

# Aplicação para análise estatística ligada à pesquisa genética

*Application for statistical analysis linked to genetic research*

**Bruno Alves Caixeta**

Graduando do curso de Sistemas de Informação (UNIPAM)

E-mail: [brunoalvescaixeta@unipam.edu.br](mailto:brunoalvescaixeta@unipam.edu.br)

**José Correa Viana**

Docente do curso de Sistemas de Informação (UNIPAM)

E-mail: [jcorrea@unipam.edu.br](mailto:jcorrea@unipam.edu.br)

**Jeyson Césary Lopes**

Docente do curso de Ciências Biológicas (UNIPAM)

E-mail: [jeysoncl@unipam.edu.br](mailto:jeysoncl@unipam.edu.br)

---

**Resumo:** Este artigo tem como objetivo propor o uso de um *software* para facilitar e agilizar o modelo da pesquisa de compostos mutagênicos. O método utilizado atualmente no processamento e no registro dos dados resultantes de experimentos é lento e propenso a erros, dificultando o trabalho dos pesquisadores. Considerando-se esses fatores, o uso do software pode melhorar significativamente a qualidade do processo como um todo, desde os cálculos estatísticos utilizados na determinação de um composto causador de mutação genética até a armazenagem dos resultados de forma segura. Assim, os usuários poderão ter um repositório de suas pesquisas e chegar a conclusões mais concretas sobre as causas da mutagênese.

**Palavras-chave:** Mutagênese. Teste SMART. Bioinformática

**Abstract:** This article aims to propose the use of a software to facilitate and streamline the research model of mutagenic compounds. The method currently used in the processing and recording of data resulting from experiments is slow and prone to errors, making it difficult for researchers to work. Considering these factors, the use of the software can significantly improve the quality of the process as a whole, from the statistical calculations used to determine a compound that causes genetic mutation to the safe storage of results. Thus, users will be able to have a repository of their research and reach more concrete conclusions about the causes of mutagenesis.

**Keywords:** Mutagenesis. SMART Test. Bioinformatics.

---

## 1 INTRODUÇÃO

Os estudos dos processos pelos quais as células são mutadas, tanto de maneira espontânea como por agentes ambientais, datam do início do século XX. A importância desses eventos para a saúde e bem-estar da espécie humana tornou-se evidente com a

descoberta de que danos genéticos podiam ser induzidos por raios-X, gás mostarda e vários outros agentes físicos e químicos. (RIBEIRO; SALVATORI; MARQUES, 2003, p.9)

Conforme descrito por Graf *et al* (1984, p. 347), vários testes foram desenvolvidos para auxiliar as pesquisas sobre a mutação genética, entre eles está o teste SMART (*Somatic Mutation and Recombination Test*), que utiliza como cobaia a mosca *Drosophila melanogaster*. O teste consiste na aplicação de fármacos no embrião da mosca para provocar danos ao seu material genético. As células afetadas se multiplicam e se manifestam como manchas nas asas do inseto.

O teste SMART com *Drosophila melanogaster* foi primeiramente descrito por Graf e seus colaboradores em 1984. É caracterizado por ser rápido, barato, produzir resultados confiáveis e facilmente reproduzíveis (Graf *et al.*, 1984, p. 347).

Este trabalho teve como objetivo descrever o desenvolvimento de um software, com o papel de facilitador da pesquisa de fármacos com capacidades mutagênicas, realizando cálculos estatísticos. Tal objetivo permite que estudantes e pesquisadores realizem a análise de resultados de forma rápida e objetiva, minimizando a chance de erros no processamento dos resultados da observação no microscópio.

Para atingir esse objetivo, são apresentadas as seguintes funcionalidades:

- *gerar relatórios de resultados*: mostrar ao usuário um relatório informando os resultados obtidos a partir dos dados observados no teste SMART realizado por ele.
- *armazenar relatórios*: salvar os relatórios gerados para que o usuário possa consulta-los posteriormente e comparar resultados.
- *realizar cálculos estatísticos*: executar cálculos necessários para obter o resultado do teste de forma rápida e simples para o usuário.

Quanto à abordagem metodológica, o presente trabalho seguiu a linha quantitativa, pois se baseou em procedimentos sistemáticos e no levantamento de estatísticas.

## 2 REFERENCIAL TEÓRICO

Esta seção tem por objetivo tratar dos seguintes conteúdos: genética e câncer, mosca *Drosophila melanogaster* e teste SMART, análise estatística, bioinformática, *framework .Net Core* e *framework Vue.js*.

### 2.1 GENÉTICA E CÂNCER

O câncer provém de mutações genéticas em que as células passam a adquirir capacidades especiais, como descontrole na reprodução celular, evasão de apoptose,

invasão de outros tecidos e angiogênese (formação de novos vasos sanguíneos) (MARTINEZ *et al.*, 2006, p.405).

Conforme afirmam Rang *et al.* (2007, p.718), o “O câncer é caracterizado como uma das principais causas de morte nos países desenvolvidos, sendo que um terço da população será vítima dessa doença durante a vida”.

## 2.2 *Drosophila melanogaster*

Como apontado por Vogel *et al.* (1999, p.427), a *Drosophila melanogaster* é utilizada como objeto de estudo da área genética há mais de um século. Conhecida como mosca da fruta, o pequeno inseto permite identificar elementos capazes de afetar os genes.

O teste SMART de asa em *D. melanogaster* fundamenta-se na premissa de que, durante o desenvolvimento embrionário, grupos de células (os discos imaginais das asas) proliferam mitoticamente até o ponto em que se diferenciam, durante a metamorfose, em estruturas que originam as asas das moscas adultas. Esse bioensaio faz uso de dois genes marcadores para a forma dos pelos das asas: pelos múltiplos (*mwh*, 3-0.3) e pelos cujo formato lembra uma ‘chama de vela’, do inglês *flare* (*flr<sup>3</sup>*, 3-38.8) – baseando-se na indução de alterações genéticas que originam a perda de heterozigose em células larvais, que são heterozigotas para estes dois genes recessivos (Graf *et al.*, 1984, p. 347).

Quando ocorre uma alteração genética em uma das células que estão se dividindo por mitose, é formado um clone de células mutantes, que pode ser detectado fenotipicamente como uma mancha mutante na superfície das asas dos adultos. A análise das lesões induzidas é feita pela observação de grupos de células (clones mutantes), que expressam fenotipicamente os genes marcadores *flr<sup>3</sup>* ou *mwh*, responsáveis por mudanças na forma dos pelos ou tricomas (Graf *et al.*, 1984, p. 347).

## 2.3 ANÁLISE ESTATÍSTICA

O registro e análise de dados têm sido, desde muitos séculos, instrumentos essenciais à compreensão do mundo que nos rodeia. Os físicos, por exemplo, registravam os dados resultantes de suas experiências e, posteriormente, analisavam-nos em busca de uma lei que explicasse os resultados obtidos (MARTINS *et al.*, 2007, p.11).

Assim como dito por Kurichi e Sonnad (2006, p. 476), a análise estatística reportada nas pesquisas científicas proporciona aos pacientes e profissionais ligados à área da saúde a oportunidade de interpretar os dados gerados pela pesquisa e de utilizarem-nos em benefício de todos.

Para testar as hipóteses formuladas, vários testes são aconselhados: (i) o teste Binomial Condicional (Teste de Kastenbaum e Bowman) é o mais recomendado, podendo ser utilizado mesmo que o número de manchas seja pequeno; (ii) o teste Qui-quadrado para proporções pode ser utilizado somente quando o número esperado de mutações, tanto no controle como no tratado, não for menor ou igual a 5; (iii) o Teste U de Wilcoxon, Mann-Whitney com correção para as caudas, é recomendado nos casos em que a variabilidade individual (entre experimentos e/ou entre sexos) contribui significativamente para a dispersão dos dados (FREI; WÜRGLER, 1988, p. 297).

## 2.4 BIOINFORMÁTICA

Conforme descrito por Feltes *et al.* (2014, p. 3), as origens da bioinformática datam da década de 50 com a publicação da pesquisa de James Watson e Francis Crick sobre a molécula de DNA em forma de hélice. Com diversos estudos sendo realizados na área, surgiu a necessidade de visualizarem-se as moléculas de forma tridimensional para agilizar a análise delas. Isso foi possível com o desenvolvimento de programas de computador já na década de 60.

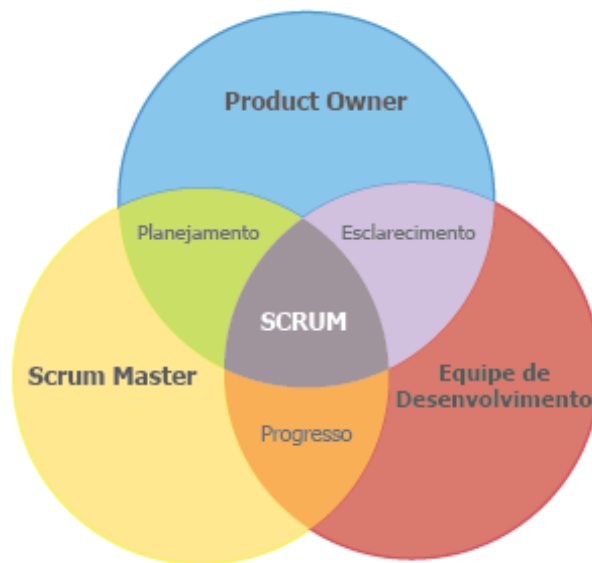
Ainda segundo Feltes *et al.* (2014, p. 3), a quantidade de dados vem aumentando em um ritmo acelerado e não dá indícios de que irá parar. Novas descobertas e estudos geram um volume maior de dados, sendo necessário que a bioinformática acompanhe, na mesma proporção, com melhorias de hardware, software e modelos de construção e análise de dados.

## 3 METODOLOGIA

A metodologia que foi utilizada para o desenvolvimento do trabalho segue padrões de metodologia ágil, mais especificamente do Framework Scrum.

Um dos conceitos-chave do Scrum é que os membros da equipe decidem, eles mesmos, como irão executar o trabalho. A responsabilidade da direção da empresa é estabelecer os objetivos estratégicos, e o trabalho da equipe é decidir como atingi-los (SUTHERLAND, 2014, p. 38).

O termo Scrum vem do jogo de rúgbi, em que o time se junta com um único objetivo: levar a bola no campo adversário. Todo o seu objetivo fica às claras, um alinhamento previsível e cuidadoso, tudo se unindo entre os membros. Uma tática perfeita para o que uma equipe deseja fazer (SUTHERLAND, 2014, p. 38). A Figura 1 apresenta os papéis presentes no *Scrum*.

**Figura 1** — Papéis do Scrum

Fonte: Elaborado pelos autores, 2019

Cada membro do time é alocado em uma função conforme apresentado na Figura 1. Um dos desenvolvedores é eleito “*Scrum Master*”, cujo papel se assemelha a um gerente de projetos. O *Scrum Master* trabalha para que o processo Scrum aconteça e para que não existam impedimentos para os membros da equipe realizarem seu trabalho (CARVALHO; MELLO, 2012, p. 557).

Outro papel importante no método é o do Dono do Produto (*Product Owner*). Esse membro do time geralmente representa o cliente (interno ou externo). Ele define quais são os requisitos e qual é o grau de importância e prioridade de cada um deles. Esse membro necessita conhecer muito bem as regras de negócios do cliente, de forma que ele possa tirar qualquer dúvida que o time possa ter em relação às funcionalidades do produto (CARVALHO; MELLO, 2012, p. 557).

Geralmente esses métodos dividem o desenvolvimento em diversas iterações de ciclos mais curtos (no caso do *Scrum* e para o presente trabalho, estes ciclos são chamados de *Sprints*) e realizam entregas ao final de cada uma delas (CARVALHO; MELLO, 2012, p. 557).

Um exemplo de organização de *Sprints* é apresentado no Quadro 1. São definidos os objetivos finais que se deseja atingir ao término da *Sprint*.

**Quadro 1** – Relação dos Sprints conforme framework Scrum

Sprint	Atividades
Sprint 1	Definir cálculos que a aplicação deve realizar.
	Implementar cálculos e testar.
Sprint 2	Definir Layout e estrutura da aplicação.
	Implementar tela inicial do aplicativo.
Sprint 3	Implementar lógica de login.
	Desenvolver tela de login.
	Definir modelo do relatório.
Sprint 4	Implementar layout do relatório.
	Definir layout da tela de relatório.
Sprint 5	Implementar tela de relatório.
	Validação das funcionalidades.

Fonte: Dados da pesquisa, 2019.

Para o desenvolvimento da aplicação, foram utilizadas as seguintes ferramentas:

- *Visual Studio Code*: IDE (Ambiente de Desenvolvimento Integrado) com recursos voltados para o desenvolvimento em diversas linguagens de programação.
- *Visual Studio 2019*: IDE (Ambiente de Desenvolvimento Integrado) oficial da *Microsoft* para desenvolvimento de aplicações utilizando *.NET*.
- *.NET Framework*: iniciativa da *Microsoft* com objetivo de ser uma plataforma única de desenvolvimento e execução de sistemas; códigos gerados para *.NET* podem ser executados em qualquer dispositivo que possua um framework dessa plataforma.
- *.NET Core*: variação da plataforma *.NET Framework* com suporte para sistemas operacionais Linux e MacOS.
- *C#*: linguagem de programação multiparadigma, de tipagem forte, desenvolvida junto da plataforma *.NET*.
- *ITextSharp*: biblioteca da linguagem C# para geração e formatação de arquivos.
- *Vue.js*: biblioteca *Javascript* para desenvolvimento de componentes reativos para interfaces web
- *Vuetify.js*: biblioteca de componentes para aplicativos *Vue.js*.
- *GIT*: sistema de controle de versão para gerenciamento do código fonte e obtenção de todo o histórico dele.

O Guia do *.NET Core* é uma plataforma de desenvolvimento de código aberto de uso geral mantida pela Microsoft e pela comunidade *.NET* no *GitHub*. É uma plataforma cruzada (compatível com *Windows*, *macOS* e *Linux*), que pode ser usada no desenvolvimento de dispositivos, na nuvem (*.NET CORE*, 2019).

A versão mais recente é o *.NET Core 3,0*. Os novos recursos incluem o suporte à área de trabalho do Windows com o Windows Presentation Foundation (C# WPF) e o Windows Forms, o desenvolvimento da Web de pilha completo com mais de um novo C# C# e muito mais (*.NET CORE*, 2019).

Vue (pronuncia-se /vju:/, como *view*, em inglês) é um framework progressivo para a construção de interfaces de usuário. Ao contrário de outros frameworks monolíticos, Vue foi projetado desde sua concepção para ser adotável incrementalmente. A biblioteca principal é focada exclusivamente na camada visual (view layer), sendo fácil adotar e integrar com outras bibliotecas ou projetos existentes. Por outro lado, Vue também é perfeitamente capaz de dar poder a sofisticadas Single-Page Applications quando usado em conjunto com ferramentas modernas e bibliotecas de apoio (VUE.JS, 2019).

O sistema de componentes também é outro importante conceito no Vue, por ser uma abstração que proporciona a construção de aplicações de larga escala, compostas por pequenos componentes, autocontidos e frequentemente reutilizáveis. Geralmente qualquer tipo de interface de uma aplicação pode ser abstraída em uma árvore de componentes. No Vue, um componente é essencialmente uma instância Vue com opções predefinidas. (VUE.JS, 2019).

#### 4 DESENVOLVIMENTO E RESULTADOS

Na fase inicial do trabalho, foi necessário analisar o fluxo utilizado para processar os resultados dos testes realizados com as moscas e realizar pesquisas sobre os conceitos ligados à área da biologia envolvidos no desenvolvimento para garantir que o resultado final fosse compreendido com clareza e exatidão pelos usuários.

As moscas são expostas ao composto que se deseja estudar ainda durante a sua fase embrionária. Ao se tornarem adultas, suas asas são observadas com a ajuda de um microscópio, buscando manchas causadas por mutação genética.

A partir da análise ao microscópio, foi montado um arquivo de texto conforme a Figura 2. Cada linha do arquivo representa uma asa de uma mosca, apresentando a seção da asa e a quantidade de manchas de cada tipo identificadas nela. As linhas em branco apontam asas em que não foram identificadas manchas.

**Figura 2** — Arquivo de texto para extração de dados

```
* Tratamento: Descendentes MH do cruzamento ST controle negativo H20
* Lâmina 1
*
1
2
3
4
5
6
7 CD1-0
8
9
10 E2-0
11
12
13
14
15 D3-0
16 C1-0
17
18
19 A1-0 C1-0 C2-0 C2-0 CD2-0 C2-0 E2-0 E1-0
20 B1-0 B1-0 B2-0 C2-0 C1-0 C1-0 D1-0 E1-0 E1-0 E2-0 E1-0 E1-0
* Tratamento: Descendentes MH do cruzamento ST controle negativo H20
* Lâmina 2
*
1 D1-0 D83-0 E1-0
2
3
4 E2-0 E3-0
5
6
7 E25-0
8
9
10
11
12
13
14
15 C4-0
16 C1-0
17
18
19
20
* Tratamento: Descendentes MH do cruzamento ST controle negativo H20
* Lâmina 3
*
1
2 B1-0
3 A14-0
4
```

**Fonte:** Dados da pesquisa, 2019.

Os dados do arquivo de texto eram então transferidos para uma tabela *Excel*, onde eram executadas duas macros responsáveis por contar as manchas de cada tipo e separá-las em classes de acordo com regras que eram desconhecidas dos próprios pesquisadores. Em seguida, essas informações eram inseridas em uma segunda tabela para serem realizados cálculos estatísticos e para se chegar a uma conclusão se o composto testado era mutagênico. Esse método se mostrou ineficiente por ser desnecessariamente complexo e pouco intuitivo.



Muitas vezes, os arquivos de texto eram excluídos ou perdidos, assim como os resultados. Esse tipo de pesquisa é frequentemente usado por alunos de ciências biológicas em seus trabalhos e os professores precisavam acompanhar o processo com muita atenção para que não fossem cometidos erros na utilização das tabelas.

As tabelas foram criadas no ano de 1999, e a lógica responsável pela classificação estava protegida por uma senha que se perdeu com o tempo. Para acessá-la, foi necessário quebrar a segurança padrão do *Excel*. Através de uma técnica que consiste em converter as tabelas em arquivos hexadecimais e remover ou alterar as linhas que contêm a senha, foi possível acessar os códigos na linguagem *Visual Basic*, que continham as lógicas para reproduzi-las na aplicação.

Após se obterem as regras das tabelas, foi criada uma aplicação em C# para receber os dados. A pedido da equipe de pesquisa, o uso do arquivo de texto da Figura 2 foi mantido por ser um modelo padronizado que é repassado aos alunos no laboratório. Dessa forma, a aplicação faz a leitura e o processamento do arquivo, salvando os resultados em um banco de dados relacional.

A Figura 3 mostra os dados obtidos do arquivo da Figura 2, em que MSP, MSG e MG são tipos de mancha e cada mancha é atribuída a uma classe (de 1 a 10) de acordo a quantidade de células do tipo MSP. É possível também marcar o resultado como um controle, indicando que será utilizado como base para determinar se algum composto é mutagênico em comparação a ele.

**Figura 3** — Resultado da Extração

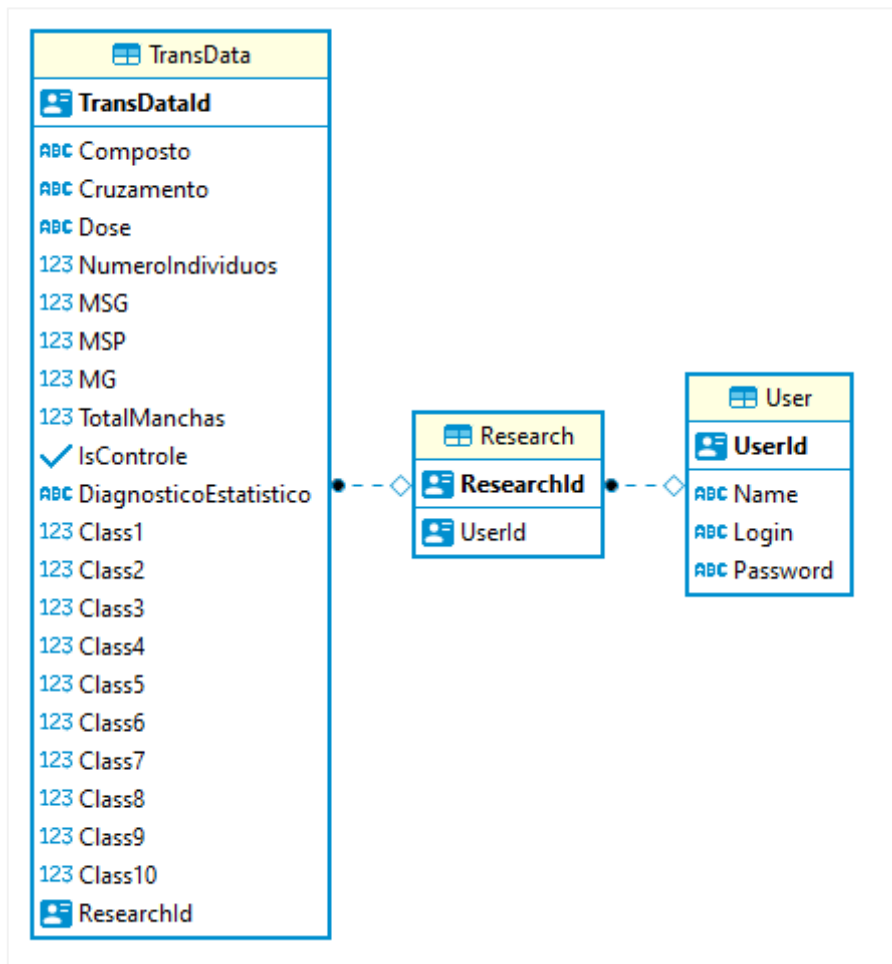
```
"msg": 9,  
"msp": 61,  
"mg": 0,  
"totalManchas": 70,  
"isControle": false,  
"diagnosticoEstatistico": null,  
"class1": 37,  
"class2": 24,  
"class3": 6,  
"class4": 0,  
"class5": 1,  
"class6": 1,  
"class7": 0,  
"class8": 1,  
"class9": 0,  
"class10": 0,
```

**Fonte:** Dados da pesquisa, 2019.

Para atribuírem-se os resultados a um usuário específico e agrupá-los de acordo com o estudo realizado, foi criado o modelo relacional exposto na Figura 4.

Cada usuário pode criar várias *Researches* (pesquisas) que são formadas por várias *TransData* (dados transformados). Dessa maneira, é possível, por exemplo, testar o mesmo composto com diferentes concentrações e comparar os resultados dentro da mesma pesquisa, além de melhorar a organização.

**Figura 4** – Modelo do Banco Relacional



Fonte: Dados da pesquisa, 2019.

Após criar uma pesquisa e incluir os dados referentes a ela, o usuário pode gerar um relatório no formato PDF com os resultados de cálculos estatísticos de distribuição binomial, como exposto na Figura 5. Caso os dados de análise sejam alterados, basta que o usuário gere um novo relatório.



A aplicação foi desenvolvida com foco no teste SMART, porém a proposta de informatização poderá futuramente abranger outros tipos de experimentos das ciências biológicas, sendo necessária a implementação das funcionalidades requeridas pelo teste que se deseja incluir na aplicação.

Os objetivos deste trabalho foram alcançados, uma vez que o uso das tabelas *Excel* foi removido do fluxo de trabalho, e os resultados obtidos foram os mesmos, com maior praticidade e menor risco de erro.

O desafio do trabalho consistiu na interpretação de todos os cálculos estatísticos utilizados para se obter o resultado final, já que, em alguns casos, valores arredondados ou fatores em ordem errada podem afetar a conclusão sobre a natureza do composto analisado. Foram realizados testes com vários dados para validar os resultados, todos obtiveram sucesso, comprovando a eficácia da aplicação em reproduzir o fluxo das tabelas.

A mutagênese está atrelada ao desenvolvimento de diversos tipos de câncer, uma das maiores causas de morte da atualidade. A busca por um tratamento que seja menos agressivo e que erradique completamente a doença é um dos grandes desafios da ciência moderna.

A quantidade de pesquisas cresce, acompanhando a revelação de novas descobertas. Assim sendo, aumenta-se a necessidade de se processarem os dados rapidamente e de se armazenarem os resultados de modo organizado e de fácil acesso.

## REFERÊNCIAS

.NET CORE. **Guia do .NET Core**. 2019. Disponível em: <https://docs.microsoft.com/pt-br/dotnet/core/>.

CARVALHO, B. V.; MELLO, C. H. P. Aplicação do método ágil *scrum* no desenvolvimento de produtos de *software* em uma pequena empresa de base tecnológica. **Gest. Prod.**, São Carlos, v.19, n. 3, p. 557-573, 2012.

FELTES, B. C. *et al.* Origens. **Bioinformática: da biologia à flexibilidade molecular**. São Paulo: Sociedade Brasileira de Bioquímica e Biologia Molecular, 2014. v.1, p. 3-6.

FREI, H.; WÜRGLER, F. E. Statistical methods to decide whether mutagenicity test data from *Drosophila* assays indicate positive, negative or inconclusive result. **Mutation Research**, v. 203, p.297-308, 1988.

GRAF, U.; WURGLER, F. E.; KATZ, A. J.; FREI, H.; JUON, H.; HALL, C. B.; KALE, P. G. Somatic Mutation and Recombination Test in *Drosophila melanogaster*. **Environmental Mutagen**, v. 6, p. 347-377, 1984.

KURICHI, J. E.; SONNAD, S. S. Statistical methods in the Surgical Literature. **Journal of the American College Surgeons**, v. 202, p.476-484, 2006.

MARTINEZ, M. A. R. *et al.* Molecular genetics of non-melanoma skin câncer. **Anais Brasileiros de Dermatologia**, v. 8, n. 5, p. 405-419, 2006.

MARTINS, M. E. G *et. al.* Dados e variáveis. *In: Análise de dados: texto de apoio para os professores do 1.º ciclo.* Lisboa: Ministério da Educação, 2007.

RANG, H. P. *et al.* Quimioterapia do câncer. **Farmacologia**, v. 6, p. 718-736, 2007.

RIBEIRO, L. R.; SALVATORI, D. M. F.; MARQUES, E. K. **Mutagênese Ambiental.** Canoas: Editora da ULBRA, 2003.

SUTHERLAND, J. **Scrum: a arte de fazer o dobro de trabalho na metade do tempo.** São Paulo: LeYa, 2014

VOGEL, E. W.; GRAF, U.; FREI, H. J.; NIVERD, M. M. J. The results of assay in *Drosophila* of exposure to carcinogens. **International for research on Cancer**, v.146, p. 427-470, 1999.

VUE.JS. **Introdução.** Guia do Vue.js, 2019. Disponível em: <https://br.vuejs.org/v2/guide/index.html>.