nome do objeto que contém a tabela) do pacote psych e clicamos enter para termos o cálculo automático das estatísticas descritivas como na figura 23 (BRANCO & HARNIK, 2017). De acordo com Centeno (1990) o erro padrão é calculado através da divisão do desvio padrão pela raiz quadrada do “n” amostral e posteriormente pela multiplicação pelo valor Z conforme  $\alpha$ .



```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console
11  190  65
12  190  65
> install.packages("psych")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/psych_2.0.9.zip'
Content type 'application/zip' length 4200447 bytes (4.0 MB)
downloaded 4.0 MB

package 'psych' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\ELTON\AppData\Local\Temp\RtmpkXWY2\downloaded_packages
> require("psych")
Carregando pacotes exigidos: psych
Warning message:
package 'psych' was built under R version 3.6.3
> comp<-a$comp.
> comp
[1] 123 145 156 167 145 178 178 178 189 190 190 190
```

Figura 23: Instalação, carregamento do pacote “psych” e o uso da função describe para obtenção da estatística descritiva de variáveis em tabelas e conjuntos

Fonte. Autores (2021)

Podemos calcular a estatística descritiva utilizando o pacote “psych” e a função describeBy (nome do objeto, group=nome do objeto\$variavel citada) (BRANCO & HARNIK, 2017) conforme a figura 24.

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda
[Icons]

R Console
> a<-read.table("profissoes.txt",T)
> a
      sexo  religiao profissao anos peso altura
1  casado  catolico professor   21  50   147
2  casado  catolico professor   24  56   148
3  casado  catolico professor   25  52   145
4  casado  catolico professor   63  89   147
5  casado  catolico professor   36  78   148
6  casado  espirita professor   45  89   149
7  casado  espirita professor   41  85   175
8  solteiro espirita  lojista   44  85   182
9  solteiro espirita  lojista   45  85   178
10 solteiro espirita  lojista   45  85   177
11 solteiro espirita  lojista   16  85   157
12 solteiro espirita  lojista   14  85   165
13 solteiro espirita  lojista   10  88   198
14 solteiro evangelico lojista   78  88   199
15 separado evangelico lojista   85  89   147
16 separado evangelico operario  14  65   145
17 separado evangelico operario  45  63   135
18 separado evangelico operario  42  65   152
19 separado evangelico operario  47  66   150
20 separado evangelico operario  85  63   180
21 separado evangelico operario  65  63   158

> ac<-read.table("tabela comp e peso.txt",T)
> a
      comp. peso
1    123    45
2    145    67
3    156    54
4    167    87
5    145    87
6    178    87
7    178    87
8    178    87
9    189    98
10   190    76
11   190    65
12   190    65

> describe(a)
      vars  n  mean  sd median trimmed  mad min max range  skew kurtosis  se
comp.  1 12 169.08 22.17 178.0 171.6 17.79 123 190 67 -0.69 -0.84 6.40
peso   2 12  75.42 16.10  81.5  76.2 14.83  45  98  53 -0.43 -1.21 4.65

```

Figura 23: Usando o pacote “psych” e a função describe para calcular a estatística descritiva no R

Fonte. Autores (2021)

```

> # install.packages("psych")
> install.packages("psych")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/psych'
Content type 'application/zip' length 4200447 bytes (4.0 MB)
downloaded 4.0 MB

package 'psych' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:/Users/ELTON/AppData/Local/Temp/RtmpIrl1e3/downloaded_packages
> require("psych")
Carregando pacotes exigidos: psych
Warning message:
package 'psych' was built under R version 3.6.3
> # usando o pacote psych e a funcao "describeBy" para a fazer a estatistica dos
> describeBy(a,group=a$profissao)

Descriptive statistics by group
group: lojista

```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew
sexo*	1	8	2.88	0.35	3.0	2.88	0.00	2	3	1	-1.86
religiao*	2	8	2.25	0.46	2.0	2.25	0.00	2	3	1	0.95
profissao*	3	8	1.00	0.00	1.0	1.00	0.00	1	1	0	NaN
anos	4	8	42.12	28.41	44.5	42.12	43.74	10	85	75	0.28
peso	5	8	86.25	1.75	85.0	86.25	0.00	85	89	4	0.51
altura	6	8	175.38	18.42	177.5	175.38	24.46	147	199	52	-0.10

```

      kurtosis      se
sexo*          1.70  0.12
religiao*     -1.21  0.16
profissao*      NaN  0.00
anos          -1.62 10.08
peso          -1.83  0.62
altura        -1.53  6.51
-----
group: operario

```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis
sexo*	1	6	2.00	0.00	2	2.00	0.00	2	2	0	NaN	NaN
religiao*	2	6	3.00	0.00	3	3.00	0.00	3	3	0	NaN	NaN
profissao*	3	6	2.00	0.00	2	2.00	0.00	2	2	0	NaN	NaN
anos	4	6	49.67	23.85	46	49.67	17.05	14	85	71	0.02	-1.33
peso	5	6	64.17	1.33	64	64.17	1.48	63	66	3	0.18	-2.05
altura	6	6	153.33	15.17	151	153.33	9.64	135	180	45	0.59	-1.04

```

      se
sexo*    0.00
religiao* 0.00
profissao* 0.00
anos     9.74
peso     0.54
altura   6.19
-----

```

Figura 24: Usando o pacote “psych” e a função describeBy para calcular a estatística descritiva de grupos no R

Fonte: Autores (2021)

```

group: professor
      vars n   mean    sd median trimmed   mad min max range skew
sexo*   1 7   1.00  0.00      1   1.00  0.00  1  1    0  NaN
religiao* 2 7   1.29  0.49      1   1.29  0.00  1  2    1  0.75
profissao* 3 7   3.00  0.00      3   3.00  0.00  3  3    0  NaN
anos     4 7  36.43 14.85     36  36.43 16.31 21  63   42  0.54
peso     5 7  71.29 17.89     78  71.29 16.31 50  89   39 -0.16
altura   6 7 151.29 10.53    148 151.29  1.48 145 175   30  1.57

      kurtosis   se
sexo*           NaN 0.00
religiao*     -1.60 0.18
profissao*           NaN 0.00
anos          -1.23 5.61
peso          -2.12 6.76
altura         0.70 3.98
> |

```

Figura 24: Usando o pacote “psych” e a função describeBy para calcular a estatística descritiva de grupos no R

Fonte. Autores (2021)

Usando o pacote “modeest” e a função mlv (nome do objeto method = “mfv”) para o cálculo da moda em tabelas e conjuntos.

Podemos calcular a moda ou o valor mais frequente para tabelas e grupos através do pacote “modeest” criado por Poncet (2019) usa-se a função *mlv* (nome do objeto, method = “mfv”) e conforme ilustra a figura 25 (SOUZA ET AL. 2018).

```

> mlv(comp,method="mfv")
[1] 178 190
> mlv(a$peso,method="mfv")
[1] 87

```

Figura 25: Cálculo da moda para variáveis de tabelas ou grupos com o pacote “modeest” e a função mlv

Fonte. Autores (2021)

## FAZENDO GRÁFICOS NO R

### Gráfico de dispersão de pontos em conjuntos no R com a função plot

Para fazermos gráficos no R com a função plot segundo Souza et al. (2018) iremos considerar a amostra A (ver figura 26) com dados de peso e comprimento. O peso (68,69,74,59,53,46) e o comprimento (178, 189,171,165, 185,174), onde o y é o peso (Kg) e o x o comprimento (cm). Usaremos a função plot (nome do objeto no eixo do x, nome do objeto no eixo do y, xlab = “comprimento”, ylab = “peso”) para gerar o gráfico de dispersão de pontos (ver figura 27). Os argumentos da função plot xlab e ylab fornecem o nome para

cada eixo.

```
> # conjunto de peso (68,69,74,59,53,46)
> # conjunto de comprimento (178, 189,171,165, 185,174)
> peso<-c(68,69,74,59,56,46)
> peso
[1] 68 69 74 59 56 46
> length(peso)
[1] 6
> comprimento<-c(178,189,171,165,185,174)
> comprimento
[1] 178 189 171 165 185 174
> length(comprimento)
[1] 6
> plot(comprimento,peso,xlab="Comprimento",ylab="Peso")
```

Figura 26: Valores de peso e comprimento para a plotagem do gráfico de dispersão de pontos no R .

Fonte. Autores (2021)

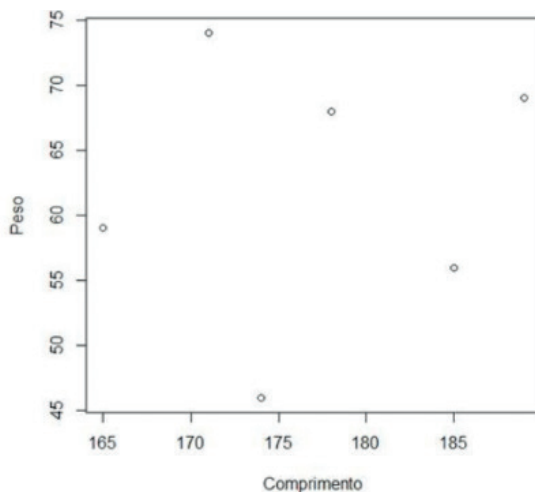
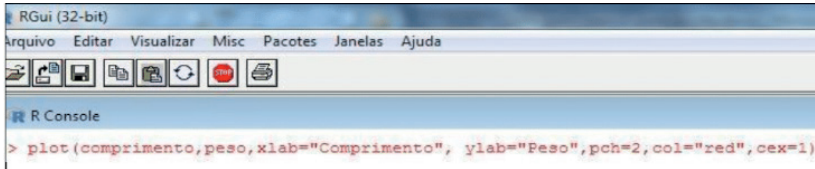


Figura 27: Gráfico de dispersão de pontos com a função plot no R.

Fonte. Autores (2021)

Conforme Souza et al. (2018) podemos acrescentar outros argumentos para variar a forma, a cor e o tamanho do ponto que é formado pelo par  $(x,y)$ . Estes argumentos são opcionais. Os argumentos são adicionados entre vírgulas. Para variar a forma usaremos o argumento,  $pch$  = número. Na cor usaremos ( $col$  = "nome da cor") e para o tamanho,  $cex$  número. Para um gráfico onde o ponto é simbolizado por um triangulo, a cor do ponto é vermelha e o tamanho é 4 (a função plot ficará conforme a figura 28). O gráfico com estas características será visualizado na figura 29.



```
RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda
R Console
> plot(comprimento,peso,xlab="Comprimento", ylab="Peso",pch=2,col="red",cex=1)
```

Figura 28: A função plot para o gráfico de dispersão de pontos com variação nos argumentos de tipo, cor e tamanho de pontos.

Fonte. Autores (2021)

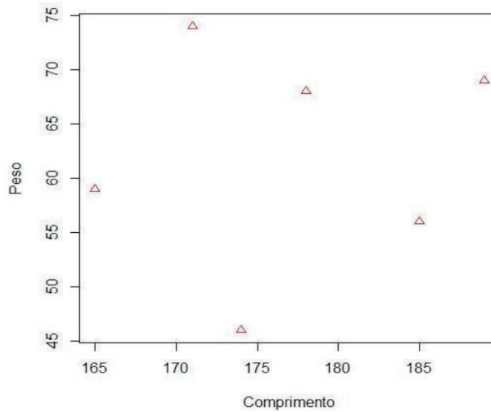


Figura 29: Gráfico de dispersão de pontos com variação nos argumentos de tipo, cor e tamanho de pontos

Fonte. Autores (2021)

Conforme Souza et al. (2018) os padrões de pontos com o argumento *pch* varia segundo a figura 30.



Figura 30: Padrões de pontos com o argumento pch no R

Fonte. Autores (2021)

De acordo com Crawley (2013) o software R apresenta um quantitativo de 657 cores. Para descobrir qual é cada tipo de cor digita-se no R console a função *colors* () e para que uma das cores possa ser utilizada no gráfico basta acrescentar ao argumento da função *plot* () denominado *col* o nome da cor entre aspas. Quanto ao tamanho, usa-se o

argumento `cex` = número do tamanho do ponto. O `cex` tem como padrão geral o número 1, representando 100% do tamanho, de forma que 0.5 representara 50% do tamanho e 1.5 consiste em 150% do tamanho.

## Gráfico de linhas em conjuntos no R com a função `plot`

Conforme Souza et al. (2018) para utilizar o gráfico de linhas adiciona-se ao argumento da função `plot` o argumento denominado `type` "l" que representa a presença da linha no gráfico. As linhas segundo os autores supracitados variam segundo outro argumento, o `lty` número. O número de variações são de 0 até 6, sendo o zero representando a ausência de linhas e de 2 a 6 tipos variados de linhas. A função `plot` com os argumentos para o gráfico de linhas é `plot` (`comprimento`, `peso`, `xlab` = "Comprimento", `ylab` = "Peso", `col` = "red", `cex` = 1, `lty` = 2, `type` = "l"). O gráfico de linhas gerado pela função e argumento está na figura 31.

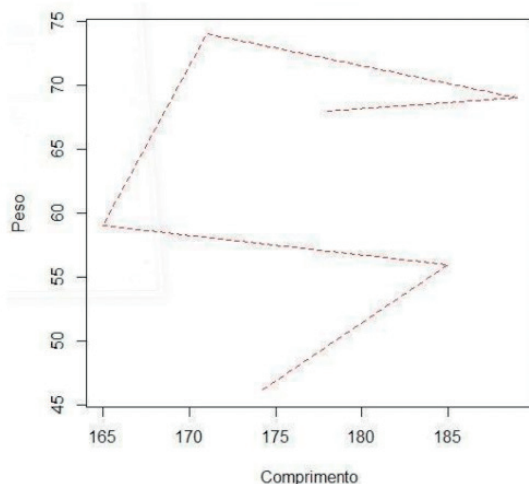


Figura 31: Gráfico de linhas no R com a variável x o comprimento e a variável y o peso

Fonte. Autores (2021)

## Gráfico de barras em conjuntos no R com a função `barplot` com o pacote `ggplot2`

Conforme Martins & Alcoforado (2018) para criarmos o gráfico de barras é necessário obtermos o somatório de cada coluna com a função `sum` (`nome do objeto$nome da variável`) que soma valores de uma coluna ou de um conjunto. Posteriormente criaremos um conjunto com os valores dos somatórios e associaremos com a função `names` (`nome do objeto`) os nomes das colunas com o conjunto com os somatórios conforme os passos exibidos na figura 32.

- 1º passo: fazendo os somatórios com a função `sum` (`nome do objeto$variável`) para cada coluna.



```
> sum(nome do objeto$nome da variável)
```

- 2º passo: construindo um conjunto com os somatórios de cada coluna.

```
> dados<-c( )
```

- 3º passo: associando o nome aos valores do conjunto numérico com os somatórios.

```
> names(dados)<-c("nome da coluna 1","nome da coluna2"," nome da coluna 3")
```

Seguindo Crawley (2013), pode-se instalar e carregar o R com o pacote ggplot2 criado por Wickham (2016) e depois construir o gráfico com a programação como na figura 33.

```
> barplot (dados, col=" nome da cor das colunas", horiz=F, ylim=c (valor mínimo para representar as colunas, valor máximo para representar as colunas))
```

Dos argumentos: o *col* indicara a cor das colunas, *horiz* indicara com o comando F de FALSE onde as barras ficarão dispostas na vertical; o *xlab* e *ylab* informarão a legenda do eixo x e y (respectivamente) e o *ylim* delimitara o eixo do y sendo no conjunto *c* (limite inferior do eixo dos y, limite superior do eixo do y).

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

[148] "varios dados.txt"
> a<-read.table("estadocivill.txt",T)
> a
  casado solteiro divorciado
1     45         41         70
2     12         41         87
3     45         45         77
4     42         42         72
5     42         41         49
6     42         42         72
7     42         41         49
8     42         42         72
9     42         41         49
> sum(a$casado)
[1] 354
> sum(a$solteiro)
[1] 376
> sum(a$divorciado)
[1] 597
> dados<-c(354,376,597)
> nomes(dados)<-c("casado", "solteiro", "divorciado")
> dados
  casado  solteiro divorciado
    354     376     597
> # instalando o pacote ggplot2
> install.packages("ggplot2")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/ggplot2_3.6.3.zip'
Content type 'application/zip' length 4069458 bytes (3.9 MB)
downloaded 3.9 MB

package 'ggplot2' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\ELTON\AppData\Local\Temp\RtmpWcML2T\downloaded_packages
> require("ggplot2")
Carregando pacotes exigidos: ggplot2
Warning message:
package 'ggplot2' was built under R version 3.6.3
> barplot(dados,col="red",horiz=F,xlab="Estado Civil",ylab="Contagem Total",ylim=c(0,600))

```

Figura 31: Programação para a construção do gráfico de barras construído com a função `barplot` do pacote `ggplot2`.

Fonte. Autores (2021)

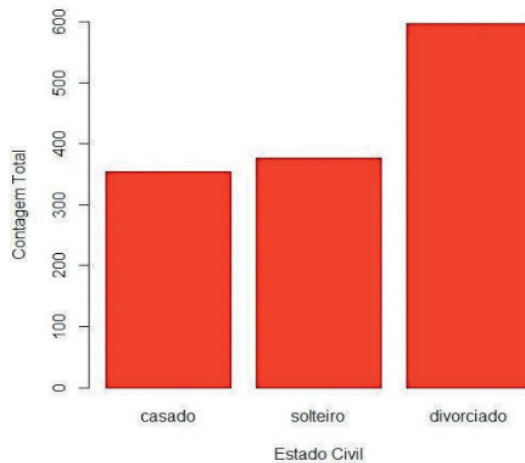


Figura 32: Gráfico de barras construído com a função `barplot` do pacote `ggplot2`  
 Fonte. Autores (2021)

## HISTOGRAMA COM A FUNÇÃO `HIST()`

Para fazermos um histograma utilizaremos a função `hist` (nome do objeto\$variável, `xlab`= “legenda do eixo do x”, `ylab`= “legenda do eixo do y”, `col`= “cor das colunas”) (RIBEIRO ET AL., 2009). A figura 33 exhibe a programação do histograma e a figura 34 o histograma.

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console
> a<-read.table("estadocivill.txt" ,T)
> a
  casado solteiro divorciado
1     45      41         70
2     12      41         87
3     45      45         77
4     42      42         72
5     42      41         49
6     42      42         72
7     42      41         49
8     42      42         72
9     42      41         49
> # fazendo um histograma com a funcao hist() no R
> # A frequencia da Contagem Total do grupo divorciado
> hist(a$divorciado,main="Histograma",xlab="Grupo dos divorciados",ylab="Frequencia Absoluta",s
  
```

Figura 33: Programação do gráfico histograma com a função `hist` no R  
 Fonte. Autores (2021)

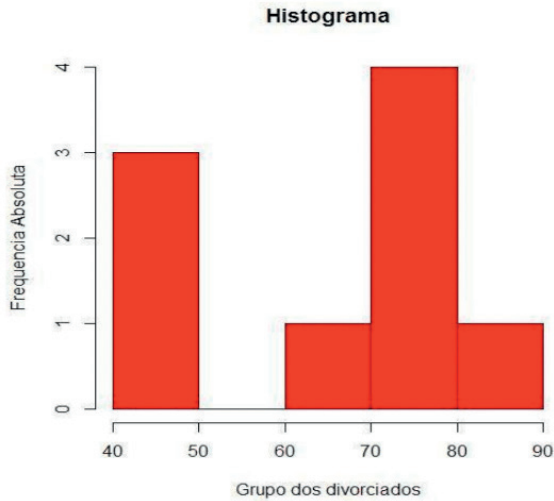


Figura 34: Gráfico histograma no R

Fonte. Autores (2021)

## COMO CONSTRUIR UMA TABELA DE DADOS AGRUPADOS NO R E CALCULAR AS FREQUÊNCIAS (ABSOLUTAS, RELATIVA, FREQUÊNCIAS ACUMULADAS, ABSOLUTAS E RELATIVAS)?

Conforme Jelihovschi (2014) podemos construir uma tabela de dados agrupados e calcular as frequências absolutas, relativas, acumuladas absolutas e relativas aplicando o pacote “fdth” criado por Faria et al. (2019) usando a função *fdt* (). Especificamente, a função *fdt* apresenta os argumentos *fdt* (nome do objeto\$variável citada, start = limite inferior da primeira classe, end = limite superior da última classe, h = amplitude da classe). Podemos ver os comandos na figura 35 e a tabela na figura 36.

```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

[146] "twowayrep.txt"
[147] "twowayrep.xlsx"
[148] "varios dados.txt"
> a<-read.table("tabela comp e peso.txt",T)
> a
  comp. peso
1   123   45
2   145   67
3   156   54
4   167   87
5   145   87
6   178   87
7   178   87
8   178   87
9   189   98
10  190   76
11  190   65
12  190   65

> install.packages("fdth")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/fdth_5
Content type 'application/zip' length 293571 bytes (286 KB)
downloaded 286 KB

package 'fdth' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Users\ELTON\AppData\Local\Temp\RtmpAdgvqg\downloaded_packages
> require("fdth")
Carregando pacotes exigidos: fdth

Attaching package: 'fdth'

The following objects are masked from 'package:stats':

  sd, var

Warning message:
package 'fdth' was built under R version 3.6.3
```

Figura 35: Instalação do pacote “fdth” e os argumentos da função fdt para a construção da tabela de distribuição de frequências no R

Fonte. Autores (2021)

```

> a
  comp. peso
1    123   45
2    145   67
3    156   54
4    167   87
5    145   87
6    178   87
7    178   87
8    178   87
9    189   98
10   190   76
11   190   65
12   190   65
> sort(a$peso)
[1] 45 54 65 65 67 76 87 87 87 87 87 98
> |
> # vamos construir uma tabela de distribuicao
> # usaremos o grupo de peso
> t<-fdt(a$peso,start=40,end=100,h=10)
> t
  Class limits f   rf rf(%) cf  cf(%)
  [40,50) 1 0.08  8.33  1   8.33
  [50,60) 1 0.08  8.33  2  16.67
  [60,70) 3 0.25 25.00  5  41.67
  [70,80) 1 0.08  8.33  6  50.00
  [80,90) 5 0.42 41.67 11  91.67
  [90,100) 1 0.08  8.33 12 100.00

```

Figura 36: Instalação do pacote “fdth” e os argumentos da função *fdt* para a construção da tabela de distribuição de frequências no R

Fonte. Autores (2021)

Conforme a tabela de dados agrupados gerada na figura 36 pela função *fdt* (), a tabela foi salva e copiada no objeto *t* e o class limits representa que o intervalo de classe tem 10 unidades e inclui ou conta-se o limite inferior e exclui ou não o limite superior. Ainda nos resultados exibidos, temos *f* é frequência absoluta, *rf* é a proporção relativa, *rf* corresponde à frequência relativa em porcentagem, *cf* é a frequência acumulada absoluta crescente e *cf* indica a frequência acumulada relativa crescente.

## TESTES DE HIPÓTESES PARA DISTRIBUIÇÃO PARAMÉTRICA NO R

### Como calcular a premissa da normalidade e construir as hipóteses?

De acordo com Ayres et al. (2003) o teste de Shapiro-Wilk serve para detectar a normalidade e pode ser utilizado para “n” amostrais maiores que 2 e menores que 51. Segundo Crawley (2013) esse teste utiliza a função *shapiro.test* (nome do objeto \$ variável resposta) conforme a figura 37.

```

> a
  comp. peso
1    123   45
2    145   67
3    156   54
4    167   87
5    145   87
6    178   87
7    178   87
8    178   87
9    189   98
10   190   76
11   190   65
12   190   65

> shapiro.test(a$peso)

      Shapiro-Wilk normality test

data:  a$peso
W = 0.90549, p-value = 0.1867

```

Figura 37: Cálculo da normalidade com o shapiro.test () no R

Fonte. Autores (2021)

O resultado do teste é interpretado pela comparação do  $\alpha$  teste que geralmente é 5% e que fornece um  $p$ -valor=0,05 com o  $p$ -valor do teste. Se  $p$ -valor do teste  $>0.05$  não rejeitamos a hipótese  $h_0$ , porém se o  $p$ -valor for  $<0.05$  rejeitamos a hipótese  $h_0$  (VIEIRA, 2010).

## TESTE DE HIPÓTESE SEGUNDO CRAWLEY (2013)

- $H_0$ : há distribuição normal nos valores de peso;
- $H_1$ : não há distribuição normal nos valores de peso.

Na figura 37 o  $p$ -valor do teste foi  $p$ -valor0.1867 comparado ao  $\alpha = 5\%$ , de modo que a distribuição dos valores de peso tem distribuição normal porque o  $p$ -valor 0,1836 é maior que o  $p$ -valor 0,05.

## COMO VISUALIZAR A DISTRIBUIÇÃO NORMAL POR HISTOGRAMA, UMA OPÇÃO VISUAL

Podemos fazer o gráfico de histograma com a função `hist ()` conforme a figura 38 onde constam os comandos para tal propósito. A figura 39 mostra o gráfico de histograma.

```

RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda

R Console
> a<-read.table("tabela comp e peso.txt",T)
> a
  comp. peso
1    123   45
2    145   67
3    156   54
4    167   87
5    145   87
6    178   87
7    178   87
8    178   87
9    189   98
10   190   76
11   190   65
12   190   65

> hist(a$peso,main="Histograma",xlab="Peso Total",ylab="Frequencia Absoluta",col="red")

```

Figura 38: Tabela e a função hist () com os argumentos para o histograma avaliado tem distribuição normal

Fonte. Autores (2021)

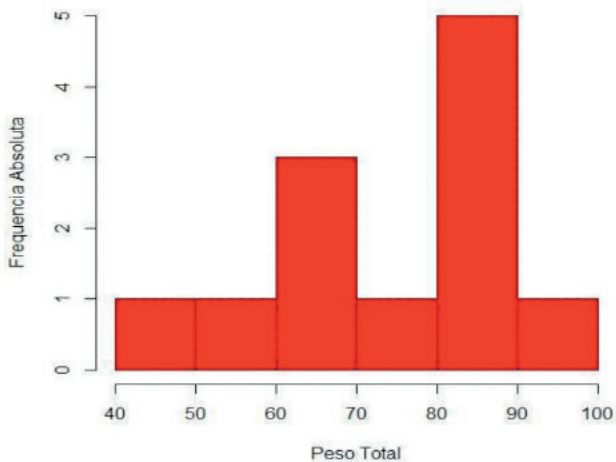


Figura 39: Histograma para avaliação da distribuição normal

Fonte. Autores (2021)

## COMO CALCULAR A HOMOGENEIDADE DAS VARIÂNCIAS E CONSTRUIR AS HIPÓTESES?

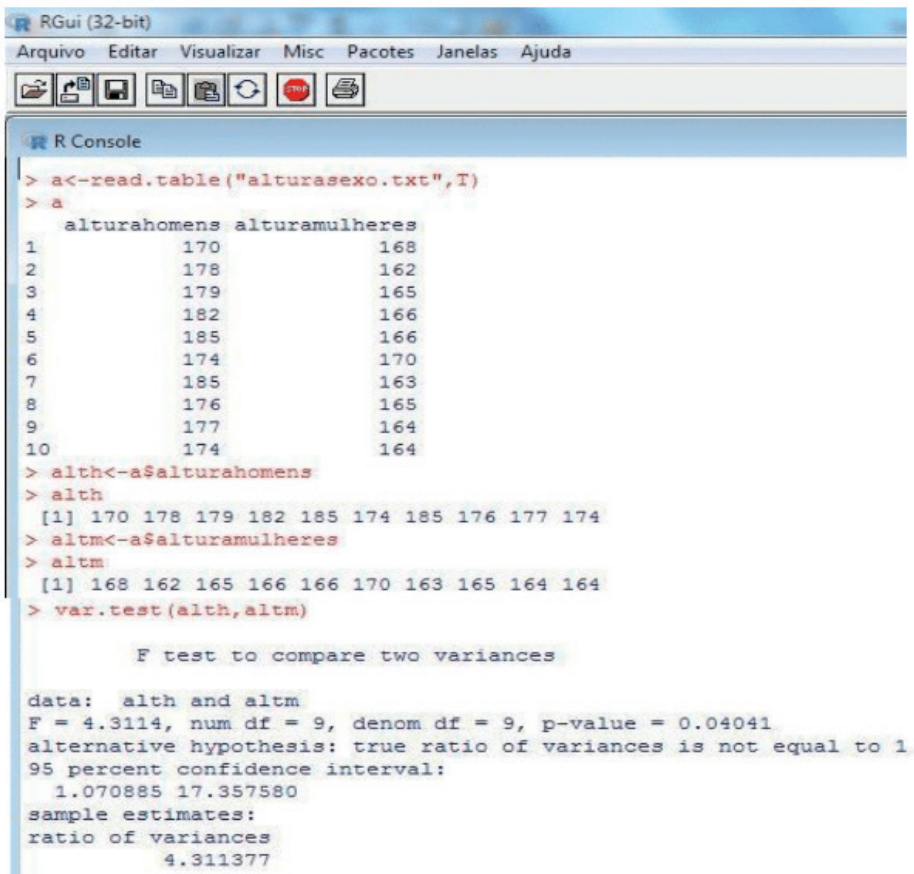
Para o teste de homogeneidade das variâncias amostrais, calcula-se a diferença entre duas variâncias mediante a função *var.test* (nome do objeto1, nome do objeto2) (BEASLEY, 2004). A decisão é feita com base no teste de hipóteses comparando-se *p*-valor e o  $\alpha$  crítico com o *p*-valor dos dados observados (VIEIRA, 2010).



Tomada da decisão pelo teste de hipóteses pela função `var.test()` segundo Crawley (2013):

- $H_0$ : não há diferença entre as variâncias amostrais ou as variâncias são independentes;
- $H_1$ : há diferença entre as variâncias amostrais ou as variâncias são dependentes;

Na figura 40, constam a tabela e a programação relativas à função `var.test()` e o resultado do teste de homogeneidade das variâncias.



```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console
> a<-read.table("alturasexo.txt",T)
> a
  alturahomens alturamulheres
1           170             168
2           178             162
3           179             165
4           182             166
5           185             166
6           174             170
7           185             163
8           176             165
9           177             164
10          174             164
> alth<-a$alturahomens
> alth
 [1] 170 178 179 182 185 174 185 176 177 174
> altm<-a$alturamulheres
> altm
 [1] 168 162 165 166 166 170 163 165 164 164
> var.test(alth,altm)

      F test to compare two variances

data:  alth and altm
F = 4.3114, num df = 9, denom df = 9, p-value = 0.04041
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 1.070885 17.357580
sample estimates:
ratio of variances
 4.311377
```

Figura 40: Comparação entre a altura de homens e de mulheres e aplicação do teste de homogeneidade das variâncias com a função `var.test()` do R

Fonte. Autores (2021)

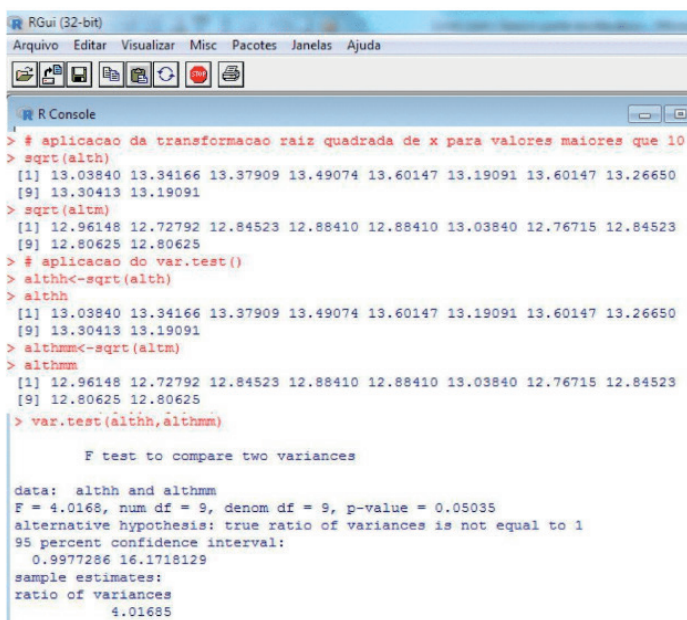
A interpretação do teste da figura 40 é que  $p$ -valor do teste entre a altura de homens e mulheres é menor ou igual que o  $p$ -valor crítico ( $p$ -valor = 0.05), assim com a decisão

$p$ -valor observado  $p$ -valor = 0.040, rejeitamos a hipótese  $h_0$ , concluindo que as variâncias não são homogêneas ou são diferentes. Segundo Centeno (1990) pode ser aplicada uma transformação aos dados e após ser feito o teste `var.test()` para verificar se o resultado não rejeitaria a hipótese  $h_0$ .

## TRANSFORMAÇÃO DOS DADOS COM AS FUNÇÕES LOG () E SQRT () NO R

No R a transformação para  $\sqrt{x}$  é feita com a função `sqrt` (nome do objeto) (LANDEIRO, 2011). Usamos esta transformação aos valores de altura de homens e mulheres, aplicando posteriormente o `var.test()` aos dados transformados para verificar se não há rejeição da hipótese  $h_0$  (ver figura 41).

Segundo Centeno (1990) para contagens e de uma forma geral para valores maiores que 10 aplicamos a transformação da  $\sqrt{x}$  onde  $x$  é cada valor do conjunto. Este tipo de transformação pode ser aplicado a variáveis contínuas ou medidas. Conforme este autor pode ser aplicado também quando as porcentagens variam entre 0% e 20% e 80% e 100%. Para valores de contagens menores que 10 e porcentagens baixas próximas a zero aplicam-se as transformações a cada valor do conjunto ( $\sqrt{x+0.5}$  ou  $\sqrt{x+1}$ ) (CENTENO, 1990). Considerando a transformação logarítmica  $\log(x)$ , utiliza-se-á quando as variáveis são números positivos ou no caso de porcentagens quando estas variam muito. Para valores “zero” utilizam-se  $\log(x+1)$ . Enfatiza-se que pode ser aplicada qualquer base de logaritmo (CENTENO, 1990).



```
RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda

R Console
> # aplicacao da transformacao raiz quadrada de x para valores maiores que 10
> sqrt(alth)
[1] 13.03840 13.34166 13.37909 13.49074 13.60147 13.19091 13.60147 13.26650
[9] 13.30413 13.19091
> sqrt(althm)
[1] 12.96148 12.72792 12.84523 12.88410 12.88410 13.03840 12.76715 12.84523
[9] 12.80625 12.80625
> # aplicacao do var.test()
> althh<-sqrt(alth)
> althh
[1] 13.03840 13.34166 13.37909 13.49074 13.60147 13.19091 13.60147 13.26650
[9] 13.30413 13.19091
> althmm<-sqrt(althm)
> althmm
[1] 12.96148 12.72792 12.84523 12.88410 12.88410 13.03840 12.76715 12.84523
[9] 12.80625 12.80625
> var.test(althh,althmm)

      F test to compare two variances

data:  althh and althmm
F = 4.0168, num df = 9, denom df = 9, p-value = 0.05035
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.9977286 16.1718129
sample estimates:
ratio of variances
 4.01685
```

Figura 41: Transformação das variáveis contínuas as alturas de homens e mulheres para raiz quadrada com aplicação do `var.test()` aos dados transformados.

Fonte. Autores (2021)

Conforme indica a figura 41 foi aplicada a transformação de  $\sqrt{x}$  aos dados de altura cuja interpretação do var.test () com estes dados transformados mostra que a resposta do teste rejeita a hipótese  $H_0$  porque o  $p$ -valor do teste é menor ou igual a 0.05 que é o valor crítico.

## TESTE ESTATÍSTICO DE DISTRIBUIÇÃO PARAMÉTRICA NO R

### Calculando o teste t de Student um teste paramétrico e interpretando o teste

Segundo Zar (1999) o teste t de Student ou teste T serve para comparar dois grupos que tenham distribuição normal ou paramétrica. As premissas deste teste exigem que ambos os grupos devem ter distribuição normal, variâncias iguais e serem independentes.

No R o teste é realizado pela função t.test (objeto 1, objeto 2, var.equal = TRUE, alternative = "two-sided") segundo Beasley (2004). De acordo com Zar (1999) o teste T de Student tem as seguintes premissas:

### TESTE DE HIPÓTESES SEGUNDO TRIOLA (2008)

- Teste bilateral:  $H_0$  (média do grupo 1 = média do grupo 2) e  $H_1$  (média do grupo 1  $\neq$  média do grupo 2);
- Teste Unilateral à direita:  $H_0$  (média do grupo 1  $\leq$  média do grupo 2) e  $H_1$  (média do grupo 1  $>$  média do grupo 2);
- Teste Unilateral à esquerda:  $H_0$  (média do grupo 1  $\geq$  média do grupo 2) e  $H_1$  (média do grupo 1  $<$  média do grupo 2).

A seguir, mostraremos as etapas do teste T de Student com a variável resposta: tempo de coagulação sanguínea após a aplicação de duas drogas (B e G). Serão aplicados os testes das premissas da normalidade e teste das variâncias e o teste t de Student para verificar se o tempo médio de coagulação sanguínea diferirá ou não entre os grupos para  $\alpha=5\%$  (ver figura 42).

```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

[158] "varios dados.txt"
> a<-read.table("tcs.txt",T)
> a
  Droga  tcs
1      b  8.8
2      b  8.4
3      b  7.9
4      b  8.7
5      b  9.1
6      b  9.6
7      g  9.9
8      g  9.0
9      g 11.1
10     g  9.6
11     g  8.7
12     g 10.4
13     g  9.5

> # teste da normalidade
> # para o grupo b
> shapiro.test(a$tcs[a$Droga=="b"])

      Shapiro-Wilk normality test

data:  a$tcs[a$Droga == "b"]
W = 0.99456, p-value = 0.9973

> # para o grupo g
> shapiro.test(a$tcs[a$Droga=="g"])

      Shapiro-Wilk normality test

data:  a$tcs[a$Droga == "g"]
W = 0.97211, p-value = 0.9132
```

Figura 42: Tabela do tempo de coagulação sanguínea=tcs (variável resposta) droga b e g (variável independente); teste da normalidade; teste das variâncias e o teste T de Student no R

Fonte. Autores (2021)

```

> # extracao do grupo b
> a[c(1:6),]
  Droga tcs
1      b 8.8
2      b 8.4
3      b 7.9
4      b 8.7
5      b 9.1
6      b 9.6
> # extracao do grupo g
> a[c(7:13),]
  Droga tcs
7      g 9.9
8      g 9.0
9      g 11.1
10     g 9.6
11     g 8.7
12     g 10.4
13     g 9.5

```

Figura 42: Tabela do tempo de coagulação sanguínea=tcs (variável resposta) droga b e g (variável independente); teste da normalidade; teste das variâncias e o teste T de Student no R

Fonte. Autores (2021)

```

> #grupo b
> b<-a[c(1:6),]
> b
  Droga tcs
1      b 8.8
2      b 8.4
3      b 7.9
4      b 8.7
5      b 9.1
6      b 9.6
> tcsb<-b$tcs
> tcsb
[1] 8.8 8.4 7.9 8.7 9.1 9.6
> length(tcsb)
[1] 6
> # grupo g
> gc<-a[c(7:13),]
> g
  Droga tcs
7      g 9.9
8      g 9.0
9      g 11.1
10     g 9.6
11     g 8.7
12     g 10.4
13     g 9.5
> tcsg<-g$tcs
> tcsg
[1] 9.9 9.0 11.1 9.6 8.7 10.4 9.5
> length(tcsg)
[1] 7
> var.test(tcsb,tcsg)

      F test to compare two variances

data:  tcsb and tcsg
F = 0.50633, num df = 5, denom df = 6, p-value = 0.4722
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.084856359 3.53301988
sample estimates:
ratio of variances
 0.50633
> t.test(tcsb,tcsg,var.equal=TRUE,alternative="two.sided")

      Two Sample t-test

data:  tcsb and tcsg
t = -2.4765, df = 11, p-value = 0.03076
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.8752609 -0.1104534
sample estimates:
mean of x mean of y
 8.750000  9.742857

```

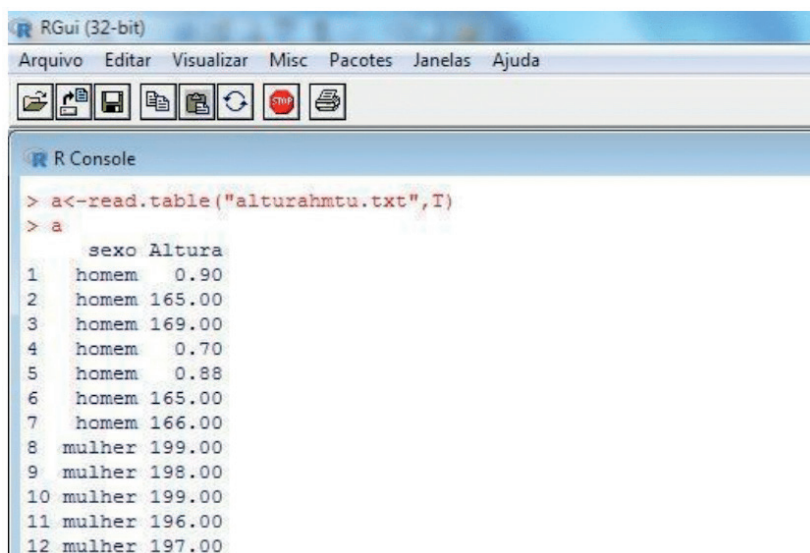
Figura 42: Tabela do tempo de coagulação sanguínea=tcs (variável resposta) droga b e g (variável independente); teste da normalidade; teste das variâncias e o teste T de Student no R

Fonte. Autores (2021)

## TESTE ESTATÍSTICO EQUIVALENTE NÃO PARAMÉTRICO NO R DO TESTE DE T DE STUDENT

### Calculando o teste U de Mann-Whitney e interpretando o teste

De acordo com Zar (1999) o teste U de Mann-Whitney é um teste não paramétrico, representando o equivalente não paramétrico do teste T de Student. O teste U é um teste aplicado a dois grupos onde pelo menos um não apresenta a distribuição normal (ZAR, 1999). No R o teste de Mann-Whitney é calculado como o teste de Wilcoxon Rank Sum teste (HEIBERGER & HOLLAND, 1989). Conforme Beasley (2004) este teste é calculado com a função `wilcox.test` (objeto 1, objeto2).Mostraremos um exemplo modificado do Zar (1999) para o teste U de Mann-Whitney comparando a altura de estudantes homens e mulheres. Neste teste conforme Gasnier (2011) a variável resposta é a altura e a variável independente é o sexo (homem e mulher) (ver figura 43).



```
> a<-read.table("alturahmtu.txt",T)
> a
  sexo  Altura
1 homem  0.90
2 homem 165.00
3 homem 169.00
4 homem  0.70
5 homem  0.88
6 homem 165.00
7 homem 166.00
8 mulher 199.00
9 mulher 198.00
10 mulher 199.00
11 mulher 196.00
12 mulher 197.00
```

Figura 43: Grupo de alturas de homens e de mulheres a partir da modificação de um exemplo retirado do livro Zar (1999); teste de Normalidade e teste U aplicando-se o teste de Wilcoxon

Fonte. Autores (2021)

```

> # realizando o teste da normalidade por grupo
> # grupo homem
> shapiro.test(a$Altura[a$sexo=="homem"])

      Shapiro-Wilk normality test

data:  a$Altura[a$sexo == "homem"]
W = 0.67572, p-value = 0.002002

> # grupo mulher
> shapiro.test(a$Altura[a$sexo=="mulher"])

      Shapiro-Wilk normality test

data:  a$Altura[a$sexo == "mulher"]
W = 0.90202, p-value = 0.4211

> # extraindo os grupos
> # extraindo o grupo homem
> a[c(1:7),]
  sexo Altura
1 homem  0.90
2 homem 165.00
3 homem 169.00
4 homem  0.70
5 homem  0.88
6 homem 165.00
7 homem 166.00
> m<-a[c(1:7),]
> m
  sexo Altura
1 homem  0.90
2 homem 165.00
3 homem 169.00
4 homem  0.70
5 homem  0.88
6 homem 165.00
7 homem 166.00
> alth<-h$Altura
> alth
[1] 0.90 165.00 169.00 0.70 0.88 165.00 166.00
> # extraindo o grupo mulher
> a[c(8:12),]
  sexo Altura
8 mulher  199
9 mulher  198
10 mulher 199
11 mulher 196
12 mulher 197
> m<-a[c(8:12),]
> m
  sexo Altura
8 mulher  199
9 mulher  198
10 mulher 199
11 mulher 196
12 mulher 197
> altm<-m$Altura
> altm
[1] 199 198 199 196 197

```

Figura 43: Grupo de alturas de homens e de mulheres a partir da modificação de um exemplo retirado do livro Zar (1999); teste de Normalidade e teste U aplicando-se o teste de Wilcoxon

Fonte. Autores (2021)



```

> # realizando o teste U
> wilcox.test(alth, altm)

      Wilcoxon rank sum test with continuity correction

data:  alth and altm
W = 0, p-value = 0.005597
alternative hypothesis: true location shift is not equal to 0

Warning message:
In wilcox.test.default(alth, altm) :
  não é possível computar o valor de p exato com o de desempate
> |

```

Figura 43: Grupo de alturas de homens e de mulheres a partir da modificação de um exemplo retirado do livro Zar (1999); teste de Normalidade e teste U aplicando-se o teste de Wilcoxon

Fonte. Autores (2021)

Observa-se da figura 43 na comparação da distribuição normal da altura de homens e de mulheres, somente os dados de altura de homens não apresentam distribuição normal conforme o  $p$ -valores, 0.002002 e 0.4211, respectivamente. Assim foi aplicado o teste U usando-se a função `wilcox.test()` cujo resultado do teste indica que houve diferença estatística significativa de acordo com o  $p$ -valor do teste (0.005597).

## TESTE ESTATÍSTICO DE DISTRIBUIÇÃO PARAMÉTRICA NO R

### Calculando a ANOVA one way com a função `aov` e construindo as hipóteses do teste

Conforme Zar (1999) o teste de análise de variância - ANOVA one way ou de um fator é realizado para uma variável independente categórica (nominal ou ordinal) e uma variável resposta (contagem ou contínua). A ANOVA tem premissas que são: distribuição normal, variâncias homogêneas e grupos independentes. No R utilizaremos a função `aov` (variável dependente ~ variável independente, `data = nome do objeto que contém a tabela com os dados para ser feita a ANOVA`) para fazer a ANOVA one way. Os resultados da ANOVA serão visualizados no R com a função `summary` (nome do objeto em que foi copiado e salvo a função `aov`) (BEASLEY, 2004). O teste de normalidade será feito com a função `shapiro.test` (variável resposta [variável independente == "nome do grupo da variável independente"]) (BEASLEY, 2004). Por sua vez, o teste de homogeneidade das variâncias será feito com o teste de Bartlett com a função `bartlett.test` (variável resposta ~ variável independente, nome do objeto que contém a tabela com as variáveis para ser feita a ANOVA) (SOUZA, ET AL., 2018). Teste de hipóteses segundo Zar (1999).

- As hipóteses do teste de normalidade:  $H_0$  (há distribuição normal dos pesos dos grupos a, b, c e d) e  $H_1$  (não há distribuição normal nos pesos dos grupos a, b, c e d);
- As hipóteses do teste da homogeneidade de variâncias:  $H_0$  (não há diferença entre as variâncias dos grupos a, b, c e d) e  $H_1$  (há diferença entre as variâncias dos grupos a, b, c e d)

- As hipóteses da ANOVA são:  $H_0$  (não há diferença entre os pesos médios dos grupos a, b, c e d) e  $H_1$  (há diferença entre os pesos médios dos grupos a, b, c e d).

Utilizaremos um exemplo do livro Zar (1999) para realizar o teste de ANOVA one way. Neste exemplo, o peso de porcos será a variável resposta e o tipo de ração a variável independente com quatro níveis (a, b, c e d). Serão realizados os testes de normalidade para cada grupo, e o teste da diferença entre as variâncias e o teste de ANOVA one way conforme a figura 44.

```

RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda

R R Console

[170] "varios dados.txt"
> b<-read.table("racaopigs.txt",T)
> b
  racao peso
1     a  60.8
2     a  57.0
3     a  65.0
4     a  58.6
5     a  61.7
6     b  68.7
7     b  67.7
8     b  74.0
9     b  66.3
10    b  69.8
11    c 102.6
12    c 102.4
13    c 100.2
14    c  96.5
15    d  87.9
16    d  84.2
17    d  83.1
18    d  85.7

> # realizando o teste de normalidade entre grupos
> # grupo a
> shapiro.test(b$peso[b$racao=="a"])

      Shapiro-Wilk normality test

data:  b$peso[b$racao == "a"]
W = 0.97924, p-value = 0.9305

> # grupo b
> shapiro.test(b$peso[b$racao=="b"])

      Shapiro-Wilk normality test

data:  b$peso[b$racao == "b"]
W = 0.92341, p-value = 0.5521

> # grupo c
> shapiro.test(b$peso[b$racao=="c"])

      Shapiro-Wilk normality test

data:  b$peso[b$racao == "c"]
W = 0.86259, p-value = 0.2696

```

Figura 44: Tabela com os dados para o teste de ANOVA one way com a variável resposta o peso de porcos e a variável independente o tipo de ração em quatro níveis: a, b, c e d; teste de normalidade por grupo e teste de homogeneidade entre as variâncias.

Fonte. Autores (2021)

```

> # grupo d
> shapiro.test(b$peso[b$racao=="d"])

      Shapiro-Wilk normality test

data:  b$peso[b$racao == "d"]
W = 0.97246, p-value = 0.8567

      Bartlett test of homogeneity of variances

data:  b$peso by b$racao
Bartlett's K-squared = 0.47087, df = 3, p-value = 0.9252

```

Figura 44: Tabela com os dados para o teste de ANOVA one way com a variável resposta o peso de porcos e a variável independente o tipo de ração em quatro níveis: a, b, c e d; teste de normalidade por grupo e teste de homogeneidade entre as variâncias.

Fonte. Autores (2021)

Segundo a figura 44 os resultados indicam que os pesos dos grupos a, b, c e d apresentam distribuição normal pelo  $p$ -valores, sendo 0.9305, 0.5521, 0.2696 e 0.8567, nesta ordem. O teste de Bartlett indicou que as variâncias são homogêneas entre os grupos pelo  $p$ -valor = 0.9252.

## COMO VISUALIZAR OS RESULTADOS DA ANOVA COM A FUNÇÃO SUMMARY

No R, segundo Beasley (2004) os resultados da análise de variância com a função `avov ()` podem ser visualizados com a função `summary ()` (vide figura 45).

```

RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda
[Icons]
R Console
> anova<-aov(b$peso~b$racao,data=b)
> anova
Call:
aov(formula = b$peso ~ b$racao, data = b)

Terms:
          b$racao Residuals
Sum of Squares 4101.361  108.863
Deg. of Freedom      3      14

Residual standard error: 2.788535
Estimated effects may be unbalanced
> summary(anova)
          Df Sum Sq Mean Sq F value Pr(>F)
b$racao   3  4101  1367.1  175.8 2.4e-11 ***
Residuals 14   109    7.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

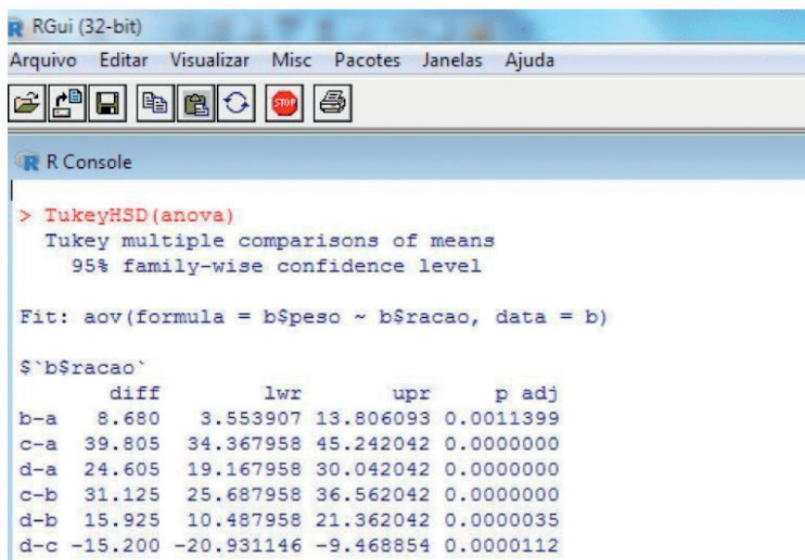
Figura 45: Teste de ANOVA com a função `aov()` e a visualização dos resultados com a função `summary()`

Fonte. Autores (2021)

Segundo a figura 45 o teste de ANOVA one way mostrado com a função `summary()` indicou que houve diferença estatística significativa conforme o  $p$ -valor =  $2.14 \times 10^{-1}$ .

## CALCULANDO OS TESTES *A POSTERIORI* NO R COM A FUNÇÃO `TUKEYHSD`

De acordo com Beasley (2004) no R o teste de Tukey *a posteriori* pode ser realizado com a função `TukeyHSD` (Nome do objeto que contém o teste de ANOVA pela função `aov()`). Na figura 46 temos o teste de Tukey.



```
> TukeyHSD(anova)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = b$peso ~ b$racao, data = b)

$b$racao`
      diff      lwr      upr      p adj
b-a    8.680  3.553907 13.806093 0.0011399
c-a   39.805 34.367958 45.242042 0.0000000
d-a   24.605 19.167958 30.042042 0.0000000
c-b   31.125 25.687958 36.562042 0.0000000
d-b   15.925 10.487958 21.362042 0.0000035
d-c  -15.200 -20.931146 -9.468854 0.0000112
```

Figura 46: Resultado do teste de Tukey *a posteriori* para a verificação das diferenças entre as médias dos porcos alimentados com ração de quatro tipos.

Fonte. Autores (2021)

## UTILIZANDO O PACOTE “PSYCH” E A FUNÇÃO `DESCRIBEBY()` PARA CALCULAR A MÉDIA E O ERRO PADRÃO DE GRUPOS

Podemos calcular a estatística descritiva utilizando o pacote “psych” e a função `describeBy` (nome do objeto, `group = nome do objeto $ variável citada`) (BRANCO & HARNIK, 2017). Para os mesmos dados é possível representá-las em uma tabela, as médias e os erros padrões de cada tratamento, citando as letras para diferenciar as médias com ou sem diferença significativa (ver figura 47).

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

> # instalando o pacote psych para usar a funcao describeBy()
> install.packages("psych")
Installing package into 'C:/Users/ELTON/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
tentando a URL 'https://brieger.esalq.usp.br/CRAN/bin/windows/contrib/3.6/psych_
Content type 'application/zip' length 4199681 bytes (4.0 MB)
downloaded 4.0 MB

package 'psych' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\ELTON\AppData\Local\Temp\Rtmp42RjXz\downloaded_packages
> require("psych")
Carregando pacotes exigidos: psych
Warning message:
package 'psych' was built under R version 3.6.3

Descriptive statistics by group
group: a
  vars n mean  sd median trimmed  mad min max range skew kurtosis  se
X1    1 5 60.62 3.06  60.8   60.62 3.26  57  65    8 0.21   -1.74 1.37
-----
group: b
  vars n mean  sd median trimmed  mad min max range skew kurtosis  se
X1    1 5 69.3 2.93  69.7   69.3 1.63 66.3  74  7.7 0.58   -1.43 1.31
-----
group: c
  vars n mean  sd median trimmed  mad min max range skew kurtosis  se
X1    1 4 100.42 2.83 101.3 100.42 1.78 96.5 102.6  6.1 -0.47   -1.93 1.42
-----
group: d
  vars n mean  sd median trimmed  mad min max range skew kurtosis  se
X1    1 4 85.22 2.08  84.95  85.22 1.93 83.1 87.9  4.8 0.24   -2.02 1.04
-----
>

```

Figura 47: Usando o pacote psych e a função describeBy () para o cálculo da média e do erro padrão entre os grupos

Fonte. Autores (2021)

De acordo com Centeno (1990) o erro padrão é calculado pela fórmula  $e = \pm z * \frac{sd}{\sqrt{n}}$  conforme  $\alpha$  utilizado no teste. No caso, se utilizarmos  $\alpha=5\%$  usaremos aproximadamente 2 desvios padrões, desta forma multiplicaremos o resultado pelo valor 1.96 que corresponde ao  $\alpha=5\%$ .

Conforme apresenta a figura 47 a função describeBy () gerou dentre os valores, a média representada pela palavra “mean” e o erro padrão sem multiplicar pelo valor de z representada pela sigla “se”. Por exemplo, o erro padrão do grupo a foi  $se = 1.37$ . Se dividirmos o desvio padrão (sd) deste grupo que está representada pela sigla “sd” pela  $\sqrt{n}$  teremos o valor  $se = 1.37$  (arredondado). Considerando o  $\alpha = 5\%$  o erro padrão para  $se=2.68$  segundo a formula de Centeno (1990). A tabela final da ANOVA será representada a seguir.

Grupo	a	b	c	d
$\mu \pm se^* \alpha = 5\%$	$60.62 \pm 2.68_a$	$69.3 \pm 2.57_{bb}$	$100.42 \pm 2.77_{ccc}$	$85.22 \pm 2.04_{ddd}$

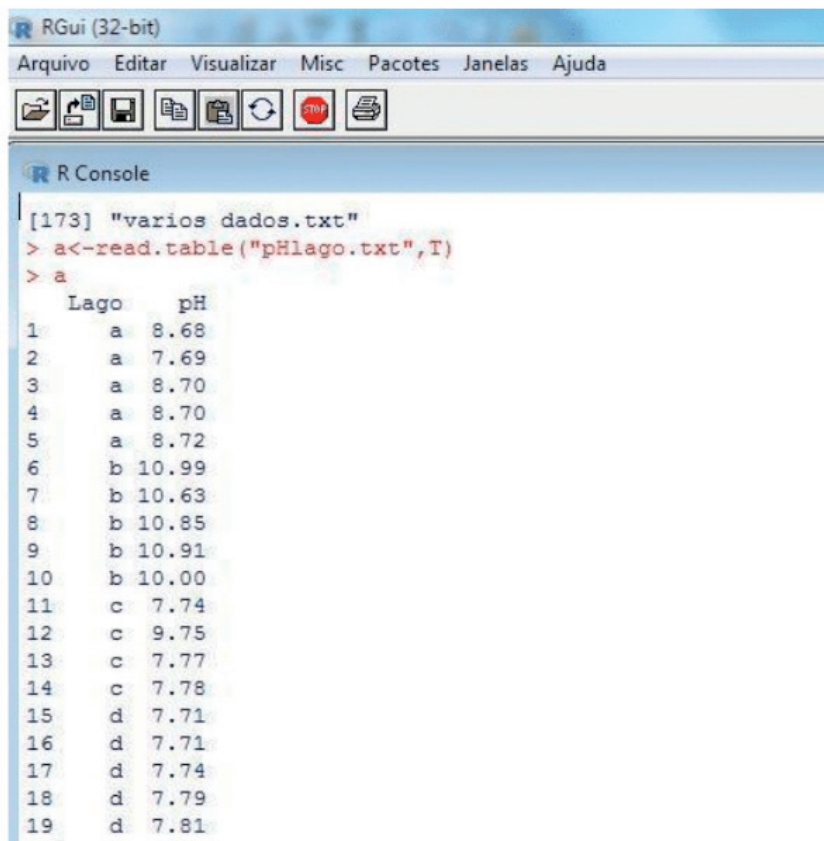
Tabela 1. Tabela representativa do teste de ANOVA e do teste *a posteriori* com a média, o erro padrão e as letras.

De acordo com a tabela 1 o grupo C seguido do grupo D foram os grupos com os maiores pesos médios após a utilização das rações; neste caso, seleciona-se a ração do tipo C por apresentar um melhor desempenho zootécnico. Esta função pode ser utilizada para serem obtidos os valores de média e erro padrão para representar a tabela para o teste T de Student.

## TESTE ESTATÍSTICO EQUIVALENTE NÃO PARAMÉTRICO DA ANOVA ONE WAY NO R

### Calculando o teste de Kruskal-Wallis no R com a função `kruskal.test()`

Conforme Dalgaard (2008) o teste de Kruskal-Wallis no R é realizado com a função `kruskal.test` (variável resposta ~ variável independente). De acordo com Zar (1999) o teste de Kruskal-Wallis é um teste não paramétrico equivalente à ANOVA one way. Pra fins didáticos, utilizaremos um exemplo obtido de Zar (1999) onde a variável resposta é o pH da água de lagos e a variável independente é o tipo de lago em quatro níveis ou tipos (a, b, c e d). Iremos realizar o teste da normalidade por grupos e em seguida aplicar a função `kruskal.test ()` (ver figura 48).



```
[173] "varios dados.txt"
> a<-read.table("pHlago.txt",T)
> a
  Lago  pH
1    a 8.68
2    a 7.69
3    a 8.70
4    a 8.70
5    a 8.72
6    b 10.99
7    b 10.63
8    b 10.85
9    b 10.91
10   b 10.00
11   c  7.74
12   c  9.75
13   c  7.77
14   c  7.78
15   d  7.71
16   d  7.71
17   d  7.74
18   d  7.79
19   d  7.81
```

Figura 48: Tabela contendo os dados de pH de lagos em quatro tipos a, b, c e d e teste de normalidade e de Kruskal-Wallis no R

Fonte. Autores (2021)

Verifica-se (vide figura 48) que pelo menos um grupo não apresentou distribuição normal; por exemplo, os grupos a e c não apresentaram distribuição normal conforme os  $p$ -valores, 0.0003535 e 0.002236 respectivamente. Desta forma, sem o pré-requisito da distribuição normal em ambos os grupos, aplica-se o teste não paramétrico de Kruskal-Wallis. O teste de Kruskal-Wallis indica que há diferença significativa segundo o  $p$ -valor = 0.007794.

```

> # realizando o teste de normalidade por grupos
> # grupo a
> shapiro.test(a$pH[a$Lago=="a"])

      Shapiro-Wilk normality test

data:  a$pH[a$Lago == "a"]
W = 0.58182, p-value = 0.0003535

> # grupo b
> shapiro.test(a$pH[a$Lago=="b"])

      Shapiro-Wilk normality test

data:  a$pH[a$Lago == "b"]
W = 0.81911, p-value = 0.1149

> # grupo c
> shapiro.test(a$pH[a$Lago=="c"])

      Shapiro-Wilk normality test

data:  a$pH[a$Lago == "c"]
W = 0.64602, p-value = 0.002236

> # grupo d
> shapiro.test(a$pH[a$Lago=="d"])

      Shapiro-Wilk normality test

data:  a$pH[a$Lago == "d"]
W = 0.86757, p-value = 0.2567

> # realizando o teste de Kruskal-Wallis
> kruskal.test(a$pH~a$Lago)

      Kruskal-Wallis rank sum test

data:  a$pH by a$Lago
Kruskal-Wallis chi-squared = 11.883, df = 3, p-value = 0.007794

```

Figura 48: Tabela contendo os dados de pH de lagos em quatro tipos a, b, c e d e teste de normalidade e de Kruskal-Wallis no R

## REALIZANDO A COMPARAÇÃO DE MEDIAS *A POSTERIORI* COM A FUNÇÃO DUNNTEST NO PACOTE FSA

De acordo com Ogle et al. (2020) para realizar o teste de Dunn é necessário instalar e carregar o pacote FSA criado por Ogle et al. (2020). Este teste é um teste não paramétrico que compara a diferença entre postos médios após o teste de Kruskal-Wallis. Ainda segundo Ogle et al., (2020), o Teste de Dunn é realizado com a função `dunnTest` (variável resposta~variável independente, `data` = nome do objeto que contém a tabela dos dados que foi feito o Kruskal-Wallis) (vide figura 49).



```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console
> dunnTest(a$P~a$Lago,a)
Dunn (1964) Kruskal-Wallis multiple comparison
p-values adjusted with the Holm method.

  Comparison      Z      P.unadj      P.adj
1   a - b -2.1382310 0.0324980023 0.129992009
2   a - c  0.4045150 0.6858340674 0.685834067
3   b - c  2.4204585 0.0155009501 0.077504750
4   a - d  1.1535193 0.2486972854 0.746091856
5   b - d  3.2917503 0.0009956597 0.005973958
6   c - d  0.6830335 0.4945856844 0.989171369
>

```

Figura 49: Teste de Dunn *a posteriori* realizado com a função `dunnTest()` para determinar os postos médios diferentes do teste de Kruskal-Wallis.

Fonte. Autores (2021)

Observa a partir da figura 48 que o teste de Dunn indicou uma diferença estatística significativa entre b e d. De acordo com Vieira (2010) é possível, mesmo em testes não paramétricos, utilizar os valores de média e erro padrão para se comparar, apenas de forma ilustrativa os resultados. Partindo desta premissa também podemos utilizar o pacote “psych” e a função `describeBy()` para obtermos a estatística descritiva dos valores de média e erro padrão, no entanto não iremos ilustrar estes resultados aqui.

## TESTE ESTATÍSTICO DE DISTRIBUIÇÃO PARAMÉTRICA NO R

### Calculando a correlação de Pearson no R e construindo as hipóteses do teste

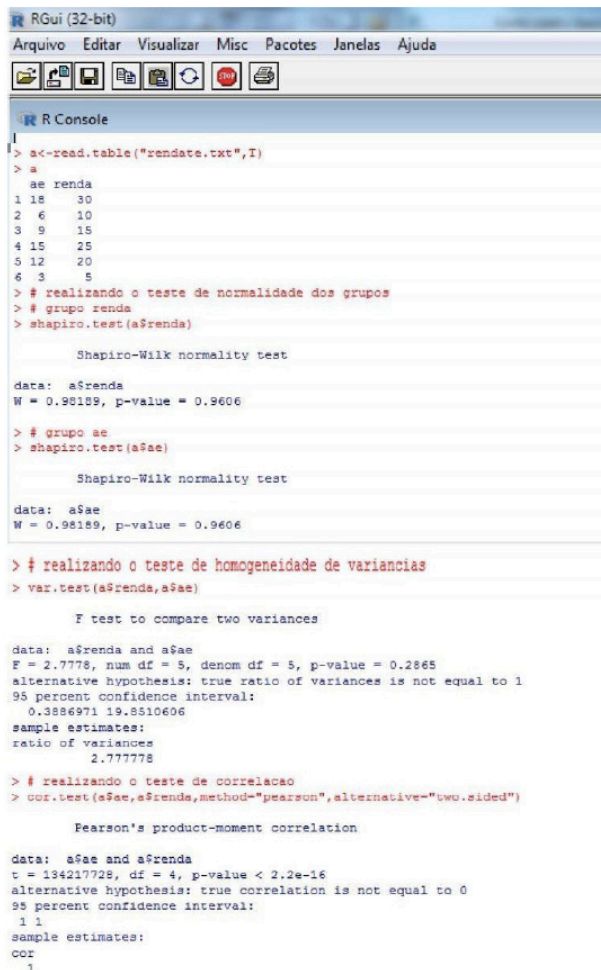
De acordo com Zar (1999) a correlação de Pearson é uma associação entre a variável resposta ( $y$ ) e a variável independente ( $x$ ) em que ambas as variáveis tem distribuição normal, variâncias homogêneas e são independentes. O cálculo de correlação produz um coeficiente de correlação denominado de  $R$  que varia entre  $-1$ ,  $0$  e  $+1$  indicando as correlações perfeita negativa ( $r=-1$ ); ausência de correlação ( $r=0$ ) ou próximo de zero e a correlação perfeita positiva ( $r=1$ ) respectivamente. O teste de normalidade será o de Shapiro-Wilks segundo Ayres et al. (2003) realizado pela função `shapiro.test` (nome do objeto que contém a variável), enquanto o teste de homogeneidade de variâncias será realizado com a função `var.test` (objeto da variável 1, objeto da variável2) (CRAWLEY, 2013). No R a correlação de Pearson é calculada pela função `cor.test` (variável independente, variável dependente, `method = "pearson"`, `alternative = "two.sided"`) (BEASLEY, 2008). Utilizaremos um exemplo adaptado de Levin (1987) para a correlação linear de Pearson onde a variável resposta ( $y$ ) é a renda ( $\times 1000$  reais) e a variável independente ( $x$ ) tempo de escola (anos) (vide figura 49).

## Teste de hipóteses da distribuição normal segundo Crawley (2013)

- Valores de  $y$ :  $H_0$  (os valores de  $y$  apresentam distribuição normal) e  $H_1$  (os valores de  $y$  não apresentam distribuição normal);
- Valores de  $x$ :  $H_0$  (os valores de  $x$  apresentam distribuição normal) e  $H_1$  (os valores de  $x$  não apresentam distribuição normal)

Teste de hipóteses da homogeneidade das variâncias:  $H_0$  ( $\text{var1}=\text{var2}$ ) e  $H_1$  ( $\text{var1}\neq\text{var2}$ ) (ZAR, 1999).

Teste de hipóteses da correlação  $H_0$  (não há correlação entre os valores  $y$  e  $x$ ) e  $H_1$  (há correlação entre os valores de  $y$  e  $x$ ) (AYRES ET AL., 2003).



```
RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda

R Console

> a<-read.table("rendate.txt",I)
> a
  ae renda
1 18     30
2  6     10
3  9     15
4 15     25
5 12     20
6  3      5
> # realizando o teste de normalidade dos grupos
> # grupo renda
> shapiro.test(a$renda)

      Shapiro-Wilk normality test

data:  a$renda
W = 0.98189, p-value = 0.9606

> # grupo ae
> shapiro.test(a$ae)

      Shapiro-Wilk normality test

data:  a$ae
W = 0.98189, p-value = 0.9606

> # realizando o teste de homogeneidade de variancias
> var.test(a$renda,a$ae)

      F test to compare two variances

data:  a$renda and a$ae
F = 2.7778, num df = 5, denom df = 5, p-value = 0.2865
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.3886971 19.8510606
sample estimates:
ratio of variances
 2.777778

> # realizando o teste de correlacao
> cor.test(a$ae,a$renda,method="pearson",alternative="two.sided")

      Pearson's product-moment correlation

data:  a$ae and a$renda
t = 134217728, df = 4, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 1 1
sample estimates:
cor
1
```

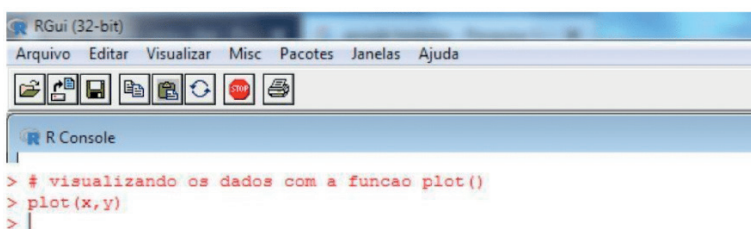
Figura 49: Tabela com os dados  $y$ =renda e  $x$ =dias; teste de normalidade com a função shapiro.test (); teste de homogeneidade das variâncias com a função var.test() e o teste de correlação com a função cor.test()

Fonte. Autores (2021)

Visualiza-se na figura 49 que os dados de  $y$  e  $x$  apresentaram distribuição normal conforme os  $p$ -valores 0.9606 e 0.9606, respectivamente. O teste de homogeneidade das variâncias indicou que as estas são homogêneas pelo  $p$ -valor = 0.2865. O teste de correlação de Pearson, por sua vez, rejeitou a hipótese  $h_0$  indicando que a correlação é significativa pelo  $p$ -valor =  $2.2e^{-16}$  e o valor do coeficiente de correlação foi  $r=+1$ , sinalizando uma correlação positiva, ou seja, o valor de  $y$  aumenta com o aumento do valor de  $x$ .

### Visualizando os dados submetidos à correlação previamente pelo gráfico scatterplot com a função plot()

Uma forma preliminar de verificar a tendência da relação  $y$  em função de  $x$  é construir um gráfico de pontos ou scatterplot. De acordo com Beasley (2004), o gráfico de pontos é feito com a função plot (variável independente, variável dependente) (vide figura 50).



```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda
[Icons]
R Console
> # visualizando os dados com a funcao plot()
> plot(x,y)
> |
```

Figura 50: Visualizando a relação  $y$  em função de  $x$  da correlação utilizando a função plot ()

Fonte. Autores (2021)

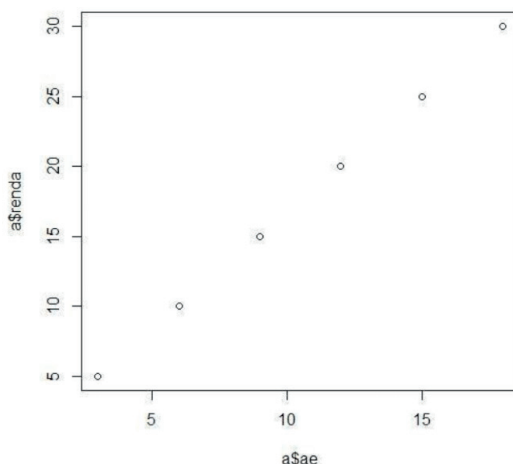


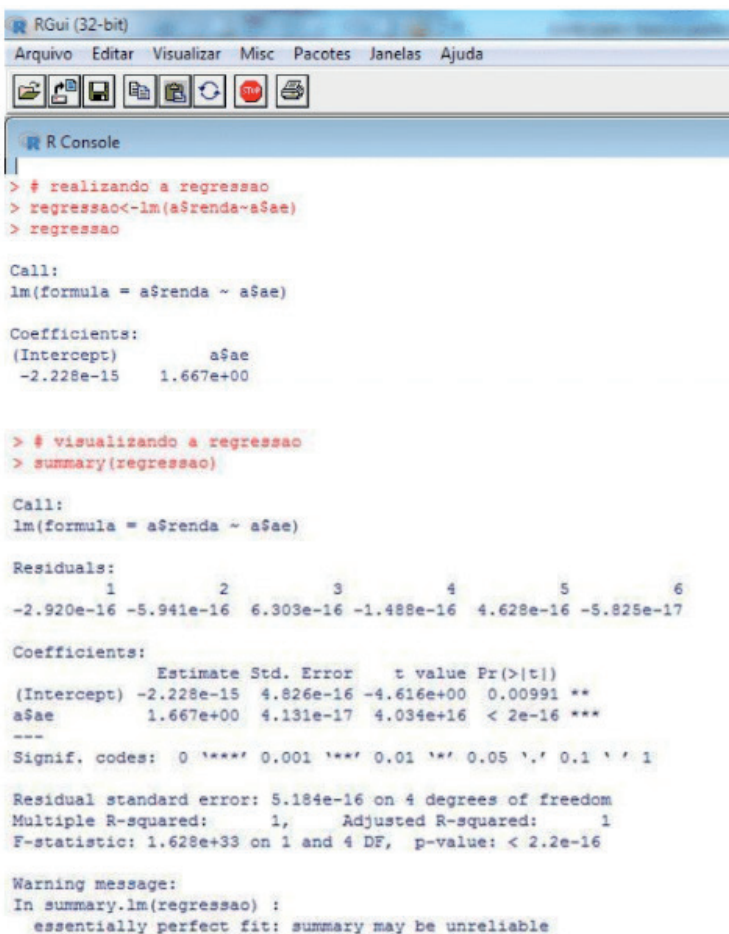
Figura 50: Gráfico scatterplot no R após aplicar a função plot ()

Fonte. Autores (2021)

De posse da figura 50, visualiza-se que o gráfico scatterplot indica uma tendência positiva onde à medida que aumentam os valores de  $x$  os valores de  $y$  também aumentam.

## Calculando a equação de regressão linear no R com a função `lm()` e visualização da regressão com a função `summary()`

De acordo com Zar (1999) havendo uma correlação significativa determinam-se os valores de “a” (intercepto) que corresponde ao coeficiente linear (isto é, o ponto em que a reta corta o eixo dos y) e o valor de “b” que é o coeficiente angular, indicando pelo sinal positivo a tendência dos valores y aumentarem em função do aumento dos valores de x, e pelo sinal negativo a tendência dos valores de y diminuir com o aumento dos valores de x. No R, segundo Beasley (2004) os valores de “a” e “b” da equação de regressão linear são calculados usando a função `lm(y~x)` copiada e salva em um objeto que a represente. A visualização da regressão é realizada com a função `summary()` (objeto que contém a regressão da função `lm()`) (ver figura 51).



```
RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

> # realizando a regressao
> regressao<-lm(a$renda~a$ae)
> regressao

Call:
lm(formula = a$renda ~ a$ae)

Coefficients:
(Intercept)      a$ae
-2.228e-15      1.667e+00

> # visualizando a regressao
> summary(regressao)

Call:
lm(formula = a$renda ~ a$ae)

Residuals:
    1      2      3      4      5      6
-2.920e-16 -5.941e-16  6.303e-16 -1.488e-16  4.628e-16 -5.825e-17

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept) -2.228e-15  4.826e-16 -4.616e+00  0.00991 **
a$ae         1.667e+00  4.131e-17  4.034e+16 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.184e-16 on 4 degrees of freedom
Multiple R-squared: 1, Adjusted R-squared: 1
F-statistic: 1.628e+33 on 1 and 4 DF, p-value: < 2.2e-16

Warning message:
In summary.lm(regressao) :
  essentially perfect fit: summary may be unreliable
```

Figura 51: Aplicação da função `lm()` para o cálculo dos valores “a” e “b” e da função `summary()` para visualizar a regressão

Fonte. Autores (2021)

Da figura 51 verifica-se que a regressão é significativa conforme o  $p$ -valor:  $2.2e^{-1}$ . O valor do “a” (intercepto) é  $-2.228e^{-1}$ , enquanto o valor de “b” (coeficiente angular) é  $1.667e^{+0}$  indicando uma tendência de aumento dos valores de  $y$  em função dos valores de  $x$ . Observa-se que o coeficiente de determinação ajustado foi ( $r^2 = +1$ ). Nesse sentido, a equação  $y = -2.228e^{-1} + 1.667x$  é adequada para estimar valores de  $y$  entre valores de  $x$  não coletados. O coeficiente de determinação ajustado ( $r^2 = 100\%$ ) avalia 100% do aumento de  $y$  em função do aumento de  $x$ .

## TESTE ESTATÍSTICO EQUIVALENTE NÃO PARAMÉTRICO DA CORRELAÇÃO DE PEARSON NO R

### Calculando a correlação de Spearman no R e interpretando os resultados

De acordo com Zar (1999) a correlação de Spearman é uma associação entre a variável resposta ( $y$ ) e a variável independente ( $x$ ) em que pelo menos uma não tem distribuição normal. O cálculo de correlação produz um coeficiente de correlação denominado de  $r$  que varia de  $-1$ ,  $0$  e  $+1$ , indicando as correlações perfeita negativa onde o  $r$  é um valor negativo, ausência de correlação o  $r = 0$  ou próximo de zero, e a correlação perfeita positiva  $r$  é um valor positivo respectivamente. No R, a correlação de Spearman é calculada pela função `cor.test` (variável independente, variável dependente, `method = "spearman"`, `alternative = "two.sided"`) (BEASLEY, 2008). Para ilustrarmos o uso dessa correlação, utilizaremos um exemplo modificado de Bhujel (2008) onde a variável resposta ( $y$ ) é a porcentagem de ovos viáveis e a variável independente ( $x$ ) é a concentração de amônia tóxica (ppm) (ver figura 52).

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

> a<-read.table("viabatox.txt",T)
> a
  atox viab
1  38.0 74.9
2  38.8 77.6
3  37.1 30.2
4  38.2 51.7
5  34.5 68.2
6  33.2 85.3
7  32.1 81.1
8  31.9 85.6
9  32.4 93.0
10 31.9 96.4
11 30.5 95.3
12 29.3 97.2

> # realizando o teste de normalidade dos grupos
> # grupo viab
> shapiro.test(a$viab)

      Shapiro-Wilk normality test

data:  a$viab
W = 0.85688, p-value = 0.04469

> # grupo atox
> shapiro.test(a$atox)

      Shapiro-Wilk normality test

data:  a$atox
W = 0.90427, p-value = 0.1801

```

Figura 52: Tabela com os dados de y=porcentagem de ovos viáveis e x=concentração de amônia tóxica (ppm); teste de normalidade com a função shapiro.test()

Fonte. Autores (2021)

```

RGui (32-bit)
Arquivo  Editar  Visualizar  Misc  Pacotes  Janelas  Ajuda

R Console

> # grafico de pontos com a funcao plot()
> plot(a$atox,a$viab)
> |

```

Figura 53: Uso da função plot () para fazer o gráfico de pontos dos valores de y=porcentagem de ovos viáveis e x=concentração de amônia tóxica (ppm)

Fonte. Autores (2021)

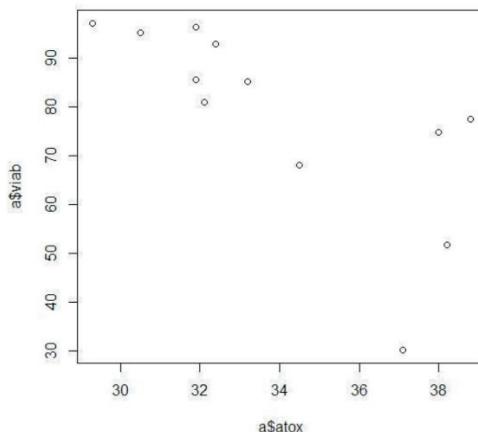


Figura 54: Gráfico de pontos no R com a função plot() das variáveis y=porcentagem de ovos viáveis e x=concentração de amônia tóxica (ppm)

Fonte. Autores (2021)

De acordo com a figura 54 verifica-se uma tendência negativa de redução da porcentagem de ovos viáveis com o aumento da concentração de amônia tóxica.

```

RGui (32-bit)
Arquivo Editar Visualizar Misc Pacotes Janelas Ajuda

R Console

> # realizando o teste de correlacao de Spearman
> cor.test(a$atox,a$viab,method="spearman",alternative="two.sided")

Spearman's rank correlation rho

data: a$atox and a$viab
S = 527.42, p-value = 0.0005543
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
-0.8441344

Warning message:
In cor.test.default(a$atox, a$viab, method = "spearman", alternative = "two.sided")
Impossível calcular o valor exato de p com empates

```

Figura 55: Teste de correlação de Spearman para as variáveis y=porcentagem de ovos viáveis e x=concentração de amônia tóxica (ppm) com a função cor.test()

Fonte. Autores (2021)

A figura 55 evidencia que a porcentagem de ovos viáveis não apresentou distribuição normal e a concentração de amônia tóxica apresentou a distribuição normal conforme os

p-valores, sendo 0.04469 e 0.1801 respectivamente. O teste de correlação de Spearman indicou uma correlação significativa pelo p-valor 0.0005543 e um coeficiente de correlação de Spearman igual a - 0.84 indicando que a porcentagem de ovos viáveis diminui com o aumento da concentração de amônia tóxica.

## ABORDAGEM EPR

De acordo com Gasnier (2014), a abordagem EPR auxilia os estudantes a transformar problemas biológicos em estudos estatísticos. Na sigla EPR, a letra “E” significa entidade e como sinônimo objeto de estudo ou organismo quanto à área biológica, variando conforme a área (por exemplo, na área da física a entidade pode ser uma coisa simples como um automóvel ou um conjunto como uma Floresta).

A letra “P”, por sua vez, significa propriedade cujos sinônimos são característica e atributo que representam as características (denominadas variáveis) de cada entidade. Assim, uma árvore tem espécie, altura, tipo de solo, etc. As propriedades variam segundo a entidade. No caso do ser humano as propriedades podem ser sexo, altura, meio de transporte e etc. Já a letra “R” refere-se à relação entre as variáveis. Na abordagem EPR determina-se a variável “x” ou a preditora ou a variável independente e a variável resposta ou dependente. Nesta relação, tem-se o entendimento inicial para o estudo estatístico no qual a variável dependente “depende” da variável independente passando a variação da variável independente influir sobre a variação da variável dependente. Esta relação segundo Gasnier (2014) pode ser causal indicando que há uma relação verdadeira ou consequente que indique uma relação de “causa” versus “efeito”. Com as variáveis de “causa” e “efeito” podemos construir tabelas ou planilhas EPR’s e gráficos EPR’s. Na tabela EPR as entidades ficam em cada linha, as variáveis tanto “causa” quanto “resposta” ficam nas colunas e os valores destas variáveis em cada linha representativa de um objeto de estudo. Nos gráficos EPR’s a variável resposta fica no eixo do “y” e as variáveis preditoras no eixo do “x”.



## REFERÊNCIAS

ALCOFORADO, L. F. **Utilizando a Linguagem R: Conceitos, Manipulação, Visualização, Modelagem e Elaboração de Relatórios**. Rio de Janeiro: Alta Books, 2021. 384p.

AYRES, M. et al. **BIOESTAT 3.0. Aplicações Estatísticas nas Áreas das Ciências Biológicas e Médicas**. Sociedade Civil Mamiraurá. Brasília: CNPq, 2003. 290p.

BEASLEY, C. R. Bioestatística usando R: apostila de exemplo para o biólogo. UFPA, 2004.

BRANCO, M. D.; HARNIK, S. Introdução ao R para a Análise Exploratória de dados – Parte 1. USP, 2017.

BHUGEL, R. C. **Statistics for Aquaculture**. 1ª ed. Wiley-Blackwell, 2008. 221p.

CENTENO, A. J. **Curso de Estatística Aplicada a Biologia**. 1ª ed. Centro Editorial e Gráfico. Goiânia: UFG, 1990. 188p.

CRAWLEY, M. **The R book**. 2ª ed. John Wiley & Sons, Ltd. 2013, 975p.

DALGAARD, P. **Introductory Statistics with R**. 2ª ed. Springer, 2008. 364p.

DA SILVA, B. F.; DINIZ, J.; BORTOLUZZI, M. A. Minicurso de Estatística Básica: introdução ao software R. UFSM, 2009.

FARIA, J. C.; JELIHOVSHI, E. G.; ALLAMAN, I. B. fdth: frequency distribution tables, histograms and polygons. UESC. Bahia. Brazil, 2019.

GASNIER, T. R. A Condução da Análise em Ciências Biológicas I – Base Conceitual. UFAM, 2014.

HADLEY, W. ; JENNIFER, B. (2019). readxl: Read Excel Files. R package version 1.3.1. <https://CRAN.R-project.org/package=readxl>.

HEIBERGER, R. M.; HOLLAND, B. **Statistical Analysis and Data Display: an intermediate course with examples in R**. 2ª ed. Springer, 2009. 889p.

JELIHOVSCHI, E. G. **Análise Exploratória de Dados usando o R**. Editus, 2014. 85p.

LANDEIRO, V. L. Introdução ao Uso do Programa R. INPA, 2011.

MARTINS, P. S.; ALCOFORADO, L. F. Desenvolvendo Atividades de Ensino Com o Auxílio do Software R: treinando habilidades de elaboração de gráficos com R. UFF, 2018.

MARTINS, G. A.; DONAIRE, D. **Princípios de Estatística: 900 exercícios resolvidos e propostos**. 4ª ed. ATLAS, 1990. 255p.

OGLE, D. H.; WHEELER, P.; DINNO, A. 2020. FSA: fisheries stock analysis. R Package version 0.8.31, <https://github.com/droglenc/FSA>.

PONCET, P. 2019. Modeest: mode estimation. R package version 2.4.0. <https://CRAN.R-project.org/package=modeest>.

REVELLE, W. 2020. Psych: procedures for Personality and Psychological Research, Northwestern University, Evanston, Illinois, USA, [https:// CRAN.R-project.org/package=psych=2.0.9](https://CRAN.R-project.org/package=psych=2.0.9),

R Development Core Team R: **A Language and Environment for Statistical Computing**, 2008. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-00-3. <http://www.R-project.org>.

RIBEIRO, A. J. F. (Coord.); FERREIRA, E. F.; REIS, A. I.; MONTENEGRO, L. C. C. Bioestatística Básica Usando o Ambiente Computacional R. UFMG, 2009.

SOUZA, D. V. et al. **Introdução ao R: Aplicações Florestais**. Curitiba. Editora do Autor, 2018. 136p.

TRIOLA, M. F. **Introdução à Estatística**. 1ª ed. Rio de Janeiro, LTC, 2008. 696p.

VICENTINI, A.; PERDIZ, R. Preparação de Dados para Análises Estatísticas: introdução ao R. INPA, 2020.

VIEIRA, S. **Bioestatística: Tópicos avançados**. 3ª ed. Rio de Janeiro, Elsevier, 2010. 278p.

VITAL, M. V. C. Introdução ao Uso do Software R para as Ciências Biológicas. UFAL, 2015.

WICKHAM, H. 2016. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York.

ZAR, J. H. **Biostatistical analysis**. 4ªed. New Jersey, Prentice-Hall, Inc, 1999. 663p.

## **SOBRE O AUTOR**

**ELTON NUNES BRITTO** - graduou-se em Oceanologia pela Fundação Universidade do Rio Grande – FURG/RS na cidade de Rio Grande/RS em 1998 e em Licenciatura em Ciências Biológicas pelo Centro Universitário Leonardo Da Vinci - UNIASSELVI – Indaial/SC em Manaus/AM em 2016. Concluiu o Mestrado em 2003 e o Doutorado em 2008, ambos em Biologia de Água Doce e Pesca Interior – BADPI pelo Instituto Nacional de Pesquisas da Amazônia – INPA/AM em Manaus. Durante o mestrado e doutorado cursou disciplinas de estatística aplicada trabalhando com os softwares estatísticos: STATISTICA, SYSTAT e BIOESTAT. Neste período do INPA participou também como Monitor da disciplina Fundamentos de Estatística para discentes dos cursos de Pós-graduação em Entomologia e Biotecnologia do convênio INPA/UFAM em 2006, e ministrou o curso de Estatística Aplicada a Microbiologia em 2007. Depois do doutorado entre 2008 à 2015 lecionou estatística em diversas faculdades de Manaus para discentes das áreas: Biológica, Saúde, e de Ciências Sociais. Desde de 2016 faz parte do quadro dos servidores efetivos do Instituto Federal de Educação, Ciência e Tecnologia do Pará - IFPA - Campus Santarém - PA como professor do Curso Técnico em Nível Médio em Aquicultura Subsequente ensinando dentre outras disciplinas estatística básica desde 2017. Em 2016 no IFPA também criou o Projeto de Extensão intitulado “Bioestatística Aplicada com o software BIOESTAT para biólogos e áreas afins”, ministrando cinco cursos de extensão de bioestatística aplicada, beneficiando os discentes do ensino médio do IFPA e finalistas de cursos de Graduação das Instituições de Ensino Públicas de Santarém, como a Universidade Federal do Oeste do Pará – UFOPA dos cursos de Engenharia de Pesca, Agronomia e Biologia; da Universidade Estadual do Pará – UEPA dos cursos de Biologia e Enfermagem; das Instituições Privadas como a Universidade Luterana do Brasil – ULBRA dos cursos de Biologia e Agronomia. De 2019 a 2021 cursou seis cursos de Estatística à distância sobre o software R. Em 2020 inseriu o software R ao projeto de Extensão supracitado, e em 2021 ministrou o curso intitulado “Curso de Bioestatística com Rstudio EAD” para servidores do Laboratório Federal de Defesa Agropecuária do Ministério da Agricultura Pecuária e Abastecimento – MAPA, Campinas- SP. <http://lattes.cnpq.br/7630462808118280>

**DAMIÃO PEDRO MEIRA FILHO** - graduou-se em Física pela Universidade Federal do Pará – UFPA/PA em 2003 e em Matemática pela Universidade Estadual do Pará – UEPA/PA em 2002. Concluiu o mestrado em Física Teórica pela Universidade Federal do Pará em 2006. No Mestrado estagiou no Instituto de Física Teórica da Universidade Estadual Paulista – IF/UNESP/SP. Concluiu o Doutorado em Física-Matemática pela Universidade de São Paulo – USP/SP em cooperação com Tomsk State University e Herzen State University, ambas na Rússia em 2010. Desde 2009 faz parte do quadro dos servidores efetivos como professor do Instituto Federal de Educação, Ciência e Tecnologia do Pará - IFPA - Campus Santarém – PA. Em 2010 criou o projeto de extensão intitulado “Minicursos sobre Tópicos de Física Básica e Matemática Fundamental” e lecionou a disciplina de Estatística Básica no IFPA de 2013 à 2016 e em 2018. Ensinou disciplinas das áreas da matemática e da física no

curso de Licenciatura em Matemática/Física da UFOPA de 2012 à 2016; lecionou Tópicos de Estatística e Inferência estatística, Probabilidade e Tópicos de Estatística Básica no Curso de Especialização em Matemática e Estatística pela FENATEFIB em 2015. Ensinou Estatística Física no Mestrado Profissional em Física da UFOPA e frequentou o curso de extensão de Estatística Aplicada “Curso de Estatística Aplicada utilizando o software BIOESTAT”, ambos em 2019; Ensina Tópicos em Ensino de Física para o Curso de Especialização em Ensino de Ciências e Matemática do IFPA desde 2021. Atualmente é membro docente do projeto de extensão de Bioestatística Aplicada com o software BIOESTAT e R para biólogos e áreas afins, onde dedicou-se ativamente na confecção desse caderno, contribuindo nas análises e aplicações do software R. <http://lattes.cnpq.br/2267580178817640>

**MARIA RAIMUNDA SILVA REGO** - concluiu o ensino médio em 2001 e desde 2003 faz parte do quadro de servidores públicos efetivos da Prefeitura Municipal de Santarém como Chefe do Setor de Recursos Humanos. Desde 2019 é discente do Curso Técnico Subsequente em Aquicultura do Instituto Federal de Educação Ciência e Tecnologia do Pará – IFPA, Campus Santarém e neste período do IFPA participou do III Congresso de Tecnologia e Desenvolvimento da Amazônia e da VIII Jornada Científica e Semana Integrada do IFPA. cursou a disciplina de Estatística Básica e atualmente é membro discente do projeto de extensão de Bioestatística Aplicada com o software BIOESTAT e R para biólogos e áreas afins, onde dedicou-se ativamente na confecção desse caderno, contribuindo nas análises e aplicações do software R. <http://lattes.cnpq.br/7618720047491204>

CADERNO Nº 1

ELTON NUNES BRITTO  
DAMIÃO PEDRO MEIRA FILHO  
MARIA RAIMUNDA SILVA REGO

# ESTATÍSTICA APLICADA À BIOLOGIA

USANDO O



ESTATÍSTICA DESCRITIVA E  
INFERENCIAL



[www.atenaeditora.com.br](http://www.atenaeditora.com.br)



[contato@atenaeditora.com.br](mailto:contato@atenaeditora.com.br)



[@atenaeditora](https://www.instagram.com/atenaeditora)



[facebook.com/atenaeditora.com.br](https://www.facebook.com/atenaeditora.com.br)

 **Atena**  
Editora

Ano 2021

CADERNO Nº 1

ELTON NUNES BRITTO  
DAMIÃO PEDRO MEIRA FILHO  
MARIA RAIMUNDA SILVA REGO

# ESTATÍSTICA APLICADA À BIOLOGIA

USANDO O 

ESTATÍSTICA DESCRITIVA E  
INFERENCIAL

 [www.atenaeditora.com.br](http://www.atenaeditora.com.br)

 [contato@atenaeditora.com.br](mailto:contato@atenaeditora.com.br)

 @atenaeditora

 [facebook.com/atenaeditora.com.br](https://facebook.com/atenaeditora.com.br)

 Atena  
Editora

Ano 2021