

**Universidade Federal do Espírito Santo**  
**Centro Tecnológico**  
**Programa de Pós Graduação em Engenharia Elétrica**



**Bruna Rodrigues Vidigal**

# **Modelos para Previsão de Potência Gerada e Identificação de Falha em Painéis Fotovoltaicos**

**Vitória - ES, 2022**



Bruna Rodrigues Vidigal

## **Modelos para Previsão de Potência Gerada e Identificação de Falha em Painéis Fotovoltaicos**

Dissertação apresentada ao Programa de Pós Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica.

Universidade Federal do Espírito Santo - UFES  
Centro Tecnológico  
Programa de Pós Graduação em Engenharia Elétrica

Orientador: Marcia Helena Moreira Paiva  
Coorientador: Helder Roberto de Oliveira Rocha

Vitória - ES

2022

Ficha catalográfica disponibilizada pelo Sistema Integrado de Bibliotecas - SIBI/UFES e elaborada pelo autor

---

V653 Vidigal, Bruna Rodrigues, 1995-  
m Modelos para Previsão de Potência Gerada e Identificação de Falha em Painéis Fotovoltaicos / Bruna Rodrigues Vidigal. - 2023.  
77 f. : il.

Orientadora: Marcia Helena Moreira Paiva.  
Coorientador: Helder Roberto de Oliveira Rocha.  
Dissertação (Mestrado em Engenharia Elétrica) -  
Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Sistemas Fotovoltaicos. 2. Previsão. 3. Grafos. 4. Redes Neurais. 5. Classificação. I. Paiva, Marcia Helena Moreira. II. Rocha, Helder Roberto de Oliveira. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 621.3

---

Bruna Rodrigues Vidigal

## **Modelos para Previsão de Potência Gerada e Identificação de Falha em Painéis Fotovoltaicos**

Dissertação apresentada ao Programa de Pós Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica.

Trabalho aprovado. Vitória - ES, 15 de dezembro de 2022:

*M. Paiva*

---

**Marcia Helena Moreira Paiva**  
Orientador

*Agustin*

---

**Dr. Augusto César Rueda Medina**  
Universidade Federal do Espírito Santo  
Examinador Interno

*Jugurta Rosa Montalvão Filho*

---

**Dr. Jugurta Rosa Montalvão Filho**  
Universidade Federal de Sergipe  
Examinador Externo

Vitória - ES  
2022

*Aos meus pais Sandra Maria Rodrigues Vidigal e Vinicius Vidigal,  
por todo amor, apoio e incentivo.*

*Aos meus irmãos Victor Rodrigues Vidigal e Vinicius Vidigal Junior,  
pelos exemplos e contra-exemplos de vida.*

# AGRADECIMENTOS

Agradeço a Deus por me dar saúde e sabedoria para conseguir desenvolver o presente trabalho. Sou grata pelos livramentos e pelas pessoas abençoadas colocadas em meu caminho até o presente momento.

Aos meus pais sempre compreensivos, por abrir mão da minha presença em momentos importantes de reunião familiar e por me apoiar me dando o sustento, o suporte psicológico e emocional. À minha mãe Sandra Maria Rodrigues Vidigal, pelas noites em claro me acompanhando e dando apoio, por cada palavra de incentivo. Ao meu pai Vinicius Vidigal, pelos momentos de distração proporcionados, que me ajudaram a desanuviar e tornaram a vida mais agradável durante minha caminhada.

Aos meus irmãos, por me apoiarem em meus estudos e por incentivarem meu sucesso. Vinicius Vidigal Jr. e Victor Rodrigues Vidigal, vocês foram incríveis, sendo sempre presente e me dando apoio nos meus piores momentos.

Agradeço a minha orientadora, a Professora Marcia Paiva, pela oportunidade oferecida em trabalhar em um projeto com uma proposta singular. Sou grata pelo apoio e conselhos sábios que foram extremamente importantes nos momentos de tomada de decisão, e também pelos conselhos de vida compartilhados durante essa trajetória. Ao meu professor coorientador, Professor Helder Rocha, muito obrigado pelos ensinamentos e amparo.

Aos meus amigos, agradeço por estarem ao meu lado nos momentos de dificuldade e por se alegrarem nos momentos de vitória. Às amigas construídas ao longo da vida, Nathalia Andrade e Nathiely Soares, vocês são uma parte muito importante dessa conquista, agradeço pelos momentos de lazer indispensáveis, por ouvirem os desabafos aleatórios, pela compreensão pelos furos por toda ajuda que direta ou indiretamente vocês me forneceram. Aos amigos que trabalharam comigo e mesmo após seguirmos caminhos diferentes continuaram seguindo com a amizade, Beatriz Freire, Bruno Evaldt, Eliza Chagas e Leevan Dias, vocês foram fundamentais para o meu crescimento profissional, intelectual e pessoal, admiro muito vocês. À amiga Marcella Freire, que sempre insistiu que eu arranjassem algum tempo na correria do dia a dia para me exercitar, fazendo com que eu tivesse mais disposição nos meus afazeres e uma melhor qualidade de vida. Aos amigos adquiridos no mestrado, Roseli Wenzel e Yruí Neris, muito obrigado por toda ajuda, conselho, momentos de alegria, compartilhamento de conhecimento e vivências extraordinárias, vocês fizeram essa caminhada muito mais agradável e bem menos estressante. Às amigas conquistadas na graduação, Clarice Arruda e Nathalia Veríssimo, vocês têm um espaço especial no meu coração, vocês foram verdadeiros presentes que a UFES me concedeu.

Agradeço aos professores que, com paciência, ensinaram e tiraram as infinitas dúvidas que surgiam nas disciplinas ministradas durante o mestrado. Aos bolsistas e professores do CPID, por todo suporte e orientação.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ), à Fundação de Amparo à Pesquisa e Inovação do Espírito Santo (FAPES) e à Secretaria da Ciência, Tecnologia, Inovação, Educação Profissional e Desenvolvimento Econômico do Espírito Santo (SECTIDES) agradeço pelo apoio financeiro para realização deste trabalho de pesquisa.



*“Experiência não é o que acontece com um homem;  
é o que um homem faz com o que lhe acontece.”*  
*(Aldous Huxley)*



# RESUMO

A energia solar é uma alternativa promissora para atender às crescentes demandas energéticas do mundo. Um sistema fotovoltaico que possui tecnologia de armazenamento de dados de medição de carga elétrica (como potência, corrente e tensão), permite a criação de metodologias para monitorar e prever seu funcionamento, de modo a reduzir a perda de energia e o custo de manutenção. Neste trabalho, uma técnica utilizando um modelo híbrido de rede neural convolucional de grafos e redes neurais recorrentes é proposta com o objetivo de prever a potência gerada de um sistema fotovoltaico. Além disso, um modelo de classificação utilizando redes neurais multicamadas foi avaliado para detecção de falha do sistema com base na avaliação do sistema de monitoramento da usina. O ajuste do modelo híbrido considerando a estrutura temporal e topológica dos dados foi realizado e comparado com modelos de estudos anteriores. O modelo desenvolvido apresentou um resultado 96% melhor que outros modelos usados para comparação, mostrando que considerar a estrutura temporal e topológica do sistema é muito relevante na análise de desempenho. O desempenho do modelo classificador foi análogo ao desempenho do modelo híbrido, porém a ordem cronológica e a informação topológica dos dados foi desconsiderada na construção do modelo classificador.

**Palavras-chaves:** Sistemas Fotovoltaicos. Previsões. Grafos. Convolução. Rede Neural. Classificação.



# ABSTRACT

Solar energy is a promising alternative to meet the world's growing energy demands. A photovoltaic system that has data storage technology allows the creation of a methodology for monitoring its operation, in order to reduce energy losses and maintenance costs. In this study, a technique is proposed using a hybrid model of graph convolutional neural network and recurrent neural networks with the objective to forecast the generated power of system. The model adjustment was performed and compared with models from previous studies. The developed one presented a result about 96% better than the other models, showing that considering the temporal and topological structure of the system is very relevant in the performance analysis.

**Keywords:** Photovoltaic System. Forecasting. Graph. Convolutional. Neural Network. Classification.



# LISTA DE ILUSTRAÇÕES

Figura 1 – Efeito Fotovoltaico . . . . .	7
Figura 2 – Lingotes de Silício . . . . .	8
Figura 3 – Componentes do Sistema Fotovoltaico. . . . .	9
Figura 4 – Subgrafo . . . . .	11
Figura 5 – Grafos . . . . .	11
Figura 6 – Grafo Desconexo: Representação da topologia lógica de um sistema fotovoltaico com dois inversores e dezoito módulos. . . . .	12
Figura 7 – Problema de Euler . . . . .	13
Figura 8 – Esquema Operacional do Aprendizado Profundo de Máquina . . . . .	18
Figura 9 – Esquema Operacional Básico da Rede Neural Perceptron . . . . .	22
Figura 10 – Esquema Operacional Básico da Rede Neural Convolutiva . . . . .	23
Figura 11 – Esquema Operacional Básico da Camada LSTM. . . . .	25
Figura 12 – Unidade de Entrada da Camada LSTM. . . . .	26
Figura 13 – Unidade de “Esquecimento” da Camada LSTM. . . . .	27
Figura 14 – Unidade de Saída da Camada LSTM. . . . .	28
Figura 15 – Usina experimental de geração de energia alternativa - CPID. . . . .	29
Figura 16 – Relação lógica do sistema fotovoltaico do CPID. . . . .	32
Figura 17 – Gráfico Boxplot da Geração de Energia dos Módulos do Sistema . . . . .	34
Figura 18 – Transformação Box-Cox . . . . .	35
Figura 19 – Gráficos diagnósticos dos resíduos do modelo . . . . .	36
Figura 20 – Estrutura do Modelo TGCN . . . . .	37
Figura 21 – Representação da estrutura de grafo da topologia lógica da usina experimental de geração de energia alternativa - CPID. Cada $Z_t$ indica a imagem das medições de potência no instante $t$ , que é armazenado em uma matriz de atributos de nós. A série temporal apresenta a série $z_v$ de um módulo do sistema fotovoltaico. . . . .	41
Figura 22 – Treinamento do modelo proposto na Tabela 4. . . . .	43
Figura 23 – Comparação dos valores reais <i>versus</i> previsões um passo à frente da série temporal. . . . .	44
Figura 24 – Treinamento do modelo proposto na Tabela 6 . . . . .	46
Figura 25 – Matriz de Confusão . . . . .	46





# LISTA DE TABELAS

Tabela 1 – Dispositivos Eletrônicos: <i>Software</i> . . . . .	30
Tabela 2 – Dispositivos Eletrônicos: <i>Hardware</i> . . . . .	31
Tabela 3 – Blocos espaço-temporal . . . . .	37
Tabela 4 – Arquitetura T-GCN . . . . .	42
Tabela 5 – Desempenho dos modelos . . . . .	44
Tabela 6 – Arquitetura Modelo de Classificação . . . . .	45



# LISTA DE ABREVIATURAS E SIGLAS

AR	<i>Autoregressive</i>
ARIMA	<i>Autoregressive Integrated Moving Average</i>
CA	Corrente Alternada
CC	Corrente Contínua
CNN	<i>Convolutional Neural Network</i>
Colab	Google <i>Collaboratory</i>
CPID	Centro de Pesquisa, Inovação e Desenvolvimento
GCNN	<i>Graph Convolutional Neural Network</i>
<i>LeakyReLU</i>	<i>Leaky Rectified Linear Hidden Units</i>
MA	<i>Moving Average</i>
MLPE	Eletrônica de Potência a nível de Módulo
MPPT	Rastreador de Máximo Ponto de Potência
<i>ReLU</i>	Restricted Linear Unit
SARIMA	<i>Seasonal-ARIMA</i>
<i>Tanh</i>	Tangente Hiperbólica
TGF	<i>Trivial Graph Format</i>



# SUMÁRIO

	<b>Lista de ilustrações</b>	<b>XIII</b>
	<b>Lista de tabelas</b>	<b>XV</b>
	<b>Lista de abreviaturas e siglas</b>	<b>XIX</b>
<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
1.1	Justificativa	2
1.2	Estado da Arte	3
1.3	Objetivos	5
1.4	Método de Desenvolvimento do Trabalho	5
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>7</b>
2.1	Sistemas Fotovoltaicos	7
2.2	Teoria de Grafos	10
2.3	Séries Temporais	13
2.4	Modelos Estatísticos	14
2.4.1	Processos sazonais autoregressivos integrados de médias móveis (SARIMA)	15
2.5	<b>Aprendizado Profundo de Máquina</b>	<b>17</b>
2.5.1	Representação dos Dados para uma Rede Neural	19
2.5.2	Tratamento de Dados para uma Rede Neural	19
2.5.3	Funções de Ativação	20
2.5.4	Otimização Baseada em Gradiente	21
2.5.5	Algoritmo de Retropropagação	21
2.5.6	Perceptron Multicamadas	22
2.5.7	Convolução de Grafos	22
2.5.8	Aprendizado Profundo para Previsão de Séries Temporais	24
<b>3</b>	<b>METODOLOGIA EXPERIMENTAL</b>	<b>29</b>
3.1	Base de Dados	29
3.2	Informações Computacionais	30
3.3	Modelagem dos dados	31
3.4	Implementação das Soluções	33
3.4.1	Análise Exploratória dos Dados	33
3.4.2	Modelo Estatístico - SARIMA	35
3.4.3	Rede Neural Espaço Temporal de Grafos - STGCN	36
3.5	<b>Estrutura do Modelo</b>	<b>37</b>

3.6	Metodologia de Avaliação do Desempenho da Rede Neural . . . . .	38
3.7	Métricas de desempenho dos modelos . . . . .	38
4	RESULTADOS E DISCUSSÃO . . . . .	41
5	CONCLUSÕES E TRABALHOS FUTUROS . . . . .	49
	REFERÊNCIAS . . . . .	51

# 1 INTRODUÇÃO

O crescimento da demanda de energia elétrica ao redor do mundo tem intensificado o uso de fontes de energia renováveis, considerando a preocupação com o esgotamento dos recursos utilizados por fontes de energia não renováveis, também conhecidas como fontes de energia convencionais. A utilização do vento (energia eólica), da água (energia hidráulica) e do sol (energia solar), como fontes de energia, são exemplos comumente conhecidos de geradores de energia sustentável. O sistema de energia solar, entre os geradores de energia limpa, é o sistema mais vantajoso em termos de facilidade de utilização levando em conta que a radiação solar não necessita de uma estrutura complexa para ser utilizada, podendo ser implantado no âmbito doméstico e no âmbito industrial ([ABDULMAWJOOD; REFAAT; MORSI, 2018](#)). Um sistema de painéis fotovoltaicos, termo técnico empregado ao sistema de energia solar, consiste em um equipamento composto de materiais semicondutores, capaz de transformar a radiação solar em energia elétrica ([RENOVÁVEIS, 2019](#)).

Os sistemas de painéis fotovoltaicos podem ser categorizados em dois grupos, de acordo com o tipo de conexão realizada para controle da eletricidade gerada: os sistemas *on-grid* e os sistemas *off-grid*. Os sistemas *off-grid* possuem uma estrutura mais simples que os sistemas *on-grid*, atuando desconectados da rede elétrica pública, sendo sua energia utilizada para alimentar sistemas de armazenamentos, como baterias. Sob outra perspectiva, os sistemas *on-grid* possuem uma composição mais robusta, operando conectados à rede elétrica pública, permitindo a aquisição de créditos caso a energia produzida seja maior do que a capacidade de consumo do imóvel em um período determinado ([BRASIL, 2021](#)). A complexidade do sistema de painéis fotovoltaicos *on-grid* permite a transmissão de altos níveis de tensão, demandando um processo rigoroso de monitoramento das condições gerais do sistema, visto que anomalias geram riscos de danificação na construção base de sustentação dos painéis ([DHERE; SHIRADKAR, 2012](#)).

O projeto de implementação de sistemas de painéis fotovoltaicos está sendo modernizado, com a inserção de sensores nas placas fotovoltaicas capazes de medir atributos do sistema em tempo real e alimentar um banco de dados composto por grandes quantidades de dados históricos. A criação de uma metodologia computacional capaz de verificar se o sistema de energia solar está funcionando de forma regular é de grande utilidade nesse cenário, visto que o monitoramento realizado por técnicos especializados pode ser subjetivo e custoso.

## 1.1 Justificativa

O uso de sistemas de painéis fotovoltaicos é uma alternativa promissora para atender as futuras demandas de energia elétrica do mundo, considerando que seu principal recurso, a radiação solar, não necessita de grandes mudanças no ambiente para ser utilizado, contribuindo para a proteção do meio ambiente de uma forma geral. Além disso, o fato desses geradores possuírem a possibilidade de serem implementados no âmbito doméstico, distribuindo o excedente de energia gerado pelo sistema em troca de créditos na concessionária elétrica local, pode ser um grande aliado para aumentar a renda nacional da população de um país através da economia no gasto mensal da conta de luz ([INTELBRAS, 2021](#)).

Os sistemas fotovoltaicos são compostos por componentes conectados em série e paralelo, e cada uma das conexões elétricas possui uma empregabilidade distinta na geração de energia. O acoplamento em série é realizado para aumentar a tensão gerada, e o acoplamento em paralelo para aumentar a corrente. As conexões elétricas são necessárias para gerar uma quantidade de potência a um nível utilizável, devido ao fato que o menor componente do sistema, a célula, sozinha gera uma quantidade mínima de potência. As células conectadas em série formam os módulos do sistema, a conexão em série dos módulos dão origem às *strings* e as *strings* conectadas em paralelo formam o arranjo fotovoltaico, alcançando assim a tensão e corrente final desejada ([SINGH, 2013](#)).

A maioria dos sistemas possuem inversores de frequência responsáveis pela conversão da corrente contínua do sistema em corrente alternada, e pelo registro de informações de tensão, corrente e potência das *strings*. No entanto, há sistemas mais modernos que possuem rastreador de máxima potência a nível de módulo, capazes de registrar as medidas elétricas de forma mais detalhada, módulo a módulo ([SOLAREEDGE, 2021](#)). Essa tecnologia permite o acompanhamento mais preciso do comportamento do sistema, tendo sido pouco explorada na literatura.

O sistema fotovoltaico discutido nesse estudo foi construído utilizando os rastreadores de máxima potência a nível de módulo, conhecidos como otimizadores de potência ([SOLAREEDGE, 2021](#)). O projeto do sistema foi desenvolvido e implementado pensando em explorar as vantagens dessa tecnologia.

As metodologias tradicionais de monitoramento de desempenho em sistemas de painéis fotovoltaicos utilizam o gráfico da curva tensão-corrente como principal ferramenta de análise, sendo necessário um profissional competente da área para averiguar as conclusões pertinentes ([PEI; HAO, 2019](#)). Estudos buscando maneiras de aprimorar o processo de monitoramento sugeriram a utilização do aprendizado profundo de máquina para realizar esta tarefa. Os resultados dos estudos desenvolvidos se mostraram satisfatórios, considerando que os níveis de acurácia apresentaram valores maiores que 80%, contudo



restrições na estrutura dos métodos explorados tornam a implementação dos modelos em um sistema real complexa (SILVA, 2020; ZHAO; LEHMAN; BALL, 2013; FAN et al., 2020a).

## 1.2 Estado da Arte

O monitoramento do sistema fotovoltaico é de grande importância para a redução de perda de energia e custo de manutenção. Apesar de existirem várias propostas mais recentes de técnicas de monitoramento, o processo mais comum utilizado consiste no registro das informações de potência, corrente e tensão dos componentes do sistema, conhecidos como módulos, que são dispostas de forma gráfica para análise (MEYER; DYK, 2004). Além disso, a análise visual da estrutura do sistema também é utilizada como forma de monitoramento. Esses métodos são eficazes para detectar alterações físicas na estrutura do sistema e intercorrências ambientais, que influenciam de forma negativa a captação de energia, mas são ineficientes na detecção de anomalias elétricas que possam desencadear uma falha irreversível no sistema (ALAM et al., 2015).

Os sistemas mais modernos de painéis fotovoltaicos são construídos utilizando um sistema de armazenamento de dados, capaz de guardar uma grande quantidade de dados históricos. Estudos de modelos utilizando técnicas estatísticas e de aprendizagem de máquina surgem como alternativa para criação de um programa de monitoramento automatizado, de forma a aproveitar essa tecnologia. Modelos de previsão de séries temporais, utilizando abordagem estatística e aprendizado de máquina, foram explorados, considerando o benefício de conseguir detectar as anomalias antes de se tornarem de fato uma falha ou detectar o princípio da falha (SHARADGA; HAJIMIRZA; BALOG, 2020).

A modelagem de grafos é utilizada para representar matematicamente relações entre objetos de um mesmo conjunto. Os grafos podem ser utilizados como extratores de características similares em um banco de dados de imagem, para auxiliar na detecção de irregularidades em superfícies (WANG et al., 2022), assim como para representar estruturas físicas ou lógicas de um sistema fotovoltaico (FAN et al., 2020b), funcionando de maneira similar ao processamento de imagens, sendo capaz de abranger um número maior de possíveis comportamentos atípicos dessa maneira. Utilizar grafos para representar estruturas relacionais em modelos de previsão de séries temporais tem se mostrado bastante eficaz em algumas áreas de estudo como, por exemplo, o estudo da previsão de tráfego de carros considerando a disposição das rodovias (YU; YIN; ZHU, 2017), que utiliza um modelo de rede neural espaço-temporal de grafo para prever a quantidade média de carros que passa em determinada rodovia, em determinado intervalo de tempo. O problema de previsão de potência de um sistema de painéis fotovoltaicos é semelhante ao explorado no estudo de previsão de tráfego de carros (YU; YIN; ZHU, 2017), adaptando os métodos

propostos a um conjunto de dados de potência (VIDIGAL et al., 2022). O estudo utilizando os dados de potência de um sistema fotovoltaico foi promissor, visto que o modelo de rede neural espaço-temporal apresentou um resultado três vezes melhor que o resultado do modelo estatístico de séries temporais, considerando as medidas de desempenho definidas para comparação.

Uma outra abordagem avaliada para implementação em um programa de monitoramento automatizado de sistemas fotovoltaicos é a aplicação de modelos de classificação. A análise da curva característica tensão-corrente (curva característica I-V) foi proposta para identificar padrões de comportamento da energia gerada no sistema de painéis fotovoltaicos, utilizando a comparação do comportamento da curva I-V do sistema em funcionamento normal e funcionando em condições instáveis (ALI et al., 2017). A metodologia, apesar de começar a incluir em sua solução uma análise do fluxo da energia gerada pelo sistema, depende de uma análise gráfica crítica de um técnico especializado, visto que os valores ideais para cada cenário (funcionamento normal, funcionamento irregular) foram gerados por simulação, não levando em consideração possíveis fatores externos que podem alterar essas medidas.

Modelos de classificação de aprendizado de máquina supervisionado (BASNET; CHUN; BANG, 2020; SILVA, 2020) e semi-supervisionado (FAN et al., 2020b) foram explorados considerando algumas falhas elétricas e físicas do sistema, como circuito aberto, degradação dos módulos, entre outras. Os resultados foram satisfatórios para classificação, contudo os métodos desenvolvidos não consideram a dependência temporal dos dados e cada modelo tratou de um número pequeno de anomalias possíveis dentro do sistema. Além disso, o armazenamento de dados em um dos estudos foi realizado de forma totalmente controlada, considerando uma taxa de amostragem impraticável de ser realizada em tempo real durante o funcionamento do sistema (SILVA, 2020).

A vantagem dos modelos de previsão em relação aos modelos de classificação é a possibilidade de identificar o momento em que o comportamento das métricas de avaliação foi alterado e, definido um limite de variação, se a alteração indica um mau funcionamento do sistema, sendo a desvantagem a não identificação da falha que está impactando o sistema. O problema de identificação da falha pode ser resolvido através de um estudo para categorizar os dados de série temporal do modelo. Os modelos preditores apresentaram resultados satisfatórios de desempenho. No entanto, apesar de ser possível identificar o momento de intercorrência do sistema utilizando modelos de previsão, os modelos construídos não possibilitam a localização do erro (identificação do módulo problemático).

## 1.3 Objetivos

O objetivo principal do trabalho é gerar um modelo computacional capaz de prever a potência gerada pelo sistema fotovoltaico, em um determinado momento do dia, utilizando aprendizado profundo de máquina aplicando teoria de grafos como método de pré-processamento dos dados.

Alcançar o objetivo principal consiste em executar os seguintes objetivos específicos:

- Definir a modelagem de grafo adequada para o conjunto de dados
- Análisar de forma descritiva os dados
- Definir e construir o modelo de rede neural
- Comparar o modelo gerado com outros modelos existentes na literatura

## 1.4 Método de Desenvolvimento do Trabalho

A exploração de modelos de previsão utilizando grafos para monitorar o funcionamento de um sistema fotovoltaico ainda é pouco discutida. O presente estudo utiliza a arquitetura de uma rede neural que aprende com os dados da topologia lógica de um sistema fotovoltaico e séries temporais de mudanças históricas de potência da rede para prever a potência de um determinado módulo em um momento futuro. Um modelo híbrido de convolução de grafos e redes neurais recorrentes *long short term memory* (LSTM) e um modelo de redes neurais recorrentes LSTM foram aplicados a um conjunto de dados de potência de uma rede fotovoltaica, com o objetivo de realizar uma comparação e definir os benefícios da aplicação da teoria de grafos no problema. Além disso, o conjunto de dados utilizado é o mesmo conjunto do estudo desenvolvido utilizando redes neurais espaço-temporal (VIDIGAL et al., 2022), tornando possível a comparação dos resultados de ambos os estudos. As comparações são realizadas utilizando métricas de desempenho baseadas nos erros de cada modelo.

Considerando a desvantagem do modelo de previsão, um modelo de classificação começou a ser desenvolvido e a versão inicial é apresentada no estudo. O projeto final, que necessita de mais experimentação, é descobrir um modelo ótimo capaz de prever e classificar uma falha em um sistema fotovoltaico.

O trabalho foi dividido em três partes: referencial teórico, metodologia experimental, análises e resultados. A revisão bibliográfica contém um resumo da literatura de pesquisas específicas sobre funcionamento de um sistema de painéis fotovoltaicos, séries temporais e modelos estatísticos, teoria de grafos e inteligência artificial. Na etapa de metodologia experimental são apresentados o conjunto de dados, tecnologias utilizadas no

desenvolvimento dos modelos, métricas para modelagem e avaliação do desempenho dos modelos, e todo escopo do experimento. O encerramento do estudo é realizado na etapa de resultados, que consiste na exposição das principais soluções, resultados do modelo de previsão e do modelo de classificação, análises dos resultados, considerações finais e trabalhos futuros.

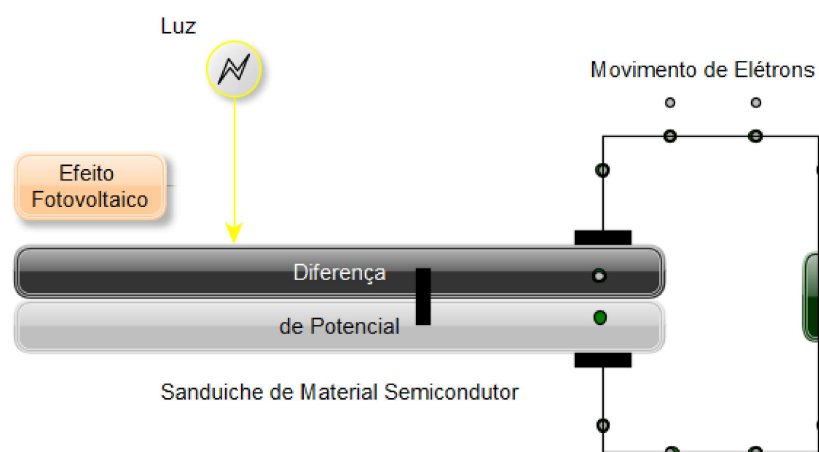
## 2 FUNDAMENTAÇÃO TEÓRICA

Essa parte do trabalho apresentará o referencial teórico utilizado no desenvolvimento do estudo e a descrição formal dos métodos utilizados. O objetivo é fornecer ao leitor uma base mínima de conhecimento das metodologias aplicadas no estudo.

A Seção 2.1 apresenta alguns conceitos para o entendimento do funcionamento de um sistema de painéis fotovoltaicos. A teoria de grafos e sua aplicação na análise dos dados é apresentada na Seção 2.2. A contextualização sobre dados do tipo série temporal é apresentada na Seção 2.3. O modelo estatístico e os métodos de aprendizagem profunda de máquina utilizados para manipulação e análise dos dados são apresentados nas Seções 2.4 e 2.5, respectivamente.

### 2.1 Sistemas Fotovoltaicos

O efeito fotovoltaico consiste na conversão direta da luz solar em energia elétrica, através de um fenômeno físico capaz de gerar uma diferença de potencial, ou uma tensão elétrica, sobre uma célula formada por um sanduíche de materiais semicondutores (BERTICELLI et al., 2017). A célula conectada a dois eletrodos permite que seja gerada uma tensão elétrica, ou uma corrente elétrica caso exista um caminho elétrico entre eles, como ilustrado na Figura 1.



Fonte: (BERTICELLI et al., 2017).

Nota: Adaptado pelo autor

Figura 1 – Efeito Fotovoltaico

O material semiconductor mais utilizado na fabricação das células fotovoltaicas é o silício, em razão de sua matéria prima, o quartzo, ser barata e abundante. Há um grande número de tecnologias bem desenvolvidas para a fabricação da célula, sendo as mais encontradas no mercado a do silício monocristalino e a do silício policristalino.

O silício monocristalino é produzido através do aquecimento, em altas temperaturas, de blocos de silício ultrapuro em um processo de formação de cristal conhecido como método de *Czochralski*, resultando em um lingote de silício monocristalino homogêneo, como exibido na Figura 2a. O lingote é serrado e fatiado para produzir *wafers*, que são discos finos de silício puro. Os *wafers* são submetidos a processos químicos nos quais recebem impurezas em ambas as faces, formando as camadas de silício *P* e *N* que constituem a base para o funcionamento da célula fotovoltaica. As células de silício monocristalino são as mais eficientes, mas tem o custo de produção mais elevado comparado aos outros tipos de células. O silício policristalino tem processo de fabricação similar ao silício monocristalino, sendo formado por um aglomerado de pequenos cristais, com tamanho e orientações diferentes, resultando em um lingote de silício policristalino heterogêneo, como apresentado na Figura 2b. As células de silício policristalino são inferiores, em termos de eficiência, às células de silício monocristalino, entretanto seu custo de fabricação é mais acessível, compensando no preço a diferença de eficiência (BERTICELLI et al., 2017).



Fonte: (CHAN, 2021).

(a) Silício Monocristalino



Fonte: (PLANAS, 2018).

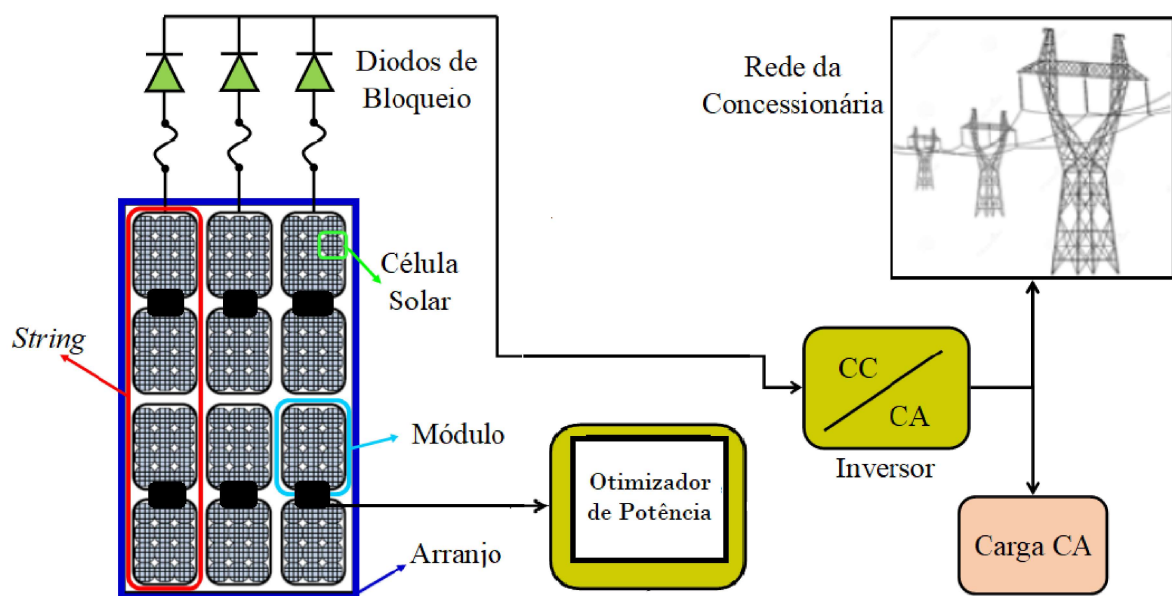
(b) Silício Policristalino

Figura 2 – Lingotes de Silício

Uma única célula de silício gera uma tensão pequena, em torno de  $0,5V$  a  $0,9V$ , e sua corrente gerada é proporcional à área, visto que a corrente elétrica depende diretamente da luz recebida pela célula, com valores de densidade de corrente entre  $16$  a  $42mA/cm^2$ , podendo gerar  $4,5W$  a  $8,1W$  em uma área de dimensão  $15 \times 15cm$  (SILVA, 2020). Considerando que a potência gerada por uma única célula é pequena, comparada à outros equipamentos elétricos, é necessário utilizar um número suficiente de células para gerar uma quantidade de potência a um nível utilizável. O agrupamento de células pode ser realizado de duas formas, com conexões elétricas realizadas em série e paralelo. A conexão em série é realizada para aumentar a tensão gerada, e a conexão em paralelo para aumentar

a corrente.

Um módulo fotovoltaico, também conhecido como painel ou placa, é o primeiro nível de agrupamento das células. Os módulos são compostos por dezenas de células conectadas em série e encapsuladas. Os módulos solares conectados em série constituem o segundo nível de agrupamento do sistema, esse agrupamento é conhecido como *string*, sendo a tensão final desejada alcançada nesse estágio do sistema. O terceiro e último agrupamento é realizado conectando as *strings* em paralelo para obter o nível de corrente desejado, formando o arranjo fotovoltaico com a capacidade de potência requerida (SINGH, 2013). A Figura 3 apresenta os componentes de um sistema fotovoltaico.



Fonte: (SILVA, 2020).

Nota: Adaptado pelo autor

Figura 3 – Componentes do Sistema Fotovoltaico.

Além do arranjo fotovoltaico, responsável por converter a energia solar em energia elétrica, um sistema fotovoltaico possui outros componentes importantes, sendo eles os diodos de bloqueio, o otimizador de potência e o inversor interativo. Os diodos de bloqueio tem a função de evitar que a corrente flua em sentido contrário em períodos de baixa irradiação solar, considerando que em condições normais a corrente elétrica flui sempre no mesmo sentido, comportamento característico de uma corrente contínua. Os otimizadores de potência são dispositivos projetados tomando como base a tecnologia *module level power electronics* (MLPE), que permite a divisão da produção e monitoramento de energia módulo a módulo, sendo instalados a cada grupo de dois módulos fotovoltaicos, tendo como princípio básico de funcionamento o rastreamento do ponto de máxima potência, do inglês *maximum power point tracking* (MPPT), que realiza o monitoramento contínuo dos parâmetros elétricos do módulo, otimizando a energia e buscando a melhor potência para

o funcionamento da placa solar, evitando assim a perda de potência de toda a *string* caso haja a perda de potência em um determinado módulo (SOLAREEDGE, 2020). O inversor interativo é um equipamento eletrônico que converte a corrente contínua em corrente alternada.

## 2.2 Teoria de Grafos

A teoria de grafos é uma área da matemática utilizada para modelar relações entre objetos de determinado conjunto. Tem aplicação em diversas áreas como engenharia, logística, sociologia, entre muitas outras (BENJAMIN; CHARTRAND; ZHANG, 2015). Algumas definições importantes para entendimento dos resultados deste estudo são apresentadas nesta seção (BONDY; MURTY et al., 1976).

**Definição 1** (*Grafo*). *A estrutura de grafo é definida como  $G = (V, E)$ , onde  $V$  é o conjunto de nós/vértices, e  $E$  o conjunto de arestas, sendo uma aresta definida como um subconjunto de dois vértices  $e_{ij} = \{v_i, v_j\}$  com  $v_i, v_j \in V$ . A identificação da vizinhança de um nó é facilmente realizada, sendo definida como  $N(v) = \{u \in V | \{u, v\} \in E\}$ .*

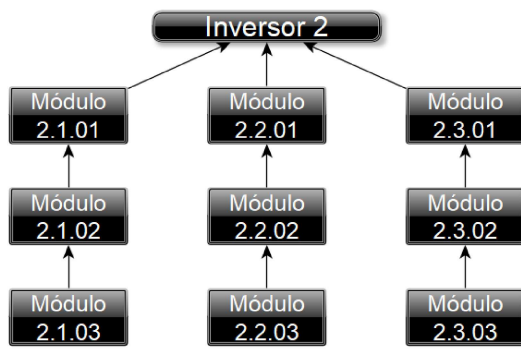
A modelagem de grafos é utilizada para descrever objetos e suas relações, consequentemente, cada vértice e cada aresta do grafo tem um significado, sendo possível indentificar e rotular cada componente (NETTO; JURKIEWICZ, 2017). Na representação de grafos os vértices são definidos como adjacentes quando há uma aresta entre eles, enquanto que uma aresta é incidente aos vértices que ela conecta, e são consideradas adjacentes quando incidem em um mesmo vértice.

**Definição 2** (*Grau*) *O grau de um vértice  $v$ , denotado por  $\text{grau}(v)$ , é a quantidade de arestas que nele incidem.*

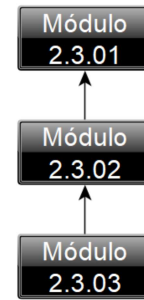
As partes de um grafo que originam outros grafos, formalmente, são denominadas subgrafos. Por definição, o conjunto de vértices de um subgrafo  $H$  de  $G = (V, E)$  está contido no de  $G$  e seu conjunto de arestas também está contido no de  $G$  (BONDY; MURTY et al., 1976).

**Definição 3** (*Subgrafo*) *Seja  $H = (V_H, E_H)$  e  $G = (V_G, E_G)$ , o grafo  $H$  é considerado um subgrafo do grafo  $G$  quando  $V_H \subseteq V_G$  e  $E_H \subseteq E_G$ . A Figura 4b apresenta um subgrafo  $H$  do grafo  $G$  exibido na Figura 4a.*





Fonte: Autor.

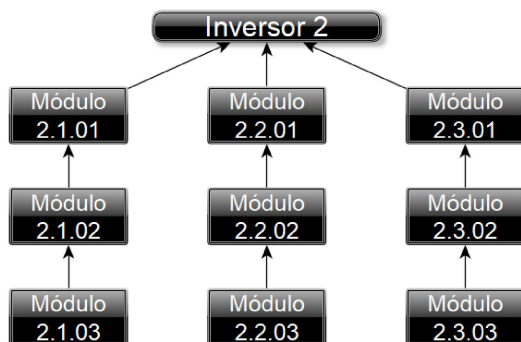
(a) Grafo  $G$ 

Fonte: Autor.

(b) Subgrafo  $H$  de  $G$ 

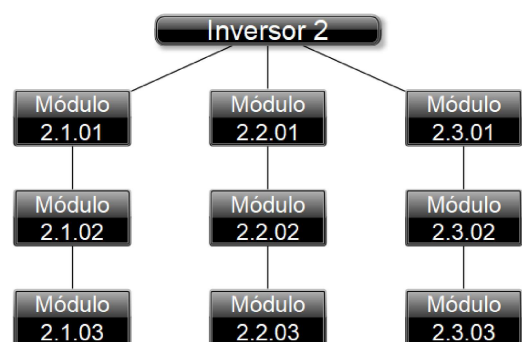
Figura 4 – Subgrafo

**Definição 4** (*Grafo Direcionado*). Um grafo  $G = (V, E)$  é dito direcionado quando o conjunto  $E$  de arestas é definido por pares  $\{v_i, v_j\} \in E$  de vértices ordenados, ou seja,  $\{v_i, v_j\} \neq \{v_j, v_i\}$ . A relação de interesse é indicada graficamente por uma seta, como apresentado na Figura 5a. Consequentemente, um grafo é dito não direcionado quando o conjunto  $E$  de arestas é definido por pares  $\{v_i, v_j\} \in E$  de vértices não ordenados, sendo  $\{v_i, v_j\} = \{v_j, v_i\}$ , como apresentado na Figura 5b.



Fonte: Autor.

(a) Grafo Direcionado

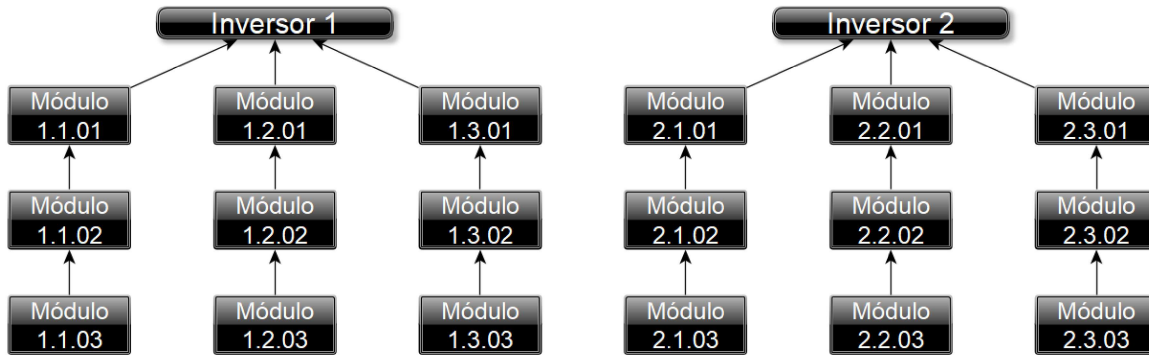


Fonte: Autor.

(b) Grafo não Direcionado

Figura 5 – Grafos

O número  $n$  de vértices em um grafo é chamado de ordem do grafo, enquanto que o número  $m$  de arestas é conhecido como o tamanho do grafo. Um caminho em um grafo é uma sequência linear de vértices conectados por uma sequência de arestas. Um grafo é definido como conexo quando há pelo menos um caminho entre cada par de vértices, como os grafos apresentados na Figura 5, e é denominado desconexo caso contrário, como mostra a Figura 6. É intuitivo concluir que os grafos conexos, que compõem o grafo desconexo, são subgrafos do mesmo.



Fonte: Autor.

Figura 6 – Grafo Desconexo: Representação da topologia lógica de um sistema fotovoltaico com dois inversores e dezoito módulos.

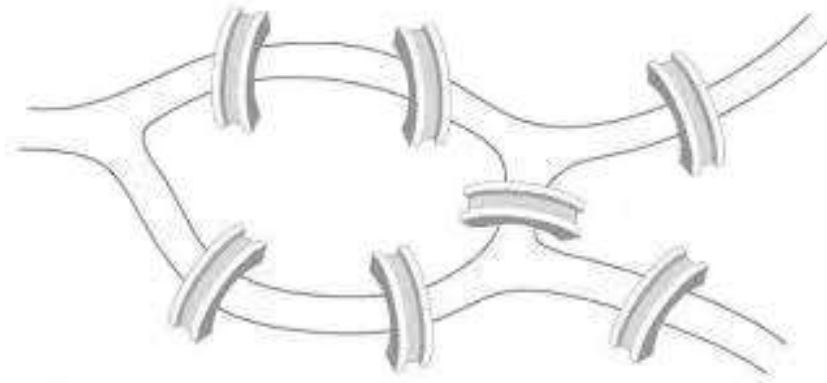
**Definição 5** (*Matriz de Adjacência*). Um grafo pode ser representado pela sua matriz de adjacência  $\mathbf{A}^{n \times n}$ , com  $n$  igual a quantidade de nós do grafo e as componentes da matriz seguindo a lógica apresentada na equação 2.1.

$$\begin{cases} \mathbf{A}_{ij} = 1 & , \text{ se } e_{ij} \in E \\ \mathbf{A}_{ij} = 0 & , \text{ se } e_{ij} \notin E \end{cases} \quad (2.1)$$

Os modelos relacionais representados em forma de grafos podem conter informações adicionais sobre seus objetos e relações, sendo essas informações representadas em uma matriz de atributos de nós ou matriz de informações de arestas.

**Definição 6** (*Matriz de Atributos*) Uma matriz de atributos de nós pode ser definida como  $\mathbf{Z}$ , onde  $\mathbf{Z} \in \mathbb{R}^{t \times n}$ , sendo  $t$  o intervalo de tempo e  $n$  a quantidade de nós, com  $\mathbf{z}_v \in \mathbb{R}^t$  representando o vetor de característica do nó  $v$ . Quando há mudança dinâmica dos valores dos atributos dos nós ao longo do tempo, o grafo atribuído é nomeado de grafo espaço-temporal e sua definição é dada pela expressão  $G^{(t)} = (\mathbf{V}, \mathbf{E}, \mathbf{Z}^{(t)})$  com  $\mathbf{Z}^{(t)} \in \mathbb{R}^{n \times d}$ .

O primeiro registro de uma estrutura matemática usada para modelar relações entre objetos de um mesmo conjunto, dando origem ao que hoje é conhecido como teoria dos grafos, remonta a 1736 (NETTO; JURKIEWICZ, 2017). O problema consistia em duas ilhas que eram ligadas entre si por uma ponte e se ligavam ainda as margens por mais seis pontes ao todo, como apresentado na Figura 7.



Fonte: (JR, 2022).

Figura 7 – Problema de Euler

O desafio era encontrar um caminho para passar pelas sete pontes, mas sem passar duas vezes pela mesma ponte. O que o matemático Euler conseguiu foi, através da representação da topologia das regiões e pontes da cidade, provar que não existia uma solução para o problema. Ele provou que para que o passeio desejado fosse possível, seria necessário um rearranjo das pontes, tornando assim o problema conhecido como Problema de Euler.

Desde Euler, a teoria dos grafos cresceu continuamente, se tornando um campo de interesse crescente, cujas aplicações vão de problemas de localização ao projeto de processadores eletrônicos.

## 2.3 *Séries Temporais*

Uma série temporal é uma sequência ordenada de dados, que pode ser determinada no domínio do tempo ou espaço. As séries temporais podem ser encontradas em várias áreas, entre elas a agricultura (séries de produção agrícola e preço), a medicina (exames de eletro-encefalograma e eletrocardiograma), as engenharias (sinais sonoros, elétricos e de vibrações), entre muitas outras (WEI, 2006). Séries temporais memorizadas em intervalos de tempo específicos como, por exemplo, representação diária do volume de vendas de um comércio, são consideradas séries temporais discretas. Sinais de eletricidade, vibração e tensão costumam ser memorizados de forma contínua no tempo, e são definidos como séries temporais contínuas.

Há vários objetivos no estudo de séries temporais, entre eles, realizar previsão de valores futuros, controle ótimo de processos, dentre outros. O interesse do presente estudo é entender e descrever o mecanismo de geração da série de potência de um sistema fotovoltaico, e prever o dado um passo à frente. A implementação do processo apresentado utilizando técnicas computacionais é denominado previsão de séries temporais. A previsão

de séries temporais pode ser realizada utilizando diversas técnicas computacionais, sendo duas delas a implementação de modelos estatísticos e o aprendizado profundo de máquina.

## 2.4 Modelos Estatísticos

Um processo estocástico é uma família de variáveis aleatórias, que supõe-se ser definida em um mesmo espaço de probabilidade  $(\Omega, A, P)$ , seja  $\Omega$  o espaço amostral,  $A$  o evento ou coleção de possíveis resultados do experimento e  $P$  a função de probabilidade. Sua representação matemática é dada pela equação (2.2), sendo  $T \subset \mathbb{R}$ .

$$X = \{X_t, t \in T\} \quad (2.2)$$

Uma série temporal é uma realização ou trajetória de um processo estocástico e o objetivo ao se analisar dados em série com técnicas estatísticas é investigar seu mecanismo gerador, ou seja, o processo estocástico e suas características, permitindo a modelagem e extrapolação do comportamento da série para valores futuros (MORETTIN; TOLOI, 2018). A utilização de modelos estatísticos de séries temporais para previsão é indicada em casos que as suposições realizadas na construção do modelo se mostrem verdadeiras.

Uma série pode ser decomposta de forma aditiva como a soma de três componentes não observáveis, da forma apresentada pela equação (2.3), com  $\epsilon_t$  sendo uma componente aleatória não correlacionada no tempo, usualmente conhecida como ruído branco, que tem como condição necessária ser estacionária, com média zero e variância  $\sigma_\epsilon^2$ , e  $\mathbf{t}_t$  e  $\mathbf{s}_t$  representando a tendência e a sazonalidade, respectivamente.

$$\mathbf{z}_t = \mathbf{t}_t + \mathbf{s}_t + \epsilon_t \quad (2.3)$$

A análise gráfica da série temporal é importante para identificar possíveis padrões de comportamento dos dados. Conjuntos de dados que possuem a variância da série em função da sua média  $\mu_t$ , como apresentado na equação (2.4), sendo  $c$  um valor constante e  $f(\cdot)$  a função de transformação, necessitam de alteração para estabilizar sua variância. Esse comportamento é confirmado realizando um teste estatístico para detectar os pontos de mudança dos parâmetros e é denominado teste CUSUM (BROWN; DURBIN; EVANS, 1975). A transformação *boxcox*, apresentada na equação (2.5), normalmente, é utilizada como tratamento. A transformação *boxcox* é aplicada antes de qualquer outra análise e só faz sentido em dados positivos. O valor de  $\lambda$  da equação (2.5) pode ser estimado utilizando um otimizador de estimação de máxima verossimilhança, sendo o critério maximizar o coeficiente de correlação do gráfico de quantis da probabilidade normal (BOX; COX, 1964).

$$\text{Var}(\mathbf{z}_t) = c \times f(\mu_t) \quad (2.4)$$

$$T(\mathbf{z}_t) = \frac{z_t^\lambda - 1}{\lambda} \quad (2.5)$$

Modelos estatísticos de série temporal necessitam que o conjunto de dados seja estacionário, sendo a estacionariedade a garantia da precisão dos valores previstos. Um série é estacionária nos momentos de 2ª ordem quando a média  $\mu_t$  e a variância  $\sigma_t^2$  são invariantes ao longo do tempo e a covariância entre valores defasados  $\gamma(t, t - k)$  da série depende apenas da defasagem  $k$  (WEI, 2006), como apresentado nas equações (2.6) e (2.7). A verificação da estacionariedade dos dados pode ser feita utilizando o teste Dickey Fuller Aprimorado (ADF) e o teste Kwiatkowski-Phillips-Schmidt-Shin (KPSS) (SEABOLD; PERKTOLD, 2010). A rejeição da estacionariedade de ambos os testes indica que os dados são não estacionários por diferença, tornando necessário o uso da diferenciação para tornar a série estacionária.

$$\mu_t = \mu, \quad \forall t \in \mathbb{Z} \quad (2.6)$$

$$\gamma(t, t - k) = \gamma_k, \quad \forall k \in \mathbb{Z} \quad (2.7)$$

### 2.4.1 Processos sazonais autoregressivos integrados de médias móveis (SARIMA)

Uma metodologia bastante utilizada na análise de modelos paramétricos é conhecida como abordagem de Box e Jenkins (JERE; KASENSE; BWALYA, 2017). Tal metodologia consiste em ajustar modelos auto-regressivos integrados de médias móveis (ARIMA) ao conjunto de dados, atentando para as informações descritivas exploradas.

O modelo ARIMA é uma classe de modelos lineares que utiliza valores históricos para prever valores futuros e cada uma das técnicas (processos autoregressivos, integração e processos médias móveis) contribui para a previsão do valor final.

No modelo de autoregressão (AR) a previsão da variável de interesse é realizada utilizando uma combinação linear de valores históricos da variável. O termo autoregressão indica que é uma regressão da variável com ela mesma. Em outras palavras, os valores defasados da variável de destino é utilizado como dados de entrada para prever valores para o futuro (WEI, 2006). O processo  $\mathbf{z}_t$  satisfaz o modelo AR de ordem  $p$  se é solução da equação de diferenças apresentada na Equação 2.8, sendo  $\phi_1, \phi_2, \dots, \phi_p \in \mathbb{R}$  os coeficientes da regressão e  $\epsilon_t$  o ruído branco gaussiano.

$$z_t = \phi_1 z_{t-1} + \phi_2 z_{t-2} + \dots + \phi_p z_{t-p} + \epsilon_t \quad (2.8)$$

O processo de integração representa qualquer diferenciação que deva ser aplicada para tornar os dados estacionários. A estacionariedade dos dados pode ser validada utilizando testes estatísticos como descrito na introdução da Seção 2.4. Atestado a não estacionariedade dos dados, diferentes fatores  $d$  de diferenciação são testados na construção do modelo, de modo a definir a ordem de diferenciação que torne os dados estacionários.

O modelo de médias móveis (MA) utiliza os erros das previsões passadas, ao invés do histórico de valores passados em um modelo semelhante a regressão para prever valores futuros. O processo  $\mathbf{z}_t$  satisfaz o modelo MA de ordem  $q$  se satisfaz a equação de diferenças apresenta na Equação 2.9, sejam  $\theta_1, \theta_2, \dots, \theta_q$  os coeficientes da regressão.

$$z_t = \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \dots - \theta_q \epsilon_{t-q} \quad (2.9)$$

A variável  $\{\epsilon_i | i \in [t - q, t]\}$  é chamada de erro e representa os desvios residuais aleatórios entre o modelo e a variável alvo. É preciso destacar que o valor  $\epsilon_i$  só pode ser determinado após o ajuste do modelo, fazendo com que esse seja um parâmetro não observável na modelagem. Assim, para resolver a equação 2.9, técnicas iterativas como Estimativa de Máxima Verossimilhança são usadas (WEI, 2006).

No caso em que se identifique a presença de sazonalidade no conjunto de dados o mais indicado é o ajuste de modelos SARIMA  $(p, d, q) \times (P, D, Q)_s$ , onde  $p$  refere-se à ordem autoregressiva do modelo,  $d$  o grau de diferenciações,  $q$  a ordem da parte de média móveis do modelo. Os valores  $(P, D, Q)$  tem a mesma interpretação dos valores  $(p, d, q)_s$  aplicados à parte sazonal do modelo, sendo  $s$  a ordem de sazonalidade. Uma das suposições do modelo SARIMA é a de que os resíduos  $\hat{\epsilon}_t$  sejam um ruído branco (RB), ou seja, o comportamento de  $\hat{\epsilon}_t$  deve ser semelhante ao comportamento de um processo não correlacionado, estacionário com média zero e desvio padrão  $\sigma_\epsilon^2$ .

O procedimento de Portmanteau (PANKRATZ, 2012) é utilizado para verificar se as correlações e autocorrelações dos resíduos do modelo são significativamente iguais a zero. Para o teste utiliza-se a estatística de Ljung-Box Portmanteau (LJUNG; BOX, 1978) definida na equação (2.10), onde  $n$  é o tamanho do conjunto de dados,  $r_{\epsilon\epsilon}^2(h)$  a correlação dos resíduos com defasagem  $h$ . Quando o modelo explica adequadamente a média e a variabilidade presente nos dados,  $\mathcal{Q}(\mathcal{H})$  converge para uma variável com distribuição Qui-Quadrado com  $(\mathcal{H} - p - q)$  graus de liberdade. A hipótese nula de correlação igual a zero será rejeitada quando  $\mathcal{Q}(\mathcal{H})$  assumir valores altos.

$$\mathcal{Q}(\mathcal{H}) = n(n+2) \sum_{h=1}^{\mathcal{H}} \frac{r_{\epsilon\epsilon}^2(h)}{n-h} \quad (2.10)$$

Além da verificação da hipótese de autocorrelação nula, uma outra suposição deve

ser verificada, a de que os resíduos sejam normalmente distribuídos. Existem ferramentas gráficas de verificação da distribuição dos dados tais como o histograma, o box-plot e o gráfico de quantis, no entanto são úteis apenas numa análise descritiva. É necessário verificar a normalidade dos dados por meio de um teste de hipóteses.

## 2.5 Aprendizado Profundo de Máquina

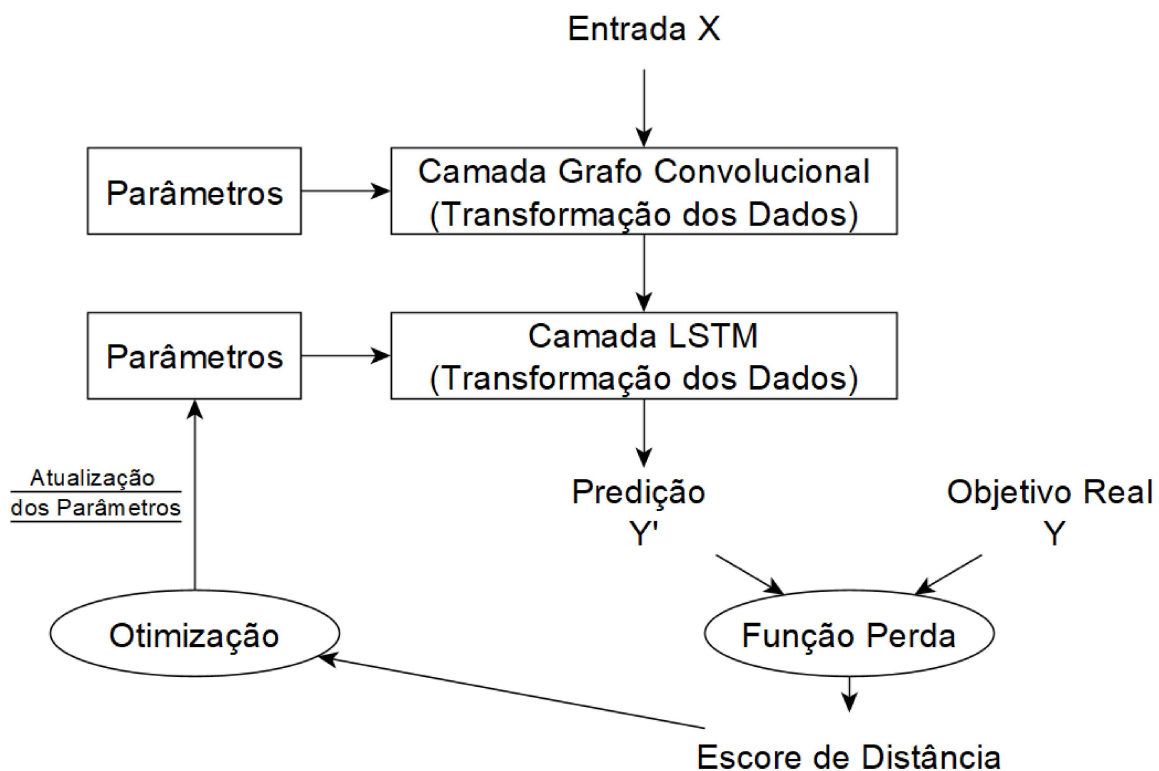
O aprendizado de máquina (*Machine Learning*) é uma área da inteligência artificial que utiliza técnicas computacionais para habilitar computadores a aprender sozinhos como realizar tarefas específicas, adquirindo conhecimento de forma automática (CHOLLET; ALLAIRE, 2018). Tecnicamente, o aprendizado de máquina busca representações úteis para um conjunto de dados de entrada, com um espaço de possibilidades predefinido, utilizando orientação de algum sinal de *feedback*. A habilidade de aprendizado permite a generalização do conhecimento adquirido para um conjunto de dados jamais visto pelo algoritmo computacional implementado na máquina (MITCHELL; MITCHELL, 1997).

Uma rede neural é uma técnica de aprendizagem de máquina inspirada na maneira como o cérebro humano realiza uma tarefa particular ou uma função de interesse. Análogo à estrutura cerebral, a rede neural consiste em um processador constituído de unidades de processamento simples (neurônios), distribuídos de forma paralela e condensada (camadas), que tem a capacidade de armazenar conhecimento experimental e torná-lo disponível para uso. O processo utilizado para realizar o processo de aprendizagem é chamado de algoritmo de aprendizagem (HAYKIN, 2007).

O algoritmo de aprendizagem utiliza dados de entrada  $x_1, x_2, x_3, \dots, x_n$ , criando ligações de acesso que são dinamicamente reforçadas ou enfraquecidas com base na frequência com que são usadas. Matematicamente, é atribuído a cada dado de entrada um peso específico  $w_1, w_2, w_3, \dots, w_n$ . A força das respectivas conexões, representada pelos pesos, realiza uma ponderação de cada dado, gerando o que se pode chamar de *logit* do neurônio,  $z = \sum_{i=1}^n x_i w_i$ , o *logit* sempre contém um viés agregado a ele. O *logit* é então submetido a uma função  $f$ , produzindo um saída  $y = f(z)$ , podendo essa saída ser os dados de entrada do próximo *logit* ou ser o resultado final do algoritmo (BUDUMA; BUDUMA; PAPA, 2022).

O aprendizado profundo de máquina compreende uma rede neural composta por uma grande quantidade de camadas, conseqüentemente, uma grande quantidade de neurônios. As camadas intermediárias da rede, também chamadas de camadas ocultas, são definidas de acordo com o objetivo do modelo e características dos dados. No esquema operacional apresentado na Figura 8 as camadas intermediárias são do tipo Grafo Convolutacional e *Long Short-Term Memory* (LSTM), alternativas razoáveis para previsão de dados de séries temporais.

O método fundamenta-se no mapeamento de dados de entrada aos seus respectivos alvos, utilizando um processador extenso de camadas de neurônios. A especificação do que cada camada faz com seus dados é armazenada em pesos, também chamados de parâmetros da camada. O controle da saída da rede neural e a avaliação da precisão do seu mapeamento é realizado pela função de perda ou função objetivo da rede. A função objetivo considera a predição da rede neural e o verdadeiro objetivo e computa o escore de distância, sendo essa medida capaz de determinar o afastamento entre o valor de saída e o valor esperado. O escore de distância é então utilizado como sinal de *feedback*, pela função de otimização, para ajustar os valores dos parâmetros, buscando minimizar o valor do escore (CHOLLET; ALLAIRE, 2018). O esquema operacional é apresentado na Figura 8.



Fonte: (CHOLLET; ALLAIRE, 2018).  
Nota: Adaptado pelo autor

Figura 8 – Esquema Operacional do Aprendizado Profundo de Máquina

No momento inicial da retropropagação, os pesos das redes neurais são valores aleatórios e, naturalmente, as saídas da rede neural são muito distantes do valor real. Uma série de transformações aleatórias é implementada pela rede para ajustar esses parâmetros em um treinamento em ciclos. A cada processo da rede, os pesos são ajustados na direção correta e o escore de distância reduz seu valor. O treinamento é realizado um número suficiente de vezes, produzindo os valores dos pesos que minimizam a função de perda. A segmentação do conjunto de dados no processo de treinamento é essencial para garantir a



confiança do resultado, sendo normalmente subdividido em três subgrupos: treino, validação e teste. A rede utiliza os dados de treino e validação para realizar o ajuste dos parâmetros e os dados de teste para garantir a veracidade do resultado do ajuste.

### 2.5.1 Representação dos Dados para uma Rede Neural

As redes neurais lidam com dados arranjados na forma de tensores. Os tensores, no âmbito da computação, são uma generalização de vetores e matrizes em um número arbitrário de dimensões, e são determinados por três atributos chave: número de eixos, forma e tipo de dado.

O número de eixos consiste no número de dimensões do tensor, uma matriz por exemplo pode ser representada como um tensor bidimensional ou  $2D$ , logo, o número de eixos de uma matriz é 2. O atributo forma compreende um vetor de números inteiros que descreve a quantidade de dimensões que cada eixo possui, considerando que a matriz do exemplo anterior tenha duas linhas e três colunas, então a forma da matriz é dada por  $(2, 3)$ . É importante que as definições de número de eixos e forma do tensor não sejam embaralhadas. Por exemplo, um cubo composto por agregação de cinco matrizes, contendo duas linhas e três colunas cada uma, é um tensor  $3D$  de forma  $(2, 3, 5)$ . O tipo de dado pode ser definido de maneira intuitiva, consiste na definição do tipo de dado contido no tensor, podendo assumir os valores *integer*, *double (float)*, ou *character*.

A representação de um grafo utilizando matriz de adjacência é um tensor  $2D$  de forma (número de vértices, número de vértices), sendo o tipo de dado *integer*. Dados de série temporal contínua ou de sequência são tensores  $3D$  de forma (número de amostras, número de passos no tempo, número de características). Quando a natureza dos dados é univariada, a quantidade de características da série é igual a um. Isso implica que a série temporal poder ser apresentada por um tensor  $2D$  de forma (número de amostras, número de passos no tempo), sendo o tipo de dado *double*.

### 2.5.2 Tratamento de Dados para uma Rede Neural

O pré-processamento dos dados é necessário quando se utilizam métodos de redes neurais, dado que não é adequado alimentar uma rede com valores muito grandes ou dados que são heterogêneos. Isso se deve ao fato que a rede neural, para convergir os parâmetros para um mínimo global, necessita de valores padronizados. Normalmente, as variáveis de entrada da rede possuem duas características: são valores entre zero e um e são homogêneos entre si.

A normalização é uma técnica de pré-processamento dos dados que tem como objetivo a redução de escala e do *range* de variação dos dados, sendo uma das formas de implementação apresentada na equação 2.11, com  $x_i$  referente a cada dado de entrada da

rede.

$$\tilde{x}_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \quad (2.11)$$

### 2.5.3 Funções de Ativação

As transformações aprendidas por uma rede neural de aprendizado profundo podem ser reduzidas a um conjunto de operações de tensores como, por exemplo, adição de tensores, multiplicação de tensores, entre outras. Essas operações são aplicadas na rede neural através das funções de ativação. As funções de ativação são essenciais para dar capacidade representativa às redes neurais, introduzindo um componente de não linearidade. A utilização de uma função de ativação pode ser justificada com base no cenário e dados disponíveis.

A função de ativação ReLU, conhecida como unidade linear restrita, é uma função não linear, e pode ser definida como apresentado na equação (2.12). Nas equações (2.12), (2.13) e (2.14)  $\mathbf{x}$  é o dado de entrada,  $\mathbf{w}$  o vetor de pesos da rede neural e  $\mathbf{b}$  o vetor de vies.

$$f(\mathbf{x}) = \max[(\mathbf{w} \cdot \mathbf{x} + \mathbf{b}), 0], \quad (2.12)$$

A função na equação (2.12) tem sido utilizada em vários estudos, principalmente em cenários que abordam temas de visão computacional e classificação de séries temporais, que apresentam problemas de múltiplas classificações. Uma desvantagem da função ReLU, é que os neurônios tendem a expirar durante o treinamento, ou seja, começam a produzir somente saídas igual a zero. Isso acontece quando a soma ponderada antes da aplicação da ReLU se torna negativa, em consequência os parâmetros em  $\mathbf{w}$  tem suas atualizações interrompidas (MAAS; HANNUN; NG, 2013). A proposta para solucionar o problema da função ReLU é acrescentar uma pequena inclinação  $\alpha$  na parte negativa do seu domínio,

$$f(\mathbf{x}) = \begin{cases} (\mathbf{w} \cdot \mathbf{x} + \mathbf{b}) & , \text{ se } (\mathbf{w} \cdot \mathbf{x} + \mathbf{b}) > 0 \\ \alpha(\mathbf{w} \cdot \mathbf{x} + \mathbf{b}) & , \text{ c.c} \end{cases} . \quad (2.13)$$

A equação (2.13) é denominada função de ativação *LeakyReLU*. O valor  $\alpha$  da equação pode ser definido como um valor fixo igual a 0.01 (MAAS; HANNUN; NG, 2013), ou pode ser definido após uma análise de valor ideal do hiperparâmetro (XU et al., 2015).

As funções *sigmoid* e tangente hiperbólica são funções empregadas geralmente para modelar dados com comportamento binário, sendo a função tangente hiperbólica a escolha mais utilizada, sua definição é apresentada na equação (2.14).

$$f(\mathbf{x}) = \tanh(\mathbf{w} \cdot \mathbf{x} + \mathbf{b}), \quad (2.14)$$

É possível se definir uma função de ativação para cada camada ou neurônio da rede. Os métodos de seleção da função de ativação das camadas intermediárias e da camada de saída do modelo costumam ter critérios diferentes, sendo a função de ativação da última camada definida com base no objetivo da rede. Uma possibilidade dentro de problemas de classificação, é a última camada da rede neural retornar como saída um vetor de probabilidades estimadas, sendo que, cada valor do vetor corresponde à probabilidade do exemplo pertencer à uma das distintas categorias. O vetor de saída é da forma  $[p_1 \ p_2 \ \dots \ p_c]$ , onde  $c$  corresponde a quantidade de categorias definidas, e  $\sum_{i=1}^c p_i = 1$ . A categoria do objeto é então definida considerando o maior valor  $p_i$ ,  $i = 1, \dots, c$ . Esse vetor é alcançado utilizando a função de ativação *softmax*, e no caso da classificação binária, onde  $c = 2$ , esse vetor é alcançado utilizando a função de ativação *sigmoid*. Problemas de regressão ou previsão de valores, utilizam a função de ativação linear na última camada, sendo essa função também conhecida como identidade.

#### 2.5.4 Otimização Baseada em Gradiente

Os pesos ou parâmetros do modelo, representados nas equações (2.12), (2.13) e (2.14) pelos tensores  $\mathbf{w}$  e  $\mathbf{b}$ , inicialmente são preenchidos com valores aleatórios bem pequenos, procedimento conhecido como inicialização aleatória. Naturalmente, os valores definidos na inicialização aleatória não fornecem representações úteis. O processo que sucede à inicialização aleatória é o ajuste gradual dos parâmetros, baseado no sinal de *feedback*, denominado treinamento ou treinamento em *loop*.

A atualização dos parâmetros é o trabalho mais árduo da rede neural. No entanto, é possível explorar o fato de que todas as operações utilizadas nas redes neurais são diferenciáveis, calculando o vetor gradiente de perda em relação aos coeficientes das redes neurais e movendo os coeficientes na direção oposta do gradiente, decrescendo a perda.

#### 2.5.5 Algoritmo de Retropropagação

Na prática, a função da rede neural consiste em muitas operações de tensores encadeadas, cada uma delas com uma derivada simples e conhecida. É conhecido que,

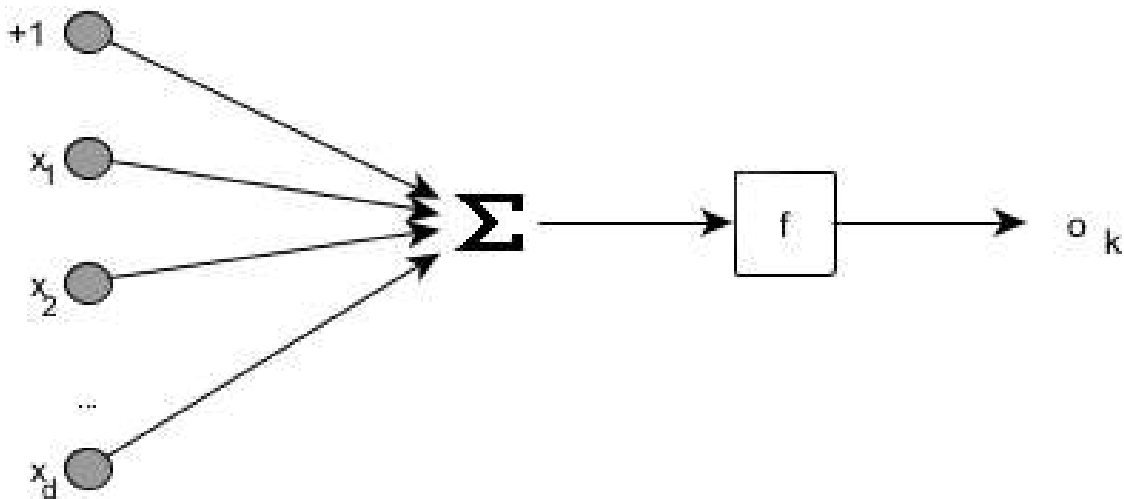
$$f(g(x)) \rightarrow f'(g(x)) = f'(g(x))g'(x), \quad (2.15)$$

desde que  $f(\cdot)$  e  $g(\cdot)$  sejam funções diferenciáveis. A identidade matemática apresentada pela Equação (2.15) é intitulada regra da cadeia. O algoritmo conhecido como retropropagação

é originado da aplicação da regra da cadeia para calcular os valores gradiente da rede neural. A retropropagação trabalha com o valor final da perda, das camadas superiores para as camadas inferiores, aplicando a regra da cadeia para calcular a contribuição de cada parâmetro no valor da perda.

### 2.5.6 Perceptron Multicamadas

A rede neural perceptron multicamadas, também conhecida como rede neural padrão, é uma rede neural composta por múltiplas camadas de perceptron. O perceptron é a rede neural mais simples, sendo composta por um único neurônio. A Figura 9 apresenta o esquema operacional de uma rede neural perceptron, sendo  $\Sigma$  o componente integrador da rede,  $f$  a função de ativação,  $+1$  o valor constante do modelo e os valores  $x_1, x_2, \dots, x_d$  e  $o_k$  os dados de entrada e de saída do modelo, respectivamente.



Fonte: (HAYKIN, 2007).  
Nota: Adaptado pelo autor

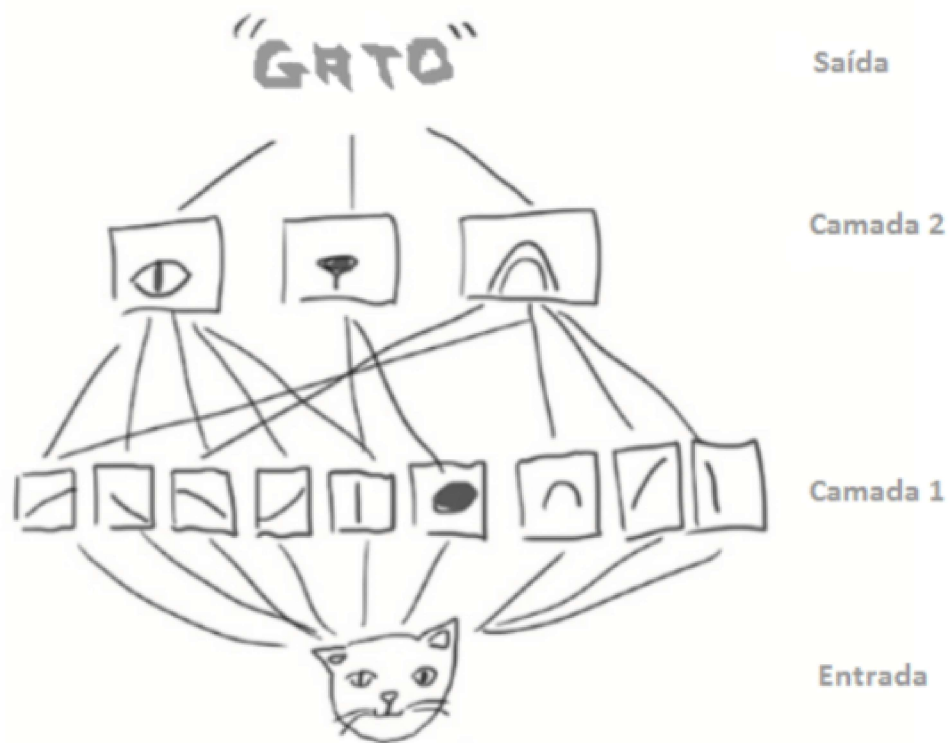
Figura 9 – Esquema Operacional Básico da Rede Neural Perceptron

A rede neural perceptron possui fácil implementação. Diferente da rede perceptron, as redes neurais multicamadas podem ser usadas e são eficientes como solução para problemas de classificação e regressão não-linear (HAYKIN, 2007).

### 2.5.7 Convolução de Grafos

Redes neurais convolucionais (CNN) são comumente definidas como algoritmos de aprendizado de máquina, que tem como dados de entrada um conjunto de imagens. As redes convolucionais são capazes de atribuir importância a vários aspectos da figura e, com

base na significância definida para cada aspecto, diferenciar padrões distintos (CHOLLET; ALLAIRE, 2018). Os algoritmos de aprendizado profundo CNN possuem duas propriedades interessantes: os padrões aprendidos pelo algoritmo são invariantes à translação e o modelo é capaz de aprender padrões espaciais hierárquicos. Invariância a translação é equivalente a identificar os padrões aprendidos pela rede, independentemente do local onde a característica se encontra na imagem. O aprendizado de maneira hierárquica consiste no fato que a primeira camada geralmente aprende pequenos padrões locais, como bordas, a segunda compreende padrões maiores construídos com as características da primeira camada e assim sucessivamente. A Figura 10 apresenta uma representação gráfica de uma rede neural convolucional.



Fonte: (CHOLLET; ALLAIRE, 2018).

Figura 10 – Esquema Operacional Básico da Rede Neural Convolucional

É importante ressaltar que os dados de entrada da rede neural convolucional não se restringem somente a imagens. Dados sequenciais também podem ser utilizados como dados de entrada da rede. Nesse contexto, similar a definição comum, as redes neurais convolucionais são desenvolvidas conforme a ideia de conectividade local. Os neurônios são conectados, cada um a apenas uma região dos dados de entrada. No algoritmo a conectividade local é alcançada substituindo as somas ponderadas por convoluções. Em cada camada da CNN, a entrada é convoluída com a matriz de peso para criar um mapa de recursos. Todos os mapas de recursos de saída compartilham os mesmos pesos, isso garante

que todos os neurônios de saída detectem exatamente o mesmo padrão. O aspecto da conectividade local e dos pesos compartilhados é favorável na redução do número total de parâmetros que podem ser aprendidos pela rede resultando em treinamentos e aprendizado mais eficiente.

As redes neurais convolucionais de grafos (GCNN) são modelos de redes neurais convolucionais que tem como dados de entrada a representação matricial de um grafo, podendo ser distribuídas em duas categorias, redes neurais de convolução espectral e convolução espacial. A abordagem convolução espacial é uma abordagem análoga à rede neural convolucional regular. Considerando que um grafo pode ser representado como uma imagem, a rede utiliza filtros para capturar padrões de relacionamentos e conseguir propagar a informação dos nós através das arestas.

O método de interesse neste trabalho é a abordagem convolução espectral, considerando que um grafo pode ser representado como um sinal de processamento, as camadas convolucionais da rede são interpretadas como multiplicadores do sinal  $x \in \mathbb{R}^n$  no espaço  $\theta$ , como apresentado na equação (2.16) (WU et al., 2020). Seja  $\Theta_{*G}x$  o sinal processado, a equação (2.16) apresenta  $\mathbf{U}$  como a matriz de autovetores da transformada laplaciana do grafo  $\mathbf{L} = \mathbf{I}_n - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ , com  $\mathbf{A}$  igual a matriz de adjacência do grafo,  $\mathbf{I}_n$  matriz identidade e  $\mathbf{D} \in \mathbb{R}^{n \times n}$  matriz diagonal de graus dos nós do grafo ( $\sum_j \mathbf{A}_{ij}$ ).

$$\Theta_{*G}x = \Theta(\mathbf{L})x = \Theta(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T)x = \mathbf{U}\Theta(\mathbf{\Lambda})\mathbf{U}^T x = \mathbf{U}g_\theta\mathbf{U}^T x \quad (2.16)$$

A função  $g_\theta$  presente na equação (2.16) pode assumir várias formas. A rede neural convolucional espectral *Chebyshev* (Cheb) utiliza o filtro  $g_\theta = \sum_{i=0}^k \theta_i T_i(\tilde{\mathbf{\Lambda}})$ , com  $\tilde{\mathbf{\Lambda}} = 2\mathbf{\Lambda}/\lambda_{\max} - \mathbf{I}_n$  variando no intervalo  $[-1, 1]$ , sendo  $T_i(\mathbf{x}) = 2\mathbf{x}T_{i-1}(\mathbf{x}) - T_{i-2}(\mathbf{x})$  o polinômio de *Chebyshev* com  $T_0(\mathbf{x}) = 1$  e  $T_1(\mathbf{x}) = \mathbf{x}$ . A aproximação de primeira ordem (1st) da convolução Cheb é definida utilizando  $k = 1$ .

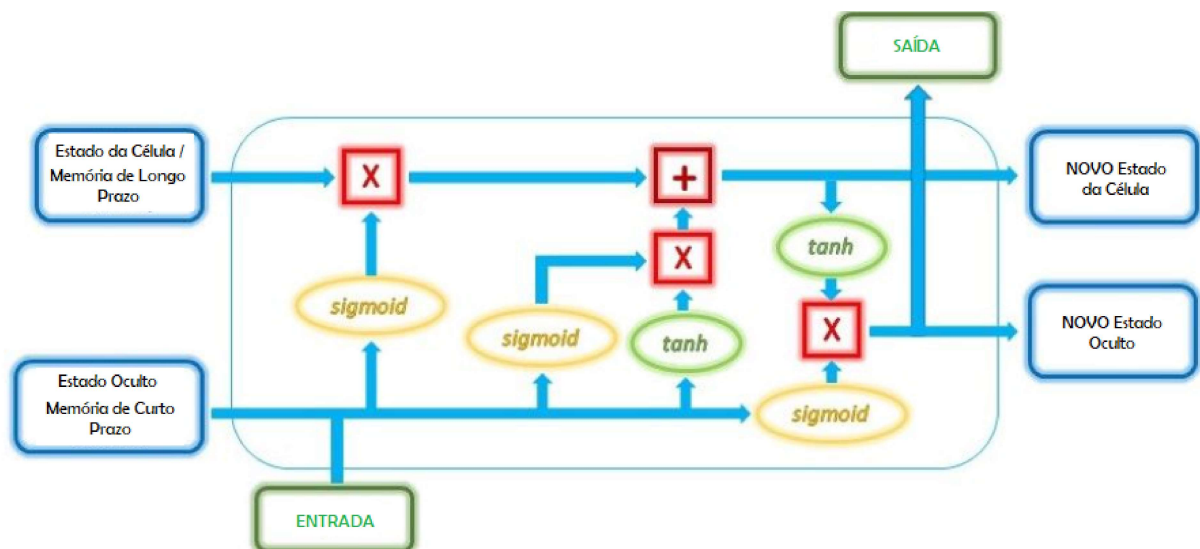
### 2.5.8 Aprendizado Profundo para Previsão de Séries Temporais

Uma característica importante das redes neurais convolucionais é que, as popularmente utilizadas, não têm memória. As entradas são processadas de forma independente, sem nenhum estado entre elas, tornando o procedimento com dados de entrada sequenciais eficaz em tarefas de classificação, mas improdutivo em tarefas de previsão (CHOLLET; ALLAIRE, 2018).

As redes tradicionais, conhecidas como *feed-forward*, realizam a alimentação das camadas de forma sequencial, ou seja, a saída de uma camada é automaticamente entendida como entrada para a camada subsequente. As redes neurais recorrentes são algoritmos de aprendizado de máquina que possuem conexões ponderadas dentro de uma camada que

funcionam como *loops*, isso permite o armazenamento de informações ao processar novas entradas (KIM, 2020).

Redes neurais (LSTM), são um tipo de rede recorrente, capaz de armazenar e utilizar a dependência temporal do dados, considerando uma grande janela temporal. Uma camada LSTM consiste em um conjunto de blocos conectados de forma recorrente, conhecidos como bloco de memórias, cada um contendo uma ou mais células de memórias e três unidades multiplicativas: a unidade de entrada, unidade de saída e a unidade do “esquecimento”. As unidades multiplicativas fornecem operações de escrita, leitura e *reset* para as células (BROWNLEE, 2017). A Figura 11 apresenta o esquema operacional de uma camada LSTM.



Fonte: (LOYE, 2019).  
Nota: Adaptado pelo autor

Figura 11 – Esquema Operacional Básico da Camada LSTM.

A unidade de entrada é responsável pela seleção da nova informação que será armazenada na memória de longo prazo da camada. Matematicamente, a seleção é realizada utilizando duas camadas conectadas. A primeira camada exerce a função de filtro, e é criada considerando a função de ativação *sigmoid* ( $\sigma$ ). A função *sigmoid* utiliza as informações da entrada atual e da memória de curto prazo do intervalo de tempo anterior para avaliar as informações relevantes que devem ser escolhidas para serem armazenadas e também os valores que devem ser descartados, como demonstra a Equação 2.17. O treinamento é realizado por meio da retropropagação, isso permite que os pesos  $\mathbf{W}$  sejam atualizados de forma que o modelo aprenda a deixar apenas o útil passar, enquanto descarta os recursos menos críticos.

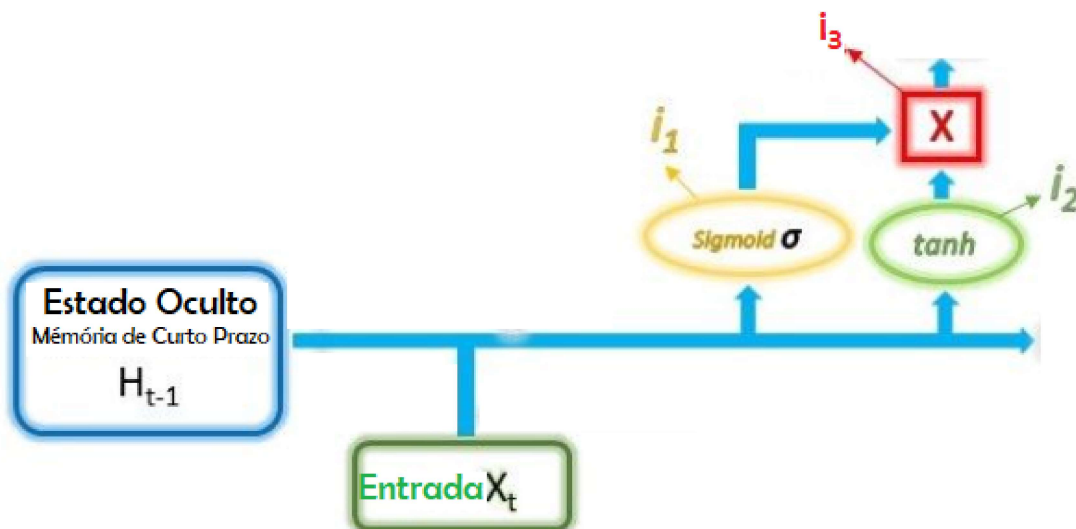
$$i_1 = \sigma(W_{i1} \cdot (H_{t-1}, x_t) + viés) \quad (2.17)$$

A segunda camada da unidade de entrada, assim como a primeira, utiliza as informações da entrada atual e da memória de curto prazo do intervalo de tempo anterior. A função de ativação utilizada, geralmente, é a função  $\tanh$ , e esse processo funciona como um regulador da rede. A Equação 2.18 apresenta a lógica matemática do procedimento.

$$i_2 = \tanh(W_{i_2} \cdot (H_{t-1}, x_t) + viés_{i_2}) \quad (2.18)$$

O resultado final do processamento da unidade de entrada da camada consiste no produto das saídas das duas camadas apresentadas, Equação 2.19, e representa a informação a ser mantida na memória de longo prazo e usada como saída (LOYE, 2019). A Figura 12 apresenta o esquema operacional da unidade de entrada da camada LSTM.

$$i_3 = i_1 \times i_2 \quad (2.19)$$



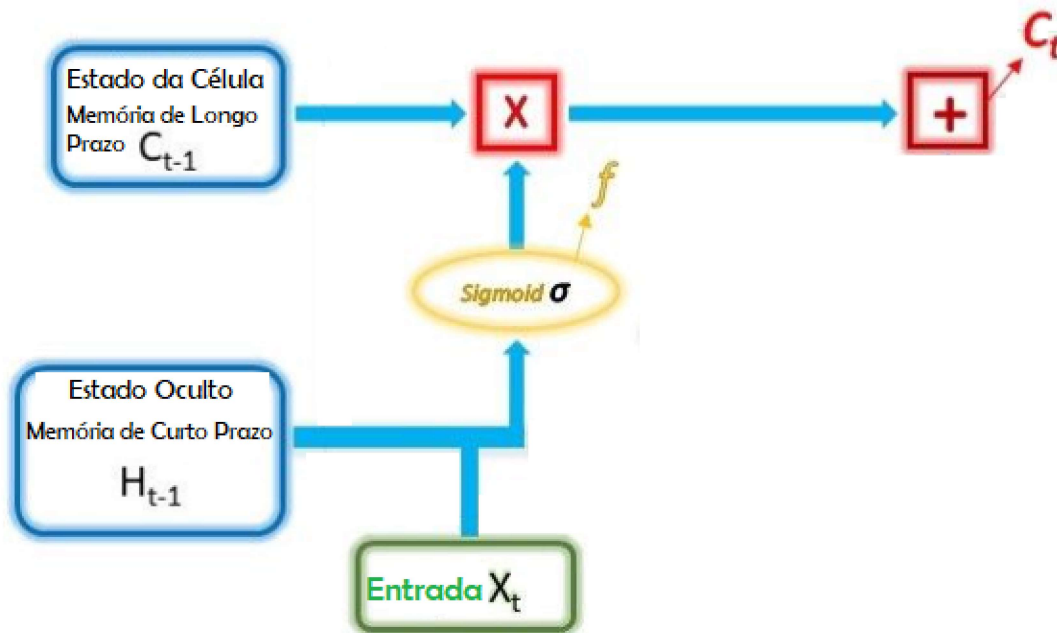
Fonte: (LOYE, 2019).  
Nota: Adaptado pelo autor

Figura 12 – Unidade de Entrada da Camada LSTM.

Análogo à primeira camada da unidade de entrada, a unidade do “esquecimento” trabalha como um filtro, indicando quais informações da memória de longo prazo devem ser mantidas ou descartadas (LOYE, 2019). A operação é realizada multiplicando a informação da memória de longo prazo do intervalo de tempo anterior por um vetor de esquecimento gerado pela entrada atual e o estado oculto, como demonstrado na Figura 13 e descrito na Equação 2.20.



$$f = \tanh(W_{\text{esquecido}} \cdot (H_{t-1}, x_t) + \text{viés}_{\text{esquecido}2}) \quad (2.20)$$



Fonte: (LOYE, 2019).  
Nota: Adaptado pelo autor

Figura 13 – Unidade de “Esquecimento” da Camada LSTM.

Uma adição pontual é realizada com as saídas da unidade de entrada e da unidade de esquecimento, fornecendo uma nova versão da memória de longo prazo, que será transportada para a próxima célula, o processo é apresentado Equação 2.21.

$$C_t = C_{t-1} \times f + i_3 \quad (2.21)$$

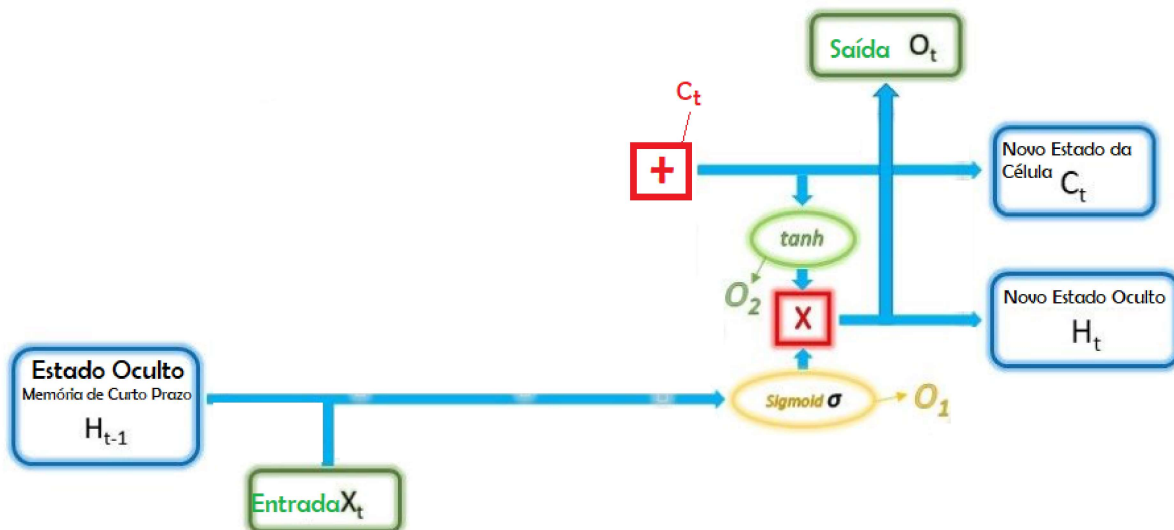
A unidade de saída é a última unidade multiplicativa da camada. Ela utiliza a entrada atual, o estado oculto e a memória de longo prazo recém-computada para produzir um novo estado oculto e a saída da camada, como apresentado na Figura 14. O terceiro e último filtro utilizando a função *sigmoid* é criado, utilizando a lógica da Equação 2.22, operando a memória de curto prazo anterior e a entrada atual. A nova memória de longo prazo é adicionada ao processo por meio de uma camada que utiliza a função de ativação

$\tanh$ , como mostra a Equação 2.23, e esse valor é multiplicado ao valor de saída do filtro para produzir a nova memória de curto prazo e a saída atual, como descrito na Equação 2.24. Esse processo se repete para todos os instantes  $t$  de tempo .

$$O_1 = \sigma(W_{saída1} \cdot (H_{t-1}, x_t) + viés_{saída2}) \quad (2.22)$$

$$O_2 = \tanh(W_{saída2} \cdot (H_{t-1}, x_t) + viés_{saída2}) \quad (2.23)$$

$$H_t, O_t = O_1 \times O_2 \quad (2.24)$$



Fonte: (LOYE, 2019).  
Nota: Adaptado pelo autor

Figura 14 – Unidade de Saída da Camada LSTM.

As redes neurais LSTM, diferente das redes convencionais, não realizam o ajuste de parâmetros através do processamento em ciclos. O ajuste de parâmetros é realizado utilizando as novas entradas como pesos, ponderando cada peso com as informações obtidas através da realimentação das saídas de um ou mais neurônios (HAYKIN, 2007).

## 3 METODOLOGIA EXPERIMENTAL

Nesta parte do trabalho, apresenta-se o escopo do experimento. Informações sobre o conjunto de dados utilizado, e a justificativa de cada escolha são expostos, com o objetivo de fornecer um melhor entendimento do cenário e posteriormente melhor interpretação dos resultados.

### 3.1 Base de Dados

O Centro de Pesquisa, Inovação e Desenvolvimento (CPID) é um instituto de pesquisa localizado no sudeste brasileiro, e tem como objetivo fomentar pesquisas e projetos de desenvolvimento tecnológico visando a implantação de ações estratégicas para o desenvolvimento do Estado do Espírito Santo (SECTIDES, 2020). O centro possui uma usina experimental de geração de energia alternativa com uma potência instalada de  $264kWp$  que gera em média  $35MWh/mês$ , sendo essa energia utilizada no empreendimento. A Figura 15 apresenta a estrutura física da usina solar.



Fonte: (SECTIDES, 2020).

Figura 15 – Usina experimental de geração de energia alternativa - CPID.

A usina teve seu funcionamento iniciado em janeiro de 2021, no entanto os dados só

começaram a ser armazenados, pelo sistema *SolarEdge Monitoring Plataforma*, em fevereiro de 2021. As medições de potência, corrente e tensão são efetuadas por sensores, presentes no sistema de painéis fotovoltaicos, são dispositivos MLPE denominados otimizadores de potência P730/P850, e sua função é monitorar de forma contínua a energia gerada em cada módulo, buscando seu respectivo ponto de máxima potência (LEITE, 2020).

A usina fotovoltaica é composta por 840 módulos, e há um otimizador de potência para cada dois módulos do painel. Os dados utilizados nesse estudo são medições de potência de cada otimizador, e foram coletados no período entre 12/02/2021 e 20/06/2021. O intervalo de tempo considerado no dia foi de 07h às 17h, considerando o fato que entre os horários de 05h às 07h e após as 17h as medições apresentavam valores muito próximos de zero ou iguais a zero.

O sistema de monitoramento *SolarEdge Monitoring Plataforma* possui um sistema de alerta de falha, que abrange algumas falhas relacionadas a comunicação dos sensores com o sistema e de incompatibilidade de tensão dos módulos. No espaço de tempo considerado para coleta dos dados, o sistema identificou uma falha que teve início no dia 17/03/2021 às 13h e foi reparada no dia 05/04/2021 às 10h. Essa informação foi utilizada como classificador do conjunto de dados, para testes com modelos de classificação. Isso foi pensado considerando que há interesse não somente em desenvolver um modelo de previsão de potência para um painel fotovoltaico, mas também a partir dos dados de previsão conseguir determinar se um sistema fotovoltaico apresenta ou não indicação de falha. Os testes iniciais foram realizados considerando apenas duas classificações: Normal e Falha.

## 3.2 Informações Computacionais

As informações de *software* e *hardware* dos dispositivos utilizados no desenvolvimento do estudo são apresentadas nas Tabela 1 e 2, respectivamente. No total, dois dispositivos físicos foram utilizados, aferido que o modelo de previsão Rede Neural Espaço Temporal de Grafos (STGCN) necessitava de grande capacidade de processamento computacional para ser experimentado. Além dos dispositivos físicos, o ambiente virtual Google *Collaboratory* (Colab) foi usado, para compartilhamento dos resultados dos experimentos.

Tabela 1 – Dispositivos Eletrônicos: *Software*

Dispositivos Eletrônicos	Sistema operacional/suporte	Linguagem de programação	Compilador
Acer Aspire A515-54G	WINDOWS 11	Python	Pycharm
A+ Server 2124GQ-NART	LINUX-UBUNTU	Python	Python

Tabela 2 – Dispositivos Eletrônicos: *Hardware*

Dispositivos Eletrônicos	CPU	GPU	RAM
Acer Aspire A515-54G	Intel(R) Core(TM) i5-10210U	NVIDIA GeForce MX250	8 G
A+ Server 2124GQ-NART	AMD EPYC	NVIDIA HGX A100 4-GPU	8 T
Google <i>Collaboratory</i>	-	RTX 2080 Ti	32 G

O ambiente virtual Colab permite escrita e execução de código python arbitrário, e funciona como um serviço de notebook jupyter, com *hardware* com alto poder de processamento que pode ser utilizado por um tempo limitado (BISONG, 2019). Além disso, o colab possibilita o compartilhamento de documentos entre pessoas sem precisar baixar, instalar ou executar nada. O tempo de limitação de uso do colab, tornou inviável o uso para os experimentos do modelo SARIMA e STGCN, mas todos os outros modelos foram testados na plataforma.

O CPID disponibilizou acesso à máquina supermicro do laboratório (A+ Server 2124GQ-NART), como alternativa para o problema de processamento encontrado na experimentação dos modelos SARIMA e STGCN. Assim como exibido na Tabela 2, a linguagem utilizada na investigação dos modelos propostos foi a linguagem de programação Python (ROSSUM; DRAKE, 2009).

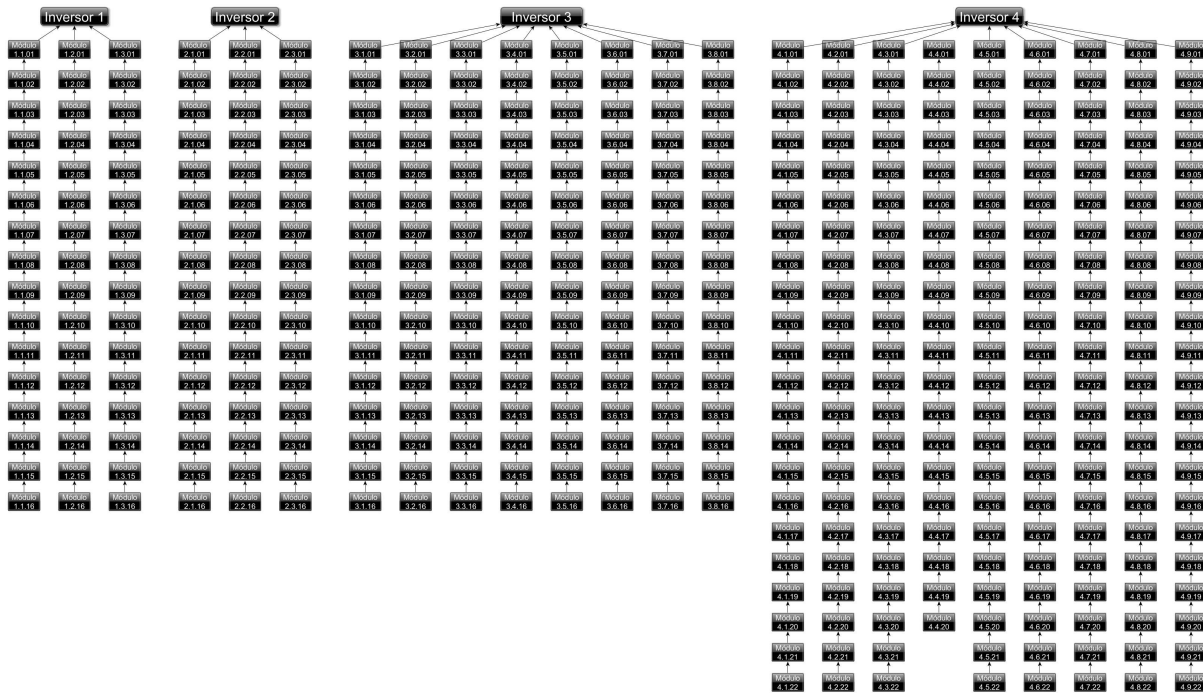
A manipulação nos dados de série temporal foi executada utilizando as bibliotecas Pandas (TEAM, 2020), Numpy (HARRIS et al., 2020), Statsmodels (SEABOLD; PERKTOLD, 2010) e Keras (CHOLLET et al., 2015). As estruturas de grafo foram manuseadas com as bibliotecas Networkx (HAGBERG; SCHULT; SWART, 2008) e Stellargraph (DATA61, 2018). É importante destacar que as bibliotecas Statsmodels, Keras e Stellargraph são responsáveis pela estruturação dos modelos SARIMA, STGCN e Rede Neural Temporal Convolutacional de Grafo (TGCN). As apresentações gráficas dos dados foram realizadas utilizando a biblioteca Matplotlib (HUNTER, 2007).

### 3.3 Modelagem dos dados

A representação lógica do sistema fotovoltaico do CPID foi desenhada na aplicação yEd (yWorks, 2019), e é apresentada na Figura 16. A aplicação yEd, além de facilitar a visualização do grafo, permite salvar o arquivo na extensão *Trivial Graph Format* (TGF), facilitando o *input* do grafo rotulado no algoritmo do estudo. O grafo tem um total de 420 vértices e 397 arestas, sendo que cada vértice corresponde a um otimizador de potência do painel fotovoltaico e as arestas as conexões elétricas.

Os dados de potência do painel são atributos dos vértices do grafo, e estão organizados no formato de séries temporais, sendo um total de 420 séries, cada série referente a um otimizador. As séries temporais abrangem um total de 129 dias, sendo a informação

de potência agregada a nível de hora, totalizando 1419 observações. De forma a facilitar a manipulação computacional do dados, as séries temporais foram estruturadas em uma matriz de atributos de dimensão  $1419 \times 420$ , sendo cada coluna da matriz a série temporal referente a um módulo.



Fonte: Autor.

Figura 16 – Relação lógica do sistema fotovoltaico do CPID.

A formatação dos dados de entrada do algoritmo de processamento é definida conforme o modelo a ser testado. No geral, os algoritmos de processamento trabalham com duas formatações, um formato para os modelos estatísticos e um formato para os modelos de aprendizagem profundo de máquina. Os modelos estatísticos trabalham com as séries uma a uma armazenadas na matriz de atributos do grafo, ou seja, os dados são utilizado no formato de vetores. Sob outro ponto de vista, os modelos de redes neurais utilizam os dados de entrada no formato tensorial. A informação da topologia do grafo, é inserida pela matriz de adjacência no algoritmo, sendo a matriz de adjacência gerada a partir do arquivo de extensão TGF utilizando a biblioteca Python *stellargraph*. Enquanto que, a informação da matriz de atributo é inserida na forma de uma tensor tridimensional, sendo a terceira dimensão gerada através de manipulação na forma da matriz de atributos para identificar a janela temporal de agregação diária dos dados. Os dados da matriz de atributos estão agregadas a nível de hora, e o intervalo de tempo considerado no dia foi de 07h às 17h. Isso significa que a terceira dimensão do tensor da matriz de atributos será de tamanho 11.

## 3.4 Implementação das Soluções

O levantamento bibliográfico foi realizado pensando em metodologias que permitissem a detecção de falha em um painel fotovoltaico em tempo real, utilizando os dados apresentados na Seção 3.1. A aplicação de técnicas de aprendizado de máquina utilizando modelagem de grafos em dados de sistemas de painéis fotovoltaicos é uma área pouco explorada, como descrito na Seção 1.2, e na maioria dos estudos são utilizados dados simulados para a extração das características pertinentes às falhas de interesse. Além da simulação dos dados, alguns estudos utilizaram taxa de amostragem impraticável para uma avaliação em tempo real, considerando que os dados históricos precisam ser armazenados em uma estrutura de banco de dados.

A solução encontrada para o problema de definição das falhas relevantes foi gerar um modelo de previsão de potência, de modo que as variações de potência identificadas pelo modelo de previsão, posteriormente, fossem utilizadas como característica de falhas. Dessa maneira, o estudo foi separado em duas partes: construção do modelo de previsão e construção do modelo de classificação. A construção do modelo de previsão para os dados de potência do sistema fotovoltaico foi elaborada utilizando três metodologias diferentes: SARIMA, STGCN e TGCN. O modelo de classificação, em contrapartida, em função do escopo do trabalho, foi construído utilizando as informações de falhas relacionadas à comunicação dos sensores com o sistema e de incompatibilidade de tensão dos módulos, sendo sua metodologia a aplicação de uma rede neural *perceptron* multicamadas ao conjunto de dados.

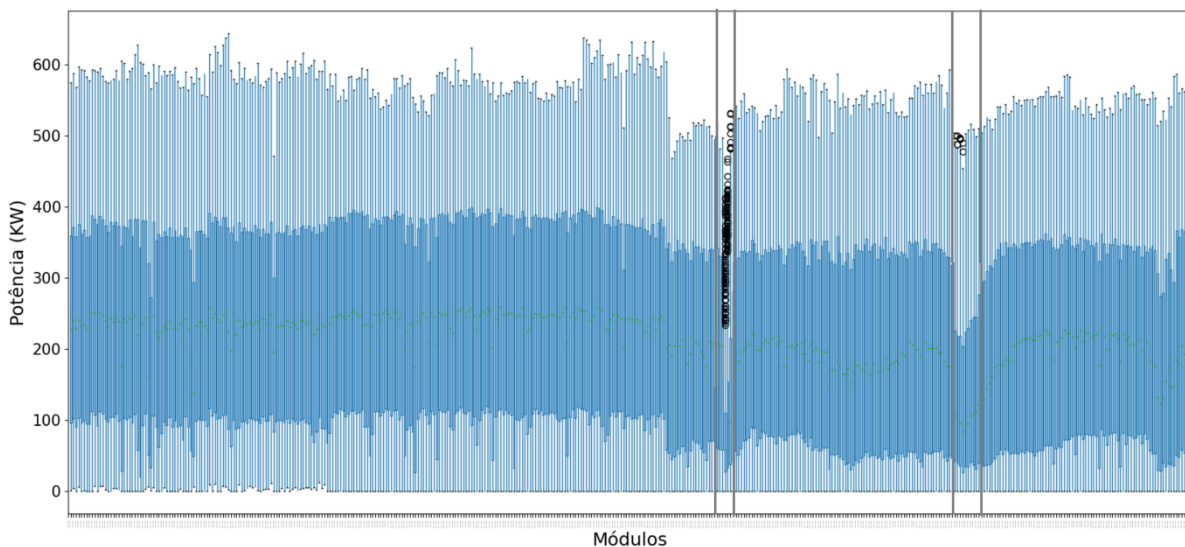
### 3.4.1 Análise Exploratória dos Dados

A análise exploratória de dados é uma das etapas mais onerosas e importantes em um projeto de modelagem de dados (WEI, 2006). Essa etapa é necessária para entender o domínio analisado e o comportamento aparente dos dados. Uma análise geral inicial foi realizada utilizando os gráficos boxplots das séries temporais dos módulos e o resultado é apresentado na Figura 17. É possível notar que, embora a maioria dos módulos tenham um comportamento similar de geração de energia, existem módulos que possuem um comportamento conflitante, como os casos destacados em cinza.

É possível notar que a potência gerada pelos módulos destacados em cinza na Figura 17, foi muito menor que a dos outros módulos. A distribuição da geração de potência desses módulos no período de tempo considerado, ficou entre  $50KW - 200KW$ , enquanto que a do restante dos módulos, ficou entre  $100KW - 350KW$ . Tornando a geração de potência comum dos módulos um valor outlier para os módulos em destaque.

Um estudo examinando os módulos de forma individual foi realizado, tendo como objetivo identificar as principais características das séries e possíveis causas para o mal

desempenho identificado. A Figura 21, explicada de maneira mais detalhada na Seção 4, apresenta o gráfico gerado da série original de potência de um módulo do sistema fotovoltaico. Apesar do resultado de apenas um módulo ser apresentado, a mesma lógica foi aplicada as outras séries, e foi observado que a maioria das séries possuía quebra nas médias, indicando um comportamento de não estacionariedade da variância e autocovariância da série.

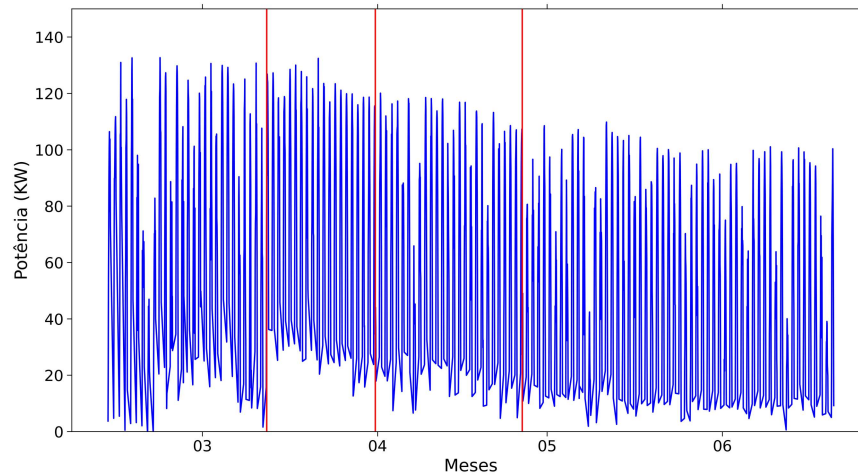


Fonte: Autor.

Figura 17 – Gráfico Boxplot da Geração de Energia dos Módulos do Sistema

O teste CUSUM (BROWN; DURBIN; EVANS, 1975) foi aplicado para validar se de fato as séries eram não estacionárias e se constatou que a série de todos os módulos possuía mudança de níveis com 95% de confiança. No módulo analisado na Figura 21 o valor- $p$  obtido para o teste foi igual a  $5,132 \times 10^{-9}$ . A transformação Box-Cox foi aplicada em cada série e os pontos de quebra de média foram identificados no conjunto de dados transformados, como exibido na Figura 18.





**Fonte:** Autor.

Figura 18 – Transformação Box-Cox

As quebras de média foram identificadas nos dias 12, 31 de março e 26 de abril de 2021. Houve um crescimento na energia gerada na primeira quebra, que se manteve durante o período entre a primeira e segunda quebra, que pode ser atribuída à maior incidência de radiação solar que ocorre no fim do verão. A segunda quebra foi um decréscimo da energia gerada, também podendo ser explicada pela mudança de estação, que ocorre dia 21/03/2021, data que marca o encerramento do verão e início do outono no Brasil. No entanto, é preciso lembrar que, essa análise foi realizada pontualmente para um único módulo. A comprovação das suposições a cerca da mudança de níveis do conjunto de dados necessita de uma análise mais crítica, considerando um período maior de observação dos dados e a identificação das diferentes estações dentro do modelo ajustado.

Um dos modelos testados nesse estudo é o modelo estatístico de séries temporais, por esse motivo, é necessário que a série esteja em conformidade com algumas premissas. Os testes ADF e KPSS foram realizados para testar a estacionariedade dos dados, sendo a hipótese nula de ambos os testes rejeitada, tornou-se indispensável utilizar a diferenciação dos dados no modelo de ajuste estatístico.

É necessário destacar que os dados processados com a transformação Box-Cox foram utilizados como dados de entrada para o modelo estatístico SARIMA e o modelo STGCN.

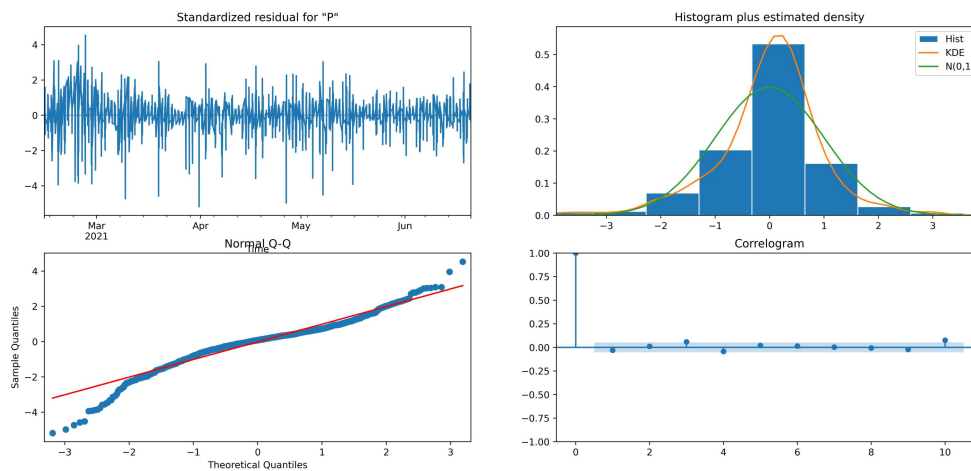
### 3.4.2 Modelo Estatístico - SARIMA

Ajustes dos modelos  $SARIMA(p, d, q) \times (P, D, Q)_{11}$ , com  $0 \leq p \leq q \leq d \leq P \leq Q \leq D \leq 2$  foram realizados. A decisão a favor do melhor modelo foi baseada

no Critério de *AKAIKE* (AIC) e o Critério de Informação Bayesiano (BIC). O modelo  $SARIMA(1, 1, 1) \times (1, 0, 1)_{11}$  apresentou o menor valor em ambos os critérios, sendo então definido para o ajuste dos dados.

Os resíduos do modelo ajustado foram calculados para o módulo apresentado na Figura 21 e os gráficos de diagnóstico são apresentados na Figura 19. O diagnóstico do modelo indica que a suposição que os resíduos são não correlacionados e seguem uma distribuição normal padrão com média 0 e variância  $\sigma^2$  é verdadeira. A fim de confirmar os resultados encontrados nos gráficos de diagnóstico, avaliaremos se os resíduos do modelo ajustado atendem as hipóteses de ruído branco e normalidade, aplicando o teste de normalidade de *Jarque-Bera* e o teste de autocorrelação de *Ljung-Box* com nível de significância  $\alpha = 0,05$ .

O teste de *Ljung-Box*, com valor- $p$  igual a 0,301335, não rejeita a hipótese de ruído branco. No entanto, o teste de *Jarque-Bera* resultou em um valor- $p \leq 0$ , rejeitando a hipótese de normalidade dos resíduos. Nesse cenário, a previsão dos valores foi realizada utilizando o método de reamostragem *sieve bootstrap*.



Fonte: Autor.

Figura 19 – Gráficos diagnósticos dos resíduos do modelo

### 3.4.3 Rede Neural Espaço Temporal de Grafos - STGCN

Além do modelo estatístico, um modelo de rede neural espaço temporal de grafos foi testado antes do modelo TGCN. A arquitetura base da rede foi construída conforme modelo desenvolvido para previsão de tráfego em rodovias (YU; YIN; ZHU, 2017), em consequência da similaridade das séries presentes na matriz de atributos dos vértices. A configuração dos blocos convolucionais espaço-temporal são apresentadas na Tabela 3.

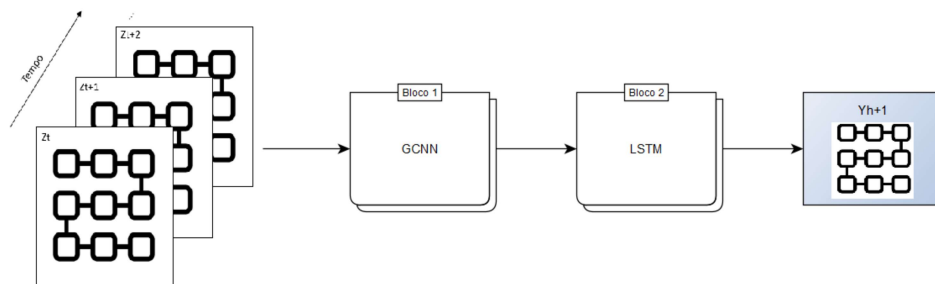
Tabela 3 – Blocos espaço-temporal

Blocos	Configuração dos canais
1	[1, 32, 64]
2	[64, 32, 128]

O treinamento foi executado com número de épocas e tamanho do lote igual a 100 e 50, respectivamente. A taxa inicial de treinamento definida foi  $10^{-3}$  com taxa de decaimento igual 0,7 a cada 5 épocas.

### 3.5 Estrutura do Modelo

A estrutura do modelo de aprendizado profundo TGCN (ZHAO et al., 2020), adaptada para previsão de potência em sistemas de painéis fotovoltaicos, é representada na Figura 20.



Fonte: Autor.

Figura 20 – Estrutura do Modelo TGCN

O modelo TGCN consiste em quatro componentes principais:

1. Os dados de entrada: A representação topológica dos módulos em conjunto com as medições de potência do sistema fotovoltaico.
2. A operação de convolução de grafo: Bloco responsável por aprender a correlação espacial do vértice central e seus vizinhos.
3. Camadas LSTM: Bloco responsável pela captura da dependência temporal do estado de potência do vértice e seus vizinhos.
4. Camada de Saída da rede: Responsável por fornecer as previsões de potência um passo a frente do período estudado.

Detalhes sobre o funcionamento dos componentes 2 e 3 são descritos no Capítulo 2. A arquitetura final é definida através de uma avaliação de desempenho da rede como descrito na Seção 3.6.

### 3.6 Metodologia de Avaliação do Desempenho da Rede Neural

O conjunto de dados foi dividido em três subconjuntos: treino, validação e teste. A rede é treinada no conjunto de treino e avaliada no conjunto de validação. Definido o modelo, é realizado um exame final do seu desempenho no conjunto de dados de teste.

A estrutura do modelo de rede neural é definida através de ensaios, que geralmente são iniciados com uma estrutura de rede neural utilizada em estudos que tem proposta de estudo similar ao proposto. A estrutura de ensaio inicial utilizada, para o modelo de previsão, foi a desenvolvida no estudo de tráfego de carros (YU; YIN; ZHU, 2017; ZHAO et al., 2020). O modelo de classificação teve sua estrutura construída com base em um otimizador bayesiano (CHOLLET et al., 2015), que auxiliou tanto na escolha da quantidade de camadas e neurônios de cada camada da rede, como na seleção das funções de ativação de cada camada, e na definição dos melhores parâmetros e hiperparâmetros para treinamento do modelo.

Desenvolver um modelo de rede neural sempre envolve afinação (*tuning*) na configuração. A afinação do modelo consiste na escolha dos parâmetros e hiperparâmetros, e é realizada utilizando o sinal de *feedback* do desempenho do algoritmo nos dados de validação. Em resumo, a afinação é uma pesquisa para uma boa configuração do modelo. No entanto, o processo de afinação com base no desempenho do modelo no conjunto de dados de validação pode resultar em um sobreajuste nesse conjunto, mesmo que o modelo não esteja sendo treinado diretamente nele. O centro desse fenômeno é a noção de vazamento de informação. A fim de garantir o bom desempenho do modelo em dados nunca vistos pelo modelo, após definidos os parâmetros, é realizada uma avaliação em um conjunto totalmente desconhecido para o modelo: o conjunto teste.

Garantir que não haja sobreposição desses conjuntos é essencial, visto que, o algoritmo não pode ter acesso a nenhuma informação do conjunto teste, mesmo que indiretamente. O particionamento do conjunto pode ser realizado de várias formas. Nesse estudo foi utilizado o particionamento simples, preservando as informações sequenciais dos dados.

### 3.7 Métricas de desempenho dos modelos

Métricas de desempenho são medidas que determinam de forma quantitativa o rendimento assertivo dos modelos, sendo utilizadas como parâmetro de comparação para

definir o modelo com melhor desempenho. As medidas geralmente utilizadas em modelos de previsão de valores futuros são o erro médio absoluto (MAE) e a raiz do erro quadrático médio (RMSE) dada pelas equações (3.1) e (3.2), onde  $\hat{y}_j$  o valor estimado do modelo.

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j| \quad (3.1)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2} \quad (3.2)$$

Modelos de classificação também podem ser avaliados segundo as métricas apresentadas pelas equações (3.1) e (3.2). No entanto, existem mais algumas métricas que fornecem informações interessantes para interpretação dos resultados. É importante ressaltar que essas métricas são mais apropriadas para problemas que manipulam dados desbalanceados. As métricas são baseadas na matriz de confusão do modelo, ou seja, utilizam os valores de verdadeiro positivo ( $tp$ ), verdadeiro negativo ( $tn$ ), falso positivo ( $fp$ ) e falso negativo ( $fn$ ).

A medida precisão mede numericamente a fração de instâncias verdadeiras que são relevantes e é definida pela equação 3.3. A medida revocação é compreendida pela fração de instância relevantes que são verdadeiras e é dada pela equação 3.4

$$Pr = \frac{tp}{(tp + fp)} \quad (3.3)$$

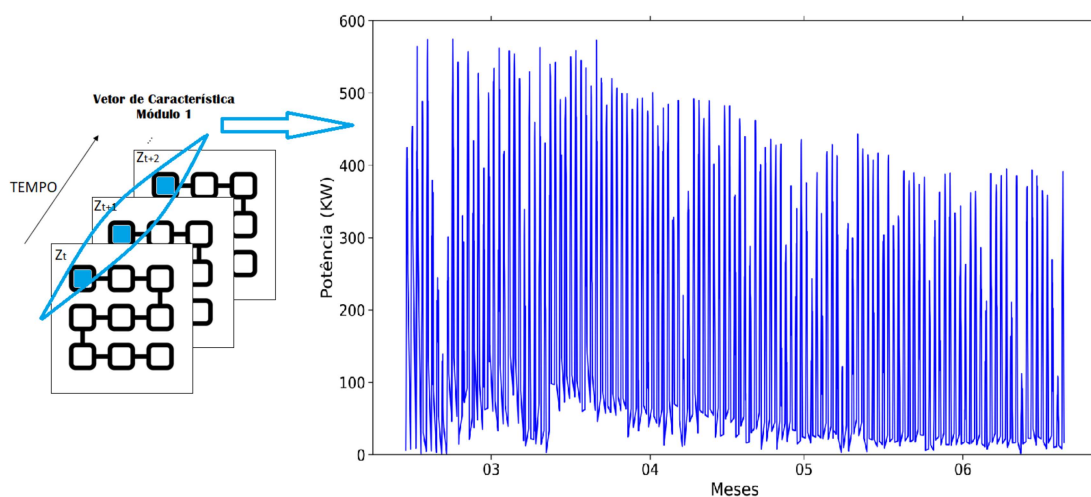
$$revoc = \frac{tp}{(tp + f)} \quad (3.4)$$



## 4 RESULTADOS E DISCUSSÃO

Nesta parte da dissertação, apresentam-se os resultados obtidos com os experimentos descrito no Capítulo 3. A metodologia foi desenvolvida utilizando a linguagem Python (ROSSUM; JR, 1995). Os pacotes *keras* (CHOLLET et al., 2015), *scipy* (VIRTANEN et al., 2020), *scikit-learn* (PEDREGOSA et al., 2011) auxiliaram na construção dos modelos propostos, como descrito na Seção 3.5.

A estrutura de grafo da topologia lógica da usina fotovoltaica contém 420 vértices, correspondente aos 420 otimizadores do sistema, e 397 arestas. A matriz de atributos dos vértices possui 1419 observações, sendo as informações agregadas por hora, referente a um período de 129 dias (12/02/2021 a 20/06/2021). A representação da estrutura de grafo utilizada e os dados de potência, armazenados na matriz de atributos, referente a um módulo do sistema fotovoltaico são apresentados na Figura 21. O gráfico apresentado na Figura 21 exibe a série original de potência de um módulo, sendo possível identificar possíveis diferenças de médias no gráfico. No início do mês de março houve um crescimento da potência gerada, que pode ser atribuída à maior incidência de radiação solar que ocorre no final do verão. O decréscimo subsequente pode ser explicado pela mudança de estação verão/outono, que ocorreu no Brasil dia 21/03/2021. É necessário uma análise mais crítica do dados, considerando um período maior de tempo, para confirmar o prognóstico dado.



Fonte: Autor.

Figura 21 – Representação da estrutura de grafo da topologia lógica da usina experimental de geração de energia alternativa - CPID. Cada  $Z_t$  indica a imagem das medições de potência no instante  $t$ , que é armazenado em uma matriz de atributos de nós. A série temporal apresenta a série  $z_v$  de um módulo do sistema fotovoltaico.

O conjunto de dados foi submetido a normalização descrita na equação 2.11 e dividido nos subconjuntos de treino, validação e teste, cada um possuindo 90, 19 e 20 dias, respectivamente. O interesse do experimento é realizar a previsão de potência no decorrer do dia. Para fins de comparação com resultados de outros estudos (VIDIGAL et al., 2022), foi fixada a previsão no horário das 16h, utilizando a previsão um passo a frente. A arquitetura base do algoritmo de aprendizagem foi construída conforme modelo T-GCN (ZHAO et al., 2020). Manipulações da estrutura do modelo foram realizadas com o objetivo de encontrar um algoritmo parcimonioso que obtivesse resultados ótimos da previsão de valores. A arquitetura do algoritmo T-GCN que apresentou melhor resultado é apresentada na Tabela 4.

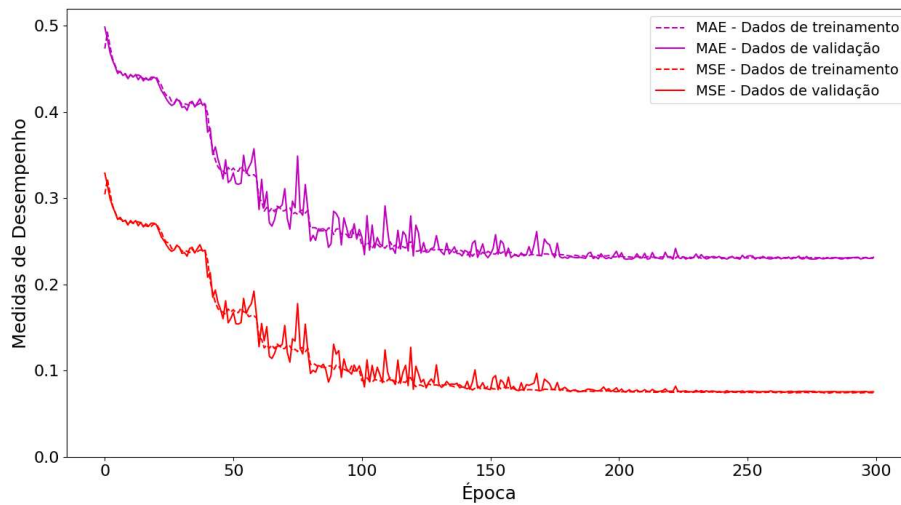
Tabela 4 – Arquitetura T-GCN

Camada	Tipo	Mapa de Característica	Dropout
0	Dados	$1 \times 420 \times 11$	–
1	Grafo Convolutacional	$1 \times 420 \times 16$	0.1
2	Grafo Convolutacional	$1 \times 420 \times 32$	0.1
3	LSTM	$1 \times 32 \times 200$	–
4	LSTM	$1 \times 200$	–
5	Camada Densa	$1 \times 420$	–

A arquitetura final do modelo T-GCN, apresentado na Tabela 4, consiste em um modelo de 2 camadas grafo convolutacional, 2 camadas LSTM e 1 camada densa. Embora a arquitetura tenha sido baseada no modelo T-GCN (ZHAO et al., 2020), os parâmetros do modelo, como quantidade de neurônios das camadas e funções de ativação, foram selecionados baseados no modelo STGCN (YU; YIN; ZHU, 2017). O mapa de característica indica a forma dos tensores de entrada de cada camada, sendo a estrutura modificada através da função *reshape*.

O treinamento do algoritmo foi realizado durante 300 iterações, também conhecidas como épocas, manipulando lotes de dados de tamanho 50. O valor do erro médio absoluto e do erro quadrático médio foi calculado em cada iteração considerando os subconjuntos de treino e validação, os resultados são apresentados na Figura 22. A taxa inicial de aprendizagem utilizada foi  $10^{-1}$ , com taxa de decaimento de 0,7 a cada 8 épocas.





Fonte: Autor.

Figura 22 – Treinamento do modelo proposto na Tabela 4.

Analisando o resultado apresentado na Figura 22, é possível observar que o erro absoluto médio e o erro quadrático médio decrescem à medida que o modelo é treinado. Além disso, a medida de desempenho avaliada no conjunto de treinamento tem um decréscimo mais acentuado do que no conjunto de validação, isso ocorre devido ao sobreajuste da rede, ou seja, a rede costuma ser muito melhor para prever os dados de treinamento, devido ao fato que seu conhecimento é adquirido utilizando esses dados. O valor de iterações foi limitado a 300, visto que as medidas de desempenho após essa quantidade de iterações não apresentavam diferença significativa, indicando que houve uma estagnação no aprendizado da rede.

A análise final dos algoritmos consiste em comparar o desempenho de cada modelo definido, realizando o teste em uma mesma base de dados, utilizando as métricas de desempenho como parâmetro de escolha do melhor modelo. Lembrando que, as métricas de desempenho medem a qualidade do ajuste do modelo de modo quantitativo, variando seus valores entre 0 (melhor desempenho) e 1 (pior desempenho).

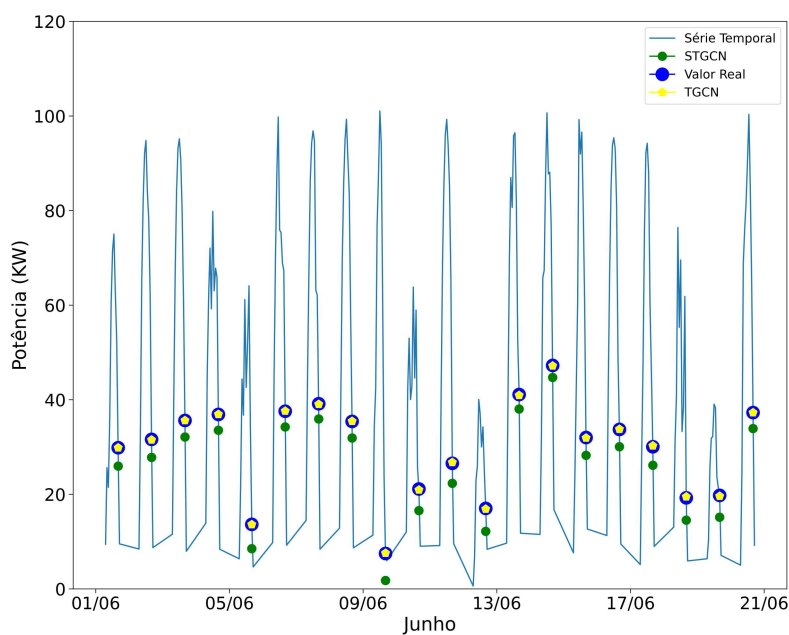
A configuração do modelo com menor erro absoluto médio foi armazenada, e o modelo foi utilizado para calcular as previsões no conjunto teste. Os resultados das medidas de desempenho de interesse e das previsões são apresentadas na Tabela 5 e na Figura 23, respectivamente. Para fins de comparação, além dos valores do modelo de rede neural desenvolvido, a Tabela 5 apresenta os resultados de previsão do conjunto teste de outros modelos desenvolvidos (VIDIGAL et al., 2022), sendo o modelo SARIMA um modelo estatístico de série temporal e o STGCN um modelo espaço temporal convolucional de grafo.

O modelo T-GCN, embora tenha apresentado o tempo de processamento no treinamento da rede cerca de 34 vezes maior que o modelo mais rápido, demonstrou melhor desempenho quando comparado aos outros modelos testados. O erro quadrático médio e o erro médio absoluto do modelo T-GCN desenvolvido, foi reduzido cerca de 97% e 96%, respectivamente, quando comparado ao valor do modelo mais rápido. Logo, dado a grande melhora na performance do modelo de previsão é justificável a escolha do T-GCN para a tarefa de previsão dos valores de potência de um sistema fotovoltaico.

Tabela 5 – Desempenho dos modelos

Modelos	MAE	RMSE	Tempo
SARIMA	17,258	15,689	4832,788s
STGCN(Cheb)	4,198	4,386	23,085s
STGCN(1st)	4,189	4,349	<b>21,994s</b>
T-GCN	<b>0,128</b>	<b>0,182</b>	757,785s

O gráfico comparativo dos valores preditos pelos modelos STGCN(1st), T-GCN e os valores reais é apresentado na Figura 23, e reforça as deduções realizadas analisando a Tabela 5. A acurácia da predição melhorou de forma que, graficamente, o valor real e o valor predito pelo modelo T-GCN são próximos a ponto de suas representações se sobreporem.



Fonte: Autor.

Figura 23 – Comparação dos valores reais *versus* previsões um passo à frente da série temporal.

Além do modelo de previsão utilizando aprendizado profundo grafo temporal, a matriz de atributos normalizada foi submetida a um modelo inicial de classificação. A arquitetura base do classificador é composta por 4 camadas densas, que são descritas na Tabela 6.

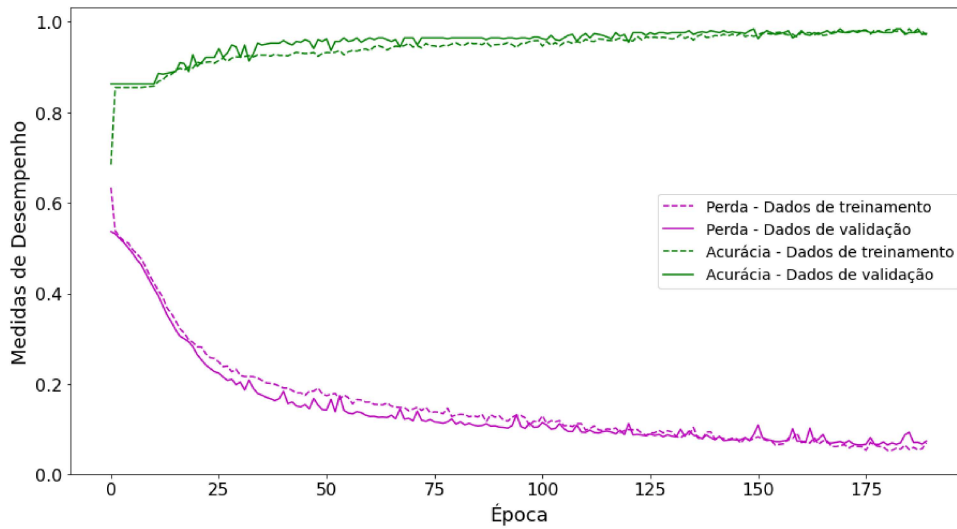
Tabela 6 – Arquitetura Modelo de Classificação

Camada	Tipo	Mapa de Característica	Dropout
0	Dados	$1 \times 420$	–
1	Camada Densa	$1 \times 420$	–
2	Camada Densa	$1 \times 128$	0.1
3	Camada Densa	$1 \times 128$	0.1
4	Camada Densa	$1 \times 2$	–

Os dados de entrada do modelo de classificação foi aleatorizado, perdendo assim a lógica temporal. Essa manipulação foi realizada considerando que para o conjunto de dados organizado em uma sequência temporal, o modelo de classificação não tinha desempenho relevante ou satisfatório. Um dos possíveis motivos para esse comportamento do modelo é o momento e o curto período de tempo em que foi identificado a falha, só sendo possível assim levar essa característica para o conjunto de treino do modelo.

O treinamento do modelo foi realizado utilizando um algoritmo de otimização bayesiana. Inicialmente o treino foi realizado considerando quatro camadas densas, variando a quantidade de neurônios de cada camada no intervalo de 16 a 128, com incrementos de tamanho 16. Três funções de ativação foram avaliadas, sendo elas as funções *Relu*, *Tanh* e *Sigmoid*. A taxa de *dropout* do modelo foi investigada para valores de 0,1 a 0,5 e a taxa de aprendizagem no intervalo entre  $1 \times 10^{-2}$  e  $1,5 \times 10^{-4}$ . A arquitetura apresentada na Tabela 6 apresentou o melhor desempenho entre as estruturas testadas, utilizando a função de ativação *Relu* e a taxa de aprendizagem igual  $1,5 \times 10^{-4}$ . O aprendizado do algoritmo foi avaliado durante 180 iterações, manipulando lotes de tamanho 50. O valor de perda, medida apresentada como o erro quadrático médio nesse projeto, juntamente com a acurácia do modelo foram calculados em cada iteração considerando os subconjuntos de treino e validação, e o resultados são apresentados na Figura 24.

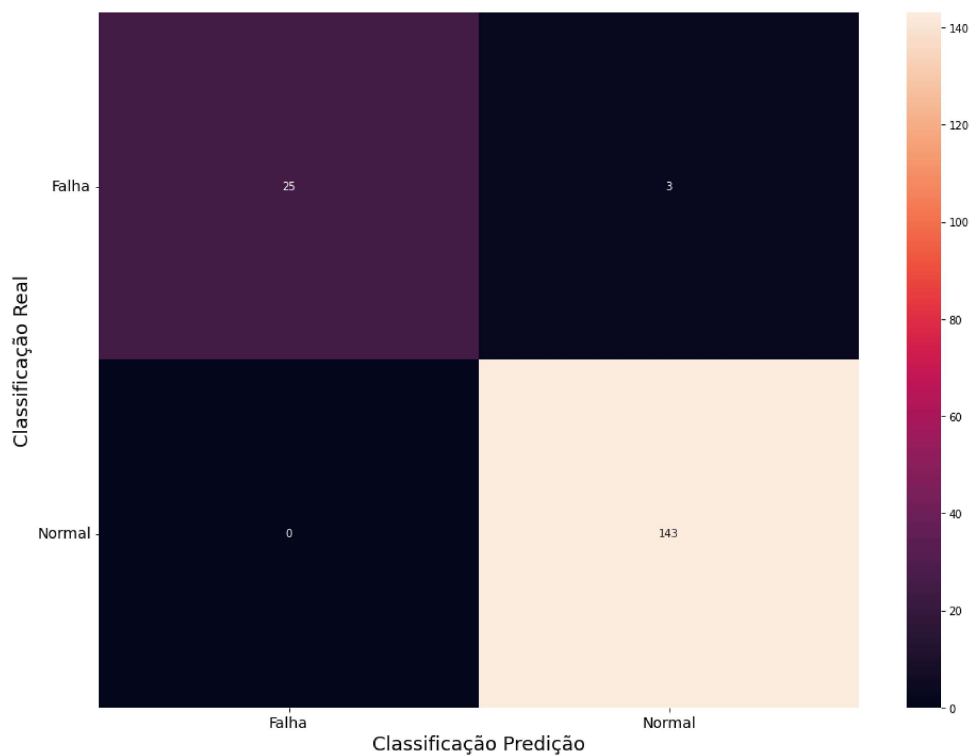
Os resultado apresentados na Figura 24 mostram que o valor da perda decresce significativamente antes da iteração 100, assumindo um comportamento mais brando nas ultimas iterações. É possível observar também que os valores das métricas para o subconjuntos de treino e validação são bem próximos, com os dados de treino aparentando ter um acréscimo mais acentuado, caracterizando um possível sobreajuste da rede, que é reparado com a definição da taxa de *dropout* nas camadas ocultas do modelo.



Fonte: Autor.

Figura 24 – Treinamento do modelo proposto na Tabela 6

A matriz de confusão do modelo aplicado ao conjunto teste, é construída e apresentada na Figura 25. A diagonal da matriz apresenta a quantidade de objetos que foram classificados corretamente, enquanto que, valores fora da diagonal foram classificados de maneira equivocada.



Fonte: Autor.

Figura 25 – Matriz de Confusão

Buscando entender melhor os resultados dipostos na matriz de confusão, os cálculos das métricas de precisão e revocação foram executados e os valores de 0,9767 e 0,9824 foram encontrados, respectivamente. Os valores de revocação e precisão são considerados altos quando próximos de um. Isso significa que o classificador está sendo exigente ao realizar a categorização dos dados.



## 5 CONCLUSÕES E TRABALHOS FUTUROS

O monitoramento de dados, em uma usina solar, auxilia na redução de perda de energia do sistema e na redução do custo de manutenção. Visando encontrar um modelo de previsão ótimo para um sistema de painéis fotovoltaicos, uma técnica utilizando um modelo híbrido de rede neural convolucional de grafo e redes neurais recorrentes foi proposto com o objetivo de prever a potência gerada do sistema.

A utilização de um modelo híbrido de convolução de grafos e redes neurais recorrentes LSTM apresentou melhor desempenho quando comparado a outros modelos de previsão de valores de potência de sistemas fotovoltaicos. A estrutura da rede neural T-GCN, embora mais complexa e com desempenho computacional mais dispendioso, efetuou previsões mais precisas que os demais modelos. Além disso, é possível concluir que a estrutura de grafo gera um melhor aprendizado para a rede neural, considerando que o modelo de redes neurais recorrentes LSTM apresentou um resultado inferior ao modelo híbrido T-GCN.

A análise do comportamento de dados é essencial na análise de séries temporais. Um conjunto de dados maior, contendo mais informações da potência ao longo do tempo, poderia trazer muito mais riqueza de informação ao estudo. Além disso, é interessante realizar testes alterando o horário de previsão do modelo, uma vez que realizar somente a previsão da última hora do dia não é uma análise interessante para o administrador do sistema. Explorar variáveis que possam influenciar as medições de potência como temperatura, estação do ano, entre outras, é uma abordagem que pode garantir a generalização e aplicação do modelo em qualquer época do ano.





# REFERÊNCIAS

- ABDULMAWJOOD, K.; REFAAT, S. S.; MORSI, W. G. Detection and prediction of faults in photovoltaic arrays: A review. In: IEEE. *2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018)*. [S.l.], 2018. p. 1–8. Citado na página 1.
- ALAM, M. K. et al. A comprehensive review of catastrophic faults in pv arrays: types, detection, and mitigation techniques. *IEEE Journal of Photovoltaics*, IEEE, v. 5, n. 3, p. 982–997, 2015. Citado na página 3.
- ALI, M. H. et al. Real time fault detection in photovoltaic systems. *Energy Procedia*, Elsevier, v. 111, p. 914–923, 2017. Citado na página 4.
- BASNET, B.; CHUN, H.; BANG, J. An intelligent fault detection model for fault detection in photovoltaic systems. *Journal of Sensors*, Hindawi, v. 2020, 2020. Citado na página 4.
- BENJAMIN, A.; CHARTRAND, G.; ZHANG, P. *The fascinating world of graph theory*. [S.l.]: Princeton University Press, 2015. Citado na página 10.
- BERTICELLI, R. et al. Energia solar fotovoltaica: Conceitos e aplicações. *I Seminário de Pesquisa Científica e Tecnológica*, v. 1, n. 1, 2017. Citado 2 vezes nas páginas 7 e 8.
- BISONG, E. Google colab. In: \_\_\_\_\_. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Berkeley, CA: Apress, 2019. p. 59–64. ISBN 978-1-4842-4470-8. Disponível em: <[https://doi.org/10.1007/978-1-4842-4470-8\\_7](https://doi.org/10.1007/978-1-4842-4470-8_7)>. Citado na página 31.
- BONDY, J. A.; MURTY, U. S. R. et al. *Graph theory with applications*. [S.l.]: Macmillan London, 1976. v. 290. Citado na página 10.
- BOX, G. E.; COX, D. R. An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, Wiley Online Library, v. 26, n. 2, p. 211–243, 1964. Citado na página 14.
- BRASIL, E. *ON-GRID OU OFF-GRID: AFINAL, QUAL O MELHOR SISTEMA SOLAR PARA A SUA CASA?* 2021. <<https://www.energybrasilsolar.com.br/sistema-solar-on-grid-off-grid>> [Online; acessado em 02-Junho-2021]. Citado na página 1.
- BROWN, R. L.; DURBIN, J.; EVANS, J. M. Techniques for testing the constancy of regression relationships over time. *Journal of the Royal Statistical Society. Series B (Methodological)*, JSTOR, p. 149–192, 1975. Citado 2 vezes nas páginas 14 e 34.
- BROWNLEE, J. *Long short-term memory networks with python: develop sequence prediction models with deep learning*. [S.l.]: Machine Learning Mastery, 2017. Citado na página 25.
- BUDUMA, N.; BUDUMA, N.; PAPA, J. *Fundamentals of deep learning*. [S.l.]: "O'Reilly Media, Inc.", 2022. Citado na página 17.

- CHAN, V. *Cristal de silício tipo P ou N com uma orientação de (100), (110) ou (111)*. 2021. <<https://www.powerwaywafer.com/pt/silicon-crystal.html>>. [Online; accessed 21-October-2021]. Citado na página 8.
- CHOLLET, F.; ALLAIRE, J. J. *Deep Learning mit R und Keras: Das Praxis-Handbuch von den Entwicklern von Keras und RStudio*. [S.l.]: MITP-Verlags GmbH & Co. KG, 2018. Citado 4 vezes nas páginas 17, 18, 23 e 24.
- CHOLLET, F. et al. *Keras*. GitHub, 2015. Disponível em: <<https://github.com/fchollet/keras>>. Citado 3 vezes nas páginas 31, 38 e 41.
- DATA61, C. *StellarGraph Machine Learning Library*. [S.l.]: GitHub, 2018. <<https://github.com/stellargraph/stellargraph>>. Citado na página 31.
- DHERE, N. G.; SHIRADKAR, N. S. Fire hazard and other safety concerns of photovoltaic systems. *Journal of Photonics for Energy*, International Society for Optics and Photonics, v. 2, n. 1, p. 022006, 2012. Citado na página 1.
- FAN, J. et al. Fault classification in photovoltaic arrays using graph signal processing. In: *2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*. [S.l.: s.n.], 2020. v. 1, p. 315–319. Citado na página 3.
- FAN, J. et al. Fault classification in photovoltaic arrays using graph signal processing. In: IEEE. *2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*. [S.l.], 2020. v. 1, p. 315–319. Citado 2 vezes nas páginas 3 e 4.
- HAGBERG, A. A.; SCHULT, D. A.; SWART, P. J. Exploring network structure, dynamics, and function using networkx. In: VAROQUAUX, G.; VAUGHT, T.; MILLMAN, J. (Ed.). *Proceedings of the 7th Python in Science Conference*. Pasadena, CA USA: [s.n.], 2008. p. 11 – 15. Citado na página 31.
- HARRIS, C. R. et al. Array programming with NumPy. *Nature*, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357–362, set. 2020. Disponível em: <<https://doi.org/10.1038/s41586-020-2649-2>>. Citado na página 31.
- HAYKIN, S. *Redes neurais: princípios e prática*. [S.l.]: Bookman Editora, 2007. Citado 3 vezes nas páginas 17, 22 e 28.
- HUNTER, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007. Citado na página 31.
- INTELBRAS. *Créditos de energia solar: 6 perguntas e respostas, howpublished = "<<https://blog.intelbras.com.br/creditos-de-energia-solar/>>"*. 2021. [Online; acessado em 12-Novembro-2022]. Citado na página 2.
- JERE, S.; KASENSE, B.; BWALYA, B. B. Univariate time-series analysis of second-hand car importation in zambia. *Open Journal of Statistics*, Scientific Research Publishing, v. 7, n. 4, p. 718–730, 2017. Citado na página 15.
- JR, M. V. R. B. *O problema de Euler – Grafos*. 2022. <<https://valci.com.br/home/2022/07/09/o-problema-de-euler-grafos/>>. [Online; acessado em 02-Junho-2022]. Citado na página 13.

KIM, S. Deep learning with r, François Chollet, Joseph J. Allaire, Shelter Island, NY: Manning. Wiley Periodicals, Inc., 2020. Citado na página 25.

LEITE, N. M. *A diferença entre a solução SolarEdge e outras topologias de inversores solares*. 2020. <<https://www.solaredge.com/br/solaredge-blog/solaredge-tipos-de-inversor-solar>>. [Online; accessed 19-November-2021]. Citado na página 30.

LJUNG, G. M.; BOX, G. E. On a measure of lack of fit in time series models. *Biometrika*, Oxford University Press, v. 65, n. 2, p. 297–303, 1978. Citado na página 16.

LOYE, G. *Long Short-Term Memory: From Zero to Hero with PyTorch*. 2019. <<https://blog.floydhub.com/long-short-term-memory-from-zero-to-hero-with-pytorch/>>. [Online; acessado em 28-Novembro-2022]. Citado 4 vezes nas páginas 25, 26, 27 e 28.

MAAS, A. L.; HANNUN, A. Y.; NG, A. Y. Rectifier nonlinearities improve neural network acoustic models. In: *Proc. icml*. [S.l.: s.n.], 2013. v. 30, n. 1, p. 3. Citado na página 20.

MEYER, E. L.; DYK, E. E. V. Assessing the reliability and degradation of photovoltaic module performance parameters. *IEEE Transactions on Reliability*, IEEE, v. 53, n. 1, p. 83–92, 2004. Citado na página 3.

MITCHELL, T. M.; MITCHELL, T. M. *Machine learning*. [S.l.]: McGraw-Hill New York, 1997. v. 1. Citado na página 17.

MORETTIN, P. A.; TOLOI, C. M. *Análise de séries temporais: modelos lineares univariados*. [S.l.]: Editora Blucher, 2018. Citado na página 14.

NETTO, P. O. B.; JURKIEWICZ, S. *Grafos: introdução e prática*. [S.l.]: Editora Blucher, 2017. Citado 2 vezes nas páginas 10 e 12.

PANKRATZ, A. *Forecasting with dynamic regression models*. [S.l.]: John Wiley & Sons, 2012. v. 935. Citado na página 16.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011. Citado na página 41.

PEI, T.; HAO, X. A fault detection method for photovoltaic systems based on voltage and current observation and evaluation. *Energies*, Multidisciplinary Digital Publishing Institute, v. 12, n. 9, p. 1712, 2019. Citado na página 2.

PLANAS, O. *Silício policristalino*. 2018. <<https://pt.solar-energia.net/energia-solar-fotovoltaica/elementos/painel-fotovoltaico/celula-fotovoltaica/silicio/silicio-policristalino>>. [Online; accessed 21-October-2021]. Citado na página 8.

RENOVÁVEIS, E. E. *Placa Solar Fotovoltaica: processo de fabricação e insumos!* 2019. <<https://www.ecoenergias.com.br/2019/10/11/placa-solar-fotovoltaica-processo-de-fabricacao-e-insumos/>>. [Online; acessado em 02-Junho-2021]. Citado na página 1.

ROSSUM, G. V.; DRAKE, F. L. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. ISBN 1441412697. Citado na página 31.

- ROSSUM, G. V.; JR, F. L. D. *Python reference manual*. [S.l.]: Centrum voor Wiskunde en Informatica Amsterdam, 1995. Citado na página 41.
- SEABOLD, S.; PERKTOLD, J. Statsmodels: Econometric and statistical modeling with python. In: AUSTIN, TX. *Proceedings of the 9th Python in Science Conference*. [S.l.], 2010. v. 57, p. 61. Citado 2 vezes nas páginas 15 e 31.
- SECTIDES. *Centro de Pesquisa, Inovação e Desenvolvimento (Cpid)*. 2020. <<https://inovacaoedesenvolvimento.es.gov.br/centro-de-pesquisa-inovacao-e-desenvolvimento-cpid>>. [Online; accessed 19-November-2021]. Citado na página 29.
- SHARADGA, H.; HAJIMIRZA, S.; BALOG, R. S. Time series forecasting of solar power generation for large-scale photovoltaic plants. *Renewable Energy*, Elsevier, v. 150, p. 797–807, 2020. Citado na página 3.
- SILVA, A. Identificação de funcionamento atípico de painel fotovoltaico. 2020. Citado 4 vezes nas páginas 3, 4, 8 e 9.
- SINGH, G. Solar power generation by pv (photovoltaic) technology: A review. *Energy*, v. 53, p. 1–13, 2013. ISSN 0360-5442. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0360544213001758>>. Citado 2 vezes nas páginas 2 e 9.
- SOLAREEDGE. *Otimizador de Potência Adicionável ao Módulo*. 2020. <[https://www.solaredge.com/br/products/power-optimizers#/>](https://www.solaredge.com/br/products/power-optimizers#/). [Online; accessed 15-November-2022]. Citado na página 10.
- SOLAREEDGE. *Otimizador de Potência Adicionável ao Módulo, howpublished = "<<https://www.solaredge.com/br/products/power-optimizers/>>"*. 2021. [Online; acessado em 12-Novembro-2022]. Citado na página 2.
- TEAM, T. pandas development. *pandas-dev/pandas: Pandas*. Zenodo, 2020. Disponível em: <<https://doi.org/10.5281/zenodo.3509134>>. Citado na página 31.
- VIDIGAL, B. R. et al. Modelo inteligente utilizando grafos para previsão da potência gerada por módulos de painéis fotovoltaicos. *Aceito para Publicação no Simpósio Brasileiro de Sistemas Elétricos, Santa Maria, RS*, 2022. Citado 4 vezes nas páginas 4, 5, 42 e 43.
- VIRTANEN, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, v. 17, p. 261–272, 2020. Citado na página 41.
- WANG, Y. et al. A graph guided convolutional neural network for surface defect recognition. *IEEE Transactions on Automation Science and Engineering*, IEEE Transactions on Automation Science and Engineering, p. 1–13, 2022. ISSN 1545-5955. Citado na página 3.
- WEI, W. W. Time series analysis. In: *The Oxford Handbook of Quantitative Methods in Psychology: Vol. 2*. [S.l.: s.n.], 2006. Citado 4 vezes nas páginas 13, 15, 16 e 33.
- WU, Z. et al. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, IEEE, v. 32, n. 1, p. 4–24, 2020. Citado na página 24.

XU, B. et al. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015. Citado na página 20.

YU, B.; YIN, H.; ZHU, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017. Citado 4 vezes nas páginas 3, 36, 38 e 42.

yWorks. *yEd*. 2019. [Online; acessado em 02-Fevereiro-2022]. Disponível em: <<https://www.yworks.com/products/yed>>. Citado na página 31.

ZHAO, L. et al. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, v. 21, n. 9, p. 3848–3858, 2020. Citado 3 vezes nas páginas 37, 38 e 42.

ZHAO, Y.; LEHMAN, B.; BALL, R. Graph-based semi-supervised learning for fault detection and classification in solar photovoltaic arrays. In: . [S.l.: s.n.], 2013. v. 30, p. 1628–1634. Citado na página 3.