

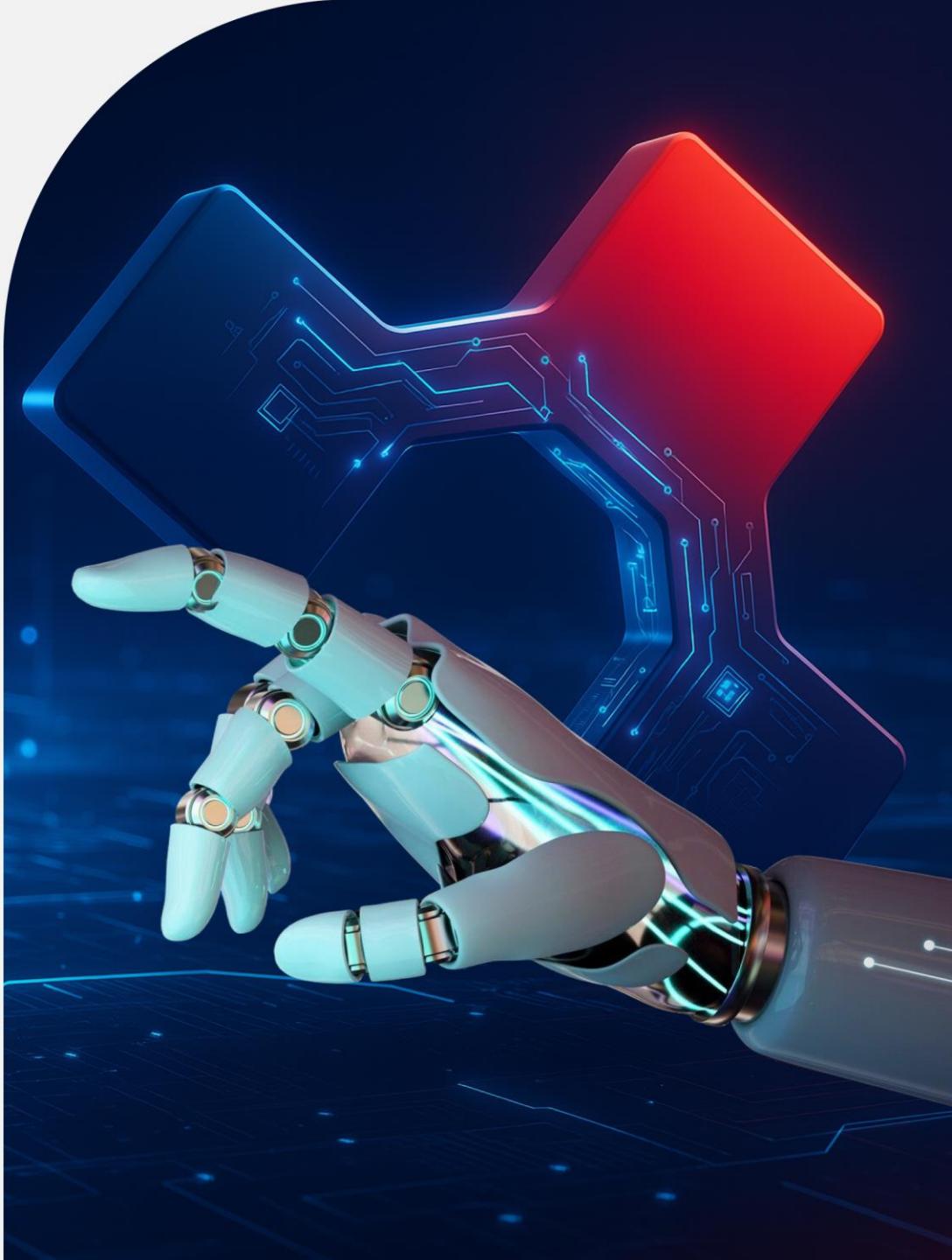
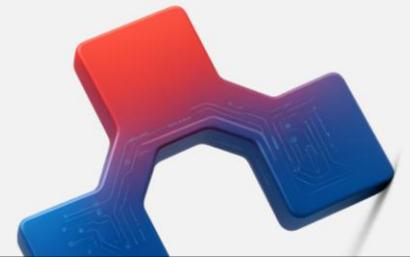


NCIA

Núcleo de Capacitação em Inteligência Artificial

# Classificação de Series Temporais

Minicurso



# Quem Sou?



## Hendrio Bragança

- Engenheiro de Machine Learning
- Cientista de Dados
- Doutorado em Computação
- Áreas de atuação: Processamento de Sinais, Visão computacional, NLP

1

**TIPOS DE DADOS**

2

**SÉRIES TEMPORAIS**

3

**SENSORES E DADOS DE SENsores**

4

**CLASSIFICAÇÃO DE SÉRIES TEMPORAIS**

5

**PROCESSAMENTO DE SINAIS**

6

**PIPELINE DE MACHINE LEARNING PARA SÉRIES TEMPORAIS**

7

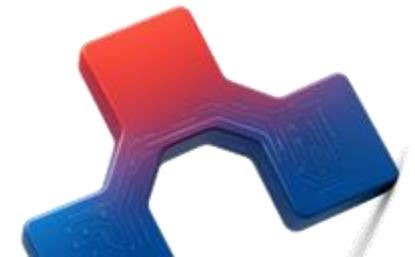
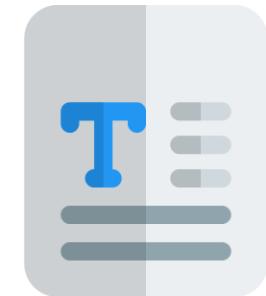
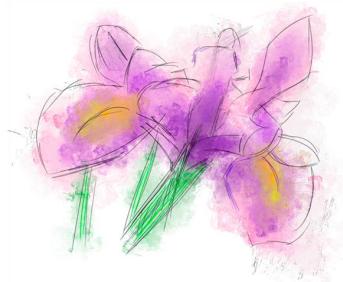
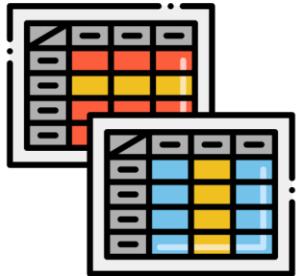
**DISCUSSÃO**

8

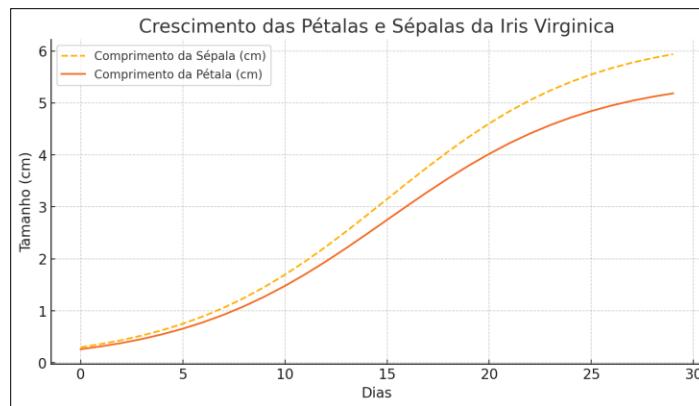
**AULA PRÁTICA**



# Tipos de Dados em Ciência de Dados e Machine Learning



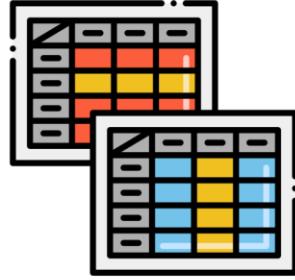
# Tipos de Dados



A **Iris virginica** é a maior das três espécies (setosa, versicolor), destacando-se por suas pétalas longas e largas, com um comprimento médio de **5.5 cm** e largura de **2.0 cm**.

- Suas sépalas também são extensas, medindo cerca de **6.3 cm** de comprimento e **3.0 cm** de largura.
- Apresenta uma coloração predominantemente **roxa escura**, podendo exibir variações em tons de azul e violeta.
- É comumente encontrada em solos úmidos, como prados alagados e áreas de pântano.

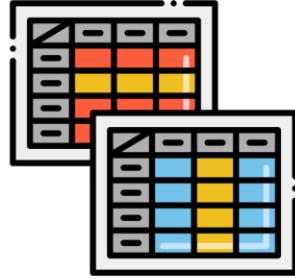
Espécie	Comprimento da Sépala (cm)	Largura da Sépala (cm)	Comprimento da Pétala (cm)	Largura da Pétala (cm)
Iris Setosa	5.0	3.4	1.5	0.2
Iris Versicolor	6.0	2.8	4.5	1.3
Iris Virginica	6.3	3.0	5.5	2.0



# Dados Tabulares

- **Definição:** Dados organizados em linhas e colunas
  - (ex: planilhas, bancos relacionais).
- **Características:**
  - Estrutura clara e padronizada.
  - Fácil integração com bibliotecas (pandas, SQL).
- **Prós:**
  - Alto suporte em ferramentas de ML.
  - Facilidade de limpeza e manipulação.
- **Contras:**
  - Pode não capturar bem contextos complexos (ex: imagem ou linguagem natural).

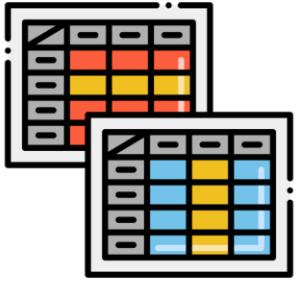




# Dados Tabulares

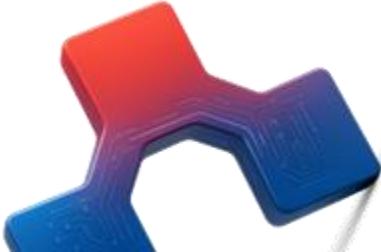
- **Definição:** Dados organizados em linhas e colunas
  - (ex: planilhas, bancos relacionais).
- **Características:**
  - Estrutura clara e padronizada.
  - Fácil integração com bibliotecas (pandas, SQL).
- **Prós:**
  - Alto suporte em ferramentas de ML.
  - Facilidade de limpeza e manipulação.
- **Contras:**
  - Pode não capturar bem contextos complexos (ex: imagem ou linguagem natural).





# Dados Tabulares

Espécie	Comprimento da Sépala (cm)	Largura da Sépala (cm)	Comprimento da Pétala (cm)	Largura da Pétala (cm)
Iris Setosa	5.0	3.4	1.5	0.2
Iris Versicolor	6.0	2.8	4.5	1.3
Iris Virginica	6.3	3.0	5.5	2.0



# Dados de Imagem

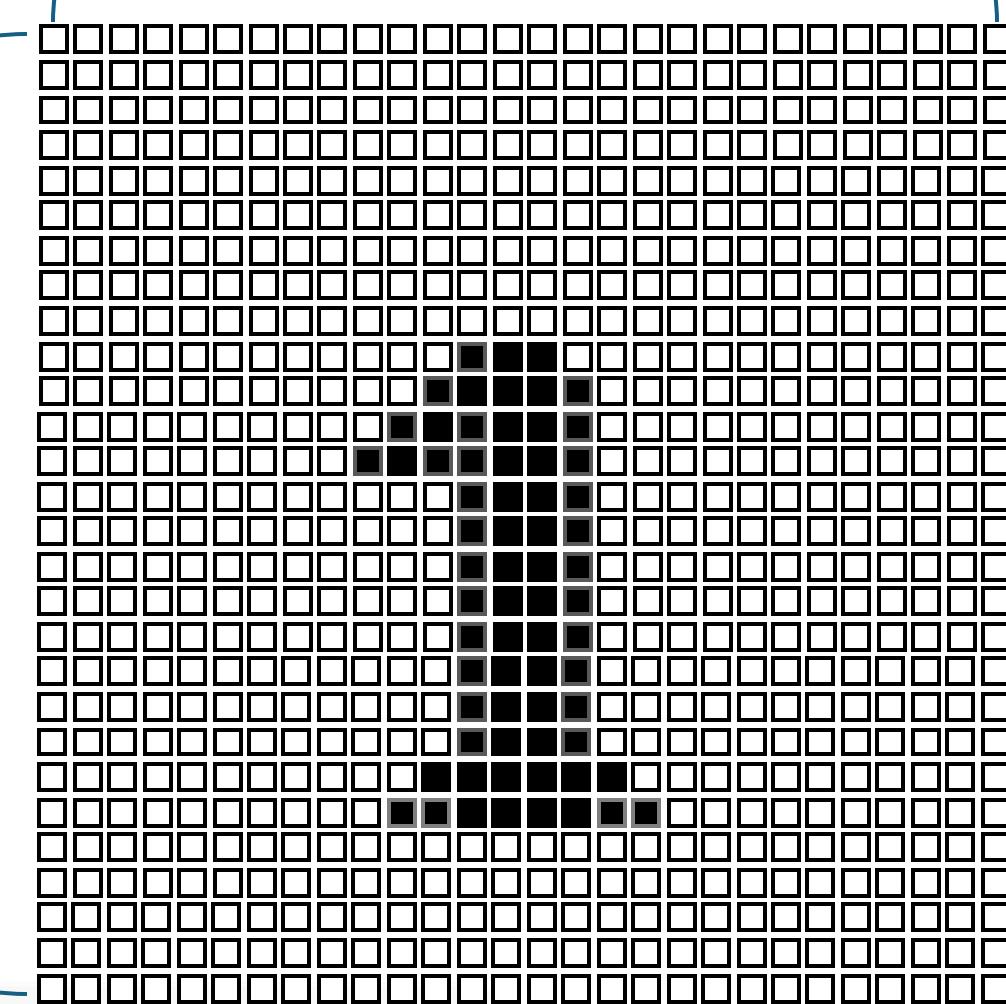
- **Definição:** Matrizes de pixels (RGB, escala de cinza).
- **Características:**
  - Alta dimensionalidade (milhares/milhões de pixels).
  - Informação espacial importante.
- **Prós:**
  - Permite detectar padrões visuais complexos.
  - Aplicações vastas (saúde, visão computacional, satélites).
- **Contras:**
  - Requer alto poder computacional.
  - Necessidade de anotações rotuladas para treino.
- **Exemplos:**
  - MNIST (dígitos manuscritos),
  - APTOS 2019 (retinopatia),
  - COCO dataset.



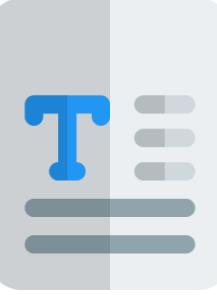
# Dados de Imagem

28

28



A imagem ao lado possui  $28 \times 28 = 784$  pixels, cada um com um valor de brilho entre 0 (preto) e 1 (branco).

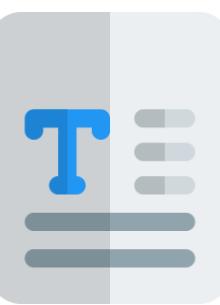


# Dados Textuais

- **Definição:** Dados não estruturados em linguagem natural.
- **Características:**
  - Ambiguidade semântica.
  - Representações comuns: Bag-of-Words, TF-IDF, embeddings.
- **Prós:**
  - Riqueza de contexto e informação.
  - Modelos de NLP (LLMs) conseguem entender semântica.
- **Contras:**
  - Pré-processamento complexo (limpeza, tokenização).
  - Dependência de idioma/contexto.
- **Exemplos:**
  - Tweets, reviews, artigos científicos.

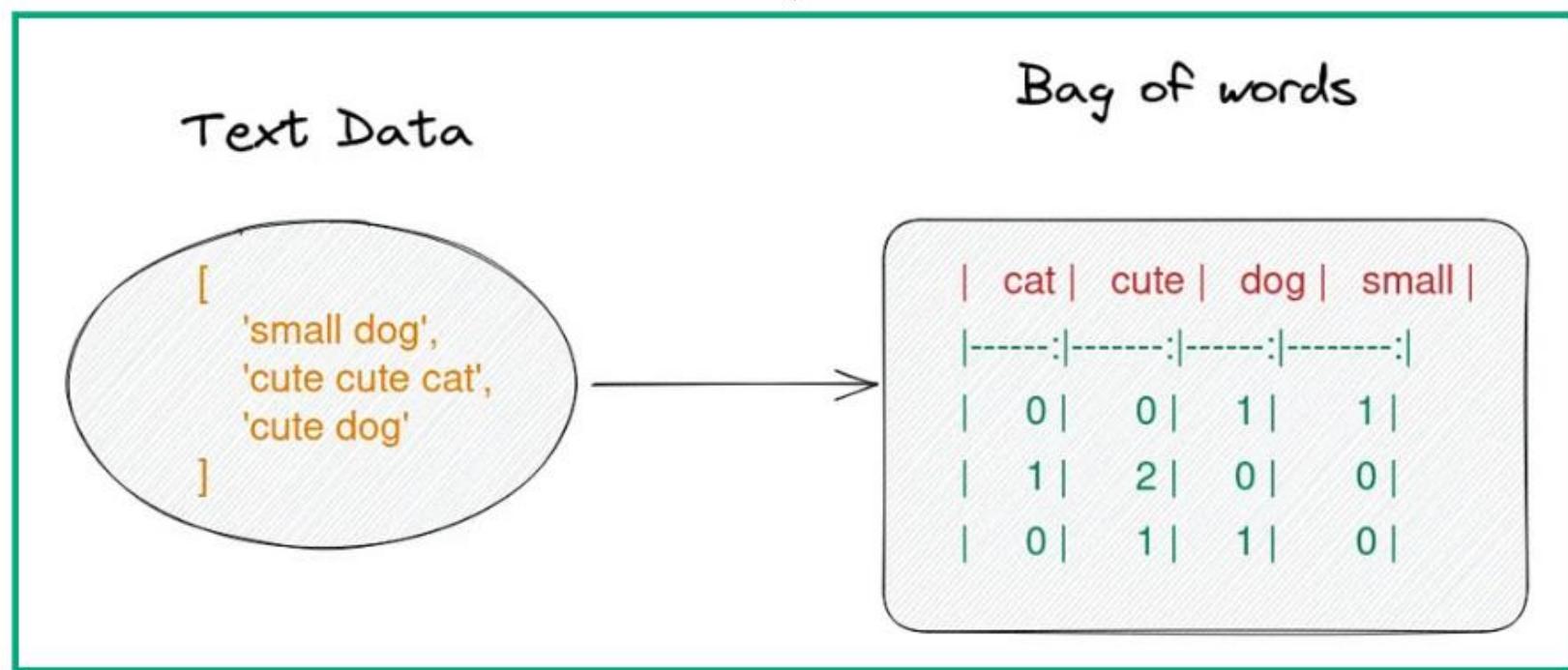
A **Iris virginica** é a maior das três espécies (setosa, versicolor), destacando-se por suas pétalas longas e largas, com um comprimento médio de **5.5 cm** e largura de **2.0 cm**.

- Suas sépalas também são extensas, medindo cerca de **6.3 cm** de comprimento e **3.0 cm** de largura.
- Apresenta uma coloração predominantemente **roxa escura**, podendo exibir variações em tons de azul e violeta.
- É comumente encontrada em solos úmidos, como prados alagados e áreas de pântano.



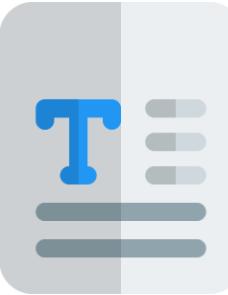
# Dados Textuais

- **Transformação de tipos de dados**
  - Textual para Tabular



**Bag of Words (BoW):** Representa um texto como uma matriz onde cada coluna é uma palavra única e os valores são suas frequências no documento.

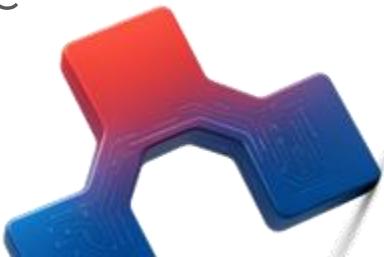




# Dados Textuais

## Principais Métodos

- 1. Bag of Words (BoW)**: Representa um texto como uma matriz onde cada coluna é uma palavra única e os valores são suas frequências no documento.
- 2. TF-IDF (Term Frequency - Inverse Document Frequency)**: Calcula a importância de uma palavra em relação a um conjunto de documentos.
- 3. Word Embeddings (Word2Vec, GloVe, FastText)**: Modelos mais avançados que capturam o significado semântico das palavras transformando-as em vetores densos.
- 4. Transformers (BERT, GPT, T5)**: Modelos baseados em deep learning que compreendem melhor o contexto.





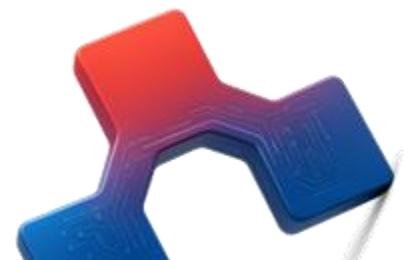
# Dados Temporais

- **Definição:** Dados coletados em sequência ao longo do tempo.
- **Características:**
  - Ordem importa.
  - Sazonalidade, tendência, ruído.
- **Prós:**
  - Essenciais para previsão (\*financeira, climática, IoT).
  - Forte aplicabilidade prática.
- **Contras:**
  - Difícil lidar com outliers e dados faltantes.
  - Dependência de granularidade (segundos, dias, anos).
- **Exemplos:**
  - Preço de ações, sensores de IoT, dados meteorológicos.





# Dados Multimodais

- **Definição:** Combinação de múltiplos tipos de dados
    - exemplo: imagem + texto.
  - **Características:**
    - Exigem integração entre representações diferentes.
  - **Prós:**
    - Modelos mais ricos e completos.
    - Base para sistemas avançados (ex: ChatGPT com visão).
  - **Contras:**
    - Complexidade alta na modelagem.
  - **Exemplos:**
    - Radiologia (imagem + laudo), vídeos legendados, sistemas de e-commerce (foto + descrição + reviews).
- 

# Fontes de Dados

**Welcome to the UC Irvine Machine Learning Repository**

We currently maintain 674 datasets as a service to the machine learning community. Here, you can donate and find datasets used by millions of people all around the world!

[VIEW DATASETS](#) [CONTRIBUTE A DATASET](#)

### Popular Datasets

- Iris**  
A small classic dataset from Fisher, 1936. One of the earliest known datasets used for ev...  
 Classification 150 Instances 4 Features
- Heart Disease**  
4 databases: Cleveland, Hungary, Switzerland, and the VA Long Beach  
 Classification 303 Instances 13 Features
- Wine Quality**  
Two datasets are included, related to red and white vinho verde wine samples, from the ...  
 Classification, Regressi... 4.9K Instances 12 Features

### New Datasets

- Lattice-physics (PWR fuel assembly neutronics simulation results)**  
This dataset encompasses lattice-physics parameters—the infinite multiplication factor (...  
 Regression 24K Instances 39 Features
- Gas sensor array low-concentration**  
This dataset contains 6 gas responses collected by a sensor array consisting of 10 metal ...  
 Classification, Regressi... 90 Instances
- Twitter Geospatial Data**  
Seven days of geo-tagged Tweet data from the United States with exact GPS location an...  
 Classification, Regressi... 14.26M Instances 4 Features

<https://archive.ics.uci.edu/datasets>

# Fontes de Dados

The screenshot shows the Hugging Face Hub documentation page for datasets. The left sidebar has a dark theme with a navigation bar for 'Hub' (selected), 'Datasets' (highlighted in red), and other sections like 'Models', 'Spaces', and 'Community'. The main content area has a light background. It features a section titled 'Datasets' with a brief introduction about the Hub's dataset collection and how to interact with them. Below this is a 'Contents' section with a list of links to various dataset-related topics. At the bottom of the page is a GitHub update link.

**Datasets**

The Hugging Face Hub is home to a growing collection of datasets that span a variety of domains and tasks. These docs will guide you through interacting with the datasets on the Hub, uploading new datasets, exploring the datasets contents, and using datasets in your projects.

This documentation focuses on the datasets functionality in the Hugging Face Hub and how to use the datasets with supported libraries. For detailed information about the 😊 Datasets python package, visit the 😊 [Datasets documentation](#).

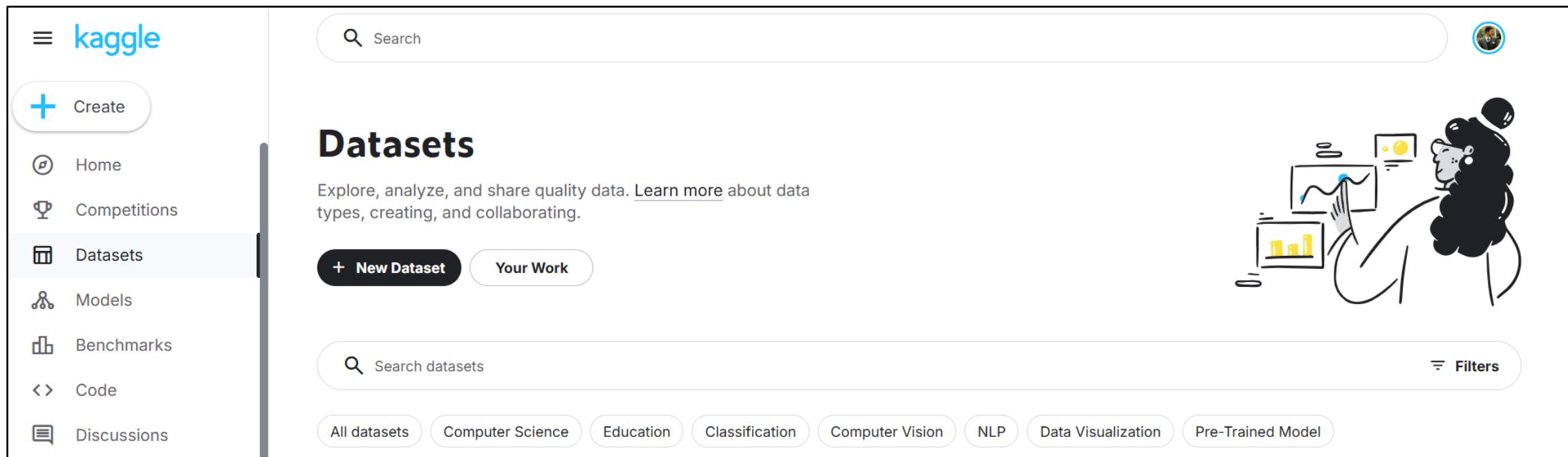
**Contents**

- [Datasets Overview](#)
- [Dataset Cards](#)
- [Gated Datasets](#)
- [Uploading Datasets](#)
- [Uploading Datasets \(for LLMs\)](#)
- [Downloading Datasets](#)
- [Integrated Libraries](#)
- [Data Studio](#)
- [Datasets Download Stats](#)
- [Data files Configuration](#)
- [Spaces](#)
- [Spaces Overview](#)
- [Spaces GPU Upgrades](#)
- [Spaces ZeroGPU](#)

⇠ [Update on GitHub](#)

<https://huggingface.co/docs/hub/en/datasets>

# Fontes de Dados



The screenshot shows the Kaggle website's datasets page. On the left, there's a sidebar with navigation links: Create, Home, Competitions, Datasets (which is selected and highlighted in light blue), Models, Benchmarks, Code, and Discussions. A search bar at the top right contains the placeholder "Search". Below the sidebar, the main content area has a title "Datasets" and a sub-instruction "Explore, analyze, and share quality data. [Learn more](#) about data types, creating, and collaborating." It features a "New Dataset" button and a "Your Work" button. Further down are a search bar labeled "Search datasets", a "Filters" button, and several category filters: All datasets, Computer Science, Education, Classification, Computer Vision, NLP, Data Visualization, and Pre-Trained Model. To the right of the filters is a cartoon illustration of a person with curly hair working on a laptop.

<https://www.kaggle.com/datasets>

# Fontes de Dados

The screenshot shows the figshare homepage. At the top left is the figshare logo. To its right are navigation links for "Browse", a search bar with placeholder text "Search on figshare...", and user account links "Log in" and "Sign up". The main background image is a vibrant, abstract 3D rendering of molecular structures in red, orange, and purple. A central white rectangular overlay contains the tagline "store, share, discover research" in a bold, sans-serif font. Below this, smaller text reads "get more citations for all of the outputs of your academic research over 80,000 citations of figshare content to date". Further down, it says "ALSO FOR INSTITUTIONS & PUBLISHERS". At the bottom of this overlay is a quote from WIRED: "'figshare wants to open scientific data to the world'". The footer of the page includes links for "About", "Solutions", "Tools", "Blog", "Contact", "Help", "Privacy Policy", "Cookie Settings", "Terms", and "Sitemap". Social media icons for Facebook, Twitter, and YouTube are also present. The footer also features the text "figshare. credit for all your research.", the "Port of DIGITALscience" logo, and logos for DataCite, COPAC, and OPEN ACCESS.

<https://figshare.com/>

# Introdução a Séries Temporais





# Séries Temporais

- **Definição:** Dados organizados como observações ao longo do tempo (diário, mensal, anual, etc.).
  - **Exemplos práticos:**
    - Mercado financeiro (ações, câmbio)
    - Sensores IoT (temperatura, pressão)
    - Dados de saúde (batimentos cardíacos, glicemias)
    - Demanda de produtos e energia elétrica
  - **Características-chave:**
    - Ordem importa.
    - Correlação entre observações vizinhas.
    - Dependência temporal (passado influencia o futuro).
- 

# Componentes de uma Série Temporal

- **Tendência (Trend):** Movimento de longo prazo.
- **Sazonalidade (Seasonality):** Padrões cíclicos regulares (dia, semana, ano).
- **Ciclos (Cycles):** Oscilações irregulares de médio/longo prazo (ex: crises econômicas).
- **Ruído (Noise):** Variações aleatórias/imprevisíveis.

# Problemas Clássicos em Séries Temporais

- **Não estacionariedade:**

- Média e variância mudam ao longo do tempo.
- Solução: diferenciação, transformações log/Box-Cox.

- **Valores ausentes:**

- Comuns em sensores e registros.
- Técnicas: interpolação, forward/backward fill.

- **Outliers:**

- Picos anômalos (ex: falha em sensor).
- Técnicas: detecção robusta (IQR, z-score, Isolation Forest).

- **Granularidade inadequada:**

- Dados coletados em frequências diferentes precisam ser reamostrados.

# Técnicas Tradicionais de Modelagem

- **Modelos estatísticos clássicos:**
  - **AR (Autoregressive)** – prevê com base em valores passados.
  - **MA (Moving Average)** – usa erros passados.
  - **ARMA / ARIMA** – combinação para séries estacionárias.
  - **SARIMA** – ARIMA com sazonalidade.
  - **ETS (Exponential Smoothing)** – suavização exponencial.
- **Limitações:** funcionam bem em séries univariadas, mas têm dificuldade com séries de alta complexidade ou dados multivariados.



# Técnicas Modernas de Modelagem

- **Machine Learning:**
  - Regressão, Random Forest, XGBoost aplicados em features extraídas.
- **Deep Learning:**
  - **RNN / LSTM / GRU:** capturam dependências de longo prazo.
  - **1D CNNs:** extração de padrões locais temporais.
  - **Transformers (ex: Temporal Fusion Transformer, TimeGPT):** estado da arte em previsão de séries temporais.
- **Modelos Probabilísticos:** Prophet (Facebook), GluonTS.

# Feature Engineering em Séries Temporais

- **Lag Features:** criar variáveis com valores defasados ( $t-1$ ,  $t-7$ , etc.).
- **Janela Móvel (Rolling Window):** médias móveis, somas acumuladas, desvio padrão.
- **Extração de Sazonalidade:** dia da semana, mês, feriados.
- **Transformações de Frequência:**
  - Fourier Transform, Wavelets → capturam padrões cílicos.
- **Agregações Multiescala:**
  - Reamostrar em diferentes granularidades (diário, semanal, mensal).



# Avaliação de Modelos

- Métricas comuns:
  - MAE (Mean Absolute Error)
  - RMSE (Root Mean Squared Error)
  - MAPE (Mean Absolute Percentage Error)
- **Métricas de classificação:**
  - Acurácia
  - AUC
  - F1-Score

# Avaliação de Modelos no Contexto de Séries Temporais

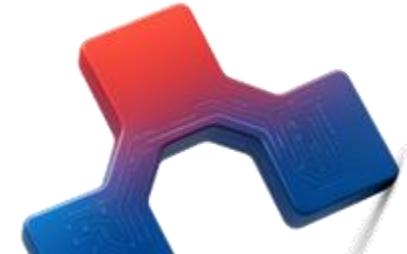
Method	Temporal Order	Computational Cost	Data Efficiency	Suitable for Time Series	Risk of Data Leakage
K-fold	✗	Medium	High	✗	High
LOOCV	✗	Very High	Very High	✗	High
Bootstrap	✗	High	High	✗	High
Nest Cross Validation	✗	Very High	High	✗	High
Hold-Out	✓	Low	Low	Partially	Low
Cross-Validation	✓	High	Medium	✓	Low
Walk-Forward Validation	✓	Medium	Medium	✓	Low



# Conclusão

- Séries temporais têm particularidades próprias e exigem técnicas específicas.
  - Podemos usar modelos estatísticos clássicos até DL de última geração.
- **Feature engineering e avaliação temporal** são fundamentais.
- Cada aplicação define a melhor abordagem.

# Sensores e Dados de Sensores





# Dados de Sensores

- Os **dados de sensores** são informações coletadas por **dispositivos eletrônicos** que monitoram variáveis físicas ou químicas.
- Se um sensor só registra **eventos específicos** (exemplo: um alerta quando uma temperatura crítica é atingida), ele **não gera uma série temporal clássica**

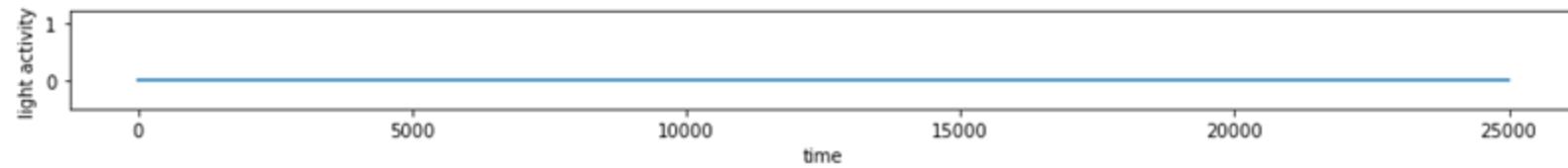
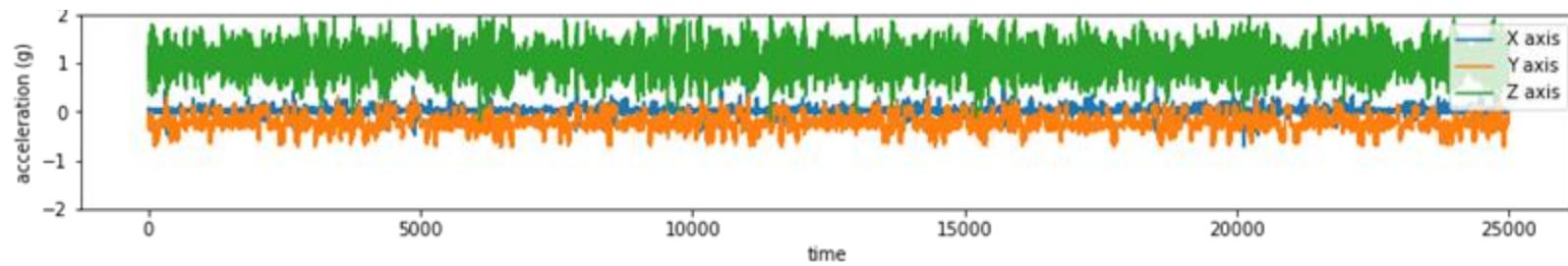
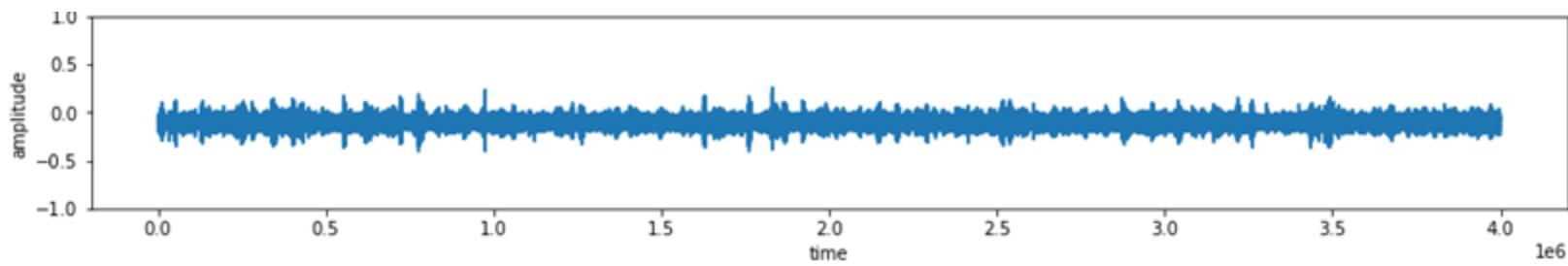


# Dados de Sensores

- Podem ser **contínuos** ou **discretos**
  - Exemplo: um sensor de temperatura pode registrar a cada segundo (contínuo) ou só quando há variação significativa (discreto).
- **Multidimensionais**
  - Um sistema pode coletar dados de vários sensores ao mesmo tempo.
- **Geralmente precisam de pré-processamento**
  - Como remoção de ruído, calibração e normalização.
- Usados para **controle** de sistemas e **monitoramento** em tempo real.

# Dados de Sensores

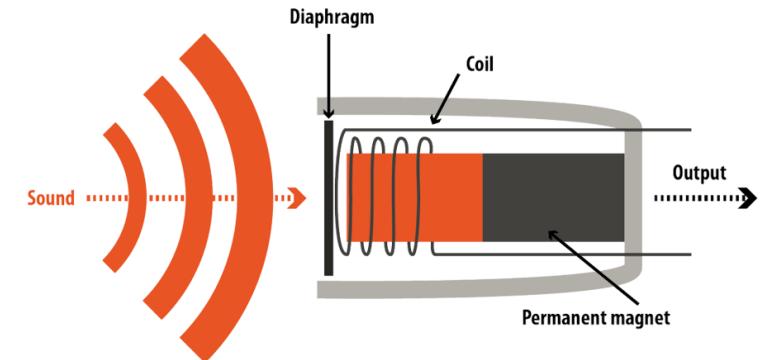
- Áudio, Acelerômetro e Infravermelho



# Dados de Sensores

- **Microfone**

- As ondas sonoras atingem o **diafragma** do microfone
- O diafragma se move, **movimentando** também **a bobina** que está presa a ele
- A bobina se move em torno de um ímã, variando o fluxo magnético
- **A variação do fluxo magnético gera uma corrente elétrica**
- A corrente elétrica é enviada para um amplificador, gravador ou outro equipamento

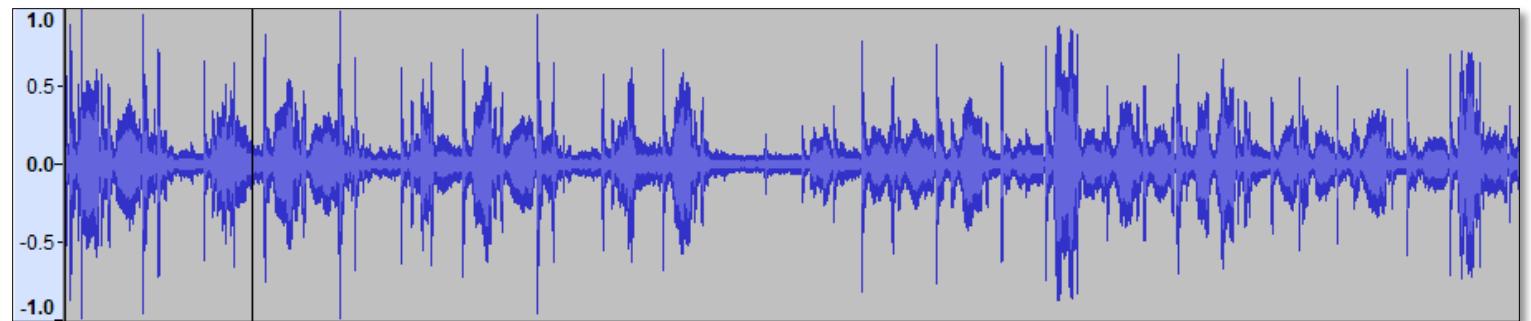


# Dados de Sensores

- O microfone pode ser usado em analise de padrões de audio



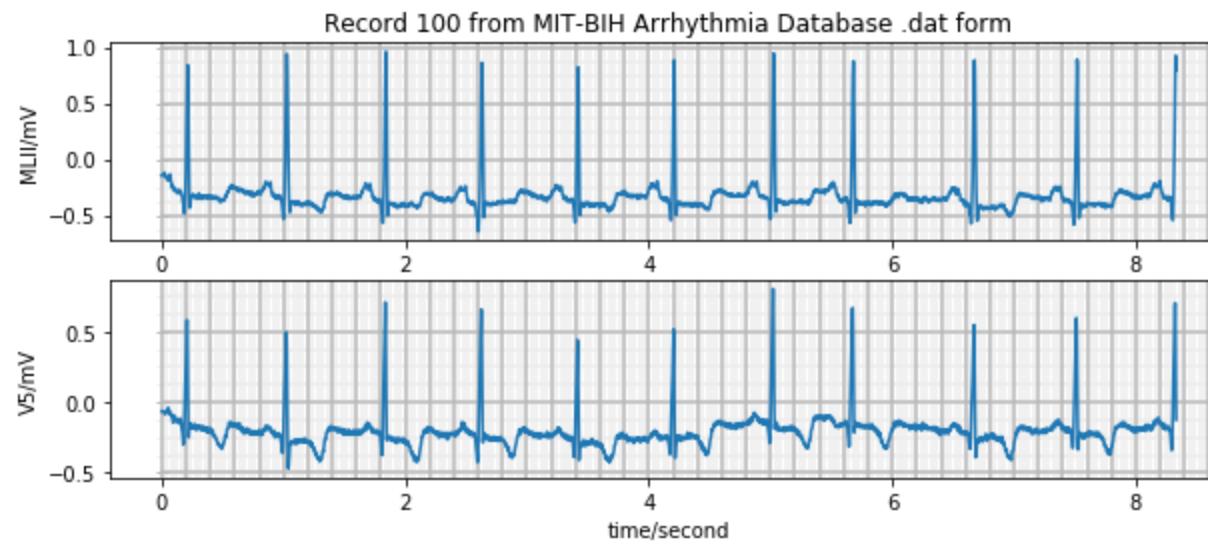
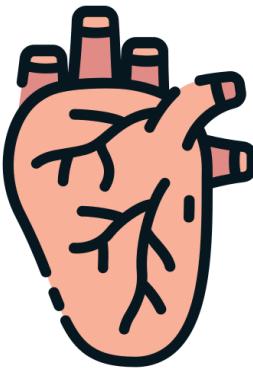
Uirapuru-ferrugíneo



Amostragem de 48khz

# Dados de Sensores

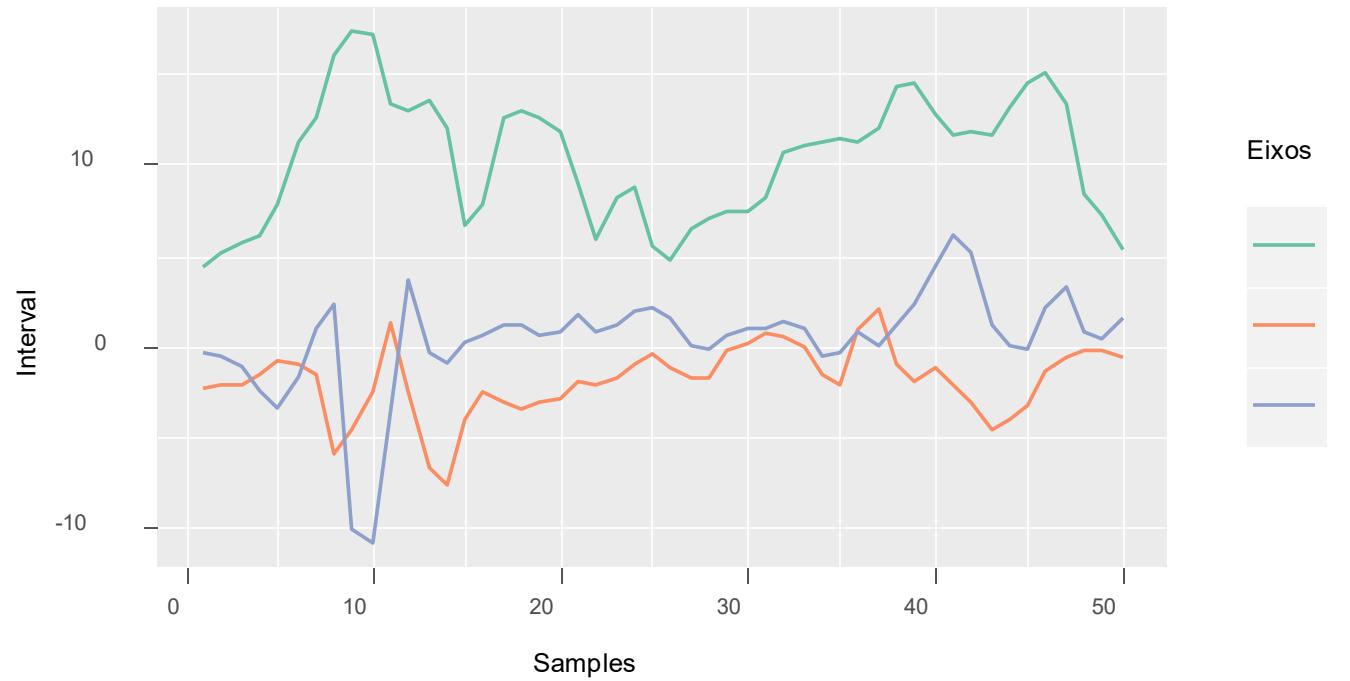
- **Electrocardiogram (ECG)**





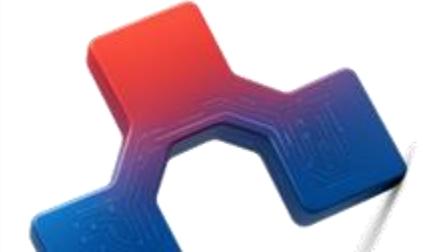
# Dados de Sensores

- Acelerômetro e Giroscópio



Eixos

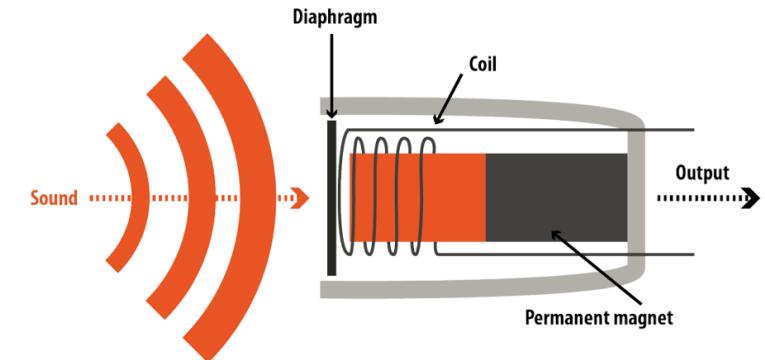
X  
Y  
Z



# Dados de Sensores

- **Microfone**

- As ondas sonoras atingem o **diafragma** do microfone
- O diafragma se move, **movimentando** também **a bobina** que está presa a ele
- A bobina se move em torno de um ímã, variando o fluxo magnético
- **A variação do fluxo magnético gera uma corrente elétrica**
- A corrente elétrica é enviada para um amplificador, gravador ou outro equipamento





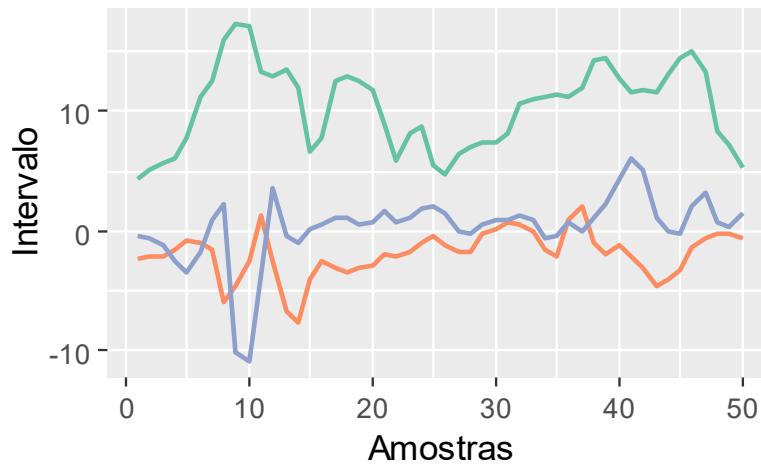
# Dados de Sensores

- A **frequência de amostragem** define **quantas amostras por segundo** são capturadas de um sinal analógico para convertê-lo em um sinal digital.
  - A escolha da frequência de amostragem depende da aplicação.
  - Se for **muito baixa**, pode perder informações;
  - Se for **muito alta**, pode aumentar desnecessariamente o tamanho dos dados e a necessidade de processamento.
- O teorema de Nyquist afirma que a frequência mais alta que pode ser representada com precisão em uma gravação digital é metade da taxa de amostragem.

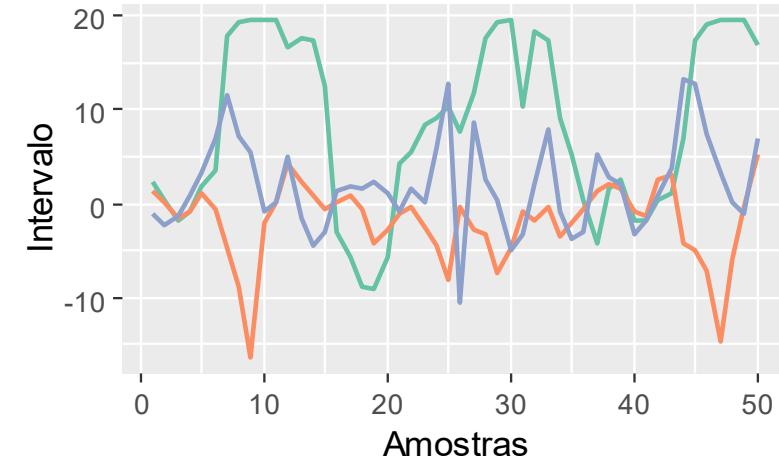
# Dados de Sensores

- Como extrair informações?

Classe 1: Andar



Classe 2: Correr



Eixos

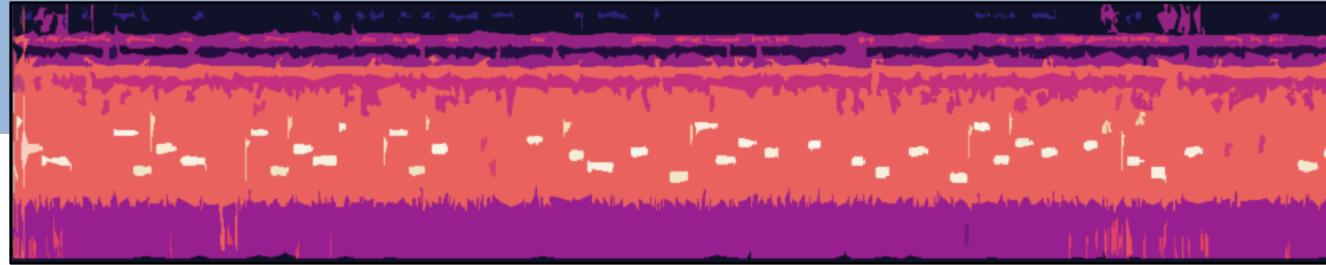
X
Y
Z



# Classificação de Séries Temporais

			<ul style="list-style-type: none"><li>• Média</li><li>• Máximo</li><li>• Mínimo</li><li>• Desvio Padrão</li></ul>			<ul style="list-style-type: none"><li>• Acertos</li><li>• Erros</li></ul>
Fonte de Dados	Pre-Processamento / Segmentação	Especialista	Extração de Características	Criação de regras	Categorização	Avaliação

# Classificação de Sérias Temporais



# Introdução

- **Definição:**
  - O objetivo da classificação de séries temporais é atribuir uma **classe** a uma série (ou segmento temporal).
  - Exemplo: detectar se um sinal de ECG indica “normal” ou “arritmia”.
- **Diferença para Previsão (Forecasting):**
  - Previsão – predizer valores futuros.
  - Classificação – identificar a categoria a que pertence uma série ou subsequência.

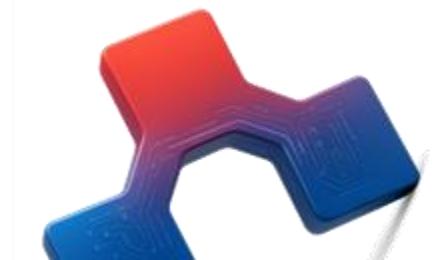


# Características de Séries Temporais para Classificação

- **Dependência temporal:** ordem dos dados é essencial.
- **Multivariadas ou univariadas:** pode haver múltiplos sensores/variáveis.
- **Dimensionalidade alta:** longas sequências tornam modelos pesados.
- **Ruído e variabilidade:** séries reais têm outliers, atrasos, mudanças de escala.
- **Invariâncias desejadas:**
  - Translação (mesmo padrão em momentos diferentes).
  - Escala (mesmo padrão em intensidades diferentes).
  - Deformação temporal (mesmo padrão em diferentes velocidades).



# Abordagens Clássicas

- **Baseadas em Distância:**
    - DTW (Dynamic Time Warping) – mede similaridade entre sequências.
    - Euclidean Distance – simples, mas pouco robusto.
  - **Baseadas em Features:**
    - Extrair estatísticas (média, variância, entropia, picos).
    - Usar essas features em classificadores tradicionais (SVM, Random Forest, XGBoost).
  - **Shapelets:** subsequências representativas que melhor discriminam classes.
- 



# Abordagens Modernas

- **Deep Learning:**
  - **RNN / LSTM / GRU:** capturam dependências temporais longas.
  - **CNNs 1D:** detectam padrões locais invariantes.
  - **Transformers temporais:** modelos de última geração, capturam dependências de longo alcance.
- **Ensembles:** HIVE-COTE, TS-CHIEF (state of the art em benchmarks UCR/UEA).
- **Self-supervised learning:** usar grandes quantidades de séries não rotuladas para pré-treino.

# Feature Engineering Específica

- **Estatísticas básicas:** média, desvio padrão, máximos/mínimos.
- **Transformadas:**
  - Fourier → captura periodicidade.
  - Wavelet → padrões locais.
- **Derivadas temporais:** variações de primeira e segunda ordem.
- **Encoding simbólico:** SAX (Symbolic Aggregate approXimation).



# Desafios na Classificação

- **Dados desbalanceados:** algumas classes muito raras (ex.: falhas industriais).
- **Alta dimensionalidade:** séries longas → custo de computação elevado.
- **Poucos dados rotulados:** rotulação manual é cara e demorada.
- **Ruído e missing values:** sensores falhando ou inconsistências.
- **Generalização:** modelos precisam funcionar em diferentes contextos e dispositivos.



# Avaliação de Modelos

- **Métricas**

- Acurácia.
- F1-Score.
- AUC-ROC.

- **Validação**

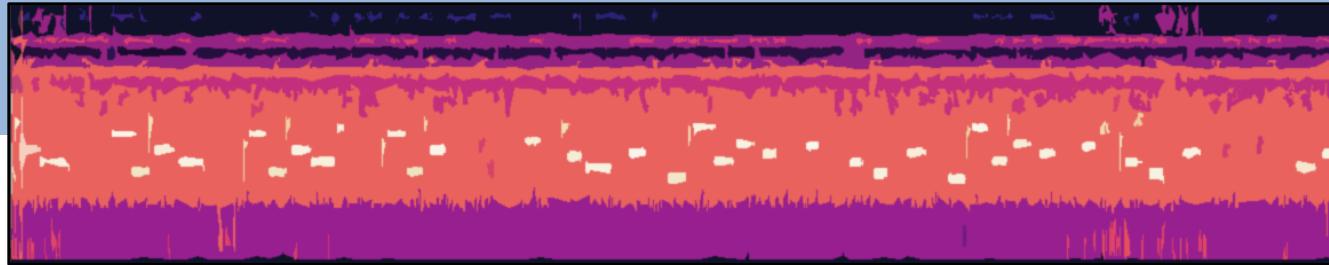
- Cross-validation estratificada por série.
- Divisão treino/teste considerando continuidade temporal.

# Conclusão

- Classificação de séries temporais é um campo essencial em aplicações reais.
- Métodos vão desde técnicas baseadas em distância até modelos de deep learning e transformers.
- Desafios principais: lidar com ruído, alta dimensionalidade e classes raras.
- **Tendência:** self-supervised + transformers para aprender representações robustas.



# Processamento de Sinais e Técnicas Relevantes





# Processamento de Sinais

- Um **sinal** é qualquer grandeza que varia no tempo, espaço ou outra variável independente.
- Pode ser uma onda sonora, uma imagem, uma série temporal ou um sinal de telecomunicação
- O processamento de sinais é o **conjunto de técnicas** e algoritmos que analisa e **modifica sinais para extrair informações úteis**.
- O objetivo é tornar os sinais mais adequados para uma aplicação específica.

# Domínios do Processamento de Sinais

- Um **sinal** é qualquer grandeza que varia no tempo, espaço ou outra variável independente.
- Pode ser uma onda sonora, uma imagem, uma série temporal ou um sinal de telecomunicação
- O processamento de sinais é o **conjunto de técnicas** e algoritmos que analisa e **modifica sinais para extrair informações úteis**.
- O objetivo é tornar os sinais mais adequados para uma aplicação específica.

# Domínios do Processamento de Sinais

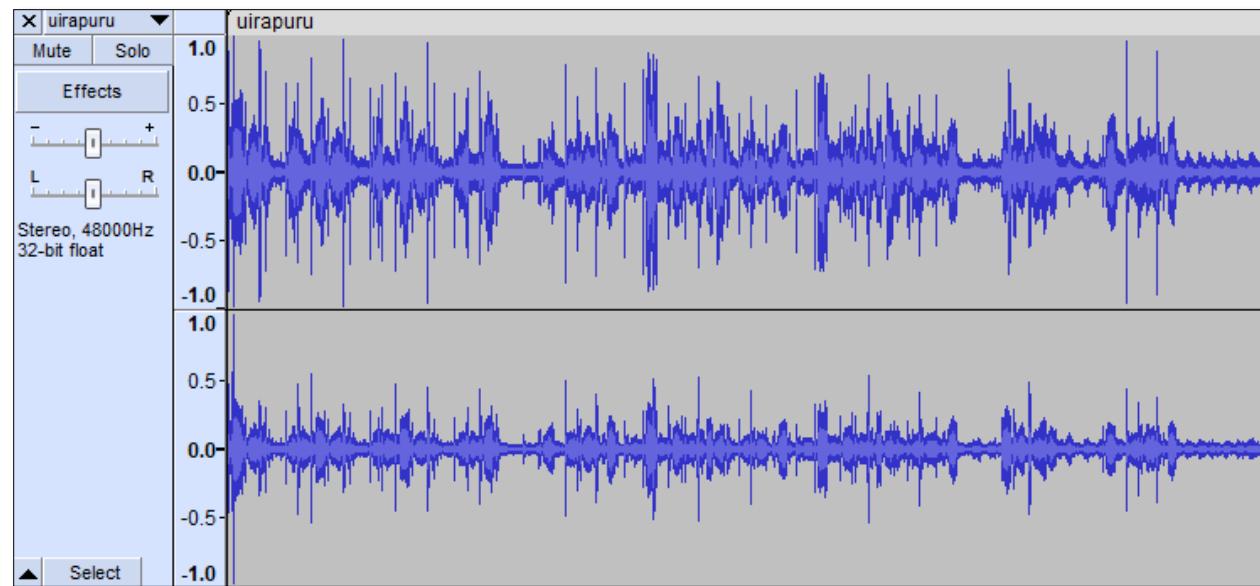
Domínio	Descrição	Aplicação
Tempo ⏳	O sinal é analisado em sua forma original (ex.: áudio bruto)	Monitoramento de sensores, análise temporal
Frequência 🎵	Analisamos quais frequências compõem o sinal usando a Transformada de Fourier	Análise espectral, filtragem, compressão
Tempo-Frequência ⚡	Mistura informações do tempo e da frequência (ex.: Wavelets)	Reconhecimento de padrões em sinais dinâmicos

# O Domínio do Tempo

- O processamento de sinais no **domínio do tempo** envolve a análise e manipulação de sinais diretamente em sua **forma original**, onde os valores do sinal são expressos em função do tempo.



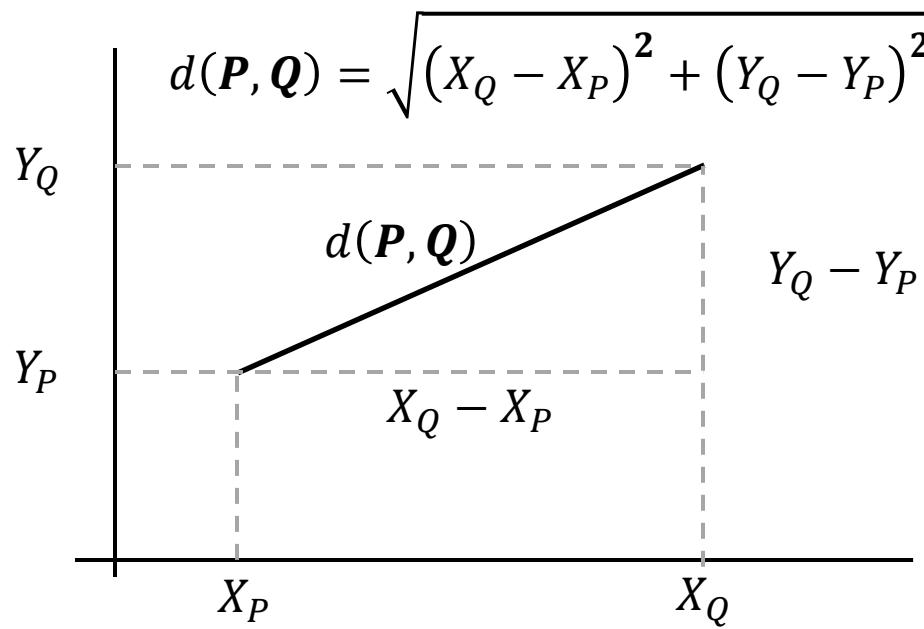
Uirapuru-ferrugíneo



# Abordagens Baseadas em Distância

# Distância Euclidiana

- Mede a similaridade ponto a ponto entre duas séries.
- Simples e rápido, mas sensível a distorções no tempo.

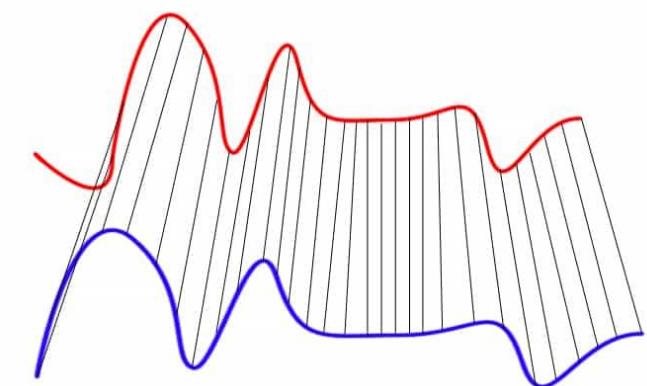
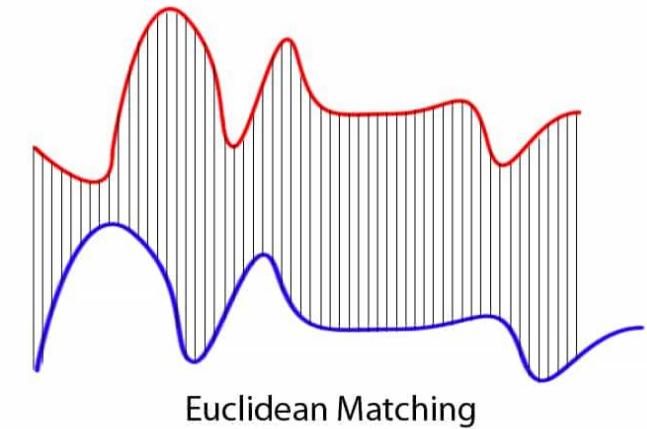


$$d(x, y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2}$$
$$x = (x_1, \dots, x_N)$$
$$y = (y_1, \dots, y_N)$$



# DTW (Dynamic Time Warping)

- DTW é um método para medir a **semelhança entre duas séries temporais** mesmo quando elas estão **desalinhadas no tempo** (uma é mais rápida/lenta, tem atrasos locais, alongamentos/encurtamentos).
- Em vez de comparar ponto a ponto (mesmo índice), o DTW **alinha elasticamente** as séries para encontrar o emparelhamento de menor custo.





# DTW (Dynamic Time Warping)

- **Como funciona (intuição)**
  1. Constrói-se uma **matriz de custo local**  $C(i, j)$  medindo a diferença entre  $x_i$  e  $y_j$ 
    - (ex.:  $\|x_i - y_j\|$  ou distância Euclidiana no caso multivariado).
  2. Procura-se um **caminho de warping** da célula (1,1) até (N,M) que minimize o **custo acumulado**, podendo andar:
    - diagonal (alinha  $x_i$  com  $y_j$ )
    - para cima (repete  $y_j$ )
    - para a direita (repete  $x_i$ )
  3. Esse caminho pode “esticar”/“comprimir” trechos para alinhar picos/vales que ocorrem em instantes diferentes.



# DTW (Dynamic Time Warping)

Característica

**Complexidade Temporal**

**Complexidade de Espaço**

**Precisão**

**Velocidade**

**Uso Ideal**

**DTW Padrão (Exato)**

$O(N^2)$

$O(N^2)$

**Exata.** Garante encontrar o caminho de alinhamento ótimo global.

Lento para séries longas.

Quando a precisão absoluta é necessária e as séries temporais são curtas. Análises acadêmicas ou científicas que exigem a solução ótima garantida.

**FastDTW**

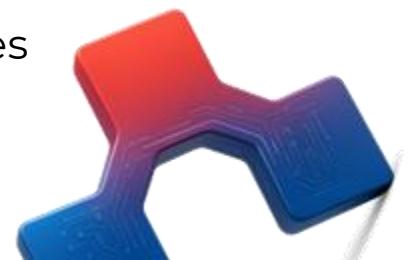
$O(N)$  (Linear, na prática)

$O(N)$

**Aproximada.** Encontra um caminho muito próximo do ótimo. A precisão depende do parâmetro `radius`.

Extremamente mais rápido, especialmente para séries longas.

Na grande maioria das aplicações práticas: clustering, classificação, busca de similaridade em grandes bancos de dados de séries temporais.

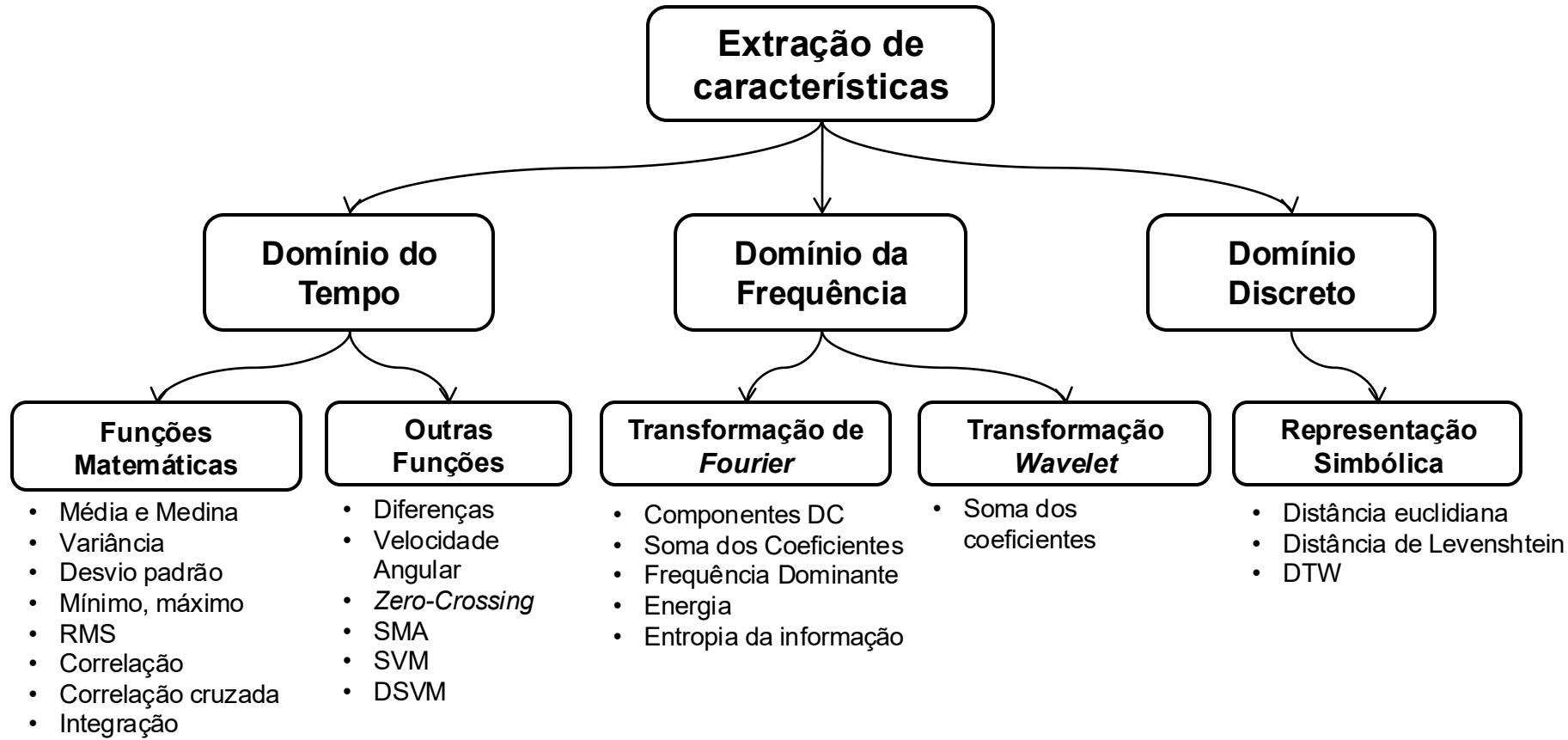




# Engenharia de Características (Feature Engineering)

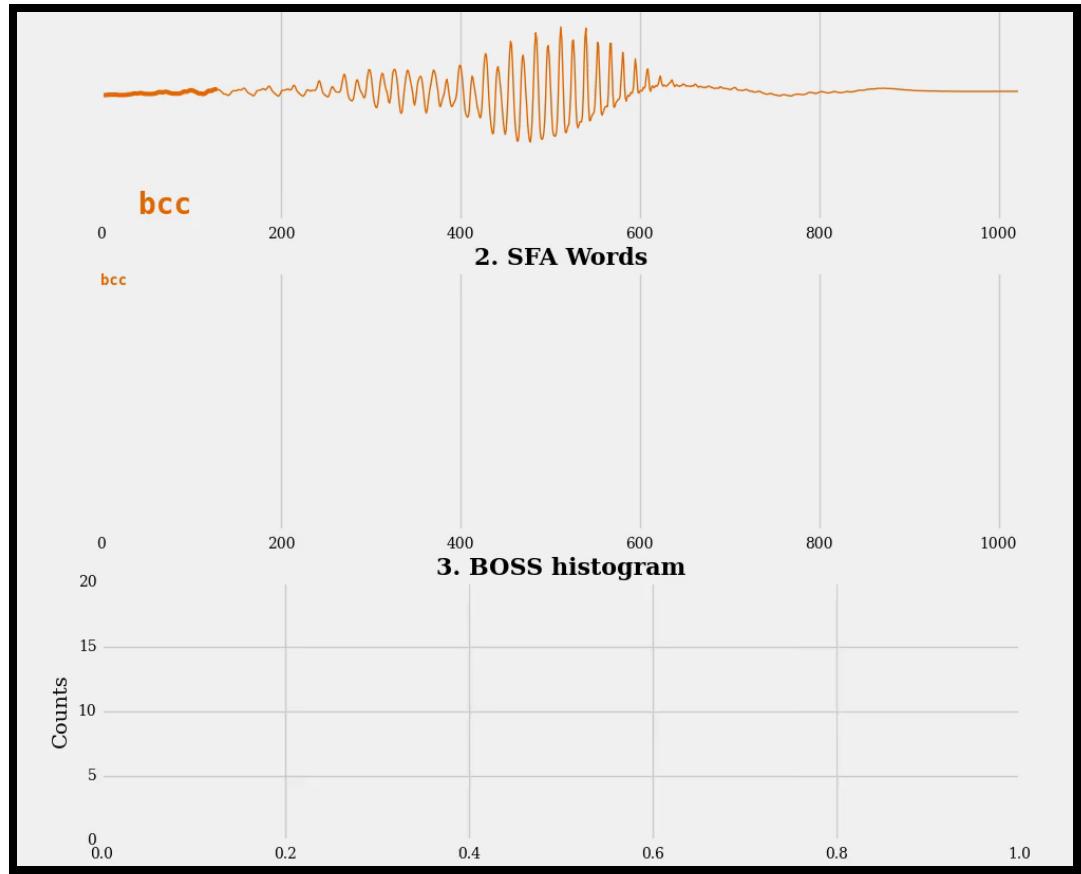
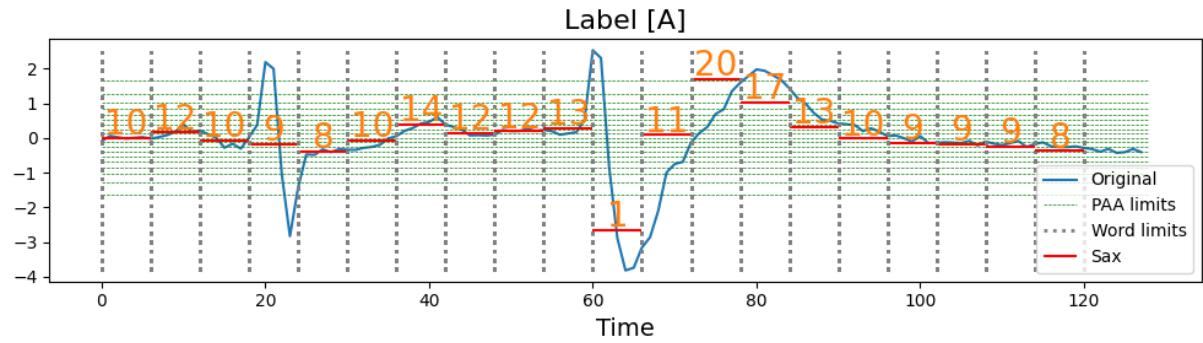
- **Estatísticas globais:** média, desvio padrão, kurtosis, entropia.
- **Lags e janelas móveis:** média móvel, variância em janelas, autocorrelação.
- **Shapelets:** subsequências curtas que melhor diferenciam classes.
- **SAX (Symbolic Aggregate approXimation):** converte séries em cadeias simbólicas → reduz dimensionalidade.

# Engenharia de Características (Feature Engineering)



# Engenharia de Características (Feature Engineering)

## Domínio Discreto

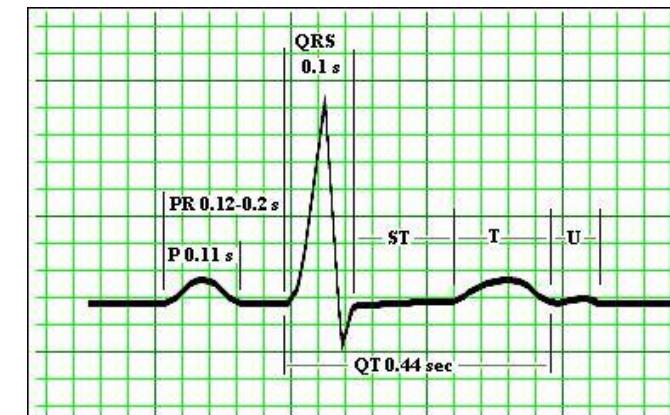
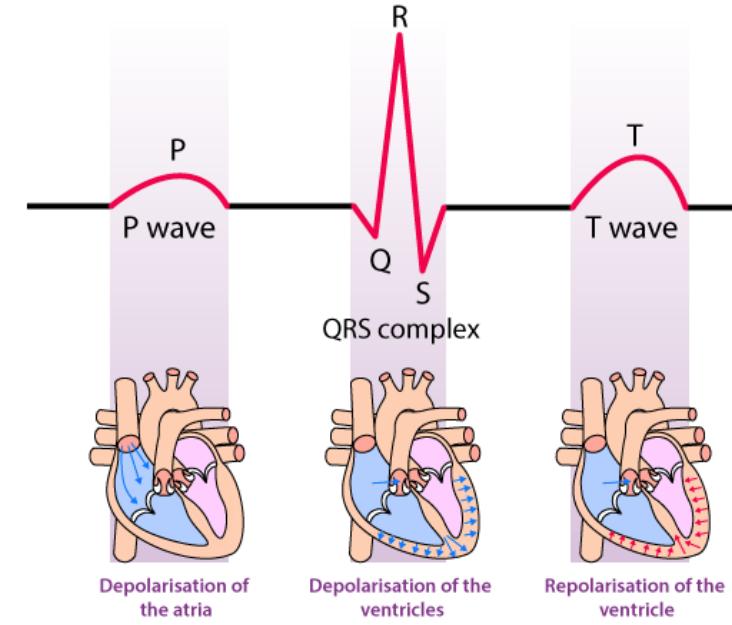


# Engenharia de Características

## Caso de uso - ECG

Como identificar o **complexo QRS** de um sinal ECG?

1. Para calcular a duração do complexo QRS em uma fita de EKG, siga estes passos:
2. Identifique o complexo QRS: Comece a medir do final do intervalo PR até o final da onda S.
3. Conte as caixas: Conte o número de pequenas caixas entre as medições.
4. Calcule a duração: Cada caixa representa 0,04 segundos. Some as caixas para obter a duração do complexo QRS. Por exemplo, se você medir 2 caixas, a duração do complexo QRS seria 0,08 segundos ( $0,02 + 0,04 + 0,04 + 0,02 = 0,08$  segundos).
5. Interprete o resultado: Uma duração normal do complexo QRS é tipicamente entre 0,06 a 0,12 segundos, o que corresponde a 1,5 a 3 pequenas caixas na fita de EKG.



# Processamento de Sinais no Domínio da Frequência

# Desvantagens do Domínio do tempo

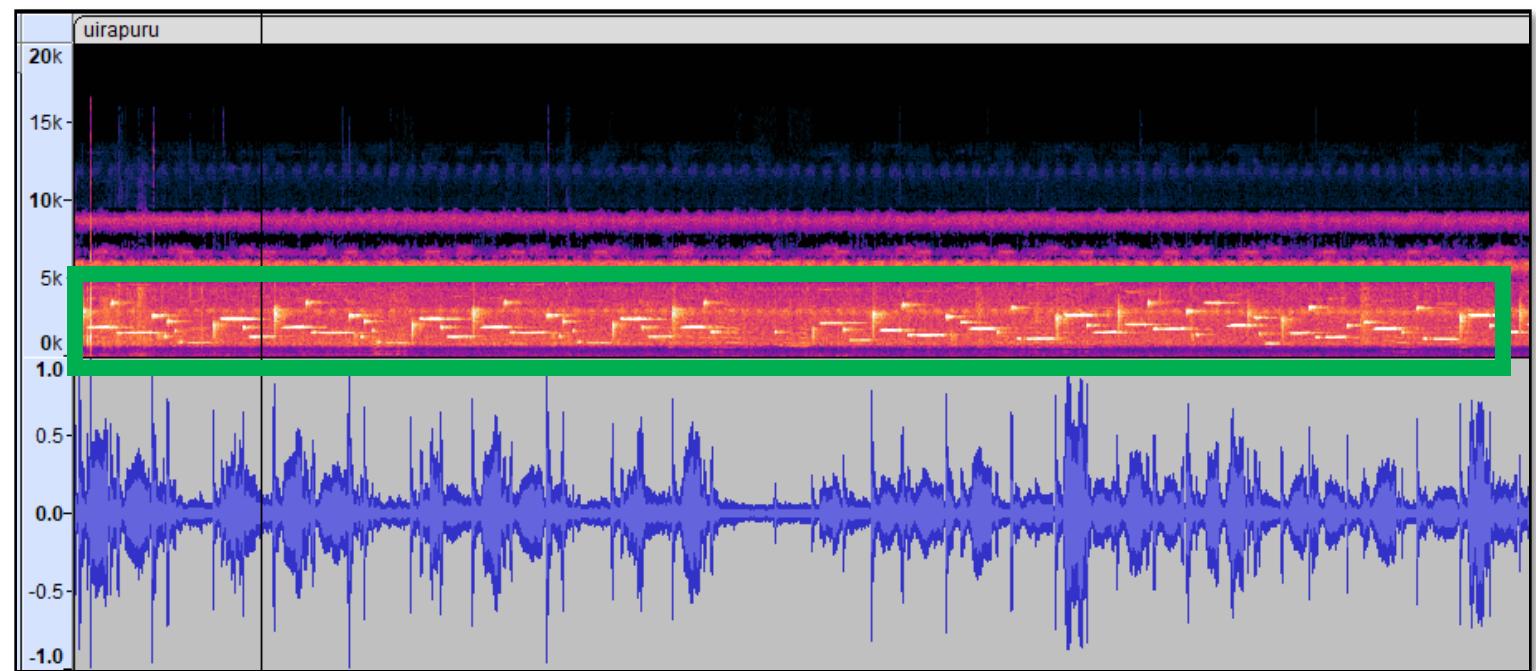
Desvantagem	Problema	Alternativa
Não revela frequências	Difícil identificar componentes espectrais	FFT, spectrograma
Padrões ocultos	Difícil reconhecer padrões repetitivos	Transformada Wavelet
Sensível a ruído	Pequenas variações mascaram informações	Filtragem digital
Não separa sinais	Mistura de sons ou eventos	Análise espectral, decomposição de sinais
Difícil detectar periodicidade	Eventos recorrentes ficam confusos	FFT, análise espectral

# O Domínio da Frequência

- Padrões aparecem claramente no espectro de frequências



Uirapuru-ferrugíneo

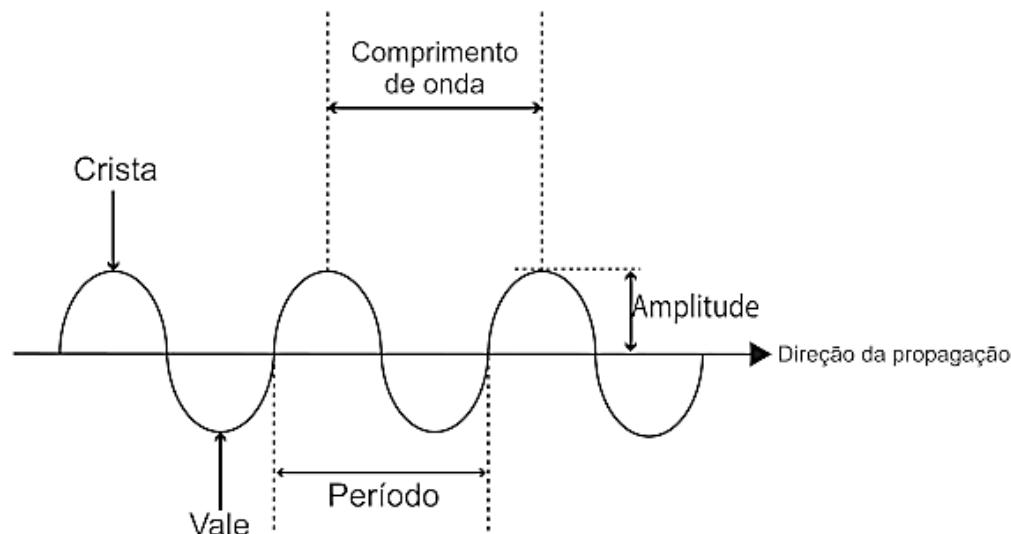


# Transformadas no Domínio da Frequência

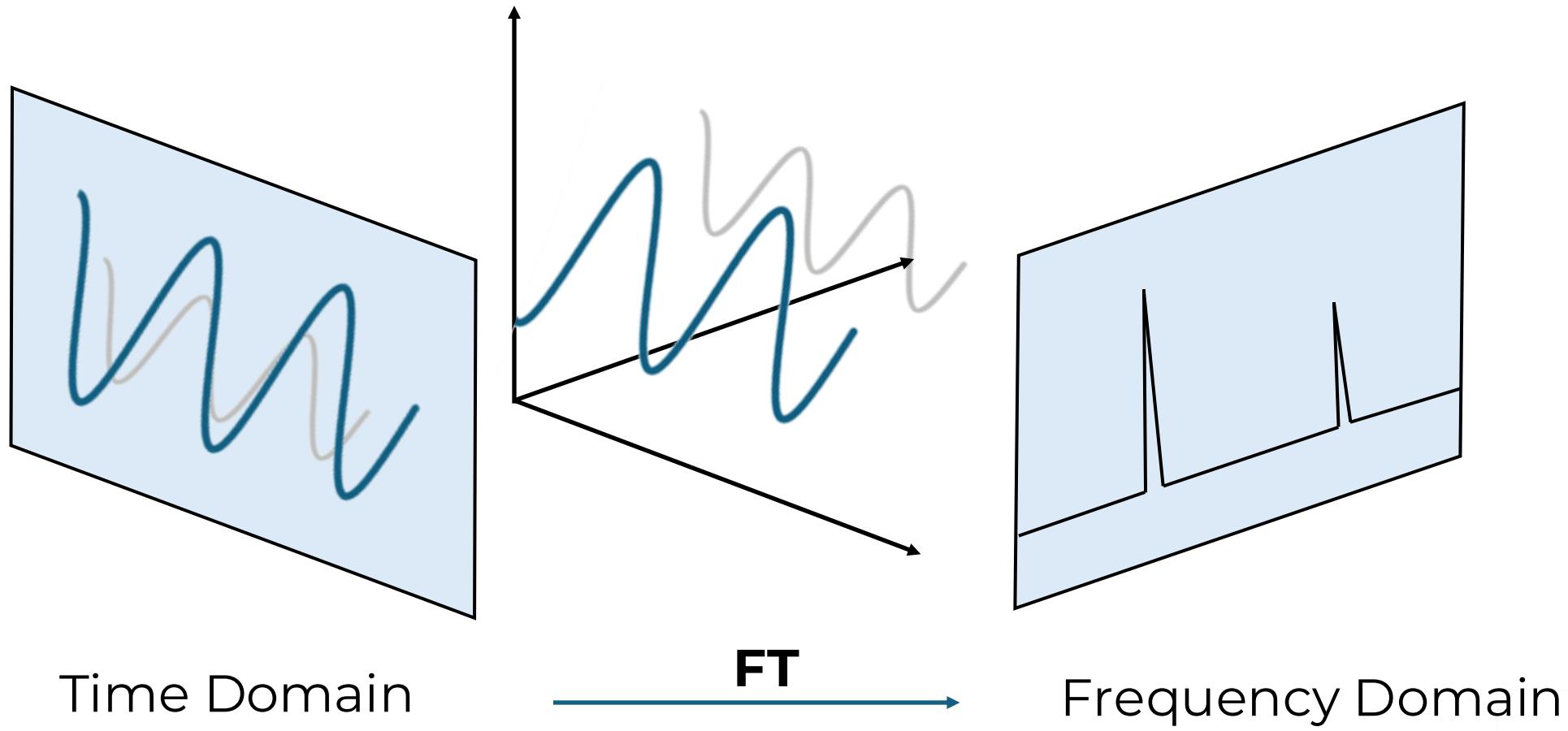
- **Transformada de Fourier (FFT):**
  - Converte a série do tempo → frequência.
  - Identifica periodicidades e componentes dominantes.
  - Útil para sinais estacionários.
- **Transformada Wavelet:**
  - Analisa série em múltiplas escalas (tempo + frequência).
  - Detecta padrões locais, útil para séries não estacionárias.
  - Aplicação: EEG, falhas mecânicas, análise de vibração.

# Transformadas no Domínio da Frequência

- **Amplitude:** são as posições de oscilação máxima (crista) e mínima (vale).
- **Período:** é o tempo necessário para a formação de um comprimento de onda, medido em segundos.
- **Frequência:** é o número de vezes que esse comprimento de onda se repete no intervalo de um segundo.



# Domínio da Frequência





# Domínio da Frequência

- A transformação do domínio do Tempo para o domínio da Frequência é feita por meio da **Transformada de Fourier**
  - **Transformada de Fourier Contínua**
  - **Transformada Discreta de Fourier (DFT)**
  - A DFT pode ser computada de forma eficiente usando a **Transformada Rápida de Fourier (FFT)**.



# Domínio da Frequência: Filtragem

## Filtros Passa-Baixa:

- **Objetivo:** Permitir a passagem de frequências abaixo de um determinado corte e atenuar frequências mais altas.
- **Aplicações:** Remoção de ruídos de alta frequência, suavização do sinal

## Filtros Passa-Alta:

- **Objetivo:** Permitir a passagem de frequências acima de um certo limite e atenuar frequências mais baixas.
- **Aplicações:** Eliminação de ruídos de baixa frequência, remoção de interferências de fundo (ex.: ruídos de vento ou zumbidos).

## Filtros Passa-Banda e Rejeita-Banda:

- **Passa-Banda:** Permite apenas um intervalo específico de frequências, útil para isolar componentes sonoras de interesse.
- **Rejeita-Banda:** Atenua um intervalo específico, eliminando frequências indesejadas (por exemplo, cancelamento de frequência de 50/60 Hz de redes elétricas).

# Domínio da Frequência: Filtragem

<b>Tipo de Filtro</b>	<b>Descrição</b>	<b>Aplicação</b>
Passa-baixa	Mantém frequências baixas e elimina as altas	Remoção de ruído de alta frequência
Passa-alta	Mantém frequências altas e elimina as baixas	Detecção de bordas em imagens
Passa-banda	Mantém um intervalo de frequências	Aplicação em rádio e telecomunicações

# Métodos do Domínio da Frequência



# FFT (Fast Fourier Transform)

- A DFT transforma um sinal do **domínio do tempo** (como um áudio bruto, onde vemos amplitude ao longo do tempo) para o **domínio da frequência** (quais frequências compõem aquele sinal).
- A FFT é um **algoritmo eficiente** para calcular a **Transformada Discreta de Fourier (DFT)**.

# FFT (Fast Fourier Transform)

Transformada	Natureza	Custo computacional
Fourier Contínua (TFC)	Integral analítica	Não aplicável (só conceito matemático)
Discreta de Fourier (DFT)	Somas duplas	$O(N^2)$
Rápida de Fourier (FFT)	Algoritmo otimizado	$O(N \log N)$

A FFT é apenas uma maneira **rápida e eficiente** de calcular essa soma (reduz o custo de  $O(N^2)$  para  $O(N \log N)$ )

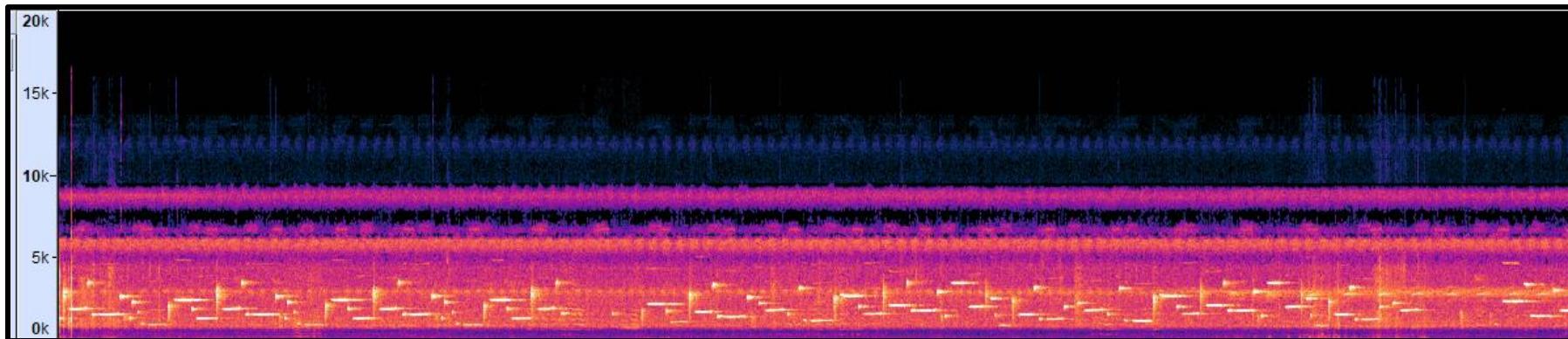
$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt$$

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}, k = 0, 1, \dots, N - 1$$

$x[n]$	Sinal no tempo
$X[k]$	Componente da frequência
$N$	Número de pontos
$e^{-j\omega n}$	Parte da transformada de Fourier

# STFT (Short-Time Fourier Transform)

- A **STFT** é a **Transformada de Fourier de Curto Prazo**.
- Enquanto a Transformada de Fourier (FFT) analisa um sinal inteiro de uma vez, a STFT divide o sinal em **janelas pequenas** no tempo e aplica a FFT em cada pedaço.
- Assim conseguimos ver **como as frequências mudam ao longo do tempo**.



$$X(\tau, \omega) = \sum_{n=-\infty}^{\infty} x[n] \cdot w[n - \tau] \cdot e^{-j\omega n}$$

$x[n]$	Sinal no tempo
$w[n - \tau]$	Janela centrada no tempo $\tau$
$H_m(k)$	Filtros triangulares na escala Mel
$e^{-j\omega n}$	Parte da transformada de Fourier

# STFT (Short-Time Fourier Transform)

## 1. Dividir o sinal em janelas

O áudio é cortado em pedaços curtos (20–40 ms). Isso porque a fala e sons mudam rápido.

## 2. Aplicar janela de suavização

Multiplicamos cada pedaço por uma função janela (*Hamming, Hann, etc.*) para evitar “bordas” artificiais.

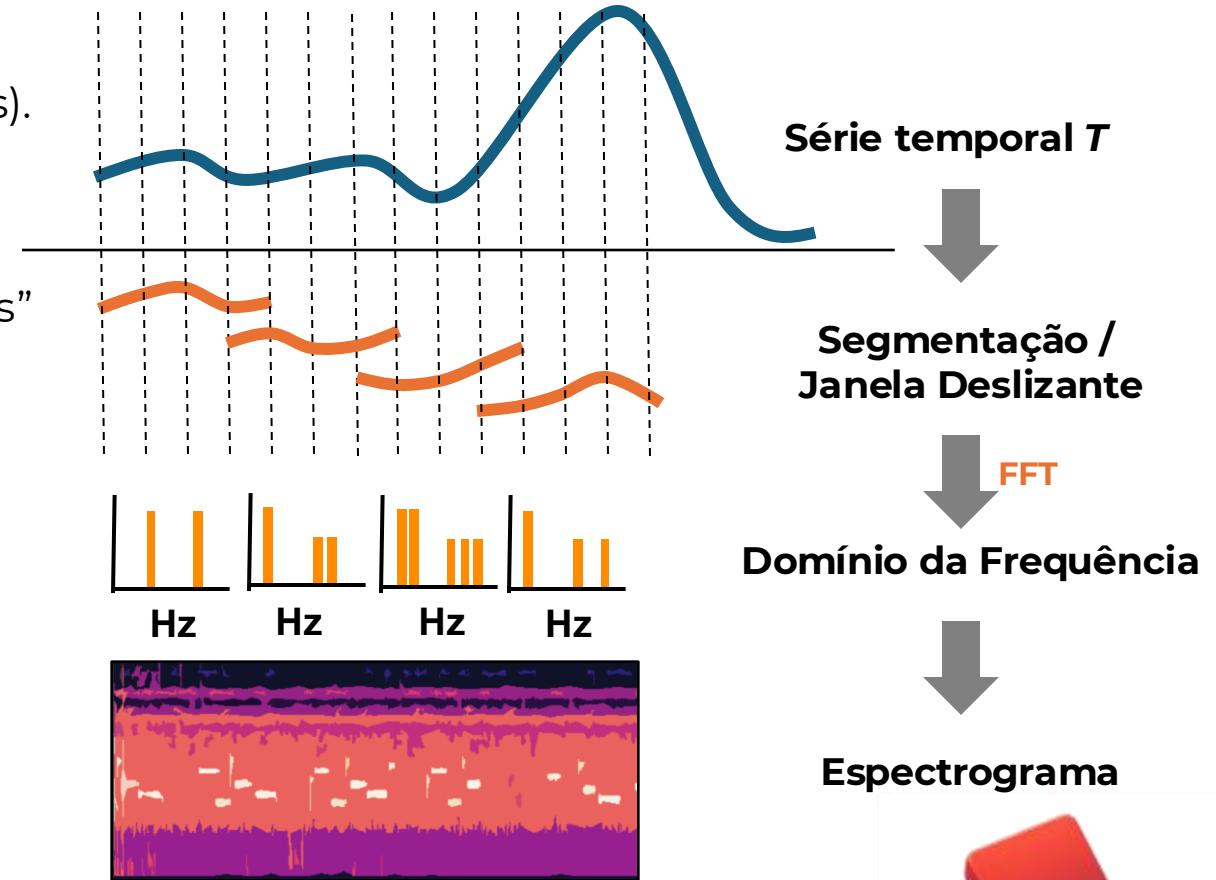
## 3. Aplicar FFT em cada janela

Isso gera o espectro de frequências de cada pedaço.

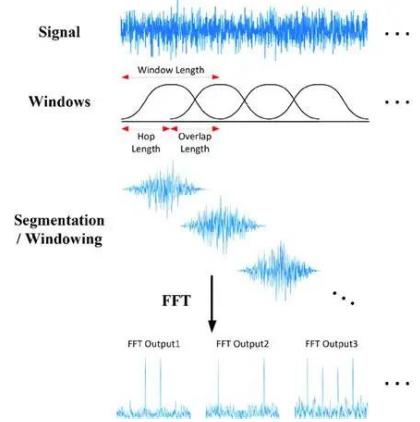
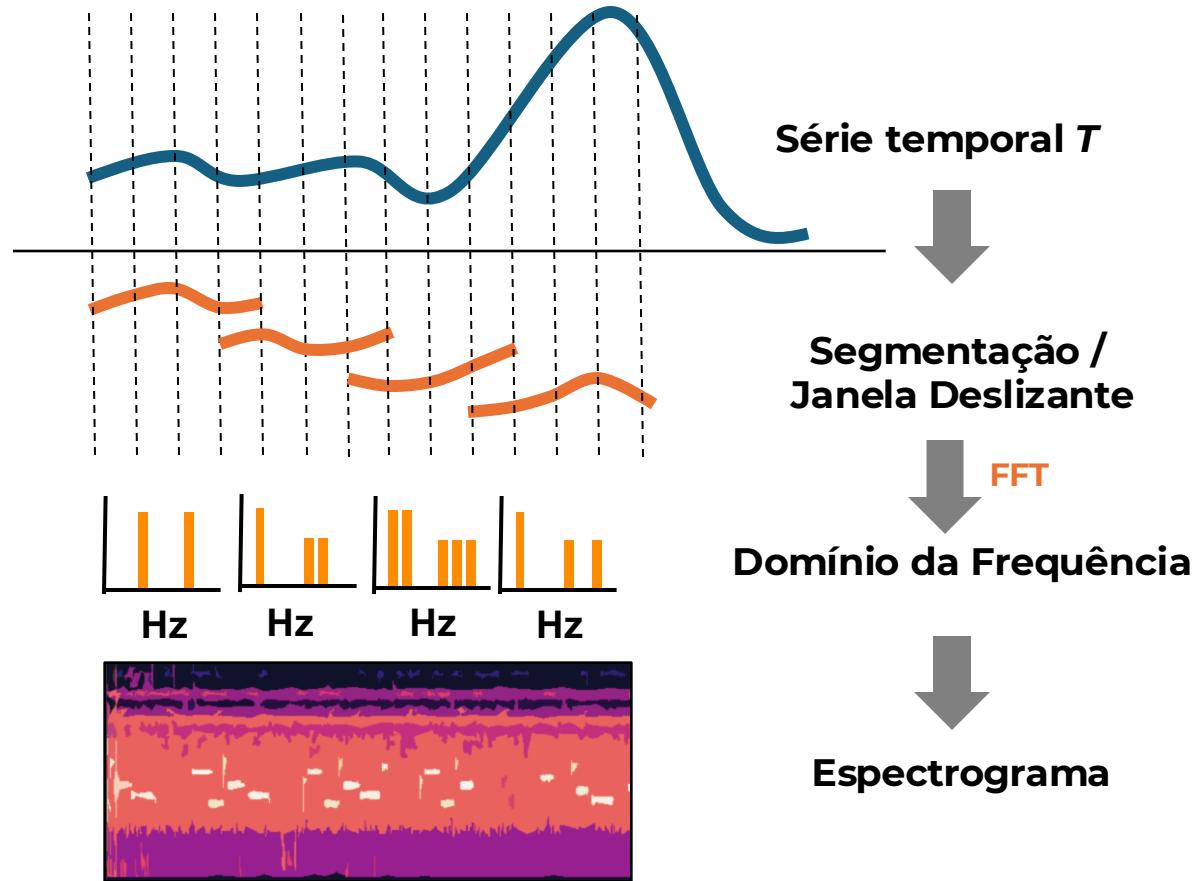
## 4. Montar o espectrograma

Empilhamos os espectros no tempo → criamos uma matriz (tempo × frequência).

A intensidade das cores mostra a energia em cada frequência



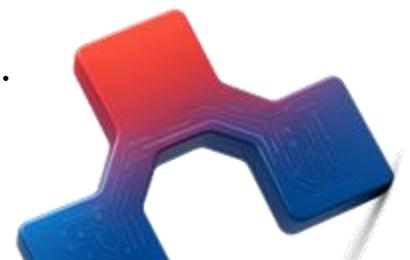
# STFT (Short-Time Fourier Transform)





# MFCC (Mel-Frequency Cepstral Coefficients)

- MFCC é uma **técnica de extração de características** (features) usada em processamento de sinais de áudio, principalmente em **fala, música e sons ambientais**.  
Ele transforma o sinal de áudio cru em uma representação numérica que captura **as características perceptíveis pelo ouvido humano**.
- ✓ **Reconhecimento de fala** (ex.: assistentes virtuais como Alexa, Siri).
- ✓ **Reconhecimento de música ou sons** (ex.: Shazam).
- ✓ **Processamento biométrico de voz** (identificação de pessoas).
- ✓ **Classificação de sons** (ex.: detectar ruídos em máquinas, sons da natureza etc.).



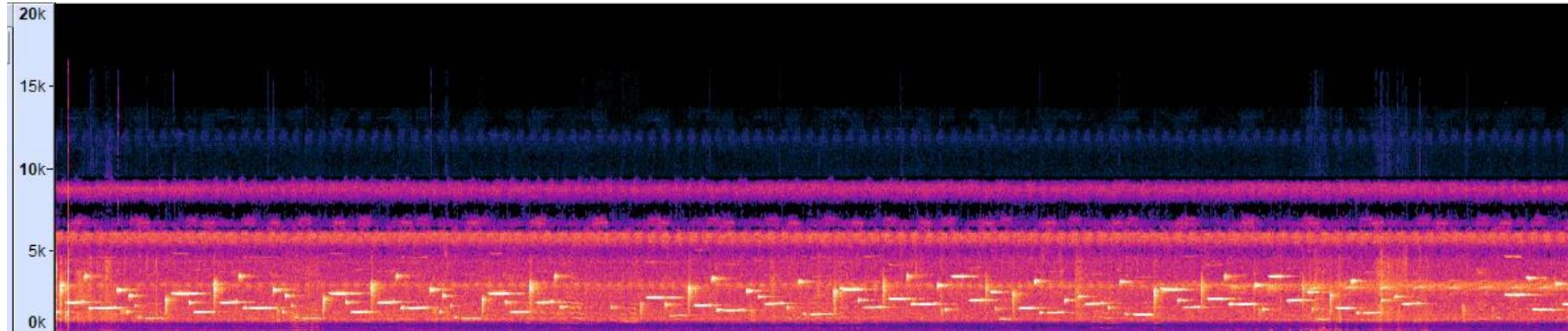


# MFCC (Mel-Frequency Cepstral Coefficients)

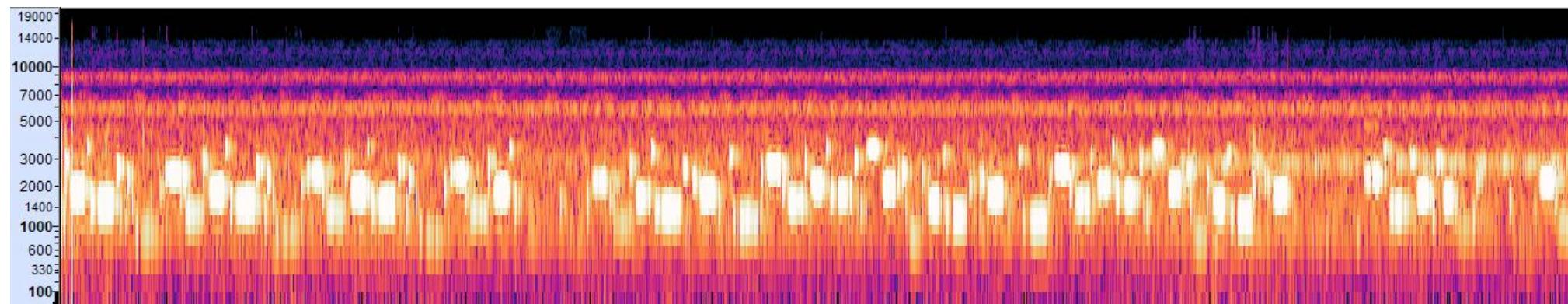
Característica	STFT 	MFCC 
Saída	Espectrograma completo (tempo × frequência × intensidade).	Vetor pequeno de coeficientes (resumo do espectro).
Foco	Física/matemática do sinal.	Percepção humana do som (psicoacústica).
Tamanho dos dados	Muito grande (matriz 2D).	Muito compacto (12–13 valores por frame).
Usado em	Análise de áudio, música, engenharia de som.	Reconhecimento de fala, classificação, ML.



# STFT vs MFCCs



**STFT**



**MFCC**



# MFCC (Mel-Frequency Cepstral Coefficients)

Imagine que você grava alguém falando. O MFCC faz o seguinte:

1. **Divisão em janelas (frames)**

O sinal de áudio é muito longo e varia no tempo. Então dividimos em **pequenos pedaços (20–40 ms)** para analisar melhor.

2. **Transformada de Fourier (FFT)**

Cada pedaço é transformado para o **espectro de frequências** (descobre quais frequências compõem aquele som).

3. **Escala Mel**

Como o ouvido humano **não percebe todas as frequências da mesma forma**, convertemos o espectro para a **escala Mel**, que é mais parecida com nossa percepção auditiva (por exemplo, percebemos melhor graves e médios do que agudos muito altos).

4. **Filtro triangular (Mel filterbank)**

Aplica-se um conjunto de filtros que simulam como o ouvido "enfatiza" certas bandas de frequência.

5. **Logaritmo da energia**

O ouvido humano percebe intensidade de forma logarítmica (um som 10x mais intenso não parece 10x mais alto). Por isso aplicamos log.

6. **Transformada Discreta do Cosseno (DCT)**

Comprime e remove redundâncias, gerando os **coeficientes cepstrais**. Normalmente usamos os primeiros 12–13 números, que resumem bem o espectro.



# MFCC (Mel-Frequency Cepstral Coefficients)

Imagine que você grava alguém falando. O MFCC faz o seguinte:

**1. Divisão em janelas (frames)**

O sinal de áudio é muito longo e varia no tempo. Então dividimos em **pequenos pedaços (20–40 ms)** para analisar melhor.

**2. Transformada de Fourier (FFT)**

Cada pedaço é transformado para o **espectro de frequências** (descobre quais frequências compõem aquele som).

**3. Escala Mel**

Como o ouvido humano **não percebe todas as frequências da mesma forma**, convertemos o espectro para a **escala Mel**, que é mais parecida com nossa percepção auditiva (por exemplo, percebemos melhor graves e médios do que agudos muito altos).

**4. Filtro triangular (Mel filterbank)**

Aplica-se um conjunto de filtros que simulam como o ouvido "enfatiza" certas bandas de frequência.

**5. Logaritmo da energia**

O ouvido humano percebe intensidade de forma logarítmica (um som 10x mais intenso não parece 10x mais alto). Por isso aplicamos log.

**6. Transformada Discreta do Cosseno (DCT)**

Comprime e remove redundâncias, gerando os **coeficientes cepstrais**. Normalmente usamos os primeiros 12–13 números, que resumem bem o espectro.



# Resumo

Representação	Retém Informação Original?	Mostra Frequências?	Preserva Tempo?	Aplicações
Dado Bruto (Raw Audio)	<input checked="" type="checkbox"/> Sim	<input checked="" type="checkbox"/> Não	<input checked="" type="checkbox"/> Sim	Processamento direto do sinal
FFT (Transformada de Fourier)	<input checked="" type="checkbox"/> Não	<input checked="" type="checkbox"/> Sim (espectro)	<input checked="" type="checkbox"/> Não	Análise espectral, filtragem
MFCC	<input checked="" type="checkbox"/> Não	<input checked="" type="checkbox"/> Sim (perceptível)	<input checked="" type="checkbox"/> Parcialmente	Reconhecimento de fala, classificação de áudio

# Resumo

Representação	Informação Preservada	Vantagens ✓	Desvantagens ✗
Dado Bruto (Raw Audio)	● ● ● Máxima informação (tempo, amplitude)	Retém todos os detalhes, ideal para reconstrução do áudio	Alto custo computacional, difícil extrair padrões
FFT (Transformada de Fourier)	● ● Todas as frequências (mas perde tempo)	Permite analisar espectro completo	Não mantém informações temporais
MFCC	● Apenas frequências relevantes à audição humana	Compacta dados para ML, robusto para reconhecimento de voz	Perde detalhes do áudio, não pode ser reconstruído



# Wavelets

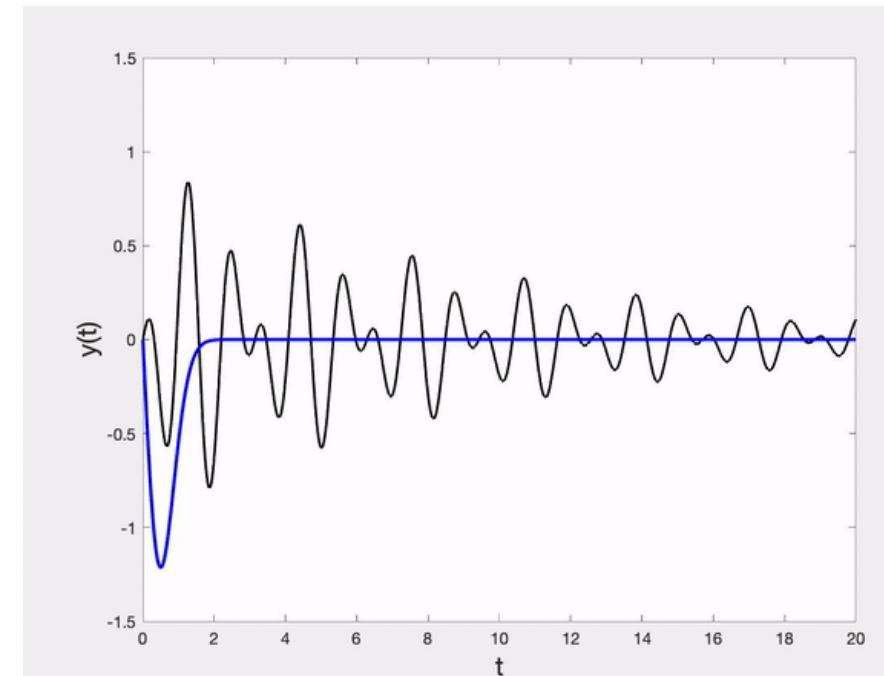
Calcular uma wavelet significa projetar o sinal em “ondinhas” (wavelets) em **várias escalas (a)** e posições no tempo **(b)**, para descobrir **quanta energia** do sinal aparece em cada escala e em que instante. Existem dois jeitos principais:

- **CWT (Continuous Wavelet Transform)**: análise “densa” em muitas escalas e deslocamentos.
- **DWT (Discrete Wavelet Transform)**: análise eficiente via **banco de filtros** (passa-baixas/altas + dizimação), base da compressão e da denoising.
- **Regras rápidas de escolha**
  - Quer mapa tempo-freq simples e dados localmente estáveis? → **STFT**.
  - Tem transientes/escala múltipla (picos, “chirps”, bordas) ou precisa denoising/compactação? → **Wavelet** (DWT p/ eficiência; CWT p/ análise rica).

# Wavelets

Calcular uma wavelet significa projetar o sinal em “ondinhas” (wavelets) em **várias escalas (a)** e posições no tempo **(b)**, para descobrir **quanta energia** do sinal aparece em cada escala e em que instante. Existem dois jeitos principais:

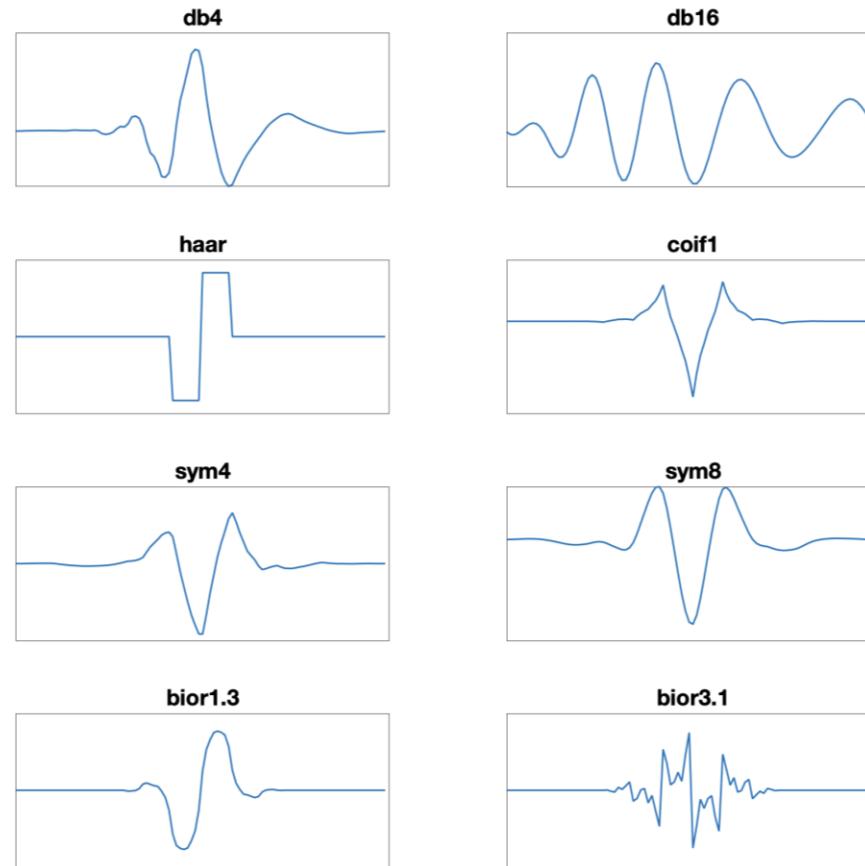
1. Escolhemos uma wavelet de uma escala particular (como a wavelet azul no gif).
2. Em seguida, deslizamos essa wavelet por todo o sinal, ou seja, variamos sua localização, onde a cada passo do tempo multiplicamos a wavelet e o sinal.
3. O produto dessa multiplicação nos dá um coeficiente para aquela escala de wavelet naquele passo do tempo.
4. Depois, aumentamos a escala da wavelet (por exemplo, as wavelets vermelha e verde) e repetimos o processo.



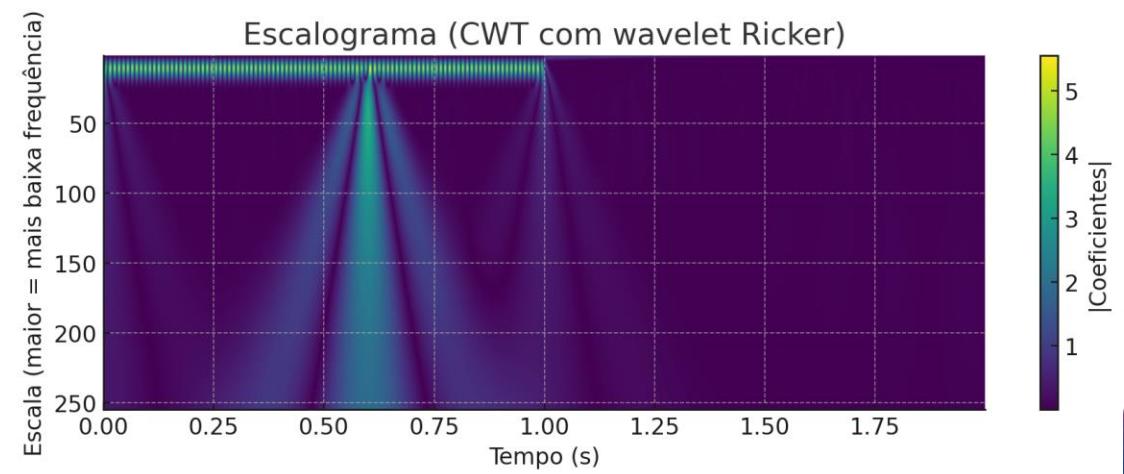
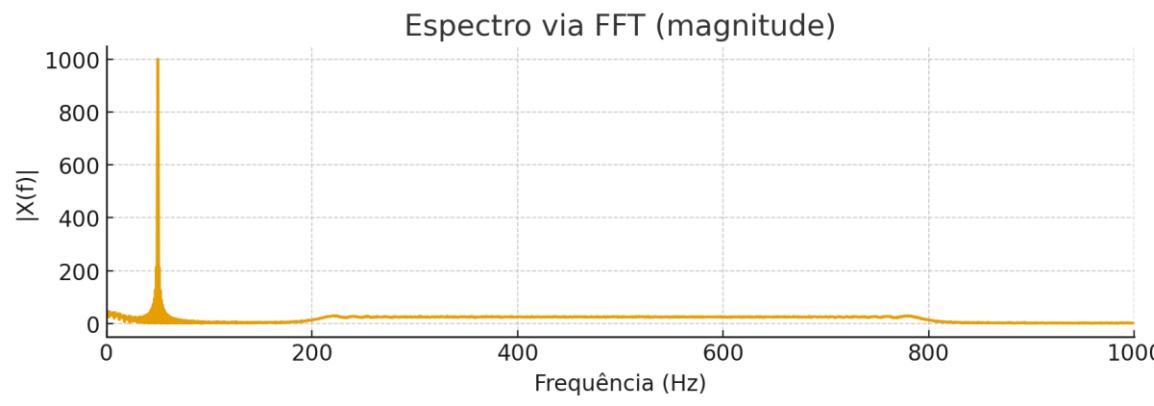
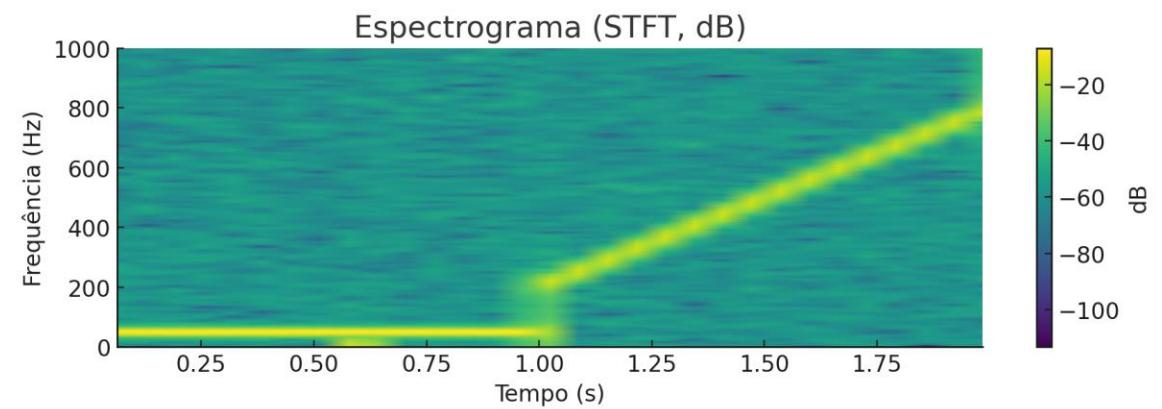
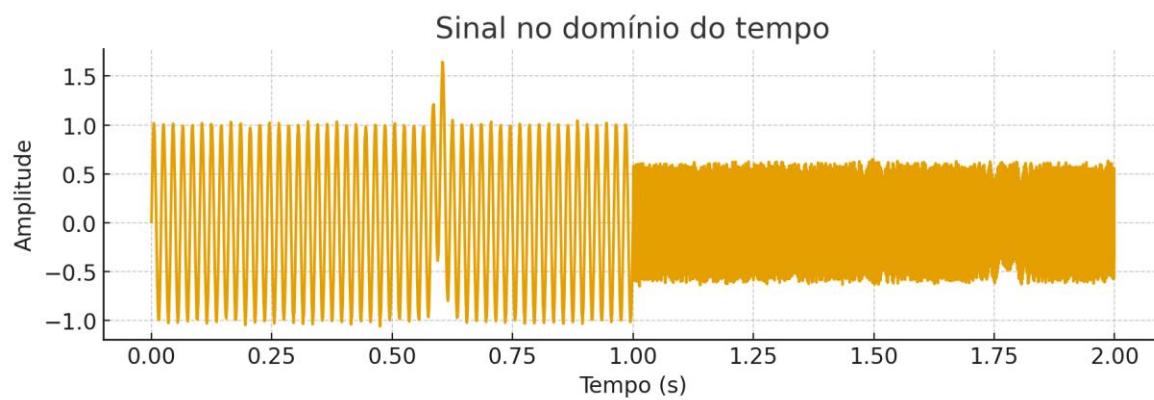
# Wavelets

A lição aqui é que se você souber qual forma característica está tentando extrair do seu sinal, há uma grande variedade de wavelets para escolher para melhor corresponder a essa forma.

A **onda-mãe é o “molde”**: escalando e deslizando esse molde você “varre” o sinal, medindo **semelhança local por banda**. Se o molde combina bem com a estrutura do seu sinal, os coeficientes “acendem” no tempo e na escala certos — revelando **onde e em que largura/frequência** os eventos acontecem.



# Wavelets



# Wavelets

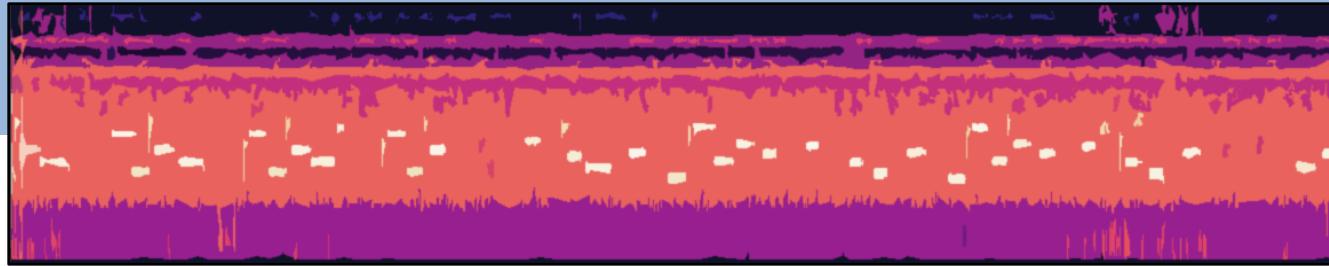
Aspecto	STFT (Short-Time Fourier Transform)	Wavelet (CWT/DWT)
Ideia central	FFT aplicada em janelas fixas ao longo do tempo.	Correlaciona o sinal com cópias escaladas e deslocadas de uma wavelet-mãe.
Saída típica	Espectrograma (tempo × frequência × energia).	Escalograma (tempo × escala/frequência) na CWT; coeficientes multi-nível (A/D) na DWT.
Resolução tempo-freq.	Fixa (trade-off da janela: janela curta → boa no tempo/ruim em freq; janela longa → o inverso).	Adaptativa: boa no tempo p/ altas freq. e boa em freq. p/ baixas (análise multi-resolução).
Parâmetros-chave	Tipo/tamanho de janela (Hann, Hamming), overlap, tamanho da FFT.	Wavelet-mãe (Haar, Daubechies, Morlet, etc.), escalas (CWT) ou níveis (DWT).
Kernels/base	Senos/cossenos “infinitos” janelados.	Wavelets localizadas e média zero; família escalada/shiftada.
Sensibilidade a transitórios	Pode “espalhar” energia se a janela for inadequada.	Muito boa (wavelets captam transientes e bordas).
Representação compacta	Matriz densa (muitos bins).	DWT é bem compacta (coeficientes A/D); CWT é densa (sem dizimação).
Custo computacional	Aproximadamente $O(F \cdot N_w \log N_w) O(F \cdot N_w \log N_w)$ (FFT por frame: tamanho $N_w \times N_w \times$ n° de frames FF).	CWT: $O(N \cdot S) O(N \cdot S)$ ( $S =$ n° de escalas). DWT: $\sim O(N) O(N)$ (muito eficiente).

# Wavelets

Aspecto	STFT (Short-Time Fourier Transform)	Wavelet (CWT/DWT)
Reconstrução perfeita	Possível, mas menos usada no dia a dia do espectrograma.	DWT (bancos de filtros) suporta reconstrução perfeita; CWT também, com condições.
Shift-invariance	Depende do desenho (overlap elevado ajuda).	DWT padrão não é invariante a deslocamentos; SWT (undecimated) melhora isso.
Casos ideais	Sinais quase-estacionários localmente (fala, notas musicais sustentadas).	Sinais não-estacionários com eventos rápidos (ECG/EEG, vibrações, falhas, transientes).
Exemplos de uso	MFCC, análise de formantes, música, ruído de máquinas.	Denoising, detecção de QRS em ECG, compressão (JPEG2000), detecção de bordas.
Vantagens	Simples, amplamente disponível, interpretação direta em frequência.	Multi-resolução, ótimo para transientes, DWT é leve e com reconstrução perfeita.
Limitações	Trade-off fixo da janela; pode perder detalhes locais.	Escolha da wavelet/escala impacta muito; CWT pode ser custosa.

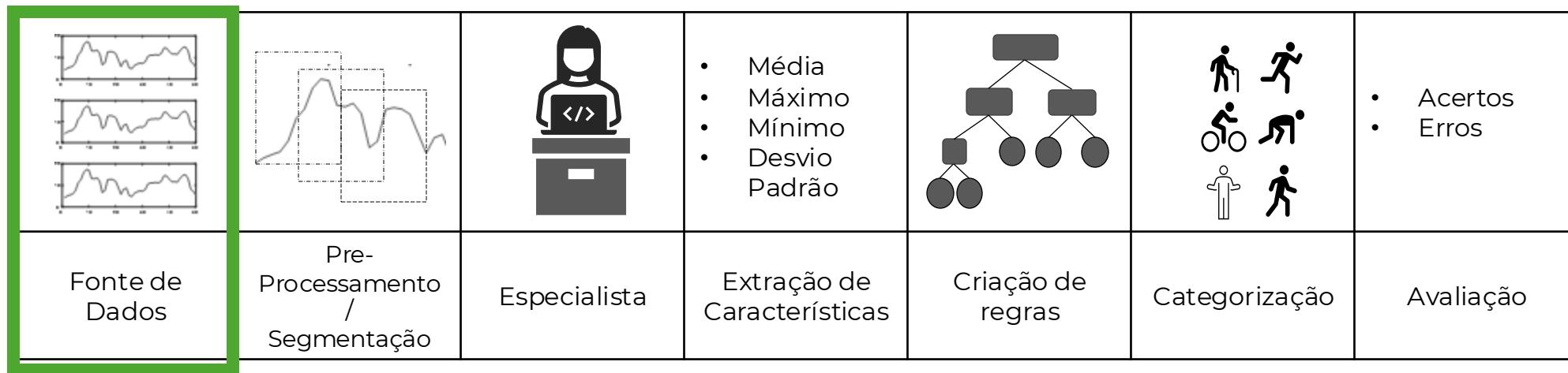


# Pipeline de Machine Learning para Classificação de Séries Temporais



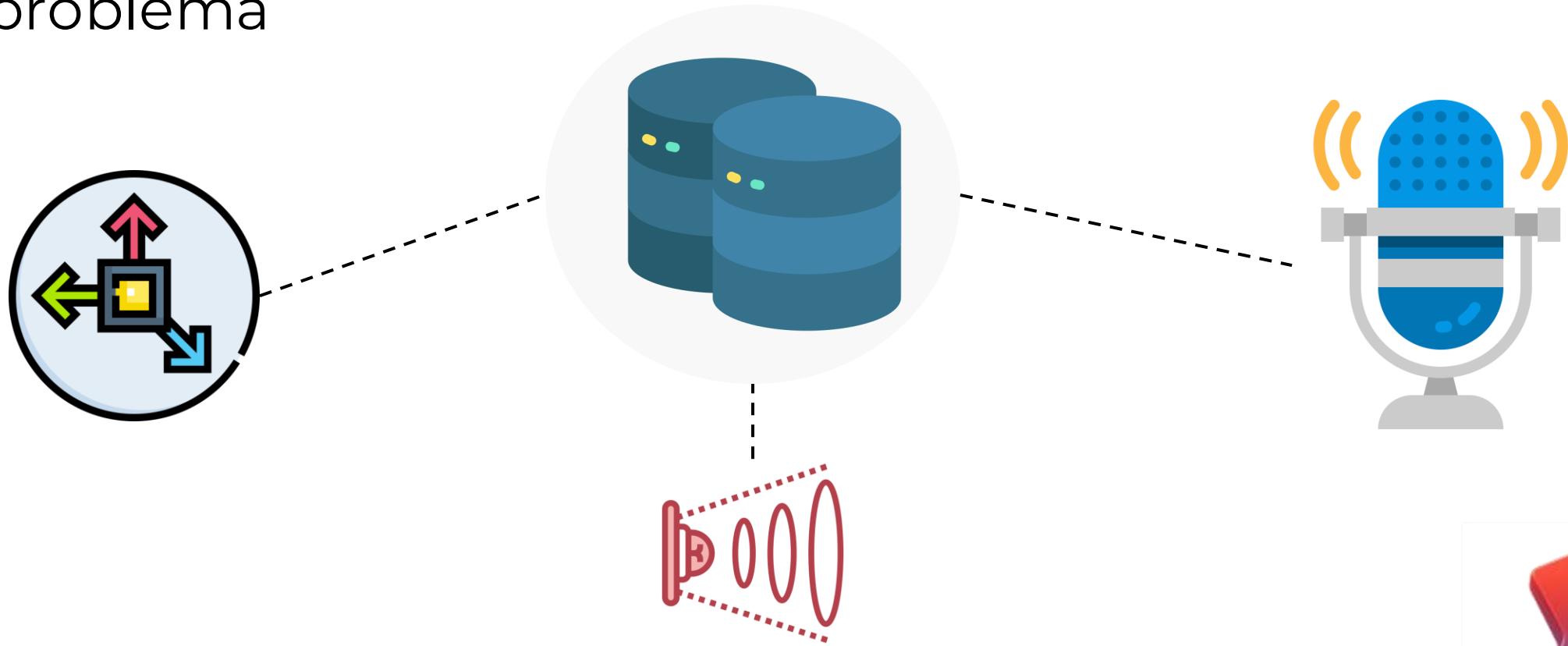
# Overview

- Visão geral de uma pipeline de processamento de sinais



# Aquisição de Dados

- Dados adquiridos de acordo com a natureza do problema



# Datasets

Class	Symbols	Description
N	N, ., L, R, e, j	Non-ectopic beats
S	A, a, J, S, x	SVEB (Supraventricular ectopic beat)
V	V, E, !	VEB (Ventricular ectopic beat)
F	F	Fusion beat
Q	P, /, f, u, Q	Unknown beat

Table 5-2: Beat Classifications according to AAMI standard.

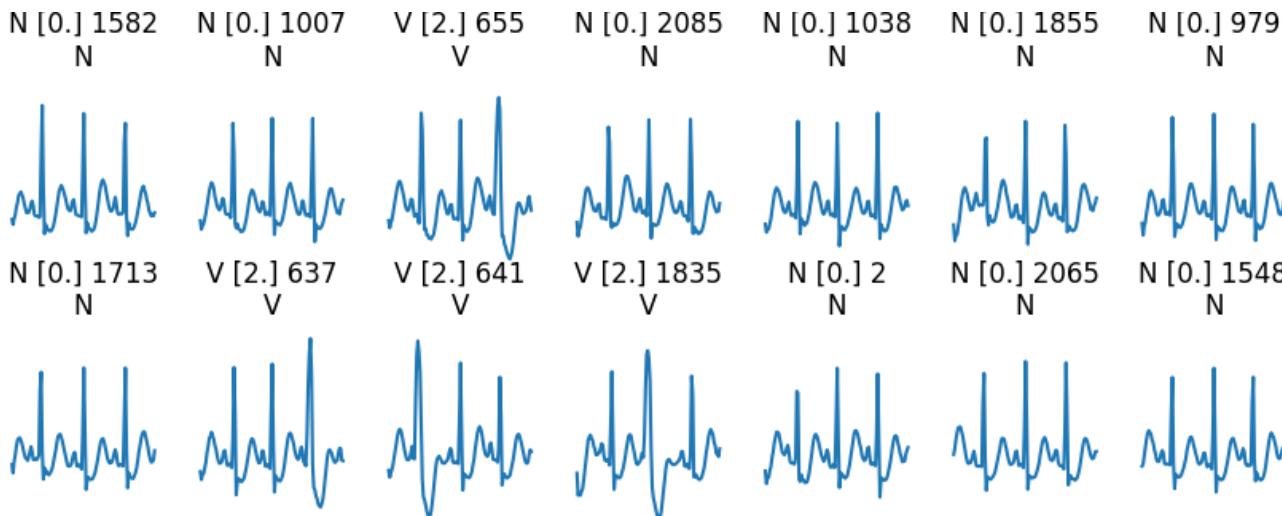


Figure 5-2: ECG heartbeats samples from processed PhysioNet arrhythmia dataset

Symbol	Description
N	Normal beat
.	Normal beat
L	Left bundle branch block beat
R	Right bundle branch block beat
A	Atrial premature beat
a	Aberrated atrial premature beat
J	Nodal (junctional) premature beat
S	Supraventricular premature beat
V	Premature ventricular contraction
F	Fusion of ventricular and normal beat
[	Start of ventricular flutter-fibrillation
!	Ventricular flutter wave
]	End of ventricular flutter-fibrillation
e	Atrial escape beat
j	Nodal (junctional) escape beat
E	Ventricular escape beat
/	Paced beat
f	Fusion of paced and normal beat
x	Non-conducted P-wave (blocked APB)
Q	Unclassifiable beat

Table 5-3: Symbol Definitions for ECG Beat Types.

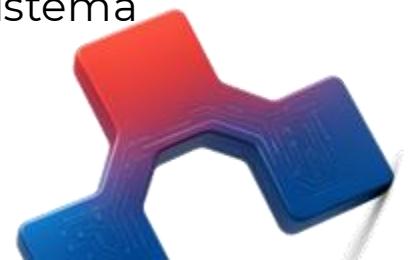
# Pré-Processamento

- Visão geral de uma pipeline de processamento de sinais





# Pré-Processamento

- A **normalização** ajusta os valores do sinal para um **intervalo específico**, geralmente  $[0,1]$  ou  $[-1,1]$
  - Vantagens:
    - Mantém a proporção dos valores originais.
    - Evita que um valor domine os outros ao treinar modelos de ML.
  - Desvantagens:
    - Sensível a outliers, pois valores extremos distorcem a escala.
  - Exemplo de Aplicação:
    - Áudio digitalizado é frequentemente normalizado para o intervalo  $[-1,1]$  para evitar distorções.
    - Sinais de sensores podem ser normalizados para ajustar à faixa de operação de um sistema eletrônico.
- 



# Pré-Processamento

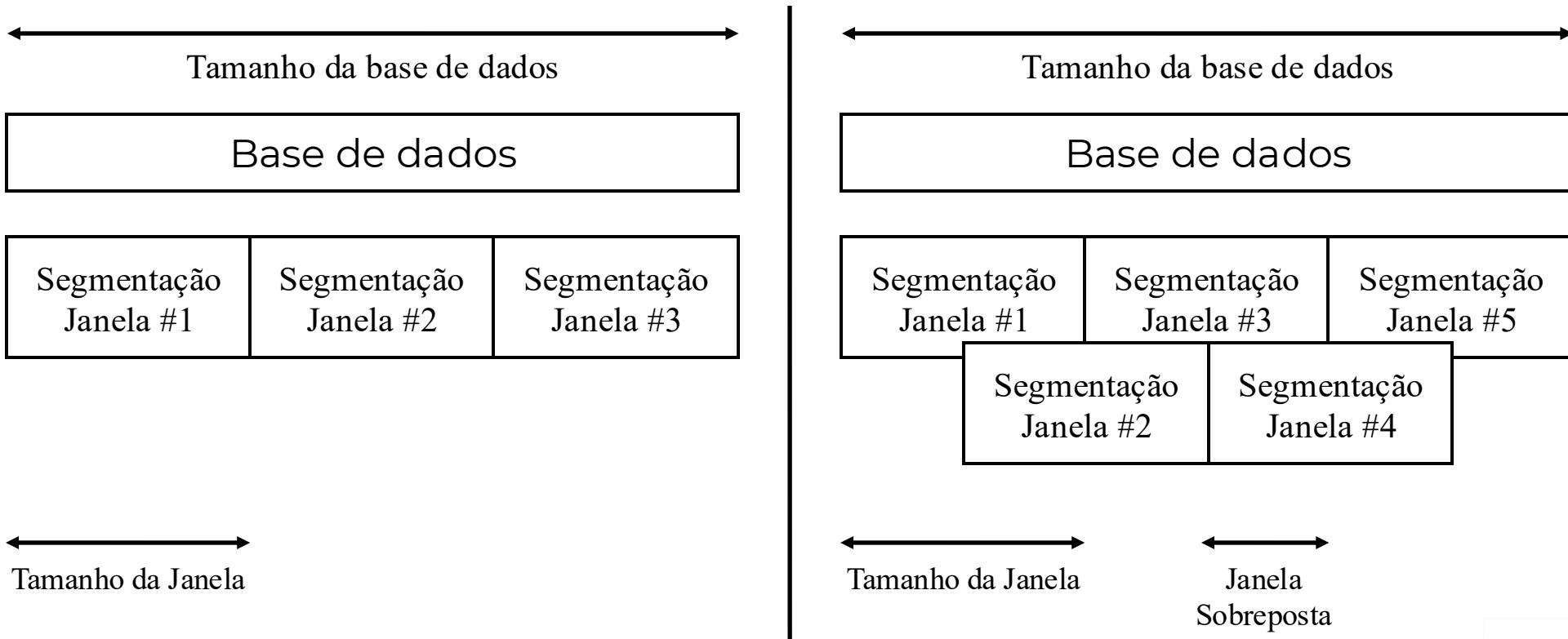
- A **padronização** transforma os dados para terem **média zero e desvio padrão unitário**
  - Vantagens:
    - Reduz influência de escalas diferentes, útil para algoritmos como PCA e SVM.
    - Menos sensível a outliers do que a normalização.
  - Desvantagens:
    - Pode distorcer a amplitude original do sinal.
    - Difícil interpretar os valores após a transformação.
  - **Exemplo de Aplicação:**
    - Sinais biomédicos (ECG, EEG) são padronizados para remover variações entre diferentes pacientes.
    - Sinais de sensores podem ser padronizados antes da análise de anomalias.
- 

# Pré-Processamento

- Muitos sinais contêm **ruídos indesejados**, que podem ser removidos com **filtros digitais**.

Tipo de Filtro	Descrição	Aplicação
Passa-baixa	Mantém frequências baixas e elimina as altas	Remoção de ruído de alta frequência
Passa-alta	Mantém frequências altas e elimina as baixas	Detecção de bordas em imagens
Passa-banda	Mantém um intervalo de frequências	Aplicação em rádio e telecomunicações

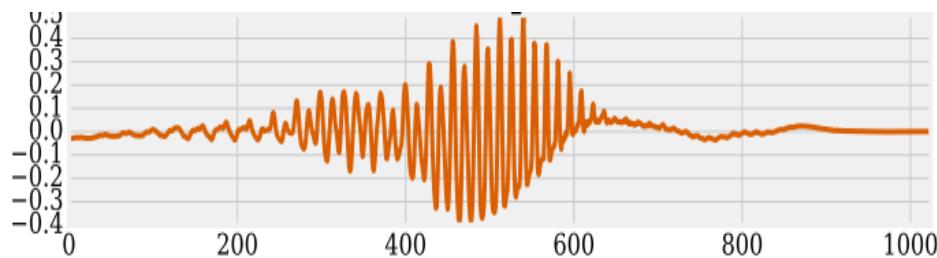
# Segmentação



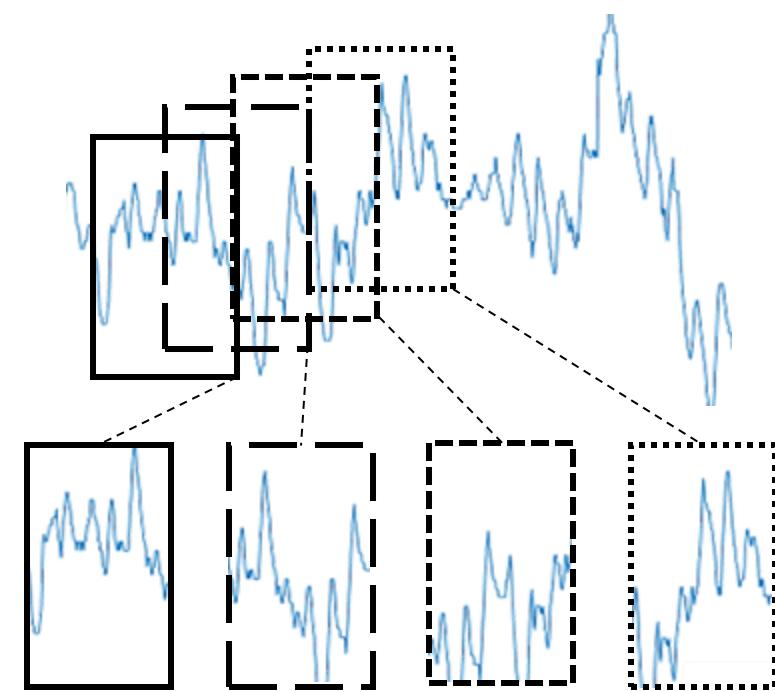
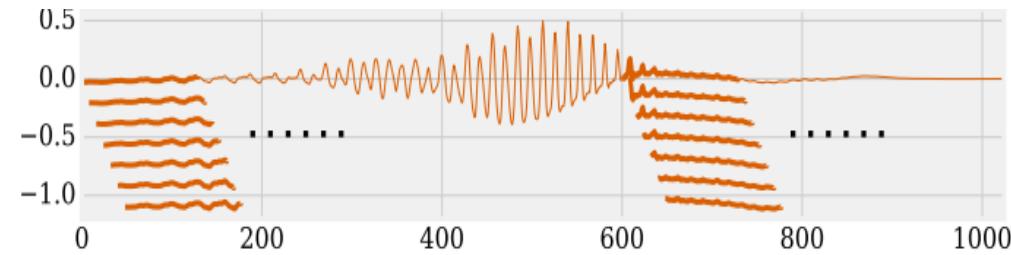
# Segmentação

- Dados adquiridos de acordo com a natureza do problema

Sinal do sensor



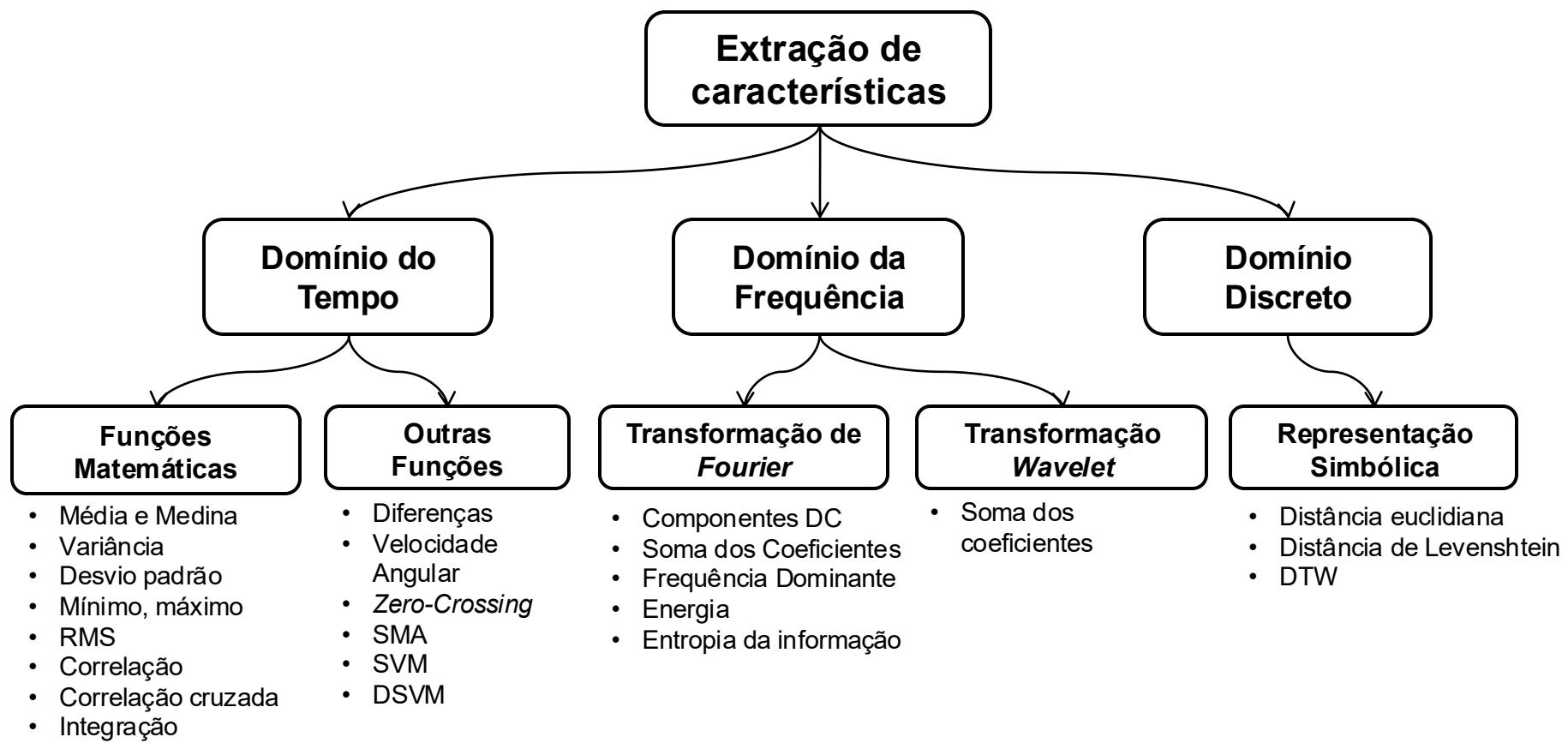
Janela deslizante



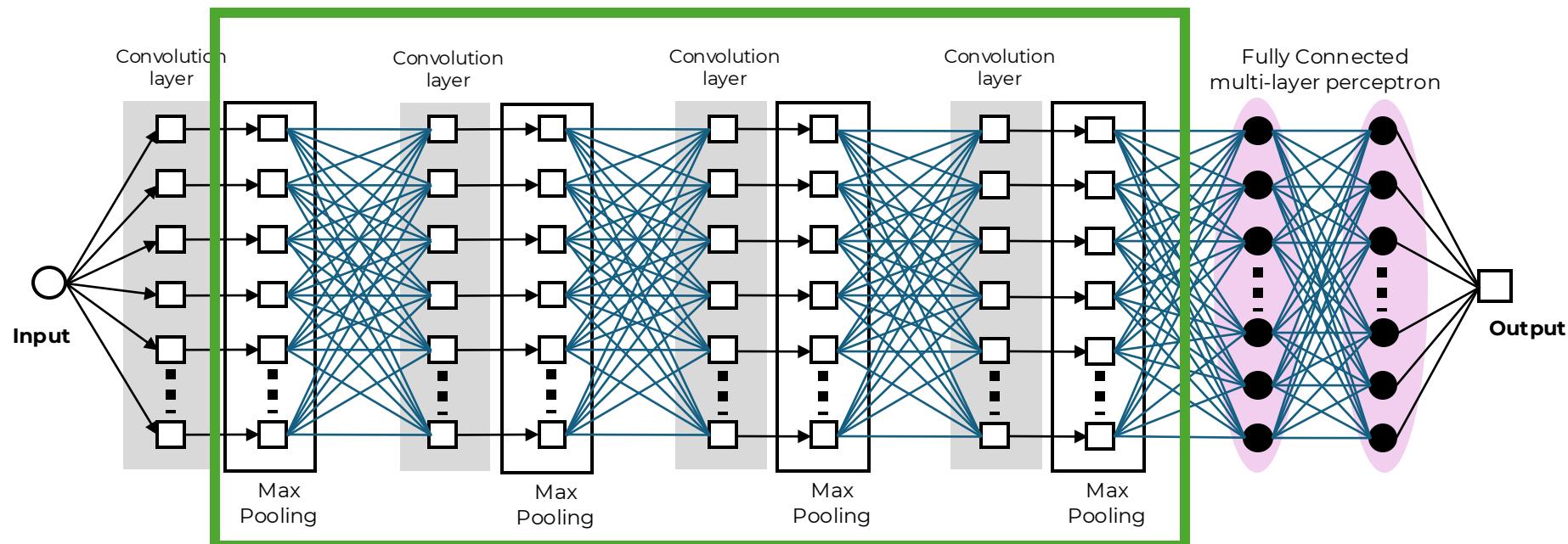
# Extração de Características

			<ul style="list-style-type: none"><li>• Média</li><li>• Máximo</li><li>• Mínimo</li><li>• Desvio Padrão</li></ul>			<ul style="list-style-type: none"><li>• Acertos</li><li>• Erros</li></ul>
Fonte de Dados	Pre-Processamento / Segmentação	Especialista	Extração de Características	Criação de regras	Categorização	Avaliação

# Extração de Características

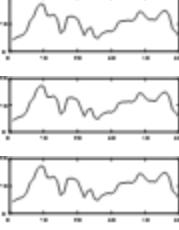
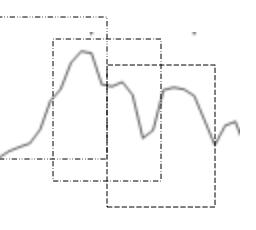
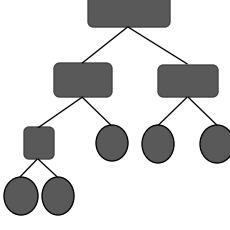


# Extração de Características



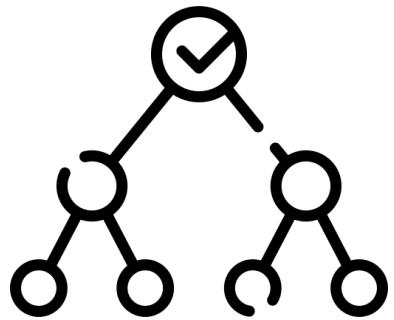
*Basic architecture of a deep neural network model.*

# Classificação

			<ul style="list-style-type: none"><li>• Média</li><li>• Máximo</li><li>• Mínimo</li><li>• Desvio Padrão</li></ul>			<ul style="list-style-type: none"><li>• Acertos</li><li>• Erros</li></ul>
Fonte de Dados	Pre-Processamento / Segmentação	Especialista	Extração de Características	Criação de regras	Categorização	Avaliação

# Classificação

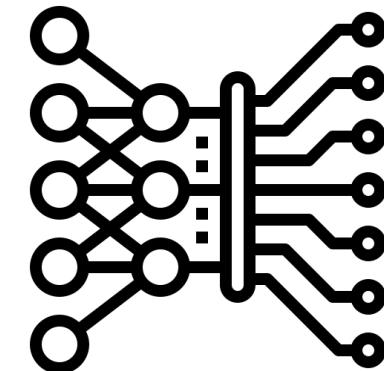
- **XGBoost:** Ensemble of gradient-boosted decision trees.
- **FCN:** Deep network using only convolutions for feature extraction.
- **DeepConvLSTM:** Hybrid CNN-LSTM architecture capturing spatial and temporal patterns.



**XGBoost**



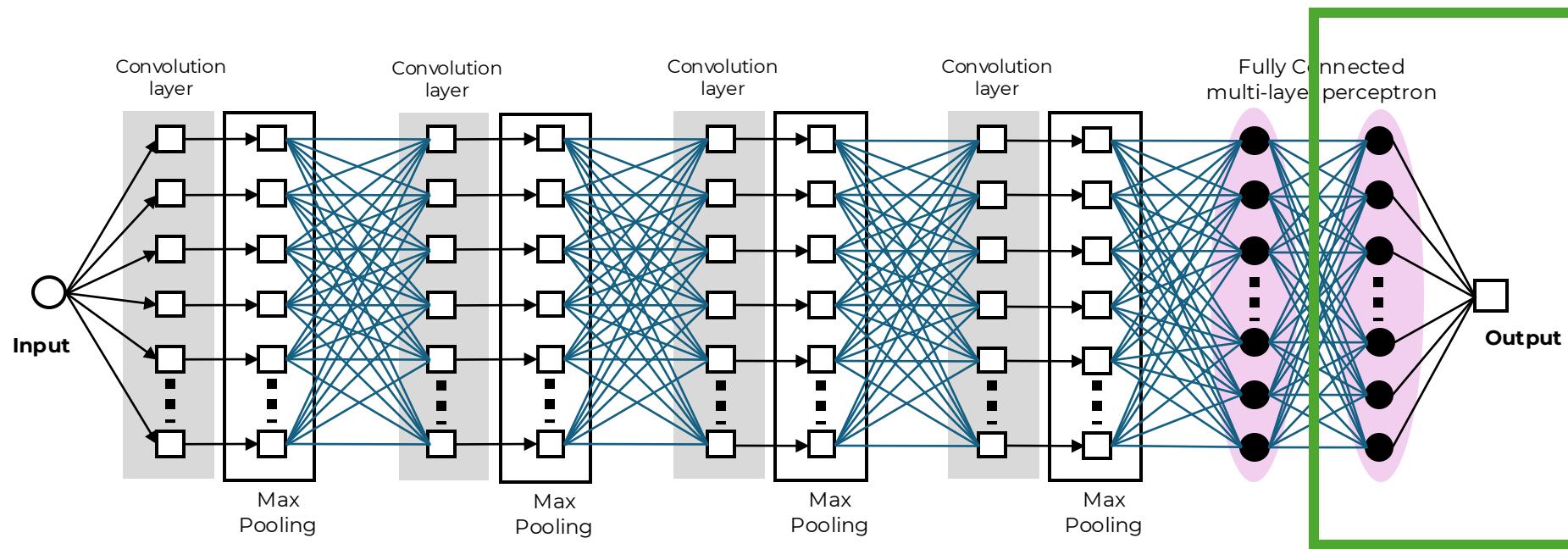
**FCN**



**DeepConvLSTM**

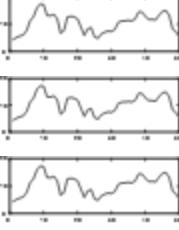
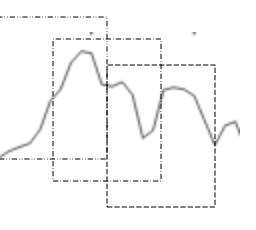
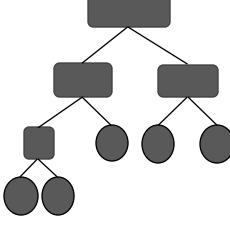
Figure 5-3: Models used in this study.

# Extração de Características



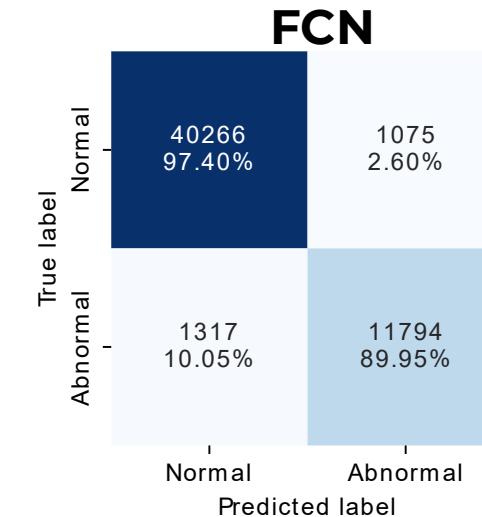
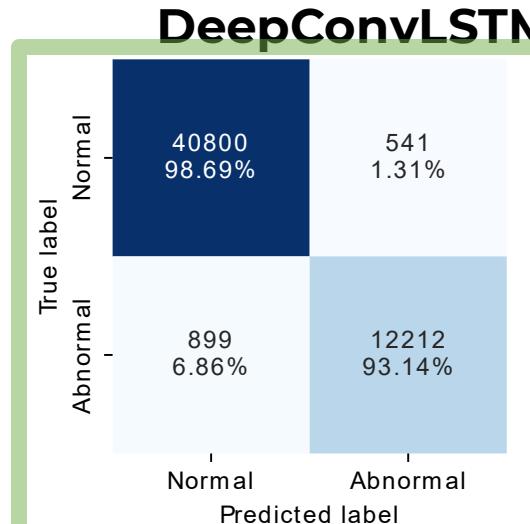
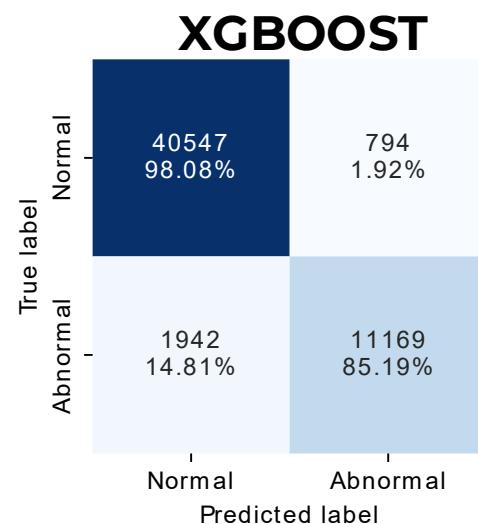
*Basic architecture of a deep neural network model.*

# Avaliação

			<ul style="list-style-type: none"><li>• Média</li><li>• Máximo</li><li>• Mínimo</li><li>• Desvio Padrão</li></ul>			<ul style="list-style-type: none"><li>• Acertos</li><li>• Erros</li></ul>
Fonte de Dados	Pre-Processamento / Segmentação	Especialista	Extração de Características	Criação de regras	Categorização	Avaliação

# Análise usando Matriz de Confusão

**MITBIH  
+SVDB**



**INCART**

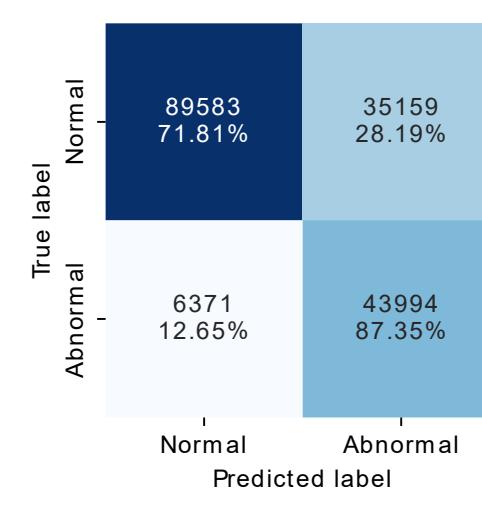
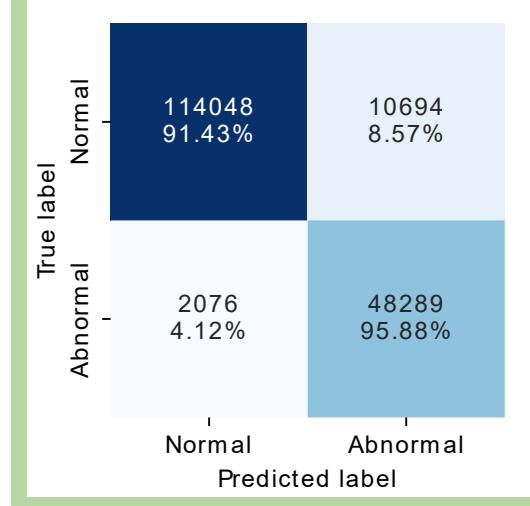
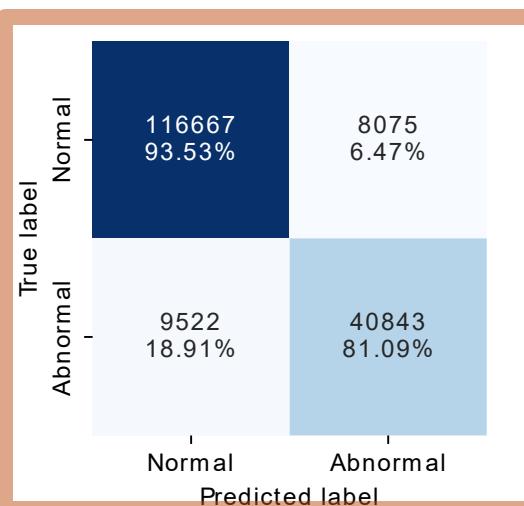
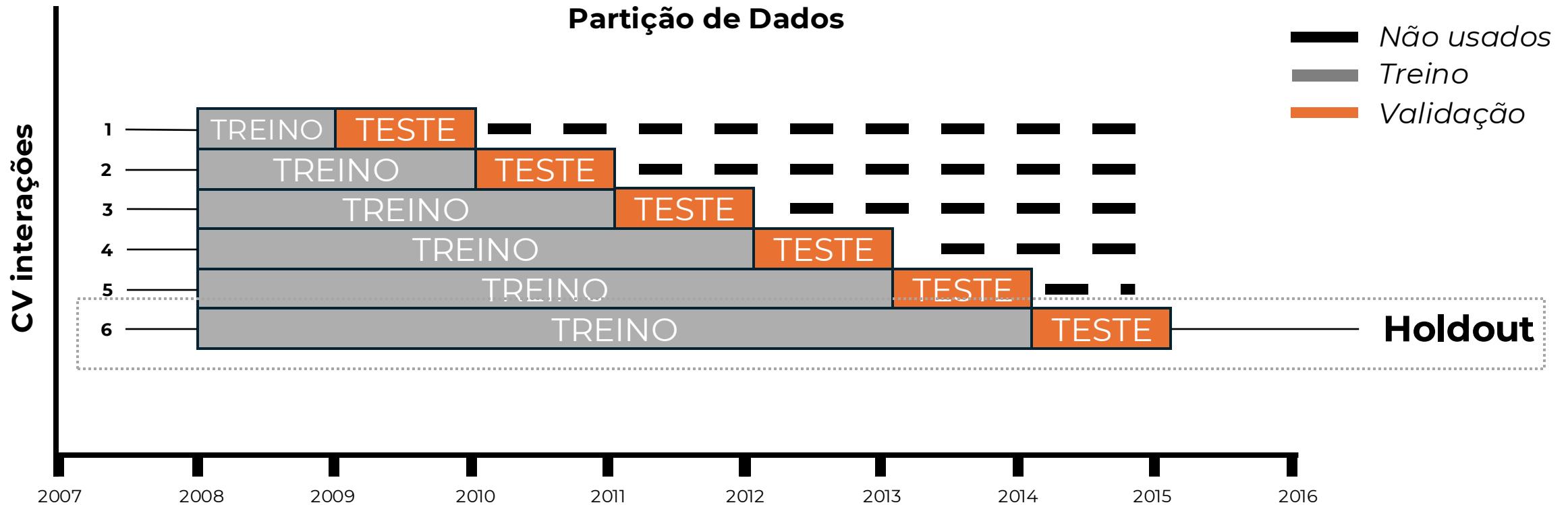
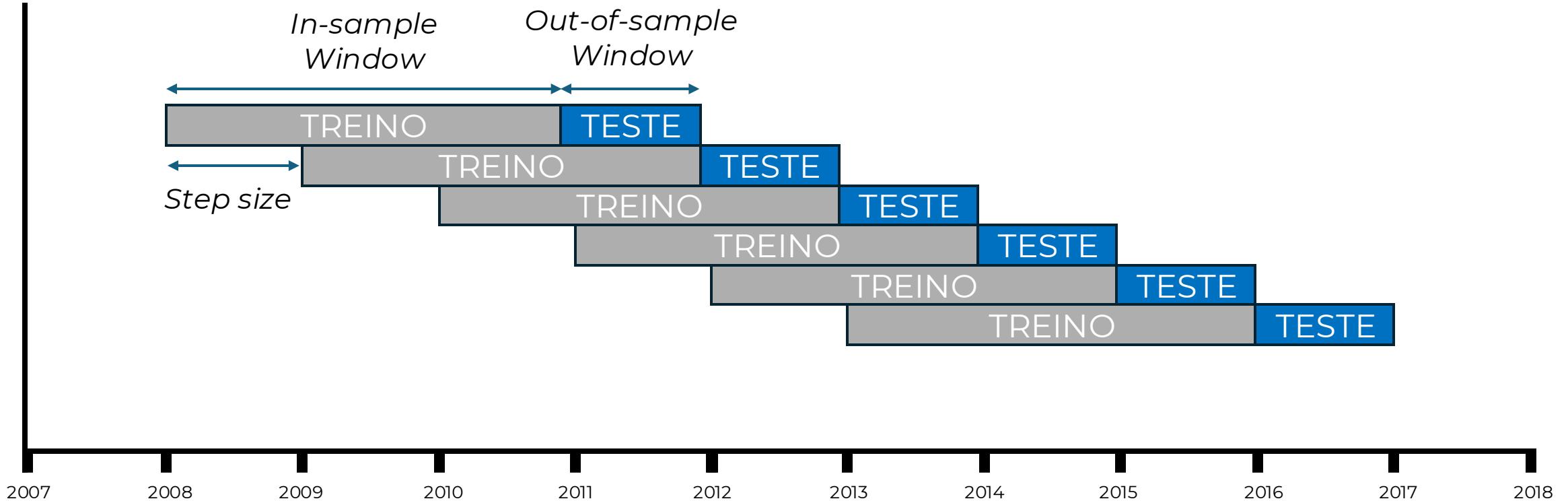


Figure 6-1: XGBoost, DeepConvLSTM and FCN model results on MITBIH+SVDB (training) and INCART (test) datasets for anomaly heartbeats classification.

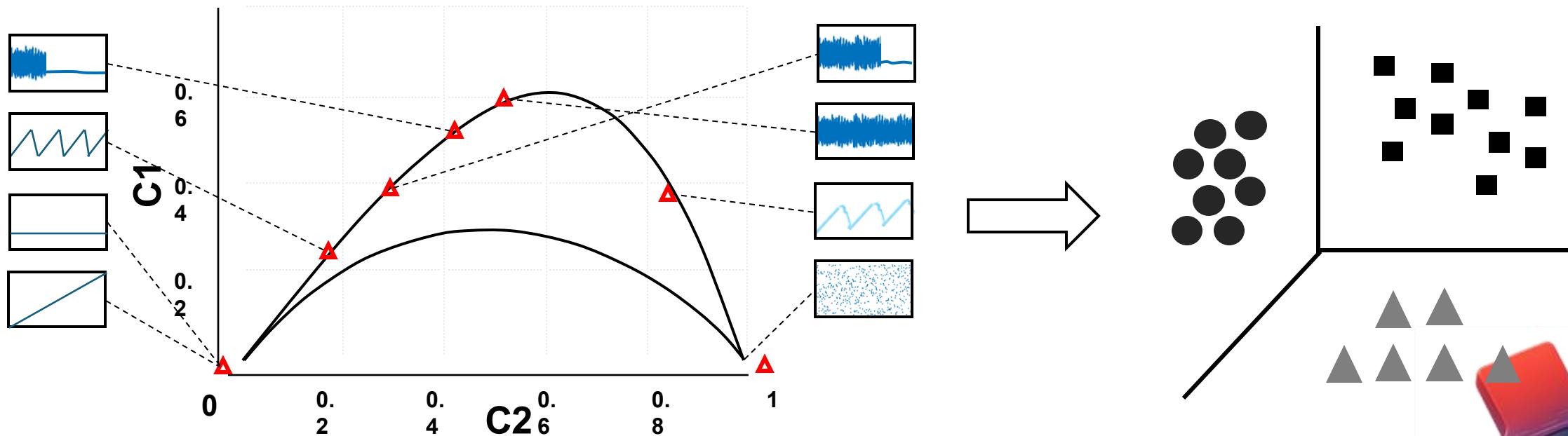
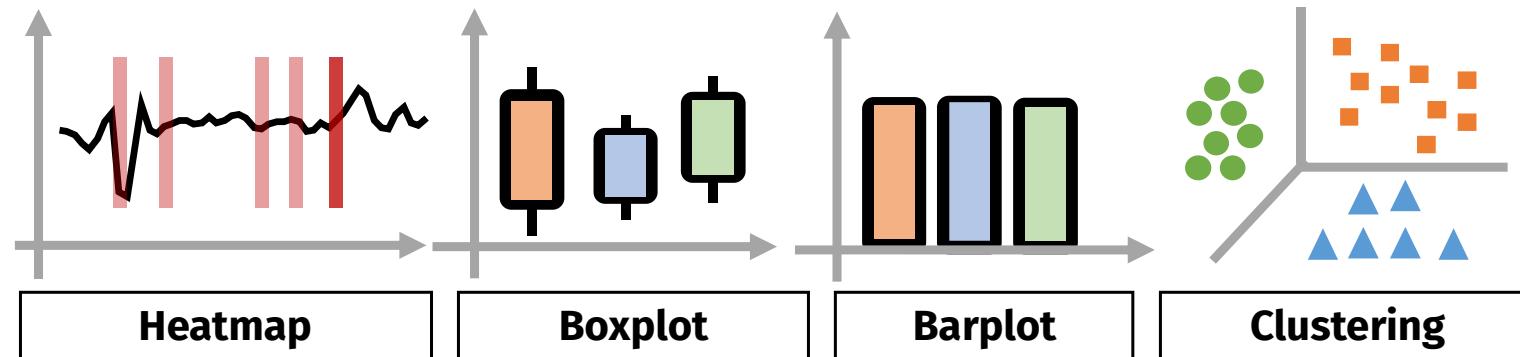
# Avaliação: TS Cross-Validation



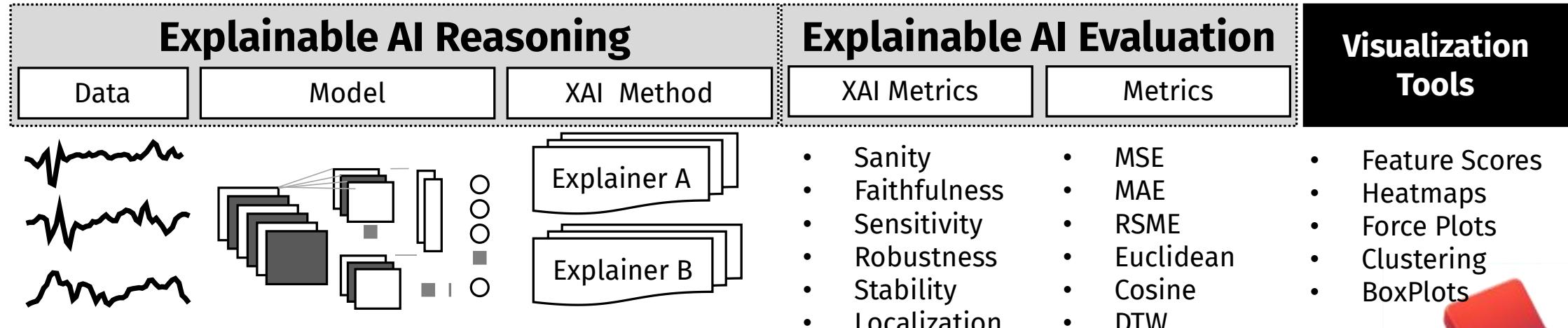
# Avaliação: Walk Forward



# Visualização



# Explicação



# XAI: Métrica de Sanidade

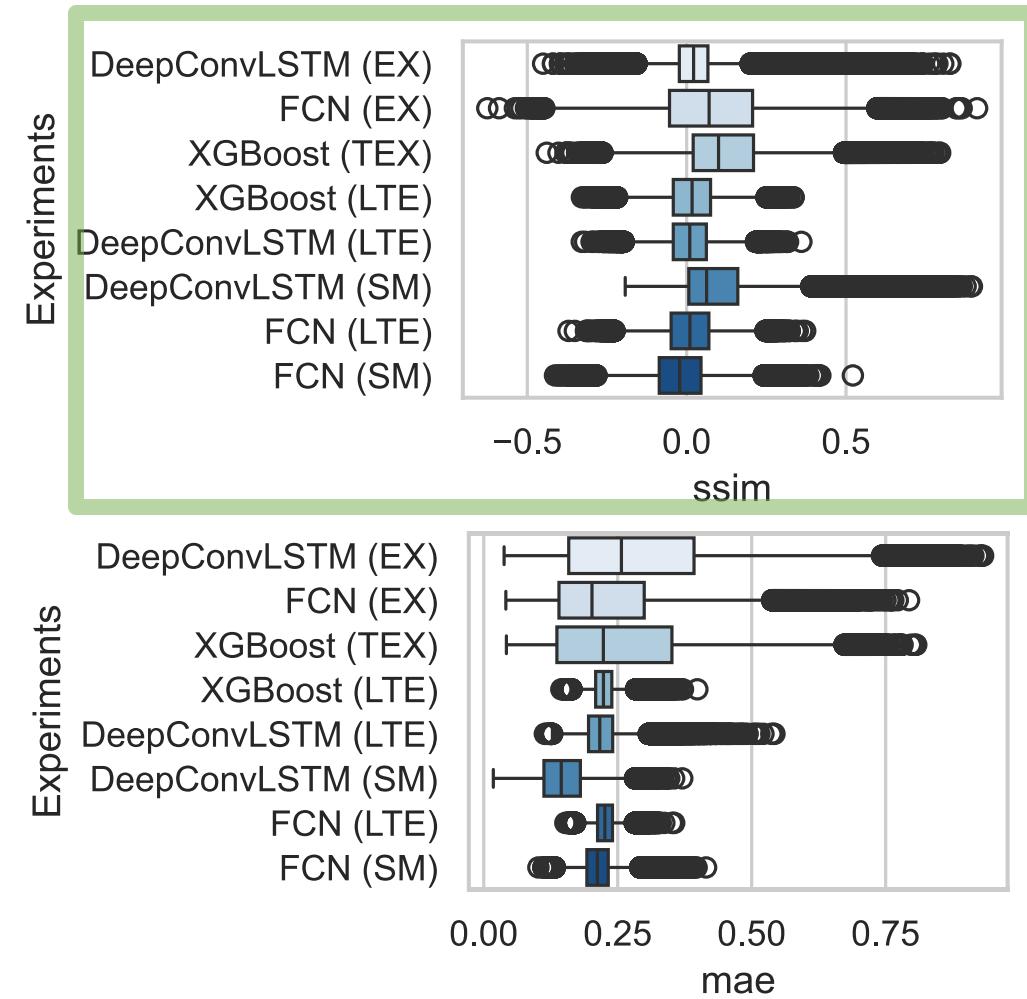
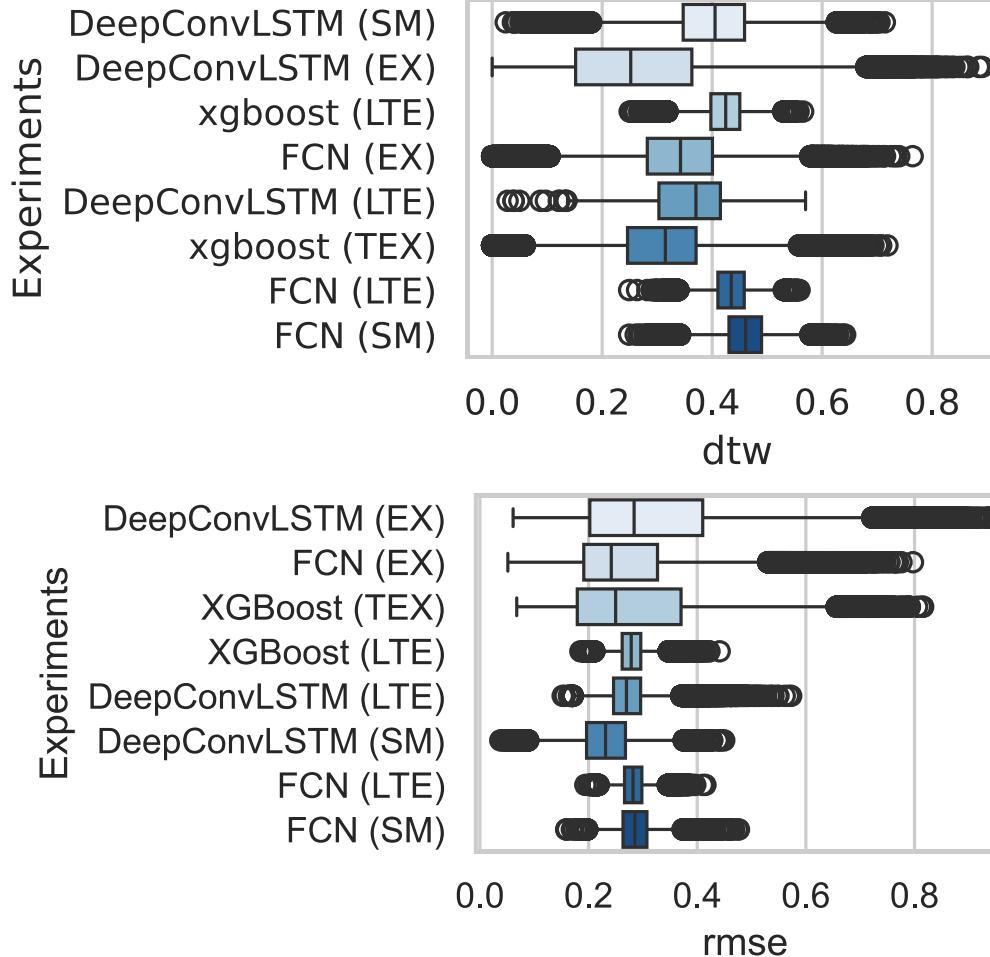


Figure 6-2: Sanity Evaluation of the six similarity metrics across DeepConvLSTM, FCN, and XGBoost model architectures and four XAI methods.

# Os perigos de confiar modelos *black-box*

Pessoas sem ideia sobre IA  
dizendo que ela vai dominar  
o mundo



Pesquisadores em aprendizado de máquina se  
perguntando por que esta rede neural está  
classificando um gato como um cachorro.

# Perguntas e Discussão

# Próximos Passos: Aula Prática

- Demonstrações práticas para o processamento de sinais.
- Hands-On.