



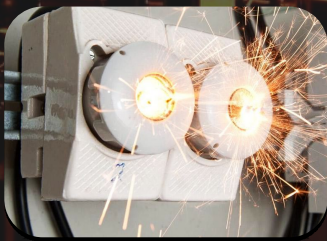
# Detecção de Falhas Elétricas

Renan Massena de Oliveira  
Vinicius Branco Scortegagna

Jun/2022

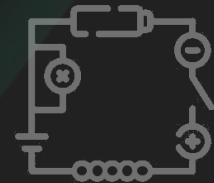
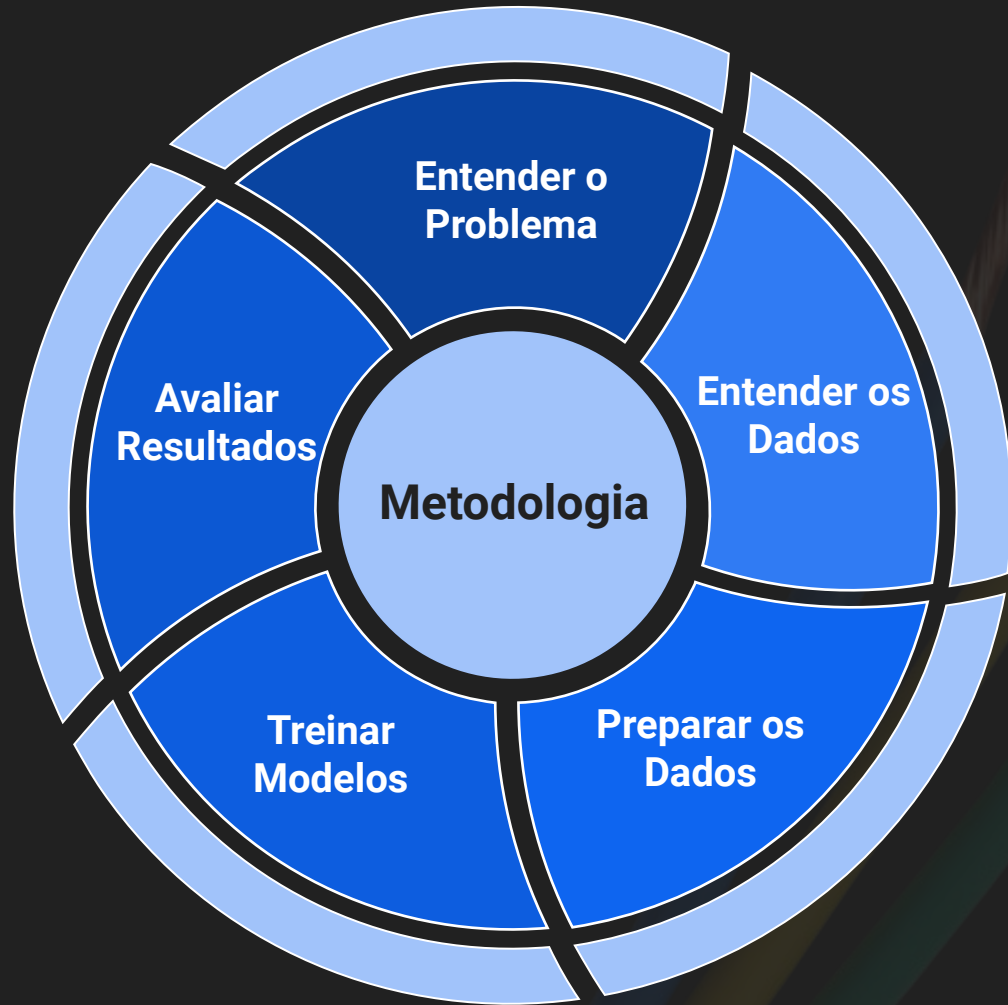
# Problemas

- Explosões
- Incêndios
- Mortes
- Apagões
- Prejuízos



# Pitch

<b>Produto</b>	Inteligência artificial para detecção automática de falhas elétricas
<b>Público</b>	que ajuda indústrias, comércios e residências
<b>Problema</b>	a prevenir acidentes, prejuízos e atrasos no restabelecimento da energia
<b>Solução</b>	utilizando uma rede neural que aprende o padrão das falhas
<b>Diferencial</b>	sendo esta uma solução melhor do que usar apenas um disjuntor geral
<b>Motivo</b>	porque pode ser utilizada em vários pontos da rede elétrica
<b>Objetivo</b>	facilitando a identificação do local em que a falha ocorre.



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- Dados obtidos no Kaggle.
- Circuito elétrico trifásico simulado no MATLAB.
  - Séries temporais de corrente e voltagem.
  - Curtos-circuitos são impostos ao sistema.
- Problema de classificação binária:
  - Conjuntura de valores de corrente e voltagem.
  - Cada instante é independente.
  - Ordem sequencial não importa.



Entender o  
Problema

Entender os  
Dados

Preparar os  
Dados

Treinar  
Modelos

Avaliar  
Resultados

- Estrutura do dataset:
  - 12 mil linhas (instantes).
  - 7 colunas (variáveis).
    - 6 Features:
      - Corrente nas 3 fases:  $I_a$ ,  $I_b$ ,  $I_c$ .
      - Voltagem nas 3 fases:  $V_a$ ,  $V_b$ ,  $V_c$ .
    - 1 Target:
      - 0: sem falha.
      - 1: com falha.



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- Correntes: evolução temporal.



Falhas identificadas pelas linhas vermelhas

**Entender o  
Problema**

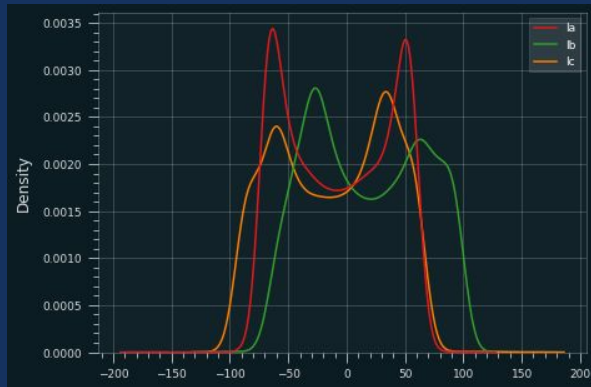
**Entender os  
Dados**

**Preparar os  
Dados**

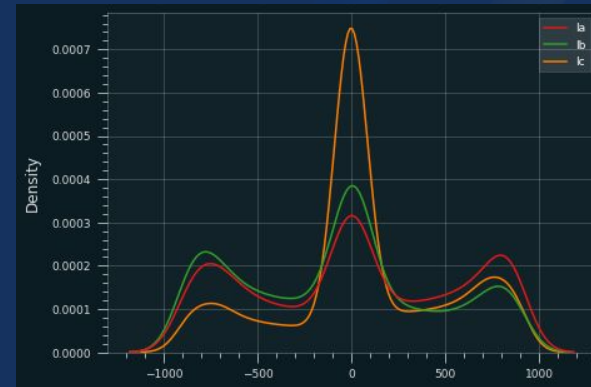
**Treinar  
Modelos**

**Avaliar  
Resultados**

- Correntes: distribuição de valores.



Sem Falha



Com Falha



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- **Voltagem: evolução temporal.**



Falhas identificadas pelas linhas vermelhas

**Entender o  
Problema**

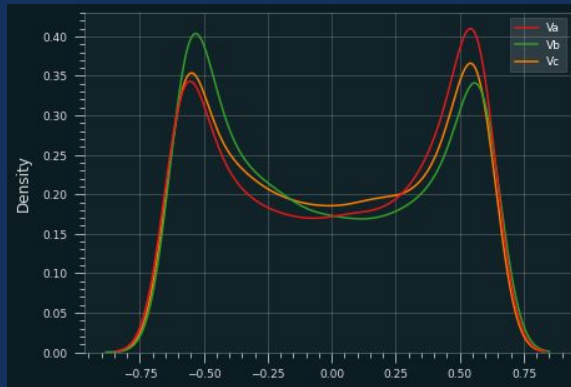
**Entender os  
Dados**

**Preparar os  
Dados**

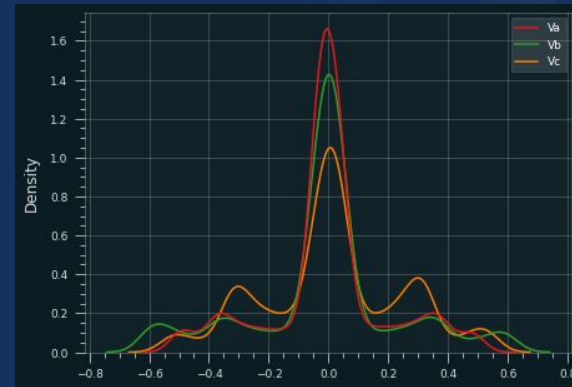
**Treinar  
Modelos**

**Avaliar  
Resultados**

- **Voltagem: distribuição de valores.**



Sem Falha



Com Falha

**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- Tratamento de dados não foi necessário.
- Dados separados em treino e teste:
  - 80/20
  - 70/30
- Dados normalizados:
  - MinMax Scaler:
    - $X_{mm} = (X - X_{min}) / (X_{max} - X_{min})$
  - Standard Scaler:
    - $X_{ss} = (X - X_{avg}) / (X_{std})$



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- Clusterização:
  - K-means:
    - 2 e 3 grupos.
  - DBSCAN:
    - Raio: 0.1.
    - Amostras: 50.



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- Classificação:
  - Rede neural:
    - Camada de entrada: 6 neurônios, função RELU.
    - Camada oculta: 6 neurônios, função RELU.
    - Camada de saída: 1 neurônio, função SIGMÓIDE.
    - Otimizador: Adam.
    - Função de custo: entropia cruzada binária.
    - Métrica: acurácia.
    - Tamanho de lote: 32.



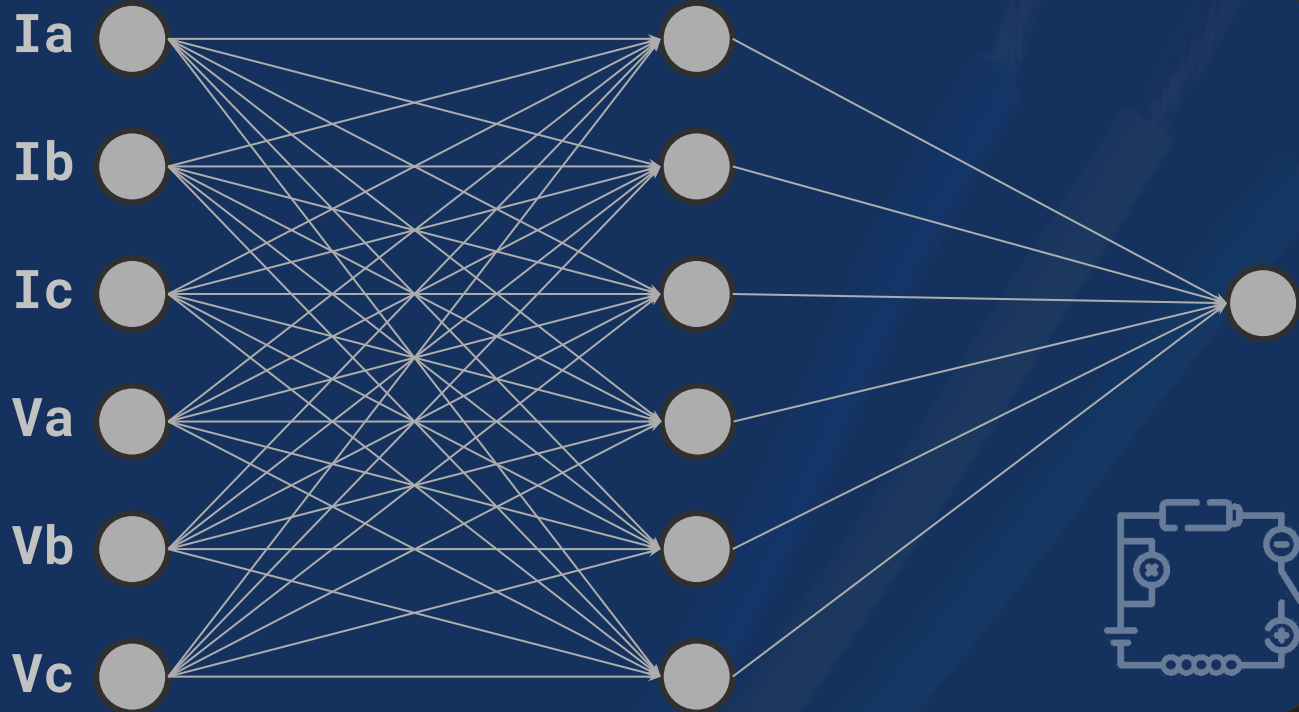
**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- K-means: não consegue clusterizar adequadamente.



Resultado da clusterização em linhas vermelhas  
(0: sem falha, 1: com falha)

**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- DBSCAN: resultado satisfatório.



Resultado da clusterização em linhas vermelhas  
(0: sem falha, 1: com falha)



Entender o  
Problema

Entender os  
Dados

Preparar os  
Dados

Treinar  
Modelos

Avaliar  
Resultados

- DBSCAN – Train Report

	precision	recall	f1-score	support
0	1.00000	0.97156	0.98558	6505
1	0.96744	1.00000	0.98345	5496
accuracy			0.98458	12001
macro avg	0.98372	0.98578	0.98451	12001
weighted avg	0.98509	0.98458	0.98460	12001

Entender o  
Problema

Entender os  
Dados

Preparar os  
Dados

Treinar  
Modelos

Avaliar  
Resultados

- DBSCAN – Matriz de confusão



**Entender o  
Problema**

**Entender os  
Dados**

**Preparar os  
Dados**

**Treinar  
Modelos**

**Avaliar  
Resultados**

- Rede Neural:
  - Resultado satisfatório com validação cruzada.
  - Épocas: 150.
  - Treino/Teste: 80/20.
  - K-folds:
    - 4 splits
    - Acurácia média: 99.30% (+/- 0.15%)



Entender o  
Problema

Entender os  
Dados

Preparar os  
Dados

Treinar  
Modelos

Avaliar  
Resultados

- Rede Neural – Test Report (split 80/20)

	precision	recall	f1-score	support
0	0.99387	0.99769	0.99578	1301
1	0.99726	0.99273	0.99499	1100
accuracy			0.99542	2401
macro avg	0.99557	0.99521	0.99538	2401
weighted avg	0.99543	0.99542	0.99542	2401

Entender o  
Problema

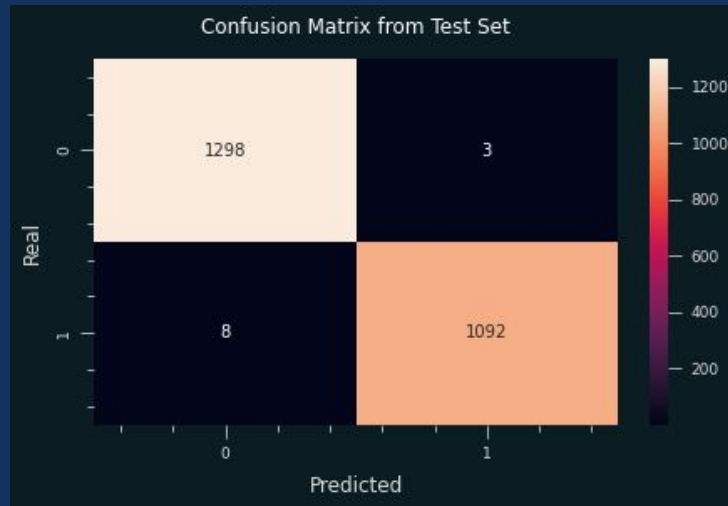
Entender os  
Dados

Preparar os  
Dados

Treinar  
Modelos

Avaliar  
Resultados

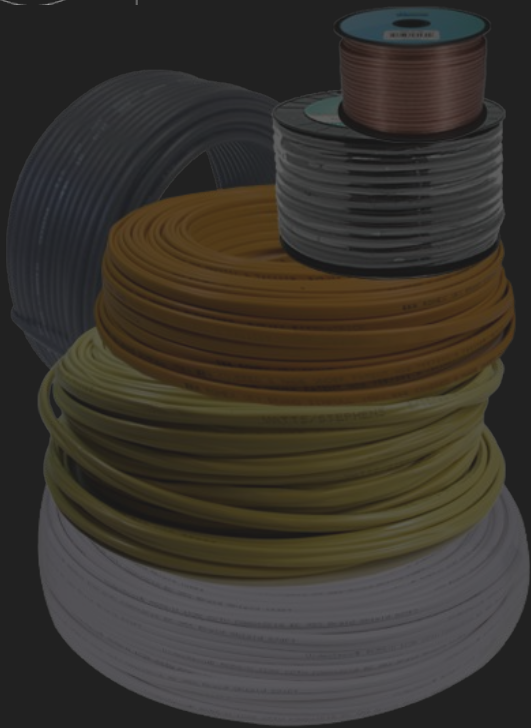
- Rede Neural – Matriz de confusão





Instituto Infnet

MIT Data Science, Data Analytics & Machine Learning 2022-1



# OBRIGADO!



<https://www.linkedin.com/in/renan-massena>  
<https://www.linkedin.com/in/vinicius-br-sc>