

# 10° EiC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de  
dezembro de 2025 no  
IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

## DESENVOLVIMENTO DE API PARA AVALIAÇÃO DE RISCO DE MORTALIDADE POR DOENÇA DE CHAGAS COM APRENDIZADO DE MÁQUINA

**RIBEIRO, Alessandro Carneiro <sup>1</sup>; GONÇALVES, Arthur Márcio Silva <sup>2</sup>; PEREIRA, Edmar  
de Brito <sup>2</sup>; JESUS, Renata Morgana de <sup>2</sup>**

<sup>1</sup>Docente do Curso de Sistemas de Informação do IFNMG Campus Pirapora;

<sup>2</sup> Discente do Curso de Sistemas de Informação do IFNMG Campus Pirapora;

**Palavras-chave:** Doença de Chagas. Predição de Mortalidade. API. Aprendizado de Máquina. Regressão Logística.

### Introdução

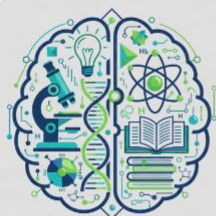
A Doença de Chagas, causada pelo protozoário *Trypanosoma cruzi*, é uma doença tropical negligenciada de grande impacto social e epidemiológico na América Latina, cuja distribuição e persistência no Brasil se relacionam à ocupação territorial, desigualdades sociais e moradias rurais precárias, com domiciliação de triatomíneos favorecendo a transmissão vetorial (VINHAES; DIAS, 2000). Ao mesmo tempo em que políticas de controle vetorial e vigilância reduziram parte do problema, as formas crônicas ainda evoluem para cardiopatias graves e risco de morte, exigindo estratégias que permitam identificar precocemente pacientes de maior risco (VINHAES; DIAS, 2000).

O prognóstico permanece baseado em procedimentos clínicos tradicionais, exames complementares e na experiência do profissional, enquanto métodos avançados, como a ressonância magnética cardíaca, não estão amplamente disponíveis, mantendo muitos serviços dependentes de abordagens convencionais (SIMÕES et al., 2018). Mesmo quando biomarcadores, dados eletrocardiográficos e variáveis clínicas existem, costumam ser avaliados de forma isolada, o que mantém o prognóstico fortemente sujeito à interpretação individual e dificulta a padronização na estratificação de risco (SIMÕES et al., 2018).

Diante disso, ganha relevância o uso de ferramentas que integrem de forma sistemática informações clínicas, laboratoriais e eletrocardiográficas, favorecendo modelos prognósticos mais objetivos e reprodutíveis; nesse cenário, técnicas de aprendizado de máquina (ML) mostram potencial para apoiar a estratificação de risco em saúde (NUNES, 2018). Entre os métodos disponíveis, a regressão logística se destaca pela interpretabilidade e ampla aceitação em estudos clínicos e epidemiológicos, permitindo estimar a probabilidade de desfechos binários, como mortalidade em Doença de Chagas, a partir de múltiplos preditores.

Sua capacidade de converter marcadores clínicos, exames e dados do paciente em probabilidades claras de risco contribui para avaliações mais padronizadas e menos dependentes da interpretação isolada do profissional. Para que essa estimativa seja utilizável na prática, o modelo foi incorporado a uma Interface de Programação de Aplicativos (API) (KOTSTEIN, 2021) desenvolvida em Python com o micro-framework Flask e containerizada com Docker, de modo a padronizar o ambiente de execução e facilitar a implantação em diferentes sistemas.

Assim, o objetivo central do projeto é desenvolver, implementar e containerizar uma API preditiva baseada em regressão logística para estimar o risco de mortalidade em pacientes com Doença de Chagas, oferecendo uma ferramenta tecnológica de apoio à decisão clínica e à estratificação de risco, com potencial para melhorar o acompanhamento dos pacientes e subsidiar futuras ações em saúde pública.



# 10° EiC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de  
dezembro de 2025 no  
IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

## Material e métodos /Metodologia

A metodologia adotada para o desenvolvimento da *API* de predição de risco de mortalidade em pacientes com Doença de Chagas baseia-se em um processo estruturado desde a aquisição e tratamento dos dados até a implementação do modelo preditivo e sua integração em um serviço *WEB*. Para o treinamento do modelo de *ML* adotado neste projeto foram utilizados dados do estudo denominado SaMi-Trop (Ferreira et al., 2022). O modelo de Regressão Logística utilizado no desenvolvimento da *API* foi descrito pelo trabalho de Ribeiro e Santos (2024), que apresentou resultados expressivos na predição do risco de mortalidade por doença de Chagas.

**Construção do modelo preditivo de Regressão Logística:** O processo de construção do modelo proposto por Ribeiro e Santos (2024), será apresentado a seguir. Vale destacar que para teste e validação da *API* foram descartados os dados de treinamento do modelo, sendo utilizados apenas os dados separados para teste e validação do modelo.

### *Base de Dados*

Os dados clínicos utilizados neste trabalho foram obtidos do estudo multicêntrico SaMi-Trop (Ferreira et al., 2022), que incluiu 2.161 pacientes com Doença de Chagas em Minas Gerais, acompanhados por entrevistas, coletas de sangue e exames de ECG em unidades básicas de saúde. Após pré-processamento, ficou disponível um conjunto com 1.637 pacientes, incluindo 134 óbitos, e 83 variáveis clínicas e sociodemográficas selecionadas.

### *Pré-processamento dos Dados*

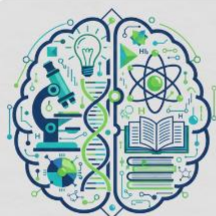
- Foi realizada a limpeza e a preparação dos dados para uso nos modelos de aprendizado de máquina, incluindo:
- Tratamento de valores ausentes por imputação (média para dados contínuos e moda para dados discretos).
- Exclusão de variáveis duplicadas ou de baixa relevância.
- Normalização dos dados numéricos e transformação de variáveis categóricas em variáveis dummy.
- Ferramentas como Pandas e Scikit-learn (PEDREGOSA et al., 2011) foram utilizadas para estas operações.

### *Seleção e Estratificação das Variáveis*

O método Random Forest foi utilizado para identificar os atributos mais relevantes para a predição da mortalidade, com destaque para o biomarcador NT-proBNP. Os dados foram estratificados por gênero para permitir análise diferenciada e desenvolvimento de modelos específicos para subgrupos.

### *Divisão dos Dados*

O conjunto de dados foi dividido em treino (75%) e teste (25%), utilizando a função *train\_test\_split* da biblioteca *Scikit-learn* (PEDREGOSA et al., 2011), garantindo que o modelo fosse avaliado em dados independentes para evitar o *overfitting*.



# 10° EIC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de  
dezembro de 2025 no  
IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

### *Balanceamento das Classes*

Devido ao desbalanceamento significativo entre os casos de morte e sobrevivência, foram aplicadas técnicas de *oversampling*, especialmente variações do *SMOTE* (CHAWLA et al., 2002), como *Borderline-SMOTE*, *ADASYN* e *SMOTEENN*, para aumentar a representatividade da classe minoritária e melhorar a capacidade preditiva dos modelos.

**Desenvolvimento do Modelo de Regressão Logística:** O modelo preditivo foi desenvolvido utilizando a técnica de regressão logística, adequada para predição de eventos binários com alta interpretabilidade (HOSMER et al., 2013). O modelo foi treinado com os dados balanceados e otimizados por validação cruzada e *Randomized Search* (BERGSTRA, 2012) para escolha dos melhores hiperparâmetros.

Dos 1152 modelos desenvolvidos por Ribeiro e Santos (2024), foi selecionado o modelo de Regressão Logística que obteve o melhor desempenho com o conjunto de dados de teste. O mesmo apresentou uma G-média de 73%, entre especificidade e sensibilidade, para o conjunto de dados geral (masculino e feminino) combinado com a técnica de balanceamento de dados *SMOTEENN* (CHAWLA, 2002).

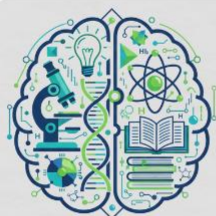
A Tabela 1 apresenta a lista de atributos contidos no modelo que foram utilizados na implementação da *API*.

**Desenvolvimento da API:** A *API* foi desenvolvida com o objetivo de disponibilizar o modelo de Regressão Logística em um ambiente acessível e com potencial de integração com outros sistemas, permitindo que a estimativa de risco fosse realizada de forma padronizada. Para isso, utilizou-se o *micro-framework Flask*, que oferece uma estrutura leve e adequada para aplicações web baseadas no padrão *REST*.

No arquivo *init.py*, é criada a instância principal da aplicação *Flask* e registrado o *blueprint* responsável pelas rotas, o que contribui para a organização e modularização do sistema. As rotas foram implementadas no arquivo *routes.py*, onde se destacam duas funcionalidades: a rota inicial (/), responsável por carregar a interface web em *HTML*, e a rota */predict*, que executa o processo de predição.

A rota de predição foi configurada para receber requisições *POST* contendo os dados clínicos em formato *JSON*. Esses dados correspondem aos dez atributos utilizados pelo modelo preditivo: NT-proBNP, Fumante, Duração do QRS, Gênero, Classe Funcional, Estado Civil, Renda, Escolaridade, Idade e Hipertensão arterial. Após o recebimento da requisição, os valores são convertidos para o formato numérico e organizados conforme a estrutura exigida pelo modelo.

O arquivo *snn\_dc\_melhorrl\_rf10.pkl*, que contém o modelo previamente treinado, é carregado dinamicamente com a biblioteca *joblib*, e a previsão é realizada por meio da função *model.predict()*. O resultado é devolvido em formato *JSON*, permitindo que a interface *WEB* exiba ao usuário a indicação de risco, “0” para baixo risco e “1” para necessidade de atenção. Esse retorno é representado visualmente por cores e mensagens, conforme programado no arquivo *index.html* e demonstrado na Figura 1.



# 10º EIC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de  
dezembro de 2025 no  
IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

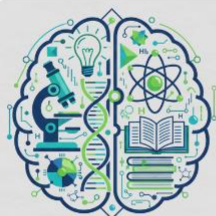
**Tabela 1 - Atributos definidos pelo modelo de ML Regressão Logística**

| Atributo             | Tipo                   | Domínio                                                                                             |
|----------------------|------------------------|-----------------------------------------------------------------------------------------------------|
| NT-proBNP            | Binário                | Até 300 pg/dl - 0<br>Acima de 300 pg/dl - 1                                                         |
| Fumante              | Binário                | Nunca fumou ou já fumou, mas não fuma mais - 0<br>Atualmente fuma - 1                               |
| Duração do QRS       | Binário                | Inalterado - 0<br>Alterado - 1                                                                      |
| Gênero               | Binário                | Masculino - 0<br>Feminino - 1                                                                       |
| Classe Funcional     | Binário                | É capaz de realizar atividades diárias - 0<br>Não é capaz de realizar atividades diárias - 1        |
| Estado Civil         | Binário                | Solteiro (sem companheiro/a) - 0<br>Casado (com companheiro/a) - 1                                  |
| Renda per capita     | Binário                | Até meio salário mínimo por membro da família - 0<br>Acima de meio mínimo por membro da família - 1 |
| Escolaridade         | Categórico<br>numérico | Alfabetizado - 0<br>Ensino Fundamental - 1<br>Ensino Médio - 2                                      |
| Idade                | Binário                | Até 61 anos - 0<br>Acima de 61 anos - 1                                                             |
| Hipertensão arterial | Binário                | Não - 0<br>Sim - 1                                                                                  |

**Fonte: Desenvolvido pelos autores.**

Essa comunicação ocorre de forma integrada com a interface *WEB*, que não apenas apresenta o resultado ao usuário, mas também envia os dados clínicos para a *API*. Quando o formulário é preenchido, um *script JavaScript* coleta as informações e realiza uma requisição *POST* através da função *fetch()* para a rota */predict*, enviando elas em formato *JSON*. A resposta retorna à própria página, permitindo a atualização imediata do resultado sem recarregamento, o que torna a interação mais fluida e facilita os testes da aplicação.

Para garantir reprodutibilidade e facilitar a implantação da aplicação em diferentes ambientes, a *API* foi containerizada utilizando *Docker*. O arquivo *Dockerfile* define a imagem base *python:3.8-slim* e instala todas as dependências necessárias, incluindo *Flask*, *numpy*, *scikit-learn* e *joblib*. A porta 5000 é exposta para permitir o acesso externo à aplicação. O gerenciamento do contêiner foi feito por meio do *Docker Compose*.



# 10° EiC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de dezembro de 2025 no IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

**Figura 1** - Interface da API apresentando as duas possibilidades de resultado

| <b>Avaliação de Risco de Mortalidade por Doença de Chagas</b>                                                                                                            |  |                                                               |                                                                                                                                                                          |  |                                                        |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|---------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|--------------------------------------------------------|
| <b>Dados do paciente:</b>                                                                                                                                                |  | <b>Resultado:</b>                                             | <b>Dados do paciente:</b>                                                                                                                                                |  | <b>Resultado:</b>                                      |
| <b>NT-proBNP</b><br><input checked="" type="radio"/> < 300 pg/dL<br><input type="radio"/> ≥ 300 pg/dL                                                                    |  | <b>Necessidade de atenção quanto aos elementos avaliados!</b> | <b>NT-proBNP</b><br><input type="radio"/> < 300 pg/dL<br><input checked="" type="radio"/> ≥ 300 pg/dL                                                                    |  | <b>Sem preocupação quanto aos elementos avaliados!</b> |
| <b>Fumante</b><br><input checked="" type="radio"/> Atualmente fuma.<br><input type="radio"/> Nunca fumou ou já fumou, mas não fuma mais.                                 |  |                                                               | <b>Fumante</b><br><input type="radio"/> Atualmente fuma.<br><input checked="" type="radio"/> Nunca fumou ou já fumou, mas não fuma mais.                                 |  |                                                        |
| <b>Duração QRS</b><br><input type="radio"/> Inalterado.<br><input checked="" type="radio"/> Alterado.                                                                    |  |                                                               | <b>Duração QRS</b><br><input checked="" type="radio"/> Inalterado.<br><input type="radio"/> Alterado.                                                                    |  |                                                        |
| <b>Gênero</b><br><input checked="" type="radio"/> Masculino.<br><input type="radio"/> Feminino.                                                                          |  |                                                               | <b>Gênero</b><br><input type="radio"/> Masculino.<br><input checked="" type="radio"/> Feminino.                                                                          |  |                                                        |
| <b>Classe Funcional</b><br><input type="radio"/> É capaz de realizar atividades diárias.<br><input checked="" type="radio"/> Não é capaz de realizar atividades diárias. |  |                                                               | <b>Classe Funcional</b><br><input checked="" type="radio"/> É capaz de realizar atividades diárias.<br><input type="radio"/> Não é capaz de realizar atividades diárias. |  |                                                        |
| <b>Estado Civil</b><br><input checked="" type="radio"/> Solteiro (sem companheiro/a).                                                                                    |  |                                                               | <b>Estado Civil</b><br><input type="radio"/> Solteiro (sem companheiro/a).                                                                                               |  |                                                        |

**Fonte:** Elaborado pelos autores

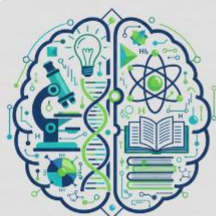
A orquestração é realizada pelo arquivo *docker-compose.yml*, que configura o serviço *WEB*, mapeia o diretório local para dentro do contêiner e libera a porta 5000. Essa abordagem garante que a aplicação possa ser executada em diferentes máquinas sem conflitos de dependências, o que é especialmente útil em ambientes clínicos ou de pesquisa.

## Resultados e discussão

O desenvolvimento da *API* permitiu transformar o modelo de Regressão Logística treinado em uma aplicação funcional e acessível para estimativa de risco em pacientes com Doença de Chagas. A integração das tecnologias utilizadas mostrou-se eficiente, especialmente devido à organização modular da aplicação em *Flask*, onde o arquivo *\_init\_.py* cria a instância da aplicação e registra o conjunto de rotas responsáveis tanto pela interface web quanto pela predição.

A rota */predict*, configurada para receber requisições POST no formato JSON, demonstrou funcionamento adequado ao processar corretamente os dez atributos clínicos utilizados pelo modelo. Após o recebimento dos dados, a *API* converte os valores, organiza-os na ordem esperada e carrega dinamicamente o modelo armazenado no arquivo *snn\_dc\_melhor\_rf10.pkl* utilizando a biblioteca *joblib*. A predição retorna valores padronizados, "0" ou "1", o que facilita a interpretação e futura integração com outros sistemas.

Além da resposta em *JSON*, a aplicação disponibiliza uma interface web acessível pela rota inicial (*/*), permitindo que o usuário realize a predição diretamente no navegador. Essa interface foi estruturada em *HTML* e estilizada com *CSS*, incluindo elementos visuais que indicam o resultado da predição por meio de cores: verde para baixo risco e vermelho para necessidade de atenção. Essa



# 10º EIC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de  
dezembro de 2025 no  
IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

apresentação visual contribuiu para validar o funcionamento da *API* durante os testes, permitindo acompanhar todas as etapas do processo de forma clara.

A containerização da aplicação com *Docker* também apresentou resultados positivos, garantindo que o ambiente de execução permanecesse consistente independentemente do sistema onde fosse executado. O *Dockerfile* definiu a imagem base *python:3.8-slim* e instalou as dependências necessárias, enquanto o *docker-compose.yml* organizou a execução do contêiner e expôs a porta 5000, permitindo acesso local à aplicação. Essa abordagem facilitou a execução da *API* sem conflitos de dependências, demonstrando portabilidade e reprodutibilidade do ambiente.

### Considerações finais

A aplicação desenvolvida atingiu o objetivo de disponibilizar o modelo preditivo em um formato acessível e padronizado, integrando de maneira funcional a lógica de predição com uma interface *WEB* e um ambiente de execução containerizado. A utilização do *Flask* permitiu estruturar a *API* de forma organizada, enquanto o retorno padronizado em *JSON* possibilitará sua integração com outros sistemas de informação em saúde.

A inclusão de uma interface web simples, porém funcional, ampliou a usabilidade da ferramenta, permitindo que a predição fosse realizada sem necessidade de conhecimentos técnicos avançados. A containerização com *Docker* tornou o processo de implantação mais simples e consistente, permitindo que a aplicação fosse executada em diferentes máquinas sem conflitos de dependências.

Como continuidade do trabalho, destaca-se a possibilidade de aprimorar a *API* para suportar a integração de algoritmos preditivos mais avançados além da Regressão Logística. A estrutura atual, que realiza o carregamento dinâmico do modelo por meio da biblioteca *joblib*, permite evoluções no sentido de incorporar novos algoritmos treinados, como Redes Neurais ou Gradient Boosting, sem alterações profundas na arquitetura. Isso possibilitaria comparar desempenhos, selecionar automaticamente o modelo mais adequado ou permitir que o usuário escolha qual modelo utilizar, aumentando a flexibilidade e o potencial de aplicação clínica da ferramenta.

Também se considera relevante aprimorar a interface gráfica para torná-la ainda mais intuitiva e ampliar a integração com sistemas externos, bem como realizar testes em ambientes reais de uso, garantindo maior confiabilidade e aplicabilidade clínica da solução.

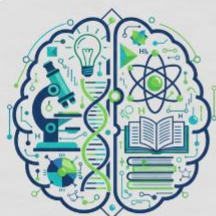
### Agradecimentos

Agradecemos ao Instituto Federal do Norte de Minas Gerais - *Campus* Pirapora, pelo apoio institucional e pela infraestrutura disponibilizada para o desenvolvimento deste projeto. Prestamos também nosso agradecimento ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo suporte financeiro, por meio de bolsa de Iniciação Científica, que viabilizou a execução desta pesquisa e contribuiu para o avanço acadêmico e tecnológico aqui apresentado.

### Referências

BERGSTRA, J.; BENGIO, Y. "Random search for hyper-parameter optimization." *Journal of Machine Learning Research*, v. 13, p. 281-305, 2012.

CHAWLA, N. V., BOWYER, K. W., HALL, L. O., & KEGELMEYER, W. P. (2002), "SMOTE: synthetic minority over-sampling technique", *Journal of artificial intelligence research*, 16, 321-357.



# 10° EiC

## Encontro de Iniciação Científica

09, 10, 11, 12 e 13 de  
dezembro de 2025 no  
IFNMG-Campus Pirapora



Realização da Coordenadoria de Pesquisa, Pós-Graduação e Inovação do IFNMG-Campus Pirapora

FERREIRA, A. M., et al. (2022). “Two-year death prediction models among patients with Chagas Disease using machine learning-based methods”, PLoS Neglected Tropical Diseases, 16(4), e0010356. doi:10.1371/journal.pntd.0010356.

KOTSTEIN, Sebastian; et al. Which RESTful API design rules are important and how do they improve software quality? A Delphi study. arXiv preprint arXiv:2108.00033, 2021.

HOSMER, D. W., LEMESHOW, S., & STURDIVANT, R. X. (2013), Applied Logistic Regression. John Wiley & Sons.

NUNES, I. C. Comparação de algoritmos de aprendizado de máquina para classificação de pacientes com risco de doença coronariana. 2018. 156 f. Dissertação (Mestrado em Engenharia Biomédica) – Universidade Federal de Minas Gerais, Belo Horizonte, 2018.

PEDREGOSA, F., et al. (2011). “Scikit-learn: Machine learning in Python”, The Journal of Machine Learning Research, 12, 2825–2830.

SIMÕES, M. V. et al. Chagas Disease Cardiomyopathy. International Journal of Cardiovascular Sciences, v. 31, n. 2, 2018. DOI: 10.5935/2359-4802.20180011.

RIBEIRO, Alessandro Carneiro; SANTOS, Laércio Ives. Predição de mortalidade na doença de chagas: impacto do balanceamento de dados e do biomarcador nt-probnp no desempenho de modelos de aprendizado de máquina.. In: Anais do Encontro Nacional de Modelagem Computacional e Encontro de Ciência e Tecnologia de Materiais. Anais, 2024. Disponível em: <https://www.even3.com.br/anais/enmc2024/915121-PREDICAO-DE-MORTALIDADE-NA-DOENCA-DE-CHAGAS--IMPACTO-DO-BALANCEAMENTO-DE-DADOS-E-DO-BIOMARCADOR-NT-PROBNP-NO-DESE>. Acesso em: 02/12/2025

VINHAES, M. C.; DIAS, J. C. P. Doença de Chagas no Brasil. Cadernos de Saúde Pública, v. 16, p. S7–S12, 2000. DOI: 10.1590/S0102-311X2000000800002.