

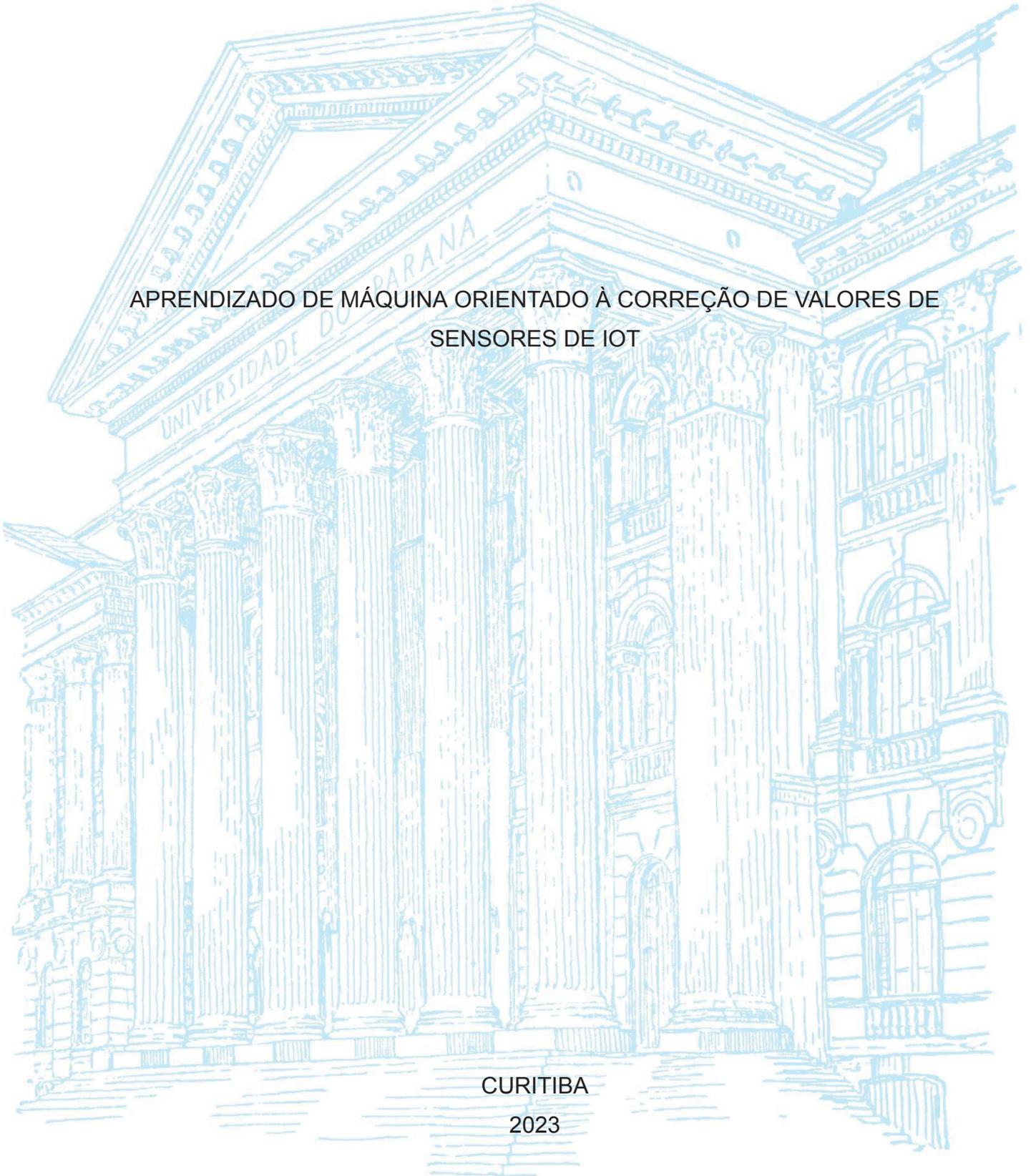
UNIVERSIDADE FEDERAL DO PARANÁ

BRUNO LUVIZOTTO CARLI

APRENDIZADO DE MÁQUINA ORIENTADO À CORREÇÃO DE VALORES DE
SENSORES DE IOT

CURITIBA

2023



BRUNO LUVIZOTTO CARLI

APRENDIZADO DE MÁQUINA ORIENTADO À CORREÇÃO DE VALORES DE SENSORES DE IoT

Trabalho de Conclusão de Curso apresentado ao curso de Especialização em Inteligência Artificial Aplicada, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Especialista em Inteligência Artificial Aplicada.

Orientador: Prof. Dr. Razer Anthom Nizer Rojas Montaña

CURITIBA

2023

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INTELIGÊNCIA ARTIFICIAL APLICADA da Universidade Federal do Paraná foram convocados para realizar a arguição da Monografia de Especialização de **BRUNO LUVIZOTTO CARLI** intitulada: **Aprendizado de Máquina orientado à correção de valores de sensores de IoT**, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa. A outorga do título de especialista está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 18 de Maio de 2023.



RAZER ANTHOM NIZER ROJAS MONTAÑO
Presidente da Banca Examinadora



JAIME WOJCIECHOWSKI
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Aprendizado de Máquina orientado à correção de valores de sensores de IoT

Bruno Luvizotto Carli
Universidade Federal do Paraná (UFPR)
Curitiba/PR, Brasil
brunolcarli@gmail.com

Razer Anthom Nizer Rojas Montaña
Universidade Federal do Paraná (UFPR)
Curitiba/PR, Brasil
razer@ufpr.br

Resumo—A internet das coisas é uma realidade contemporânea que vem sendo explorada por diversas empresas no setor da indústria, gerando dados a partir de sensores e transmitindo através da internet dando ascensão às cidades inteligentes. Sensores de dispositivos muitas vezes podem apresentar imprecisão na leitura dos valores captados do ambiente, uma forma plausível de corrigir sensores danificados ou desregulados, com menor taxa de custo e manutenção, pode ser viabilizada a partir de modelos de aprendizagem de máquina treinados sobre dados de sensores calibrados. O presente estudo de caso aborda e ilustra um cenário real da implantação de um modelo de cujo realiza a tarefa de correção de valores de corrente elétrica de dispositivos de *IoT* implantados em campo com 97% de precisão, tal metodologia demonstrou de forma positiva a viabilidade de implantação de uma solução moderna para aproximação de valores, pelo lado negativo a solução demandou demasiado tempo e esforço computacional para análise e preparação dos dados, assim como um longo tempo de treinamento dos modelos nas buscas de parâmetros, as quais podem resultar em baixas métricas de avaliação, portanto tal metodologia pode não se aplicar a todos os casos.

Palavras-chave—*iot*, aprendizado de máquina, cidades inteligentes

Abstract—The internet of things is a contemporary reality that has been explored by several companies in the industry sector, generating data from sensors and transmitting through the internet, based on this principle, taking part in the rise of smart cities. Device sensors can often present inaccuracy in reading the values captured from the environment, a plausible way to correct damaged or unregulated sensors with a lower cost and maintenance rate can be made possible from machine learning models trained on calibrated sensor data. The present case study approaches and illustrates a real scenario of the implementation of a model which performs the task of correcting the electric current values of *IoT* devices deployed in the field with a precision of 97%, such methodology positively demonstrated the feasibility of implementing a modern solution for approximation of values, on the negative side the solution demanded too much time and computational effort for data analysis and preparation, as well as a long time for training the models in parameter searches, which can result in low evaluation metrics, so this methodology may not apply to all cases.

Keywords—*iot*, machine learning, smart cities

I. DESENVOLVIMENTO

A. Internet das Coisas

A Internet das Coisas (*IoT*) é uma realidade tecnológica que vem ganhando espaço e abrangência no meio empresarial e tecnológico, contribuindo para inovação, geração de informação, solução e automação de problemas de forma distribuída e conectada [1].

Pela oferta, as empresas de base tecnológica estão sendo gradualmente mais competitivas, provendo ao mercado produtos e serviços cada vez mais sofisticados. Uma destas ofertas, que já se permite ser considerada uma revolução na inovação tecnológica, e de interconectividade, é a chamada Internet das Coisas, ou *Internet of Things (IoT)*. Esta tecnologia já estabelece conexões entre homens e máquinas, e máquinas com máquinas, proporcionando a formação de redes sociais e de mercado cada vez mais inteligentes [1].

Em resumo, a Internet das Coisas pode ser definida como redes que conectam qualquer coisa à Internet, por meio de protocolos de comunicação em equipamentos de identificação de dados, para transmitir informações coletadas pelo objeto [1].

B. Cidades Inteligentes

Há um entendimento de que cidades inteligentes são definidas pelo o acesso à computadores e implantação de internet em espaços urbanos, se caracterizam pela utilização da tecnologia e automatização de processos de maneira estratégica, para proporcionar melhor infraestrutura ao espaço urbano e otimizar soluções por meio de conectividade.

As pesquisas e estudos em torno da temática de cidades inteligentes (*smart cities*) é relativamente recente, tendo início por volta de 1990 tendo como bases de fundação o avanço tecnológico e obtendo avanços desde então, ainda que estas discussões acerca da temática nos últimos anos sejam notáveis, apresentar uma definição clara e objetiva para cidades inteligentes pode permanecer muitas vezes imprecisa devido à amplitude de todas as abstrações e contextos que ofuscam uma definição objetiva, dado que o conceito pode abordar diferentes vertentes ou escopos de aplicação o que implica na formalização de que esta ainda é uma ideia um tanto abstrata, para este escopo será definida como a automação da telegestão de ambientes urbanos informatizados e viabilizados pela internet das coisas. [2]

C. Problemática e justificativa

Este estudo descreve a análise de dois modelos de dispositivos de IoT que mensuram a luminosidade e consumo de energia elétrica em ambientes públicos onde uma versão V1 apresenta medições imprecisas do sensor de corrente elétrica quando o relé está desatracado variando entre 30mA e 80mA e a outra versão V2 do dispositivo apresenta uma precisão calibrada do sensor variando entre 7mA e 10mA quando o relé está em estado de desatracado.

A problemática em questão define-se pela possibilidade da minimização dos valores do sensor de corrente dos dispositivos V1 cuja precisão excede os níveis de eletricidade de 30mA a partir das transmissões dos dispositivos V2 como referência.

Objetiva-se aproximar os valores de corrente dos equipamentos V1 aos mesmos valores contidos na faixa conhecida de corrente transmitida pelos dispositivos V2 a partir dos dados das transmissões emitidas pelos dispositivos V1 através de técnicas de aprendizado de máquina, uma vez que ambos dispositivos transmitem as mesmas informações permitindo traçar um padrão nos dados recebidos pelo sistema de telegestão.

II. DESCRIÇÃO DOS DADOS

Foram coletados 195.744 registros de transmissões de dispositivos V1 e 134.618 registros do modelo V2 onde ambas amostras apresentam as mesmas colunas de dados estando listados nas tabelas I e II.

A coluna que apresenta o MAC descreve o endereço único do dispositivo que transmitiu a informação, o relé informa se a lâmpada estava acesa ou apagada, a corrente informa o valor de energia elétrica mensurado pelo sensor em miliamperes (mA), também foram observados os valores de hora e minutos que determinam em qual período do dia a transmissão foi emitida pelo dispositivo, o LDR é referente ao sensor de luminosidade (sensor fotovoltaico) medido em Lux.

Tabela I
EQUIPAMENTOS COM SENSOR IRREGULAR (V.1)

| MAC | HORA | RELÉ | LDR(lx) | CORRENTE(mA) |
|-------------------|-------|-------|---------|--------------|
| 804B50FFFE60AFBC | 5:15 | False | 3.549 | 22 |
| 847127FFFE2BB460 | 8:48 | False | 19.279 | 20 |
| 003C84FFFE188BBC | 12:54 | False | 1.614 | 18 |
| 5C0272FFFEFCFB7FB | 4:33 | False | 1.594 | 22 |
| 804B50FFFE2FA6EF | 5:38 | False | 8.122 | 18 |
| 847127FFFE99208F | 20:14 | True | 205 | 1636 |
| 804B50FFFE611160 | 13:16 | False | 1.312 | 18 |
| 804B50FFFE6104B2 | 9:24 | False | 1.932 | 22 |
| 5C0272FFFECC22D4 | 8:11 | False | 11.712 | 18 |

Tabela II
EQUIPAMENTOS COM SENSOR CALIBRADO (V.2)

| MAC | HORA | RELÉ | LDR (lx) | CORRENTE (mA) |
|-------------------|-------|-------|----------|---------------|
| 5C0272FFFECCCF53 | 7:50 | False | 26.653 | 10 |
| 5C0272FFFEED3CE60 | 12:20 | False | 3.944 | 9 |
| 5C0272FFFECCD356 | 7:58 | False | 19.841 | 8 |
| 5C0272FFFECCB243 | 6:32 | False | 5.677 | 8 |
| 5C0272FFFEED55243 | 5:37 | False | 28.641 | 9 |
| 5C0272FFFEDEB138 | 12:35 | False | 5.774 | 8 |
| 5C0272FFFECCB0ED | 11:13 | False | 7.842 | 8 |
| 588E81FFFE0068CB | 8:35 | False | 8.217 | 8 |
| 5C0272FFFECCCF35 | 8:19 | False | 8.402 | 10 |

Pode-se visualizar a diferença de valores da corrente para cada hora do dia. As Figuras 1 e 2 ilustram a comparação dos valores de corrente agrupados por hora do dia calculando seu quartil (Q1) e a média do valor de corrente por hora agrupados para um período de 5 dias de transmissões.

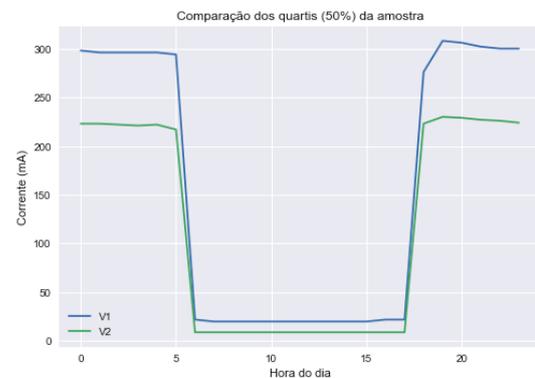


Fig. 1. Comparativo do quartil (Q1) da corrente por hora do dia.

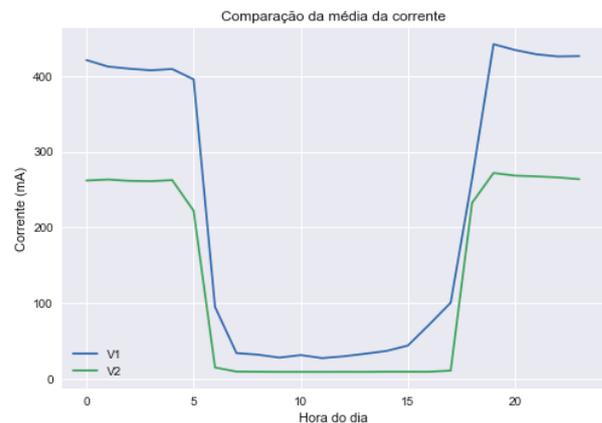


Fig. 2. Comparativo da média dos valores de corrente por hora do dia.

Ao separar apenas os registros contendo o relé desatracado pode-se visualizar a faixa de corrente elétrica no gráfico de linha da Figura 3.

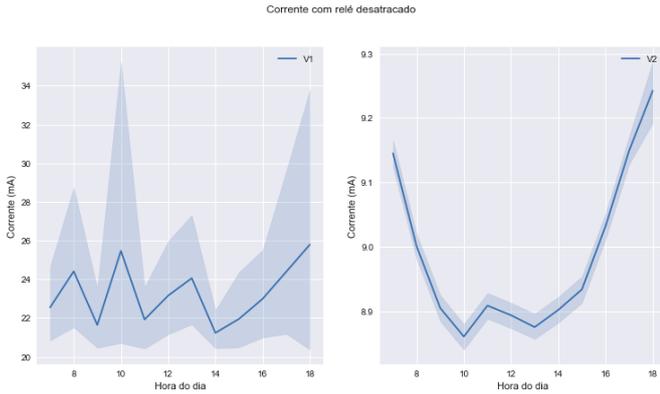


Fig. 3. Valores de corrente para amostras que contém o relé desatracado.

III. MÉTODOS

Para metodologia que define o procedimento a ser executado na análise e implementação das atividades, seguiu-se o modelo de etapas do fluxo de *machine learning* ilustrado na Figura 4. Tal fluxo é uma expansão da metodologia CRISP-DM (*Cross-Industry Standard Process for Data Mining*) [3].

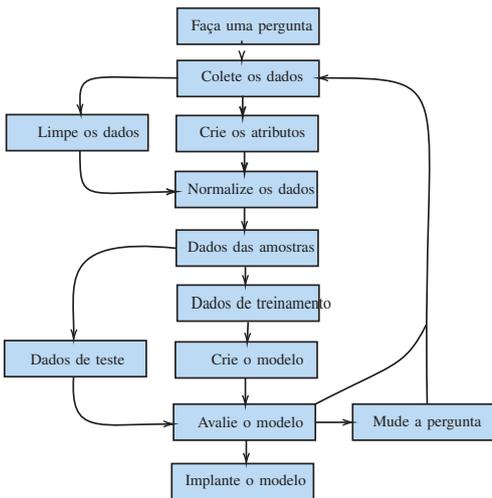


Fig. 4. Expansão da metodologia CRISP-DM [3].

Os dados foram inicialmente coletados de uma base NoSQL e separados em documentos para cada versão de dispositivos, pré-processados e formatados para arquivos no formato *csv* para evitar a busca em disco e o re-processamento dos dados em caso de perda, em seguida é realizada a visualização e análise exploratória dos dados.

A seleção de atributos trata-se de uma etapa iterativa na qual são testadas as variáveis independentes que apresentam diferentes possíveis combinações, dividindo em conjuntos de treino e teste, realizando as buscas de parâmetros e testando os modelos até que um modelo atinja a precisão aproximada de corrente dos dispositivos da versão V2 no tempo disponibilizado para a pesquisa ou então, a hipótese de que a

aproximação da precisão da corrente do dispositivo por meio de técnicas de aprendizado de máquina é refutada.

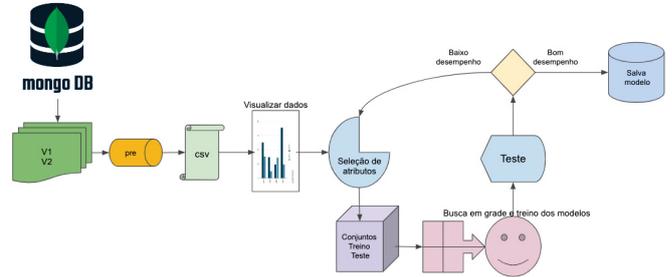


Fig. 5. Processo metodológico.

O conjunto de treino foi composto pelos dados coletados dos equipamentos V2 alocando 67% das amostras totais somando 89.714 amostras e 33% do conjunto foi dedicado para teste dos modelos utilizando 44.188 das amostras. Todo fluxo pode ser visualizado na Figura 5.

Dado que a variável dependente aqui (corrente elétrica) é um dado numérico pertencente a uma faixa padrão (entre 7 e 10) optou-se pela utilização de modelos de classificação, desta forma a partir dos valores de entrada será possível classificar a corrente esperada como valor de saída do modelo.

Foram treinados quatro modelos de classificação, usando as técnicas KNN, Floresta Aleatória, SGD e Perceptron Multicamada. Os modelos utilizados para treinamento foram importados das bibliotecas do framework scikit-learn [7] utilizando uma grade de busca para encontrar os melhores hiperparâmetros utilizando cinco validações cruzadas nos conjuntos de treino e teste, onde os parâmetros testados para cada modelo seguem listados nas Tabelas III, IV, V e VI:

Tabela III
GRADE DE BUSCA DO KNN

| Parâmetro | Valores |
|--------------------|---|
| Número de vizinhos | 3, 10, 15, 25 |
| Pesos | Uniforme, Distância |
| Algoritmo | Ball tree, kd tree, brute |
| Tamanho das folhas | 50, 100, 150 |
| p | 1, 2, 3 |
| Métrica | cityblock, euclidean, l1, l2, manhattan |

IV. MÉTRICAS DE AVALIAÇÃO

A. Precisão

O desempenho do modelos será definido pelos resultados obtidos através das métricas de precisão, revocação e pontuação *f1*.

A precisão do classificador leva em consideração a quantidade de acertos positivos pelo número de falsos positivos como definido na equação (1):

Tabela IV
GRADE DE BUSCA DO SGD

| Parâmetro | Valores |
|----------------------|---|
| Perda | <i>hinge, log, modified huber, squared hinge, perceptron, huber, epsilon insensitive, squared epsilon insensitive</i> |
| Penalidade | <i>l1, l2, elasticnet, None</i> |
| <i>Alpha</i> | 0.001, 0.01, 0.5 |
| Taxa l1 | 0.5, 0.01, 0.09, 0.001 |
| <i>Fit intercept</i> | <i>True, False</i> |
| <i>Epsilon</i> | 0.5, 0.01, 0.001 |
| Parada antecipada | <i>True</i> |
| Peso das classes | <i>None, balanceado</i> |

Tabela V
GRADE DE BUSCA DA FLORESTA ALEATÓRIA

| Parâmetro | Valores |
|---------------------------------|----------------------|
| <i>Bootstrap</i> | <i>True</i> |
| <i>CCP Alpha</i> | 0.0, 0.2 |
| Peso das classes | <i>None</i> |
| Critério | Gini |
| Profundidade máxima | <i>None, 1, 2, 5</i> |
| Características máximas | auto |
| Máximo de nós folha | <i>None, 3</i> |
| Máximo de amostras | <i>None</i> |
| Decremento mínimo de impureza | 0.0, 0.2 |
| Separação mínima de impureza | <i>None</i> |
| Amostras mínimas por folha | 1, 2 |
| Separação mínima de amostras | 2, 0.1 |
| Peso mínimo por fração de folha | 0.0, 0.3 |
| Número de estimadores | 500, 1500, 5000 |
| Número de jobs | <i>None, -1</i> |
| Pontuação OOB | <i>False</i> |

$$precisão = \frac{TP}{TP + FP} \quad (1)$$

Onde TP (*True Positives*) é o número de verdadeiros positivos e FP (*False Positives*) é o número de falsos positivos [4].

B. Revocação

A revocação, também conhecida como *sensibilidade* ou *taxa de verdadeiros positivos* (TPR, do inglês) apresenta a taxa

Tabela VI
GRADE DE BUSCA DA PERCEPTRON MULTICAMADA

| Parâmetro | Valores |
|-----------------------------|--|
| Tamanho das camadas ocultas | (5, 5, 10), (50, 50, 50), (50, 100, 50), (100) |
| Ativação | tanh, ReLu, logística |
| Resolvedor | SGD, Adam |
| <i>Alpha</i> | 0.0001, 0.05 |
| Taxa de aprendizagem | Constante, Adaptativa |
| Iterações máximas | 2000 |

de instâncias positivas que são corretamente detectadas pelo classificador, sendo definida pela equação (2) [4]:

$$revocação = \frac{TP}{TP + FN} \quad (2)$$

Onde FN (*False Negatives*) corresponde ao número de falsos negativos [4].

C. Pontuação f1

É conveniente combinar a precisão e revocação m uma única métrica chamada pontuação f1. Também conhecida como *média harmônica* da precisão e revocação, dando maior peso aos valores mais baixos. Como resultado, o classificador só obterá uma pontuação f1 alta se a revocação e a precisão forem altas [4].

$$F_1 = \frac{precisão \times revocação}{precisão + revocação} = \frac{TP}{TP + \frac{FN+TP}{2}} \quad (3)$$

V. RESULTADOS E DISCUSSÕES

Devido à quantidade massiva de dados contida no conjunto de treino os modelos levaram um tempo consideravelmente alto para a busca de parâmetros, algumas buscas da SGD chegaram a levar cerca de 12 a 16 horas de treino, revelando que esta técnica não foi uma das melhores escolhas para este conjunto de dados A melhor combinação de parâmetros encontrado pela busca para cada modelo estão especificados nas Tabelas VII, VIII, IX e X:

Tabela VII
HIPERPARÂMETROS DO KNN

| Parâmetro | Valores |
|--------------------|------------------|
| Número de vizinhos | 25 |
| Pesos | Uniforme |
| Algoritmo | <i>Ball tree</i> |
| Tamanho das folhas | 100 |
| p | 1 |
| Métrica | <i>euclidean</i> |

Tabela VIII
HIPERPARÂMETROS DO SGD

| Parâmetro | Valores |
|----------------------|-------------------|
| Perda | <i>huber</i> |
| Penalidade | <i>elasticnet</i> |
| <i>Alpha</i> | 0.5 |
| Taxa l1 | 0.001 |
| <i>Fit intercept</i> | <i>False</i> |
| <i>Epsilon</i> | 0.01 |
| Parada antecipada | <i>True</i> |
| Peso das classes | <i>None</i> |

Os resultados sob o conjunto de teste para os parâmetros encontrados, apesar de apresentarem a classificação dentro da faixa solicitada, não alcançaram métricas convencionalmente

Tabela IX
HIPERPARÂMETROS DA FLORESTA ALEATÓRIA

| Parâmetro | Valores |
|---------------------------------|--------------|
| <i>Bootstrap</i> | <i>True</i> |
| <i>CCP Alpha</i> | <i>0.0</i> |
| Peso das classes | <i>None</i> |
| Critério | Gini |
| Profundidade máxima | 5 |
| Características máximas | auto |
| Máximo de nós folha | <i>None</i> |
| Máximo de amostras | <i>None</i> |
| Decremento mínimo de impureza | 0.0 |
| Separação mínima de impureza | <i>None</i> |
| Amostras mínimas por folha | 1 |
| Separação mínima de amostras | 2 |
| Peso mínimo por fração de folha | 0.0 |
| Número de estimadores | 1500 |
| Número de jobs | <i>None</i> |
| Pontuação OOB | <i>False</i> |

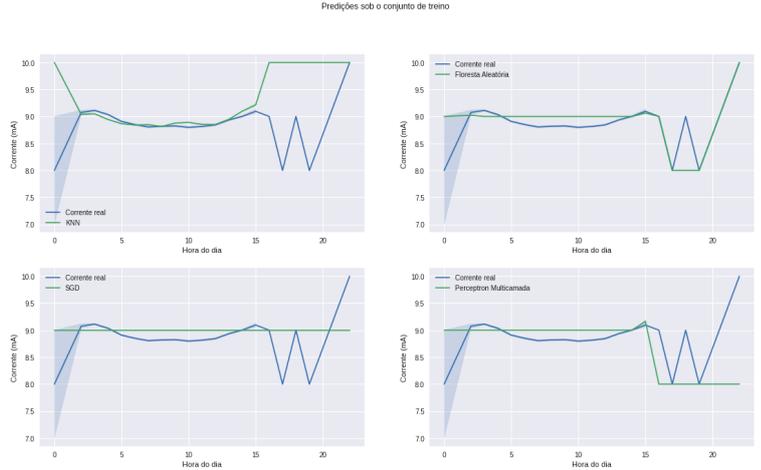


Fig. 6. Resultado dos modelos no conjunto de treino.

Tabela X
HIPERPARÂMETROS DA PERCEPTRON MULTICAMADA

| Parâmetro | Valores |
|-----------------------------|-----------|
| Tamanho das camadas ocultas | (100) |
| Ativação | tanh |
| Resolverdor | Adam |
| <i>Alpha</i> | 0.0001 |
| Taxa de aprendizagem | Constante |
| Iterações máximas | 2000 |

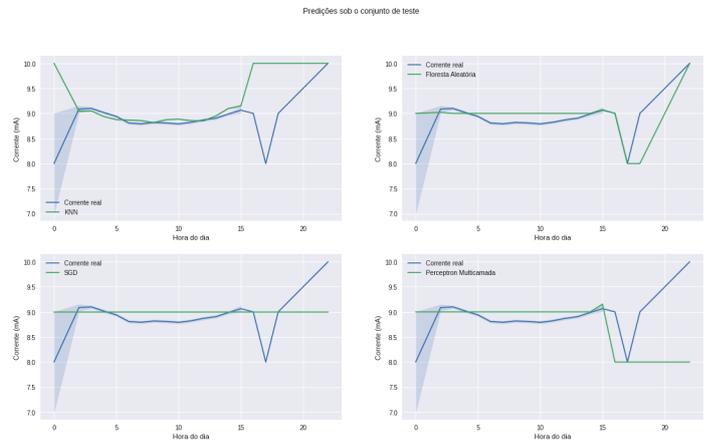


Fig. 7. Resultado dos modelos no conjunto de teste.

aceitáveis, resultando em pontuações abaixo de 50% de acertos. Os resultados obtidos estão elencados na Tabela XI:

As previsões dos modelos para as amostras dos conjuntos de treino e teste foram ilustrados nos gráficos das Figuras 6 e 7.

Tabela XI
MÉTRICAS DE CLASSIFICAÇÃO NO CONJUNTO DE TESTE

| Modelo | Precisão | Revocação | f1 |
|------------------------|----------|-----------|------|
| KNN | 0.36 | 0.44 | 0.35 |
| Floresta Aleatória | 0.48 | 0.46 | 0.30 |
| SGD | 0.21 | 0.46 | 0.29 |
| Perceptron Multicamada | 0.37 | 0.46 | 0.30 |

Em experimentos prévios à busca de parâmetros em grade, os modelos foram instanciados sob configuração padrão para que então fosse inicializada a busca de hiperparâmetros. A caráter de teste experimental treinou-se os modelos sob configuração padrão no conjunto de treino e a Floresta Aleatória apresentou resultados muito melhores em relação à primeira retornada pela grade de busca tendo seus hiperparâmetros apresentados na Tabela XII e suas métricas de avaliação apresentadas na Tabela XIII:

Tabela XII
HIPERPARÂMETROS DA FLORESTA ALEATÓRIA PADRÃO

| Parâmetro | Valores |
|---------------------------------|--------------|
| <i>Bootstrap</i> | <i>True</i> |
| <i>CCP Alpha</i> | 0.0 |
| Peso das classes | <i>None</i> |
| Critério | Gini |
| Profundidade máxima | <i>None</i> |
| Características máximas | auto |
| Máximo de nós folha | <i>None</i> |
| Máximo de amostras | <i>None</i> |
| Decremento mínimo de impureza | 0.0 |
| Separação mínima de impureza | <i>None</i> |
| Amostras mínimas por folha | 1 |
| Separação mínima de amostras | 2 |
| Peso mínimo por fração de folha | 0.0 |
| Número de estimadores | 100 |
| Número de jobs | <i>None</i> |
| Pontuação OOB | <i>False</i> |

Tabela XIII
MÉTRICAS DE CLASSIFICAÇÃO NO CONJUNTO DE TESTE FLORESTA ALEATÓRIA PADRÃO

| Precisão | Revocação | f1 |
|----------|-----------|------|
| 0.97 | 0.97 | 0.97 |

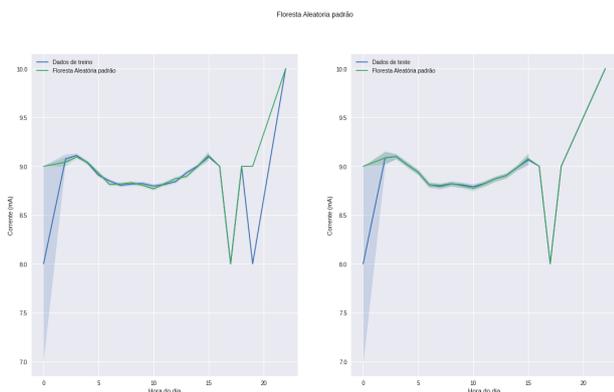


Fig. 8. Floresta Aleatória padrão nos conjuntos de treino e teste.

Para validação foi separado do conjunto de dados V1 seus horários e valores de luminosidade, os valores foram fornecidos aos modelos para que se pudesse visualizar e comparar os valores de corrente iniciais deste conjunto e os valores de corrente retornados pelos modelos treinados.

Corrente com relé desativado, comparação antes e depois

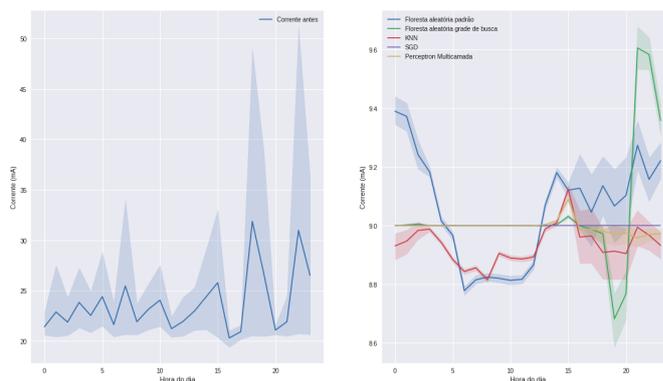


Fig. 9. Resultado dos modelos no conjunto de treino.

Para concluir o trabalho, pode-se afirmar que é possível fazer correções em valores de sensores descalibrados transmitidos por dispositivos de IoT implantados em campo a partir de amostras de dados coletadas de sensores de mesmo tipo transmitidos por equipamentos regulados sem a necessidade de realizar manutenções custosas em campo, o modelo foi implantado em um microserviço que avalia os valores de energia dos sensores para transformar o valor antes de disponibilizar os dados em seções de relatórios e gráficos, uma vez que os valores preditos são aceitáveis e reproduzíveis a solução foi aceita pela diretoria e gerência de projetos, para este caso a solução foi implantada a nível de *software* na camada de sistema de telegestão porém tais métodos podem vir a ser incluídos futuramente em nível de *hardware* e *firmware* embarcado, possibilitando a autocorreção dos sensores antecipadamente pelo próprio dispositivo.

Trabalhos futuros correlacionados à este estudo podem incluir a predição dos próximos valores de consumo de energia em caráter de análise de séries temporais e identificação de sensores defeituosos.

REFERÊNCIAS

- [1] J. E. Lara, L. J. Reis, T. A. Tissot-Lara e A. O. Silva, "Admirável mundo novo na perspectiva da tríade: Internet das Coisas, pessoas e mercados", *Perspectivas em Ciência da Informação*, vol. 26, n.º 2, pp. 124–150, junho de 2021. Consult. 2022-11-12. [Em linha]. Disponível: <https://doi.org/10.1590/1981-5344/3825>
- [2] B. D. N. JOÃO, C. L. D. SOUZA e F. A. SERRALVO, "Revisão sistemática de cidades inteligentes e internet das coisas como tópico de pesquisa", *Cadernos EBAPE.BR*, vol. 17, n.º 4, pp. 1115–1130, outubro de 2019. Consult. 2022-11-12. [Em linha]. Disponível: <https://doi.org/10.1590/1679-395174442>
- [3] M. Harrison, *Machine Learning Pocket Reference: A Quick Guide to Structured Machine Learning Techniques*. O'Reilly Media, Incorporated, 2019.
- [4] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2019.
- [5] QUININO, Roberto C.; REIS, Edna A.; BESSEGATO, Lupércio F. O Coeficiente de Determinação R² como Instrumento Didático para Avaliar a Utilidade de um Modelo de Regressão Linear Múltipla. 28 out. 2011. Disponível em: http://www.est.ufmg.br/portal/arquivos/rts/PD_28102011_Final.pdf. Acesso em: 17 jan. 2023.

- [6] BRUCE, Peter; BRUCE, Andrew. Estatística Prática para Cientistas de Dados: 50 Conceitos Essenciais. 1. ed. rev. Rio de Janeiro - RJ: Alta Books, 2019. ISBN 978-1-491-95296-2.
- [7] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011. Disponível em: <https://scikit-learn.org/0.18/index.html>.