UFRN - UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

CT - CENTRO DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO

CURSO DE GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

ESTUDO DE CASO - AI4I 2020 PREDICTIVE MAINTENANCE DATASET

Componente Eletiva: IMD3002 - Aprendizado De Máquina Supervisionado

Docentes: Dr. Daniel Sabino Amorim De Araujo

Dr. Renan Cipriano Moioli

Discente: 20210025010 Francisco De Lima Silva Filho

SUMÁRIO

‘

1. INTRODUÇÃO

A crescente digitalização dos processos industriais e a incorporação de sensores nos equipamentos têm impulsionado a adoção de soluções baseadas em aprendizado de máquina (machine learning) para otimizar a manutenção de ativos. Nesse contexto, surge a manutenção preditiva, uma abordagem que visa antecipar falhas em máquinas e equipamentos com base na análise de dados operacionais, reduzindo paradas não planejadas e maximizando a eficiência produtiva.

Este projeto tem como objetivo aplicar técnicas de aprendizado de máquina supervisionado para prever a ocorrência de falhas em máquinas industriais, bem como classificar o tipo de falha quando ela ocorre. Para isso, utilizou-se o **AI4I 2020 Predictive Maintenance Dataset**, um conjunto de dados sintético, porém realista, que simula um ambiente industrial com variáveis monitoradas por sensores e diferentes tipos de falhas operacionais.

A motivação por trás deste projeto está em alinhar os conhecimentos de modelagem supervisionada a um problema de relevância prática no setor produtivo, promovendo a interdisciplinaridade entre engenharia de produção e ciência de dados. O trabalho contempla desde a exploração dos dados (EDA), passando pela construção de modelos de classificação binária (falha vs. não falha) e multiclasse (tipo de falha), até a avaliação dos modelos treinados e sua aplicabilidade em cenários industriais.

2. ENTENDIMENTO E TRATAMENTO DOS DADOS

2.1 Descrição da base

O projeto utilizou o conjunto de dados AI4I 2020 Predictive Maintenance Dataset, disponibilizado publicamente pela UCI Machine Learning Repository. Embora sintético, o dataset foi elaborado com o objetivo de **simular condições reais de operação industrial**, incluindo medições contínuas de sensores, registros operacionais e anotações

‘

de falhas. Cada linha representa um instante no tempo para uma determinada máquina em operação.

A base contém inicialmente 10.000 registros distribuídos em 14 colunas, sendo:

- 2 identificadores
    - *UDI* (Unique Identifier)
    - *Product ID*
- 6 variáveis numéricas operacionais
    - *Temperatura do ar (K)*
    - Temperatura do processo (K)
    - Torque (Nm)
    - Velocidade rotacional (RPM)
    - Desgaste da ferramenta (Min)
- 1 variável categórica
    - Tipo (referente a qualidade do tipo da máquina)
- 2 variáveis-alvo
    - Target: indica ocorrência ou não de falha (0 = sem falha, 1 = com falha)
    - Failure Type: especifica o tipo de falha, com cinco categorias diferentes (além da classe “No Failure”)

‘

2.2 Tratamento inicial

Durante a fase de tratamento inicial dos dados, foram aplicadas as seguintes ações:

- Remoção de colunas não preditivas: *UDI* e *Product ID* foram descartadas por não contribuírem com o aprendizado do modelo.
- Verificação de qualidade dos dados:
    - Nenhum valor ausente foi identificado nas colunas
    - Não foram detectados registros duplicados
- Otimização de memória: aplicou-se *downcast* em variáveis numéricas e conversão de colunas categóricas para o tipo *category*, reduzindo significativamente o uso de memória.

2.3 Características do conjunto

Em primeiro momento, algumas características importantes do conjunto de dados puderam ser identificadas:

- A variável Type assume três categorias relacionadas à qualidade do produto: L (low), M (medium) e H (high), com distribuição não balanceada.
- As variáveis operacionais apresentam diferentes escalas e distribuições, exigindo normalização.
- A variável-alvo Target é altamente desbalanceada: apenas 3,39% dos registros indicam ocorrência de falha.

‘

- A variável Failure Type detalha os tipos de falhas possíveis: Tool Wear Failure, Heat Dissipation Failure, Power Failure, Overstrain Failure, Random Failures, além de No Failure.

3. ANÁLISE EXPLORATÓRIA DE DADOS

A análise exploratória teve como objetivo identificar padrões relevantes, relações entre variáveis e potenciais fatores preditivos de falha. Para isso, foram utilizadas visualizações como histogramas, violin plots, boxplots e matrizes de correlação. A seguir, são destacados os principais resultados da EDA.

3.1 Distribuição da variável alvo

A variável Target revelou uma **forte assimetria**:

- 96,61% dos registros correspondem a situações normais (sem falha)
- Apenas 3,39% registram falhas em operação

‘

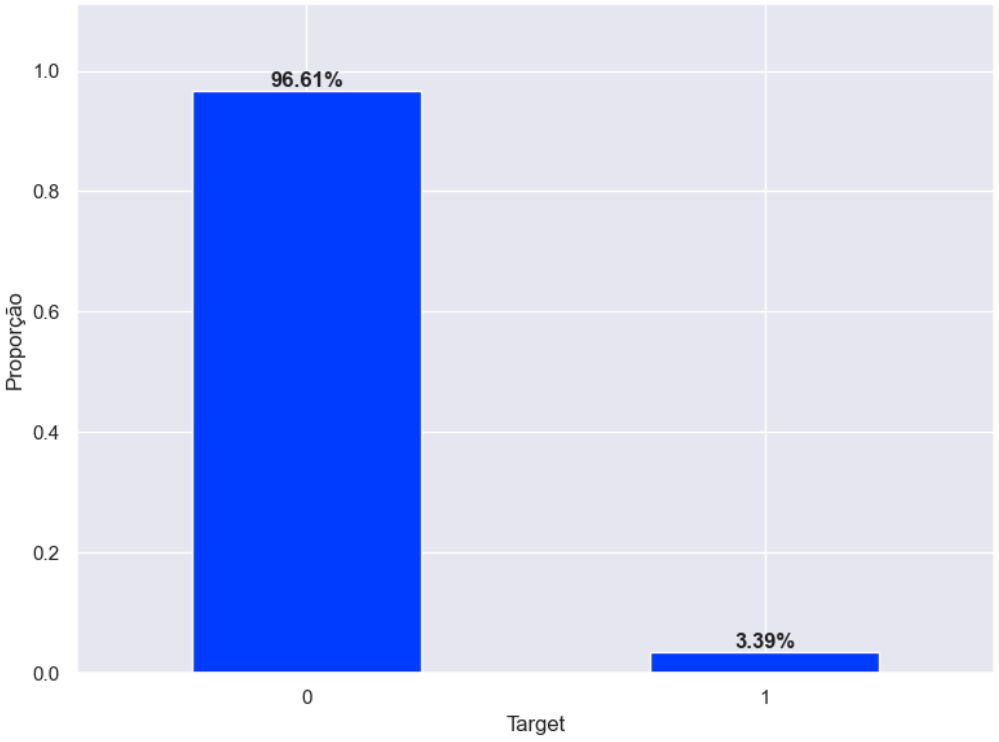


Figura 1 - Balanceamento entre classes

Esse desbalanceamento impõe desafios à modelagem supervisionada, especialmente no recall da classe minoritária.

A variável *Failure Type*, presente apenas quando há falha, foi distribuída entre cinco tipos distintos, observados na Figura 2:

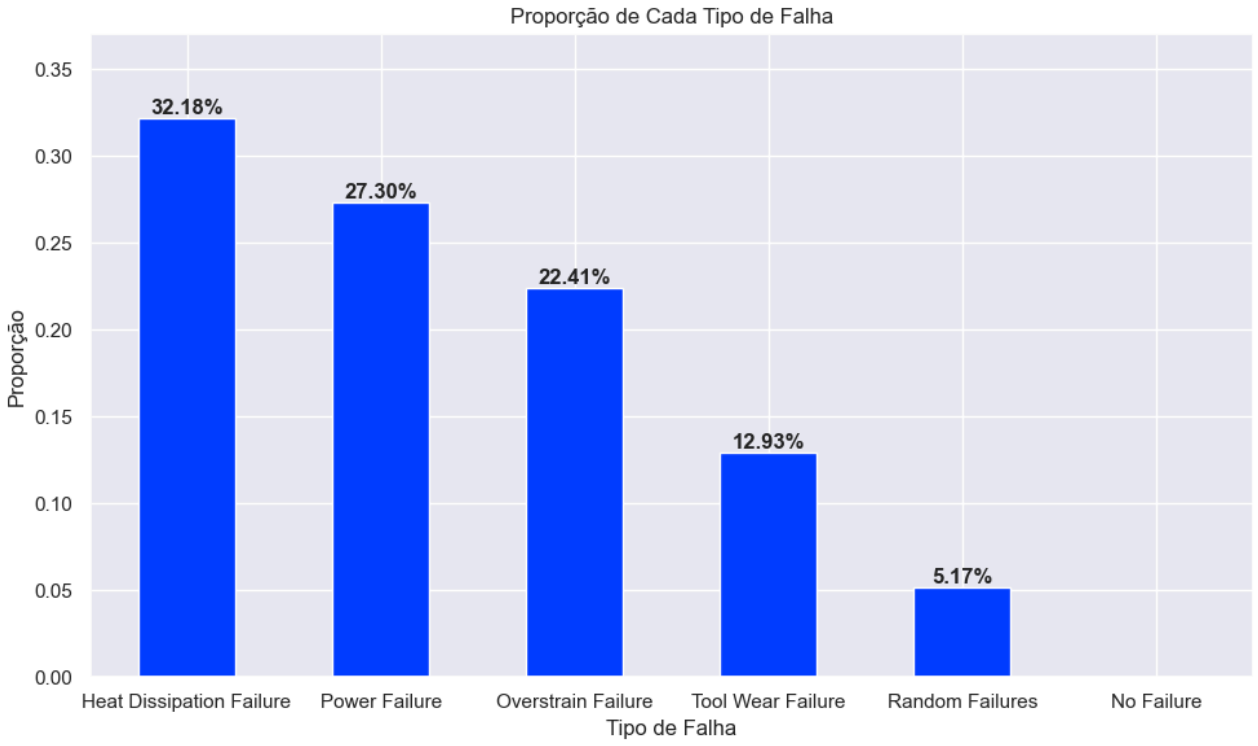


Figura 2 - Proporção de cada tipo de falha

3.2 Análise de variáveis numéricas por classe de falha

No boxplot da Figura 3, são apresentados os principais comportamentos identificados nas variáveis numéricas em relação à ocorrência de falhas (Target = 1):

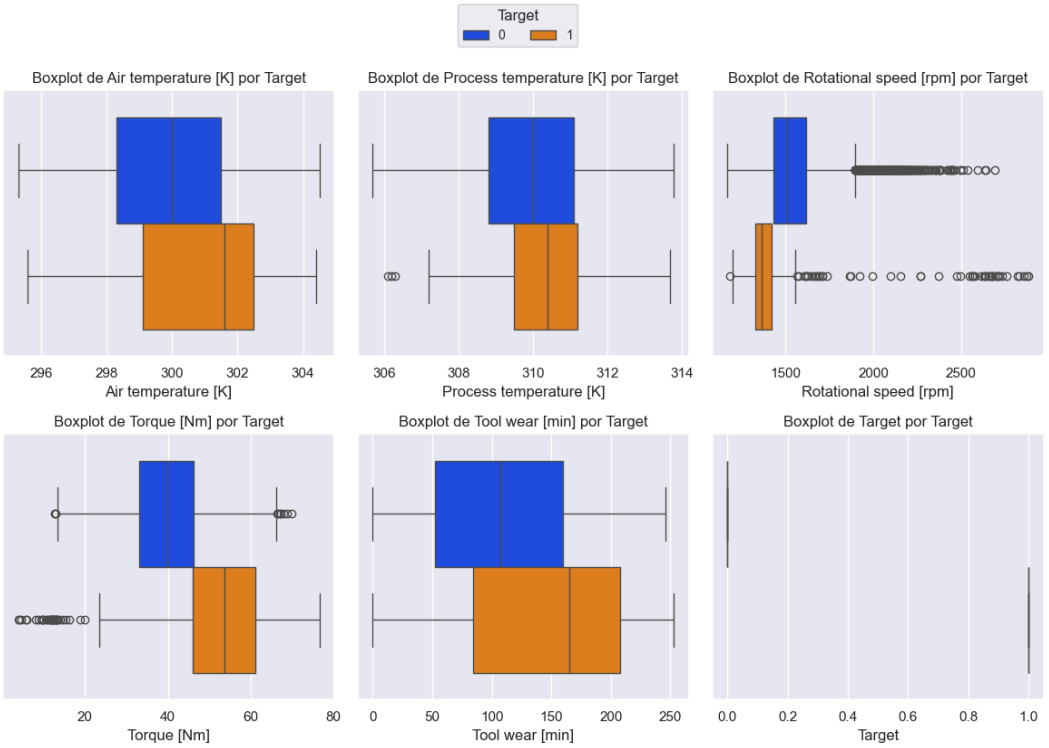


Figura 3 - Relação entre variáveis preditoras e alvo (binário)

- **Air Temperature [K]**: Falhas ocorrem em temperaturas ambiente ligeiramente superiores à média (~300 K).
- **Process Temperature [K]**: Apresenta padrão semelhante ao da temperatura ambiente, com ligeiro deslocamento para cima em casos de falha.
- **Rotational Speed [rpm]**: As falhas ocorrem com maior frequência em valores abaixo da média, mas também aparecem em extremos altos — sugerindo uma relação não linear.
- **Torque [Nm]**: Registros com falha tendem a apresentar valores mais altos de torque, indicando condições operacionais mais exigentes.

- **Tool Wear [min]**: É a variável com maior separação entre as classes. A maioria das falhas ocorre com altos níveis de desgaste.

3.3 Análise por tipo de falha (multiclasse)

A análise por Failure Type (Figura 4) permitiu identificar padrões específicos para cada tipo de falha:

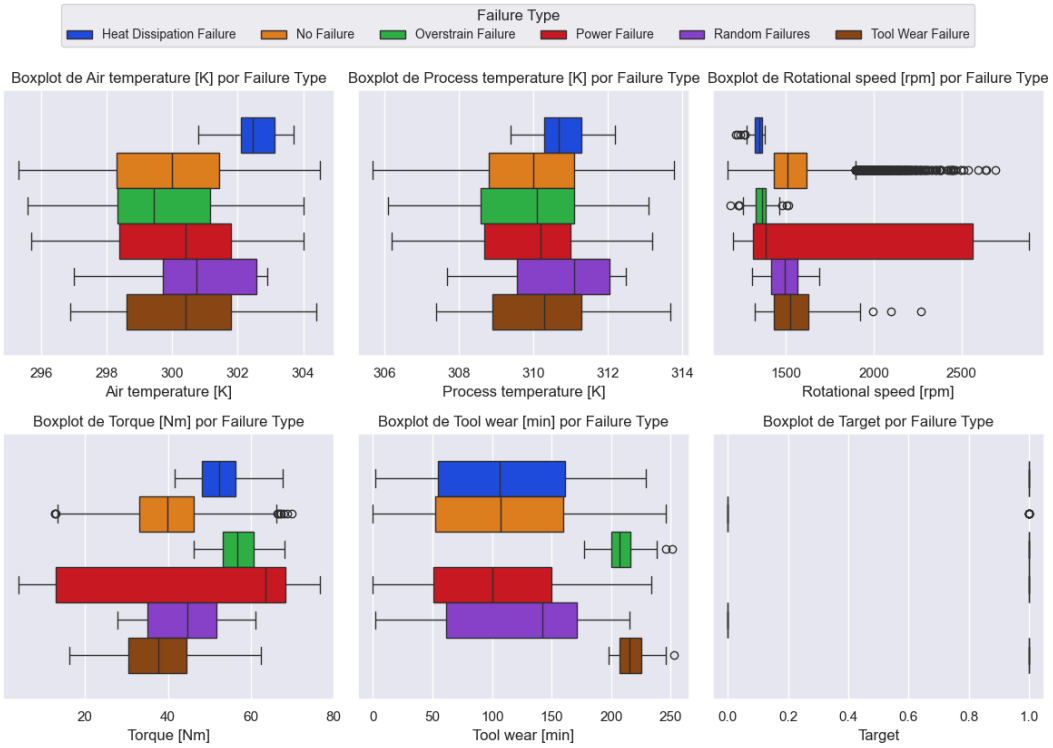


Figura 4 - Relação entre variáveis preditoras e alvo (multiclasse)

- **Heat Dissipation Failure**: Associada a temperaturas elevadas do ar e do processo.
- **Power Failure**: Ocorre em altos níveis de torque e rotação.

- **Tool Wear Failure**: Relacionada a valores altos de desgaste.
- **Overstrain Failure**: Distribuição ampla, com tendência a ocorrer sob torque e desgaste altos.
- **Random Failures**: Sem padrão definido, aparecendo de forma dispersa.

3.4 Correlação entre variáveis

A matriz de correlação (Figura 5) revelou:

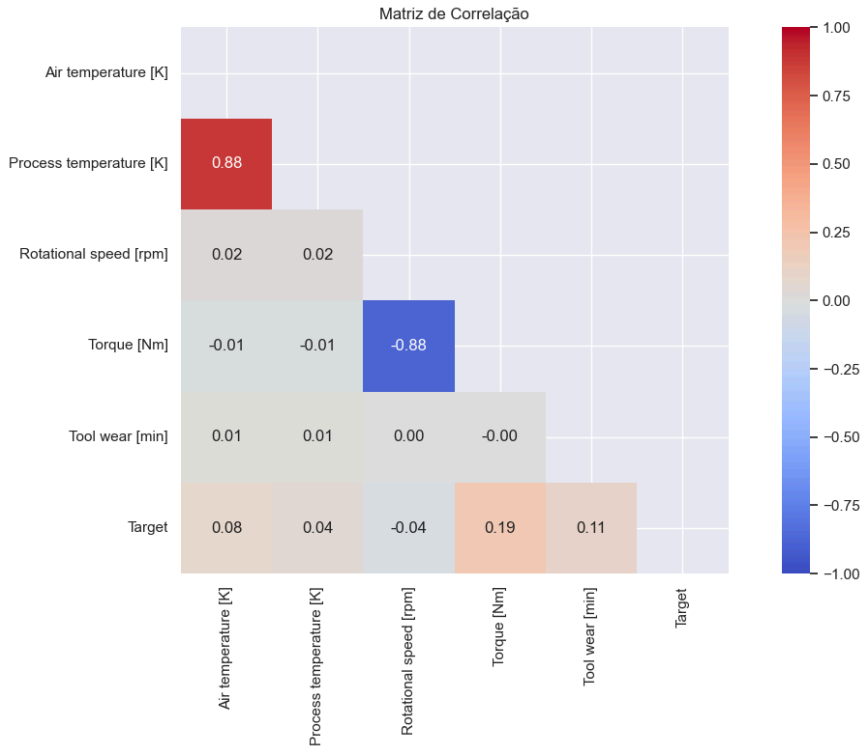


Figura 5 - Matriz de correlação

- **Correlação forte positiva** entre Air Temperature e Process Temperature (r ≈ 0.88)

- **Correlação forte negativa** entre Torque e Rotational Speed (r ≈ -0.88)
- **Baixa correlação** de Tool Wear com as demais variáveis, reforçando sua importância individual como preditora

4. MODELO DE CLASSIFICAÇÃO

4.1 Estratégia adotada

A modelagem supervisionada foi conduzida em duas etapas complementares:

1. **Classificação binária**: prever a ocorrência de falha (Target: 0 = sem falha, 1 = com falha)
2. **Classificação multiclasse**: determinar o tipo de falha quando a falha ocorre (Failure Type)

Essa abordagem reflete um cenário realista em sistemas de monitoramento industrial, nos quais a primeira decisão a ser tomada é se há falha e, posteriormente, qual é sua natureza. Cada etapa foi tratada com pipelines independentes, mas com técnicas compartilhadas.

4.2 Pré-processamento e Balanceamento

O pipeline de pré-processamento foi estruturado da seguinte forma:

- **Imputação de valores ausentes (Simple imputer)**:
  - Numéricas: média
  - Categóricas: moda

‘

- **Normalização de variáveis numéricas**: PowerTransformer para aproximação da distribuição normal
- **Codificação de variáveis categóricas**: OrdinalEncoder

Além disso, foi utilizado o método **SMOTEENN**, que combina:

- SMOTE (Synthetic Minority Over-sampling Technique) para gerar exemplos sintéticos da classe minoritária
- ENN (Edited Nearest Neighbours) para remover ruído da classe majoritária

Essa etapa foi essencial para lidar com o forte desbalanceamento presente no problema, especialmente na classificação binária (onde apenas ~3,4% dos registros indicam falha).

4.3 Modelos Utilizados

Quatro algoritmos foram avaliados em ambas as tarefas:

- RandomForestClassifier
- LogisticRegression
- GaussianNB
- KNeighborsClassifier

A seleção final de modelos foi feita via GridSearchCV, utilizando validação cruzada estratificada (5-fold). As métricas de avaliação variaram conforme a tarefa:

- Classificação binária: recall_weighted (prioridade em detectar falhas)

‘

- Classificação multiclasse: f1_weighted (balanço entre precisão e recall para múltiplas classes)

4.4 Classificação binária: Falha vs. Não Falha

Para a tarefa de classificação binária, o melhor modelo ajustado foi o **RandomForestClassifier**. A figura 6 mostra suas principais métricas no conjunto de teste:

```
=== Classificação Binária (Falha vs Não Falha) ===

Modelo: RandomForest
              precision    recall  f1-score   support

           0       0.99      0.96      0.97      1932
           1       0.40      0.82      0.54        68

    accuracy                           0.95      2000
   macro avg       0.70      0.89      0.76      2000
weighted avg       0.97      0.95      0.96      2000
```

Figura 6 - Classification Report do modelo binário

Além disso, podemos avaliar o modelo por outros parâmetros que obtiveram desempenho satisfatório:

- **Matriz de confusão** (Figura 7): indicou baixa taxa de falsos negativos (falhas não detectadas)
- **AUC-ROC** (Figura 8): 0.97 — excelente capacidade de separação entre as classes
- **Curva Precision-Recall** (Figura 9): indicou bom equilíbrio entre sensibilidade e precisão, mesmo com desbalanceamento

‘

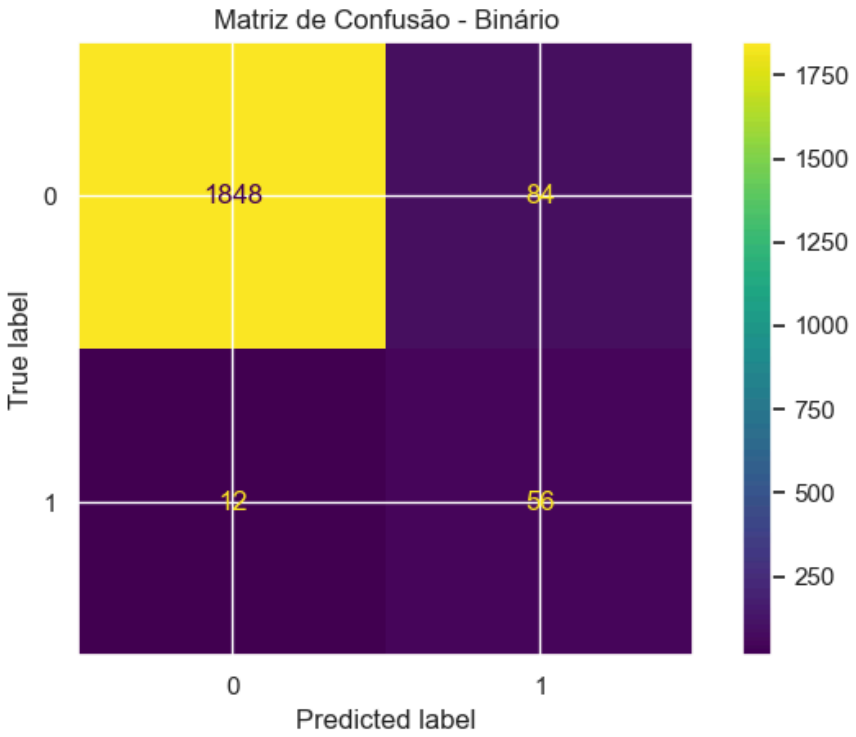
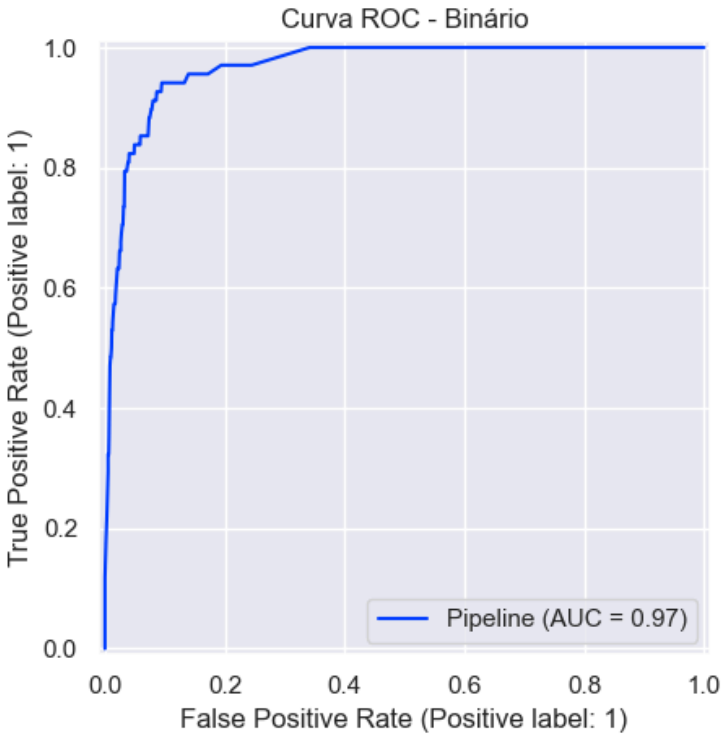


Figura 7 - Matriz de confusão do modelo binário



‘

Figura 8 - Curva ROC do modelo binário

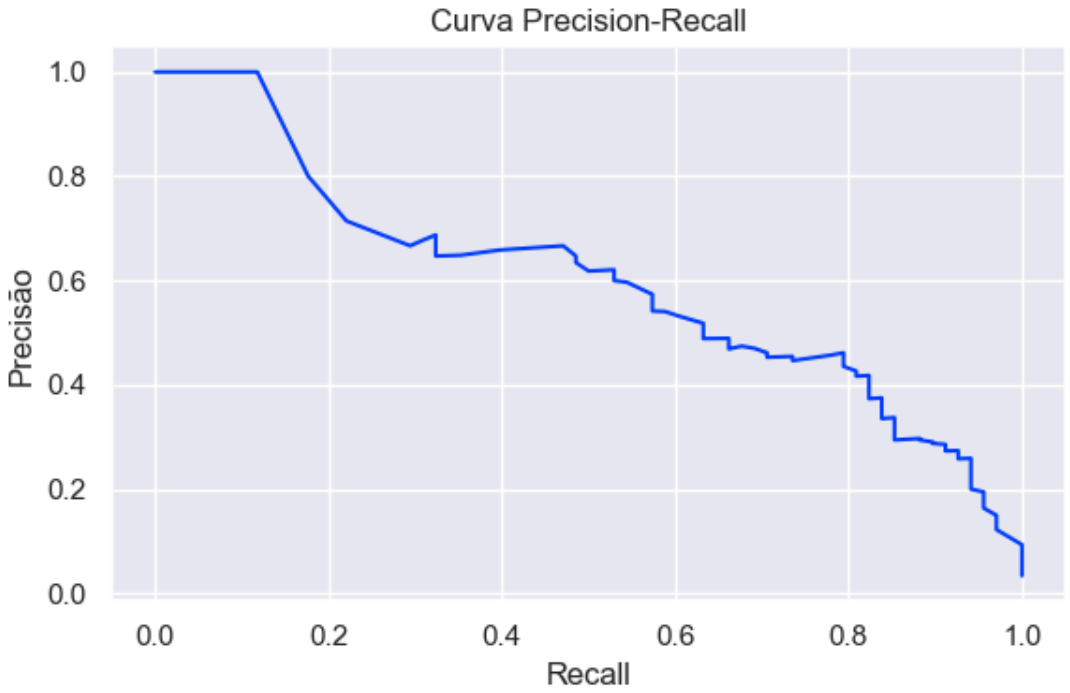


Figura 9 - Curva precisão-recall do modelo binário

Por fim, as variáveis com maior importância para o modelo binário podem ser vistas pela figura 10:

‘

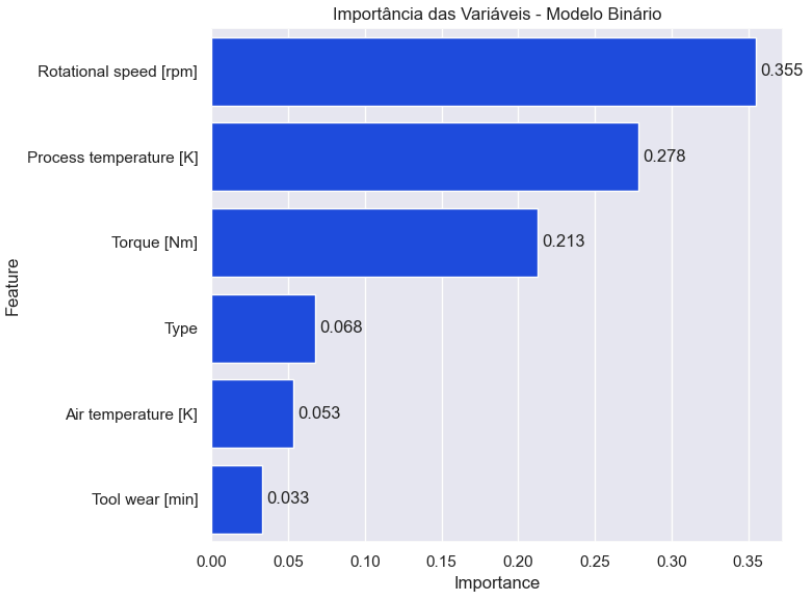


Figura 10 - Importância das variáveis do modelo binário

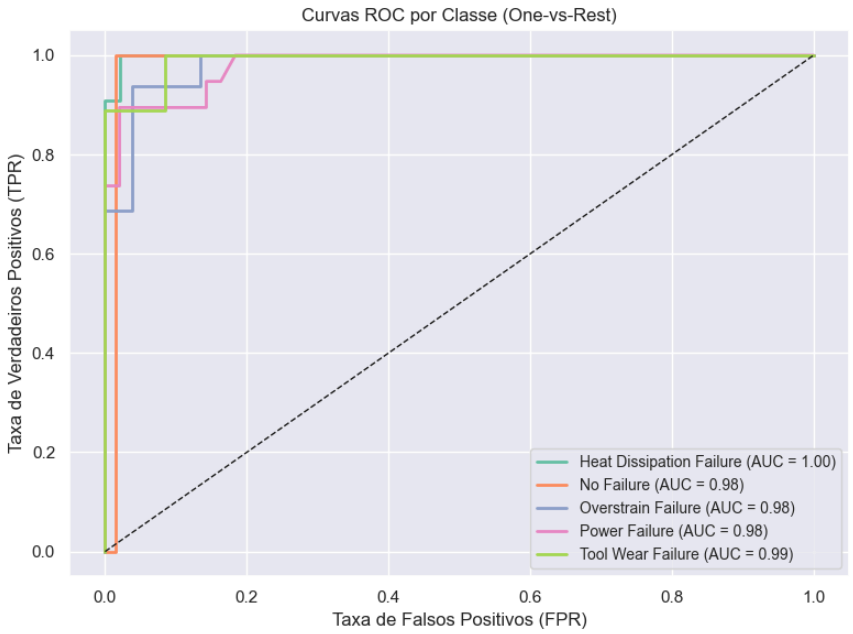
4.5 Classificação multiclasse: Tipo de Falha

A modelagem multiclasse foi realizada somente sobre os registros com falha, utilizando o mesmo pipeline e modelos. O modelo **RandomForest** também se destacou nesta etapa, com desempenho consistente entre as classes. As métricas precision, recall e f1-score foram extraídas por classe com o classification_report (Figura 11):

```
=== Classificação Multiclasses (Tipo de Falha) ===

Modelo: RandomForest
                          precision    recall  f1-score   support

Heat Dissipation Failure       0.92      1.00      0.96        22
              No Failure       0.67      1.00      0.80         2
      Overstrain Failure       0.86      0.75      0.80        16
           Power Failure       0.89      0.84      0.86        19
       Tool Wear Failure       0.89      0.89      0.89         9

                accuracy                           0.88        68
               macro avg       0.84      0.90      0.86        68
            weighted avg       0.88      0.88      0.88        68
```

Figura 11 - Classification Report do modelo multiclasse

Também foi gerada as Curvas ROC por classes (Figura 12), seguindo a abordagem One vs Rest, o que permitiu visualizar a AUC por tipo de falha e realizar uma comparação entre sensibilidade (TPR) e especificidade (FPR) de cada classe.



‘

Figura 12 - Curvas ROC por classe (One vs Rest)

5. ANÁLISE DOS RESULTADOS

5.1 Classificação binária

A etapa de classificação binária teve como foco antecipar se uma falha estaria prestes a ocorrer. O modelo final baseado em RandomForest atingiu alta acurácia (0.95) e AUC-ROC próximo a 0.97, o que demonstra forte capacidade discriminativa.

Entretanto, a principal métrica de interesse nesse contexto foi o recall da classe de falha, uma vez que em ambientes industriais, o custo de uma falha não detectada (falso negativo) é, via de regra, muito superior ao de um falso positivo.

O recall da classe minoritária (falha = 1) alcançou 82%, o que é satisfatório diante do alto desbalanceamento. Isso indica que o modelo conseguiu captar mais da metade dos eventos de falha, mesmo com representatividade inferior a 4% no conjunto original. A utilização de técnicas de balanceamento (SMOTEENN) foi fundamental nesse desempenho.

Além disso, as variáveis com maior impacto no modelo (Rotational Speed, Process Temperature e Torque) são coerentes com as expectativas industriais, sugerindo que o modelo extraiu conhecimento realista da base de dados.

5.2 Classificação multiclasse

A segunda etapa visou determinar qual tipo de falha ocorreu, considerando apenas os casos onde a falha já havia sido identificada. O desempenho do modelo multiclasse foi igualmente satisfatório:

- Acurácia: 0.88

‘

- F1-score ponderado: 0.88
- Curvas ROC com AUCs superiores a 0.98

As curvas ROC por classe demonstraram que o modelo conseguiu aprender padrões distintos de cada tipo de falha, o que amplia sua utilidade em ambientes reais, onde é fundamental não apenas detectar que algo está errado, mas saber o que exatamente está errado.

5.3 Aplicabilidade em cenários reais

O projeto desenvolvido simula com fidelidade um **sistema de manutenção preditiva**:

- A etapa binária funcionaria como um alarme preventivo que identifica anomalias no comportamento da máquina.
- A etapa multiclasse entraria em ação para diagnosticar o tipo específico de falha, permitindo intervenção direcionada.

Apesar do uso de dados sintéticos, os padrões extraídos são consistentes com a lógica industrial e podem servir de base para aplicações reais, desde que treinados com dados operacionais reais e contínuos.

6. CONCLUSÃO

Este projeto teve como objetivo aplicar técnicas de aprendizado de máquina supervisionado para a antecipação de falhas em máquinas industriais e a posterior identificação do tipo de falha, utilizando como base o conjunto de dados sintético AI4I 2020 Predictive Maintenance Dataset.

‘

A abordagem adotada dividiu o problema em duas etapas:

- **Classificação binária**: prever se uma falha ocorrerá (modelo RandomForest com recall de 82% na classe de falha)
- **Classificação multiclasse**: determinar o tipo de falha (acurácia de 88% e AUCs elevados por classe)

O pipeline completo contemplou análise exploratória, tratamento e balanceamento dos dados, modelagem com diferentes algoritmos, ajuste de hiperparâmetros e avaliação robusta com múltiplas métricas. O uso de técnicas como SMOTEENN e validação cruzada contribuiu para um desempenho confiável mesmo em um cenário de forte desbalanceamento.

Embora o dataset seja sintético, os padrões extraídos são coerentes com práticas reais de manutenção, e a estrutura desenvolvida é plenamente adaptável a bases reais com dados operacionais coletados em campo.

Como perspectivas futuras, destacam-se:

- Aplicação da metodologia em dados reais de sensores industriais
- Integração dos modelos em pipelines de monitoramento contínuo
- Exploração de abordagens baseadas em séries temporais para antecipação ainda mais precisa
- Desenvolvimento de painéis interativos para visualização das predições em tempo real

Com isso, conclui-se que a modelagem supervisionada apresenta forte potencial de apoio à manutenção preditiva, contribuindo para maior confiabilidade operacional, redução de custos e otimização dos recursos produtivos.

‘