

Centro Federal de Educação Tecnológica de Minas Gerais  
Belo Horizonte

# USO DE APRENDIZADO DE MÁQUINA PARA PREVISÃO DE PROVÁVEIS PACIENTES DESENVOLVEREM AVC

**Aluna: Gabriela Campos Gama**  
**Orientador: Prof. Dr. Edson Marchetti da Silva**

Engenharia de Computação

# Visão Geral

## Principais seções desta apresentação

Introdução

Contextualização

Objetivos

Justificativas e Contribuições

Fundamentação Teórica

Trabalhos Relacionados

Metodologia

Desenvolvimento

Implementação

Análise dos Resultados

Conclusão

Trabalhos Futuros

Referências

# Introdução

- De acordo com a OMS, o AVC é o segundo maior causador de morte do mundo.
- O melhor tratamento para o AVC ainda é a prevenção.
- A aplicação de técnicas de AM pode ajudar a identificar a ocorrência de AVC, analisando os dados dos pacientes a fim de detectar padrões e fazer previsões.



# Contextualização do estudo

Este trabalho consiste em diferentes etapas para a análise, tratamento e criação de um modelo para predição de casos AVC.

- Escolha de um dataset disponibilizado no Kaggle.
- Preparação e pré-processamento dos dados.
- Escolha de diferentes métodos de AM para classificação.
- Teinar e realizar a avaliação.
- Aplicar um método ensemble e realizar nova avaliação.
- Disponibilizar o novo modelo em um site.

# Objetivos

## **Primeiro objetivo**

Desenvolver um modelo utilizando de técnicas atuais de AM, capazes de fazer a predição de AVC.

---

## **Segundo objetivo**

Comparar e analisar os resultados dos modelos utilizados no trabalho, de forma a avaliar o desempenho.

---

## **Terceiro objetivo**

Disponibilizar o modelo em uma página web.

# Justificativas e Contribuições

O trabalho justifica-se como um estudo que aplica técnicas de AM, usando como base de dados informações de paciente reais para treinar os modelos, obtendo resultados que podem contribuir para identificação de possíveis pessoas serem propensas a ocorrer ou não um AVC.

No âmbito de contribuições, o resultado deste trabalho poderá auxiliar profissionais da área da saúde a identificarem indivíduos propensos a desenvolverem um derrame cerebral, e tomar as devidas medidas para se obter um tratamento bem sucedido.

# Fundamentação teórica

---

## Visão Geral

---

- Aprendizado de máquina
  - Aprendizado supervisionado
- Algoritmos de classificação
- Modelos Multiplos Preditivos
- Generalização em Pilha
- Validação Cruzada

# Aprendizado de Máquina

---

- É uma subárea da Inteligência Artificial.
- Estuda métodos capazes de extrair informações de uma base de dados e usá-las para classificar ou prever novos valores.
- É possível prever valores a partir de dados novos, desde que esteja no domínio em que foi treinado.

Conforme o autor Faceli cita, existem algumas aplicações de AM para problemas corriqueiros:

- reconhecimento de palavras faladas;
- predição de taxa de cura de pacientes;
- detecção de uso fraudulento de cartão de crédito;
- diagnóstico de câncer por meio da análise de dados.



# Aprendizado de Máquina Supervisionado

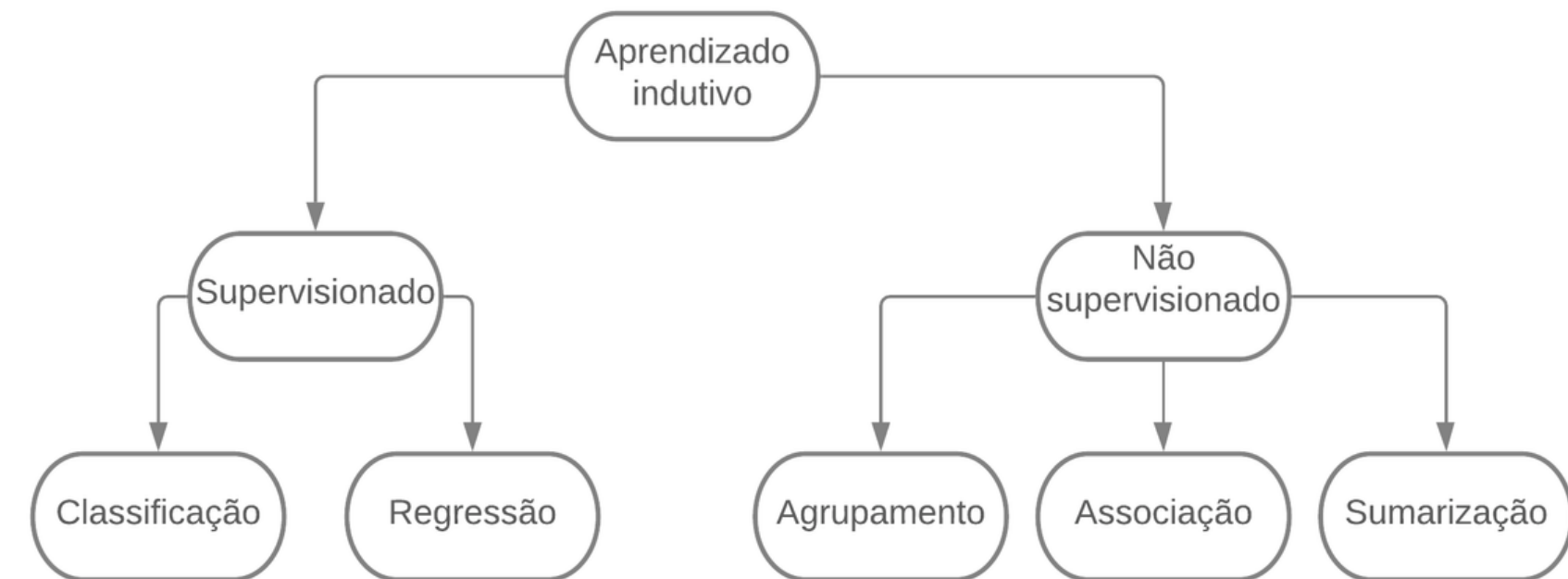
Há dois tipos de tarefas que os algoritmos de AM são capazes resolver: preditivos e descritivos.

Em problemas preditivos o objetivo principal é encontrar um modelo capaz de prever possíveis saídas para novos dados de entrada.

Para isso ser possível, em cada conjunto de dados de treinamento, as saídas precisam ser conhecidas.

Algoritmos de previsão seguem o paradigma do aprendizado supervisionado.

Figura: Hierarquia de aprendizado



Fonte: adaptado de Faceli *et al.* (2011, p.6)

# Algoritmos de Classificação

Foram escolhidos 7 algoritmos de classificação. A razão da escolha foi pelo fato de serem bem conhecidos na literatura e utilizados para resolver problemas semelhantes ao deste trabalho.

- AdaBoost
- Random Forest
- Extra Trees
- Gaussian Naive Bayes
- K-Nearest Neighbor
- Gradient Boosting
- XGBoost

# Modelos Múltiplos Preditivos (Ensemble)

---

É possível desenvolver um conjunto de classificadores que, trabalhando juntos, obtêm um melhor desempenho do que cada classificador individualmente?

---

Wolpert e Macready, com base no teorema *No Free Lunch*, fizeram a seguinte observação:

Um único algoritmo pode não realizar a previsão perfeita para um certo conjunto de dados.

Faceli *et al.* diz que algoritmos de AM possuem limitações e que produzir um algoritmo com alta acurácia é desafiador.

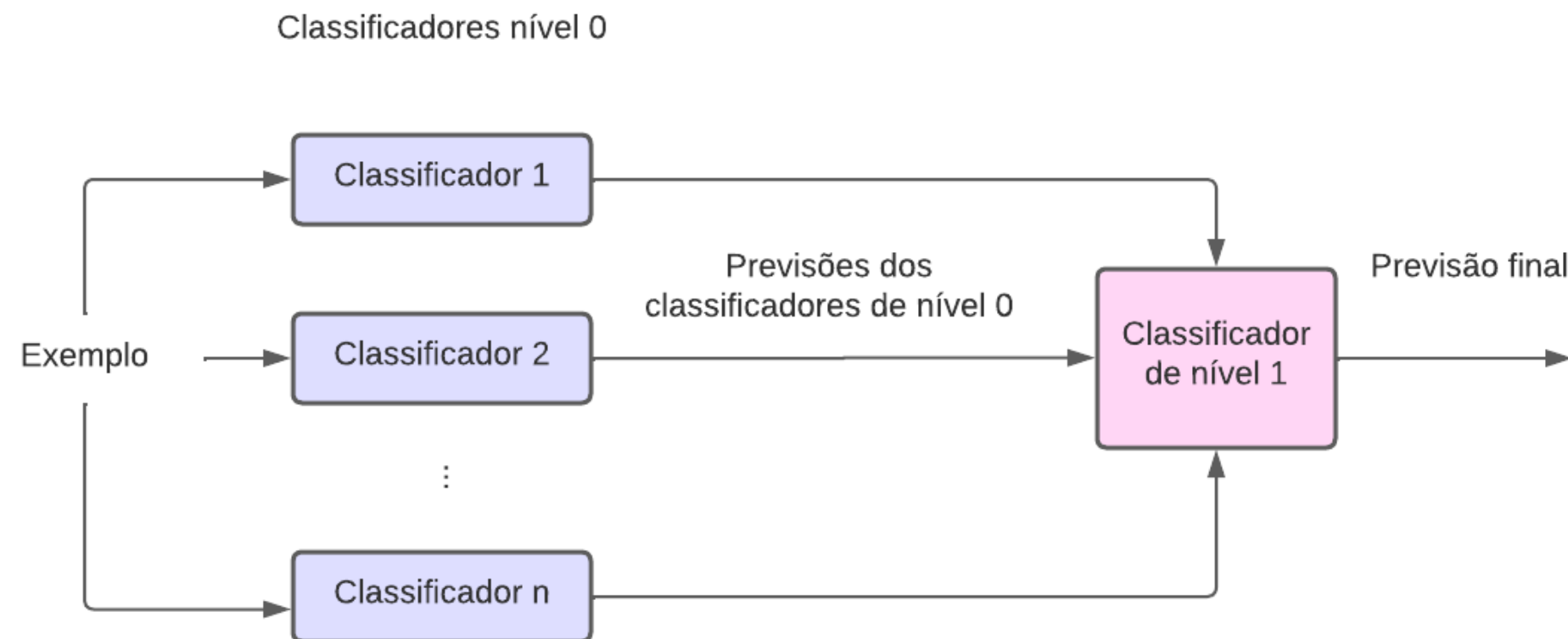
A ideia por trás da combinação de diferentes algoritmos é a de que vários algoritmos combinados produzem um resultado melhor.

# Generalização em Pilha (Stacked Generalization)

O método Generalização em Pilha foi proposto por Wolpert (1997).

Os algoritmos classificadores do nível 0 recebem como entradas dados originais e cada classificador faz uma predição.

No nível 1, o classificador recebe como entrada as predições dos classificadores anteriores, e produz uma única predição final.



Fonte: adaptado de Faceli *et al.* (2011, p.151)

# Generalização em Pilha

---

## Fase de treinamento

---

- Treinar cada classificador de nível 0 usando validação cruzada;
- Um vetor com todas as previsões dos algoritmos classificadores é criado.

## Fase de aplicação

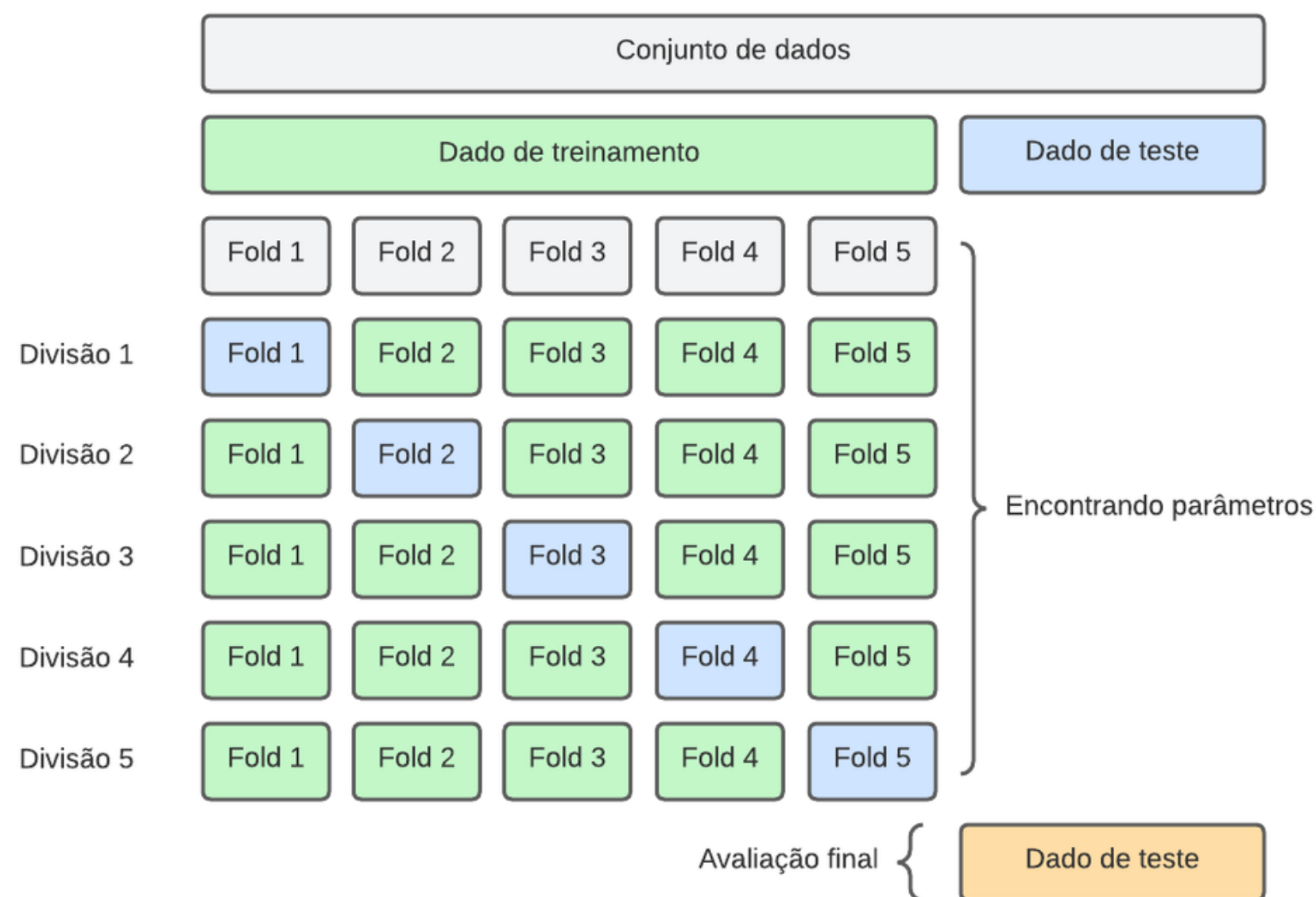
---

- O classificador do nível 1 é treinado, usando como conjunto o vetor criado pelos classificadores no nível 0;
- Quando um novo exemplo é apresentado, todos os classificadores do nível 0 produzem uma nova previsão.
- O vetor de previsão então é classificado pelo algoritmo do nível 1 que produz a previsão final.

# Validação Cruzada

A validação cruzada consiste em particionar os dados em subconjuntos (partes), onde um subconjunto é utilizado para teste e o restante para treino. A parte que é usada de teste serve para avaliação do desempenho do modelo.

Um dos métodos mais usados para validação cruzada é o k-fold, que consiste em dividir a base de dados em k subconjuntos de forma aleatória.



# Trabalhos Relacionados

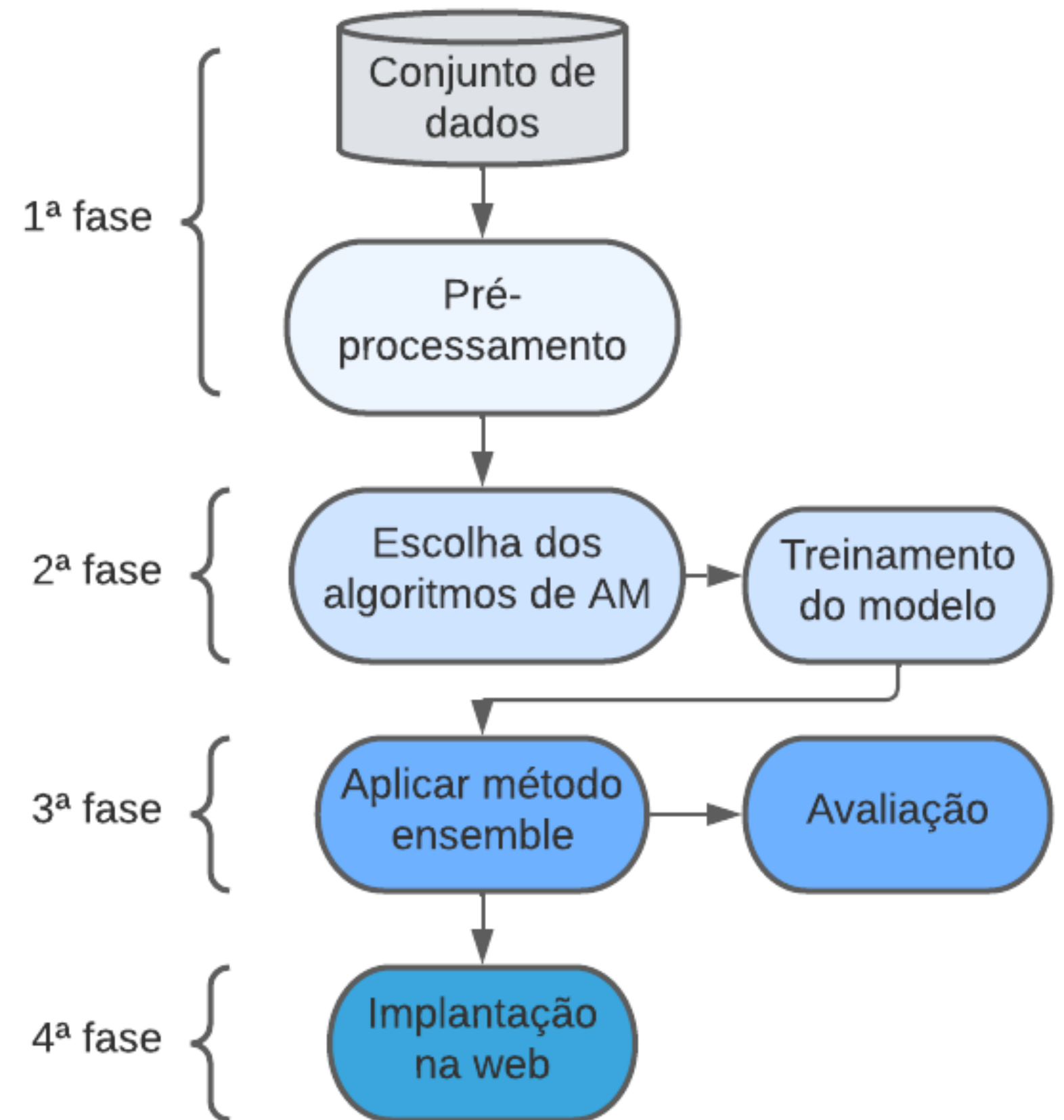
O trabalho de Emon *et al.* (2020), com o título de **Performance analysis of machine learning approaches in stroke prediction**, utilizou 10 algoritmos de AM para previsão de AVC. Os dados foram retirados de uma clínica médica de Bangladesh, totalizando 5.110 pacientes.

Os autores agregaram os resultados dos classificadores por meio de um método ensemble com abordagem de votação ponderada para alcançar maior precisão.

O modelo final obteve uma precisão de 97% mostrando que a combinação dos classificadores por meio do método ensemble de votação ponderada teve melhor desempenho que os classificadores básicos.

Os autores concluíram que o modelo obteve bom resultado para prever o AVC e pode ser usado por médicos e pacientes para prescrever e detectar precocemente um possível derrame.

# Metodologia



Fonte: a autora



# Desenvolvimento

## Pré-processamento

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
#   Column                Non-Null Count  Dtype
---  -
0   id                    5110 non-null   int64
1   gender                5110 non-null   object
2   age                  5110 non-null   float64
3   hypertension          5110 non-null   int64
4   heart_disease         5110 non-null   int64
5   ever_married          5110 non-null   object
6   work_type             5110 non-null   object
7   Residence_type        5110 non-null   object
8   avg_glucose_level     5110 non-null   float64
9   bmi                  4909 non-null   float64
10  smoking_status        5110 non-null   object
11  stroke                5110 non-null   int64
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

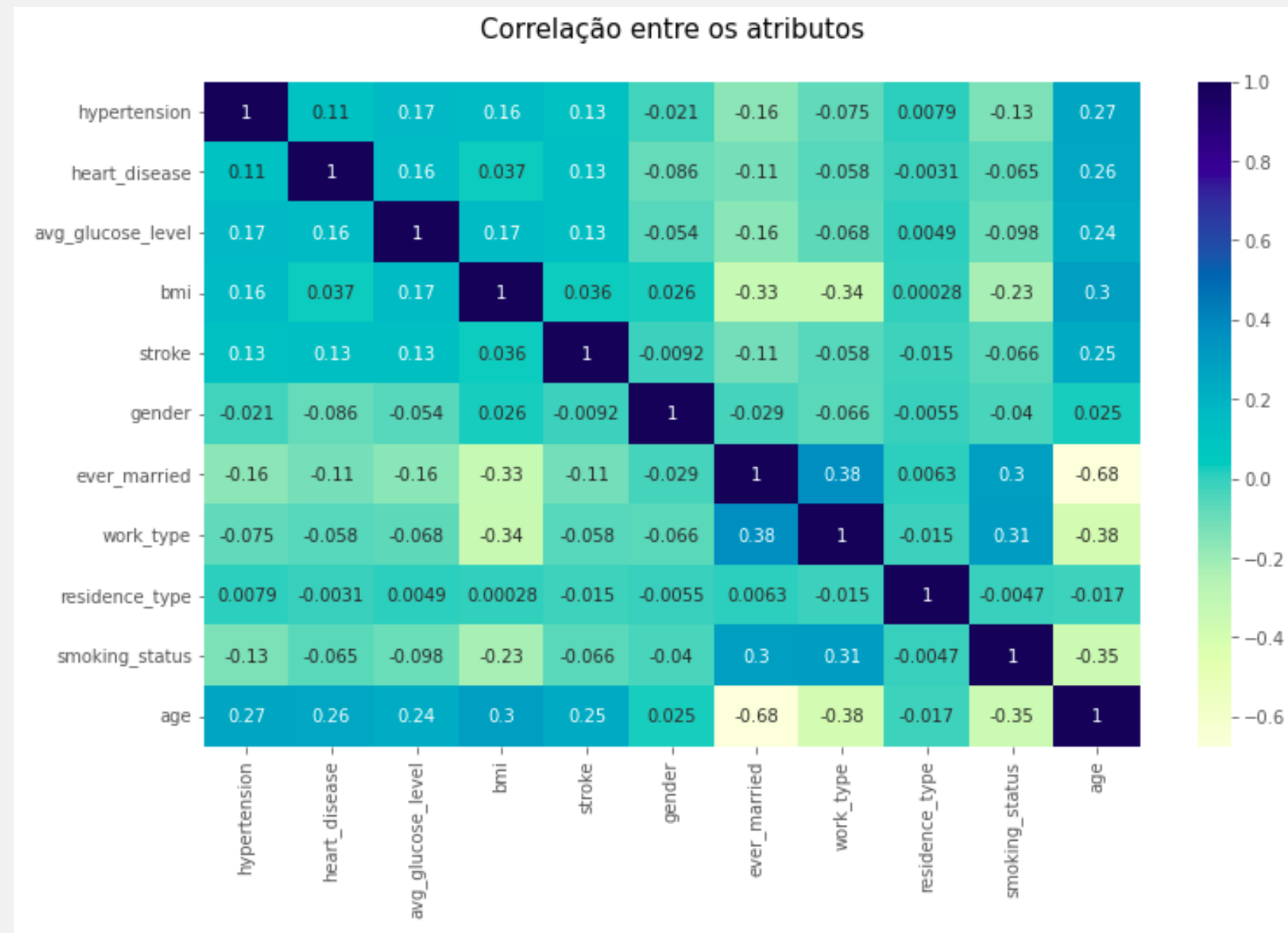
- Visão geral do conjunto de dados.
- Observou-se que há dados faltantes para bmi.
  - Eles foram preenchidos com a média dos valores conhecidos.

|   | id    | gender | age  | hypertension | heart_disease | ever_married | work_type     | Residence_type | avg_glucose_level | bmi  | smoking_status  | stroke |
|---|-------|--------|------|--------------|---------------|--------------|---------------|----------------|-------------------|------|-----------------|--------|
| 0 | 9046  | Male   | 67.0 | 0            | 1             | Yes          | Private       | Urban          | 228.69            | 36.6 | formerly smoked | 1      |
| 1 | 51676 | Female | 61.0 | 0            | 0             | Yes          | Self-employed | Rural          | 202.21            | NaN  | never smoked    | 1      |
| 2 | 31112 | Male   | 80.0 | 0            | 1             | Yes          | Private       | Rural          | 105.92            | 32.5 | never smoked    | 1      |
| 3 | 60182 | Female | 49.0 | 0            | 0             | Yes          | Private       | Urban          | 171.23            | 34.4 | smokes          | 1      |
| 4 | 1665  | Female | 79.0 | 1            | 0             | Yes          | Self-employed | Rural          | 174.12            | 24.0 | never smoked    | 1      |

Convertendo dados categóricos para numéricos.

Por exemplo, para o atributo ever\_married que possui como valores 'yes' e 'no', ao serem convertidos para números eles passam a ser 0 e 1, respectivamente.

# Correlação

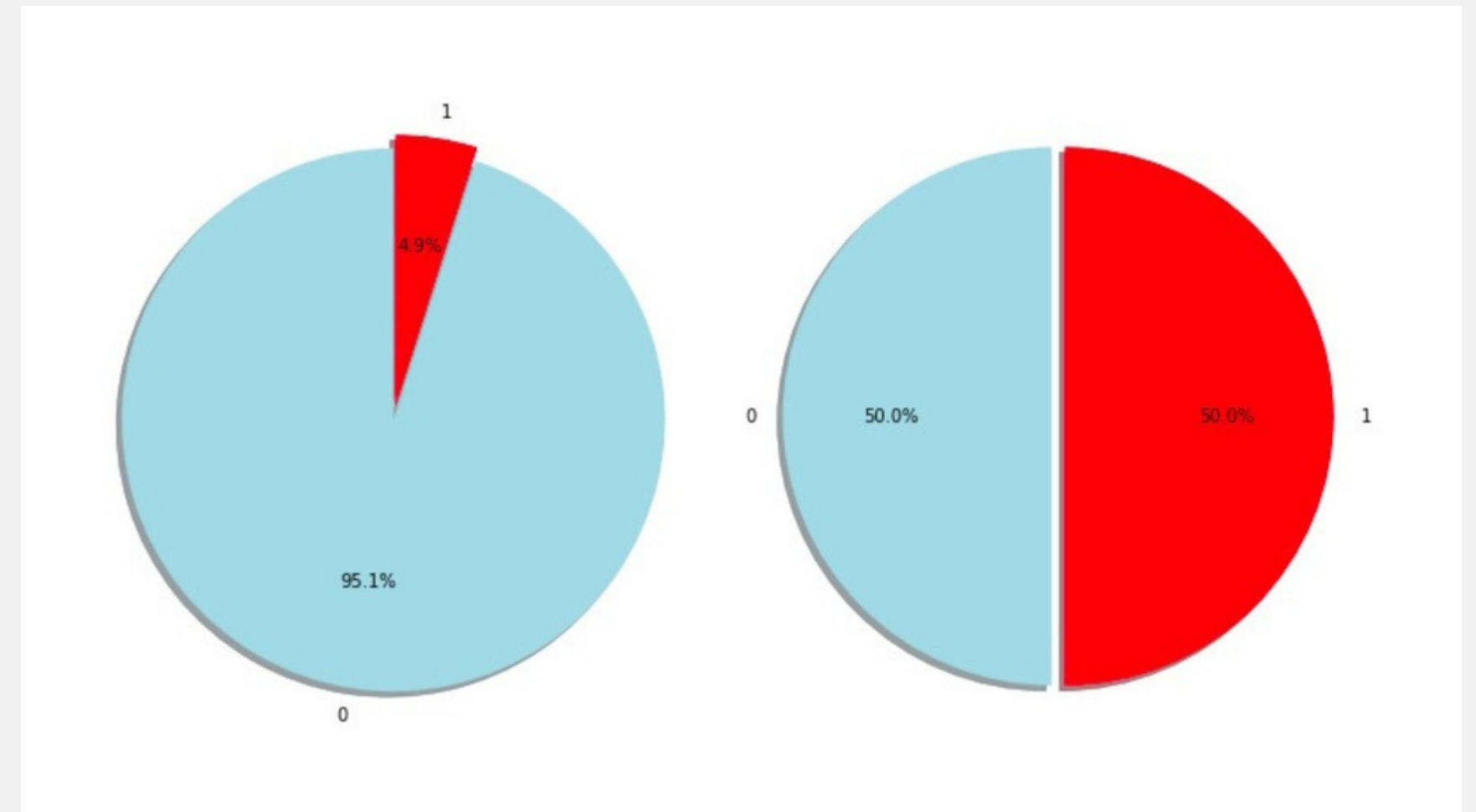


Fonte: a autora

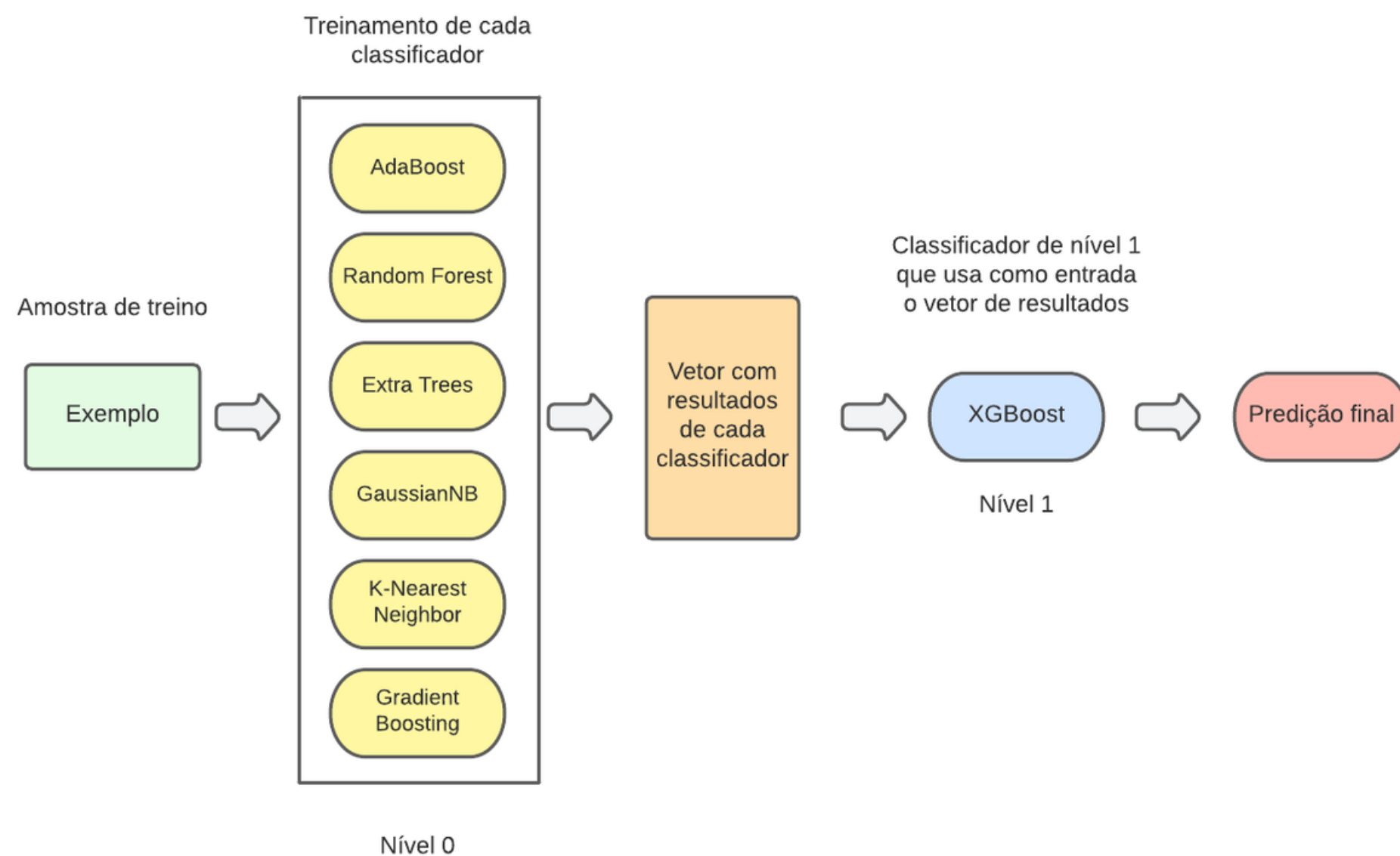
- Os valores de correlação podem variar de -1 a 1.
- Se duas variáveis tendem a diminuir ou aumentar juntas, isso indica uma correlação entre elas.
- Todos os valores conforme a figura mostram que os valores foram baixos, indicando uma relação fraca entre os atributos.

# Balanceamento

- Em todo o conjunto de dados, há 249 exemplos de casos positivos para ocorrência de AVC, e 4.861 casos negativos.
- Dados desbalanceados podem ocasionar um viés nas previsões.
- Para balancear, foi aplicado o método SMOTE nos dados de treino.
- SMOTE é uma técnica de sobreamostragem, que gera novos exemplos da classe minoritária.



# Modelagem



- A base de dados foi dividida em 70% para treino e 30% para teste.
- Seis algoritmos foram escolhidos para nível 0 e treinados com validação cruzada, k-fold com k=5.
- Os vetores de resultados das predições foram usados para treinamento do algoritmo de nível 1 (XGBoost).
- O algoritmo do nível produz uma única predição final.
- Foi aplicado a validação cruzada para avaliação da combinação das previsões.

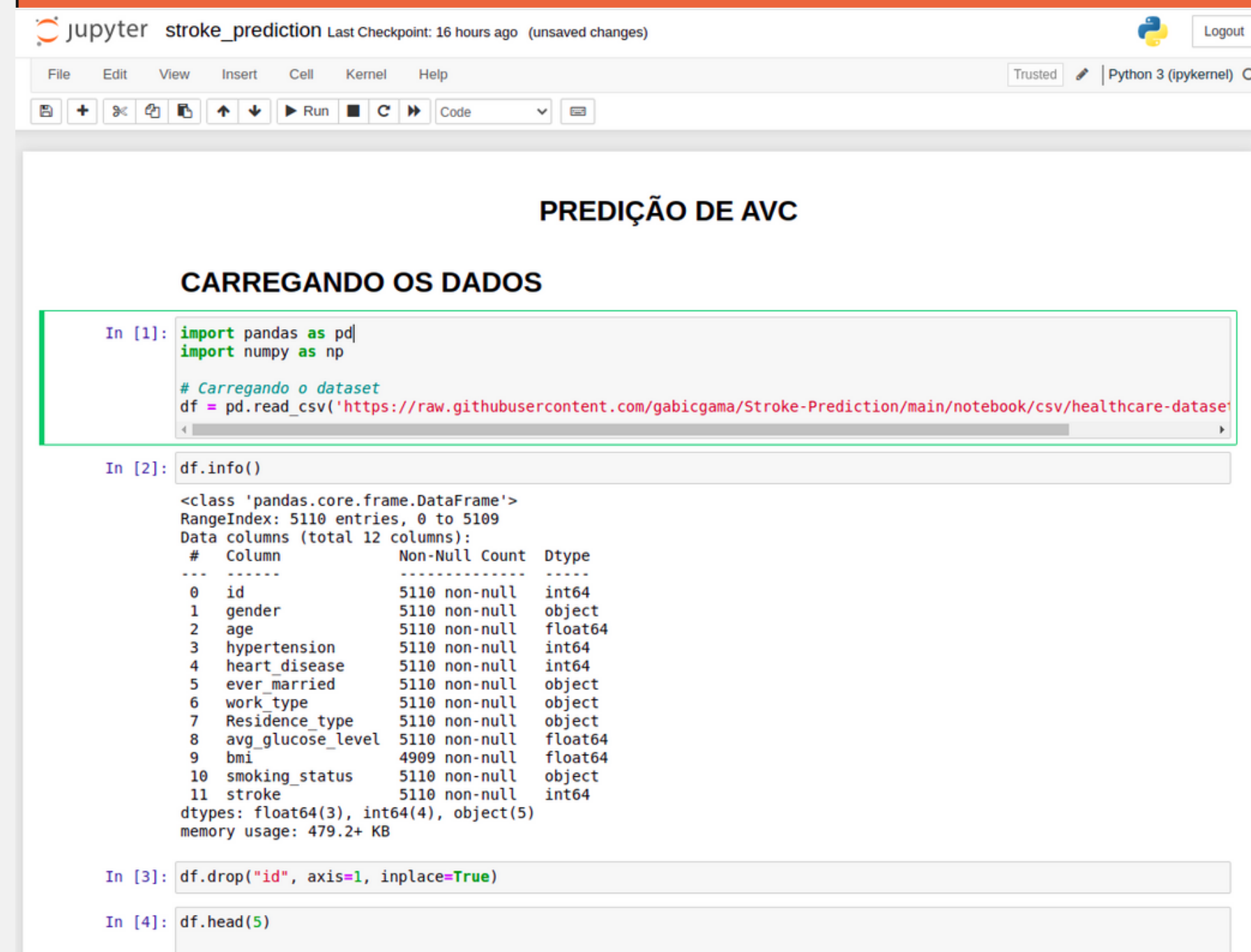
# Implantação (Jupyter Notebook)

O modelo foi desenvolvido no Jupyter Notebook, com a linguagem Python.

A principal biblioteca utilizada foi o scikit-learn, que possui todos os algoritmos implementados para fácil utilização.

Bibliotecas como Pandas e Numpy foram usadas para auxiliar na manipulação e tratamento dos dados.

Os modelos treinados foram exportados por meio do módulo pickle do Python.



The screenshot shows a Jupyter Notebook titled "stroke\_prediction" with a "Last Checkpoint: 16 hours ago (unsaved changes)" status. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with icons for file operations, running cells, and code execution. The notebook content is titled "PREDIÇÃO DE AVC" and "CARREGANDO OS DADOS". It contains four code cells:

```
In [1]: import pandas as pd
import numpy as np

# Carregando o dataset
df = pd.read_csv('https://raw.githubusercontent.com/gabigama/Stroke-Prediction/main/notebook/csv/healthcare-dataset.csv')
```

```
In [2]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  --
 0   id                    5110 non-null  int64  
 1   gender                5110 non-null  object  
 2   age                  5110 non-null  float64 
 3   hypertension          5110 non-null  int64  
 4   heart_disease         5110 non-null  int64  
 5   ever_married          5110 non-null  object  
 6   work_type              5110 non-null  object  
 7   Residence_type        5110 non-null  object  
 8   avg_glucose_level     5110 non-null  float64 
 9   bmi                   4909 non-null  float64 
10   smoking_status        5110 non-null  object  
11   stroke                5110 non-null  int64  
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

```
In [3]: df.drop("id", axis=1, inplace=True)
```

```
In [4]: df.head(5)
```

Fonte: a autora

# Implantação (Página da web)

Uma API foi desenvolvida, utilizando o Flask.

Flask é um framework web do Python que permite a fácil criação de uma aplicação web simples.

Com o Flasgger, uma extensão do Flask, foi possível criar uma interface gráfica de forma prática.

Swagger  
/apispec\_1.json Explore

## Predição de AVC <sup>0.0.1</sup>

[ Base URL: localhost:8080/apispecs ]  
/apispec\_1.json

API desenvolvida por Gabriela Campos Gama para o trabalho de conclusão do curso de Engenharia da Computação - CEFET/MG  
[Terms of service](#)  
[Contact the developer](#)

### Predição

GET /predicao\_parametros Predição utilizando os parâmetros

[Powered by [Flasgger](#) 0.9.5]

#### Predição

GET /predicao\_parametros Predição utilizando os parâmetros

Parameters

| Name   | Description |
|--|-------------|
| idade <sup>required</sup><br>number<br>(query)             | 7           |
| genero <sup>required</sup><br>number<br>(query)            | 0           |
| hipertensao <sup>required</sup><br>number<br>(query)       | 0           |
| doenca_do_coracao <sup>required</sup><br>number<br>(query) | 1           |
| ja_se_casou <sup>required</sup><br>number<br>(query)       | 0           |
| tipo_trabalho <sup>required</sup><br>number<br>(query)     | 0           |
| tipo_residencia <sup>required</sup><br>number<br>(query)   | 0           |
| nivel_glicose <sup>required</sup><br>number<br>(query)     | 228.69      |
| imc <sup>required</sup><br>number<br>(query)               | 36.6        |

Fonte: a autora



# Resultados

## Avaliação de desempenho

|   | AdaBoost | RandomForest | ExtraTrees | GaussianNB | K-NearestNeighbor | GradientBoosting | StackingModel |
|---|----------|--------------|------------|------------|-------------------|------------------|---------------|
| 0 | 0.859971 | 0.815249     | 0.790323   | 0.781525   | 0.876100          | 0.934018         | 0.943548      |
| 1 | 0.854732 | 0.853265     | 0.823184   | 0.797506   | 0.873808          | 0.937638         | 0.944974      |
| 2 | 0.840792 | 0.822450     | 0.815847   | 0.785033   | 0.873074          | 0.941306         | 0.947909      |
| 3 | 0.852531 | 0.822450     | 0.826853   | 0.777696   | 0.873074          | 0.941306         | 0.946442      |
| 4 | 0.845928 | 0.836390     | 0.820983   | 0.793837   | 0.873808          | 0.944241         | 0.939839      |

Tabela 17 – Média dos resultados da validação cruzada dos algoritmos

|        | AB        | RF        | ET        | GNB       | KNN       | GBC       |
|--------|-----------|-----------|-----------|-----------|-----------|-----------|
| média: | 0.8507908 | 0.8292215 | 0.8154434 | 0.7871194 | 0.8739728 | 0.9397018 |

Fonte: a autora

Tabela 18 – Média dos resultados da validação cruzada do modelo stacking

|        | StackingModel |
|--------|---------------|
| média: | 0.9445424     |

Fonte: a autora

# Conclusão

A aplicabilidade do método ensemble com generalização em pilha se justifica como um procedimento benéfico para obter melhores resultados, em comparação com os resultados individuais dos algoritmos.

A utilização de métodos de AM como uma ferramenta para predição de AVC mostrou-se uma opção promissora apresentando bons resultados de desempenho.

Novas predições com dados novos podem ser realizadas por meio da aplicação desenvolvida e disponibilizada na página web.





# Trabalhos Futuros

As técnicas utilizadas no trabalho podem ser aplicadas em outras bases de dados, por exemplo em um conjunto de dados coletados de um sistema de saúde brasileiro.

---

Pode-se utilizar de uma outra combinação de algoritmos para aplicar o método ensemble apresentado para resolver outros problemas de classificação ou regressão.

# Referências

- |    |   |
|----|---|
| 01 | <b>FACELI, K. et al. Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina. Rio de Janeiro: LTC, 2011.</b>   |
| 02 | <b>WOLPERT, D.; MACREADY, W. G. No free lunch theorems for optimization. IEEE Trans. Evolutionary Computation, p. 67–82, 1997.</b>  |
| 03 | <b>PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, v. 12, p. 2825–2830, 2011.</b>  |
| 04 | <b>Emon, M. U. et al. Performance analysis of machine learning approaches in stroke prediction. In: 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA). [S.l.: s.n.], 2020. p. 1464–1469.</b> |