UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

# Adoção de *random forest* e regressão linear para previsão de falhas em equipamentos agrícolas

**Márcio José Nicola**

Dissertação de Mestrado do Programa de Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria (MECAI)

ICMC USP SÃO CARLOS

**Márcio José Nicola**

# Adoção de *random forest* e regressão linear para previsão de falhas em equipamentos agrícolas

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre – Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria. *VERSÃO REVISADA*

Área de Concentração: Matemática, Estatística e Computação

Orientador: Prof. Dr. Paulino Ribeiro Villas Boas

**USP – São Carlos**
**Junho de 2021**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

Nicola, Márcio José
N634a Adoção de random forest e regressão linear para previsão de falhas em equipamentos agrícolas / Márcio José Nicola; orientador Paulino Ribeiro Villas Boas. -- São Carlos, 2021.
92 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2021.

1. Previsão de falhas. 2. Aprendizado de máquina. 3. Random forest. 4. Regressão linear. 5. Planejamento da manutenção. I. Ribeiro Villas Boas, Paulino, orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

**Márcio José Nicola**

# Random forest and regression analysis adoption for predicting failures in agricultural equipments

Dissertation submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP – in accordance with the requirements of the Professional Master's Program in Mathematics Statistics and Computing Applied to Industry, for the degree of Master in Science. *FINAL VERSION*

Concentration Area: Mathematics, Statistics and Computing

Advisor: Prof. Dr. Paulino Ribeiro Villas Boas

**USP – São Carlos**
**June 2021**

*Para Lígia, Miguel e Heloísa, a quem tanto deixei de dedicar-me para a este trabalho me dedicar.*

# AGRADECIMENTOS

A Deus, que tudo providenciou para que esta jornada fosse possível.

Aos colegas, de dentro e de fora do ICMC, que suportaram, incentivaram, compartilharam, reconheceram e colaboraram.

Aos professores, pela paciência e por repassarem, sem reservas, tamanho conhecimento.

Ao Prof. Dr. Paulino Ribeiro Villas Boas, pela paciência e orientação.

*“ A inteligência artificial está na raiz da mudança de época que estamos vivendo. A robótica pode tornar possível um mundo melhor se estiver unida ao bem comum. (...) Rezemos para que o progresso da robótica e da inteligência artificial esteja sempre a serviço do ser humano.”*

*(Papa Francisco)*

# RESUMO

NICOLA, MÁRCIO J. **Adoção de *random forest* e regressão linear para previsão de falhas em equipamentos agrícolas**. 2021. 92 p. Dissertação (Mestrado – Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

A previsão de falhas em equipamentos agrícolas do setor sucroenergético é particularmente útil para que as equipes de manutenção possam se planejar para o problema e restabelecer a condição operacional dos equipamentos no menor tempo possível, proporcionando assim o aumento dos índices de disponibilidade dos equipamentos, melhor fluxo de abastecimento de matéria-prima nas unidades produtoras, redução da necessidade de estoques de segurança de cana-de-açúcar e dos riscos de degradação da qualidade da matéria-prima em estoque. Neste contexto, explorou-se a possibilidade de técnicas de aprendizado de máquina complementarem as medidas de manutenção já adotadas e desenvolvidas no setor, aumentando assim a previsibilidade de eventos de falha. Tal exploração utilizou-se de dois conjuntos de dados: histórico de falhas de equipamentos e histórico de sensores de telemetria instalados nos equipamentos. Os dados foram extraídos, analisados, tratados e preparados para que modelos baseados em algoritmos de aprendizado de máquina pudessem ser construídos tomando-os como base; contemplam quinze equipamentos do tipo colhedora de cana-de-açúcar e foram coletados por um período de quatro safras. A preparação dos dados gerou dois novos conjuntos: um para predição da causa da próxima falha e outro para predição do tempo de operação. O primeiro modelo adotou, para fins de comparação, as técnicas de *multilayer perceptron* e florestas aleatórias, sendo que a segunda se mostrou mais efetiva. A acurácia da previsão do modelo florestas aleatórias foi de 82,80%, praticamente 20 pontos percentuais (p.p.) acima do modelo que adotou *multilayer perceptron*. O segundo modelo (previsão do tempo de operação), comparou as técnicas de regressão linear, *multilayer perceptron* e florestas aleatórias, sendo a primeira mais efetiva. O erro médio absoluto foi 2,6 horas. Os modelos precisaram ser combinados, pois de forma isolada não atenderam completamente os objetivos estabelecidos inicialmente. Estudos adicionais contemplaram ainda metodologias de gerenciamentos de projetos e o *workbench* computacional WEKA, tendo apresentado ótimos resultados no desenvolvimento desta pesquisa. Como trabalhos futuros, sugere-se o desenvolvimento de aplicações que integrem os modelos propostos e a construção de novos modelos que adotem técnicas baseadas em predição de sequências.

**Palavras-chave:** Previsão de falhas; Florestas aleatórias *(Random forest)*; Regressão linear; Planejamento da manutenção; WEKA.

# ABSTRACT

NICOLA, MÁRCIO J. **Random forest and regression analysis adoption for predicting failures in agricultural equipments**. 2021. 92 p. Dissertação (Mestrado – Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

The failures prediction for agricultural equipment in the sugar-energy sector is particularly useful for maintenance teams to plan for the problem and restore operational condition of equipment as soon as possible thus increasing the equipment's availability, better supply flow of raw materials in production units, reduction of safety stocks of sugarcane and the risks reduction of quality degradation of the raw material in stock. In this context, the possibility of machine learning techniques complementing the maintenance practices already adopted and developed in the sector was explored in order to increase the predictability of failure events. This exploration used two datasets: history of equipment failures and history of telemetry sensors installed in the equipment. Data were extracted, analyzed, processed and prepared so that machine learning models could be constructed by taking them as the basis for; comprise fifteen sugarcane harvester equipment and were collected for a period of four harvests. Data preparation generated two new datasets: the first to predict the next cause failure; and the second for predicting operating hours. The first model adopted, for comparison purposes, multilayer perceptron and random forest techniques; the second of which proved to be more effective. The accuracy of the random forest model was 82.80%, practically 20 p.p. above the model that adopted multilayer perceptron. The second model (operating hours forecast) compared linear regression, multilayer perceptron and random forest techniques; the first was the most effective. The mean absolute error was 2.6 hours. The models needed to be joined, since in an integrated way they completely met the goals. Additional studies also covered project management methodologies and the WEKA computational workbench, that added excellent results in the development of this work. As future works, it is suggested the development of applications that integrate the proposed models and the construction of new models that adopt sequence prediction techniques.

**Keywords:** Fail prediction; Random forest; Linear regression; Maintenance planning; WEKA.

# LISTA DE ILUSTRAÇÕES

# LISTA DE TABELAS

# LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| BFGS | Broyden-Fletcher-Goldfarb-Shanno |
| CRISP-DM | *Cross Industry Standard Process for Data Mining* |
| LGN | Lei dos Grandes Números |
| MLP | Multilayer Perceptron |
| OOB | Out-of-bag |
| p.p. | pontos percentuais |
| RNA | Rede Neural Artificial |

# SUMÁRIO

CAPÍTULO

1

# INTRODUÇÃO

## 1.1 Contextualização

A cultura da cana-de-açúcar no Brasil tem sido relevante fator econômico desde sua implantação, ainda nos tempos coloniais. Seus derivados são amplamente consumidos no mercado doméstico e estão entre os maiores produtos exportados pelo país, fatos que posicionam a indústria sucroenergética entre os maiores empregadores e que mais contribuem positivamente para os resultados da economia brasileira (NASCIMENTO *et al.*, 2020).

Esta vocação nacional, aliada a sucessivos cenários de demanda crescente e preços internacionais atraentes, vivenciados na década de 2010, levaram a indústria da cana-de-açúcar a buscar o aumento de sua produção tendo encontrado, por outro lado, altos custos para expansão de canaviais (custo da terra) e de operações, este último gerado, por exemplo, por pressões ambientais pelo fim das queimadas, comuns no setor até então. Tais desafios encontraram respostas para o controle de custos e aumento da produtividade nas áreas disponíveis e no investimento em mecanização, direcionados principalmente para equipamentos agrícolas do tipo colhedoras, tratores e caminhões (NASCIMENTO *et al.*, 2020).

Todos esses equipamentos são largamente utilizados nos processos agrícolas de uma agroindústria e estão expostos a um ambiente operacional bastante hostil - tráfego em vias não pavimentadas, altas temperaturas, excesso de poeira, entre outros, a condições operacionais exigentes - operação em regime 7x24 (sete dias por dia semana, vinte e quaro horas por dia) e a metas de produção arrojadas, de tal forma que se garanta o melhor desempenho possível desses ativos.

Nas indústrias mais modernas do setor agroindustrial esses equipamentos normalmente estão equipados com sistemas computacionais *(hardware e software)* embarcados, capazes de realizarem o registro detalhado e constante das condições operacionais dos equipamentos, armazenando dados como velocidade, localização e horas de operação, em intervalos regulares

de tempo. Tais sistemas geram um volume de dados expressivo e que tem sido pouco explorados. Pode-se considerar que a velocidade em se equipar os equipamentos com sistemas embarcados foi muito maior do que a velocidade de exploração dos dados gerados por tais sistemas através de tecnologias de mineração de dados. Vivencia-se atualmente uma pequena, ou inexistente, quantidade de aplicações que exploram esses dados. É perceptível também a consciência destas organizações sobre a situação de pouco uso destes dados, comparado à abundância de dados disponíveis.

A hipótese a ser validada nesta dissertação é a possibilidade de adoção de algoritmos de aprendizado de máquina (classificação e previsão) no processo de manutenção automotiva em usinas processadoras de cana-de-açúcar no Brasil. Pretende-se estimar, através do uso das técnicas de redes neurais artificiais, florestas aleatórias e análise de regressão, aplicadas aos dados gerados a partir de sistemas embarcados, quando ocorrerá e qual será um próximo evento de interesse daquele processo. Acredita-se ainda que este modelo pode ser mais eficiente se desenvolvido individualmente, ou seja, por equipamento avaliado, pois desta forma será capaz de capturar nuances específicos que um modelo generalizado não capturaria.

Este tipo de previsão tem aplicabilidade no processo de manutenção uma vez que pode estimar, por exemplo, o momento da próxima falha, o motivo desta falha, qual componente necessita de substituição, o desempenho do equipamento nas próximas horas, entre outros. Conhecendo tais informações, a equipe de manutenção pode se preparar para responder mais rapidamente a tais eventos e diminuir, ou até mesmo eliminar, as perdas ocasionadas pela indisponibilidade por eles causadas.

Note-se que amenizar essa indisponibilidade é importante pois garante a entrega de cana-de-açúcar mais uniforme e contínua nas usinas, minimizando a necessidade de estoques deste produto e os problemas de qualidade decorrentes de manter-se tal estoque por muitas horas (SILVA; ALVES; COSTA, 2011).

Considerando-se o suporte tecnológico ao processo de manutenção nas organizações, a solução proposta visa complementar os sistemas de gerenciamento da manutenção amplamente utilizados nas indústrias e que abrangem, por exemplo, funcionalidades como o gerenciamento dos ativos de manutenção, o suporte à manutenção corretiva e preventiva e a gestão de custos destes ativos.

Os recursos disponíveis para a exploração da hipótese proposta incluem:

- conjunto de dados contendo informações operacionais por intervalo de tempo: contém informações sobre as característica monitoradas do equipamento. É gerado a cada intervalo de tempo registrando seus valores naquele instante;

- conjunto de dados de manutenções de equipamentos: contém informações sobre tempos e motivos que causaram indisponibilidade do equipamento;

- subconjunto de equipamentos disponível para análise: apresenta a relação dos equipamentos que compõem o escopo da análise;
- linguagem de programação Python em ambiente de desenvolvimento Spyder, disponível na distribuição Anaconda; e
- *workbench* computacional WEKA;

Do ponto de vista operacional, o desenvolvimento de um modelo capaz de prever falhas mecânicas tem potencial para otimizar o tempo necessário para restabelecer a condição operacional do mesmo, através de uma melhor logística das equipes de manutenção responsáveis por atender ativos distribuídos por extensas regiões geográficas. O objetivo básico a ser buscado é manter os ativos em operação o maior tempo possível, contribuindo assim para o aumento da produtividades dos mesmos. Espera-se, inclusive, que melhorias em aspectos operacionais sejam refletidas em ganhos financeiros para a organização. A estimação deste ganho, entretanto, requereria acesso a dados e informações que estão fora do escopo desta dissertação.

Acredita-se, a priori, que o problema apresentado pode ser resolvido empregando-se algoritmos de classificação ou regressão, baseados em técnicas estatísticas e de aprendizado de máquina, em um cenário de aprendizado supervisionado, sendo o sucesso da hipótese definido pela capacidade do modelo realizar previsões quando comparado aos dados reais.

## 1.2 Objetivos

### *1.2.1 Geral*

O objetivo geral deste trabalho consistiu em coletar dados de falhas e de telemetria de 15 equipamentos agrícolas do tipo colhedora, em construir modelos de classificação (baseados em RNA MLP e florestas aleatórias) e regressão (baseados em regressão linear, RNA MLP e florestas aloeatórias), em realizar predições do motivo da próxima falha e do tempo restante até que a mesma ocorra e em comparar o desempenho entre as técnicas utilizadas.

### *1.2.2 Específicos*

1. Coletar e analisar dados sobre falhas e de telemetria gerados por 15 equipamentos agrícolas do tipo colhedoras durante 4 safras;
2. Gerar os conjuntos de dados necessários para serem utilizados pelos modelos de classificação e regressão;
3. Propor modelos de classificação utilizando as técnicas RNA MLP e florestas aleatórias;

4. Propor modelos de regressão utilizando as técnicas de regressão linear, RNA MLP e florestas aleatórias;

5. Realizar a predição da causa da próxima falha e do tempo de operação entre falhas para 15 equipamentos agrícolas do tipo colhedora de cana, através dos modelos de classificação e regressão propostos;

6. Comparar o desempenho obtido pelos modelos propostos e definir qual o mais eficiente, com base em medidas de acurácia das predições;

7. Propor modelos individualizados por equipamento e comparar os resultados com o modelo generalizado;

8. Aplicar a metodologia de gerenciamento de projetos de mineração de dados denominada CRISP-DM.

## 1.3 Organização

Esta dissertação está organizada em cinco capítulos: Introdução, Fundamentação Teórica, Materiais e Métodos, Resultado dos Modelos de Classificação e Regressão e Conclusão.

No segundo capítulo são apresentadas as bases teóricas sobre as quais este trabalho foi construído, a saber: (i) técnicas de desenvolvimento de modelos de classificação e regressão: regressão linear, redes neurais e florestas aleatórias; (ii) técnicas de validação de modelos: especificamente *k-fold cross validation*, matriz de confusão, estatística Kappa e técnicas de acurácia; (iii) metodologia para gerenciamento de projetos de mineração de dados;

No terceiro capítulo o leitor é apresentado ao *workbench* WEKA, aos dados disponíveis em seu estado original, à preparação necessária para seu processamento pelos modelos e à análise descritiva dos mesmos.

O quarto capítulo apresenta os modelos desenvolvidos a fim de se alcançar os objetivos propostos, bem como discuti-se os resultados atingidos por estes modelos.

No quinto capítulo é apresentada a conclusão do trabalho, as possibilidades de aplicações práticas e as oportunidades para trabalhos futuros.

CAPÍTULO

# 2

# FUNDAMENTAÇÃO TEÓRICA

## 2.1 Algoritmos de aprendizado de máquina

### 2.1.1 Zero R

Brownlee (2016) propõe a definição de uma linha de base para comparação do desempenho de diferentes modelos aplicados a um conjunto de dados e sugere a adoção, no WEKA, do modelo denominado Zero R.

O funcionamento do algoritmo Zero R, conforme explicam Witten, Frank e Hall (2011), consiste na predição da classe de maior ocorrência, para o caso de classificações, ou da predição do valor médio da variável resposta, para o caso de regressões.

### 2.1.2 Florestas aleatórias (Random forest)

Segundo Breiman (2001), florestas aleatórias *(random forest)* é um algoritmo de classificação composto de diversos classificadores do tipo árvore de decisão, onde cada árvore é formada por um subconjunto de características e de dados de treinamento definidos aleatoriamente, sendo o resultado da predição definido por um processo de votação entre as árvores do modelo.

Esclarece que pode ser aplicado a um conjunto de dados com características contínuas ou categóricas, todavia no segundo caso, a categoria deve ser transformada em valores entre $0$ $e$ $(M-1)$, onde $M$ é o número de valores possíveis que podem ser assumidos pela característica.

E conclui que florestas aleatórias são bastante efetivas para predição de classes, mas apresentaram desempenho inferior em regressões; que a efetividade obtida pelo algoritmo está relacionada com o uso adequado da aleatoriedade na definição das caraterísticas e instâncias de cada árvore; e que não ocorre overfit do modelo devido à Lei dos Grandes Números, assim definida:

> De acordo com a Lei dos Grandes Números (LGN) a média aritmética dos resultados da realização da mesma experiência repetidas vezes tende a se aproximar do valor esperado à medida que mais tentativas se sucederem. Em outras palavras, quanto mais tentativas são realizadas, mais a probabilidade da média aritmética dos resultados observados irá se aproximar da probabilidade real (WIKIPéDIA, 2018).

O pseudo-código para implementação de uma floresta aleatória com capacidade para problemas de classificação e regressão é exibido na figura 1.

1. For $b = 1$ to $B$:
   (a) Draw a bootstrap sample $\mathbf{Z}^*$ of size $N$ from the training data.
   (b) Grow a random-forest tree $T_b$ to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size $n_{min}$ is reached.
      i. Select $m$ variables at random from the $p$ variables.
      ii. Pick the best variable/split-point among the $m$.
      iii. Split the node into two daughter nodes.
2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point $x$:

*Regression:* $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.

*Classification:* Let $\hat{C}_b(x)$ be the class prediction of the $b$th random-forest tree. Then $\hat{C}_{\text{rf}}^B(x) = \textit{majority vote}\ \{\hat{C}_b(x)\}_1^B$.

Figura 1 – Pseudo-código para o algoritmo florestas aleatórias.

Fonte: Hastie, Tibshirani e Friedman (2009).

Hastie, Tibshirani e Friedman (2009) relatam bastante popularidade na aplicação deste algoritmo e compartilham experiência positiva no uso de florestas aleatórias, chamando atenção, todavia, para dois detalhes de implementação: *Out-of-bag (OOB) samples* e importância das características.

O primeiro indica que para cada observação $z_i = (x_i, y_i)$, seus preditores da floresta aleatória devam ser formados **somente** pela média das árvores que utilizam subconjunto de observações onde $z_i$ não está presente. O efeito desta estratégia é análogo à aplicação da validação *N-fold*; o segundo considera a aplicação de critérios de importância das características no momento de se subdividir uma árvore. E justificam da seguinte forma:

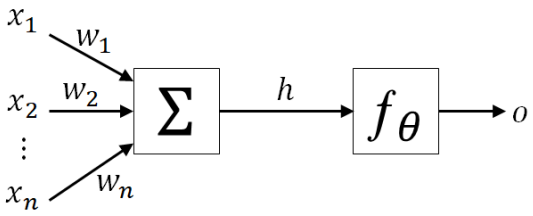
> *In data mining applications the input predictor variables are seldom equally relevant. Often only a few of them have substantial influence on the response; the vast majority are irrelevant and could just as well have not been included (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).*

Quanto à importância das características do conjunto de dados, baseando na influência que cada uma delas exerce sobre variável resposta, (HASTIE; TIBSHIRANI; FRIEDMAN, 2009; JAMES *et al.*, 2013) sugerem a adoção da medida denominada *Gini Importance*, também denominada *Mean Decrease in Impurity (MDI)*, ou impureza média, que calcula a importância de cada característica como sendo a média do número de separações *(splits)* que incluem a característica dividido pelo número de ocorrências que ela separa, considerando todas as árvores da floresta (LOUPPE, 2014).

Para definir quais características adotar, baseando-se na influência que causam na variável resposta, sugere-se a adoção da medida denominada *Gini Index* .

### 2.1.3 Redes neurais artificiais

Antes de entender o funcionamento de uma Rede Neural Artificial (RNA) é importante conhecer o modelo matemático proposto por McCulloch and Pitts em 1943 para o funcionamento de um neurônio biológico. Este esquema está representado na figura 2.




$$h = \sum_{i=1}^{m} w_i x_i \qquad o = f_\theta = g(h) = \begin{cases} 1, se\ h > \theta \\ 0, se\ h \leq \theta \end{cases}$$

(b) (c)

Figura 2 – Modelo matemático de um neurônio.

Fonte: baseada em Marsland (2015).

Este modelo é composto de: **i)** valores de entradas multiplicados por pesos, representando as sinapses; **ii)** um somatório desses valores de entrada, representando a membrana da célula que recebe os pulsos elétricos vindos das sinapses; e **iii)** uma função de ativação (por exemplo, um limiar) que decide se o neurônio dispara para os valores recebidos (MARSLAND, 2015).

Os valores de entrada vêm de outros neurônios e o valor de saída é passado para um próximo neurônio. Os pesos correspondem à intensidade que um sinal é passado para o neurônio.

Já o Perceptron pode ser apresentado como um conjunto destes neurônios conectados pelos mesmos valores de entrada, porém com pesos próprios, trabalhando sem a interferência dos outros neurônios da rede.

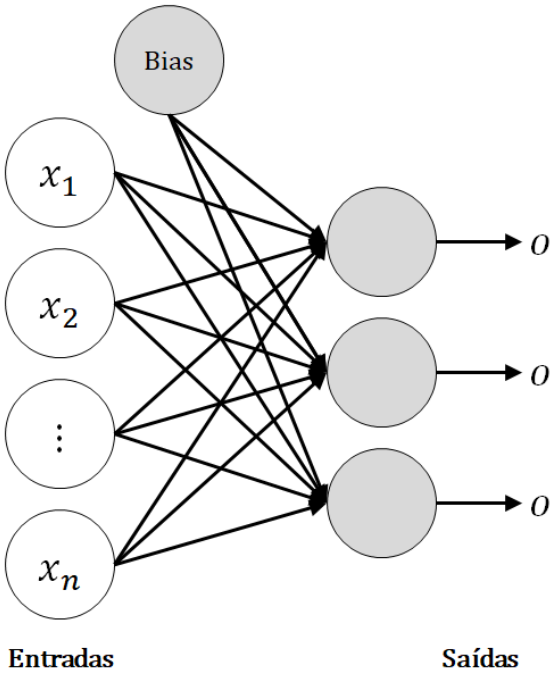



Figura 3 – RNA Perceptron.

Fonte: Elaborado pelo autor.

Para entender como o aprendizado de um Perceptron se dá é preciso considerar dois valores: (i) o valor esperado como saída daquele neurônio; e (ii) o valor calculado por ele. Se (i) e (ii) forem diferentes e considerando que os valores de entrada $x_i$ não podem ser alterados, bem como a função $g(h)$, então resta-nos ajustar o valor dos pesos $w_i$.

Esse ajuste é feito em duas etapas: primeiramente calculando-se o erro entre o valor esperado $t$ e o valor calculado $y$ para o neurônio $j$: $\Delta w_{ij} = (y_j - t_j) \cdot x_i$, onde $i$ indica qual dos pesos está sendo avaliado.

Em seguida ajusta-se o valor do peso $w$ da seguinte forma: $w_{ij} = w_{ij} - \eta \Delta w_{ij}$, onde $\eta$, chamada de taxa de aprendizado, é o valor que define a velocidade de ajuste do peso $w$.

Essas etapas podem ser repetidas por um número finito de vezes, alterando-se o valor de $w$ e $\Delta$ a cada vez. O pseudo-código para o algoritmo Perceptron é exibido na figura 4.

Witten, Frank e Hall (2011) e Marsland (2015) ampliam as variáveis $x$ com a inclusão de uma variável extra $x_0$ de valor fixo diferente de zero, chamada de *bias*, que também terá seus pesos atualizados da mesma forma que as demais.

Todavia é relatado em Witten, Frank e Hall (2011) que o Perceptron funciona perfeitamente para casos onde os dados são linearmente separáveis, ou seja, onde um hiperplano pode ser definido e sugere que problemas mais complexos sejam resolvidos através da interconexão destes neurônios, ou seja, através de uma rede Multilayer Perceptron (MLP), representada na figura 5.

Com a introdução de camadas ocultas haverá também mais pesos contribuindo para a

- **Initialisation**
  - set all of the weights $w_{ij}$ to small (positive and negative) random numbers
- **Training**
  - for $T$ iterations or until all the outputs are correct:
    - for each input vector:
      - compute the activation of each neuron $j$ using activation function $g$:

$$y_j = g\left(\sum_{i=0}^{m} w_{ij}x_i\right) = \begin{cases} 1 & \text{if } \sum_{i=0}^{m} w_{ij}x_i > 0 \\ 0 & \text{if } \sum_{i=0}^{m} w_{ij}x_i \leq 0 \end{cases}$$

      - update each of the weights individually using:

$$w_{ij} \leftarrow w_{ij} - \eta(y_j - t_j) \cdot x_i$$

- **Recall**
  - compute the activation of each neuron $j$ using:

$$y_j = g\left(\sum_{i=0}^{m} w_{ij}x_i\right) = \begin{cases} 1 & \text{if } w_{ij}x_i > 0 \\ 0 & \text{if } w_{ij}x_i \leq 0 \end{cases}$$

Figura 4 – Pseudo-código para o algoritmo Perceptron.

Fonte: Marsland (2015).

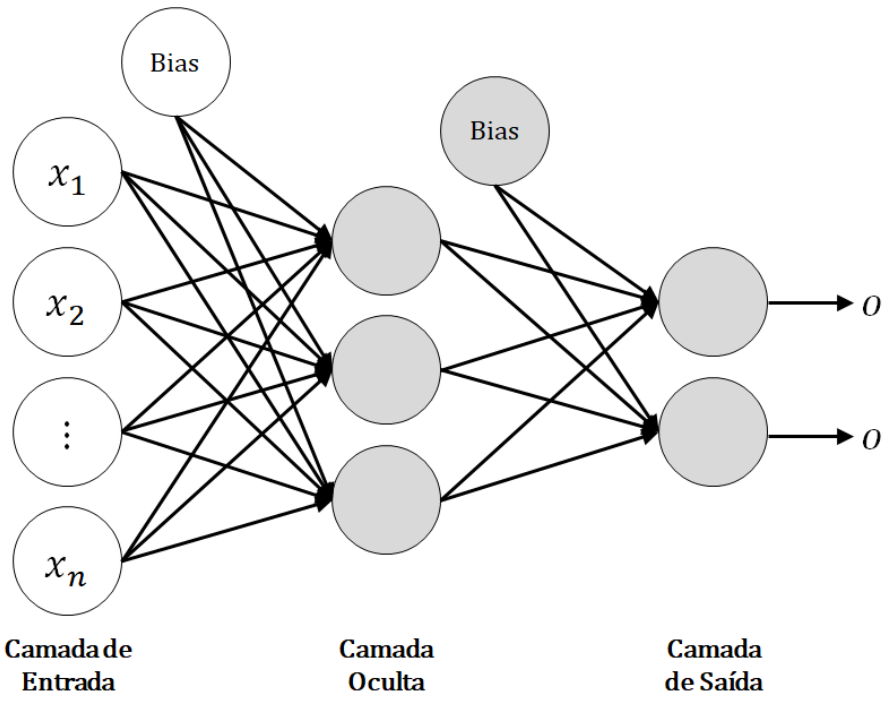



Figura 5 – *RNA Multilayer Perceptron.*

Fonte: Elaborado pelo autor.

camada de saída e por isso a função de ajuste de pesos apresentada anteriormente deixa de ser adequada. O método que ajusta todos os pesos conectados ao neurônio da camada de saída é chamado de *backpropagation error*, sendo este método baseado no algoritmo de otimização matemático chamado gradiente descendente.

Resumidamente o processo de treinamento de uma rede MLP com uma camada oculta funciona da seguinte maneira:

1. Os pesos $w$ são iniciados com valores aleatórios;
2. Um vetor com os valores de entrada é apresentado para a camada de entrada da rede;
3. Esses valores são propagados adiante *(forward)* pela rede da seguinte maneira:
   a) A camada oculta calcula seus valores de saída e os apresenta para a camada de saída;
   b) A camada de saída calcula o valor de saída da rede;
4. O erro entre o valor de saída e o valor desejado é calculado através de alguma função de erro;
5. O erro é propagado para trás *(back-propagation)*, atualizando o valor dos pesos da camada oculta e em seguida da camada de entrada, valendo-se de métodos de otimização para essa etapa.

Existe uma série de funções de ativação que podem ser utilizadas para calcular o valor de saída dos neurônios. A escolha de qual utilizar depende, em grande parte, do tipo de saída da rede: valores discretos, para problemas de regressão, ou categóricos, para problemas de classes binárias ou de múltiplas classes.

O mesmo acontece com o método de otimização do erro. Embora anteriormente tenha sido apresentado o gradiente descendente, outros métodos, como Adam, AdaMax, RMSProp e AdaGrad, podem ser utilizados para o mesmo objetivo, cada um deles, todavia, podendo ser mais adequados a determinadas situações (RUDER, 2017) e (KINGMA; BA, 2015).

## 2.2 Regressão linear

Muitos dos problemas com os quais nos deparamos no dia a dia não são de natureza determinística, e sim probabilística. Isto significa que devemos admitir que os valores envolvidos na análise do problema em questão irão variar de alguma forma aleatória e imprevisível, influenciando da mesma maneira no resultado da análise (MEYER, 1969).

Ao analisar um problema para o qual se sabe que um conjunto de variáveis estão relacionadas entre si e que tal relação é probabilística, a regressão linear tenta encontrar a melhor relação entre uma variável que será chamada de variável resposta Y e as demais variáveis da

análise, que serão chamadas de variáveis independentes (WALPOLE *et al.*, 2009). A regressão linear podem ser simples, quando há apenas uma variável independente, ou múltipla, quando a variável resposta dependerá de duas ou mais variáveis independentes. Uma estrutura de regressão múltipla normalmente é descrita como

$$Y = \alpha + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k, \tag{2.1}$$

onde: Y é a variável resposta, $\alpha$ é o intercepto, $\beta$ são as inclinações e $x_i$ as variáveis independentes.

Uma aplicação óbvia da regressão linear é a capacidade de predizer valores futuros da variável resposta através da substituição dos valores das variáveis independentes na equação.

Um caso importante a se considerar é a presença de variáveis categóricas que precisam ser consideradas no modelo. Essas variáveis ocorrem quando elas não podem ser medidas continuamente, mas assumem valores pré-determinados (região do país, tipo de componente de químico, marca do carro etc.). Nessas situações, considerando que Z é a variável categórica região do Brasil, o que se faz é designar, arbitrariamente, valores numéricos para cada possível categoria de Z, ou seja, região norte: 1, região nordeste: 2, região centro-oeste: 3, região sudeste: 4 e região sul: 5. A partir daí transforma-se essa variável em n outras, onde n representa a quantidade de valores distintos que a variável pode assumir e atribui-se os valores 1 ou 0 a essas variáveis novas, sendo 1 para a variável que representa a categoria original e 0, caso contrário.

A partir daí passa-se a considerar a variável no modelo como se numérica fosse.

A regressão linear também é bastante sensível à quantidade de variáveis inseridas no modelo. Se o problema envolver um grande número de variáveis, vale a pena considerar a aplicação de técnicas de seleção de atributos ou de redução de dimensionalidade do conjunto de dados. Os programas de computador para aplicações estatísticas normalmente implementam essa funcionalidade.

Há ainda muitos casos a serem desenvolvidos com a técnica de regressão onde a variável resposta não apresenta a distribuição normal. Para esses casos, é necessário identificar qual a distribuição da variável resposta e assim definir qual modelo aplicar. Neste ponto é oportuno voltar a citar que os programas de computador para aplicações estatísticas também são preparados para criar modelos para cada distribuição.

## 2.3 Técnica para validação de modelos

As fases de treinamento e validação dos modelos requerem a divisão dos dados disponíveis em dois subconjuntos: um para treinamento dos modelos e outro para validação dos resultados.

Neste trabalho, o conjunto de dados de treinamento foi composto por 90% das instâncias disponíveis; os 10% restantes formaram o conjunto de validação. A técnica adotada para essa divisão foi a *k-fold cross-validation*, enquanto que as técnicas para avaliação da capacidade de previsão foram a matriz de confusão e a estatística *Kappa*, além do emprego de medidas de acurácia.

### 2.3.1 K-fold cross-validation

O método de treinamento e validação conhecido como *k-fold cross-validation* permite maior confiança nos resultados pois realiza o processo de validação em todo o conjunto de dados e não somente em uma porção dele, como ocorre em outras técnicas (PROVOST; FAWCETT, 2013).

James *et al.* (2013) demonstram que estatísticas com o resultado deste processo podem ser calculados apurando-se, por exemplo, a média do erro médio quadrático ou da acurácia de cada etapa do processo.

O método consiste em dividir o conjunto de dados em $k$ partições aproximadamente do mesmo tamanho, que serão chamadas de *folds*. Na primeira iteração, a primeira partição será utilizada para teste e as demais para treinamento do modelo; na iteração *2*, a segunda partição é utilizada para testes e as $(k-1)$ partições para treinamento; e assim sucessivamente até a *k-ésima* iteração.

Cada iteração produz um modelo diferente e os resultados dos testes de cada etapa são então registrados e utilizados para calcular a performance geral do modelo.

## 2.4 Técnicas (métricas) para avaliação de modelos

### 2.4.1 Matriz de confusão

Marsland (2015) apresenta que a **matriz de confusão** deve ser utilizada em problemas de classificação. A ideia é bastante simples: constrói-se uma matriz quadrada com todas as classes listadas na vertical e na horizontal (linhas e colunas). A classe real da instância é representada pelas colunas da matriz; a classe resultante da predição pelas linhas; as respostas corretas serão quantificadas na diagonal principal da matriz.

No exemplo da tabela 1, do total de 36 instâncias de um conjunto de dados hipotético, 26 (soma dos elementos da diagonal principal) foram classificadas corretamente.

A matriz de confusão também pode ser escrita em termos de classificação para as predições para uma classe, assumindo os valores: (i) P (positivo): classe real e classe da predição iguais; (ii) N (negativo): classe real e classe da predição diferentes; (iii) FP (falso positivo): predição igual à classe esperada, mas valor real diferente; e (iv) FN (falso negativo): predição

Tabela 1 – Matriz de confusão hipotética

| | | Classes do conjunto de dados | | |
|---|---|---|---|---|
| | | Classe A | Classe B | Classe C |
| Predições | Classe A | 10 | 2 | 0 |
| | Classe B | 2 | 8 | 2 |
| | Classe C | 4 | 0 | 8 |

diferente do valor da classe esperada, mas valor real igual. A tabela 2 exibe tais classificações para a Classe B.

Tabela 2 – Classificação das predições de uma matriz de confusão

| | | Classes do conjunto de dados | | |
|---|---|---|---|---|
| | | Classe A | Classe B | Classe C |
| Predições | Classe A | N | FN | N |
| | Classe B | FP | P | FP |
| | Classe C | N | FN | N |

### 2.4.2 Acurácia

A acurácia *(accuracy)* indica qual a fração, ou percentual, de elementos foram classificados corretamente durante um processo de predição (MARSLAND, 2015). Continuando com o exemplo da tabela 1, a acurácia pode ser calculada pela soma dos elementos da diagonal principal da matriz, dividido pela quantidade total de instâncias do conjunto de dados, ou ainda como

$$Acuracia = \frac{\#P + \#N}{\#P + \#N + \#FP + \#FN},$$

onde # indica o total de ocorrências daquele tipo.

O desempenho pode ser analisado ainda por outras medidas, entre as quais são destacadas: taxa de positivos verdadeiros *(TP rate)*, taxa de falsos positivos *(FP rate)* e precisão *(precision)*. Esta análise adicional tem o objetivo de avaliar a capacidade de previsão por classe da predição (WITTEN; FRANK; HALL, 2011).

A taxa de positivos verdadeiros é um índice que indica instâncias classificadas corretamente entre o total de instâncias da classe. É definida por:

$$TP\ rate = \frac{\#P}{\#P + \#FN}$$

A taxa de falsos positivos é um índice que indica instâncias classificadas incorretamente entre o total de instâncias que não são da classe em questão. É definida por:

$$FP\ rate = \frac{\#FP}{\#FP + \#N}$$

A precisão é um índice que indica quantas instâncias realmente são daquela classe entre as que foram classificadas como tal. É definida por:

$$Precisao = \frac{\#P}{\#P + \#FP}$$

### 2.4.3 Estatística Kappa

Witten, Frank e Hall (2011) apresentam a estatística Kappa como um coeficiente utilizado para medir a concordância entre os valores de predição obtidos $(p_o)$ por um modelo e o valor previsto $(p_e)$ para o mesmo. É aplicável para modelos de classificação. O valor previsto para a predição é calculado com base na proporção entre o total do número de instâncias preditas para a classe, o número real de instâncias da classe e o número total de instâncias do conjunto de dados.

O valor máximo esperado para a estatística Kappa é 1. O coeficiente Kappa $K$ é definido como (WIKIPéDIA, 2020):

$$k \equiv \frac{p_o - p_e}{\#instancias - p_e}$$

A tabela 3 apresenta o exemplo de uma matriz de confusão dos valores obtidos e previstos para um modelo conforme a estatística Kappa. Com base nestes valores, temos $k = \frac{p_o - p_e}{\#instancias - p_e} = \frac{26-12}{36-12} = 0,5833$.

Tabela 3 – Matriz de confusão para valores obtidos e previstos segundo o coeficiente Kappa $(k)$

| Valores obtidos pelo modelo | | | | |
|---|---|---|---|---|
| | Classe A | Classe B | Classe C | Soma |
| Classe A | 10 | 2 | 0 | 12 |
| Classe B | 2 | 8 | 2 | 12 |
| Classe C | 4 | 0 | 8 | 12 |
| Soma | 16 | 10 | 10 | |

| Valores previstos com base nos valores obtidos pelo modelo | | | | |
|---|---|---|---|---|
| | Classe A | Classe B | Classe C | Soma |
| Classe A | 5,33 | 3,33 | 3,33 | 12 |
| Classe B | 5,33 | 3,33 | 3,33 | 12 |
| Classe C | 5,33 | 3,33 | 3,33 | 12 |
| Soma | 16 | 10 | 10 | |

## 2.5 Colhedoras de cana-de-açúcar

Colhedoras de cana-de-açúcar são equipamentos autopropelidos, dotados de mecanismos para separar as linhas de plantio, despontar, cortar, fracionar, limpar e carregar a cana-de-açúcar em um segundo equipamento denominado veículo de transbordo.

As características operacionais básicas da colhedora de cana-de-açúcar são as seguintes: um mecanismo denominado eliminador de ponteiros situa-se na parte frontal e superior do equipamento e é responsável por despontar os colmos; o sistema de corte é composto por discos que ficam na parte inferior do equipamento e cortam a cana-de-açúcar em sua base; após cortada, um sistema de rolos transporta a cana colhida até os sistemas de picagem e limpeza (extratores) da colhedora, sendo o primeiro responsável por picar os colmos colhidos e o segundo por extrair as impurezas vegetais resultantes da operação; finalizando o processo, um sistema de esteiras transportadoras lança a cana colhida no veículo de transbordo (MAGALHAES, 2009).


Figura 6 – *Colhedora de cana-de-açúcar.*

Fonte: (CASE, 2020) .

Na figura 6 é exibida uma colhedora de cana-de-açúcar com seus principais mecanismos destacados, sendo eles: **1)** elevador; **2)** divisor de linha; **3)** corte de base; **4)** picador; **5)** rolos; **6)** extrator; e **7)** sistema rodante.

## 2.6 Metodologia CRISP-DM para projetos de mineração de dados

Com o objetivo de avaliar se metodologias para gerenciamento de projetos são aplicáveis a projetos de mineração de dados, esta dissertação aplicou princípios da metodologia chamada CRISP-DM durante seu desenvolvimento.

Metodologia significa um "conjunto de métodos, regras e postulados usados em determinada disciplina, e sua aplicação" (FERREIRA, 2010).

"Projeto é um esforço temporário empreendido para criar um produto, serviço ou resultado único" (PMI, 2014). E sendo temporário, possui início, meio e fim, ou seja, um ciclo de vida, que "é a série de fases pelas quais um projeto passa, do início ao término" (PMI, 2014).

Estes ciclos de vida podem ser de três tipos: (a) preditivos; (b) iterativos e incrementais; e (c) adaptativos, sendo cada um destes mais adequado ao conhecimento que se tem, de antemão, do produto a ser entregue e podem ser documentados por metodologias (PMI, 2014).

Especificamente para projetos de mineração de dados existe a metodologia chamada *Cross Industry Standard Process for Data Mining* (CRISP-DM), cujo ciclo de vida proposto está representado na figura 7.

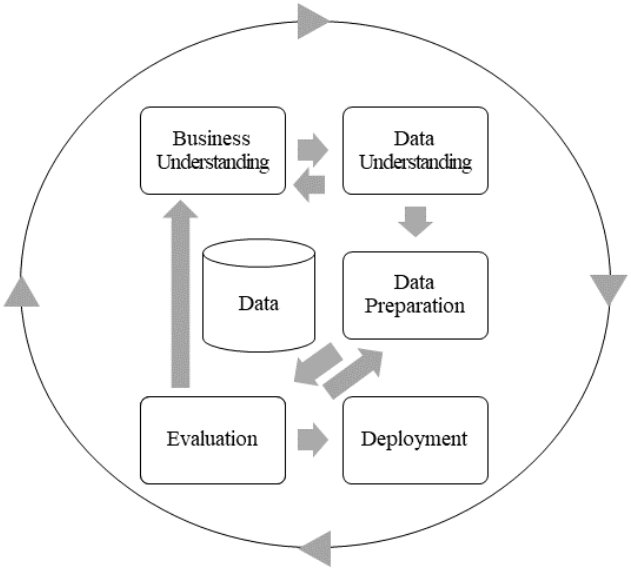



Figura 7 – Ciclo de vida de projetos proposto pela metodologia CRISP-DM.

Fonte: Chapman *et al.* (2000)

A CRISP-DM é uma metodologia de caráter iterativo. Isso acontece pois cada ciclo concluído pode dar início a um novo ciclo capaz de explorar ainda mais profundamente os resultados obtidos no ciclo anterior (PROVOST; FAWCETT, 2013). Uma breve descrição de cada fase desta metodologia, bem como as principais atividades que nelas devem ser realizadas, pode ser conferida na tabela 4.

Tabela 4 – Resumo das fases da metodologia CRISP-DM

| Fase | Descrição da Fase | Principais atividades |
|---|---|---|
| *Business Understanding* | Nesta fase acontece o mapeamento dos requisitos, o entendimento do projeto do ponto de vista do cliente e a caracterização do projeto como sendo um projeto de mineração de dados; | (i) determinação dos objetivos de negócio; (ii) avaliação da situação atual; e (iii) elaboração do plano de projeto. |
| *Data Understanding* | Nesta fase a equipe começa a se familiarizar com os dados do projeto. A qualidade dos dados é avaliada, os primeiros achados devem começar a aparecer, bem como a formulação de hipóteses. | (i) coleta dos dados iniciais; (ii) descrição dos dados; (iii) exploração dos dados; e (iv) verificação da qualidade dos dados. |
| *Data Preparation* | O produto final desta fase é a disponibilização de dados para a fase de modelagem. Suas atividades podem se repetir quantas vezes forem necessárias, dependendo das características dos conjuntos de dados disponíveis. | (i) seleção dos dados; (ii) limpeza dos dados; (iii) geração de novos dados; (iv) integração dos dados; e (v) formatação dos dados. |
| *Modeling* | Nesta fase aplica-se as técnicas de mineração de dados nos dados que foram preparados na fase anterior. Há várias técnicas disponíveis para serem aplicadas (árvores de decisão, redes neurais e regressão, por exemplo), sendo que algumas delas terão necessidades específicas para os dados. O produto final desta fase são os modelos construídos. | (i) seleção das técnicas de modelagem; (ii) definição e execução do plano de validação do modelo; (iii) construção do modelo; e (iv) avaliação da adequação do modelo. |
| *Evaluation* | Nesta fase avalia-se o resultado gerado pela modelagem, os passos para geração do mesmo e se os resultados estão alinhados como os objetivos de negócio e outros critérios de avaliação definidos na primeira fase. O produto final da fase é a aprovação do modelo/necessidade de revisão e o inventário dos achados não relacionados ao projeto. | (i) avaliação dos resultados; (ii) revisão do processo; e (iii) determinação dos próximos passos. |
| *Deployment* | Aqui obtém-se a entrega final do projeto. Pode ser um modelo da mineração de dados, um programa de computador, uma lista de problemas ou um ajuste em algum processo de negócio. Normalmente os responsáveis pela construção/desenvolvimento não serão os mesmos das fases anteriores. Um relatório resumindo os elementos utilizados, as entregas, os achados obtidos e a experiência vivenciada também pode ser produzido. | (i) planejamento da construção/desenvolvimento; (ii) planejamento do monitoramento e manutenção; (iii) produção do relatório final; e (iv) revisão o projeto. |

CAPÍTULO

# 3

# MATERIAIS E MÉTODOS

## 3.1 *Workbench* computacional WEKA

A análise dos dados e a construção dos modelos discutidos nesta dissertação foram desenvolvidos com o apoio do *workbench* computacional chamado WEKA, amplamente utilizado tanto para fins educacionais, de pesquisa e comerciais.

O WEKA, acrônimo para *Waikato Environment for Knowledge Analysis* é um *workbench* computacional que permite aos seus usuários a aplicação de diversas técnicas de aprendizado de máquina em conjuntos de dados reais ou gerados pelo próprio *workbench* (HALL *et al.*, 2009).

Para isso, disponibiliza diversas interfaces de usuário, sendo que as mais utilizadas são as *Explorer* e *Experimenter*.

A interface de usuário *Explorer* é composta por seis painéis: *Preprocess, Classify, Cluster, Associate, Select attributes* e *Visualize*, cujas funções são:

- *Preprocess*: acessar, analisar, visualizar e aplicar transformações aos dados;
- *Classify*: aplicar e avaliar algoritmos de classificação ou regressão sobre conjunto de dados disponível no painel *Preprocess*;
- *Cluster* e *Associate*: aplicar e avaliar algoritmos de agrupamento ou associação, respectivamente, sobre conjunto de dados disponível no painel *Preprocess*;
- *Select attributes*: aplicar e avaliar técnicas de identificação de importância de atributos;
- *Visualize*: visualizar um scatter plot dos atributos do conjunto de dados.

Por outro lado, a interface de usuário *Experimenter* pode ser utilizada para comparação de resultados obtidos entre diferentes algoritmos. Apesar de ser menos utilizada que as demais, a

avaliação de resultados pode ser realizada mais facilmente através desta opção (HALL *et al.*, 2009).

O WEKA oferece suporte a diversos tipos de arquivos de dados, como CSV, JSON, C4.5 e o formato próprio ARFF. Permite, inclusive, acessos a banco de dados SQL.

A partir da versão 3.0 (lançada no ano de 1.999) foi completamente desenvolvida em Java. Atualmente encontra-se na versão 3.8.4, conforme pode ser constatado na figura 8.

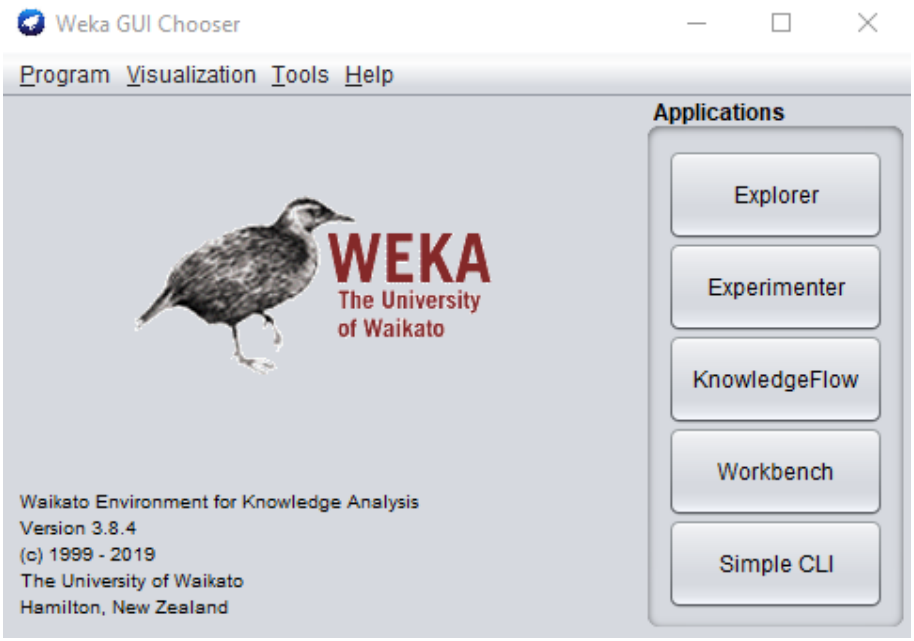



Figura 8 – Janela inicial do *workbench* WEKA.

## 3.2 Conjuntos de dados

O objetivo desta seção é o de explanar como os dados estão gravados em suas fontes originais e como foram recuperados e contextualizar, de maneira descritiva, as condições operacionais dos equipamentos analisados.

Todos os dados que foram utilizados pelos modelos desenvolvidos (características operacionais, histórico de falhas de equipamentos e relação de equipamentos) estavam armazenados em tabelas de sistemas gerenciadores de banco de dados relacional. A figura 9 representa como esses dados estão estruturados neste sistema.

O objetivo da coleta e armazenamento dos dados que foram analisados não é o de realizar as previsões que estão sendo propostas, mas sim o de avaliar a operação dos equipamentos, gerar indicadores de desempenho e rastrear rotas percorridas, por exemplo. Por esse motivo, discutiu-se nesta etapa se as fontes de dados disponíveis se adéquam à resolução do problema, ou seja, se podem ser exploradas por técnicas de aprendizado de máquina de tal forma que seja possível, através delas, obter as respostas necessárias para se atingir os objetivos pretendidos.

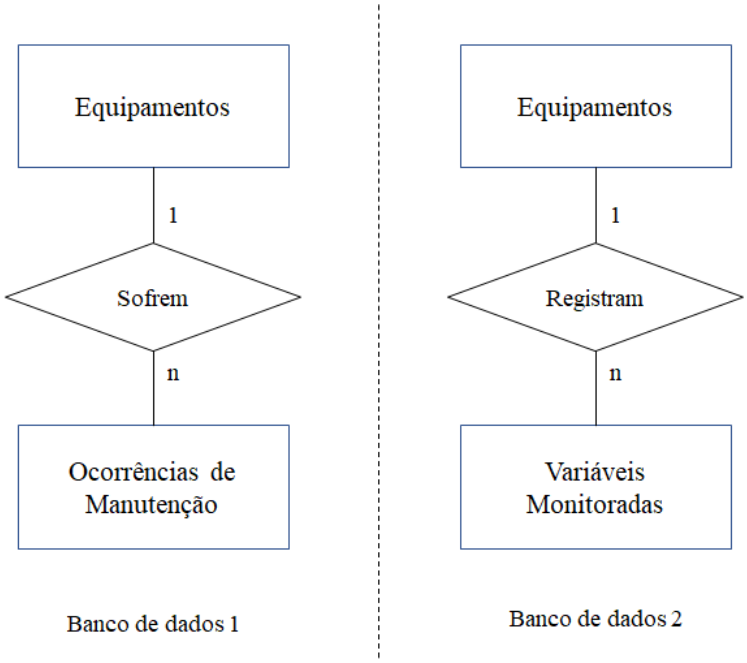



Figura 9 – Modelo entidade-relacionamento teórico das tabelas.

*Fonte: elaborada pelo autor.*

O conjunto de dados de histórico de falhas armazena o registro das manutenções realizadas nos equipamentos agrícolas. Possui cerca de 195 mil ocorrências sobre as manutenções de 780 equipamentos, realizadas durante o período de 7 anos. Este conjunto de dados é resultado da junção de duas tabelas: (i) tabela de ocorrências de manutenção; e (ii) tabela de equipamentos. É composto por dez características e sua estrutura está definida na tabela 5.

Tabela 5 – Características do conjunto de dados de manutenções de equipamentos

| Característica | Descrição | Tipo de Dado |
|---|---|---|
| 1 | Safra que ocorreu a parada | Núm. Int. |
| 2 | Equipamento onde ocorreu o problema | Núm. Int. |
| 3 | Código do problema | Núm. Int. (categórico) |
| 4 | Descrição do problema | Alfanumérico |
| 5 | Código do detalhe do problema | Núm. Int. (categórico) |
| 6 | Descrição do detalhe do problema | Alfanumérico |
| 7 | Equipe responsável pela manutenção | Alfanumérico |
| 8 | Data / hora do início da parada | Data |
| 9 | Data / hora do término da parada | Data |
| 10 | Tempo de indisponibilidade do equipamento | Núm. Real |

O método de coleta destes dados é manual e obedece ao seguinte fluxo: ao ocorrer um problema no equipamento o operador relata, através de rádio comunicador ou telefone celular, a situação à equipe responsável pela manutenção; esta, por sua vez, realiza o registro inicial da falha, informando o equipamento e o momento da ocorrência; em seguida é enviada uma equipe de campo para realizar a manutenção. Quando o equipamento é liberado para o trabalho então a

equipe de manutenção é avisada e finaliza o registro do problema, informando o motivo da falha e o momento em que o equipamento voltou a operar.

Os dados relevantes para o modelo pertencem a 50 equipamentos do tipo colhedora de cana e estão distribuídos, do ponto de vista temporal, por um período de 4 safras, totalizando aproximadamente 35.000 ocorrências. Uma vez que os dados estão organizados por safra, não haverá ocorrências em todos os meses do ano, mas apenas naqueles meses em que a colheita da cana-de-açúcar está ocorrendo - normalmente de abril até novembro.

O conjunto de dados operacionais contém informações sobre os sensores monitorados e registra seus valores a cada intervalo de tempo. Possui aproximadamente 30.000.000 de instâncias referentes a 521 equipamentos, abrangendo o período de 5 anos. É resultado da junção de duas tabelas: i) valor dos sensores; e ii) equipamentos. O volume de dados é de aproximadamente 2.000.000 de instâncias, quando restritos aos mesmos equipamentos e período descritos no conjunto de dados de manutenções de equipamentos. É composto por 13 características e sua estrutura está definida na tabela 6.

Tabela 6 – Características do conjunto de dados de sensores operacionais

| Característica | Descrição | Tipo de Dado |
|---|---|---|
| 1 | Código do Equipamento | Núm. Int. |
| 2 | Data / hora do registro do valor das características | Data |
| 3 | Duração em horas do período de apuração da característica | Núm. Real |
| 4 | Atividade que o equipamento realizava no momento da medição | Núm. Int. |
| 5 | Quilômetros percorridos no intervalo de apuração da característica | Núm. Real |
| 6 | Velocidade média apurada no intervalo de apuração da característica | Núm. Real |
| 7 | RPM do motor apurada no intervalo de apuração da característica | Núm. Int. |
| 8 | Horas de operação em marcha ré no intervalo de apuração da característica | Núm. Real |
| 9 | Horas de operação da esteira em reversão no intervalo de apuração da característica | Núm. Real |
| 10 | Horas de operação do cortador no intervalo de apuração da característica | Núm. Real |
| 11 | Horas de operação do elevador no intervalo de apuração da característica | Núm. Real |
| 12 | Horas de motor acionado no intervalo de apuração da característica | Núm. Real |
| 13 | Horas de motor ocioso no intervalo de apuração da característica | Núm. Real |

O subconjunto de equipamentos analisados foi derivado do conjunto de dados de registro de manutenções, ou seja, não foram considerados todos os equipamentos, mas somente aqueles

para os quais existiam manutenções registradas e, ao mesmo tempo, são escopo deste trabalho.

O método de coleta destes dados é híbrido. A maioria das informações não requer atividades manuais e ocorre da seguinte maneira: todos os equipamentos objeto deste estudo são equipados com dispositivos de computação embarcada, conhecidos comercialmente como computadores de bordo, equipados com sensores neles instalados e com capacidade de capturar dados em tempo real sobre suas condições operacionais. Estes dados são armazenados no computador de bordo e depois coletados por dispositivo do tipo coletor de dados, preparados especificamente para esse fim e transmitidos, via internet, para o sistema gerenciador de banco de dados instalado remotamente.

Este sistema é responsável por armazenar e processar, de forma centralizada, os dados oriundos de todos os equipamentos. A última característica que complementa este conjunto de dados é o tipo de operação que o equipamento está realizando, sendo esta informada manualmente pelo operador do equipamento. Os dados, tanto aqueles capturados pelos sensores ou pelo apontamento do operador, são armazenados a cada vez que um "gatilho" de gravação é disparado.

Cabe ressaltar que:

- por pertencerem a sistemas transacionais diferentes, os códigos de equipamentos são diferentes nos dois conjuntos de dados, logo precisaram de uma regra de mapeamento para que as informações pudessem ser combinadas entre eles;
- não existem informações textuais nos conjunto de dados;
- todas as informações necessárias estavam disponíveis de forma eletrônica.

### 3.2.1 Análise dos dados coletados

Uma vez apresentada a forma como os dados estão armazenados em sua origem e como foram coletados, foi iniciada a fase de análise destes dados, bem como a avaliação de sua utilidade para os modelos. O relato desta análise está apresentado a seguir.

#### 3.2.1.1 Dados sobre falhas

Nesta dissertação, entende-se como falha o estado no qual o equipamento é incapaz de realizar a função esperada para ele, sendo resultante de um dos mecanismos do equipamento. Como resultado, a falha gera a parada total do equipamento e requer uma manutenção corretiva.

A análise inicial dos dados, realizada a fim de orientar o modelo de previsão de falhas a ser desenvolvido, foi iniciada pelos dados de manutenção. Com a finalidade da simplificação da análise foram escolhidos, aleatoriamente, 30% dos equipamentos com informações coletadas, ou seja, 15 equipamentos.

Para estes equipamentos foram identificados 23 motivos de falha. Deste total, foram selecionados os 10 principais motivos por representarem, aproximadamente, 75% dos problemas em termos de número de ocorrência e 70% em termos de tempo. Os demais motivos foram agrupados em uma categoria denominada “Outros”. A frequência de ocorrência e o tempo total de indisponibilidade podem ser visualizados nas figuras 10 e 11.

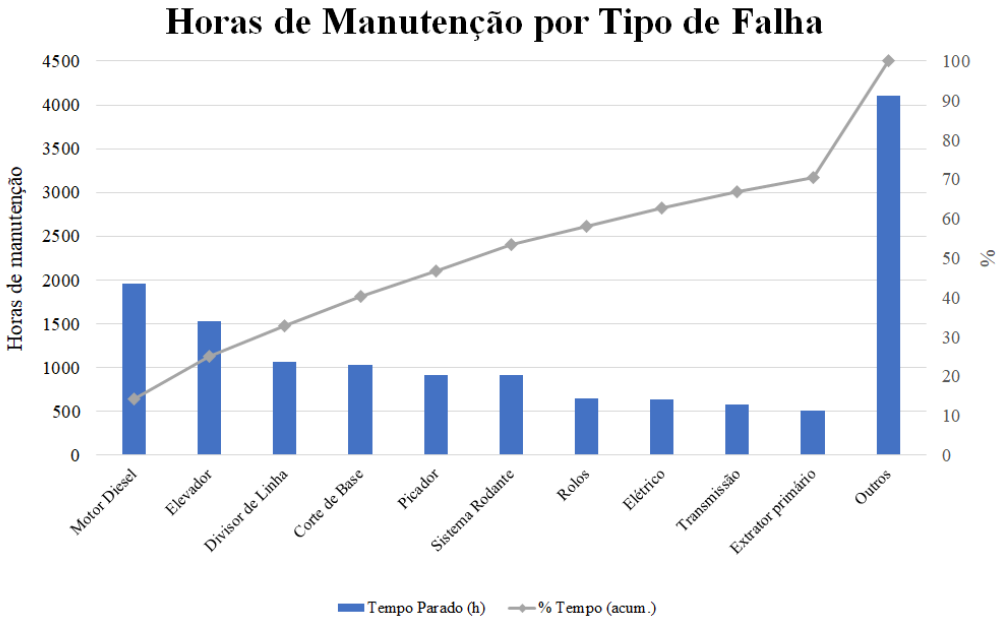



Figura 10 – Gráfico de horas de manutenção e percentual acumulado.

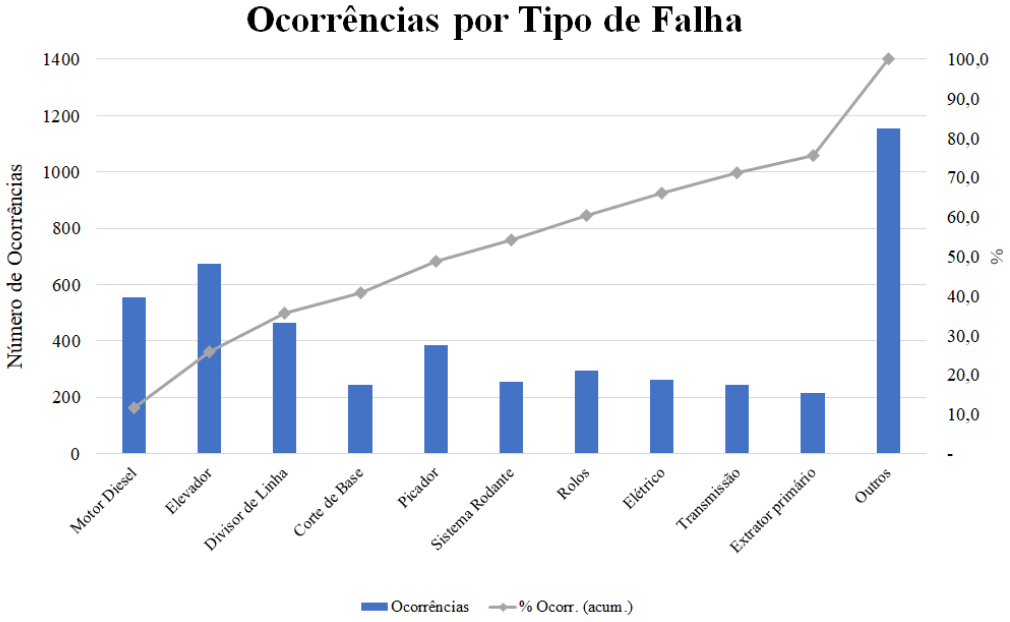



Figura 11 – Gráfico de ocorrências de manutenção por tipo de falha e percentual acumulado.

O tipo de falha Motor Diesel, mesmo não sendo a categoria com maior número de ocorrências, apresenta o terceiro maior tempo médio para reparo e o terceiro maior desvio

padrão, fato que a torna a categoria que causou a maior indisponibilidade entre equipamentos analisados. A análise do tempo médio entre falhas do mesmo tipo, bem como do desvio padrão, evidenciam grande dispersão dos dados, conforme indicam as figuras 12 e 13.

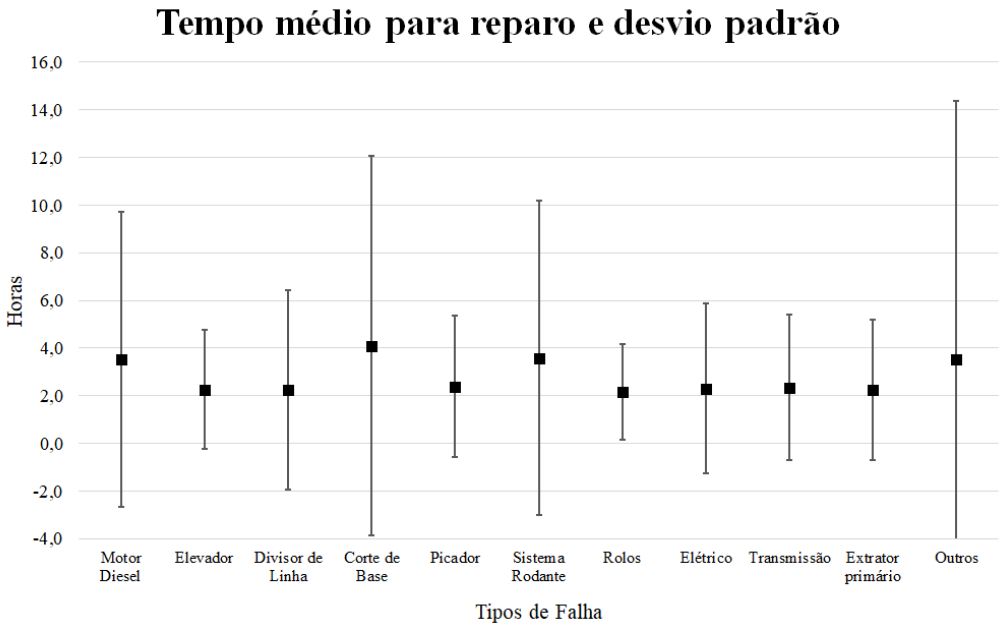



Figura 12 – Gráfico do tempo médio para reparo por tipo falha.

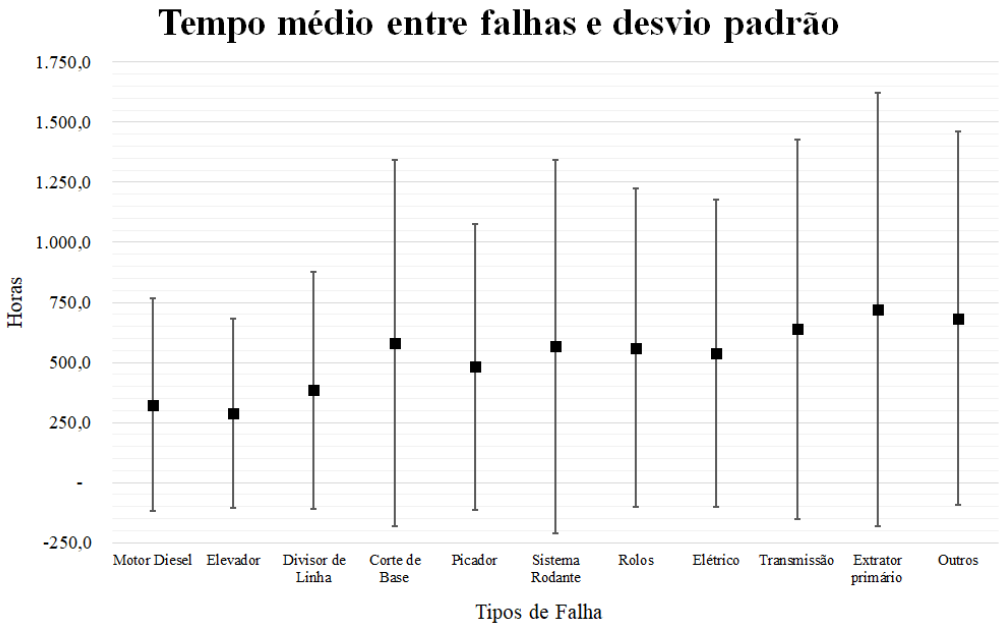



Figura 13 – Gráfico do tempo médio entre falhas.

Quando avalia-se o tempo médio entre quebras por equipamento, e não por tipo de falha, apura-se o valor de 46,8 horas e desvio padrão de 76,7 horas.

A análise descritiva evidencia também que a maioria das falhas ocorre nas primeiras horas de operação após a falha anterior, tendo ocorrido 70% delas em até 48 horas. Esta distribuição

de quebras por intervalo de tempo é exibida na figura 14; a figura 15 exibe a mesma informação, porém categorizada por intervalos de 24 horas.

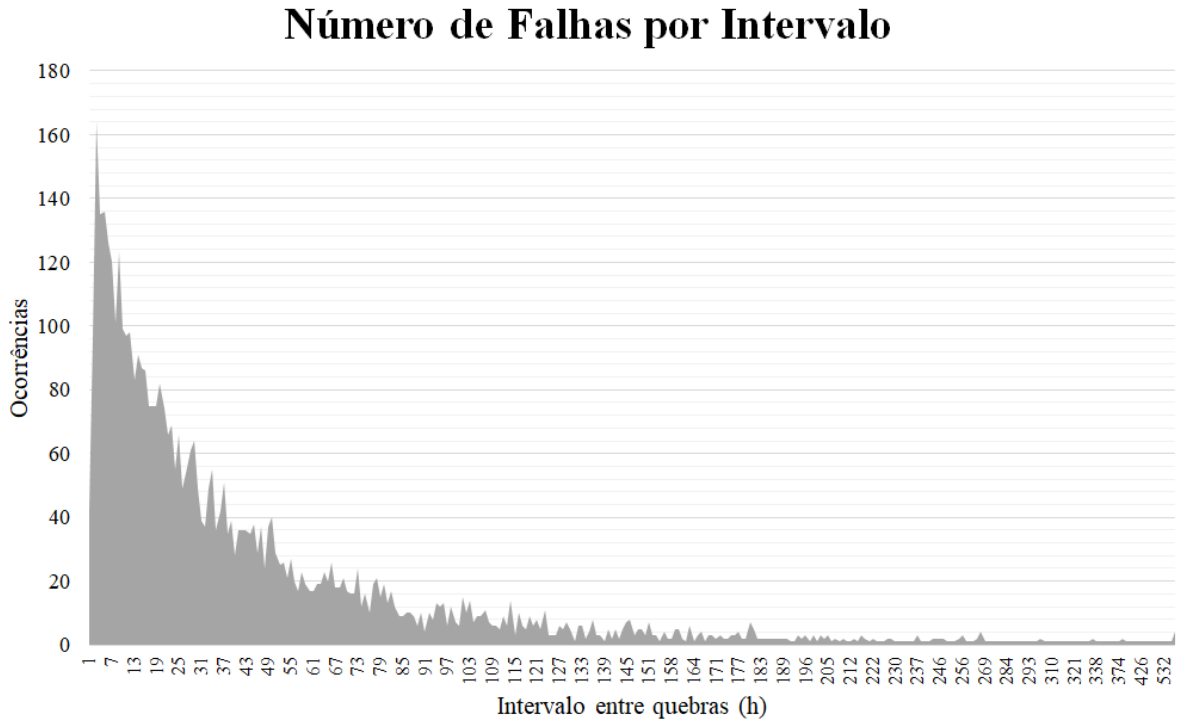



Figura 14 – Gráfico do número de falhas por intervalo de tempo (horas).

A fim de exemplificar-se a variação de motivos e intervalos constatados nos dados descritos anteriormente, a tabela 7 exibe as falhas ocorridas em cinco equipamentos durante uma semana completa de trabalho e o intervalo em relação à falha anterior, sendo que os equipamentos e a semana foram escolhidos aleatoriamente.

No conjunto de dados de manutenções estavam registradas ainda as manutenções preventivas, cujas análises não foram abordadas por se tratarem de manutenções planejadas, com periodicidade conhecida e não serem decorrentes de falhas.

#### *3.2.1.2 Dados sobre operação*

Conforme apresentado na tabela 6, este conjunto de dados é composto por treze características. As três primeiras (equipamento, data e duração da medição) referem-se a informações sobre a coleta dos dados; as demais (atividade realizada, quilômetros, velocidade, RPM, horas de operação em marcha ré, horas de operação da esteira em reversão, horas de operação do cortador, horas de operação do elevador, horas de motor acionado e horas de motor ocioso) são as informações monitoradas por sensores.

O período de tempo de monitoramento, para os 15 equipamentos analisados, abrangeu aproximadamente 630.000 quilômetros percorridos e 187.000 horas de operação. Desse total de horas, por volta de 87% do tempo foram aplicadas em atividades consideradas produtivas

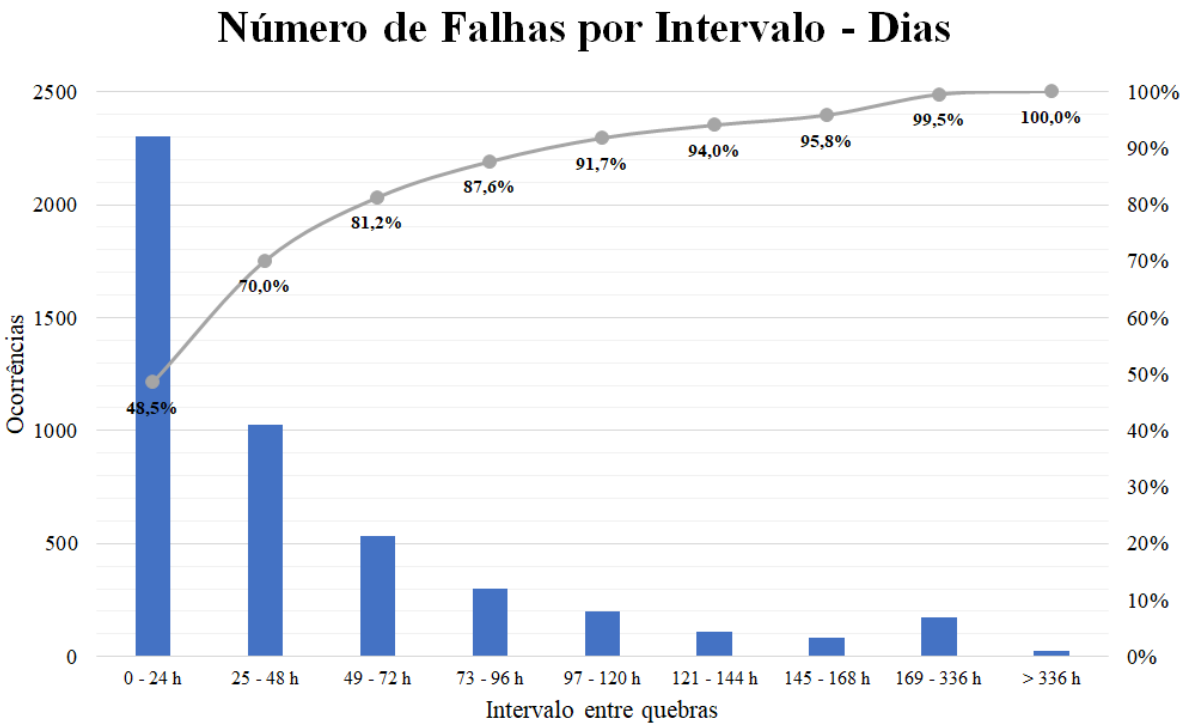



Figura 15 – Gráfico do número de falhas por intervalo de tempo (dias).

Tabela 7 – Falhas ocorridas em uma semana de trabalho de cinco equipamentos

| Dia | Equip. 1 | | Equip. 2 | | Equip. 3 | | Equip. 4 | | Equip. 5 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Falha | Int. | Falha | Int. | Falha | Int. | Falha | Int. | Falha | Int. |
| 1 | — | - | Outros | 105 | Extrator primário | 17 | — | - | — | - |
| 2 | — | - | — | - | Divisor de Linha | 14 | Extrator primário | 146 | — | - |
| | — | - | — | - | Extrator primário | 15 | Elevador | 19 | — | - |
| 3 | Extrator primário | 48 | Extrator primário | 41 | — | - | — | - | — | - |
| 4 | Elevador | 25 | — | - | — | - | Divior de linha | 35 | — | - |
| | Outros | 13 | — | - | — | - | — | - | — | - |
| | — | - | — | - | — | - | — | - | — | - |
| 5 | — | - | — | - | — | - | — | - | — | - |
| | — | - | — | - | — | - | — | - | — | - |
| 6 | — | - | — | - | Picador | 83 | — | - | — | - |
| | — | - | — | - | Picador | 14 | — | - | — | - |
| 7 | Motor Diesel | 57 | Corte de base | 87 | — | - | — | - | — | - |
| | Elevador | 9 | Corte de base | 18 | — | - | — | - | — | - |
| | Elevador | 6 | — | - | — | - | — | - | — | - |

e o restante em atividades de preparação ou períodos de inatividade por motivos climáticos. Esses tempos estão representados na figura 16 e diferenciados quanto aos tempos de produção, preparação e fatores climáticos por safra na figura 17.

Mesmo tratando-se de tempo de inatividade, as horas classificadas como "fatores climáticos" não estão agrupadas com os dados de manutenção e não foram escopo de previsão dos modelos desenvolvidos.

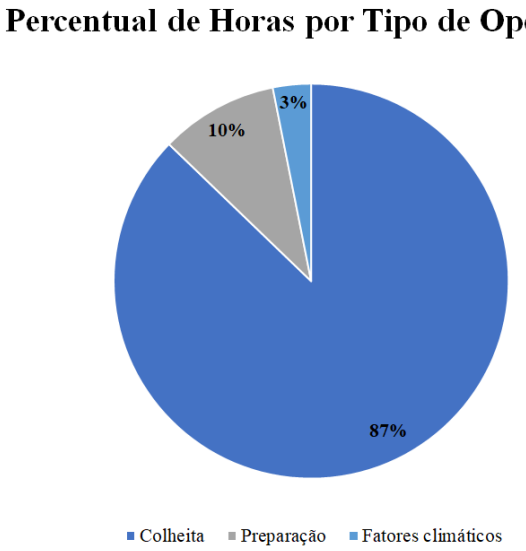



Figura 16 – Gráfico do percentual de horas por tipo de operação.

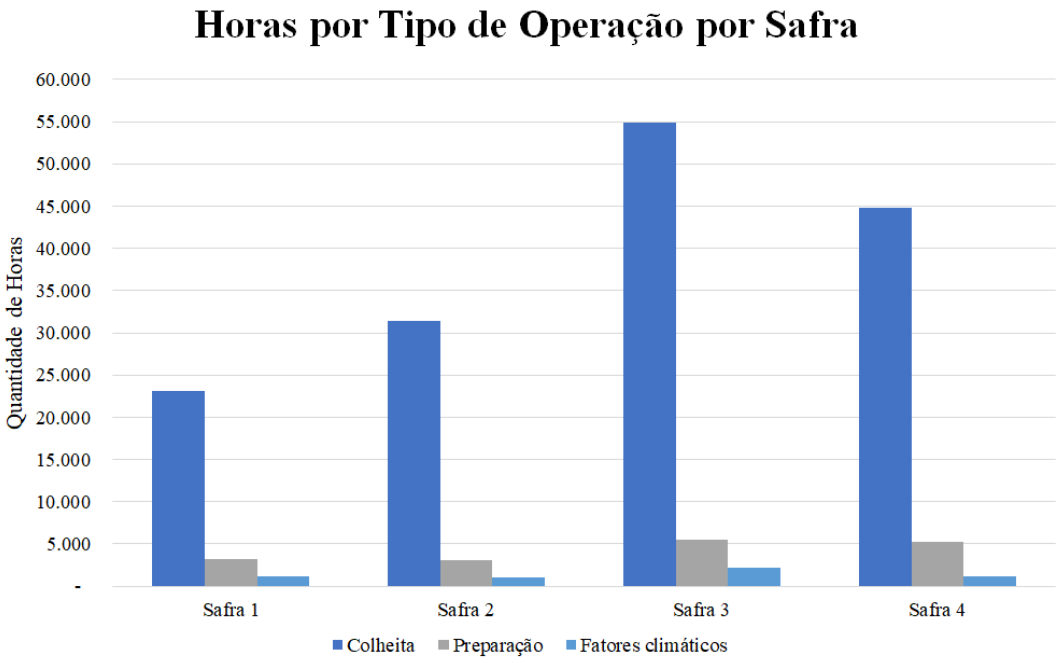



Figura 17 – Gráfico de horas por tipo de operação.

As características operacionais monitoradas pelos sensores instalados nos equipamentos sofrem influência de diversos fatores ambientais, como variedade e idade da planta que está

sendo colhida, tipo de solo, relevo do terreno, condições de visibilidade, experiência do operador, temperatura ambiente, quantidade de manobras necessárias, entre outros.

Tais condições requerem operações adequadas à situação e causam desgastes diferentes nos diversos sistemas dos equipamentos, ocasionando assim as falhas analisadas. O gráfico da figura 18 exibe os tempos de uso de cada componente apurado durante o período de monitoramento dos equipamentos.

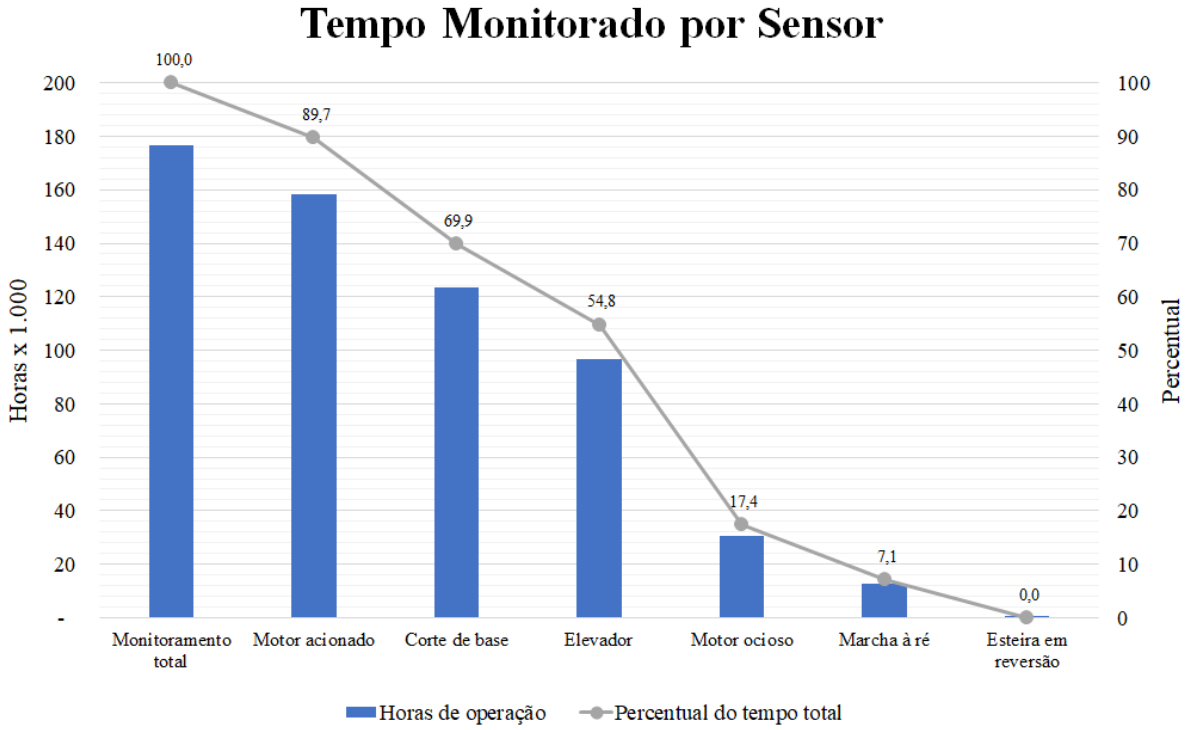



Figura 18 – Gráfico de tempos monitorados por sensor.

Além do uso disforme dos sistemas dos equipamentos devido às condições operacionais, analisando-se o uso diário dos mesmos, observa-se também bastante variação em todas as características monitoradas. Tais informações estão listadas na tabela 8.

A fim de complementar a exemplificação iniciada na tabela 7, os gráficos da figura 19 exibem as médias ou totais diários de algumas características de telemetria, para os mesmos equipamentos e semana utilizados para elaboração daquela tabela.

#### 3.2.1.3 Conclusão

As análises realizadas sobre os dois conjuntos de dados disponíveis evidenciam grande variação entre os intervalos de quebra dos equipamentos, bem como na variação de uso dos mesmos. Evidenciam também que o tempo necessário para reparar um equipamento que apresentou uma falha é bastante considerável.

É possível concluir ainda que, da forma como os dados estão estruturados, não é possível

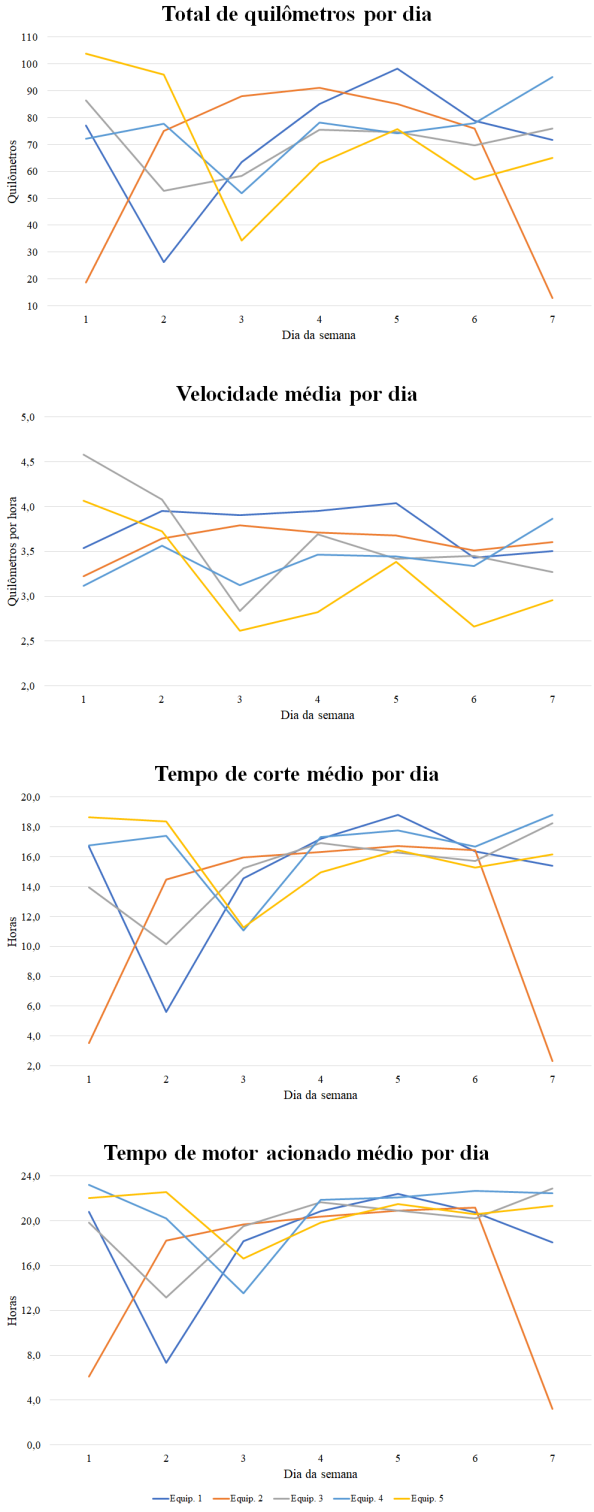



Figura 19 – Histórico da telemetria na semana exemplo.

Tabela 8 – Comportamento diário das características de telemetria

| # | Característica | Amplitude | Média | Desvio padrão |
|---|---|---|---|---|
| 3 | Duração da operação | 24,00 | 17,91 | 6,81 |
| 5 | Quilômetros rodados | 143,28 | 64,10 | 33,52 |
| 6 | Velocidade média | 8,09 | 3,54 | 1,15 |
| 7 | RPM do motor | 3.581 | 1.332 | 350 |
| 8 | Marcha à ré | 23,52 | 1,27 | 3,04 |
| 9 | Esteira em reversão | 6,66 | 0,00 | 0,09 |
| 10 | Cortador | 21,75 | 12,51 | 6,03 |
| 11 | Elevador | 19,91 | 9,81 | 5,17 |
| 12 | Motor acionado | 23,57 | 16,06 | 7,03 |
| 13 | Motor ocioso | 20,04 | 3,12 | 1,43 |

construir o modelo que está sendo proposto e que, para seguir adiante, os dados precisam passar por uma etapa de preparação que seja capaz de adequá-los ao algoritmo de predição e classificação, isso porque as perguntas que se quer responder - quando e por qual motivo ocorrerá a próxima falha - não estão relacionadas com os possíveis motivos que geraram tal evento (as condições operacionais).

#### *3.2.1.4 Considerações sobre a qualidade dos dados*

Outra análise relevante realizada foi sobre a qualidade dos dados disponíveis. É possível afirmar que, em sua grande maioria, os dados são bastante consistentes e que as análises geradas a partir deles são confiáveis. Entretanto, como era esperado, para o volume de dados avaliado e para a forma como são coletados - sensores instalados em equipamentos expostos a condições de operação severas, algumas distorções foram identificadas e corrigidas.

O resumo das constatações acerca do conjunto de dados operacionais é o seguinte:

- tamanho do conjunto de dados: 481.083 instâncias;
- velocidade: oito instâncias, ou 0,0029% do total, apresentaram velocidade incoerente (muito alta);
- quilômetros percorridos e RPM: não foram identificada inconsistências;
- quilômetros percorridos, velocidade, RPM: estão presentes em 100% das instâncias;
- em 1.260 instâncias, ou 0,26% do total, esteira em reversão, corte de base, elevador, motor ligado e motor ocioso estavam com valor nulo.
- o valor 0 (zero) é válido e esperado;
- todas as características de tempo que apresentaram medição estavam coerentes com a duração da operação.

A avaliação da qualidade dos dados das 4.750 instâncias do conjunto de dados de manutenção de equipamentos não evidenciou falhas tampouco necessidades de correção. Cabe ressaltar, porém, que por se tratarem de dados digitados, pode haver alguma defasagem ou distorção entre a ocorrência do evento e seu registro no sistema. Assumiu-se, todavia, que esta defasagem é irrelevante para o resultado do modelo proposto.

## 3.3 Preparação dos dados

Conforme explanado na seção 3.2.1, foram selecionados 15 equipamentos (30% do total) para o desenvolvimento da pesquisa. Das características disponíveis nos conjuntos de dados apresentados nas tabelas 5 e 6, foram descartadas as características 3 (código do problema), 5 (código do detalhe do problema), 6 (descrição detalhada do problema) e 7 (equipe responsável pela manutenção) do primeiro e a característica 4 (atividade que o equipamento realizava no momento da medição) do segundo.

A característica "operador do equipamento" foi desconsiderada previamente pois cada equipamento é operado sempre pelos mesmos operadores.

A característica 9 (horas de operação da esteira em reversão no intervalo de apuração da característica) do conjunto de dados operacionais (tabela 6), mesmo tendo operado por apenas 28 horas em todo o período monitorado (0,0045% do tempo total), foi considerada para o modelo.

Para os problemas de qualidade de dados abordados na seção 3.2.1.4 foram definidas as seguintes ações:

- instâncias com característica velocidade incoerente: o valor da característica foi ajustado para 20 km/h, que é a velocidade máxima aceitável;

- instâncias onde as características esteira em reversão, corte de base, elevador, motor ligado e motor ocioso estavam com valor nulo: foram mantidos pois correspondem a aproximadamente 80 horas de operação, sendo que destas somente 8 horas são operação consideradas produtivas, todavia seus valores foram considerados zero.

Outro problema a ser resolvido é sobre a linha do tempo representada pela característica 2 - data / hora do registro do valor das características do conjunto de dados operacionais (tabela 6). Os intervalos de geração desta característica não são uniformes, pois estão sujeitas a eventos (gatilhos) que podem encerrar um ciclo de coleta de dados e iniciar outro. O limite desse ciclo de coleta é de sessenta minutos, porém pode variar desde um segundo até esse limite, conforme pode ser avaliado no histograma apresentado na figura 20.

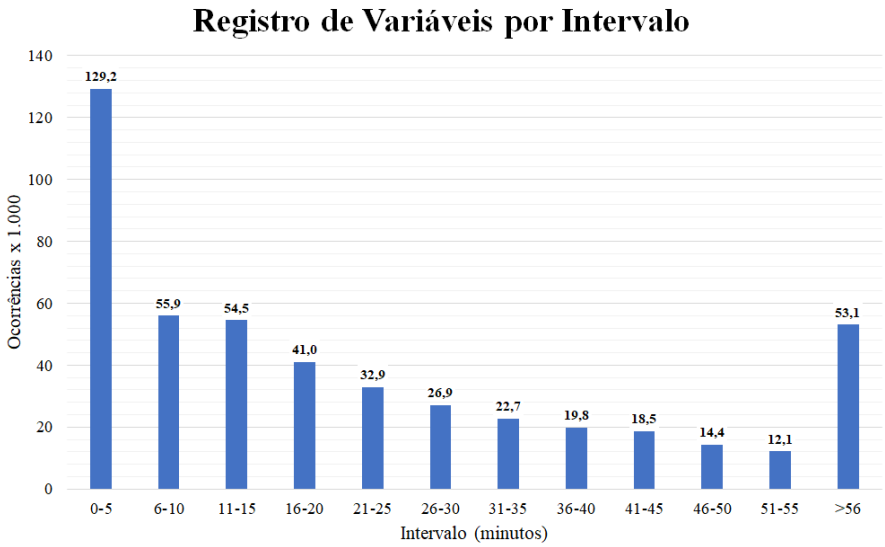



Figura 20 – Histograma de intervalos de coleta de características operacionais.

### 3.3.1 Conjunto de dados A - previsão da causa da próxima falha

Baseado na constatação prévia que, da forma como estão os conjuntos de dados, não há como utilizá-los no modelo proposto, foi adotada a construção de um conjunto de dados que fosse adequado aos objetivos que se está buscando. A estrutura inicial projetada consiste em haver variáveis dependentes e independentes em uma mesma instância, situação que requereu que as características operacionais e os dados sobre manutenção fossem combinados em um único conjunto de dados.

A estrutura de dados final com os dados mesclados está definida na tabela 9.

Este novo conjunto de dados é composto de vinte e seis características, sendo que a maior parte delas provém do conjunto de dados operacionais, porém há diferenças substanciais entre eles, conforme listado a seguir:

- a característica *1-Data / hora do registro do valor das características*, respeita intervalos regulares de tempo (horas, turnos, dias, semana etc.), conforme necessidade do modelo. Está no formato de número real, onde a data é representada pela parte inteira do número e as horas pela parte decimal. No caso da data, o número 1 representa 01/01/0001, o 2 representa 02/01/0001 e assim sucessivamente. Quanto às horas, estas são representadas por frações de 24, por exemplo, 1 hora equivale a 0,04167, 2 horas equivale a 0,0833, 12 horas equivale a 0,5;

- as características *2-Totalkm, 4-WorkTime, 6-AvgVeloc, 7-AvgRPM, 8-ReverseTime, 10-ReverseWork, 12-CropTime, 14-ElevatorTime, 16-EngineTime* e *18-IdleTime* são as características operacionais conhecidas, porém seus valores estão respeitando o intervalo de tempo da característica 1-Date, e não os intervalos de geração mencionados previamente;

Tabela 9 – Características do conjunto de dados A

| Característica | Descrição | Tipo de Dado |
|---|---|---|
| 1-Date | Data / hora do registro do valor das características | Núm. Real |
| 2-Totalkm | Quilômetros rodados no intervalo de medição | Núm. Real |
| 3-TotalkmA | Total de quilômetros rodados desde a última falha | Núm. Real |
| 4-WorkTime | Tempo de operação no intervalo de medição | Núm. Int. |
| 5-WorkTimeA | Tempo total de operação desde a última falha | Núm. Int. |
| 6-AvgVeloc | Velocidade média apurada no intervalo de medição | Núm. Real |
| 7-AvgRPM | RPM médio apurado no intervalo de medição | Núm. Int. |
| 8-ReverseTime | Horas de operação em marcha ré no intervalo de medição | Núm. Real |
| 9-ReverseTimeA | Total de horas de operação em marcha ré desde a última falha | Núm. Real |
| 10-ReverseWork | Horas de operação da esteira em reversão no intervalo de medição | Núm. Real |
| 11-ReverseWorkA | Total de horas de operação da esteira em reversão desde a última falha | Núm. Real |
| 12-CropTime | Horas de operação do cortador no intervalo de medição | Núm. Real |
| 13-CropTimeA | Total de horas de operação do cortador desde a última falha | Núm. Real |
| 14-ElevatorTime | Horas de operação do elevador no intervalo de medição | Núm. Real |
| 15-ElevatorTimeA | Total de horas de operação do elevador desde a última falha | Núm. Real |
| 16-EngineTime | Horas de motor acionado no intervalo de medição | Núm. Real |
| 17-EngineTimeA | Total de horas de motor acionado desde a última falha | Núm. Real |
| 18-IdleTime | Horas de motor ocioso no intervalo de medição | Núm. Real |
| 19-IdleTimeA | Total de horas de motor ocioso desde a última falha | Núm. Real |
| 20-Month | Mês de ocorrência da medição | Núm. Int. |
| 21-WorkShift | Turno de trabalho da medição | Núm. Int. |
| 22-TimeFLastPrev | Tempo (h) decorrido desde a última manutenção preventiva | Núm. Real |
| 23-TimeFLastFail | Tempo (h) decorrido desde a última falha | Núm. Int. |
| 24-LastFailCause | Causa da última falha | Alfanumérico |
| 25-FailType | Tipo de falha no ciclo | Alfanumérico |
| 26-NextFailCause | Causa da próxima falha | Alfanumérico |

- as características *3-TotalkmA, 5-WorkTimeA, 9-ReverseTimeA, 11-ReverseWorkA, 13-CropTimeA, 15-ElevatorTimeA, 17-EngineTimeA* e *19-IdleTimeA* representam as mesmas informações das características *2-Totalkm, 4-WorkTime, 8-ReverseTime, 10-ReverseWork, 12-CropTime, 14-ElevatorTime, 16-EngineTime* e *18-IdleTime* porém apuradas desde a última falha, não somente no intervalo de medição;

- características novas:

  – a característica *20-Month* é definida a partir da característica *1-Date*, tem a finalidade de representar condições como clima e estágio de desenvolvimento da planta colhida, uma vez que estas informações não estão disponíveis de outra forma;

  – a característica *21-WorkShift* (turno de trabalho) é definida a partir da característica 1-Date, tem por finalidade representar as diferentes condições de operação, como temperatura e visibilidade, por exemplo, durante o dia. Os turnos compreendem os seguintes horários: turno 1 entre 00:00 e 07:59, turno 2 entre 08:00 e 15:59 e turno 3 entre 16:00 e 23:59;

  – as características *22-TimeFLastPrev* e *23-TimeFLastFail* são definidas a partir da característica *1-Date* e do conjunto de dados de manutenção;

  – as características *24-LastFailCause* e *26-NextFailCause* têm seu conteúdo determinado pelos principais tipos de falha (motor diesel, elevador, divisor de linha, corte de base, picador, sistema rodante, rolos, elétrico, transmissão, extrator primário e outros), ou seja, estão na forma de classes;

  – a característica *25-FailType* indica se ocorreu alguma falha no ciclo e, tendo ocorrido, qual o tipo de falha. Os valores possíveis são os mesmos citados no item anterior. Esta variável foi inserida inicialmente com o intuito de classificar um turno com o tipo de uma falha ocorrida ou com um valor que indicasse que o equipamento estava em operação. Os resultados obtidos pelo modelo de classificação não foram satisfatórios, pois a quantidade de registros com FailType indicando ausência de falha (85% do total) influenciou negativamente na determinação de classes. Desta forma, a decisão foi manter esta variável utilizando-a apenas como variável independente para apoiar na classificação do próximo motivo de falha.

Com a estruturação apresentada na tabela 9 acredita-se ter obtido o conjunto de dados necessário à construção do modelo proposto, isso porque a variável dependente “26 - Causa da próxima falha” está, desta forma, relacionada às variáveis independentes.

Cabe ressaltar que o conjunto de dados gerado não é um dos conjuntos inicialmente disponíveis com adição de novos campos, mas sim um conjunto completamente novo gerado a partir dos dois conjuntos de dados originais em função do período determinado na característica 1-Date.

O conjunto de dados final foi gerado com o parâmetro de intervalo de tempo de oito horas, que corresponde a um turno de trabalho. Exitem, portanto, três instâncias por dia por equipamento. O resultado deste preparo foi um conjunto de dados com 29.514 instâncias. Destas, foram desconsideradas ainda 9.444 instâncias que correspondem a períodos de inatividade do equipamento ou finais de safra, em que a próxima falha ocorreu apenas na safra seguinte, o que não representa uma situação real de operação. Desta forma, o conjunto de dados final possui 20.070 ocorrências.

As estatísticas básicas do conjunto de dados A encontram-se na tabela 10. Não há valores disponíveis paras as características 19, 20 23, 24 e 25 por se tratarem de valores categóricos, por isso estas estão representadas na figura 21.

Tabela 10 – Estatísticas básicas do conjunto de dados A

| Característica | Mínimo | Máximo | Média | Desvio padrão |
|---|---|---|---|---|
| 1–Date | — | — | — | — |
| 2–Totalkm | 0,000 | 40,000 | 23,691 | 11,036 |
| 3–TotalkmA | 0,000 | 2.896,731 | 204,590 | 232,709 |
| 4–WorkTime | 0,002 | 8,800 | 6,611 | 1,989 |
| 5–WorkTimeA | 0,011 | 684,753 | 56,379 | 60,258 |
| 6–AvgVeloc | 0,000 | 17,557 | 4,347 | 1,205 |
| 7–AvgRPM | 0,081 | 3.073,836 | 1.381,093 | 351,575 |
| 8–ReverseTime | 0,000 | 8,800 | 0,475 | 1,104 |
| 9-ReverseTimeA | 0,000 | 237,851 | 4,788 | 15,990 |
| 10-ReverseWork | 0,000 | 1,519 | 0,000 | 0,014 |
| 11-ReverseWorkA | 0,000 | 2,013 | 0,002 | 0,021 |
| 12-CropTime | 0,000 | 8,800 | 4,674 | 1,989 |
| 13-CropTimeA | 0,000 | 519,789 | 39,516 | 43,233 |
| 14-ElevatorTime | 0,000 | 8,800 | 3,666 | 1,772 |
| 15-ElevatorTimeA | 0,000 | 433,989 | 31,226 | 35,424 |
| 16-EngineTime | 0,000 | 8,800 | 5,979 | 2,246 |
| 17-EngineTimeA | 0,007 | 647,362 | 50,497 | 54,556 |
| 18-IdleTime | 0,000 | 8,270 | 1,151 | 0,561 |
| 19-IdleTimeA | 0,005 | 114,926 | 9,666 | 10,617 |
| 20-Month | — | — | — | — |
| 21-WorkShift | — | — | — | — |
| 22-TimeFromLastPrev | 4,000 | 168,000 | 98,448 | 50,931 |
| 23-TimeFromLastFail | 4,000 | 336,000 | 63,979 | 75,663 |
| 24-LastFailCause | — | — | — | — |
| 25-FailType | — | — | — | — |

A figura 22 foi elaborada a fim de promover o entendimento do novo conjunto de dados. Essa figura mostra um mês de operação de todos os equipamentos avaliados, turno a turno. A cronologia (dias e turnos) dos eventos está representada na parte superior da figura; os equipamentos estão identificados na lateral esquerda; as células preenchidas com um ponto

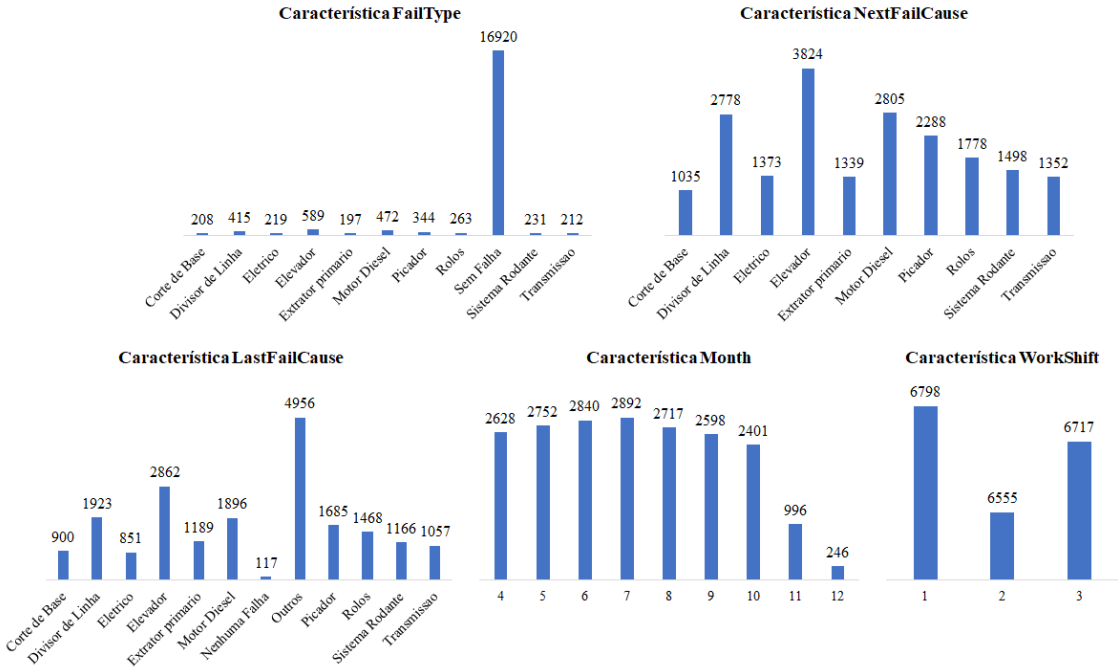



Figura 21 – Conjunto de dados A - estatística básica das variáveis categóricas.

vermelho indicam a ocorrência do evento de falha naquele turno; enquanto as células sem marcação indicam equipamentos em operação.

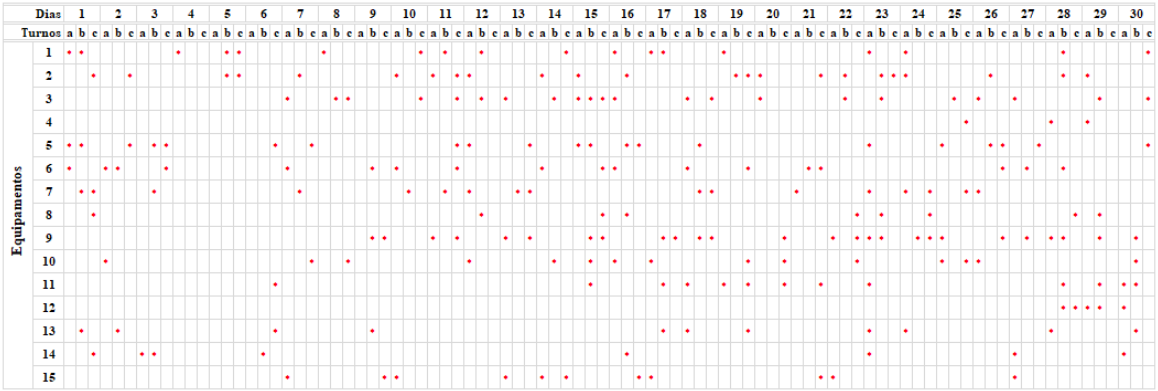



Figura 22 – Cronologia de falhas em um mês de operação.

Além dessa representação, também foram analisados os comportamentos das variáveis neste novo conjunto de dados, que resultaram nos gráficos das figuras 23, 24, 25 e 26.

Os gráficos da figura 23 representam as dispersões das características que representam os tempos de operação acumulados no turno de trabalho (eixo X) em relação às causas de falhas (eixo Y). A análise visual destas dispersões permite concluir que todas as características exibidas nos gráficos apresentam o mesmo comportamento com relação aos tipos de falha, não sendo possível afirmar que algum tipo de falha seja mais influenciado por determinada característica.

Todavia é possível identificar, ainda que com muita sutileza, que as características

relacionam-se de forma diferente com as falhas. É possível identificar quatro casos: (i) maior ocorrência de falha quanto menor o valor da variável (ReverseWork e IdleTime); (ii) maior ocorrência de falha quanto maior o valor da variável (TotalKm e WorkTime); (iii) maior ocorrência de falha nos valores intermediários (CropTime e ElevatorTime); e (iv) com maior ocorrência de falha quando a variável está próximo das extremidades (ReverseTime e EngineTime).

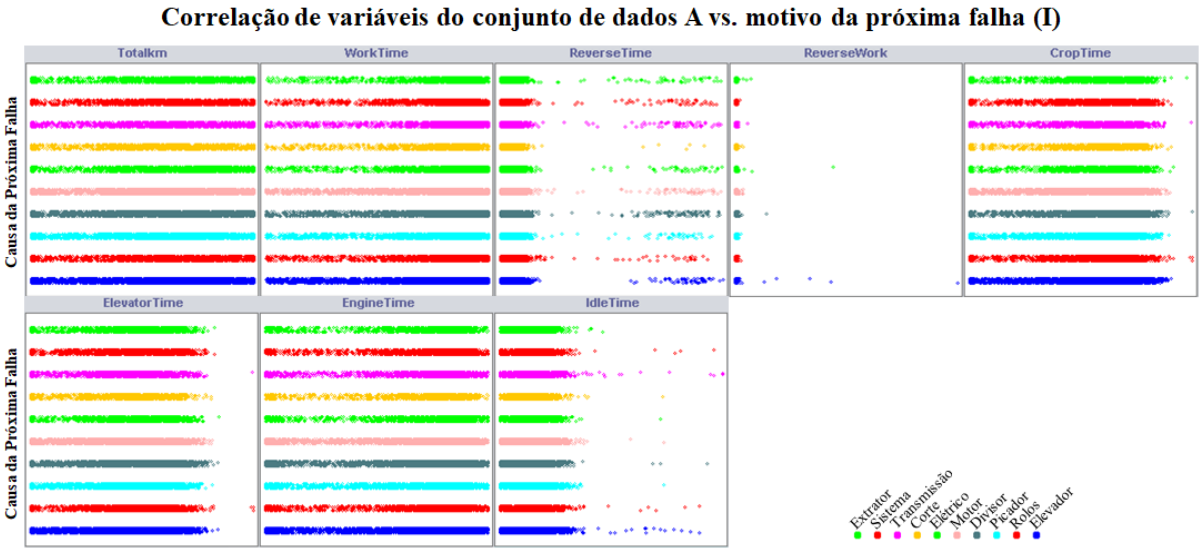



Figura 23 – Correlação do primeiro grupo de variáveis do conjunto de dados A.

Dando continuidade à análise de dispersões, os gráficos da figura 24 representam os tempos de operação acumulados desde a última falha (eixo X) em relação às causas de falhas (eixo Y). Com este conjunto de variáveis foi possível concluir que (i) a falha por motivo Rolos ocorre com os maiores tempos de operação, exceto no caso da característica ReverseTimeA; (ii) as falhas picador, divisor, motor, elétrico e transmissão são menos influenciadas pela característica ReverseTimeA; e (iii) que a falha corte ocorre com os menores tempos acumulados, para todas as características.

No último conjunto de gráficos de dispersões (figura 25) são representadas as características não relacionadas a horas de operação. As causas de falhas continuam sendo representadas no eixo Y. Neste caso identificou-se: (i) maior ocorrência de falha quanto menor o valor da variável para as características AvgVeloc, AvgRPM e TimeFromLastFail; (ii) menor ocorrência de falhas logo nas primeiras horas após a manutenção preventiva; (iii) distribuição semelhante quanto à ocorrência em Month, WorkShift e LastFailCause; (iv) em Month, a menor ocorrência de falhas no último mês está relacionada à menor quantidade de dados.

E finalmente, a análise visual das curvas de distribuição dos valores das características, representadas na figura 26, permitiu evidenciar ao menos quatro agrupamentos de distribuições: o primeiro formado pelas características AvgVeloc, IdleTime e AvgRPM; o segundo por Totalkm, WorkTime, CropTime, EngineTime, TimeFromLastPrev; o terceiro por TotalkmA; WorkTimeA, CropTimeA, ElevatorTimeA, EngineTimeA, IdelTimeA; e o quarto e último por ReverseTime,

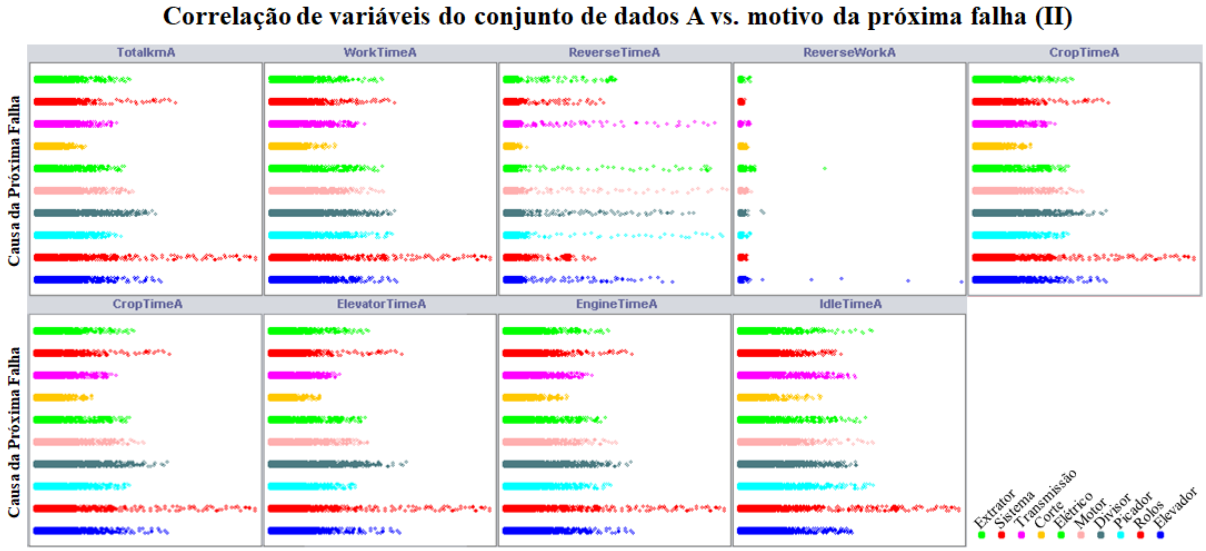



Figura 24 – Correlação do segundo grupo de variáveis do conjunto de dados A.

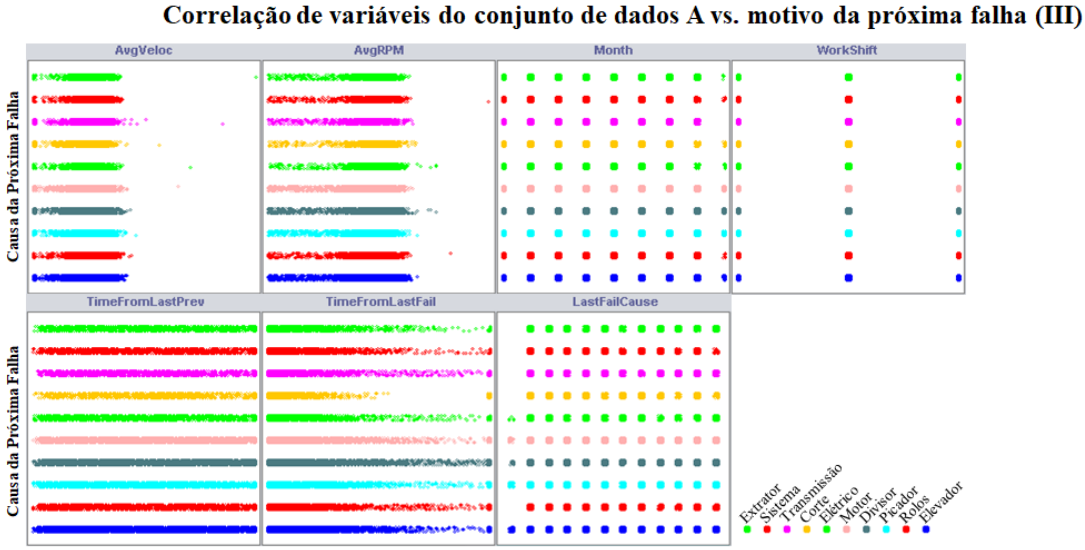



Figura 25 – Correlação do terceiro grupo de variáveis do conjunto de dados A.

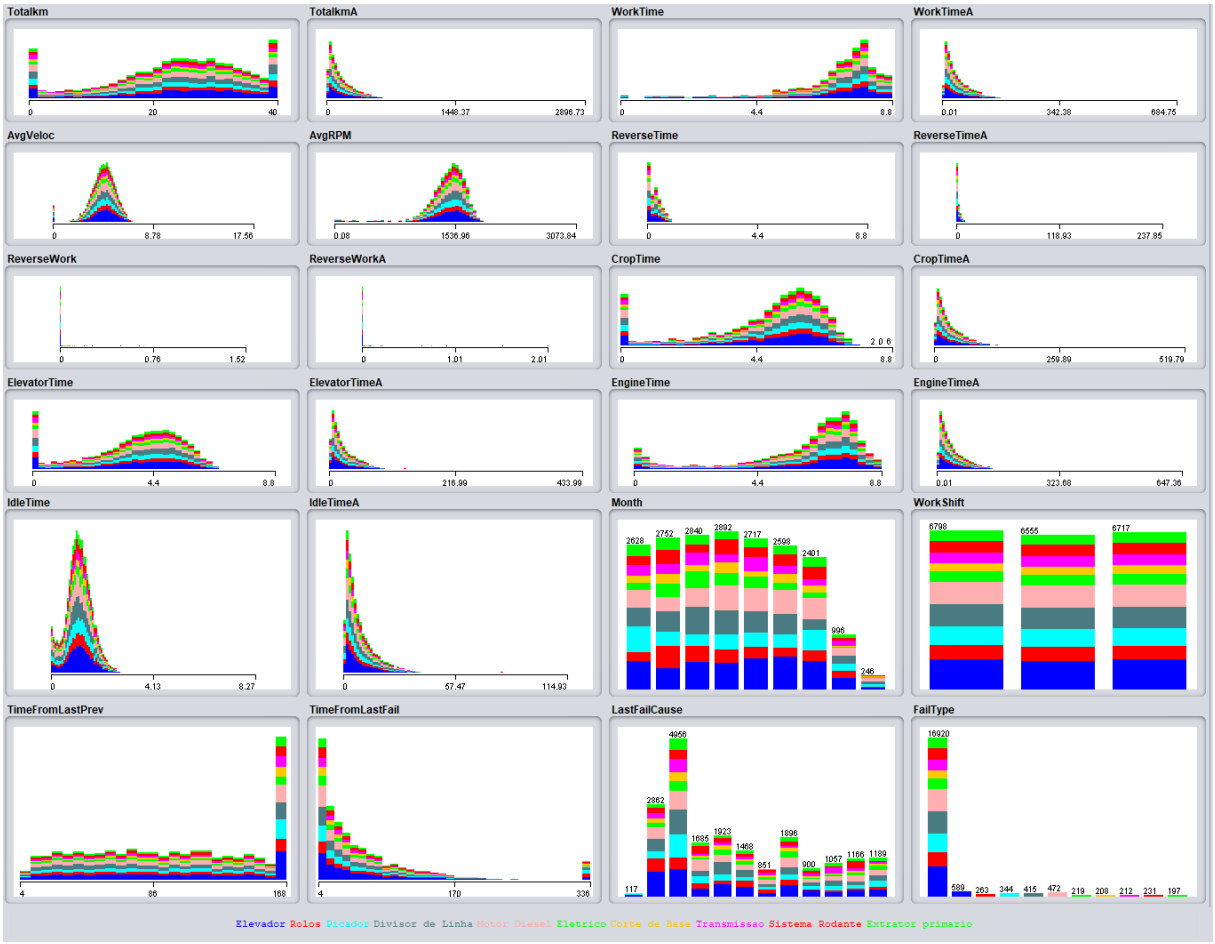



Figura 26 – Distribuição das variáveis do conjunto de dados A.

ReverseTimeA, ReverseWork, ReverseWorkA, TimeFromLastFail. Às características Month, WorkShift, LastFailCause e FailType este conceito não se aplica, uma vez que estamos diante de valores categóricos. Estas distribuições corroboram a análise de dispersões das características com as causas das falhas.

### 3.3.2 Conjunto de dados B - horas de operação

A estrutura do conjunto de dados para previsão do tempo até a próxima falha exigiu adaptações em relação àquele detalhado na tabela 9, gerando assim um conjunto diferente.

A proposta foi gerar instâncias contendo apenas as falhas e não mais as medições turno a turno. Assim sendo, foram mantidas as variáveis operacionais com valores acumuladas no momento da ocorrência da falha e desprezadas as informações relacionadas ao turno (características 2, 4, 8, 10, 12, 14, 16 e 18). As características novas (20 a 24) também foram mantidas. A característica 25-FailType foi removida pois o modelo de predição a ser construído a partir deste conjunto de dados deve prever o tempo de operação entre a última falha ocorrida e a próxima falha prevista; além disso, quando a previsão é realizada, o equipamento está em operação, e não em manutenção. Ademais, a variável resposta do conjunto de dados A (NextFailType) se

torna uma das características utilizadas para predição do tempo de operação, fato que torna este conjunto de dados dependente do resultado da predição da próxima falha.

A variável dependente deste conjunto é a característica "3-WorkTimeA Tempo total de operação desde a última falha". A tabela 11 lista a estrutura completa do segundo conjunto de dados, que será chamado de conjunto de dados B.

Tabela 11 – Características do conjunto de dados B

| Característica | Descrição | Tipo de Dado |
|---|---|---|
| 1–Date | Data / hora do registro do valor das características | Núm. Real |
| 2-TotalkmA | Total de quilômetros rodados desde a última falha | Núm. Real |
| 3-WorkTimeA | Tempo total de operação desde a última falha | Núm. Int. |
| 4-AvgVeloc | Velocidade média apurada desde a última falha | Núm. Real |
| 5-AvgRPM | RPM médio apurado desde a última falha | Núm. Int. |
| 6-ReverseTimeA | Total de horas de operação em marcha ré desde a última falha | Núm. Real |
| 7-ReverseWorkA | Total de horas de operação da esteira em reversão desde a última falha | Núm. Real |
| 8-CropTimeA | Total de horas de operação do cortador desde a última falha | Núm. Real |
| 9-ElevatorTimeA | Total de horas de operação do elevador desde a última falha | Núm. Real |
| 10-EngineTimeA | Total de horas de motor acionado desde a última falha | Núm. Real |
| 11-IdleTimeA | Total de horas de motor ocioso desde a última falha | Núm. Real |
| 12-Month | Mês de ocorrência da medição | Núm. Int. |
| 13-WorkShift | Turno de trabalho da medição | Núm. Int. |
| 14-TimeFromLastPrev | Tempo (h) decorrido desde a última manutenção preventiva | Núm. Real |
| 15-TimeFromLastFail | Tempo (h) decorrido desde a última falha | Núm. Int. |
| 16-LastFailCause | Causa da última falha | Alfanumérico |
| 17-NextFailCause | Causa da próxima falha | Alfanumérico |

As mesmas considerações sobre os registros inconsistentes aplicadas ao primeiro conjunto de dados são válidas para este. O resultado foi um conjunto de dados com 3.150 instâncias.

As estatísticas básicas do conjunto de dados B encontram-se na tabela 12. Não há valores apresentados paras as características 12, 13, 16 e 17 por se tratarem de valores categóricos, por isso estão representadas na figura 27.

Os gráficos da figura 28 exibem a correlação entre as características deste novo conjunto de dados, donde é possível identificar forte correlação entre a característica 3-WorkTimeA e pelo menos mais seis características do conjunto: 2-TotalkmA, 8-CropTimeA, 9-ElevatorTimeA,

Tabela 12 – Estatísticas básicas do conjunto de dados B

| Característica | Mínimo | Máximo | Média | Desvio padrão |
|---|---|---|---|---|
| 1–Date | — | — | — | — |
| 2-TotalkmA | 0,000 | 2.896,731 | 162,942 | 191,508 |
| 3-WorkTimeA | 0,234 | 684,753 | 45,127 | 49,592 |
| 4-AvgVeloc | 0,000 | 162,942 | 4,343 | 3,038 |
| 5-AvgRPM | 13,650 | 2.218,506 | 1.409,458 | 292,048 |
| 6-ReverseTimeA | 0,000 | 237,851 | 3,246 | 12,709 |
| 7-ReverseWorkA | 0,000 | 1.409,458 | 0,451 | 25,125 |
| 8-CropTimeA | 0,000 | 519,789 | 32,019 | 36,169 |
| 9-ElevatorTimeA | 0,000 | 433,989 | 25,029 | 29,043 |
| 10-EngineTimeA | 0,101 | 647,362 | 40,822 | 45,029 |
| 11-IdleTimeA | 0,073 | 114,926 | 7,851 | 8,624 |
| 12-Month | — | — | — | — |
| 13-WorkShift | — | — | — | — |
| 14-TimeFromLastPrev | 4,000 | 168,000 | 96,919 | 49,320 |
| 15-TimeFromLastFail | 4,000 | 336,000 | 48,369 | 59,683 |
| 16-LastFailCause | — | — | — | — |
| 17-NextFailCause | — | — | — | — |

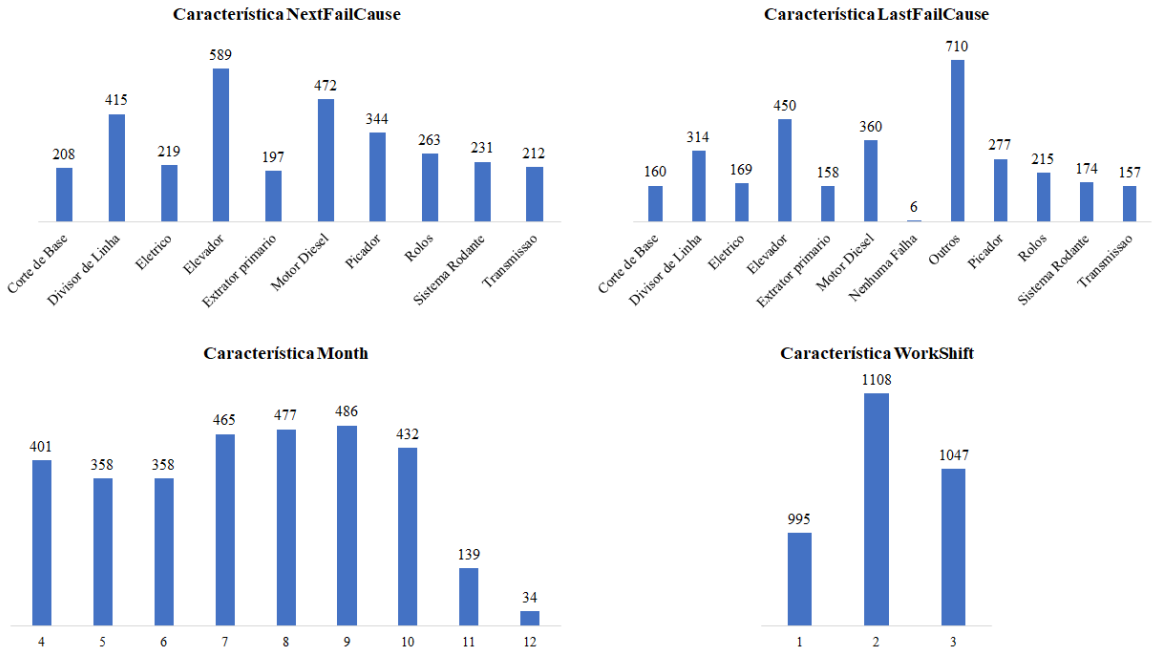



Figura 27 – Conjunto de dados B - estatística básica das variáveis categóricas.

10-EngineTimeA, 11-IdleTimeA e 15-TimeFromLastFail.

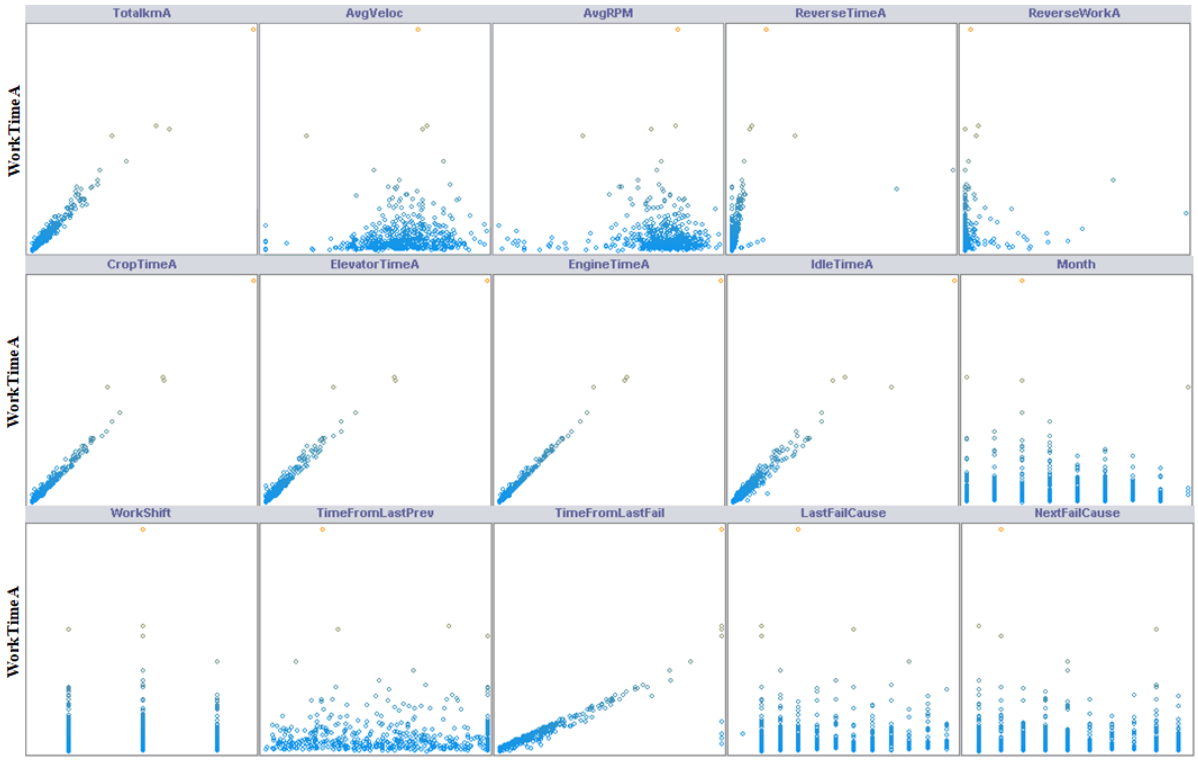



Figura 28 – Correlação entre características das variáveis conjunto de dados B.

A análise visual das curvas de distribuição dos valores das características, representadas na figura 29, permitiu evidenciar ao menos três agrupamentos de distribuições: o primeiro formado pelas características TotalkmA, ReverseTimeA, ReverseWorkA e TimeFromLastFail; o segundo formado por AvgVeloc e AvgRPM; e o terceiro por CropTimeA, ElevatorTimeA, EngineTimeA, IdleTimeA e WorkTimeA. Das características restantes, Month, WorkShift, LastFailCause e NextFailCause são categóricas e TimeFromLastPrev possui distribuição específica.

Foram apresentados, assim, os dois conjuntos de dados gerados para utilização na elaboração dos modelos apresentado no capítulo 4: o primeiro para previsão do motivo da próxima falha, discutido na seção 3.3.1, e o segundo para previsão de quando ocorrerá a próxima falha prevista, discutido nesta seção.

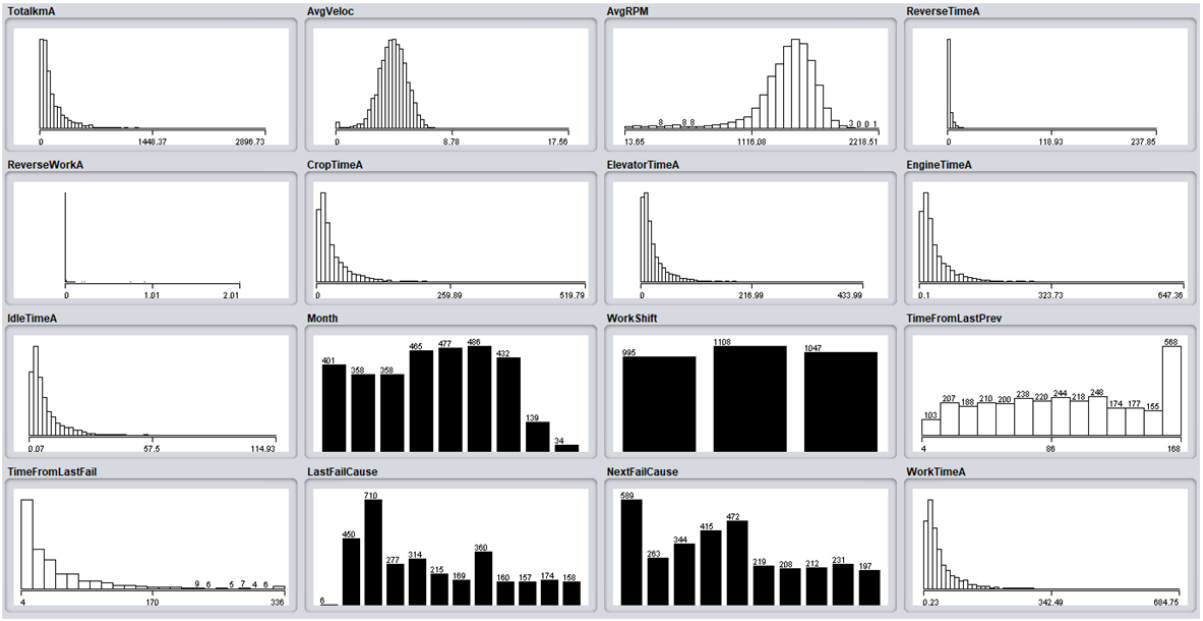



Figura 29 – Distribuição das variáveis do conjunto de dados B.

CAPÍTULO

# 4

# RESULTADOS DOS MODELOS DE CLASSIFICAÇÃO E REGRESSÃO

Algumas abordagens para a solução foram avaliadas com o intuito de atingir-se o objetivo proposto – determinar o momento e a causa da falha. Destas, a **segmentação**, ou **clusterização** (PROVOST; FAWCETT, 2013), não foi escolhida pois não estão sendo buscados subgrupos ou classes de equipamentos ou manutenções que compartilham características semelhantes. Essas informações estão suficientemente detalhadas e determinadas nos conjuntos de dados disponíveis, não identificando-se ganhos significativos no desenvolvimento de modelos que adotem tal abordagem.

Descartou-se também abordagens que identificam **associações** ou **dependências entre as instâncias** dos conjuntos de dados (CHAPMAN *et al.*, 2000), pois o resultado seriam constatações obtidas através da relação de múltiplas instâncias, porém estas também são respostas não alinhadas aos objetivos.

A avaliação seguinte foi sobre **regressão** (CHAPMAN *et al.*, 2000). Esta técnica é capaz de responder à pergunta "quando, ou em quanto tempo, ocorrerá a próxima falha?", porém não é a mais adequada para identificar o motivo da mesma, uma vez que a resposta esperada, neste caso, é um valor entre um conjunto de valores possíveis.

Para atender o objetivo de definição da causa da próxima falha optou-se então pela abordagem de **classificação** (PROVOST; FAWCETT, 2013). A capacidade de atribuir classes a instâncias não conhecidas de um conjunto de dados, baseado em modelos que avaliam instâncias previamente conhecidas e classificadas, atendem plenamente ao que se está buscando neste caso.

Quanto às técnicas existentes para desenvolvimento dos modelos a opção foi pelo emprego das técnicas Zero R, regressão linear, redes neurais MLP e florestas aleatórias.

Para validação da capacidade de previsão dos modelos foi adotada a técnica *k-fold cross validation*, sendo que sua eficácia fora avaliada, no caso dos modelos de classificação, pela

acurácia das predições geradas comparadas à classe real das instâncias, pela matriz de confusão e pela métrica *Kappa*, ambas com a finalidade de avaliação da acurácia por classe prevista. Para modelos de regressão foram adotadas as métricas coeficiente de correlação, erro médio absoluto, raiz quadrada do erro médio, erro médio relativo e raiz quadrada do erro relativo.

A rotina para geração dos conjuntos de dados A e B, conforme apresentado na seção 3.3, foi implementada através da linguagem de programação Python, versão 3.7.6, através do ambiente de desenvolvimento Spyder, versão 4.1.1, disponível na distribuição Anaconda Navigator, versão 1.9.12.

Foram utilizadas na implementação as seguintes bibliotecas Python: CSV, SYS, Numpy, Statistisc, Datetime e Pandas.

As características *24 - Causa da última falha, 25 - Tipo de falha no ciclo* e *26 - Causa da próxima falha* do conjunto de dados representado na tabela 9, bem como as características *16 - Causa da última falha* e *17 - Causa da próxima falha*, do conjunto de dados da representado na tabela 11, são características categóricas (representam uma classe) e não numéricas. Este fato requer uma etapa de transformação prévia à sua apresentação à fase de treinamento do modelo, transformação esta que ocorre da seguinte maneira:

1. substitui-se os valores das classes por um número inteiro e sequencial que represente as classes;

2. transforma-se a característica da variável independente em *n* características, onde *n* representa a quantidade de valores distintos que a característica pode assumir; e

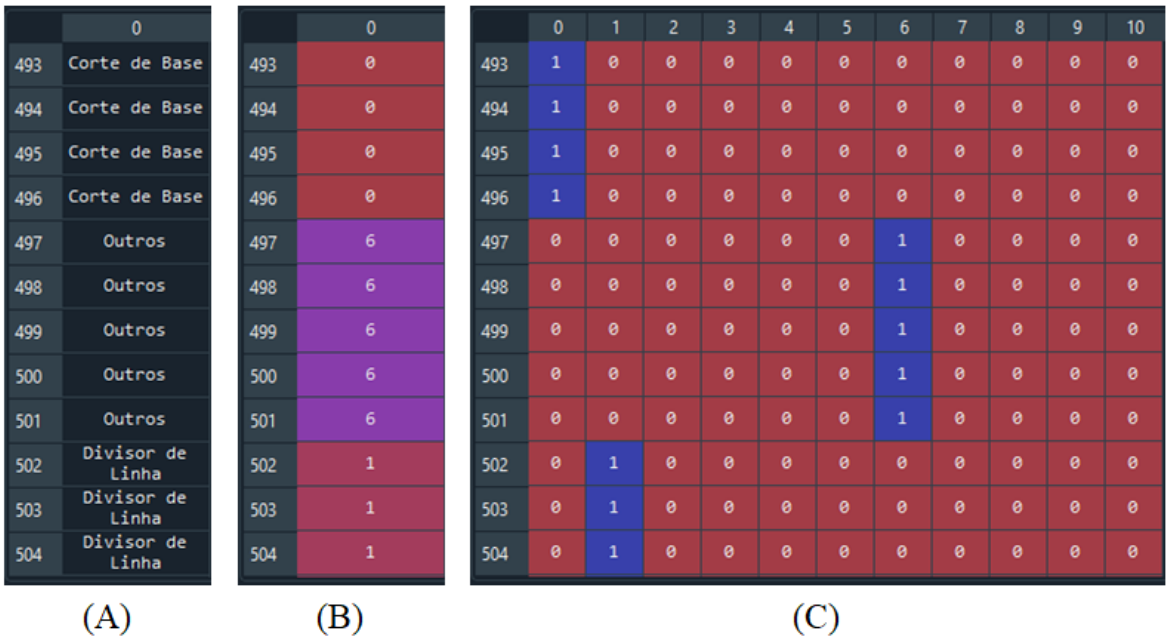
3. uma vez adicionadas as características, a estas serão atribuídos o valor 1, caso a característica corresponda à classe da instância, ou 0, caso contrário.

No Python esta transformação será realizada utilizando-se a técnica Encoder-Decoder, implementada pela classe *LabelEncoder* da biblioteca Scikit-learn. Quando da utilização do WEKA, o próprio *software* encarrega-se desta transformação.

> *LabelEncoder is a utility class to help normalize labels such that they contain only values between 0 and Nclasses-1. (...) It can also be used to transform non-numerical labels (as long as they are hashable and comparable) to numerical labels. (SCIKIT LEARN, 2019)*

As etapas deste processo estão representadas na figura 30, onde (A) apresenta a classe em seu estado original e (B) e (C) os estados após cada etapa de transformação.

Para garantir os resultados esperados optou-se pela construção de dois modelos: o primeiro, que será chamado Modelo A, para determinação da causa da próxima falha e o



| | 0 |
|---|---|
| 493 | Corte de Base |
| 494 | Corte de Base |
| 495 | Corte de Base |
| 496 | Corte de Base |
| 497 | Outros |
| 498 | Outros |
| 499 | Outros |
| 500 | Outros |
| 501 | Outros |
| 502 | Divisor de Linha |
| 503 | Divisor de Linha |
| 504 | Divisor de Linha |

(A)

| | 0 |
|---|---|
| 493 | 0 |
| 494 | 0 |
| 495 | 0 |
| 496 | 0 |
| 497 | 6 |
| 498 | 6 |
| 499 | 6 |
| 500 | 6 |
| 501 | 6 |
| 502 | 1 |
| 503 | 1 |
| 504 | 1 |

(B)

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 493 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 494 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 495 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 496 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 497 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 498 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 499 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 500 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 501 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 502 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 503 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 504 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

(C)

Figura 30 – Processo de transformação de variáveis categóricas.

Fonte: Elaborada pelo autor.

segundo, chamado Modelo B, para determinação das horas de operação no próximo ciclo entre falhas. Ambos estarão descritos nas próximas seções.

Quanto à implementação dos modelos, estes foram construídos e aplicados ao conjunto de dados utilizando-se o *workbench* WEKA, detalhado na seção 3.1 e conforme instruções de uso obtidas em Frank, Hall e Witten (2016).

## 4.1 Modelo A - causa da próxima falha

O modelo A foi construído para determinação da causa da próxima falha e utilizou o conjunto de dados definido na tabela 9.

Deste conjunto, a característica 1 - Data / hora do registro do valor das características foi descartada por não ter influência na variável dependente do modelo. As características 2 a 25 são as variáveis independentes do modelo e a característica 26, a dependente.

As técnicas empregadas na construção foram a Zero R, redes neurais MLP e florestas aleatórias.

### 4.1.1 Modelo baseado no algoritmo Zero R

O modelo utilizado como linha de base inicial para comparativo de desempenho com os demais utilizou-se do algoritmo Zero R. A acurácia geral deste modelo foi de 19,05%.

O resumo dos resultados obtidos e as medidas por classe apurados com o emprego deste modelo estão listados, respectivamente, nas tabelas 13 e 14.

Tabela 13 – Resumo dos resultados - modelo Zero R

| | |
|---|---|
| Instâncias classificadas corretamente | 3.824 |
| Instâncias classificadas incorretamente | 16.246 |
| **Total de instâncias** | **20.070** |
| Acurácia | 19,05% |
| Estatística Kappa | 0 |

Tabela 14 – Acurácia por classe - modelo Zero R

| Classe | Taxa P | Taxa FP | Precisão |
|---|---|---|---|
| Elevador | 1,000 | 1,000 | 0,191 |
| Rolos | 0,000 | 0,000 | — |
| Picador | 0,000 | 0,000 | — |
| Divisor de Linha | 0,000 | 0,000 | — |
| Motor Diesel | 0,000 | 0,000 | — |
| Elétrico | 0,000 | 0,000 | — |
| Corte de Base | 0,000 | 0,000 | — |
| Transmissão | 0,000 | 0,000 | — |
| Sistema Rodante | 0,000 | 0,000 | — |
| Extrator primário | 0,000 | 0,000 | — |
| Média Ponderada | 1,000 | 1,000 | 0,191 |

### 4.1.2 Modelo baseado em uma RNA MLP

O segundo modelo testado para predição dos motivos de falha foi uma RNA baseada no algoritmo MLP. A acurácia geral deste modelo foi de 62,96%.

A RNA foi gerada automaticamente pelo WEKA e utilizou os parâmetros decaimento = não, taxa de aprendizagem = 0,3, momentum = 0,2, épocas de treinamento = 500, função de ativação sigmoide e método de otimização Broyden-Fletcher-Goldfarb-Shanno (BFGS), todos definidos automaticamente pelo WEKA. Os valores das características foram normalizados em uma etapa de pré-processamento do modelo. A rede gerada está representada na figura 31 e o resumo dos resultados obtidos, as medidas por classe e a matriz de confusão apurados com o emprego deste modelo estão listados, respectivamente, nas tabelas 15, 16 e 17.

### 4.1.3 Modelo baseado em florestas aleatórias

O último modelo testado foi baseado no algoritmo florestas aleatórias, tendo sido gerado automaticamente pelo WEKA. Utilizou os parâmetros OOB = não, computar relevância de características = sim, tamanho máximo da árvore = não, número máximo de características por

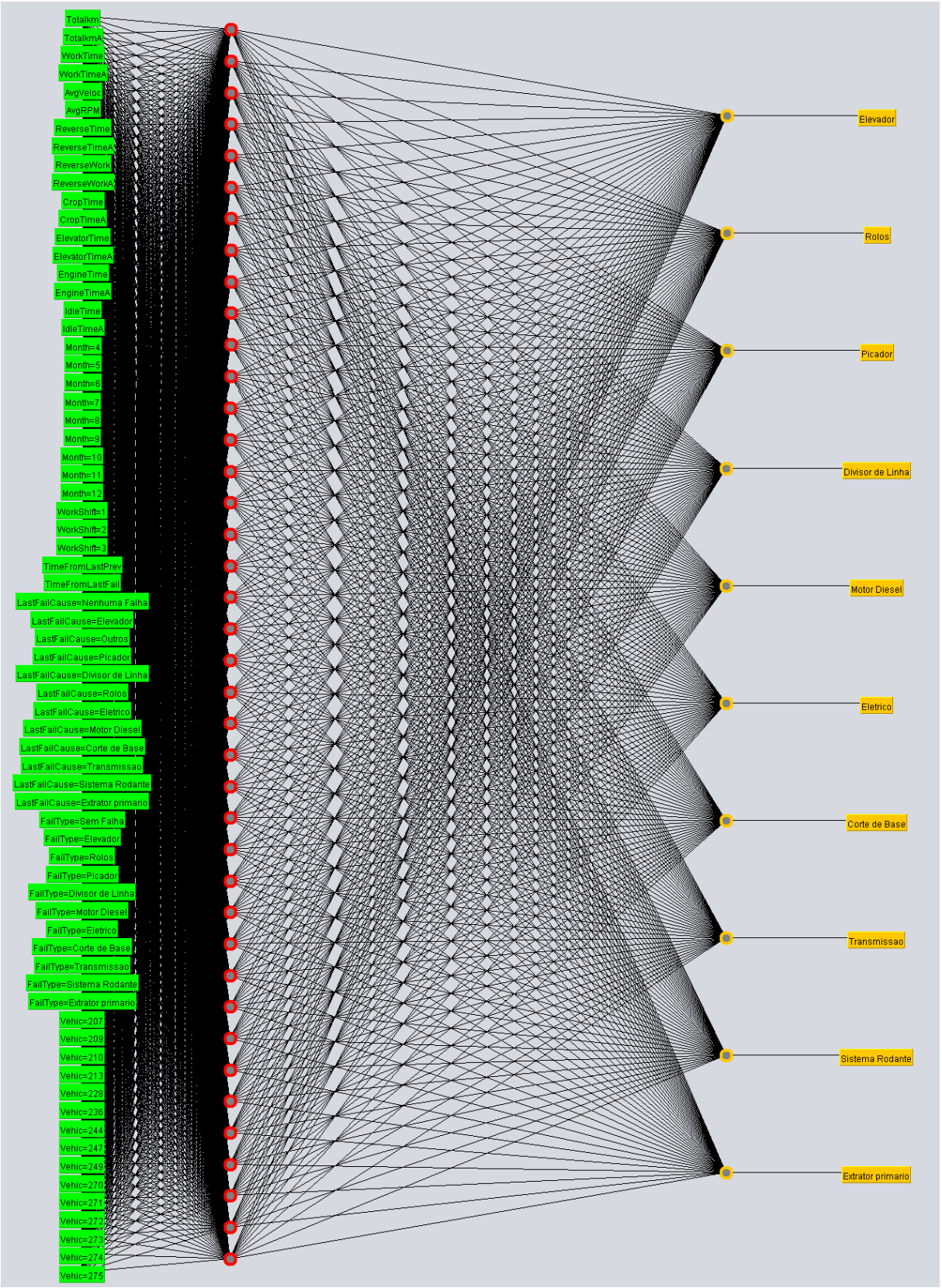



Figura 31 – RNA/MLP construída para o modelo.

Fonte: elaborada pelo autor, conforme modelo gerado pelo *workbench* WEKA.

Tabela 15 – Resumo dos resultados - modelo RNA/MLP

| | |
|---|---|
| Instâncias classificadas corretamente | 12.636 |
| Instâncias classificadas incorretamente | 7.434 |
| **Total de instâncias** | **20.070** |
| Acurácia | 62,96% |
| Estatística Kappa | 0,5763 |

Tabela 16 – Acurácia por classe - modelo RNA/MLP

| Classe | Taxa P | Taxa FP | Precisão |
|---|---|---|---|
| Elevador | 0,758 | 0,120 | 0,598 |
| Rolos | 0,582 | 0,024 | 0,698 |
| Picador | 0,618 | 0,042 | 0,656 |
| Divisor de Linha | 0,684 | 0,062 | 0,638 |
| Motor Diesel | 0,684 | 0,074 | 0,599 |
| Elétrico | 0,492 | 0,020 | 0,640 |
| Corte de Base | 0,407 | 0,014 | 0,604 |
| Transmissão | 0,578 | 0,022 | 0,657 |
| Sistema Rodante | 0,595 | 0,026 | 0,646 |
| Extrator primário | 0,523 | 0,021 | 0,644 |
| Média Ponderada | 0,630 | 0,056 | 0,633 |

Tabela 17 – Matriz de confusão - modelo RNA/MLP

| a | b | c | d | e | f | g | h | i | j | ← Classificado como |
|---|---|---|---|---|---|---|---|---|---|---|
| 2.898 | 69 | 84 | 201 | 256 | 59 | 53 | 66 | 80 | 58 | a = Elevador |
| 230 | 1.035 | 103 | 108 | 123 | 44 | 24 | 28 | 54 | 29 | b = Rolos |
| 285 | 65 | 1.415 | 137 | 151 | 42 | 30 | 72 | 47 | 44 | c = Picador |
| 260 | 62 | 124 | 1.899 | 162 | 56 | 46 | 61 | 54 | 54 | d = Divisor de Linha |
| 329 | 41 | 91 | 117 | 1.920 | 70 | 30 | 58 | 82 | 67 | e = Motor Diesel |
| 177 | 53 | 65 | 131 | 130 | 675 | 27 | 40 | 47 | 28 | f = Elétrico |
| 180 | 27 | 74 | 119 | 90 | 34 | 421 | 23 | 38 | 29 | g = Corte de Base |
| 167 | 30 | 56 | 90 | 115 | 23 | 14 | 781 | 41 | 35 | h = Transmissão |
| 168 | 42 | 79 | 76 | 117 | 18 | 28 | 35 | 892 | 43 | i = Sistema Rodante |
| 149 | 59 | 66 | 98 | 139 | 34 | 24 | 25 | 45 | 700 | j = Extrator primário |

árvore = não definido e número de árvores na floresta = 100, todos definidos automaticamente pelo WEKA. A acurácia geral obtida foi de 82,80%.

O modelo gerado é composto por 100 árvores, cujos tamanhos variam entre 8.358 e 13.299 nós.

A importância das características, baseada na impureza média e na quantidade de nós onde a característica aparece no modelo está na listada na tabela 18. O resumo dos resultados obtidos, as medidas por classe e a matriz de confusão apurados com o emprego deste modelo estão listados, respectivamente, nas tabelas 19, 20 e 21.

Tabela 18 – Importância de atributos - modelo florestas aleatórias para previsão da causa da próxima falha

| # | Impureza Média | Número de Nós | Característica |
|---|---|---|---|
| 1 | 1,31 | 8.263 | Vehic |
| 2 | 1,20 | 10.085 | LastFailCause |
| 3 | 1,13 | 12.766 | Month |
| 4 | 0,73 | 30.126 | Totalkm |
| 5 | 0,73 | 21.954 | WorkTime |
| 6 | 0,71 | 29.651 | TotalkmA |
| 7 | 0,69 | 22.004 | WorkTimeA |
| 8 | 0,69 | 22.754 | AvgVeloc |
| 9 | 0,68 | 19.417 | AvgRPM |
| 10 | 0,68 | 4.653 | FailType |
| 11 | 0,67 | 17.445 | ReverseTime |
| 12 | 0,66 | 5.332 | WorkShift |
| 13 | 0,65 | 18.660 | ReverseTimeA |
| 14 | 0,64 | 13.085 | CropTime |
| 15 | 0,63 | 15.918 | TimeFromLastPrev |
| 16 | 0,63 | 676 | ReverseWork |
| 17 | 0,63 | 6.867 | ReverseWorkA |
| 18 | 0,62 | 12.868 | CropTimeA |
| 19 | 0,62 | 11.721 | ElevatorTime |
| 20 | 0,60 | 11.457 | ElevatorTimeA |
| 21 | 0,60 | 8.422 | EngineTime |
| 22 | 0,59 | 10.070 | IdleTime |
| 23 | 0,58 | 7.504 | EngineTimeA |
| 24 | 0,57 | 9.684 | IdleTimeA |
| 25 | 0,54 | 7.196 | TimeFromLastFail |

### 4.1.4 Florestas aleatórias individualizada por equipamento

A análise da acurácia geral dos modelos, bem como a análise das medidas por classe, demonstram ampla vantagem para o modelo baseado em florestas aleatórias.

Com base nesse resultado, deu-se início a uma nova fase de testes, onde foram criados modelos individualizados para cada equipamento, buscando-se assim maior acurácia das

Tabela 19 – Resumo dos resultados - modelo florestas aleatórias

| | |
|---|---|
| Instâncias classificadas corretamente | 16.617 |
| Instâncias classificadas incorretamente | 3.453 |
| **Total de instâncias** | **20.070** |
| Acurácia | 82,80% |
| Estatística Kappa | 0,8045 |

Tabela 20 – Acurácia por classe - modelo florestas aleatórias

| Classe | Taxa P | Taxa FP | Precisão |
|---|---|---|---|
| Elevador | 0,867 | 0,051 | 0,801 |
| Rolos | 0,819 | 0,013 | 0,857 |
| Picador | 0,837 | 0,020 | 0,842 |
| Divisor de Linha | 0,830 | 0,030 | 0,818 |
| Motor Diesel | 0,842 | 0,032 | 0,812 |
| Elétrico | 0,800 | 0,011 | 0,842 |
| Corte de Base | 0,741 | 0,009 | 0,825 |
| Transmissão | 0,812 | 0,011 | 0,847 |
| Sistema Rodante | 0,805 | 0,012 | 0,842 |
| Extrator primário | 0,814 | 0,009 | 0,864 |
| Média Ponderada | 0,828 | 0,025 | 0,829 |

Tabela 21 – Matriz de confusão - modelo florestas aleatórias

| a | b | c | d | e | f | g | h | i | j | ← Classif. como |
|---|---|---|---|---|---|---|---|---|---|---|
| 3.316 | 34 | 68 | 110 | 118 | 32 | 36 | 39 | 46 | 25 | a = Elevador |
| 79 | 1.457 | 40 | 63 | 47 | 25 | 14 | 10 | 28 | 15 | b = Rolos |
| 126 | 28 | 1.916 | 52 | 56 | 26 | 15 | 22 | 24 | 23 | c = Picador |
| 145 | 47 | 54 | 2.306 | 86 | 36 | 24 | 31 | 18 | 31 | d = Div. Linha |
| 137 | 25 | 44 | 86 | 2.362 | 35 | 14 | 27 | 43 | 32 | e = Motor Diesel |
| 59 | 32 | 27 | 45 | 63 | 1.099 | 15 | 18 | 10 | 5 | f = Elétrico |
| 79 | 20 | 29 | 44 | 33 | 5 | 767 | 17 | 24 | 17 | g = Corte de Base |
| 64 | 12 | 28 | 45 | 36 | 21 | 15 | 1.098 | 18 | 15 | h = Transmissão |
| 82 | 28 | 39 | 25 | 53 | 15 | 18 | 23 | 1.206 | 9 | i = Sist. Rodante |
| 51 | 17 | 31 | 44 | 56 | 11 | 12 | 12 | 15 | 1.090 | j = Extrator prim. |

previsões ao separar-se os dados em conjuntos menores e com maior representatividade de características técnicas e operacionais específicas.

Neste cenário, o algoritmo escolhido foi o florestas aleatórias, devido ao desempenho apurado no conjunto de dados que abrangeu todos os equipamentos. Novamente os modelos foram gerados automaticamente pelo WEKA, mantendo-se os mesmos parâmetros já definidos na seção 4.1.3.

Nesta configuração, a acurácia acumulada entre todos os equipamentos avaliados com os modelos individuais foi de 82,61%.

Os resultados detalhados apurados por equipamento estão relacionados na tabela 22. A análise destes dados evidencia que a pior das acurácias apuradas foi para o veículo 4, com valor de 77,84%, inferior àquela apurada pelo modelo florestas aleatórias (82,80%) e a melhor foi para o equipamento 13, com 91,19%.

O modelo individualizado foi melhor em sete equipamentos; o geral em oito. Um comparativo de desempenho entre os modelos é exibido na figura 32.

Avaliando a estatística Kappa, o modelo individualizado foi melhor em sete equipamentos: 3, 8, 11, 12, 13, 14 e 15; o modelo generalizado, com o valor de 0,8045, nos demais.

Frente a estas constatações conclui-se que, para esta previsão, tanto o modelo generalizado quanto o individualizado atendem o objetivo que se está buscando, com ligeira vantagem para o modelo generalizado.

Tabela 22 – Resultados gerais obtidos com modelos individuais

| Equipamento | Instâncias Corretas | Instâncias Incorretas | Total de Instâncias | Acurácia | Kappa |
|---|---|---|---|---|---|
| 1 | 1.215 | 292 | 1.507 | 80,62% | 0,773 |
| 2 | 1.136 | 256 | 1.392 | 81,61% | 0,789 |
| 3 | 1.420 | 268 | 1.688 | 84,12% | 0,819 |
| 4 | 1.553 | 442 | 1.995 | 77,84% | 0,740 |
| 5 | 1.528 | 377 | 1.905 | 80,21% | 0,768 |
| 6 | 1.254 | 261 | 1.515 | 82,77% | 0,803 |
| 7 | 1.175 | 260 | 1.435 | 81,88% | 0,792 |
| 8 | 1.147 | 229 | 1.376 | 83,36% | 0,805 |
| 9 | 1.154 | 262 | 1.416 | 81,50% | 0,793 |
| 10 | 826 | 208 | 1.034 | 79,88% | 0,765 |
| 11 | 871 | 162 | 1.033 | 84,32% | 0,810 |
| 12 | 688 | 119 | 807 | 85,25% | 0,828 |
| 13 | 880 | 85 | 965 | 91,19% | 0,889 |
| 14 | 828 | 119 | 947 | 87,43% | 0,858 |
| 15 | 905 | 150 | 1.055 | 85,78% | 0,833 |
| Total | 16.580 | 3.490 | 20.070 | 82,61% | — |

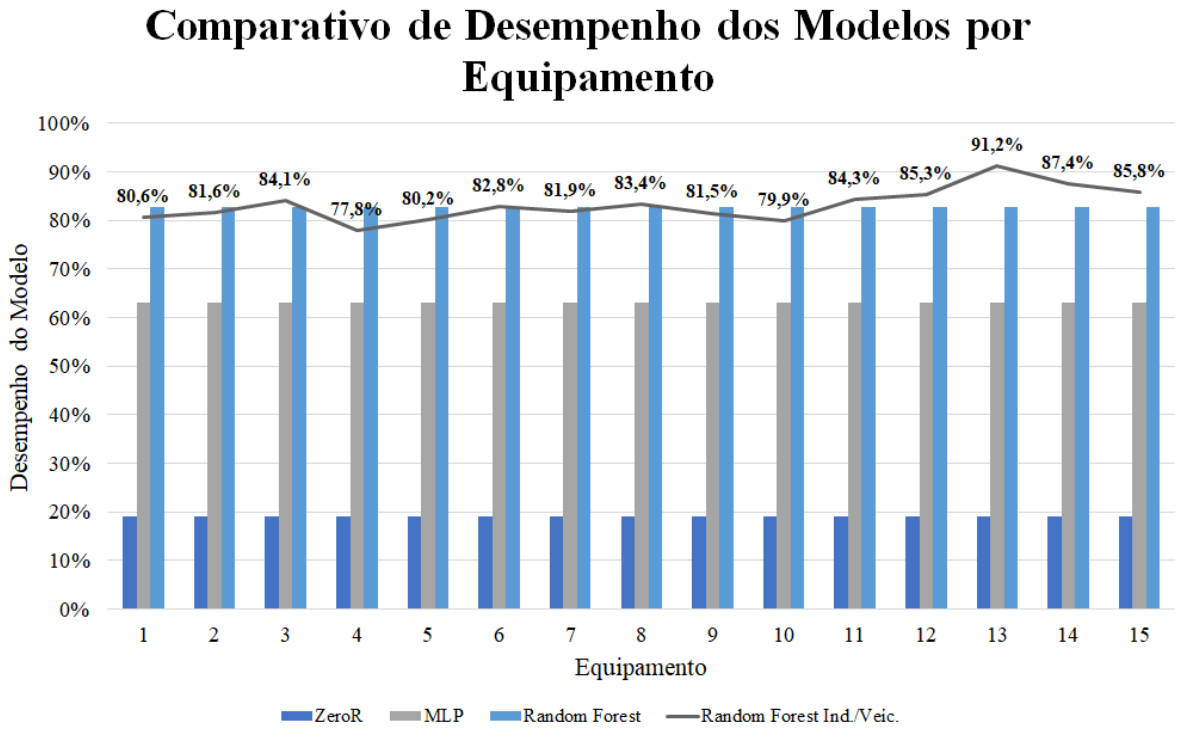



Figura 32 – Resultado dos modelos por equipamento.

## 4.2 Modelo B - horas de operação

O modelo B foi construído para prever horas de operação, ou seja, quantas horas de operação estão previstas para o ciclo atual, desde a última falha. Utilizou para isso o conjunto de dados B, definido na tabela 11.

Deste conjunto, a característica 1-Date foi descartada por não ter influência na variável dependente. As características *2-TotalkmA, 4-AvgVeloc, 5-AvgRPM, 6-ReverseTimeA, 7-ReverseWorkA, 8-CropTimeA, 9-ElevatorTimeA, 10-EngineTimeA, 11-IdleTimeA, 12-Month, 13-WorkShift, 14-TimeFromLastPrev, 15-TimeFromLastFail, 16-LastFailCause* e *17-NextFailCause* são as variáveis independentes do conjunto de dados e a característica *3-WorkTimeA*, a variável dependente.

As técnicas empregadas na construção do modelo foram a Zero R (a ser utilizada como linha de base), regressão linear, redes neurais MLP e florestas aleatórias.

### 4.2.1 Modelo baseado no algoritmo Zero R

O modelo utilizado como linha de base para comparativo de desempenho com os demais algoritmos utilizou o algoritmo Zero R. O coeficiente de correlação deste modelo foi de -0,069.

O resumo dos resultados obtidos com o emprego deste modelo está listado na tabela 24. A comparação entre as predições obtidas pelo modelo Zero R e os valores reais pode ser visualizada na figura 33.

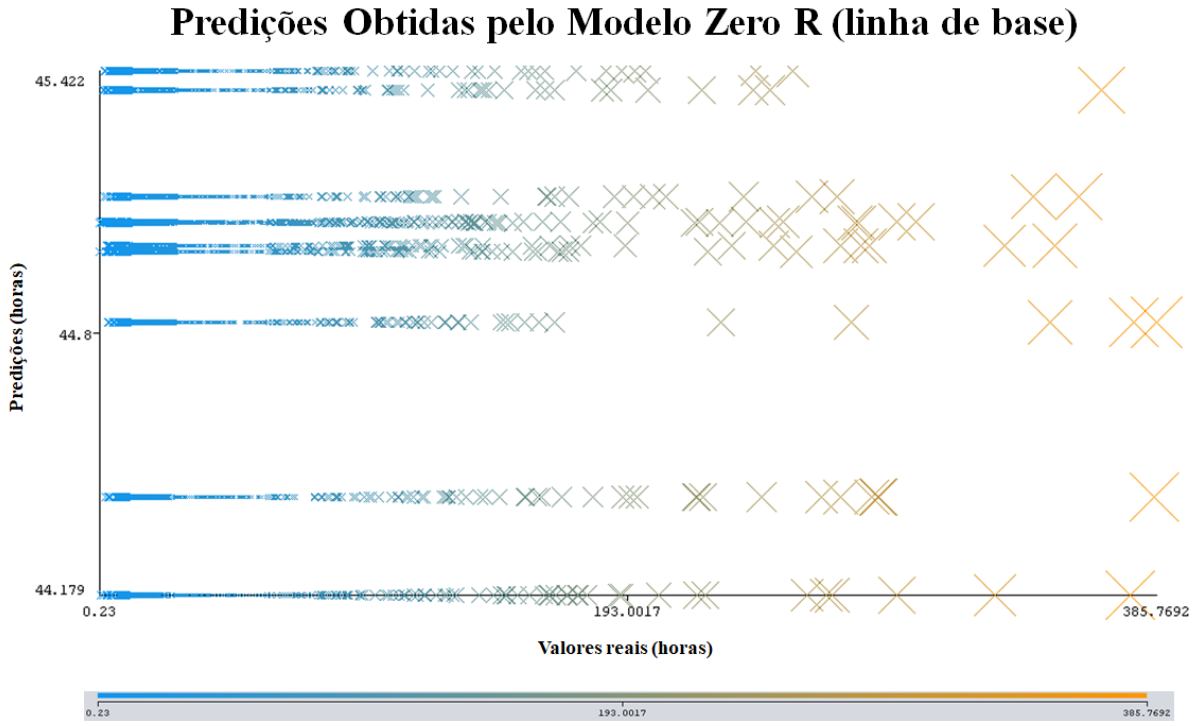



Figura 33 – Comparação entre valores reais e predições obtidas com modelo Zero R.

### 4.2.2 Modelo baseado em regressão linear

O segundo modelo testado para predição de horas de operação foi baseado em uma regressão linear. O coeficiente de correlação deste modelo foi de 0,994.

O modelo foi gerado automaticamente pelo WEKA e utilizou os parâmetros: seleção de características = método M5, eliminação de características correlacionadas = sim, parâmetro Ridge = 1.0E-8 e uso de decomposição QR = não, todos definidos automaticamente pelo WEKA.

A equação definida em 4.1 é a equação de regressão resultante do modelo. O método de seleção identificou dez variáveis relevantes, sendo elas: *TotalkmA, AvgVeloc, AvgRPM, CropTimeA, ElevatorTimeA, IdleTimeA, Month, LastFailCause, NextFailCause* e *Vehic*. Destas, quatro são categóricas: *Month, LastFailCause, NextFailCause* e *Vehic* e estão acompanhadas de um índice que identificam um subconjunto de classes que influenciam o valor que a variável pode assumir.

As variáveis numéricas podem ser substituídas diretamente pelos valores que estão em determinada instância do conjunto de dados, enquanto que as variáveis resultantes da transformação das variáveis categóricas serão substituídas por zero ou um, dependendo do valor que está na característica da instância. As regras são as seguintes:

- $Month_1 = 1$, caso classe da característica seja 11, 9, 8, 4, 7, 5, 6 ou 12. 0 caso contrário;
- $Month_2 = 1$, caso classe da característica seja 9, 8, 4, 7, 5, 6 ou 12. 0 caso contrário;

- $Month_3 = 1$, caso classe da característica seja 7, 5, 6 ou 12. 0 caso contrário;
- $Month_4 = 1$, caso classe da característica seja 5, 6 ou 12. 0 caso contrário;
- $Month_5 = 1$, caso classe da característica seja 6 ou 12. 0 caso contrário;
- $Month_6 = 1$, caso classe da característica seja 12. 0 caso contrário;
- $LastFailCause_1 = 1$, caso classe da característica seja Outros, Sistema Rodante, Rolos, Extrator primário ou Nenhuma Falha. 0 caso contrário;
- $LastFailCause_2 = 1$, caso classe da característica seja Rolos, Extrator primário ou Nenhuma Falha. 0 caso contrário;
- $LastFailCause_3 = 1$, caso classe da característica seja Extrator primário ou Nenhuma Falha . 0 caso contrário;
- $LastFailCause_4 = 1$, caso classe da característica seja Nenhuma Falha. 0 caso contrário;
- $NextFailCause_1 = 1$, caso classe da característica seja Rolos, Picador, Elevador, Sistema Rodante, Divisor de Linha ou Extrator primário. 0 caso contrário;
- $NextFailCause_2 = 1$, caso classe da característica seja Sistema Rodante, Divisor de Linha ou Extrator primário. 0 caso contrário;
- $NextFailCause_3 = 1$, caso classe da característica seja Divisor de Linha ou Extrator primário. 0 caso contrário;
- $Vehic_1 = 1$, caso classe da característica seja 1, 4, 3, 6, 5, 11, 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_2 = 1$, caso classe da característica seja 4, 3, 6, 5, 11, 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_3 = 1$, caso classe da característica seja 3, 6, 5, 11, 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_4 = 1$, caso classe da característica seja 6, 5, 11, 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_5 = 1$, caso classe da característica seja 5, 11, 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_6 = 1$, caso classe da característica seja 11, 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_7 = 1$, caso classe da característica seja 7, 9, 8, 12, 10, 13, 15 ou 14. 0 caso contrário;

- $Vehic_8 = 1$, caso classe da característica seja 12, 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_9 = 1$, caso classe da característica seja 10, 13, 15 ou 14. 0 caso contrário;
- $Vehic_{10} = 1$, caso classe da característica seja 13, 15 ou 14. 0 caso contrário;
- $Vehic_{11} = 1$, caso classe da característica seja 15 ou 14. 0 caso contrário;
- $Vehic_{12} = 1$, caso classe da característica seja 14. 0 caso contrário;

$$\begin{aligned} Y = {} & 0,0368 \times TotalkmA + 0,1748 \times AvgVeloc + -0,0045 \times AvgRPM + \\ & 0,6438 \times CropTimeA + 0,0857 \times ElevatorTimeA + 2,0642 \times IdleTimeA + \\ & 2,2532 \times Month_1 + -2,7761 \times Month_2 + -0,6076 \times Month_3 + \\ & 0,9732 \times Month_4 + -0,9089 \times Month_5 + 7,2715 \times Month_6 + \\ & 0,4654 \times LastFailCause_1 + -0,5084 \times LastFailCause_2 + 1,287 \times LastFailCause_3 + \\ & 19,3706 \times LastFailCause_4 + 0,3152 \times NextFailCause_1 + -0,47 \times NextFailCause_2 + \\ & 0,7844 \times NextFailCause_3 + -1,0441 \times Vehic_1 + 0,5113 \times Vehic_2 + \\ & -1,437 \times Vehic_3 + 0,4061 \times Vehic_4 + -1,0432 \times Vehic_5 + \\ & 3,2889 \times Vehic_6 + -2,2538 \times Vehic_7 + -1.9214 \times Vehic_8 + \\ & 3,0236 \times Vehic_9 + -1,2599 \times Vehic_{10} + 1,4583 \times Vehic_{11} + \\ & -1,5778 \times Vehic_{12} + 6,9772 \end{aligned} \quad (4.1)$$

O resumo dos resultados obtidos com o emprego deste modelo está listado na tabela 24. A comparação entre as predições obtidas pelo modelo de regressão linear e os valores reais pode ser observada na figura 34. Neste gráfico, destaca-se a predição de 103,7 horas onde o valor real é de 269,2. Esta situação está ocorrendo pois os valores de quilômetros percorridos e horas de operação por sistema estão abaixo de 50% da média destes mesmos valores quando avalia-se as instâncias que registraram valores próximos de 270 horas de operação. A mesma situação se repete para os modelos MLP e florestas aleatórias, obviamente com predições diferentes.

### 4.2.3 Modelo baseado em uma RNA MLP

O terceiro modelo testado para predição de horas foi uma RNA baseada no algoritmo MLP. O coeficiente de correlação atingido por este modelo foi de 0,990.

A RNA foi gerada automaticamente pelo WEKA e utilizou os parâmetros decaimento = não, taxa de aprendizagem = 0,3, momentum = 0,2, épocas de treinamento = 500 e método de otimização BFGS, todos definidos automaticamente pelo WEKA. Os valores das características foram normalizados em uma etapa de pré-processamento do modelo.

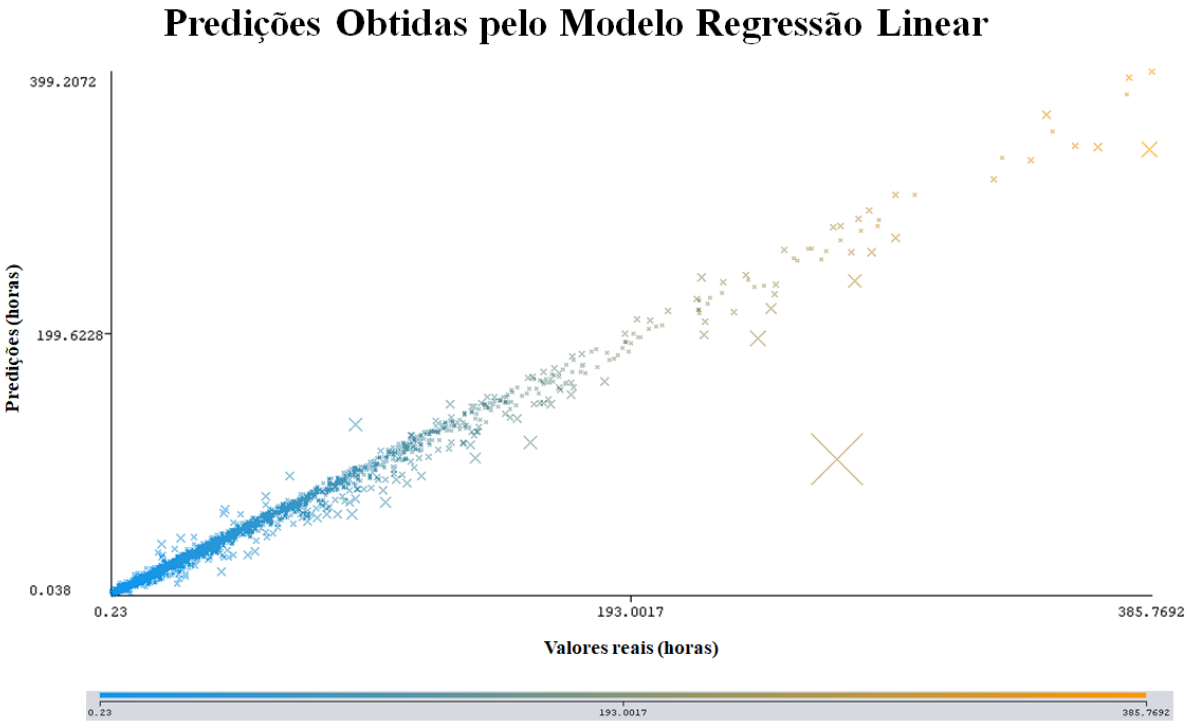



Figura 34 – Comparação entre valores reais e predições obtidas com modelo regressão linear.

A rede gerada está representada na figura 35 e o resumo dos resultados obtidos apurados com o emprego deste modelo está listado na tabela 24. A comparação entre as predições obtidas pelo modelo RNA / MLP e os valores reais pode ser observada na figura 36.

### 4.2.4 Modelo baseado em florestas aleatórias

O último modelo testado foi baseado no algoritmo florestas aleatórias, tendo sido gerado automaticamente pelo WEKA. Utilizou os parâmetros OOB = não, computar relevância de características = sim, tamanho máximo da árvore = não, número máximo de características por árvore = não definido e número de árvores na floresta = 100, todos definidos automaticamente pelo WEKA. O coeficiente de correlação obtido foi de 0,990.

O modelo gerado é composto por 100 árvores, cujos tamanhos variam entre 1.371 e 2.279 nós. A importância das características, baseada na impureza média e na quantidade de nós onde a característica aparece no modelo, está na listada na tabela 23, enquanto o resumo dos resultados obtidos está listado na tabela 24. A comparação entre as predições obtidas pelo modelo florestas aleatórias e os valores reais pode ser observada na figura 37.

### 4.2.5 Regressão linear individualizada por equipamento

Assim como no caso da predição da causa da próxima falha, também foram desenvolvidos modelos individualizados para a predição das horas de operação. A técnica adotada para o desenvolvimento de tais modelos foi a regressão linear, dado o desempenho superior obtido por

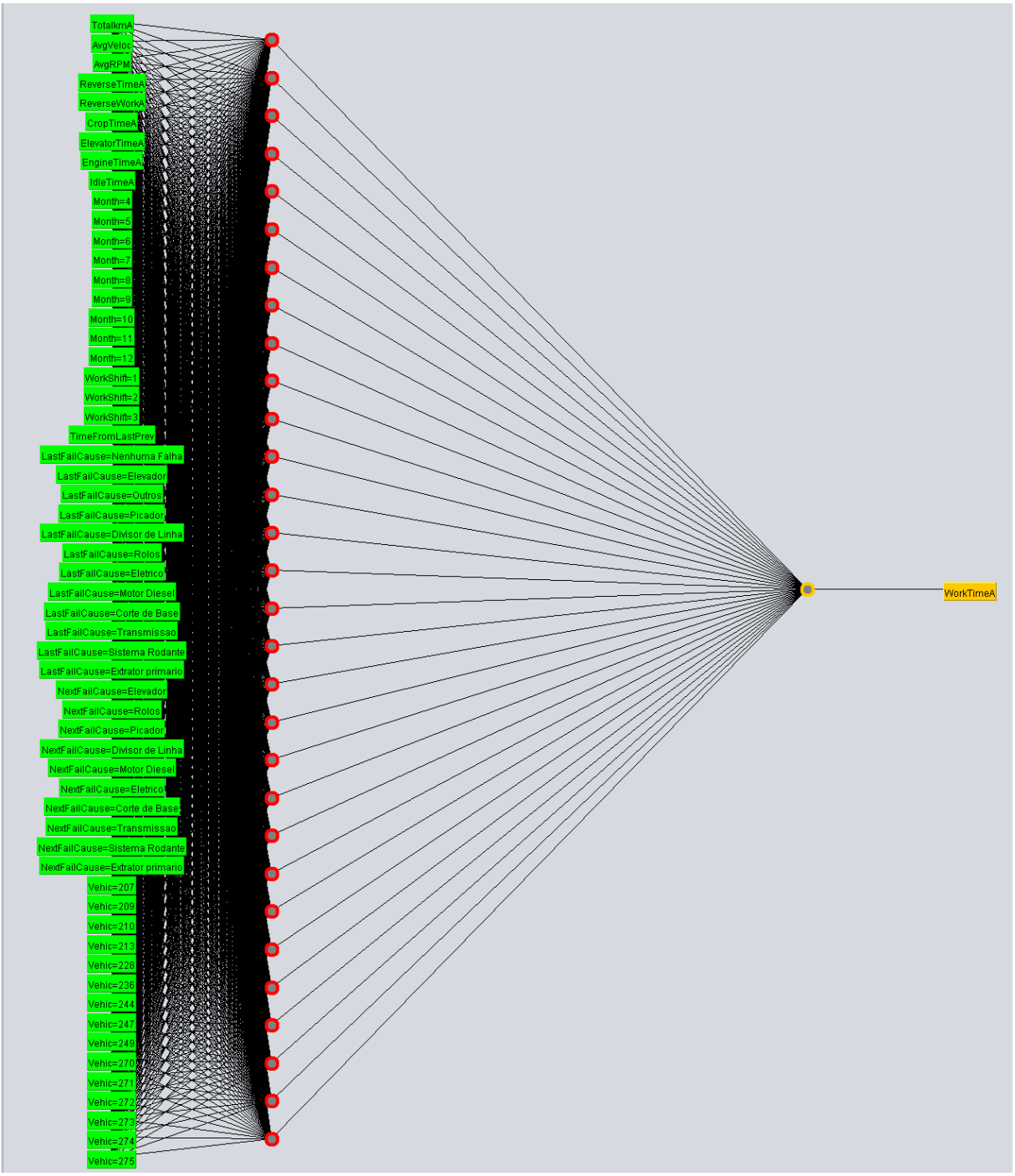



Figura 35 – RNA/MLP construída para o modelo.

Fonte: elaborada pelo autor, conforme modelo gerado pelo *workbench* WEKA.

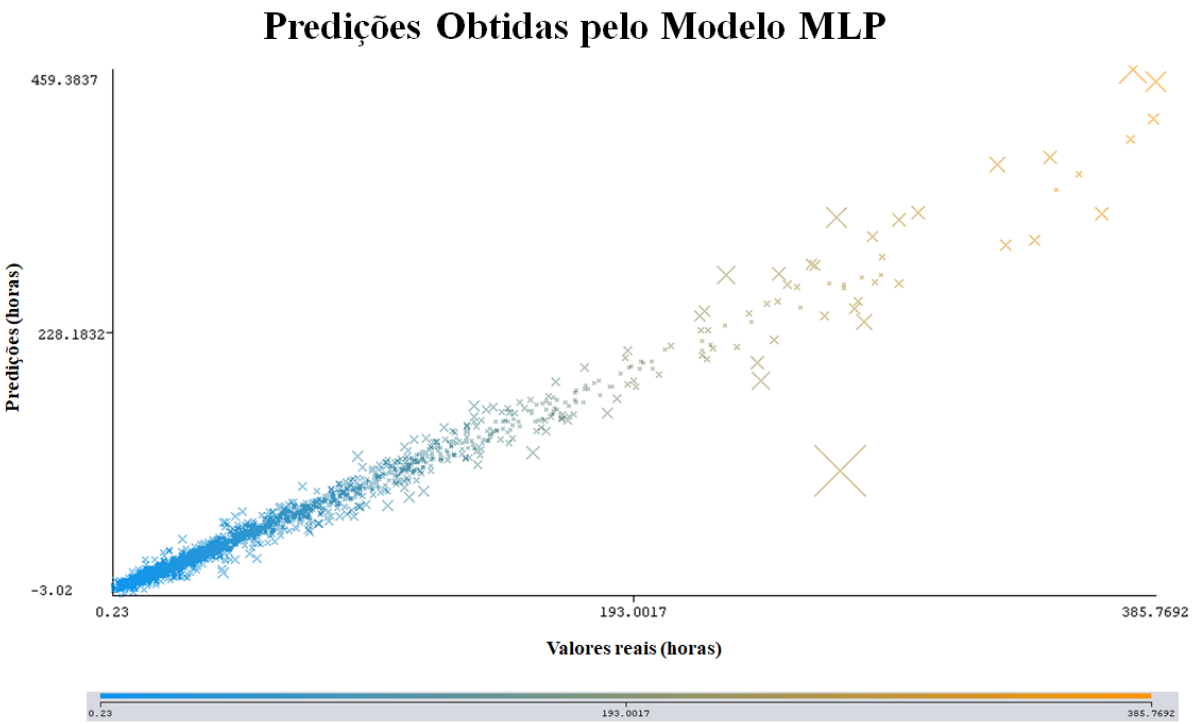



Figura 36 – Comparação entre valores reais e predições obtidas com modelo RNA / MLP.

Tabela 23 – Importância de atributos - modelo florestas aleatórias para previsão de horas

| # | Impureza Média | Número de Nós | Característica |
|---|---|---|---|
| 1 | 46.181,73 | 4.271 | CropTimeA |
| 2 | 40.518,35 | 5.008 | EngineTimeA |
| 3 | 37.329,84 | 2.702 | ElevatorTimeA |
| 4 | 25.686,56 | 3.591 | IdleTimeA |
| 5 | 19.095,19 | 4.701 | TotalkmA |
| 6 | 12.094,00 | 2.404 | ReverseTimeA |
| 7 | 1.862,17 | 3.575 | Vehic |
| 8 | 1.270,69 | 773 | ReverseWorkA |
| 9 | 942,37 | 4.096 | LastFailCause |
| 10 | 803,19 | 4.045 | Month |
| 11 | 658,31 | 4.150 | NextFailCause |
| 12 | 419,88 | 3.266 | AvgRPM |
| 13 | 389,50 | 1.288 | TimeFromLastPrev |
| 14 | 345,81 | 1.825 | WorkShift |
| 15 | 341,38 | 3.370 | AvgVeloc |

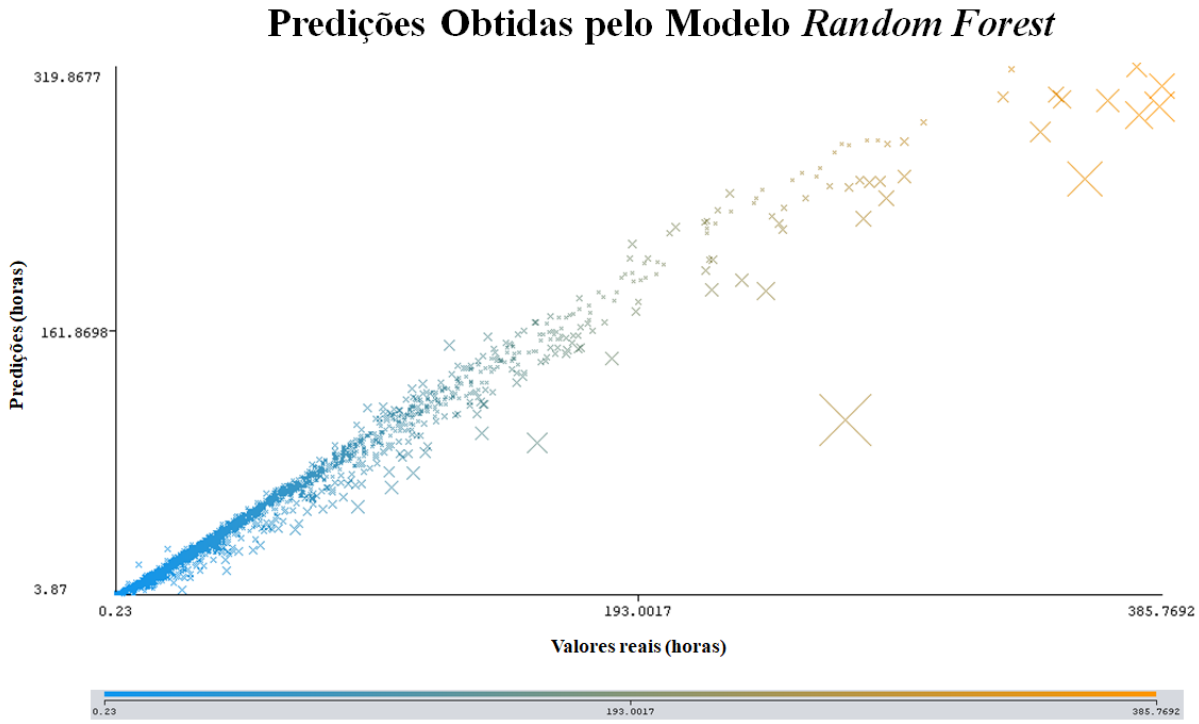



Figura 37 – Comparação entre valores reais e predições obtidas com o modelo florestas aleatórias.

esta nos testes com o conjunto de dados unificados. Estes resultados estão resumidos na tabela 25.

Tabela 24 – Resultados obtidos com modelos para predição de horas de operação na próxima falha

| Medida | ZeroR | Regressão Linear | RNA / MLP | Florestas Aleatórias |
|---|---|---|---|---|
| Coeficiente de correlação | -0,069 | 0,994 | 0,990 | 0,990 |
| Erro médio absoluto | 32,913 | 2,553 | 3,865 | 3,049 |
| Raiz quadrada do erro médio | 48,299 | 5,273 | 6,865 | 7,079 |
| Erro médio relativo | 100% | 7,756% | 11,742% | 9,264% |
| Raiz quadrada do erro relativo | 100% | 10,918% | 14,213% | 14,657% |
| Total de instâncias | 3.149 | 3.149 | 3.149 | 3.149 |

Os modelos individualizados foram gerados exatamente conforme o modelo geral: gerados automaticamente pelo WEKA, seleção de características = método M5, eliminação de características correlacionadas = sim, parâmetro Ridge = 1.0E-8 e uso de decomposição QR = não. Obviamente cada modelo gerou equações de regressão diferentes, todavia suprimiu-se a apresentação das equações individuais.

Analisando-se os resultados individuais relacionados na tabela 25 e apresentados no gráfico da figura 38, quando comparados ao modelo geral é possível constatar que:

- o coeficiente de correlação foi igual ou superior no modelo geral em sete casos (equipa-

Tabela 25 – Resultados obtidos com modelos individuais para predição de horas de operação

| Equipamento | Instâncias | Coeficiente de correlação | Erro médio absoluto | Raiz quadrada do erro médio | Erro médio relativo | Raiz quadrada do erro relativo |
|---|---|---|---|---|---|---|
| 1 | 328 | 0,993 | 2,312 | 3,915 | 10,863% | 12,088% |
| 2 | 359 | 0,991 | 2,637 | 4,567 | 15,408% | 13,778% |
| 3 | 300 | 0,994 | 2,446 | 4,660 | 9,111% | 11,235% |
| 4 | 360 | 0,993 | 2,494 | 3,948 | 10,289% | 11,586% |
| 5 | 311 | 0,997 | 2,434 | 3,467 | 7,834% | 7,720% |
| 6 | 250 | 0,995 | 3,163 | 4,892 | 9,518% | 9,511% |
| 7 | 197 | 0,998 | 2,394 | 3,523 | 6,698% | 6,780% |
| 8 | 180 | 0,996 | 2,899 | 4,445 | 7,597% | 8,448% |
| 9 | 189 | 0,994 | 3,411 | 5,230 | 9,369% | 10,914% |
| 10 | 121 | 0,995 | 3,702 | 5,738 | 8,586% | 10,142% |
| 11 | 154 | 0,958 | 6,173 | 14,935 | 16,756% | 28,496% |
| 12 | 95 | 0,996 | 3,448 | 5,526 | 7,710% | 8,763% |
| 13 | 97 | 0,997 | 3,682 | 5,723 | 7,147% | 8,085% |
| 14 | 98 | 0,994 | 4,818 | 7,006 | 9,446% | 10,498% |
| 15 | 110 | 0,990 | 5,293 | 8,661 | 11,451% | 14,522% |

mentos 1 a 4, 9, 11 e 15);

- o erro médio absoluto foi menor no modelo geral em dez casos (equipamentos 2, 6, 8, 9 a 15);

- a raiz quadrada do erro médio foi menor no modelo geral em seis casos (equipamentos 10 a 15);

- mesmo que a estratégia de validação adotada *(k-fold cross-validation)* evite a ocorrência de *overfiting*, a quantidade de registros disponíveis para treinamento pode ser insuficiente para garantir a generalização do aprendizado, se adotados os modelos individuais.

Frente a estas constatações conclui-se que, para previsão das horas de operação, tanto o modelo generalizado quanto o individualizado atendem o objetivo que se está buscando. Nas próximas seções, todavia, será adotado o modelo generalizado, levando-se em consideração, principalmente, a quantidade de registros disponíveis para os modelos individualizados.

## 4.3 Utilização conjunta dos modelos

Foram gerados até aqui dois modelos independentes: o primeiro para previsão da causa da próxima falha e o segundo para previsão de horas de operação entre falhas.

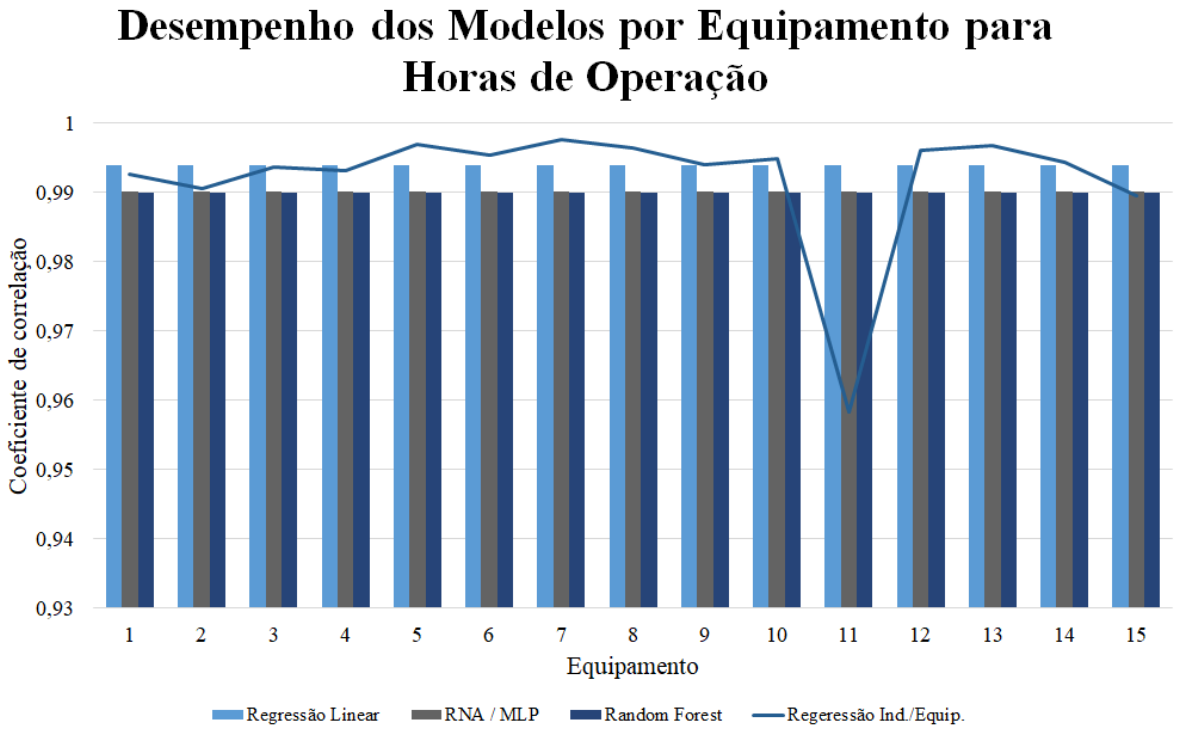



Figura 38 – Desempenho dos modelos de regressão linear por equipamento.

Observe-se que o segundo modelo tem como uma de suas características a causa da próxima falha. Ora, sendo o resultado do primeiro modelo necessário para o segundo, necessita-se, pois, da conexão entre ambos para atingir-se o objetivo geral estabelecido.

A proposta é que a predição ocorra em duas etapas:

1. cria-se um conjunto de dados com estrutura idêntica ao do conjunto de dados A, obviamente sem a variável resposta;

2. o Modelo A realiza a previsão da causa da próxima falha para o conjunto de dados criado na etapa anterior;

3. a partir do conjunto de dados resultante da predição realizada pelo modelo A, cria-se um conjunto de dados com estrutura idêntica ao do conjunto de dados B, novamente sem a variável resposta;

4. o Modelo B realiza a previsão do tempo de operação para o conjunto de dados criado na etapa anterior;

5. do conjunto de dados resultante da etapa anterior, extrai-se as informações: (i) número do veículo; (ii) causa da próxima falha; e (iii) tempo de operação previsto, que serão utilizadas para planejamento das equipes de manutenção.

Esta proposta apresenta a seguinte característica: nos casos onde a previsão não coincidir com a falha que realmente for ocorrer, e isto acontece em aproximadamente 18% dos casos, conforme relatado na tabela 19, a previsão resultante do modelo B é prejudicada por utilizar a variável com valor diferente do que deveria.

Também é verdade, porém, que este desvio é amenizado pelos conjuntos de classes que determinam os valores que podem assumir as variáveis relacionadas a características categóricas, conforme explanado nas regras de atribuição de valores das variáveis categóricas na seção 4.2.2.

Para fins de análise deste efeito, decidiu-se avaliar as predições e resultados aferidos em um dos equipamentos do estudo, tendo sido este equipamento escolhido aleatoriamente.

Essas predições foram realizadas pelo modelo A, sobre um conjunto de dados contendo 499 instâncias não apresentadas à fase de treinamento. A acurácia geral desta predição foi 78,96%, ou seja, 394 instâncias foram classificadas corretamente. A acurácia por classe está relatada na tabela 26.

Para predição das horas de operação, ajustou-se o conjunto de dados resultante do Modelo A e aplicou-se a equação de regressão 4.1 obtida pelo modelo B. O erro médio absoluto da predição do modelo foi de 3,08 horas.

Tabela 26 – Acurácia por classe para análise da utilização conjunta dos modelos

| Classe | Taxa P | Taxa FP | Precisão |
|---|---|---|---|
| Elevador | 0,794 | 0,056 | 0,794 |
| Rolos | 0,750 | 0,016 | 0,429 |
| Picador | 0,784 | 0,006 | 0,906 |
| Divisor de Linha | 0,863 | 0,040 | 0,788 |
| Motor Diesel | 0,893 | 0,078 | 0,748 |
| Elétrico | 0,659 | 0,018 | 0,784 |
| Corte de Base | 0,727 | 0,000 | 1,000 |
| Transmissão | 0,609 | 0,004 | 0,875 |
| Sistema Rodante | 0,606 | 0,013 | 0,769 |
| Extrator primário | 0,816 | 0,018 | 0,833 |
| Média Ponderada | 0,790 | 0,039 | 0,800 |

Deste conjunto foram isolados e comparados dois meses de operação, cujo resultado entre as predições e os dados reais estão relatados na tabela 27, donde é possível averiguar-se que:

- ocorreram 23 previsões no período;
- a predição para a causa da próxima falha foi diferente da real em 5 casos (instâncias 3, 4, 9, 10 e 17);
- a acurácia da predição da causa da próxima falha da amostra foi de 78,26%;

- o erro médio absoluto da previsão das horas de operação no ciclo entre falhas na amostra foi de 2 horas;
- para as instâncias 3 e 4 a diferença será irrelevante, pois tanto a causa de falha real quanto prevista acionam as mesmas variáveis *NextFailCause*;
- para as instâncias 9 e 17 a diferença será irrelevante, pois nem a causa de falha real nem a prevista acionam as variáveis *NextFailCause*;
- para a instância 10 a diferença será relevante apenas na variável $NextFailCause_1$, pois as variáveis $NextFailCause_2$ e $NextFailCause_3$ não são acionadas nem pela causa prevista e nem pela real;
- o erro médio absoluto da predição de horas de operação, nas amostras com predição de próxima falha correta, foi de 2 horas. Para o caso das amostras com predição de falha incorreta, de 1 hora. Isso ocorre pois com os coeficientes das variáveis *NextFailCause* entre -0,47 e 0,7844, esta pode causar uma diferença entre -0,47 e 1,1 horas no resulltado final da predição.

Demonstra-se, assim, que a predição incorreta da causa da próxima falha não gerou grandes distorções na predição do modelo B na amostra analisada.

Tabela 27 – Resultados reais vs. previstos em um mês de operação

| # | Falha Real | Falha Predição | Operação Real(h) | Operação Predição(h) | Erro Absoluto |
|---|---|---|---|---|---|
| 1 | Motor Diesel | Motor Diesel | 59 | 61 | 2 |
| 2 | Elevador | Elevador | 12 | 14 | 2 |
| 3 | Extrator primário | Divisor de Linha | 17 | 18 | 1 |
| 4 | Extrator primário | Divisor de Linha | 33 | 37 | 4 |
| 5 | Sistema Rodante | Sistema Rodante | 29 | 32 | 3 |
| 6 | Sistema Rodante | Sistema Rodante | 36 | 39 | 3 |
| 7 | Sistema Rodante | Sistema Rodante | 55 | 57 | 2 |
| 8 | Transmissão | Transmissão | 21 | 21 | 0 |
| 9 | Transmissão | Motor Diesel | 29 | 29 | 0 |
| 10 | Motor Diesel | Elevador | 13 | 14 | 1 |
| 11 | Motor Diesel | Motor Diesel | 36 | 37 | 1 |
| 12 | Motor Diesel | Motor Diesel | 15 | 15 | 1 |
| 13 | Elevador | Elevador | 17 | 17 | 0 |
| 14 | Divisor de Linha | Divisor de Linha | 38 | 38 | 0 |
| 15 | Divisor de Linha | Divisor de Linha | 21 | 15 | 6 |
| 16 | Elevador | Elevador | 14 | 15 | 1 |
| 17 | Elétrico | Motor Diesel | 9 | 10 | 1 |
| 18 | Elétrico | Elétrico | 14 | 15 | 1 |
| 19 | Extrator primário | Extrator primário | 73 | 71 | 3 |
| 20 | Extrator primário | Extrator primário | 81 | 79 | 2 |
| 21 | Extrator primário | Extrator primário | 98 | 95 | 2 |
| 22 | Extrator primário | Extrator primário | 113 | 112 | 1 |
| 23 | Elevador | Elevador | 10 | 12 | 2 |

CAPÍTULO

# 5

# CONCLUSÃO

Neste trabalho foram construídos dois modelos: um de classificação, que utilizou redes neurais MLP e florestas aleatórias, para realização de predições de motivos de falha em equipamentos agrícolas; e outro de regressão, através do uso de regressão linear, redes neurais MLP e florestas aleatórias, para realização de predição de tempo de operação entre falhas. Estes modelos foram construídos e testados utilizando dados de quinze equipamentos agrícolas do tipo colhedora de cana. Os modelos foram ainda comparados entre si para definir qual deles apresentou o melhor desempenho.

A coleta e entendimento inicial dos dados evidenciou a impossibilidade de utilizá-los diretamente na construção de modelos, fato que gerou a necessidade de se projetar e construir, a partir dos dados existentes, dois conjuntos de dados novos, estes sim adequados aos modelos que atenderam aos objetivos propostos. Esta etapa apresentou o problema da qualidade dos dados, ou seja, foram analisados sete aspectos e encontradas duas situações que exigiram algum tipo tratamento nos dados. Esta ação não causou prejuízos para este trabalho, mas evidenciou a importância e necessidade de uma criteriosa análise dos dados em projetos dessa natureza.

Desta etapa foi possível concluir ainda que as atividades de coleta, análise e principalmente preparação dos dados, são etapas que consomem grande parte do esforço dedicado aos projetos de mineração de dados e que, quanto melhor seu resultado, melhor será a etapa de geração de modelos. Não se deve, portanto, subestimar sua criticidade.

Observou-se grande variação nos intervalos entre falhas (de 250 a 750 horas, em média, para os dez tipos mais comuns de falhas), que quando combinadas geram 81,2% das falhas em até 72 horas desde a falha anterior; bem como no tempo médio para reparar um equipamento, que varia entre 2,2 e 4,1 horas, com desvio padrão entre 2 e 11 horas. Através da análise dos sensores dos equipamentos é possível afirmar ainda que ocorrem níveis de desgastes diferentes em seus sistemas, o que causa a variação também nos tipos de falhas.

O modelo mais adequado para predição da causa da próxima falha foi o que utilizou o

algoritmo florestas aleatórias, pois foi capaz de prever aproximadamente 82,8% das classes de falhas corretamente (9,8 p.p. acima do modelo com RNA MLP), sendo que a classe com menor acurácia obteve 74,1% e a com maior, 86,7%.

A maior adequação deste algoritmo está relacionada à aleatoriedade das variáveis em relação às falhas, uma vez que não é possível definir claramente correlações entre motivos de falhas e variáveis presentes no conjunto de dados. Ainda assim, identificou-se que as características mais relevantes para este modelo foram: (i) equipamentos, que caracteriza o comportamento próprio de cada equipamento; (ii) causa da última falha, que sugere um possível padrão sequencial de falhas; e (iii) mês que a falha ocorre, tempo de operação no turno e acumulados, quilômetros percorridos no turno e acumulados, velocidade média e RPM, que estão relacionados ao desgaste que o equipamento é submetido.

Para predição do tempo de operação, o modelo mais adequado foi o que utilizou a técnica de regressão linear, pois apresentou erro médio absoluto de 2,55 horas, seguido de florestas aleatórias, com 3,05 horas e MLP, com 3,87 horas. Esse resultado coincide com Breiman (2001), quando relata desempenho superior da técnica florestas aleatórias em modelos de classificação.

Da análise das variáveis utilizadas na equação de regressão, identifica-se o uso daquelas que representam desgaste (quilômetros percorridos, velocidade média, RPM e mês), individualidade (número do equipamento) e possível padrão sequencial de falhas (causa da última falha e causa da próxima falha).

Apesar do número do equipamento estar presente no modelos generalizados, modelos individualizados, para ambas as predições, não apresentaram vantagens significativas.

A adoção de dois modelos, onde a variável resposta do modelo para predição de causa da falha é utilizada para prever o tempo de operação, não causa prejuízos aos objetivos gerais e pode ser adotada. Desta situação pode-se afirmar que mesmo nos casos onde a predição da causa da próxima falha não coincidir com a falha que irá ocorrer, ainda assim o momento da falha será conhecido, situação esta que permitirá o adequado planejamento de curto prazo para as equipes de manutenção, o que contribuirá para diminuição dos tempos de reparo e, por consequência, maior disponibilidade dos equipamentos e do fornecimento de matéria prima nas indústrias.

Conclui-se também que a adoção de metodologias de projetos, como CRISP-DM, e *workbench* computacionais para projetos de aprendizado de máquina, como o WEKA, podem ter influência positiva no atingimento dos objetivos e metas, principalmente em ambientes empresariais.

Trabalhos futuros podem concentrar-se no desenvolvimento de aplicações computadorizadas que utilizem os dois modelos para gerar informações para as equipes de manutenção, ou ainda, experimentar modelos que explorem a sequência de falhas.

# REFERÊNCIAS


BREIMAN, L. Random forests. **Machine Learning**, v. 45, p. 5–32, 2001. Citado nas páginas 27 e 90.

BROWNLEE, J. **How To Estimate A Baseline Performance For Your Machine Learning Models in Weka**. 2016. Disponível em: <https://machinelearningmastery.com/estimate-baseline-performance-machine-learning-models-weka/>. Acesso em: 15/12/2020. Citado na página 27.

CASE. **Colhedoras de cana Case IH A8810**. Sorocaba, 2020. Disponível em: <https://assets.cnhindustrial.com/caseih/LATAM/LATAMASSETS/Folhetos/Colhedoras_e_Colheitadoras/Folheto_Técnico_A8810_PO_BAIXA.pdf>. Acesso em: 17/06/2021. Citado na página 37.

CHAPMAN, P.; CLINTON, J.; KERBER, R.; KHABAZA, T.; REINARTZ, T.; SHEARER, C.; WIRTH, R. **CRISP-DM 1.0: Step-by-Step Data Mining Guide**. 2000. Disponível em: <https://www.the-modeling-agency.com/crisp-dm.pdf>. Acesso em: 27/10/2017. Citado nas páginas 38 e 67.

FERREIRA, A. B. de H. **Mini Aurélio: o dicionário da língua portuguesa**. 8. ed. Curitiba: Positivo, 2010. Citado na página 38.

FRANK, E.; HALL, M. A.; WITTEN, I. H. **The WEKA Workbench Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques" Morgan Kaufmann, Fourth Edition**. 2016. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/Witten_et_al_2016_appendix.pdf>. Acesso em: 15/12/2020. Citado na página 69.

HALL, M.; FRANK, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; WITTEN, I. H. The weka data mining software: An update. **SIGKDD Explorations**, v. 11, n. 1, p. 10–18, 2009. Citado nas páginas 41 e 42.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediciton**. 2. ed. Stanford: Springer, 2009. Citado nas páginas 28 e 29.

JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. **An Introduction to Statistical Learning with Applications in R**. New York: Springer, 2013. Citado nas páginas 29 e 34.

KINGMA, D. P.; BA, J. L. Adam: A method for stochastic optimization. In: ICLR INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS. San Diego, 2015. Citado na página 32.

LOUPPE, G. **Understanding Random Forests: From Theory to Practice**. Tese (Doutorado) — University of Liege, Belgium, 10 2014. ArXiv:1407.7502. Citado na página 29.

MAGALHAES, P. S. G. **Ageitec. Cana-de-Açúcar / Máquinas e implementos**. 2009. Disponível em: <https://www.agencia.cnptia.embrapa.br/gestor/cana-de-acucar/arvore/CONTAG01_73_22122006154841.html>. Acesso em: 17/06/2021. Citado na página 37.

MARSLAND, S. **Machine Learning: An Algorithmic Perspective**. 2. ed. Boca Raton: CRC Press, 2015. Citado nas páginas 29, 30, 31, 34 e 35.

MEYER, P. L. **Probabilidade: Aplicações à Estatística**. Rio de Janeiro: LTC, 1969. Citado na página 32.

NASCIMENTO, D. C. do; RAMOS, P. L.; ENNES, A.; COCOLO, C.; NICOLA, M. J.; ALONSO, C.; RIBEIRO, L. G.; LOUZADA, F. A reliability engineering case study of sugarcane harvesters. **Gestão & Produção**, v. 27, n. 4, p. 0, 2020. Disponível em: <https://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-530X2020000400201&tlng=en>. Acesso em: 19/01/2021. Citado na página 23.

PMI. **Um guia do conhecimento em gerenciamento de projetos (guia PMBOK)**. 5. ed. São Paulo: Saraiva, 2014. Citado na página 38.

PROVOST, F.; FAWCETT, T. **Data Science For Business: What You Need to Now About Data Mining and Data-Analytic Thinking**. Sebastopol: O'Reilly, 2013. Citado nas páginas 34, 38 e 67.

RUDER, L. An overview of gradient descent optimization algorithms. In: INSIGHT CENTRE FOR DATA ANALYTICS, NUI AND AYLIEN, LTD. Galway and Dublin, 2017. Citado na página 32.

SCIKIT LEARN. **User Guide**: User guide. [S.l.], 2019. Disponível em: <https://scikit-learn.org/stable/modules/preprocessing_targets.html#preprocessing-targets>. Acesso em: 15/07/2020. Citado na página 68.

SILVA, J. E. A. R. da; ALVES, M. R. P. A.; COSTA, M. A. B. da. Planejamento de turnos de trabalho: uma abordagem no setor sucroalcooleiro com uso de simulação discreta. **Gestão & Produção**, v. 18, p. 73–90, 2011. Citado na página 24.

WALPOLE, R. E.; MYERS, R. H.; MYERS, S. L.; YE, K. **Probabilidade e Estatística para Engenharia e Ciências**. São Paulo: Pearson, 2009. Citado na página 33.

WIKIPéDIA. **7.2 - Lei dos Grandes Números**. 2018. Disponível em: <https://pt.wikipedia.org/wiki/Lei_dos_grandes_números#cite_note-:1-1>. Acesso em: 16/12/2020. Citado na página 28.

____. **Cohen's kappa**. 2020. Disponível em: <https://en.m.wikipedia.org/wiki/Cohen%27s_kappa>. Acesso em: 22/12/2020. Citado na página 36.

WITTEN, I. H.; FRANK, E.; HALL, M. A. **Data Mining: Practical Machine Learning Tools and Techniques**. 3. ed. Burlington: Elsevier, 2011. Citado nas páginas 27, 30, 35 e 36.


ICMC
USP
SÃO CARLOS