UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Utilização de aprendizado de máquina em sistemas de digestores comerciais

Fernanda Melo Jacques de Almeida

Monografia - MBA em Inteligência Artificial e Big Data

ICMC USP SÃO CARLOS

Fernanda Melo Jacques de Almeida

Utilização de aprendizado de máquina em sistemas de digestores comerciais

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Ronaldo Prati

Versão original

São Carlos

2023

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi, ICMC/USP, com os dados fornecidos pelo(a) autor(a)

S856m

Almeida, Fernanda Melo Jacques de
Utilização de aprendizado de máquina em sistemas de digestores comerciais / Fernanda Melo Jacques de Almeida ; orientador Ronaldo Prati. – São Carlos, 2023.
59 p. : il. (algumas color.) ; 30 cm.

Monografia (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2023.

1. LaTeX. 2. abnTeX. 3. Classe USPSC. 4. Editoração de texto. 5. Normalização da documentação. 6. Tese. 7. Dissertação. 8. Documentos (elaboração). 9. Documentos eletrônicos. I. Prati, Ronaldo, orient. II. Título.

"Não espere que vida seja 100% justa, mas esteja prepara para as oportunidades que lhe aparecem."

H.J.A.J

RESUMO

Almeida, F.M.J. **Utilização de aprendizado de máquina em sistemas de digestores comerciais**. 2023. 59p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

A crescente preocupação com o tratamento adequado de resíduos orgânicos tem estimulado a busca por soluções eficientes para lidar com esse desafio ambiental. Dentre as várias soluções disponíveis no mercado para abordar o aumento dos resíduos orgânicos, as máquinas digestoras de resíduos orgânicos se destacam como uma das opções mais sustentáveis. Essas máquinas viabilizam a decomposição desses materiais de maneira limpa, contribuindo para a redução no acúmulo de lixo em aterros sanitários e lixões, e seus efeitos adversos. Alguns dos sistemas existentes, disponibilizam na nuvem os dados do processo de digestão para que os gestores possam avaliar a eficiência e identificar possíveis falhas. Este recurso facilita muito a gestão do negócio, entretanto, quando realizado por pessoas, passa a se tornar inviável na medida que o número de digestores escala (podendo chegar a alguns milhares). Diante desse cenário, este estudo tem como objetivo auxiliar na análise e manutenção eficaz dessas máquinas, por meio da aplicação de aprendizado de máquina. Ao analisar séries temporais de dados de peso obtidos das máquinas digestoras, este trabalho procura desenvolver um sistema capaz de identificar padrões característicos e comportamentos anômalos, contribuindo, assim, para a operação eficiente e a gestão dessas máquinas.

Palavras-chave: Máquinas Digestoras de Resíduos Orgânico. Aprendizado de Máquina. Séries Temporais. Classificação.

ABSTRACT

Almeida, F.M.J. **Use of machine learning in commercial digesters systems.**. 2023. 59p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

The growing concern regarding the proper treatment of organic waste has spurred the search for efficient solutions to address this environmental challenge. Among the various solutions available in the market to tackle the increase in organic waste, organic waste digesters stand out as one of the most sustainable options. These machines enable the decomposition of these materials in a clean manner, contributing to the reduction of waste accumulation in landfills and dumpsites and their adverse effects. Some of the existing systems make the digestion process data available in the cloud so that managers can assess efficiency and identify possible failures. This feature greatly facilitates business management; however, when done by humans, it becomes unfeasible as the number of digesters scales (potentially reaching several thousand). In light of this scenario, this study aims to facilitate the monitoring and effective maintenance of these machines through the application of machine learning. By analyzing time-series weight data obtained from the digesters, this work seeks to develop a system capable of identifying characteristic patterns and anomalous behaviors, thus contributing to the efficient operation and management of these machines.

Keywords: Organic Waste Digesters. Machine Learning. Time Series. Classification.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CLP	Controlador Lógico Programável
IHM	Interface Homem-Máquina
CPU	Unidades Centrais de Processamento

SUMÁRIO

1 INTRODUÇÃO

1.1 Contextualização e Problemática

O problema de descarte do lixo no mundo é um grande desafio para a sustentabilidade ambiental e afeta a qualidade de vida de muitas pessoas. A geração de resíduos sólidos tem aumentado significativamente em todo o mundo, impulsionada pelo crescimento populacional, urbanização, mudanças nos padrões de consumo e hábitos de descarte inadequado.

Muitos países enfrentam dificuldades para lidar com o lixo de maneira adequada, com consequências ambientais e de saúde pública. O descarte inadequado de resíduos pode poluir o solo, a água e o ar, afetando a biodiversidade e a qualidade de vida das comunidades. Além disso, a presença de lixo em áreas urbanas e rurais pode atrair vetores de doenças e representar riscos para a saúde das pessoas.

> "A Política Nacional de Resíduos Sólidos (PNRS) do Brasil estabelece que somente devem ser enviados para aterros sanitários os resíduos que não tenham mais nenhuma possibilidade de recuperação ou reciclagem, ou seja os rejeitos."(RESíDUOS. . . , 2021)

> "Apesar disso, atualmente menos de 2% dos resíduos orgânicos são compostados no Brasil, o que em 2019 representou 300 mil toneladas de resíduos orgânicos reciclados. A maior parte ainda segue sendo disponibilizada para a coleta convencional e vai acabar em aterros sanitários ou, pior, em lixões."(RESíDUOS. . . , 2021)

Além do problema dos lixões, nos grandes centros urbanos o descarte de resíduos, por legislação municipal, deve ser realizado por contratação de veículos de coleta, onerando o custo do negócio associado, aumentando o tempo entre a geração e destinação do resíduo além de contribuir com aumento de emissões de gases de efeito estufa por depender de transporte de veículos motorizados para o descarte.

Visando promover uma solução para o lixo orgânico, principalmente das áreas urbanas, uma empresa desenvolveu um digestor comercial que executa uma série de processos onde micro organismos transformam as moléculas de lixo orgânico em uma água rica em nutrientes que pode servir como base para compostos fertilizantes, ou ser descartada diretamente no sistema de esgoto.

Esta solução é ideal para estabelecimentos que produzem grande quantidade de resíduos orgânicos diariamente como restaurantes, supermercados, indústrias alimentícias,

condomínios e até navios, em que não há descarte deste lixo durante a viajem. Dentre as vantagens do uso deste equipamento pode-se listar melhor limpeza/higiene do local, por não precisar armazenar lixo orgânico; economia com caminhões de coleta; redução de "*carboon footprint*", medida das emissões de gases de efeito estufa, que são liberadas na atmosfera como resultado das atividades humanas; e principalmente menos lixo orgânico nos aterros sanitários.

1.2 Justificativa

A solução do produto comercializado pelo fabricante integra um sistema de monitoramento dos digestores em tempo real. Este sistema de monitoramento coleta dados de sensores, controladores, usuários, *drivers*, etc. de todos os diferentes modelos de maquinas e seus periféricos. A coleta desses dados é armazenada em um servidor em nuvem que atualmente é acessada pelo fabricante que faz o monitoramento e atualização dos sistemas remotamente.

Nesse contexto, a aplicação de algoritmos de inteligência artificial (IA) sob as bases de dados do sistema pode ser extremamente benéfica. Os algoritmos de IA são capazes de analisar grandes quantidades de dados em tempo real, identificar padrões e prever possíveis falhas e necessidades de manutenção. A identificação precoce de problemas pode ajudar a evitar perdas econômicas significativas, reduzir o tempo de inatividade e garantir maior confiabilidade e segurança do produto.

Portanto, a justificativa e motivação para esta monografia é apresentar uma análise detalhada dos benefícios da aplicação de algoritmos de IA na previsão de falhas e manutenções e possíveis otimizações em sistemas que controlam digestores, bem como explorar as principais técnicas e metodologias utilizadas nesse contexto.

1.3 Objetivos

Este trabalho tem por objetivo principal utilizar de algoritmos de inteligência artificial na base de dados do sistema com intuito de antecipar a ocorrência de falhas e identificar padrões que possibilitam a otimização dos sistemas.

Os objetivos específicos incluem:

- Analise temporal do peso do material digerido.
- Classificação de padrões e comportamentos anômalos em séries temporais provenientes de máquinas de digestão de resíduos orgânicos

1.4 Organização do texto

Este trabalho está dividido da seguinte forma. O Capítulo 2 abordará o funcionamento de uma máquina digestora de resíduos orgânicos e os tipos de dados que o sistema de supervisão registra. No Capítulo 3, serão apresentados trabalhos relevantes que serviram de base para este projeto. No Capítulo 4, será apresentada a metodologia adotada, incluindo a preparação dos dados e a seleção do modelo de aprendizado de máquina escolhido. O Capítulo 5 se concentrará no detalhamento da implementação do experimento, suas configurações específicas e os resultados alcançados. Finalmente, o Capítulo 6 apresentará a conclusão deste trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo descreverá o funcionamento do sistema de digestores, os conceitos e as definições relacionados a este sistema.

2.1 O digestor comercial

O digestor comercial é uma máquina inspirada no estômago humano que, de uma maneira resumida, processa a comida (matéria orgânica), absorve os nutrientes e transforma o restante em outro tipo de matéria a ser descartado pelo organismo. Seguindo esta ideia, a máquina também recebe matéria orgânica depositada pelo usuário, e com os movimentos mecânicos de misturadores, água e enzimas o processo de digestão aeróbica é realizado naturalmente transformando o resíduo orgânico em água que pode ser despejada diretamente no sistema de esgoto, ou reservada para ser utilizada como base de compostos fertilizantes. Neste contexto, pode-se dizer que os movimentos mecânicos são análogos aos músculos do trato gastrointestinal e a água juntamente com as enzimas possuem a mesma funcionalidade do suco gástrico e bactérias do estômago.

2.2 Sistema de controle e supervisão

A máquina é controlada por um controlador lógico programável (CLP), que coleta os sinais de sensores da máquina e periféricos, gera outros sinais para controle e sinalização, e também disponibiliza diversos dados para um sistema de supervisão.

O CLP é basicamente um computar para controle de processos, composto por um hardware e um software. O hardware é constituído por uma ou mais Unidades Centrais de Processamento (CPU), interfaces de comunicação (seriais, ethernet, etc.). O software é responsável por implementar a lógica de controle do processo associado, podendo ser programado para realizar diversas tarefas, como monitorar sensores, controlar motores e válvulas, realizar cálculos, se comunicar com outros dispositivos entre outras funções. O CLP é amplamente utilizado em sistemas de automação industrial, pois permite a criação de soluções flexíveis e adaptáveis às necessidades específicas de cada processo produtivo, além de possibilitar a detecção de falhas e a tomada de decisões em tempo real.

Um sistema de supervisão, é um software utilizado para monitorar e controlar processos e equipamentos geralmente em um ambiente industrial. O sistema de supervisão coleta dados de sensores, dispositivos e equipamentos de campo, e exibe estas informações em uma interface gráfica para os operadores. Esses dados podem incluir temperatura, pressão, nível, fluxo, status de equipamentos, dentre outras variáveis. Além disso, o sistema de supervisão pode gerar relatórios, históricos, alertas e notificações para os operadores

e gestores, permitindo a detecção de falhas ou anomalias, além de fornecer dados para análises de desempenho e tomada de decisões.

No cenário desta aplicação, cada digestor comercial possui um CLP que é responsável por executar todo o algoritmo de controle e aquisição de dados do sistema, uma interface homem-máquina (IHM), que é uma tela que disponibiliza informações coletadas do CLP e permite que o usuário faça configurações localmente, como configurar receitas; e periféricos na qual pode-se destacar:

- Motor com velocidade variável programada;
- Sensor de temperatura;
- Sensor de peso;
- Sensor de vazão;
- Sensores digitais;

2.3 Funcionamento do sistema

Este trabalho não tem por objetivo descrever detalhadamente o algoritmo implementado para controle e monitoração do sistema, pois este é de propriedade da empresa detectora da solução. Entretanto será listado, de maneira sucinta, as ações que ocorrem durante o processo de funcionamento da máquina.

- O usuário abre a porta e deposita o lixo orgânico, a ação de abrir a porta é detectada por sensores digitais conectados ao CLP. Ao fechar a porta, o CLP inicia seu processo de decomposição do resíduo;
- Através do sensor de peso, o CLP calcula a quantidade total de lixo orgânico descartado e assim calcula a quantidade de água e a solução de bactérias digestoras necessária. O controle da água e da mistura de bactérias é realizado por válvulas e sensores de vazão;
- A velocidade do motor que é acoplado aos misturadores muda durante o processo da máquina, este é um dos parâmetros a ser programado pela IHM.
- À medida que o tempo passa a matéria orgânica é digerida, transformada em água e consequentemente o peso da solução vai diminuído.

Durantes o processo, alguns problemas podem ocorrer no sistema como:

- Falha no giro do motor, devido a descarte inadequado do material no digestor fazendo com que as hélices dos misturadores fiquem travadas, gerando uma sobrecorrente e falha no motor.
- Entupimento do ralo de vazão do líquido, também ocasionado por descarte inadequado e acúmulo de matéria não digerida.
- Deposito de maior quantidade de lixo orgânico que o sistema permite.

Muitos dos problemas que podem ocorrer no sistema podem ser detectados e notificados aos usuários pelo próprio CLP, por meio de alarmes e envio de mensagens. Porém, em vários casos, o problema não é pontual e não é detectado imediatamente pelo controlador, mas decorre de uma sucessão de eventos que se agravam com o tempo, como é o caso do entupimento do ralo, que geralmente ocorre gradativamente.

2.4 Monitoração do Peso do Sistema

O sistema de supervisão que opera em nuvem, armazena os dados coletados pelo CLP, como temperatura, velocidade do motor, corrente do motor, peso, sensor de vazão e; por meio da análise desses dados no tempo, um operador pode visualmente detectar as anomalias do processo.

As Figuras 1 e 2 representam o comportamento padrão do decaimento do peso com o tempo. Na Figura 1 o aumento rápido do peso informa que a porta da máquina foi aberta e uma quantidade de aproximadamente 200 Kg (conforme pode ser observado entre os instantes 9750 e 9080 na Figura 1) de material orgânico foi adicionado ao sistema. A medida que o tempo passa o peso diminui, com maior rapidez no início do processo. O peso nunca chega a zero pois o sistema necessita de matéria orgânica e água para manter viva a colônia de bactérias responsáveis pela decomposição. Na Figura 2, diversos ciclos normais sequenciais são ilustrados.

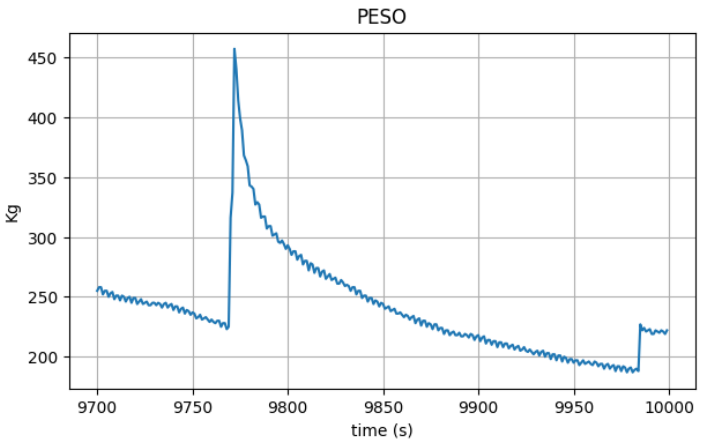


Figura 1 – Gráfico do comportamento padrão do peso (unitário)

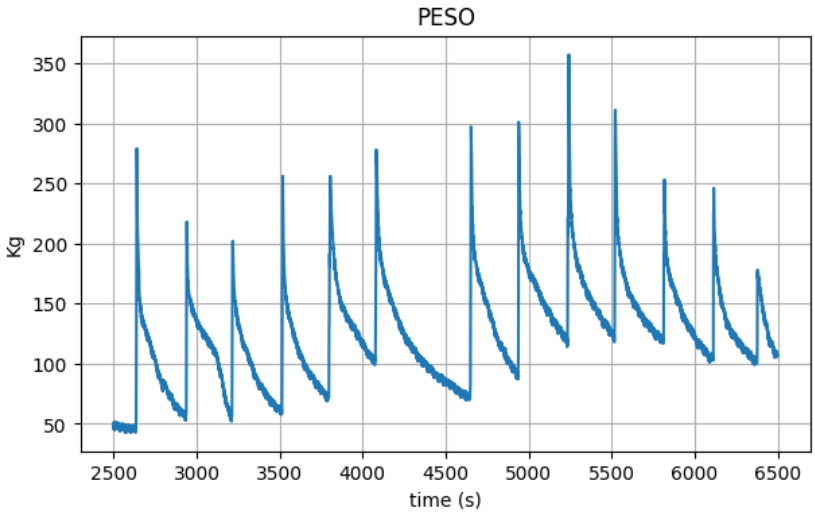


Figura 2 – Gráfico do comportamento padrão do peso no tempo

Nos casos de defeitos no sistema, temos um comportamento diferente. A partir da Figura 3, que apresenta um comportamento anômalo, evidencia-se que após a adição de material orgânico ao sistema, o peso inicial permaneceu constante e, posteriormente, aumentou. Nessa situação, pode-se deduzir um provável entupimento no sistema. No entanto, observa-se que com o decorrer do tempo, o problema foi solucionado de maneira espontânea.

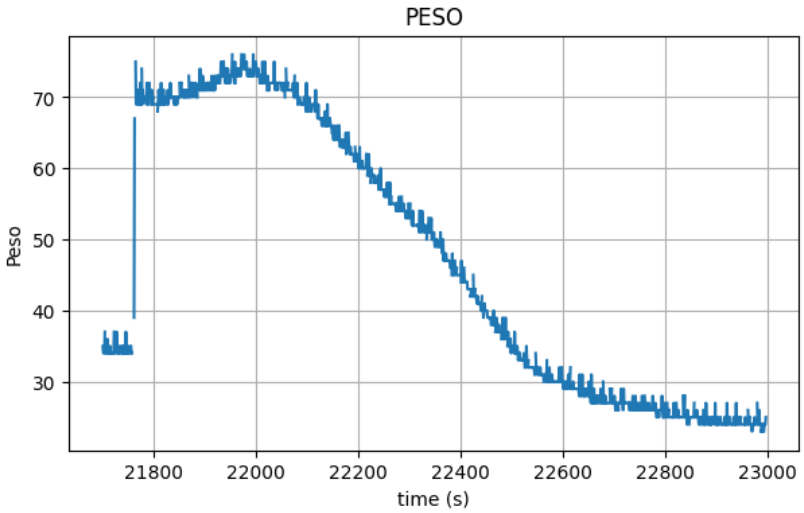


Figura 3 – Gráfico do comportamento do peso com entupimento do sistema

Na Figura 4, no intervalo de tempo 9750 a 10000, tem-se um exemplo de um provável entupimento ou falta de enzimas para decomposição acarretando a necessidade de intervenção humana para solucionar o problema.

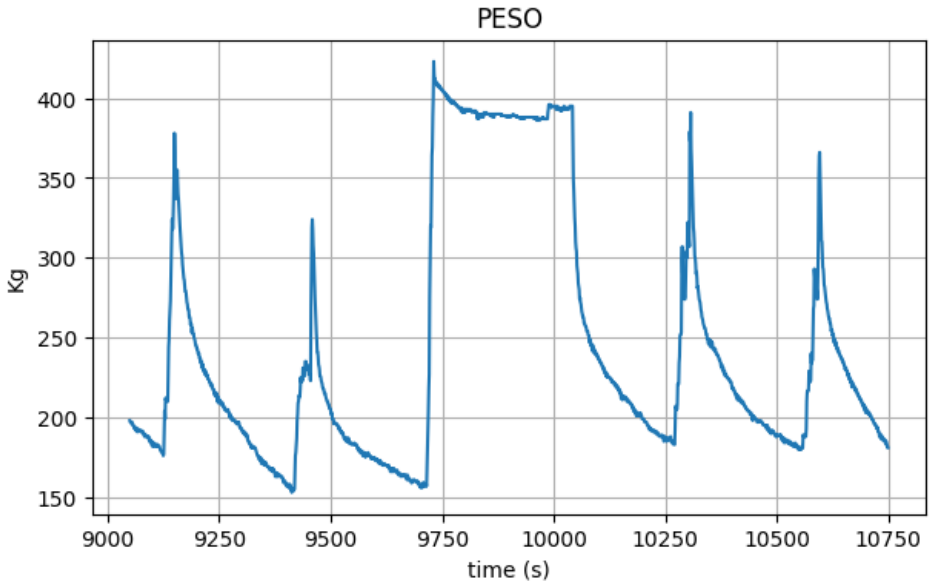


Figura 4 – Gráfico do comportamento do peso com necessidade de intervenção.

Analisar a relação peso × tempo parece trivial quando se considera um sistema conforme ilustrado na Figura 2. No entanto, na maioria das situações, o processo não é uniforme, e a operação de abertura e descarte de resíduos orgânicos varia com base no perfil do cliente onde o sistema está instalado. Existem máquinas que recebem pequenas porções de lixo orgânico várias vezes ao dia, enquanto outros sistemas executam o processo de decomposição apenas uma vez por dia, mas em grandes quantidades, como pode-se observar pela série temporal da Figura 5, no qual a máquina processa entre 50 a 170 Kg de lixo orgânico a cada ciclo. Já na Figura 6 tem-se um exemplo em que os resíduos são despejados na máquina várias vezes ao dia, em quantidades menores de 20 a 45 Kg.

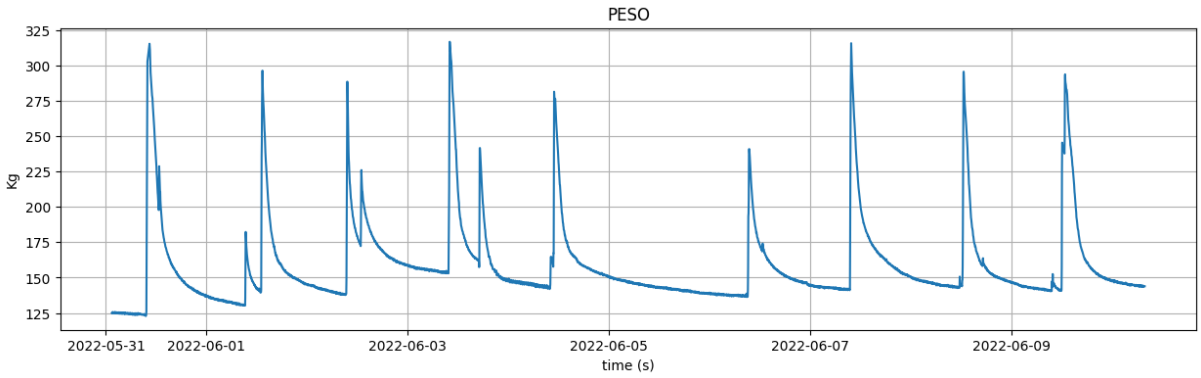


Figura 5 – Perfil do peso em que a máquina executa um ciclo em médio uma vez ao dia.

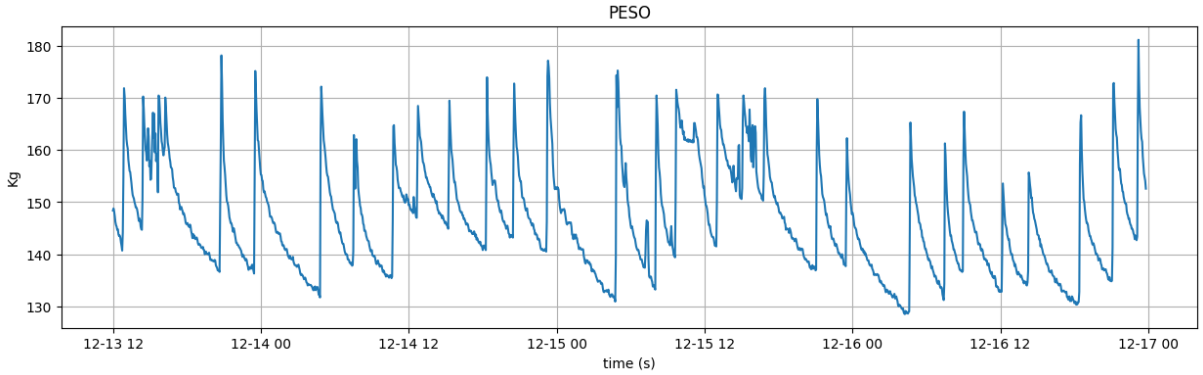


Figura 6 – Perfil do peso em que a máquina executa vários pequenos ciclos por dia.

Na Figura 7 tem-se um sistema aparentemente homogêneo porém com um ruído significativo na coleta de dados.

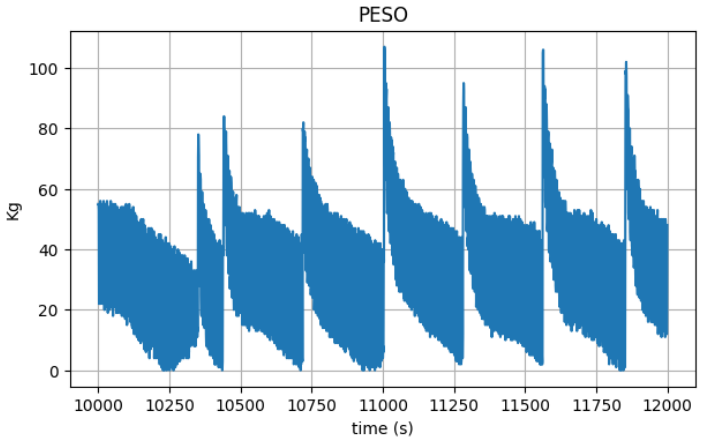


Figura 7 – Perfil do peso com ruído.

Todas essas análises e conclusões podem ser prontamente realizadas por um operador com acesso à plataforma de supervisão do sistema. Entretanto, isto exige que periodicamente o operador consulte a base de cada equipamento para realizar esta análise. Considerando que podem existir vários ciclos por dia em cada digestor e centenas de equipamentos em operação, esta tarefa de análise se torna inviável para um operador e onerosa quando se pensa em uma equipe para esta atividade. Com o objetivo de reduzir a dependência da análise humana para esses sistemas, este trabalho explora abordagens de aprendizado de máquina para identificar possíveis problemas nos sistemas através da análise da evolução do peso ao longo do tempo.

3 TRABALHOS RELACIONADOS

A seguir, serão apresentados trabalhos que foram utilizados como apoio para o desenvolvimento deste projeto.

Scachetti (2020) propôs um sistema de previsão de extravasões de esgoto em elevatórias de esgoto. Este trabalho utiliza informações de séries temporais, como a corrente dos motores e o nível da elevatória, e registros de ocorrência de extravasão e falhas de comunicação. O autor utiliza as arquiteturas de redes neurais Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) e ConvLSTM para identificar a melhor opção na previsão de eventos de extravasão. Por fim, opta pela rede neural recorrente ConvLSTM para a previsão de eventos de extravasão. A tese conclui que os objetivos propostos foram alcançados, e faz uma avaliação do desempenho dos modelos em dados de elevatórias com diferentes características das utilizadas no treinamento.

Annam, Mittapalli and Bapi (2011) apresentou um método para análise e agrupamento de séries temporais de batimentos cardíacos em Eletrocardiogramas (ECG). O método proposto utiliza o agrupamento K-medoids em combinação com a métrica de Alinhamento Temporal Dinâmico (DTW) para identificar anomalias nos batimentos cardíacos do ECG. O método é aplicado para agrupar batimentos cardíacos com base nas caraterísticas das ondas QRS (ondas que correspondem às fases de despolarização e repolarização dos ventrículos do coração), em 5 classes de tipos de batimentos cardíacos. Neste trabalho aos autores utilizaram dados do Banco de Dados de Arritmia MIT-BIH e obtiveram uma taxa de precisão de 82,08%.

No trabalho de Sugimura and Matsumoto (2011), é apresentado um sistema que adquire padrões de características e cria um classificador para dados de séries temporais sem exigir conhecimento prévio. O sistema começa pela extração de padrões de características dos dados de séries temporais usando a técnica de peso TF×IDF. O sistema recorta subsequências dos dados de séries temporais. Diversas sequências representativas são extraídas dessas subsequências usando técnicas de agrupamento e padrões de características são adquiridos a partir dessas sequências representativas. Esses padrões de características são então utilizados como atributos no processo de aprendizado de máquina. Além disso, é utilizado o algoritmo genético para aprimorar esses padrões de características, resultando em uma melhoria na precisão da classificação.

As referências mencionadas foram úteis para o estudo e a compreensão de alguns dos vários métodos aplicados em análise de séries temporais. Esses estudos não estão diretamente vinculados ao escopo deste projeto de monografia. Não foram identificados artigos ou publicações que empregassem dados semelhantes aos deste trabalho específico.

4 METODOLOGIA

Este capítulo descreve as metodologias utilizadas para o desenvolvimento do trabalho nas seguintes etapas:

- Visão geral e caracterização do problema;
- Coleta dos dados;
- Exploração e compreensão dos dados;
- Preparação dos dados;
- Exploração e escolha de modelos;
- Apresentação da solução;

4.1 Visão geral e caracterização do problema

O sistema de digestores comerciais é controlado por uma aplicação executada em um CLP. Esta aplicação já implementa vários algoritmos para prever falhas, parar a máquina quando detecta situações extremas, como sobre-corrente do motor, ou adição de maior conteúdo suportado pela máquina. Quando essas situações ocorrem alarmes são ativados e enviados à plataforma de monitoramento notificando o fabricante de maquinas e também seus clientes relacionados. Sendo assim, este trabalho tem por objetivo detectar situações de falhas que a aplicação executada no CLP não consegue detectar, ou que seja muito custoso para a aplicação.

A ocorrência de entupimento da saída de água é perceptível a um humano quando analisado o histórico do peso no tempo, como mostrado na Figura 8. Em geral, ao adicionar o lixo orgânico a ser processado na máquina, o peso aumenta significativa e instantaneamente. Porém, logo quando o processo se inicia, o peso do conteúdo da máquina decresce rapidamente nos primeiros minutos. Esse decaimento do peso diminui com o tempo até quase se estabilizar. Quando um entupimento ocorre, o peso do conteúdo na máquina pode apresentar comportamentos anômalos como aumento, decaimento muito lento ou estabilização. A implementação da detecção desses cenários na aplicação gera um custo computacional significativo para o controlador lógico programável (CLP). Porém, estas situações são relativamente simples para um ser humano detectar ao analisar o gráfico de peso ao longo do tempo.

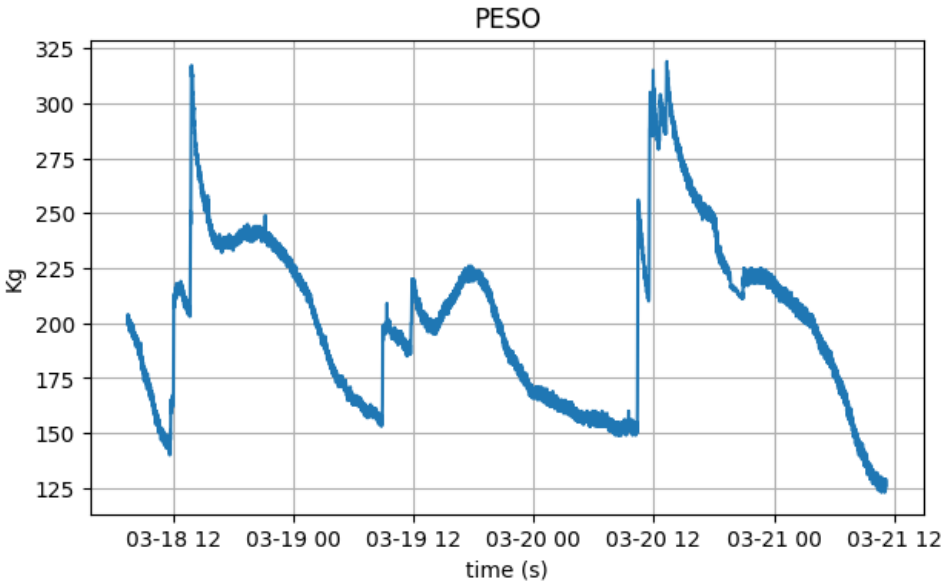


Figura 8 – Gráfico do comportamento do padrão com entupimento sucessivos

Tendo este cenário, este trabalho propõe métodos de detecção de entupimento da máquina digestora comercial através da análise da série temporal do peso de cada uma.

4.2 Coleta dos dados

Os dados de todas as máquinas são historiados em um sistema dedicado em nuvem, mas para este trabalho foi disponibilizado uma cópia de aproximadamente 80 máquinas em um banco de dados PostgreSQL[1]. O tamanho da base de dados de cada máquina variou entre 150.000 a 2.500.000 registros de peso e data-hora. Foram disponibilizados também outros registros de monitoramento que não foram utilizados como: "Temperatura", "Corrente do motor", "Status da Máquina", "Tensão do Motor", "Falha atual inversor", "Alarme Atual do Inversor", "Frequência do motor"e "Horímetro".

O acesso ao banco de dados foi realizado inicialmente pela interface de administração do PostgreSQL pgAdmin para visualização e entendimento da estrutura e tabelas do banco de dados. Posteriormente foram utilizadas as APIs "sql"e "psycopg2"do Python para realizar as consultas diretamente ao banco de dados.

Para facilitar a utilização dos dados em diferentes ambientes sem depender de instalações e recuperação de backup, os dados presentes no banco de dados foram exportados para arquivos no formato CSV (Comma Separated Values). Isto permitiu que o acesso ao banco de dados não fosse necessária durante o desenvolvimento dos algoritmos e análises.

[1] https://www.postgresql.org/

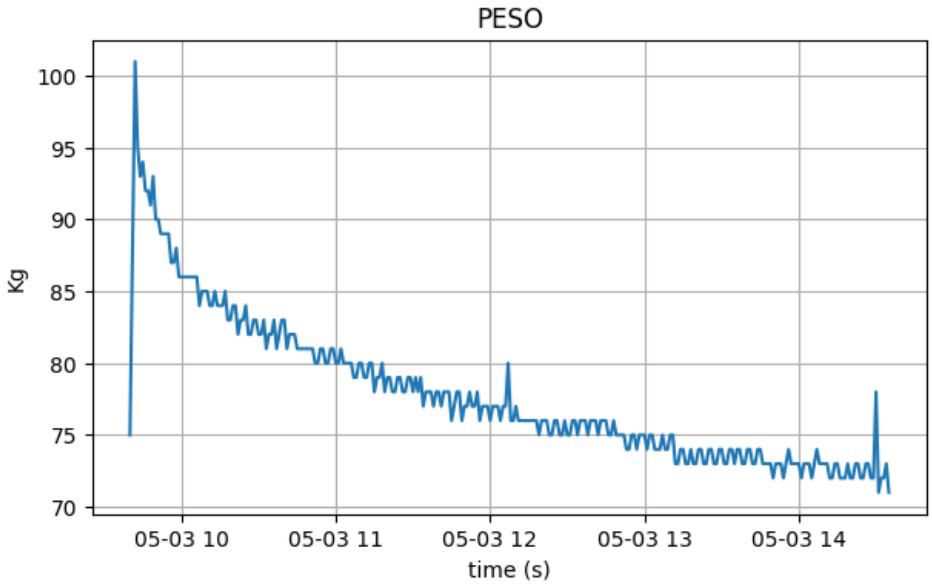
4.3 Exploração e compreensão dos dados

Para a análise inicial dos dados foi gerado o gráfico de Peso × Tempo de todas as máquinas e executada uma inspeção visual com objetivo de identificar os diferentes perfis e partes das séries onde era possível detectar regiões em que ocorreram o entupimento. Durante esta análise, foi descartadas as séries que apresentaram registros muito ruidosos ou inapropriados para este trabalho, aproximadamente 20%.

Nesta fase de análise e interpretação dos dados, foi observado que a maioria dos registros de peso apresentava um intervalo de tempo reduzido, em torno de 1 minuto ou menos, resultando em uma série com uma quantidade significativa de dados redundantes. Esta redundância foi tratada durante a etapa de preparação dos dados.

Adicionalmente, constatou-se que os registros dos pesos apresentavam um alto nível de ruído e uma considerável quantidade de valores inválidos, o que demandou a aplicação de filtro nos dados durante a etapa de preparação dos dados.

Figura 9 – Redundância de dados e ruídos presentes nos registros do peso



Constatou-se também que os sensores de monitoramento de peso frequentemente apresentaram problemas, como mau funcionamento, calibração inadequada, falhas temporárias e desconexões físicas. Nessas circunstâncias, os dados registrados tornaram-se inválidos, exibindo valores extremamente altos ou baixos.

4.4 Preparação dos dados

Como explicado na seção anterior, na análise e interpretação dos dados foram identificadas a necessidade de:

- Remoção de redundância de dados;
- Remoção de valores inválidos;
- Filtros para diminuição de ruído;

A eliminação de redundância foi conduzida empregando a estratégia de periodização da série temporal. Os dados provenientes de uma máquina variavam o intervalo de tempo de 8 segundos a 1 minuto entre registros. Inicialmente, testes foram conduzidos utilizando periodização de 3 e 5 minutos. Posteriormente, foi adotado um período de 10 minutos, resultando em uma redução considerável na quantidade de dados de cada série, sem comprometer a retenção das informações relevantes.

Com o objetivo de minimizar o ruído, empregou-se o filtro de média móvel disponibilizado pela biblioteca *pandas* (MCKINNEY, 2010) do Python, com um parâmetro de janela N. Especificamente, cada valor registrado na série temporal foi substituído pela média dos últimos N valores. Essa função proporcionou uma redução nos valores inválidos registrados. No entanto, séries que apresentavam um número significativo de dados inválidos não foram incluídas no processo de filtragem, esta seleção foi realizada de forma manual.

Aplicou-se também a padronização dos dados utilizando a técnica Z-score, que envolve a transformação dos valores para uma distribuição com média igual a zero e desvio padrão igual a um. Essa padronização foi realizada por meio da classe StandardScaler da biblioteca Scikit-learn (sklearn) (BUITINCK *et al.*, 2013). A normalização da série de dados é uma abordagem útil em diversos casos, especialmente quando os dados apresentam escalas diferentes ou quando a escala absoluta dos valores não é relevante, mas sim a relação entre eles. Vale ressaltar que uma desvantagem significativa dessas transformações é a perda de interpretabilidade dos valores individuais, uma vez que os dados não estão mais expressos nas unidades originais (KUHN; JOHNSON, 2016). Contudo ainda seria possível desfazer a normalização, caso necessário, porém no contexto deste trabalho, o objetivo principal é identificar os diferentes padrões de curvas e não os valores absolutos dos dados.

Para a remoção de dados de valores ocasionados por falha no sensor, ou desconexão, geralmente peso igual 0 ou muito altos, empregou-se a técnica de detecção e remoção de *outliers* baseada nos quartis do conjunto de dados.

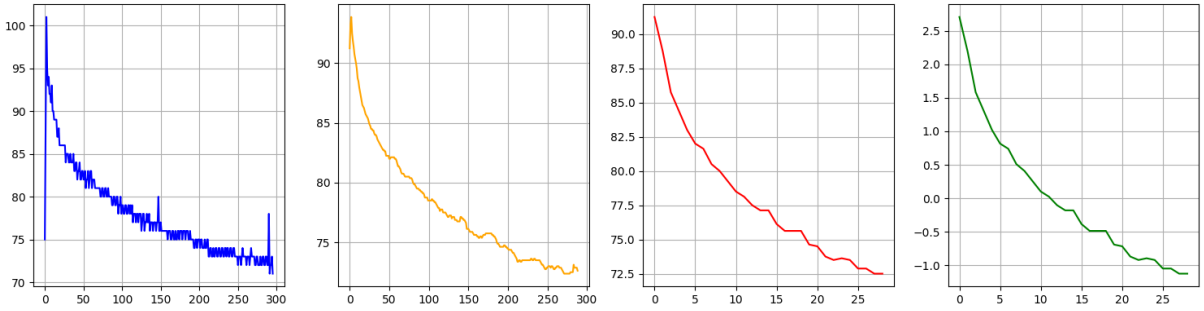
Em resumo, na etapa de preparação dos dados aplicou-se as transformações, na seguinte ordem:

- Detecção e remoção de *outliers*;
- Periodização da série;

- Filtro de média móvel;

- Normalização z-score;

Figura 10 – Transformações na série temporal durante a etapa de preparação dos dados



4.4.1 As sub-séries

Durante a escolha do modelo de aprendizado deste trabalho, que será detalhado na próxima sessão, fez-se necessário divisão dos dados de uma máquina em "sub-séries".

Para este trabalho, uma subsérie é definida como um conjunto de dados ordenados pelo tempo, correspondente ao período entre duas adições consecutivas de peso no sistema. Em outras palavras, uma sub-série representa a série temporal que abrange o intervalo de tempo entre o início de um ciclo da máquina, que ocorre quando o peso é adicionado, até o início do próximo ciclo, quando uma nova adição de peso é realizada.

Assim, foi necessário determinar as subséries a partir da análise exclusiva da série temporal de peso. A identificação foi realizada por meio dos valores da derivada da série, o método *numpy.gradient()* disponibilizado pela biblioteca Numpy [2], permitindo identificar os pontos nos quais o peso aumenta abruptamente. A partir dos índices correspondentes a essas ocorrências na série, foi possível realizar a segmentação das subséries. A Figura 11 ilustra este procedimento.

Em casos em que há ruído nos pontos de inflexão da série, ou seja índices correspondentes à deriva mínima muito próximos, considerou-se o índice de maior valor, ou seja o último índice.

Para obter bons resultados com este método, é fundamental que a série seja previamente filtrada, normalizada e livre de *outliers*.

[2] https://numpy.org/

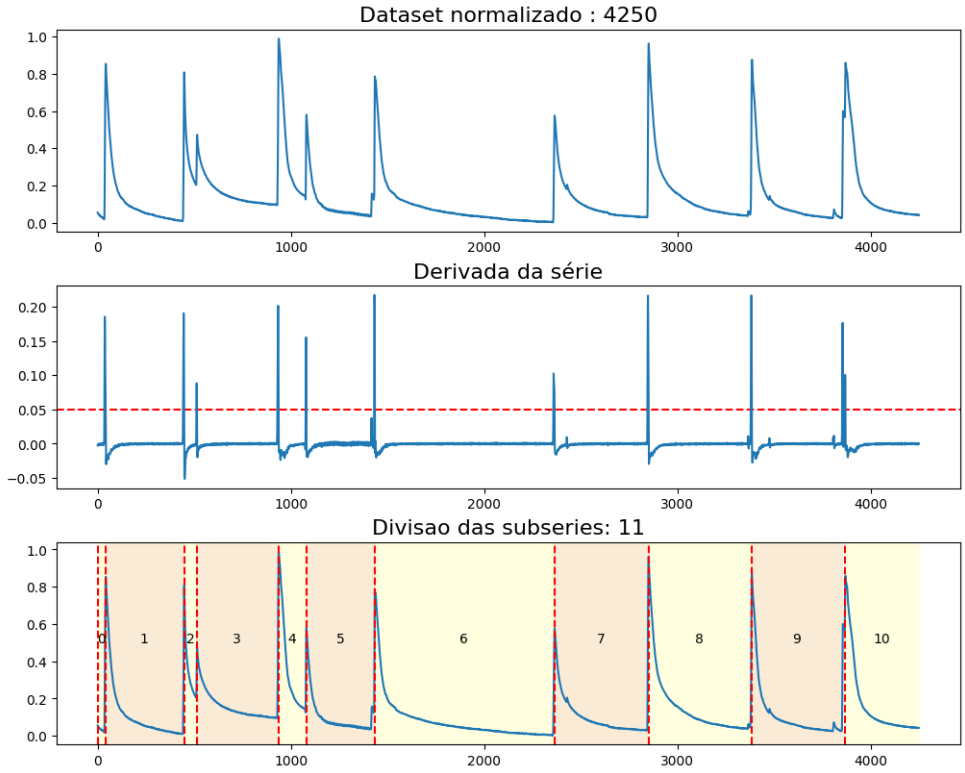


Figura 11 – Detecção de pontos de divisão das subséries

4.5 Exploração e escolha de modelos

Inicialmente, utilizou-se modelos preditivos com o intuito de avaliar a viabilidade de predição em uma série previamente classificada como boa. A proposta inicial consistia em verificar se a discrepância entre os resultados previstos e os valores reais era mínima, indicando um desempenho adequado do equipamento. Por outro lado, um alto erro poderia sinalizar a ocorrência de algum problema na máquina digestora. Contudo, esse modelo não obteve resultados satisfatórios devido à incapacidade de prever corretamente as séries consideradas boas, ou mesmo parte delas. Muitas das séries apresentaram perfis distintos mesmo sem apresentarem problemas. Consequentemente, o erro resultante entre elas foram altos invalidando a ideia inicial. As figuras Figura 12 e Figura 13 exemplificam alguns dos primeiros resultados obtidos.

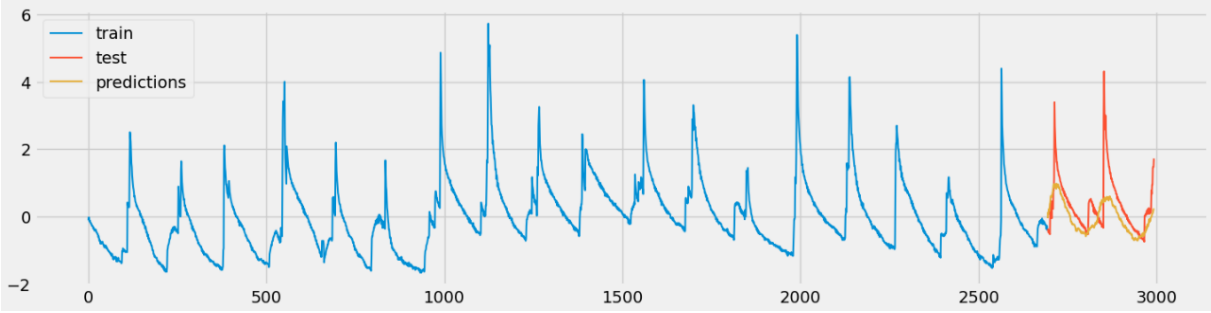


Figura 12 – Primeiros resultados de predição das séries temporais - Perfil 1.

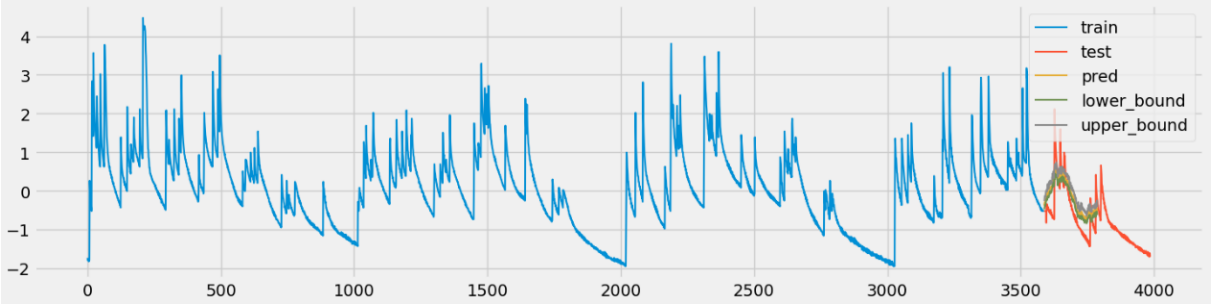


Figura 13 – Primeiros resultados de predição das séries temporais - Perfil 2.

A segunda abordagem foi utilizar o algoritmo k-means para agrupar as séries em *clusters*. O procedimento k-means é facilmente programado e é computacionalmente econômico, de modo que é possível processar amostras muito grandes em um computador com pouco processamento. Possíveis aplicações incluem métodos para agrupamento de similaridade, predição não linear, aproximação de distribuições multivariadas e testes não paramétricos para independência entre diversas variáveis (MACQUEEN, 1967).

Uma das principais vantagens do k-means é a capacidade de agrupar dados sem a necessidade de pré-classificação das amostras para o treinamento. No entanto, o k-means requer um tamanho fixo de dimensionalidade dos dados de entrada, o que pode ser problemático quando se lida com séries temporais com tamanhos variáveis.

Para incluir séries temporais em modelos estatísticos que são exclusivamente definidos em variáveis univariadas, é necessário extrair características das séries temporais. Isso denota mapeamentos das séries temporais de $\mathbb{R}^L$ para $\mathbb{R}$, a fim de obter variáveis univariadas. Um processo automatizado de agregação das séries temporais em características significativas permite o uso da teoria bem desenvolvida de aprendizado de máquina supervisionado e não supervisionado em dados temporais. (CHRIST; KIENLE; KEMPA-LIEHR, 2016)

Sendo assim, para contornar a limitação de séries temporais de tamanhos distintos, é possível extrair atributos relevantes destas e usá-los como entrada para o algoritmo de k-means. Esses atributos podem incluir medidas estatísticas, características de frequência ou outras representações que capturam informações essenciais das séries. Ao utilizar esses atributos, as séries temporais podem ser representadas por vetores de características de tamanho fixo, tornando-as adequadas para o k-means.

Inicialmente os atributos das séries escolhidos foram:

- Média;
- Desvio Padrão;
- Valor Mínimo;
- Valor Máximo;

Durante os testes iniciais, quando a qualidade do agrupamento foi avaliada nas mesmas subséries (geradas conforme descritas na seção anterior) usadas para criar os grupos, foi observado um desempenho positivo do algoritmo de agrupamento. Ou seja, as séries que apresentavam predominância de entupimento foram agrupadas em sua maioria um mesmo *cluster*. A Figura 14 e Figura 15 ilustram alguns resultados obtidos:

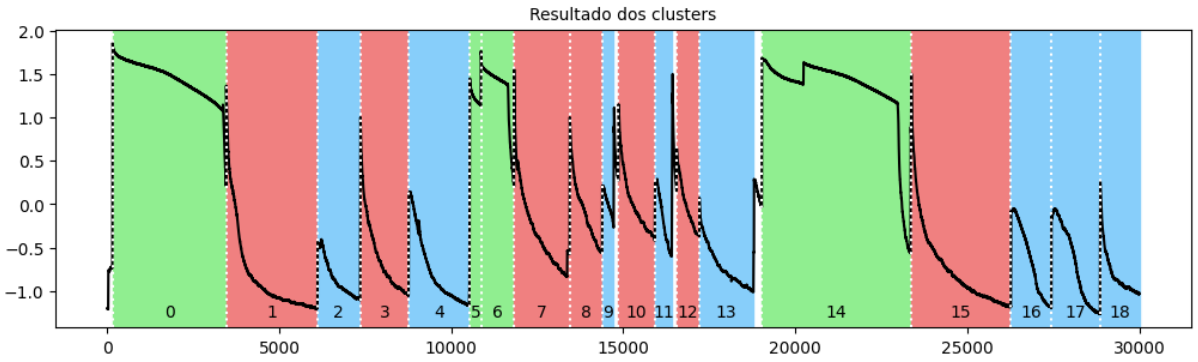


Figura 14 – Séries agrupadas com k-means e atributos básicos.

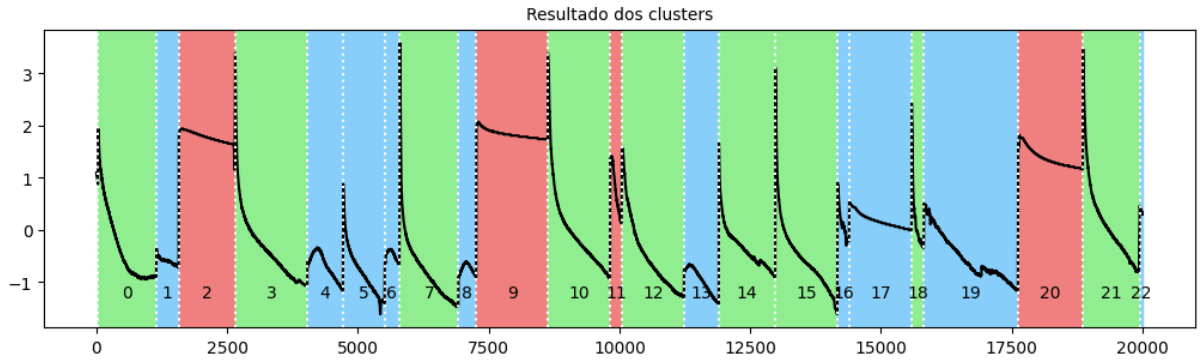


Figura 15 – Séries agrupadas com k-means e atributos básicos.

Quando séries distintas foram utilizadas nas gerar os grupos e analisar a qualidade, os resultados do agrupamento não alcançaram níveis satisfatórios. Consequentemente, foi necessário aprofundar o estudo dos atributos das séries utilizados para realizar o agrupamento.

A biblioteca de aprendizado de máquina "tsfresh", baseada em Python, é uma biblioteca de aprendizado de máquina rápida e padronizada para extração e seleção automática de características de séries temporais (CHRIST *et al.*, 2018). Esta biblioteca foi utilizada para coletar atributos mais significantes a serem enviado ao algoritmo k-means.

Os atributos a serem extraídos foram:

ts.mean_abs_change(subseries): Média sobre as primeiras diferenças. Retorna a média das diferenças absolutas entre os valores de séries temporais subsequentes.

ts.mean_change(subseries): Média ao longo das diferenças de séries temporais. Retorna a média das diferenças entre os valores de séries temporais subsequentes.

ts.index_mass_quantile(subseries, {"q": .4}): Calcula o índice relativo i da série temporal x onde q é porcentagem da massa de x que fica à esquerda de i. Neste caso, para q = 40%, esta método retornará valor próximo ao centro de massa da série temporal.

ts.quantile(subseries, 0.6): Método que calcula o quantil q = 0.6 da série temporal.

ts.quantile(subseries, 0.9): Método que calcula o quantil q = 0.9 da série temporal.

ts.time_reversal_asymmetry_statistic(subseries, 3): Este método é usado para avaliar a assimetria de reversão no tempo de uma série temporal. Em termos simples, a estatística de assimetria de reversão no tempo tenta quantificar diferença no comportamento da série quando avançamos no tempo em comparação com quando

retrocedemos no tempo, calculando uma métrica que reflete como a série e sua versão invertida no tempo diferem uma da outra.

ts.change_quantiles(subseries, ql=0.2, qh=0.8, isabs=True, f_agg ='mean'): Método utilizado para avaliar como os quantis de uma série temporal mudam ao longo do tempo. Calcula a diferença entre os quantis de uma parte inicial (ql=0.2) da série temporal e os quantis de uma parte subsequente (qh=0.8) da série temporal. Considerando as diferenças absolutas (isabs=True) e utilizando a média como função agregadora aplicada às diferenças (f_agg ='mean').

ts.change_quantiles(subseries, ql=0.1, qh=0.6, isabs=False, f_agg ='var') : Análogo à função anterior com ql=0.1, qh=0.6, desconsiderando as diferenças absolutas (isabs=False) e utilizando a variância como função agregadora aplicada às diferenças (f_agg ='var').

Estes atributos foram selecionados a partir de uma função da biblioteca tsfresh referenciada com *extract_relevant_features()*, a qual recebe como parâmetros de entrada algumas das subséries temporais e a classificação desejada de cada uma delas. O retorno desta função lista os métodos da biblioteca tsfresh que, aplicados às séries de entrada, foram mais relevantes para se obter a classificação desejada. Neste trabalho, usamos o rótulo do grupo de cada subsérie como classificação desejada.

Ao executar *extract_relevant_features()* com subséries pré-selecionas e classificadas, a função retornou de 30 a 70 métodos considerados muito relevantes para a classificação desejada. Como o k-means depende das distâncias euclidianas para atribuir pontos aos centroides, ele não é ideal para ser utilizado em conjunto de dados com muitos atributos, pois os pontos podem parecer igualmente distantes uns dos outros, resultando em agrupamentos menos precisos e mais dispersos, este fato é conhecido como a "Maldição da dimensionalidade"(RADOVANOVIć; NANOPOULOS; IVANOVIć, 2010).

Para contornar essa limitação, optou-se por selecionar apenas alguns dos atributos retornados pela função. Essa seleção foi realizada com base nos melhores resultados obtidos pelo k-means ao ser aplicado a esses atributos específicos. Dessa forma, reduziu-se a complexidade do espaço de atributos.

Por fim, foram incluídos mais dois conjuntos de atributos das subséries para melhorar a qualidade do classificador:

- Os coeficientes da curva logarítmica que mais se aproxima da subsérie
- O *p-valor* do teste de Kolmogorov-Smirnov para a distribuição exponencial da subsérie.

Esses dois conjuntos de atributos tiveram como objetivo parametrizar e quantizar a subsérie por meio de uma função exponencial.

A subsérie demonstra um comportamento ideal quando exibe uma proximidade significativa com uma função exponencial. Entretanto, em situações de entupimento, ela se distancia do padrão exponencial e tende a assumir características semelhantes a uma reta ou uma distribuição Gamma.

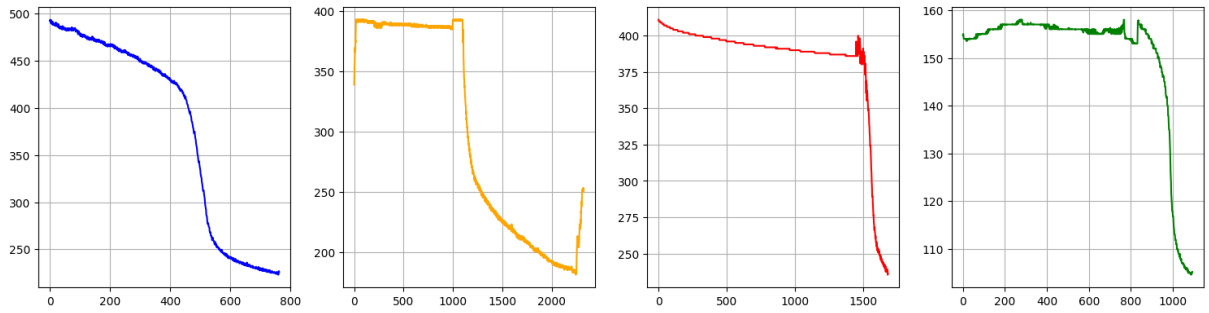


Figura 16 – Perfis de subséries com entupimento.

Ao ajustar os dados da subsérie à função logarítmica $a \cdot \log(x) + b$, obtém-se os valores de a e b. A partir dos valores de a é possível inferir a tendência da série. Os valores de b correspondem ao deslocamento vertical (*offset*) da tendência, porém estes não são relevantes para esta análise.

Parâmetro	Valor	Curva
a	~ -1	Exponencial
a	~ 0	Reta
a	> 0	Logarítmica

Tabela 1 – Exemplos de Curvas

O teste de Kolmogorov-Smirnov é uma técnica não paramétrica usada para comparar uma distribuição empírica com uma distribuição teórica. Neste trabalho, ele é aplicado para avaliar o grau de proximidade dos dados de uma subsérie com uma distribuição exponencial, com base no *p-valor* resultante do teste.

Um *p-valor* próximo de 1 indica que os dados da subsérie se aproximam de uma distribuição exponencial, enquanto um *p-valor* próximo de 0 caracteriza dados que não seguem o padrão esperado de uma distribuição exponencial.

4.6 Apresentação da solução

Definido e parametrizados os atributos que serão extraídos de cada subsérie, eles são passados como parâmetros para o método não supervisionado k-means.

Para melhorar a eficácia do algoritmo, foi essencial limitar a quantidade de dados utilizados. Subséries com um grande número de dados podem distorcer os valores dos atributos resultantes. Porém uma vez que a série foi periodizada e padronizada durante o pré-processamento dos dados, tem-se os dados necessário para a análise sempre no início de cada processo.

Outro ajuste importante realizado é a padronização da subsérie. Durante a etapa de pré-processamento, a série completa é normalizada, e, posteriormente quando a série é dividida em subséries, cada uma delas é novamente normalizada individualmente.

A seguir descreve-se o algoritmo implementado para a classificação dos dados de uma máquina digestora de resíduos orgânicos.

Data: Dados de peso no tempo de uma máquina digestora de resíduos orgânicos
Result: Dados agrupados em 3 clusters.

$original\_data \longleftarrow carregaDados()$;
$pre\_data \longleftarrow removeOutliers(original\_data)$;
$pre\_data \longleftarrow periodizacao(pre\_data)$;
$pre\_data \longleftarrow normalizacao(pre\_data)$;
$pre\_data \longleftarrow filtro(pre\_data)$;
$subseries \longleftarrow encontraSubseries(pre\_data)$;
while *subseries* **do**
    $subserie\_normalizada \longleftarrow normalizacao(subseries)$;
    $subserie\_norm\_filtrada \longleftarrow filtro(subserie\_normalizada)$;
    $atributos\_subseres \longleftarrow extraiAtributos(subserie\_norm\_filtrada)$;
    $y \longleftarrow adiciona(atributos\_subseres)$;
end
$clusters\_result \longleftarrow k\_mean\_predict(y)$;
$atribui(originalData, clusters\_result)$

Algoritmo 1: Algoritmo de agrupamento dos dados de uma máquina digestora de resíduos orgânicos

5 AVALIAÇÃO EXPERIMENTAL

Neste capítulo, será apresentada uma descrição mais detalhada da implementação do algoritmo. Serão abordados o conjunto de dados utilizado, os parâmetros adotados e os resultados obtidos.

5.1 Conjuntos de Dados

Neste estudo, foram fornecidos dados de 80 máquinas, totalizando cerca de 2.3 gigabytes. Dentro desse conjunto, foi realizada uma seleção das máquinas que predominantemente possuíam dados válidos com menor presença de ruído. Esse processo resultou em dados provenientes de 40 máquinas distintas. Dados foram considerados válidos quando as séries temporais das subséries apresentavam uma concordância significativa com os padrões dos grupos. Caso esses padrões não fossem claramente identificáveis por meio de uma inspeção visual inicial da série temporal da máquina, os dados eram descartados. Na Figura 17, exemplos de séries descartadas são ilustrados. No gráfico verde, nota-se um padrão comportado na série temporal, porém que não se encaixa nos grupos definidos para este estudo. No gráfico vermelho, os pontos de divisão das subséries são de difícil identificação. No gráfico azul, observa-se alguns dos padrões desejados, mas o sinal possui um ruído significativo.

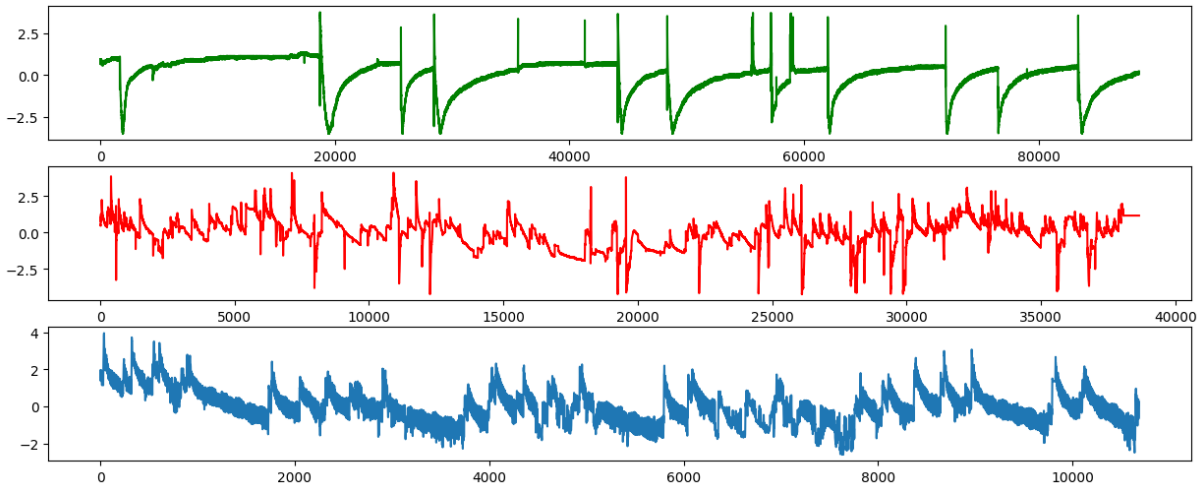


Figura 17 – Séries temporais descartadas.

As séries temporais de cada máquina possuíam períodos variando de 8 segundos a 50 segundos. Para padronizar os dados, todas as bases foram periodizadas para intervalos de 1 minuto, gerando uma cópia de cada base de dados. Ao final, obteve-se um total de 280 megabytes de dados periodizados, que foram utilizados para a etapa de agrupamento.

5.2 Configuração Experimental

Durante o desenvolvimento do projeto, foram estabelecidas algumas abordagens e limitações com o objetivo de obter resultados mais precisos. Cada uma dessas decisões será explicada a seguir:

- Janela de dados de cada subsérie = 50 a 150 registros

  Conforme mencionado no capítulo anterior, verificou-se que subséries com muitos dados distorciam o resultado de seus atributos. Uma subsérie com muitos valores indica que o tempo entre duas adições consecutivas de material orgânico em uma máquina foi muito longo. O processo de decomposição do material é iniciado na adição do peso, assim o sistema trabalha por um tempo e depois para. O restante do tempo, na qual ainda são armazenados seus dados, a máquina está parada, não sendo necessário a análise e verificação de seus valores de peso nesse período.

  Já subséries com poucos dados informam que foi adicionado peso à máquina de maneira consecutiva em um curto intervalo de tempo. Avaliar a primeira subsérie temporal não se torna mais necessário já que a próxima subsérie conterá o peso da primeira subsérie juntamente ao peso que foi adicionado.

  Para este trabalho definiu-se a quantidade entre 50 e 150 pontos de dados. Ou se seja, subséries com menos de 50 valores foram descartados, assim como os registros após o item 150 das subséries maiores. A Figura 18 uma subsérie de 48 dados seguida de uma com mais de 3500 registros.

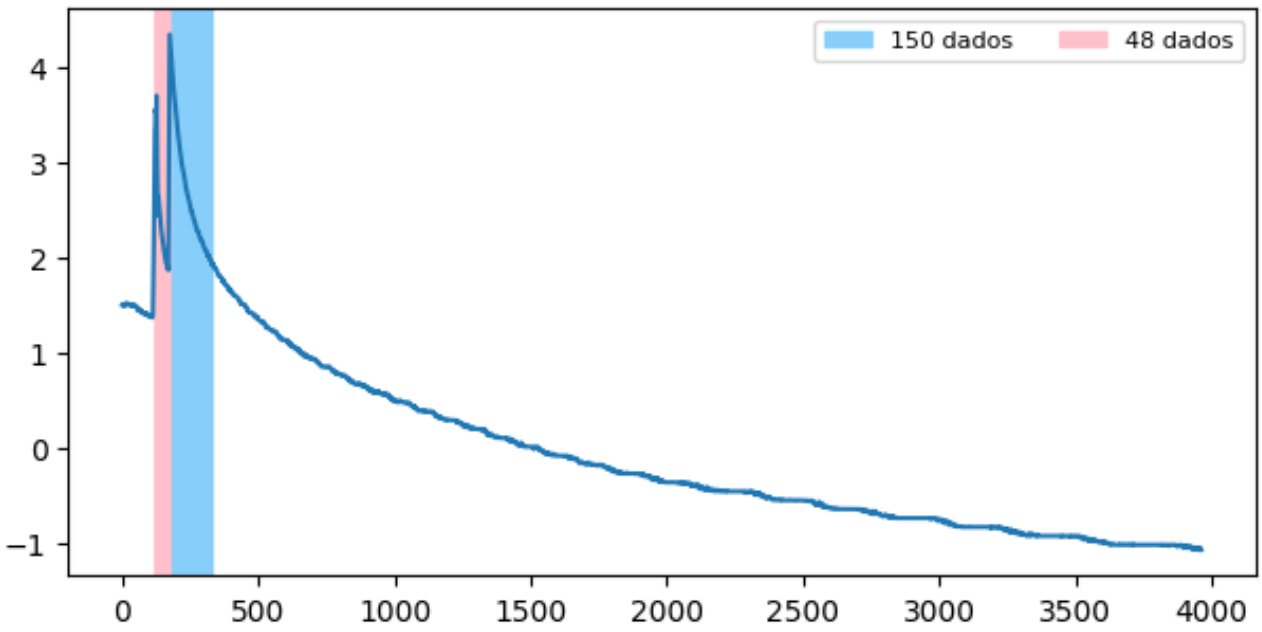


Figura 18 – Divergências de tamanhos entre subséries.

- Período = 1 minuto:

  A decisão de manter a periodização em intervalos de 1 minuto para os dados das séries temporais foi tomada visando tornar a verificação do grupo do sistema mais rápida quando implementada em tempo real. A janela de dados das subséries foi definida entre 50 a 150, isso implica que o algoritmo de atribuição do grupo deve aguardar pelo menos 50 minutos para iniciar a verificação. Se optássemos por periodizar os dados com intervalos maiores, o tempo de espera para a verificação aumentaria proporcionalmente.

- Filtro da série = 3 e filtro das subséries = 10:

  Após normalização da série temporal total é aplicado um filtro com parâmetro de janela igual 3. Esse valor foi escolhido para eliminar pequenos ruídos sem prejudicar a etapa de extração das subséries. Se aplicado filtros com janela maiores os pontos de inflexão das subséries que caracterizam a adição de peso na máquina poderiam ser mascarados. Assim, após a definição das subséries, optou-se por aplicar um filtro de janela igual a 10 para a suavização dos dados e a redução de ruídos.

- Normalização das subséries:

  A normalização das subséries foi implementada utilizando a técnica Z-score, a mesma utilizada na série temporal completa no início da preparação dos dados. Porém nesta etapa considerou-se também a subsérie anterior. Ou seja, os dados da subsérie corrente foram padronizados considerando os dados da subsérie em questão juntamente com os dados da subsérie anterior. Esse método permitiu que a subsérie corrente mantivesse sua "forma"durante a normalização, preservando o seu padrão original. A Figura 19 ilustra esse processo quando os normalizada unitariamente e considerando os dados da subsérie anterior.

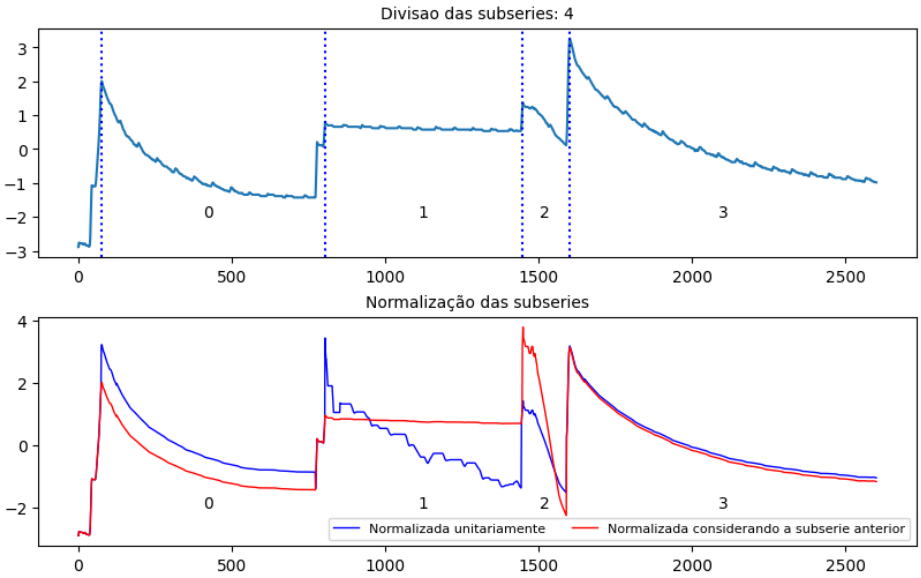


Figura 19 – Normalização unitária das subséries

- Deslocamento das subséries:

  Com o objetivo de tornar os dados que serão utilizados na geração dos atributos o mais homogêneo possível, após a normalização das subsérie, seus dados foram deslocados verticalmente para o eixo 0. Ou seja, o menor valor de cada subsérie será sempre zero. A Figura 20 ilustra primeiramente cada subsérie dividida e normalizada e posteriormente os dados limitados e deslocados para o eixo 0.

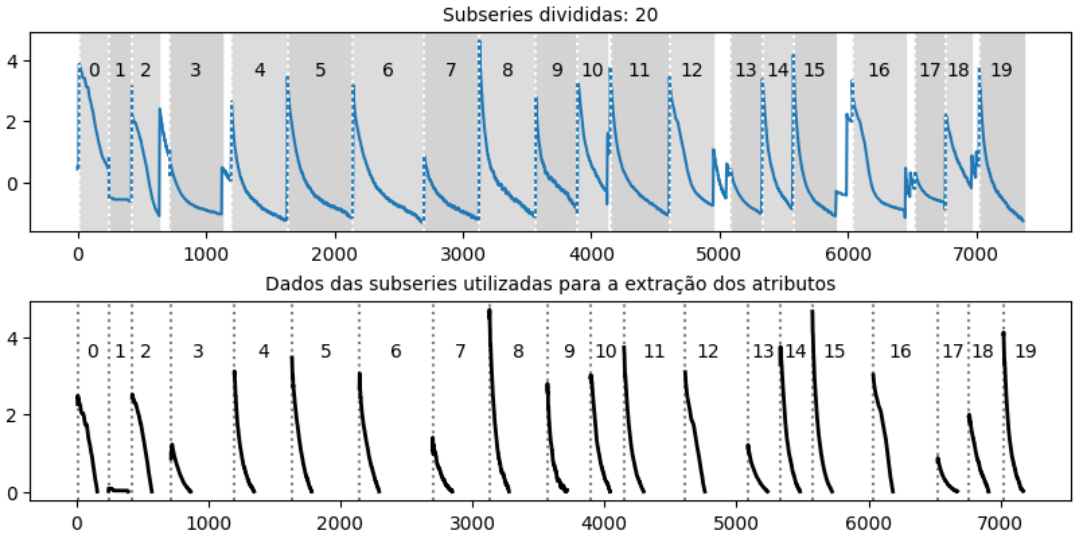


Figura 20 – Deslocamento dos dados das subséries.

- Número de grupos de classificação (*clusters*) = 3:

  O objetivo proposto para este trabalho é a detecção de ocorrências de entupimento da maquina digestora de resíduos orgânico pela análise do histórico de peso de cada

máquina. Logo deveria-se agrupar cada subsérie em um de 2 grupos: "Apresenta entupimento"e "Não apresenta entupimento". Porém durante o desenvolvimento deste trabalho percebeu-se que o comportamento dos processos seriam mais bem representados em 3 grupos:

Grupo 1: Processos em que o conteúdo da máquina é consumido rapidamente e o peso proporcional decai mais rapidamente em um curto período de tempo. Um exemplo está representado pela curva azul na Figura 21.

Grupo 2: Processos em que o conteúdo da máquina é processado em um tempo maior, porém o conteúdo é consumido corretamente. Um exemplo está representado pela curva amarela na Figura 21.

Grupo 3: Processos em que não se detecta o decaimento do peso mais acentuado no início, em alguns casos este decaimento é constante, e podendo até aumentar. Estes são os casos em que há entupimento. As curvas vermelha e verde da Figura 21 ilustram estes perfis.

Diante desta constatação optou-se por utilizar o número de *clusters* para o classificador k-means igual 3, sendo o grupo 1 e 2 sem entupimento e o grupo 3 com entupimento.

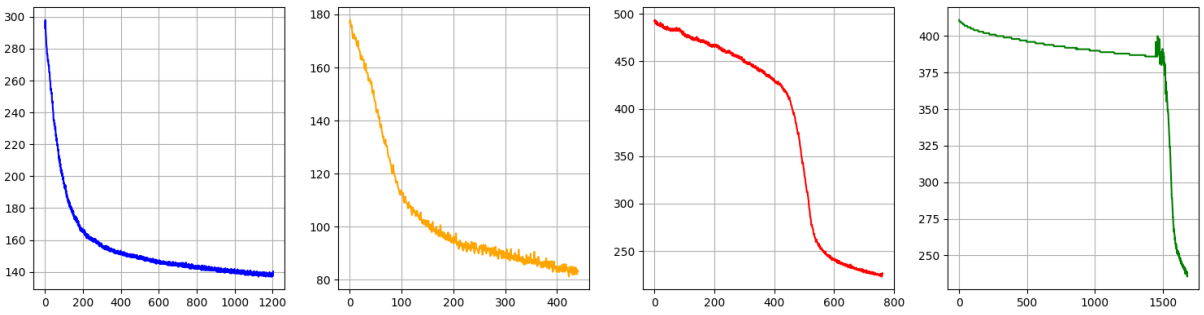


Figura 21 – Perfis de comportamento do peso

- Seleção do conjunto de treino:

  Para formar os dados do conjunto de treino foram selecionadas 32 subséries de diferentes máquinas, de tamanhos e amplitudes variadas. Estas subséries foram concatenadas para formar uma série temporal única. Neste caso, não foi necessário aplicar a remoção de *outliers*, filtro na série temporal completa, periodização e a detecção de subséries baseada na derivada. Executou-se apenas a normalização e filtro de cada subsérie, por fim extraiu-se os atributos selecionados para o algoritmo k-means, e executou os métodos de criação e atribuição de grupos. A Tabela 2 seguir descreve as características das subséries utilizadas nesta etapa de treino e suas respectivas classes.

Os grupos estão enumeradas de 0 a 2 e correspondem aos sequencialmente aos grupos 1, 2 e 3, explicados anteriormente.

Tabela 2 – Características das subséries utilizados na etapa de treino.

Subsérie	Quantidade	Mínimo	Máximo	Excursão	Cluster
0	763	223.0	493.6	270.6	2
1	1413	144.0	376.0	232.0	1
2	2320	181.0	393.0	212.0	2
3	1544	156.0	338.0	182.0	1
4	1682	235.4	411.0	175.6	2
5	1317	155.0	328.0	173.0	0
6	1538	142.0	314.4	172.4	0
7	1204	137.0	298.0	161.0	0
8	2827	183.0	324.0	141.0	2
9	817	258.0	397.0	139.0	1
10	489	202.7	327.6	124.9	2
11	1013	141.0	243.6	102.6	0
12	405	86.4	184.1	97.7	1
13	441	81.2	177.9	96.7	0
14	665	258.0	344.0	86.0	1
15	443	84.6	159.2	74.6	0
16	411	104.0	173.6	69.6	0
17	1250	274.0	329.0	55.0	0
18	1093	104.6	158.0	53.4	2
19	916	246.0	297.0	51.0	1
20	914	0.0	51.0	51.0	1
21	397	106.0	153.0	47.0	2
22	647	102.0	148.0	46.0	2
23	210	108.7	147.0	38.3	0
24	168	145.5	177.1	31.6	0
25	206	114.0	144.5	30.5	0
26	85	152.9	174.0	21.1	0
27	318	292.0	310.6	18.6	1
28	86	152.1	169.9	17.8	0
29	525	357.0	373.0	16.0	1
30	1464	252.0	354.0	102.0	1
31	321	210.0	227.0	17.0	2

- K-means:

  O algoritmo k-means é um método de aprendizado de máquina não supervisionado utilizado para agrupar um conjunto de dados em grupos, denominados *clusters*. O algoritmo opera da seguinte forma: Inicialmente, k centros são escolhidos de forma aleatória, em que cada objeto inicialmente atua como uma média ou centro de um *cluster*. Em seguida, para cada um dos objetos restantes, com base na distância entre

o objeto e o centro, o objeto é atribuído ao *cluster* mais próximo. Posteriormente, um novo centro é calculado para cada *cluster*, e esse processo se repete até que a função de critério convirja (MALKI *et al.*, 2016).

O algoritmo k-means utilizado neste trabalho foi o método *sklearn.cluster.KMeans()* disponibilizada pela biblioteca scikit-learn. Os parâmetros utilizados neste método foram:

- n_clusters = 3: O número de *clusters*, bem como o número de centróides a serem gerados.
- init = 'k-means++': Seleciona centróides iniciais de *cluster* usando amostragem com base em uma distribuição de probabilidade empírica da contribuição dos pontos para a inércia geral. Essa técnica acelera a convergência.

Os outros parâmetros não foram inicializados, sendo assim o método foi executada com valores *default* (BUITINCK *et al.*, 2013).

5.3 Resultados e Discussões

Nesta seção, serão expostos os resultados alcançados por meio do algoritmo definido nas seções anteriores, seguidos de uma análise detalhada desses resultados.

5.3.1 Resultados

Para a execução do teste cada série continha cerca de 7200 pontos de dados, o que equivale a 5 dias de dados periodizados a cada minuto. Para processar essa série, foram aplicadas etapas de remoção de outliers, normalização e filtragem. A partir dessa série transformada, as subséries foram identificadas utilizando os pontos de inflexão, os quais foram determinados com base nos valores das derivadas. Cada uma das subséries foi então submetida a um processo adicional de normalização e filtragem. Um método extraiu seus atributos, e esses atributos foram posteriormente passados para o algoritmo de agrupamento k-means.

Resultados dos teste:

- Dados processados: 7.260.204;
- Subséries agrupadas: 14.063;
- Cluster 0: 3.361;
- Cluster 1: 6.770;
- Cluster 2: 3.932;

- Imagens geradas: 836;

- Tempo de execução: 12 minutos;

Dado que o *k*-means é um algoritmo de aprendizado não supervisionado, foi necessário estabelecer um procedimento para avaliar os resultados alcançados. Após a conclusão do teste de atribuição de grupos, uma imagem foi gerada com o conjunto de dados avaliados, apresentando o resultado de cada subsérie juntamente com o respectivo *cluster*. Essa imagem foi o método utilizado a verificação dos resultados do algoritmo.

Para cada *cluster* foi atribuído uma cor para facilitar a verificação:

- Cluster 0: subsérie classificada no Grupo 1 - cor azul;

- Cluster 1: subsérie classificada no Grupo 2 - cor verde;

- Cluster 2: subsérie classificada no Grupo 3 - cor vermelha;

As imagens geradas na execução do algoritmo contém dois gráficos. No primeiro tem-se a série original avaliada, no segundo é apresentado as subséries divididas e normalizadas em branco, enumeradas, e também os dados de cada subsérie que foram utilizados na extração dos atributos. O fundo de cada gráfico é colorido de acordo com o seu *cluster* resultante. A Figura 22, Figura 23, Figura 23 e Figura 24 ilustram a apresentação final dos resultados de algumas séries:

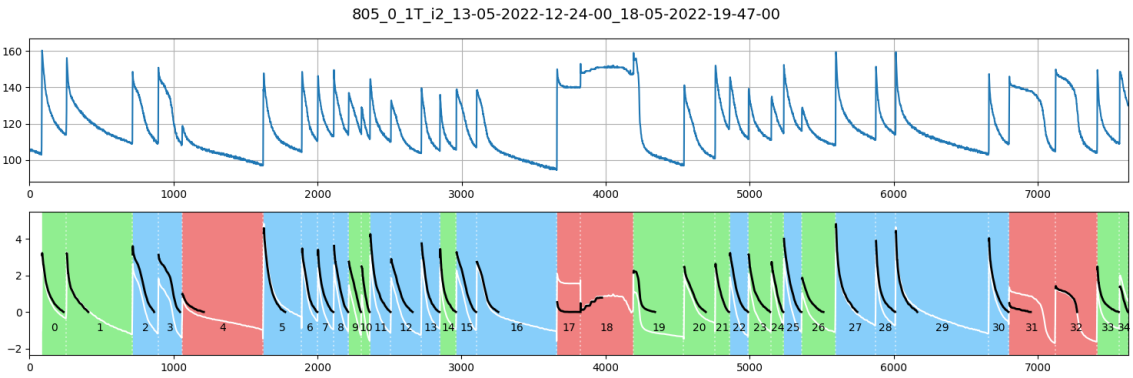


Figura 22 – Exemplo 1 de resultado do algoritmo de agrupamento.

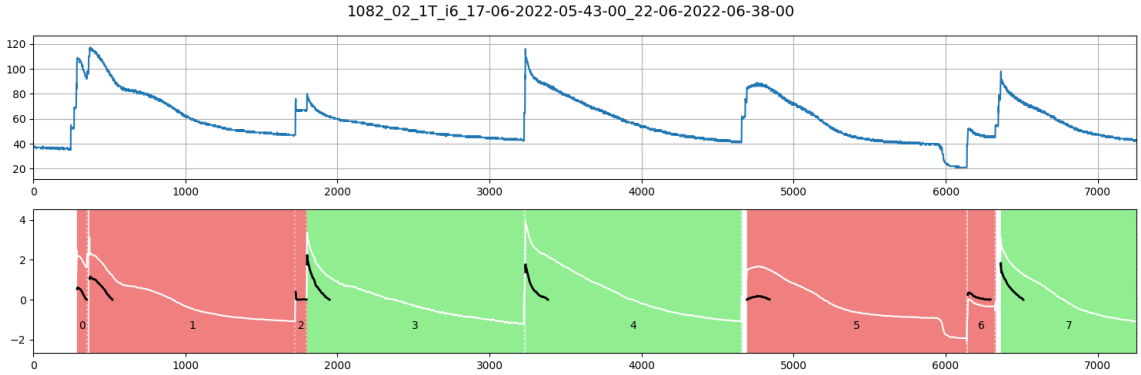


Figura 23 – Exemplo 2 de resultado do algoritmo de agrupamento.

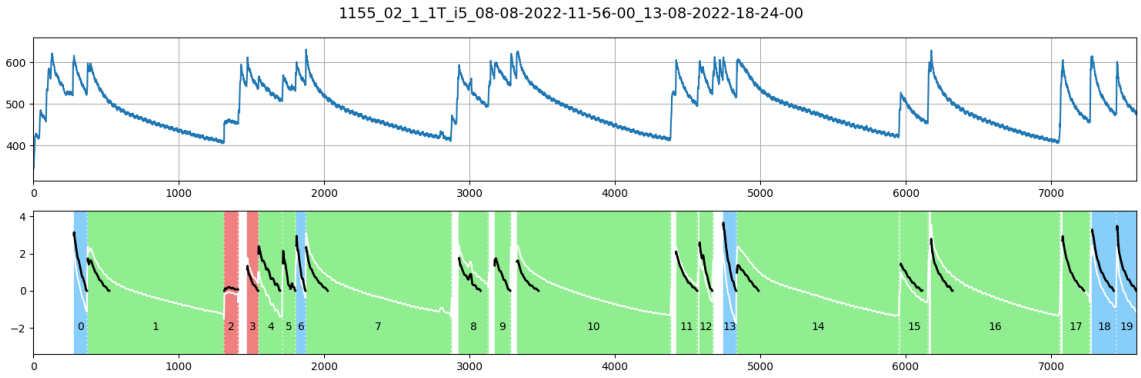


Figura 24 – Exemplo 3 de resultado do algoritmo de agrupamento.

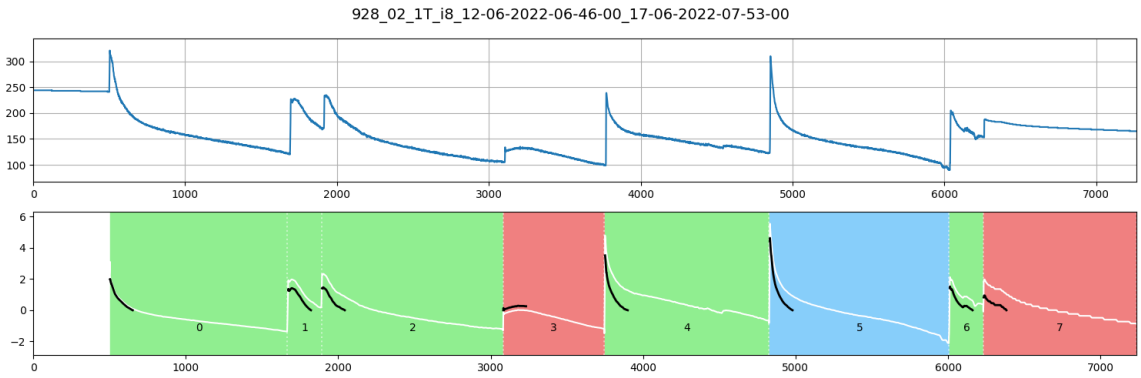


Figura 25 – Exemplo 4 de resultado do algoritmo de agrupamento.

5.3.2 Validação dos Resultados

Para validar os resultados, empregou-se a seguinte abordagem: selecionou-se aleatoriamente 20 séries a partir das imagens geradas pelo algoritmo de agrupamento. Cada série foi então subdividida em subséries e analisadas pela autora deste projeto, que realizou o agrupamento manual das subséries em duas categorias: "Sem entupimento"e "Com entupimento". O método proposto foi aplicado novamente, agrupando também as

mesmas subséries. As subséries agrupadas pelo método foram re-agrupadas em "Sem entupimento"para as subséries resultantes dos *clusters* 0 e 1, e "Com entupimento"para as subséries agrupadas no *cluster* 2.

5.3.2.1 Validação 1:

- Subséries agrupadas: 337
- Subséries "Sem entupimento": 240
  - agrupadas corretamente pelo algoritmo: 206
- Subséries "Com entupimento": 97
  - agrupadas corretamente pelo algoritmo: 64

A matriz de confusão da Figura 26 resume os resultados obtidos:

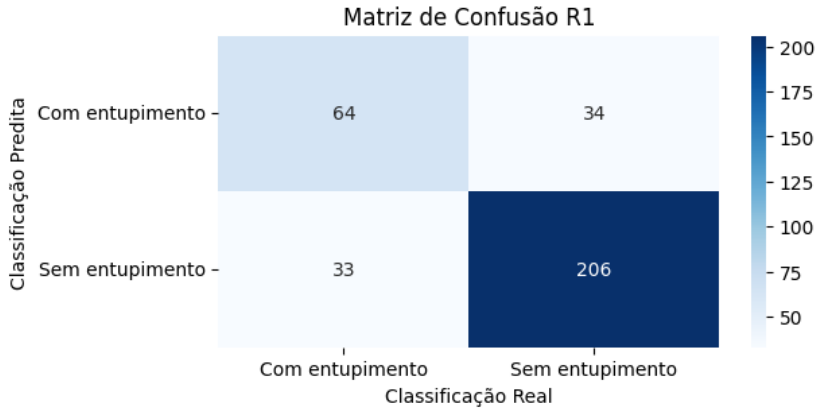


Figura 26 – Matriz de confusão dos resultados da validação 1.

A acurácia obtida neste experimento foi de 80%.

Para criar uma representação visual dos resultados obtidos na Validação 1, aplicou-se a técnica de Análise de Componentes Principais (PCA, do inglês Principal Component Analysis) para reduzir a dimensionalidade do conjunto de dados original, que continha 337 subséries com 10 atributos cada, para apenas 2 atributos. Isso permitiu a geração do gráfico Figura 27 que ilustra os resultados da Validação 1.

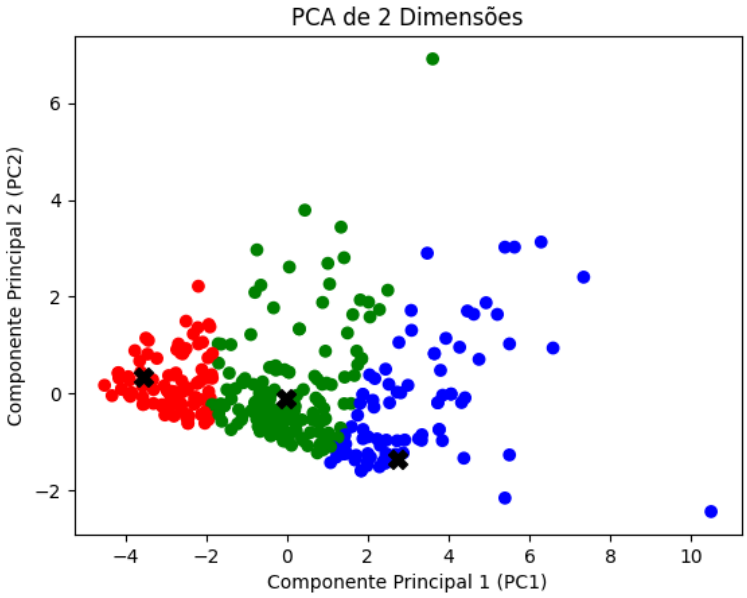


Figura 27 – Gráfico PCA dos atributos das subséries dos conjunto de Validação 1 com o resultados de seus clusters.

5.3.2.2 Validação 2:

- Subséries agrupadas: 319
- Subséries "Sem entupimento": 236
  - agrupadas corretamente pelo algoritmo: 213
- Subséries "Com entupimento": 83
  - agrupadas corretamente pelo algoritmo: 51

A matriz de confusão Figura 28 resume os resultados obtidos:

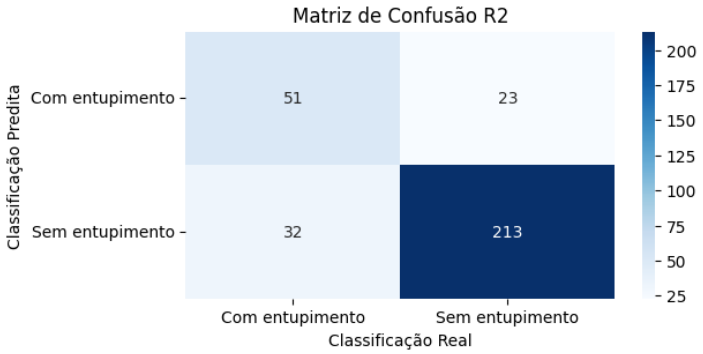


Figura 28 – Matriz de confusão dos resultados da validação 2.

A acurácia obtida neste experimento foi de 83%.

A Figura 29 ilustra uma representação visual dos resultados obtidos na Validação 2, pela a técnica PCA.

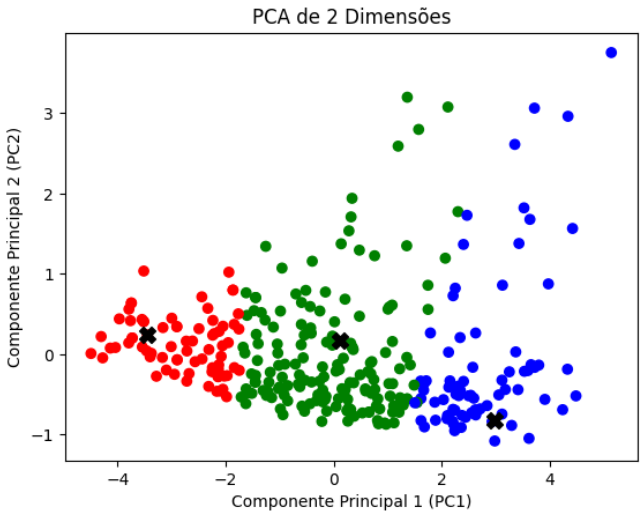


Figura 29 – Gráfico PCA dos atributos das subséries dos conjunto de Validação 2 com o resultados de seus clusters.

5.3.3 Análise dos resultados

Na análise dos resultados em que o algoritmo agrupou as subséries de maneira incorreta foram identificados algumas razões:

- Ruído na série temporal:

  Foram identificadas situações em que ruídos significativos no sinal induziam o algoritmo a detectar erroneamente um ponto de divisão para uma nova subsérie. Nesses casos, o algoritmo tratava esses dados como pertencentes a uma nova subsérie, e o comportamento atípico desses dados muitas vezes indicava erroneamente a presença de entupimento. Isso ocorria porque, na realidade, um novo processo não havia sido de fato iniciado, resultando em uma não conformidade na queda de peso esperada.

  No exemplo da Figura 30 pode-se perceber que a subsérie 3 é na realidade a continuação da subsérie 2.

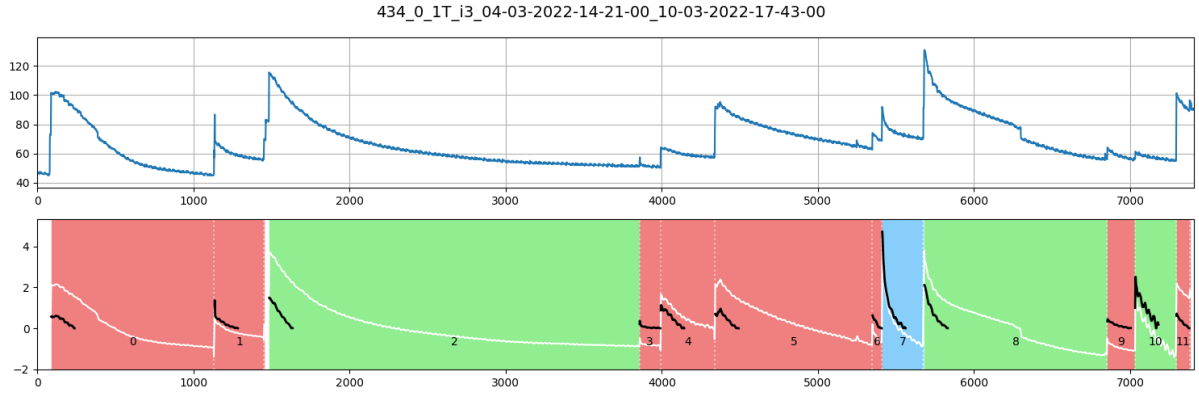


Figura 30 – Divisão incorreta da subsérie 3.

- Remoção de *outliers*:

  Em séries temporais que não possuíam ruídos excessivos, a etapa de remoção de *outliers* resultou na eliminação dos dados com valores mais elevados. Em alguns casos, isso resultou na exclusão dos dados de valores altos que, de fato, indicavam entupimento. Dessa forma, ao efetuar a classificação da série, somente os dados menores foram considerados, levando o algoritmo a não detectar o entupimento que estava presente. Na Figura 31 pode-se perceber que a subsérie 3 mudou de comportamento devido a remoção dos *outliers* da série temporal.

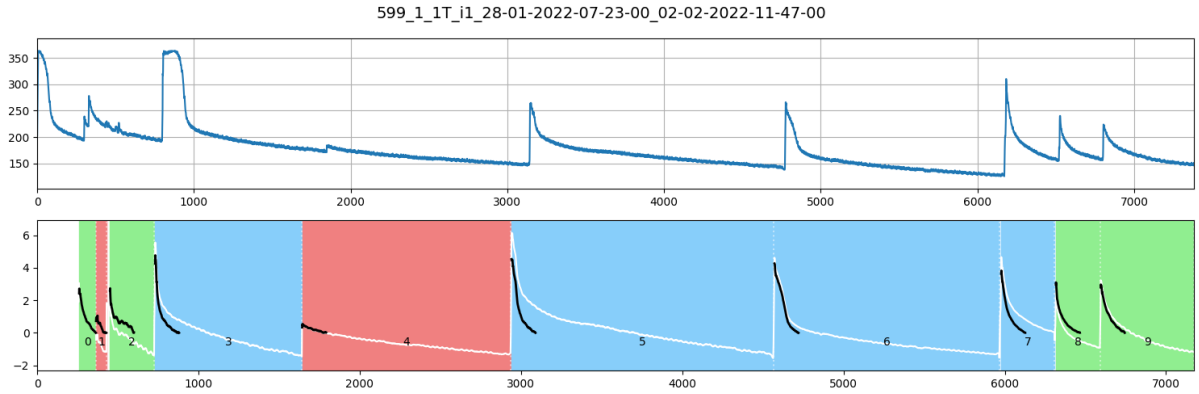


Figura 31 – Remoção dos dados que evidenciava entupimento na subsérie 3.

- Normalização das subséries:

  Também se constatou que a etapa de normalização, apesar de empregar a estratégia de considerar a subsérie anterior para preservar a fidelidade aos dados originais, frequentemente modificou-se o padrão dos dados, resultando em agrupamentos incorretos. Houveram casos em que subséries anteriores com valores altos influencia-

ram a redução dos valores na subsérie corrente, levando o algoritmo a agrupá-las incorretamente como entupimento.

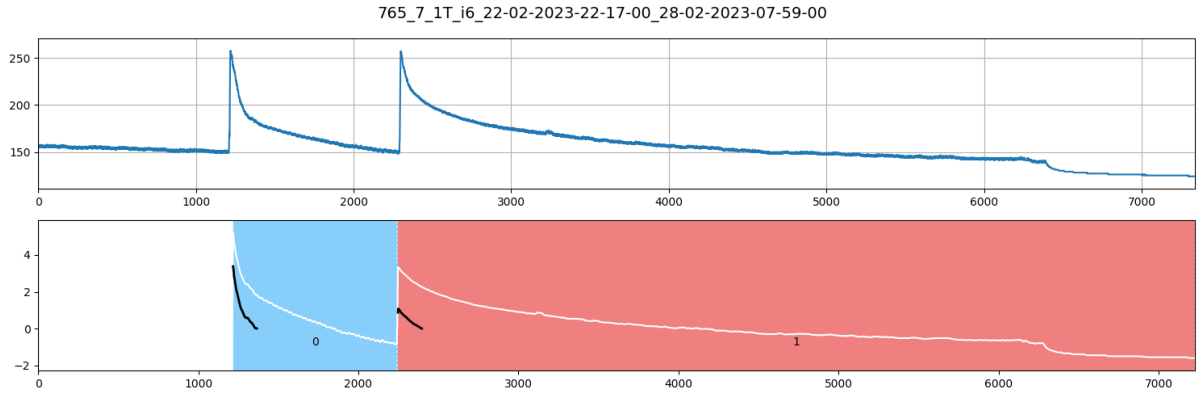


Figura 32 – Distorção dos dados pela normalização

Por fim, ao realizar a classificação manual, a autora notou que em diversas situações havia uma incerteza quanto à categorização adequada da subsérie. Nestes casos, a autora recorreu à análise do comportamento das outras subséries dentro do mesmo conjunto de dados para auxiliar na decisão. Este fato evidencia que a tarefa de classificação pode ser complexa e sujeita à ambiguidades. Isso sugere que os padrões que determinam se uma subsérie possui ou não entupimento podem não ser facilmente discerníveis e podem depender de fatores sutis e/ou contextuais além dos contidos na própria série.

5.3.4 Discussões

No capítulo de análise de resultados anterior, torna-se evidente que o algoritmo implementado apresenta algumas falhas e demanda um refinamento maior na etapa de pré-processamento dos dados. Nota-se que é preciso aplicar filtros mais amplos, mas sem comprometer a divisão das subséries. Além disso, a detecção das subséries requer um refinamento. Há também a necessidade de explorar alternativas para padronizar os dados de maneira mais eficaz, porém preservando melhor seus padrões originais.

Apesar das deficiências detectadas no método, é possível considerar satisfatória a acurácia de 80% alcançada pelo algoritmo. O propósito deste trabalho foi implementar um método de aprendizado de máquina que auxiliasse na identificação de comportamentos anômalos ou desvios do padrão esperado nas máquinas de digestão de resíduos orgânicos. Embora a classificação exata de cada processo não seja estritamente necessária, os resultados obtidos das classificações podem ser valiosos para o fabricante dessas máquinas, a fim de detectar dispositivos que não estejam funcionando adequadamente ou usuários que possam estar fazendo uso incorreto das mesmas.

6 CONCLUSÕES

Em conclusão, este trabalho explorou a aplicação do algoritmo k-means de aprendizado de máquina na detecção de padrões e comportamentos anômalos em séries temporais provenientes de máquinas de digestão de resíduos orgânicos. Mais especificamente, partes de séries temporais que apresentam um possível padrão de entupimento do peso na máquina. Embora os resultados obtidos tenham evidenciado algumas limitações no algoritmo implementado, a acurácia alcançada demonstrou ser promissora e indicativa de que a abordagem pode ser eficaz na identificação de desvios do comportamento esperado.

A análise crítica dos resultados permitiu a identificação de áreas que demandam aprimoramento, tais como a melhoria do pré-processamento dos dados para melhor tratar ruídos e preservar padrões originais, a otimização da detecção de subséries e a investigação de métodos alternativos de normalização. Mesmo assim, a capacidade do método em auxiliar na detecção de máquinas com funcionamento impróprio ou comportamento incomum representa um valor significativo para os fabricantes e usuários das máquinas.

Este estudo demonstra a complexidade inerente à análise de séries temporais em cenários reais, onde ruídos, variabilidades e padrões sutis podem influenciar os resultados. Ao mesmo tempo, ressalta o potencial das abordagens de aprendizado de máquina em contribuir para soluções automatizadas e eficazes na monitorização de sistemas industriais e detecção de anomalias.

REFERÊNCIAS

ANNAM, J. R.; MITTAPALLI, S. S.; BAPI, R. S. Time series clustering and analysis of ecg heart-beats using dynamic time warping. *In*: **2011 Annual IEEE India Conference**. [*S.l.: s.n.*], 2011. p. 1–3.

BUITINCK, L. *et al.* API design for machine learning software: experiences from the scikit-learn project. *In*: **ECML PKDD Workshop: Languages for Data Mining and Machine Learning**. [*S.l.: s.n.*], 2013. p. 108–122.

CHRIST, M. *et al.* Time series feature extraction on basis of scalable hypothesis tests (tsfresh – a python package). **Neurocomputing**, v. 307, p. 72–77, 2018. ISSN 0925-2312. Available at: <https://www.sciencedirect.com/science/article/pii/S0925231218304843>.

CHRIST, M.; KIENLE, F.; KEMPA-LIEHR, A. **Time Series Analysis in Industrial Applications**. 2016.

KUHN, M.; JOHNSON, K. **Applied Predictive Modeling**. 5th corrected printing. ed. [*S.l.: s.n.*], 2016. 31 p. ISBN 978-1-4614-6848-6.

MACQUEEN, J. Classification and analysis of multivariate observations. *In*: UNIVERSITY OF CALIFORNIA LOS ANGELES LA USA. **5th Berkeley Symp. Math. Statist. Probability**. [*S.l.: s.n.*], 1967. p. 281–297.

MALKI, A. A. *et al.* Hybrid genetic algorithm with k-means for clustering problems. **Open Journal of Optimization**, v. 5, p. 71–83, 2016. Available at: <http://dx.doi.org/10.4236/ojop.2016.52009>.

MCKINNEY Wes. Data Structures for Statistical Computing in Python. *In*: WALT Stéfan van der; MILLMAN Jarrod (ed.). **Proceedings of the 9th Python in Science Conference**. [*S.l.: s.n.*], 2010. p. 56 – 61.

RADOVANOVIć, M.; NANOPOULOS, A.; IVANOVIć, M. Hubs in space: Popular nearest neighbors in high-dimensional data. **Journal of Machine Learning Research**, v. 11, p. 2487–2531, 2010. Available at: <http://www.jmlr.org/papers/volume11/radovanovic10a/radovanovic10a.pdf>.

RESíDUOS orgânicos. *In*: . Embrapa, 2021. Available at: <https://www.embrapa.br/hortalica-nao-e-so-salada/secoes/residuos-organicos>. Access at: 21 fev. 2023.

SCACHETTI, H. A. **Usando aprendizado de máquina para prever extravasão em elevatórias de esgoto**. 2020. Master's thesis, 2020.

SUGIMURA, H.; MATSUMOTO, K. Classification system for time series data based on feature pattern extraction. *In*: **2011 IEEE International Conference on Systems, Man, and Cybernetics**. [*S.l.: s.n.*], 2011. p. 1340–1345.