



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Nicolas Moreira Branco

**Modelos de aprendizado de máquina para previsão da irradiância global horizontal de curto prazo usando dados de telemetria local e imagens *all-sky***

Florianópolis  
2023

Nicolas Moreira Branco

**Modelos de aprendizado de máquina para previsão da irradiância global horizontal de curto prazo usando dados de telemetria local e imagens *all-sky***

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina para a obtenção do título de mestre em Engenharia Elétrica.

Orientador: Prof. Danilo Silva, Ph.D.

Florianópolis

2023

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Branco, Nicolas Moreira

Modelos de aprendizado de máquina para previsão da irradiância global horizontal de curto prazo usando dados de telemetria local e imagens all-sky / Nicolas Moreira Branco ; orientador, Danilo Silva, 2023.

92 p.

2. Aprendizado de máquina. 3. Previsão. 4. Irradiância Global Horizontal. 5. Séries temporais. I. Silva, Danilo. II. Universidade Federal de Santa Catarina. Programa de Pós Graduação em Engenharia Elétrica. III. Título.

Nicolas Moreira Branco

**Modelos de aprendizado de máquina para previsão da irradiância global horizontal de curto prazo usando dados de telemetria local e imagens *all-sky***

O presente trabalho em nível de mestrado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Gerson Maximo Tiepolo, Dr.  
Instituição UTFPR

Profa. Helena Flávia Naspolini, Dra.  
Instituição UFSC

Prof. Laio Oriel Seman, Dr.  
Instituição UFSC

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de mestre em Engenharia Elétrica.

---

Prof. Telles Brunelli Lazzarin, Dr.  
Coordenação do Programa de  
Pós-Graduação

---

Prof. Danilo Silva, Ph.D.  
Orientador

Florianópolis, 2023.

Este trabalho é dedicado à minha família que sempre me apoiou e continua me apoiando durante todos esses anos.

## AGRADECIMENTOS

Primeiramente, à minha família por todo o incentivo e suporte que sempre me deram durante minha vida, especialmente aos meus pais, Juliana e Marco, as minhas tias Adelina e Goreti, minha avó Hilária e minha irmã Manuela, que estiveram sempre presentes durante esse percurso, sem eles eu não teria chegado até aqui.

Também, pela presença e apoio da minha namorada, Izabel; e amigos, Marcos, Volnei, Felipe, Patrick, Muriele, Iran, entre outros, que estiveram presentes e disponíveis durante essa e outras etapas da minha vida.

Além disso, aos professores que me incentivaram durante toda essa trajetória acadêmica em seguir esse caminho. Especialmente, ao meu orientador Danilo, por estar sempre presente para ensinar, com todo seu conhecimento, sendo uma ótima pessoa de se conviver e trabalhar. Também, ao professor Laio, por ter me auxiliado na parte técnica do *machine learning* e pela frequente disponibilidade e disposição para trabalhar junto e repassar seus conhecimentos.

Por fim, a possibilidade de ter feito esse mestrado, e ter estudado em universidades de excelência como a UTFPR e UFSC. E também, ao apoio e financiamento dessa pesquisa. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

*"Our passion for learning is the tool for our survival.  
(Carl Sagan, Cosmos ep. 11, 1980)*

## RESUMO

A energia solar vem aumentando sua participação na matriz energética de diversos países com elevadas taxas de crescimento. Essa fonte não é despachável e é afetada significativamente por fatores aleatórios relacionados às condições meteorológicas, que podem afetar muito sua geração. Desta forma, é importante realizar previsões de geração no curto prazo e em diversos horizontes para manter e melhorar a estabilidade do sistema elétrico. É comum separar a previsão de geração em duas etapas: prever a Irradiância Global Horizontal (*Global Horizontal Irradiance* - GHI) e depois prever a geração em si com base na GHI e outras variáveis. Esta dissertação se propõe a analisar e comparar modelos de aprendizado de máquina para a previsão da GHI 15 minutos a frente, utilizando somente dados de sensores, somente dados de imagem *all-sky* ou uma combinação de ambos. Para os modelos com dados de sensores, investigou-se principalmente a arquitetura de redes neurais recorrentes *Long Short-Term Memory* (LSTM). Para os modelos com imagens, utilizou-se redes neurais convolucionais (*Convolutional Neural Network* - CNN), em particular as arquiteturas ResNet, EfficientNet e VGG. Já para os modelos híbridos, investigou-se arquiteturas que combinam LSTM e ResNet, tendo como entrada dados de sensores e imagens, respectivamente. Os resultados demonstram que para modelos com dados de sensores, o melhor modelo atingiu um *forecast skill* igual a 14,2%, a partir do erro médio absoluto (*Mean Absolute Error* - MAE), em relação ao modelo persistente inteligente, obtendo resultados comparáveis com os da literatura. Além disso, percebeu-se que a adição de imagens não melhorou os resultados, de forma que o melhor modelo encontrado foi o modelo somente com dados de sensores.

**Palavras-chave:** Aprendizado de máquina; Inteligência artificial; Previsão; Irradiância Global Horizontal; Séries temporais.

## ABSTRACT

Solar energy has been increasing its share in the energy matrix of several countries with high growth rates. This source is not dispatchable and is significantly affected by random factors related to weather conditions, which can greatly affect its generation. Therefore, it is important to forecast its generation in the short term and at different horizons to maintain and improve the stability of the electrical system. It is common to separate solar generation forecasting into two steps: forecast the Global Horizontal Irradiance (GHI) and then forecast the generation itself based on the GHI and other variables. This dissertation proposes to analyze and compare machine learning models to forecast GHI 15 minutes ahead, using sensors data only, all-sky image data only, or a combination of both. For the models with sensors data, it was mainly investigated the Long Short-Term Memory (LSTM) recurrent neural network architecture. For models with images, the convolutional neural networks (CNN) were used, in particular the ResNet, EfficientNet and VGG architectures. For hybrid models, architectures that combine LSTM and ResNet were investigated, with sensors and image data as input, respectively. The results show that for models with sensors data, the best model achieved an forecast skill equal to 14.2%, with respect to the mean absolute error (MAE), in relation to the smart persistent model, obtaining results comparable to those in the literature. In addition, it was observed that the addition of images did not improve the results, so the best model found was the model with sensor data only.

**Keywords:** Machine learning; Artificial intelligence; Forecasting; Global Horizontal Irradiance; Time series.

## LISTA DE FIGURAS

Figura 1 – Conjunto de dados para previsão da GHI . . . . .	21
Figura 2 – Representação de um neurônio . . . . .	25
Figura 3 – Rede neural simples . . . . .	25
Figura 4 – Exemplo de uma rede MLP . . . . .	26
Figura 5 – Exemplos de funções de ativação . . . . .	27
Figura 6 – Conjuntos de dados . . . . .	27
Figura 7 – Bloco LSTM . . . . .	28
Figura 8 – Exemplo de rede convolucional . . . . .	29
Figura 9 – Arquitetura base da ResNet . . . . .	30
Figura 10 – Exemplo de divisão do conjunto de dados a partir de um modelo de árvore de decisão . . . . .	31
Figura 11 – Posição dos hemisférios em relação ao Sol no decorrer do ano . . .	33
Figura 12 – Possíveis recortes das imagens . . . . .	43
Figura 13 – Variação das variáveis alvo em todo o conjunto de dados . . . . .	46
Figura 14 – Histograma das variáveis alvo . . . . .	46
Figura 15 – MAE no conjunto de treinamento para possíveis variáveis alvo para modelos somente com dados de sensores . . . . .	52
Figura 16 – MAE no conjunto de validação para possíveis variáveis alvo para modelos somente com dados de sensores . . . . .	52
Figura 17 – MAE no conjunto de treinamento para modelo somente com dados de sensores com maior capacidade . . . . .	54
Figura 18 – MAE no conjunto de validação para modelo somente com dados de sensores com maior capacidade . . . . .	54
Figura 19 – MAE no conjunto de treinamento para modelo somente com dados de sensores com maior capacidade e com regularização L2 . . . . .	55
Figura 20 – MAE no conjunto de validação para modelo somente com dados de sensores com maior capacidade e com regularização L2 . . . . .	56
Figura 21 – MAE no conjunto de treinamento para modelo somente com dados de sensores com maior capacidade e com <i>dropout</i> . . . . .	56
Figura 22 – MAE no conjunto de validação para modelo somente com dados de sensores com maior capacidade e com <i>dropout</i> . . . . .	57
Figura 23 – Correlação $\phi_k$ entre atributos e a variável alvo $K_t$ . . . . .	58
Figura 24 – MAE no conjunto de treinamento para modelos MLP e LSTM so- mente com dados de sensores . . . . .	59
Figura 25 – MAE no conjunto de validação para modelos MLP e LSTM somente com dados de sensores . . . . .	59
Figura 26 – Arquitetura do melhor modelo utilizando somente dados de sensores	62

Figura 27 – MAE no conjunto de treinamento para possíveis variáveis alvo para modelos somente com o uso de imagens . . . . .	63
Figura 28 – MAE no conjunto de validação para possíveis variáveis alvo para modelos somente com o uso de imagens . . . . .	64
Figura 29 – MAE no conjunto de treinamento otimizando <i>batch_size</i> e taxa de aprendizado para modelos somente com o uso de imagens . . . . .	64
Figura 30 – MAE no conjunto de validação otimizando <i>batch_size</i> e taxa de aprendizado para modelos somente com o uso de imagens . . . . .	65
Figura 31 – MAE no conjunto de validação utilizando ou não pesos da ImageNet para modelos somente com o uso de imagens . . . . .	66
Figura 32 – MAE no conjunto de treinamento para diferentes tamanhos das imagens como entrada dos modelos somente com o uso de imagens . . . . .	67
Figura 33 – MAE no conjunto de validação para diferentes tamanhos das imagens como entrada dos modelos somente com o uso de imagens . . . . .	67
Figura 34 – MAE no conjunto de validação para modelos com diferentes arquiteturas CNN para modelos somente com o uso de imagens . . . . .	68
Figura 35 – MAE no conjunto de validação para modelos com somente uma imagem ou múltiplas imagens como entrada . . . . .	70
Figura 36 – MAE no conjunto de treinamento para modelos somente com o uso de imagens treinados por um número maior de épocas . . . . .	72
Figura 37 – MAE no conjunto de validação para modelos somente com o uso de imagens treinados por um número maior de épocas . . . . .	72
Figura 38 – Arquitetura do melhor modelo somente utilizando imagens . . . . .	73
Figura 39 – MAE para o conjunto de treinamento para modelos híbridos sem importar pesos dos melhores modelos somente com uso de sensores ou imagens . . . . .	75
Figura 40 – MAE para o conjunto de validação para modelos híbridos sem importar pesos dos melhores modelos somente com uso de sensores ou imagens . . . . .	75
Figura 41 – MAE para o conjunto de treinamento para modelos híbridos importando pesos dos melhores modelos somente com uso de sensores e imagens . . . . .	76
Figura 42 – MAE para o conjunto de validação para modelos híbridos importando pesos dos melhores modelos somente com uso de sensores e imagens . . . . .	76
Figura 43 – MAE para o conjunto de validação para modelos híbridos importando pesos do melhor modelo somente com uso de sensores e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet . . . . .	77
Figura 44 – Arquitetura do melhor modelo híbrido . . . . .	80

Figura 45 – Resultados dos modelos dia nublado de inverno - 2016-01-24 . . . .	84
Figura 46 – Resultados dos modelos dia ensolarado de inverno - 2016-01-25 . .	85
Figura 47 – Resultados dos modelos dia nublado de verão - 2016-06-04 . . . .	85
Figura 48 – Resultados dos modelos dia ensolarado de verão - 2016-06-05 . . .	86

## LISTA DE TABELAS

Tabela 1 – Momentos de imputação dos dados de sensores . . . . .	39
Tabela 2 – Momentos não imputados . . . . .	40
Tabela 3 – Comparação dos diferentes modelos de céu claro em relação à GHI de céu claro real . . . . .	40
Tabela 4 – Dados tabulares . . . . .	42
Tabela 5 – Disponibilidade dos dados . . . . .	45
Tabela 6 – Informações sobre a série de dados do GHI . . . . .	47
Tabela 7 – Informações sobre a série de dados do $K_t$ . . . . .	47
Tabela 8 – Informações sobre a série de dados do $SP_{diff}$ . . . . .	47
Tabela 9 – Métrica MAE para o conjunto de validação utilizando possíveis variáveis alvo para modelos somente com dados de sensores . . . . .	52
Tabela 10 – Métrica MAE para o conjunto de validação otimizando <i>batch_size</i> e taxa de aprendizado para modelos somente com dados de sensores . . . . .	53
Tabela 11 – Métrica MAE para o conjunto de validação com diferentes normalizações dos dados tabulares para modelos somente com dados de sensores . . . . .	53
Tabela 12 – Métrica MAE para o conjunto de validação com diferentes funções perda para modelos somente com dados de sensores . . . . .	53
Tabela 13 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores com maior capacidade . . . . .	55
Tabela 14 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores com maior capacidade e com regularização L2 . . . . .	56
Tabela 15 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores com maior capacidade e com <i>dropout</i> . . . . .	57
Tabela 16 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores variando o número de instâncias passadas . . . . .	57
Tabela 17 – Métrica MAE para o conjunto de validação utilizando somente os N atributos mais importantes para modelos somente com dados de sensores . . . . .	58
Tabela 18 – Métrica MAE para o conjunto de validação comprando com modelos MLP e LSTM somente com dados de sensores . . . . .	60
Tabela 19 – Parâmetros dos modelo Ridge . . . . .	60
Tabela 20 – Parâmetros dos modelos não neurais de árvore . . . . .	61
Tabela 21 – Métrica MAE para os conjuntos de treinamento e validação utilizando modelos não neurais somente com dados de sensores . . . . .	61

Tabela 22 – Métrica MAE para o conjunto de validação comparando possíveis variáveis alvo para modelos somente com o uso de imagens . . . . .	63
Tabela 23 – Métrica MAE para o conjunto de validação otimizando o <i>batch_size</i> e taxa de aprendizado para modelos somente com o uso de imagens	65
Tabela 24 – Métrica MAE para o conjunto de validação com diferentes funções perda para modelos somente com o uso de imagens . . . . .	65
Tabela 25 – Métrica MAE para o conjunto de validação utilizando ou não pesos da ImageNet para modelos somente com o uso de imagens . . . . .	66
Tabela 26 – Métrica MAE para o conjunto de validação com diferentes recortes das imagens para modelos somente com o uso de imagens . . . . .	66
Tabela 27 – Métrica MAE para o conjunto de validação com diferentes tamanhos das imagens para modelos somente com o uso de imagens . . . . .	67
Tabela 28 – Métrica MAE para o conjunto de validação para modelos com diferentes arquiteturas CNN para modelos somente com o uso de imagens	68
Tabela 29 – Métrica MAE para o conjunto de validação com ou sem normalização da imagem para modelos somente com o uso de imagens . . . . .	69
Tabela 30 – Métrica MAE para o conjunto de validação comparando modelos que utilizam somente imagem atual versus diferença entre imagem atual e anterior . . . . .	69
Tabela 31 – Métrica MAE para o conjunto de validação comparando modelos com somente uma imagem ou múltiplas imagens como entrada . . . . .	70
Tabela 32 – Métrica MAE para o conjunto de validação utilizando ou não diferentes DA para o treinamento de modelos somente com o uso de imagens . . . . .	71
Tabela 33 – Métrica MAE para o conjunto de validação para modelos somente utilizando imagens treinados por um número maior de épocas . . . . .	72
Tabela 34 – Métrica MAE para o conjunto de validação para modelos híbrido sem importar pesos dos melhores modelos somente com uso de sensores ou imagens . . . . .	75
Tabela 35 – Métrica MAE para o conjunto de validação para modelos híbridos importando pesos dos melhores modelos somente com uso de sensores e imagens . . . . .	77
Tabela 36 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet . . . . .	77

Tabela 37 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet, e adicionando uma MLP no depois da concatenação dos dados . . . . .	78
Tabela 38 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores (com todos os atributos ao invés dos 10 melhores) e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet, e adicionando uma MLP no depois da concatenação dos dados . . .	79
Tabela 39 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores (com todos os atributos ao invés dos 10 melhores) e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet, sem adição de uma MLP no depois da concatenação dos dados . . . . .	79
Tabela 40 – Resultados compilados referentes à etapa de seleção de hiperparâmetros (o S representa os experimentos somente com sensores, o I somente com imagens <i>all-sky</i> e o H os referentes ao modelo híbrido)	80
Tabela 41 – Métricas de erro para o retreinamento - conjunto de teste completo	81
Tabela 42 – Métricas de erro para o retreinamento - períodos ensolarados . . .	82
Tabela 43 – Métricas de erro para o retreinamento - períodos nublados . . . . .	82
Tabela 44 – Resultados no conjunto de teste dos melhores modelos treinados com função perda L1 . . . . .	83
Tabela 45 – Resultados no conjunto de teste dos melhores modelos treinados com função perda L2 . . . . .	83

## LISTA DE ABREVIATURAS E SIGLAS

ANN	<i>Artificial Neural Network</i>
CNN	<i>Convolutional Neural Network</i>
DHI	<i>Diffuse Horizontal Irradiance</i>
DL	<i>Deep Learning</i>
DNI	<i>Direct Normal Irradiance</i>
FS	<i>Forecast Skill</i>
GHI	<i>Global Horizontal Irradiance</i>
INPE	Instituto Nacional de Pesquisas Espaciais
LABENS	Laboratório de Energia Solar da UTFPR
LSTM	<i>Long Short-Term Memory</i>
MAE	<i>Mean Absolute Error</i>
MAPE	<i>Mean Absolute Percentage Error</i>
ML	<i>Machine Learning</i>
MLP	<i>Multi-Layer Perceptron</i>
NLP	<i>Natural Language Processing</i>
nMAE	<i>Normalized Mean Absolute Error</i>
NREL	<i>National Renewable Energy Laboratory</i>
nRMSE	<i>Normalized Root Mean Squared Error</i>
ONS	Operador Nacional do Sistema
ReLU	<i>Rectified Linear Unit</i>
ResNet	<i>Residual Neural Network</i>
RMSE	<i>Root Mean Squared Error</i>
RNN	<i>Recurrent Neural Network</i>
SGD	<i>Stochastic Gradient Descent</i>
SP	<i>Smart Persistent</i>
SPA	<i>Solar Position Algorithm</i>
sSolis	<i>Simplified Solis</i>
SVM	<i>Support Vector Machine</i>

## LISTA DE SÍMBOLOS

$K_t$  Índice de céu claro

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>20</b>
1.1	OBJETIVOS	22
<b>1.1.1</b>	<b>Geral</b>	<b>22</b>
<b>1.1.2</b>	<b>Específicos</b>	<b>22</b>
1.2	CONTRIBUIÇÕES	23
1.3	ORGANIZAÇÃO DO TRABALHO	23
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>24</b>
2.1	APRENDIZADO DE MÁQUINA	24
<b>2.1.1</b>	<b>Modelos neurais</b>	<b>24</b>
<b>2.1.2</b>	<b>Neurônio</b>	<b>24</b>
<b>2.1.3</b>	<b>MLP</b>	<b>24</b>
<b>2.1.4</b>	<b>Aprendizado profundo</b>	<b>25</b>
<b>2.1.5</b>	<b>Treinamento</b>	<b>26</b>
<b>2.1.6</b>	<b>Funções de ativação</b>	<b>26</b>
<b>2.1.7</b>	<b>Seleção de hiperparâmetros e divisão dos conjuntos de dados</b>	<b>27</b>
<b>2.1.8</b>	<b>Redes neurais recorrentes</b>	<b>28</b>
<b>2.1.9</b>	<b>Redes convolucionais</b>	<b>29</b>
<b>2.1.10</b>	<b>Árvore de decisão</b>	<b>30</b>
<b>2.1.11</b>	<b>Métodos combinados</b>	<b>32</b>
2.2	GERAÇÃO FOTOVOLTAICA	32
2.3	MODELO PERSISTENTE INTELIGENTE	34
2.4	MÉTRICAS DE ERRO	34
2.5	NORMALIZAÇÃO DOS DADOS TABULARES	35
<b>3</b>	<b>MATERIAIS E MÉTODOS</b>	<b>37</b>
3.1	ANÁLISE E PRÉ-PROCESSAMENTO DOS DADOS DE ENTRADA	37
<b>3.1.1</b>	<b>Conjunto de dados</b>	<b>37</b>
<b>3.1.2</b>	<b>Limpeza inicial e imputação dos dados</b>	<b>38</b>
<b>3.1.3</b>	<b>Escolha do melhor modelo de céu claro</b>	<b>39</b>
<b>3.1.4</b>	<b>Adição de variáveis de entrada</b>	<b>41</b>
<b>3.1.5</b>	<b>Imagens</b>	<b>41</b>
3.1.5.1	Opções de recortes	41
3.1.5.2	Normalização da imagem	43
<b>3.1.6</b>	<b>Verificar disponibilidade dos dados</b>	<b>44</b>
<b>3.1.7</b>	<b>Seleção de imagens a serem utilizadas</b>	<b>44</b>
<b>3.1.8</b>	<b>Definição das variáveis alvo</b>	<b>44</b>
<b>3.1.9</b>	<b>Normalização dos dados tabulares</b>	<b>45</b>
<b>3.1.10</b>	<b>Análise exploratória das variáveis alvo</b>	<b>45</b>

3.2	EXPERIMENTOS . . . . .	48
3.2.1	<b>Ferramentas e bibliotecas utilizadas . . . . .</b>	<b>49</b>
4	<b>EXPERIMENTOS E RESULTADOS . . . . .</b>	<b>51</b>
4.1	MODELOS UTILIZANDO DADOS DE SENSORES . . . . .	51
4.1.1	<b>Variável alvo . . . . .</b>	<b>51</b>
4.1.2	<b><i>Batch size</i> e taxa de aprendizado . . . . .</b>	<b>52</b>
4.1.3	<b>Normalização dos dados tabulares . . . . .</b>	<b>53</b>
4.1.4	<b>Função perda . . . . .</b>	<b>53</b>
4.1.5	<b>Aumento da capacidade do modelo e aplicação de regularização L2 ou <i>dropout</i> . . . . .</b>	<b>54</b>
4.1.6	<b>Número de instâncias passadas . . . . .</b>	<b>56</b>
4.1.7	<b>Somente com atributos mais importantes . . . . .</b>	<b>57</b>
4.1.8	<b>MLP . . . . .</b>	<b>58</b>
4.1.9	<b>Modelos não neurais . . . . .</b>	<b>60</b>
4.1.10	<b>Resumo dos modelos somente com dados de sensores . . . . .</b>	<b>61</b>
4.2	MODELOS UTILIZANDO SOMENTE IMAGENS . . . . .	62
4.2.1	<b>Variável alvo . . . . .</b>	<b>63</b>
4.2.2	<b><i>Batch size</i> e taxa de aprendizado . . . . .</b>	<b>64</b>
4.2.3	<b>Função perda . . . . .</b>	<b>64</b>
4.2.4	<b>Importar ou não os pesos da ImageNet . . . . .</b>	<b>65</b>
4.2.5	<b>Recorte da imagem . . . . .</b>	<b>65</b>
4.2.6	<b>Tamanho da imagem . . . . .</b>	<b>66</b>
4.2.7	<b>Possíveis arquiteturas . . . . .</b>	<b>67</b>
4.2.8	<b>Normalização das imagens . . . . .</b>	<b>68</b>
4.2.9	<b>Diferença entre imagem atual e anterior . . . . .</b>	<b>69</b>
4.2.10	<b>Múltiplas imagens . . . . .</b>	<b>69</b>
4.2.11	<b><i>Data augmentation</i> . . . . .</b>	<b>70</b>
4.2.12	<b>Treinar por mais épocas . . . . .</b>	<b>71</b>
4.2.13	<b>Resumo dos modelos de imagem . . . . .</b>	<b>72</b>
4.3	MODELOS COM DADOS HÍBRIDOS . . . . .	74
4.3.1	<b>Sem importar pesos dos outros testes . . . . .</b>	<b>74</b>
4.3.2	<b>Importando melhores pesos para ambos . . . . .</b>	<b>75</b>
4.3.3	<b>Importando somente pesos da LSTM . . . . .</b>	<b>76</b>
4.3.4	<b>Adicionando uma camada densa . . . . .</b>	<b>78</b>
4.3.5	<b>Com todos os atributos como entrada . . . . .</b>	<b>78</b>
4.3.6	<b>Resumo dos modelos híbridos . . . . .</b>	<b>78</b>
4.4	COMPILADOS DA SELEÇÃO DE HIPERPARÂMETROS . . . . .	79
4.5	RETREINAMENTO E RESULTADOS NO CONJUNTO DE TESTE . .	81
4.5.1	<b>Previsões dos modelos para dias específicos . . . . .</b>	<b>84</b>

<b>5</b>	<b>CONCLUSÃO E DISCUSSÃO . . . . .</b>	<b>87</b>
5.1	PRÓXIMOS TRABALHOS . . . . .	88
	<b>REFERÊNCIAS . . . . .</b>	<b>89</b>

## 1 INTRODUÇÃO

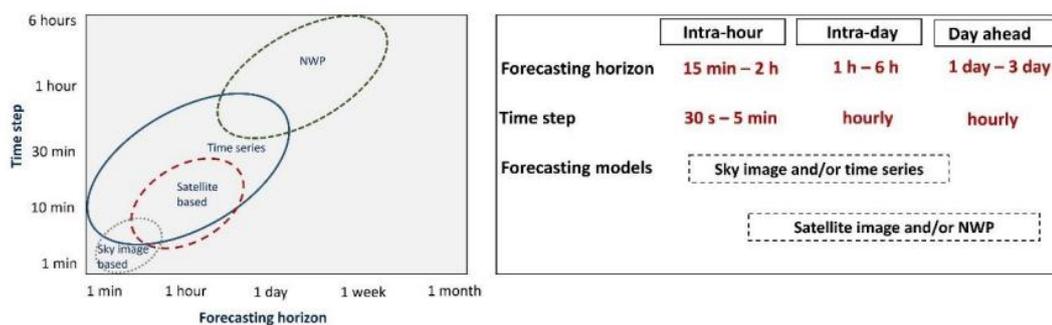
A energia solar vem aumentando sua participação na matriz energética de diversos países, em muitos casos com uma elevada taxa de crescimento. Por exemplo, no Brasil, a energia solar atingiu mais de 2,5% da capacidade instalada da matriz elétrica no ano de 2021. Isso representa um aumento de quase 190 vezes em relação ao ano de 2015, segundo dados da Empresa de Pesquisa Energética (EPE, 2022). Esse grande aumento pode estar relacionado à redução considerável dos custos dos painéis (DATA, 2023) e ao aumento do custo da energia (CORSINI, 2022), entre outros possíveis fatores.

Essa mudança significativa é relevante de ser observada e estudada, pois a energia solar, diferente de outras fontes, não é diretamente controlável. Ela é intermitente em sua natureza, sempre variando ao decorrer do dia e, entre dias, ao decorrer do ano. Além disso, e ainda mais relevante, existem variações relacionadas às condições atmosféricas, podendo variar significativamente sua geração em poucos minutos. Por exemplo, em Chu *et al.* (2015), os autores observaram que em pelo menos 1% dos casos estudados houve uma variação da irradiância em 10 minutos com amplitude de mais de 50% da irradiância de céu claro do momento. Esse contexto precisa ser considerado, buscando estratégias para compensar essas variações, por exemplo, ajustando fontes controláveis, atuando em sistemas de baterias ou outros processos similares. Isso é necessário para manter a estabilidade do sistema energético (KUMAR; PANDEY; SINHA, 2016; STEFFEL *et al.*, 2012). Para tal, a previsão da geração solar em diversos horizontes é muito relevante, e essa informação pode ser utilizada pelo Operador Nacional do Sistema (ONS), ou outras entidades mundo afora, para manter ou melhorar a estabilidade do sistema energético, mesmo com o aumento significativo da participação de fontes renováveis, em especial a solar.

O principal fator para a previsão da geração solar é a irradiância. Com isso, é comum separar a previsão da geração em duas etapas: (1) prever a irradiância futura; (2) prever a geração futura com base na irradiância prevista no item (1) e outros fatores climáticos e relacionados à cada sistema fotovoltaico em si. Os dados e métodos mais indicados variam em relação ao horizonte de previsão. A Figura 1 mostra diferentes conjuntos de dados e em quais contextos eles são mais relevantes para a previsão da Irradiância Global Horizontal (*Global Horizontal Irradiance* - GHI). Para previsões de curto prazo até duas horas, os métodos mais recomendados utilizam séries temporais, imagens *all-sky*, utilizadas individualmente ou em combinação.

Em relação aos principais modelos para a previsão de curto prazo da GHI, a partir de 2018, percebe-se um maior foco em modelos de aprendizado de máquina. Dentre esses modelos estudados estão redes neurais artificiais (*Artificial Neural Network* - ANN), perceptron multicamadas (*Multi-Layer Perceptron* - MLP), redes neurais convo-

Figura 1 – Conjunto de dados para previsão da GHI



Fonte: Rajagukguk, Ramadhan e Lee (2020).

lucionais (*Convolutional Neural Network - CNN*), LSTMs (*Long Short-Term Memory*), entre outros (JUNCKLAUS MARTINS *et al.*, 2022). O aprendizado de máquina busca reduzir a dependência de especialistas de cada área para encontrar os melhores atributos possíveis para cada problema e busca criar modelos e sistemas que consigam aprender e extrair o máximo de informações diretamente dos dados.

Em Yang *et al.* (2022), aponta-se que a área de previsão de GHI ainda é relativamente nova e sem muitos parâmetros ou boas práticas bem definidas. Além disso, muitos trabalhos utilizam conjuntos de dados distintos, metodologias e outros contextos, o que torna a comparação entre estudos em muitos casos não trivial. Vários desses trabalhos podem ser divididos em dois grupos principais: aqueles focados na área de energia solar e aqueles focados na área de aprendizado de máquina.

No primeiro grupo, os modelos de aprendizado de máquina não são tomados como o foco do estudo e, por vezes, não são muito otimizados. Nesses trabalhos, os modelos são comparados com modelos de referência da área, como o modelo persistente inteligente (*Smart Persistent - SP*). Por exemplo, em Wentz *et al.* (2022), é realizada a previsão da GHI para 1, 15 e 60 minutos à frente utilizando dados de telemetria local e modelos MLP e LSTM. Esses resultados são comparados com o modelo SP, mas os hiperparâmetros e configurações dos modelos de aprendizado não são explorados, apenas os resultados para aqueles parâmetros específicos são comparados.

Já no segundo grupo, há um foco maior nas otimizações dos modelos de aprendizado de máquina, porém esses trabalhos muitas vezes não comparam os resultados com modelos de referência da área, como o persistente. Isso ocorre em Pi *et al.* (2022), por exemplo, onde é feita a previsão da GHI de 10, 30 e 60 minutos à frente. Nesse trabalho, são comparadas nove arquiteturas, com normalização dos dados, seleção de atributos, transformações, entre outros. No entanto, esses modelos são comparados apenas entre si, sem a comparação com o modelo SP ou outro modelo de referência da área. Casos similares também podem ser observados nos trabalhos de Maciel *et al.*

(2021) e Wang *et al.* (2021).

Também existem trabalhos que fogem dessa regra, como em Paletta, Arbod e Lasenby (2021). Nesse caso, são utilizadas imagens de duas exposições diferentes e dados auxiliares, como GHI e informações relacionadas à posição do Sol. No trabalho, são comparadas diferentes arquiteturas complexas e otimizadas, além de serem postas à prova em relação ao modelo persistente. Nesse trabalho, para a previsão de 10 minutos, o resultado do melhor modelo obteve um FS em relação ao erro médio absoluto (*Mean Absolute Error* - MAE) igual a 10,4% em comparação ao SP.

Fora desses grupos, alguns trabalhos também realizam a previsão da potência gerada diretamente, como feito em Zhang *et al.* (2018). Nesse trabalho, foi realizada a previsão da potência para 1, 2, 5 e 10 minutos a frente, utilizando a potência instantânea, outras variáveis auxiliares e imagens. Para a previsão de 10 minutos, obtém-se o FS de 12,1% em relação ao modelo persistente utilizando o MAE como métrica de referência.

Mesmo nesses trabalhos mais completos e com melhorias dos modelos e comparações com a modelos de referência como o SP, acaba não sendo observado qual a contribuição de cada um dos conjuntos de dados utilizados. Para melhor entender qual parte dos modelos ou dados pode estar trazendo informações mais relevantes para a previsão da GHI, é importante a existência de trabalhos comparando a utilização apenas de sensores versus apenas de imagens *all-sky* versus ambos. Além disso, é importante seguir as boas práticas para teste e otimização dos modelos de aprendizado de máquina e comparar os resultados com o SP. No entanto, até o conhecimento do autor, não existem trabalhos contendo todos esses requisitos.

## 1.1 OBJETIVOS

### 1.1.1 Geral

Essa dissertação tem como objetivo analisar e comparar arquiteturas e possíveis dados de entrada (incluindo dados de sensores e imagens *all-sky*) para a previsão de curto prazo da GHI.

### 1.1.2 Específicos

1. Realizar as comparações desse trabalho em um conjunto de dados público e de fácil acesso;
2. Avaliar possíveis arquiteturas de aprendizado de máquina, otimizando-as, para a previsão da GHI 15 minutos a frente;
3. Analisar a previsão de cada categoria de dados de entrada (somente sensores versus somente imagens *all-sky* versus ambos);

4. Comparar os resultados obtidos com o modelo persistente inteligente.

## 1.2 CONTRIBUIÇÕES

As principais contribuições dessa pesquisa estão relacionadas a estudar os pontos propostos nos objetivos e compará-los todos de forma unificada e coerente. Isso é relevante, pois comparar trabalhos distintos da área de energia solar não é trivial. Cada conjunto de dados, período selecionado para estudo, modelo de céu claro utilizado e tantas outras variáveis podem dificultar ou impossibilitar a comparação com outras pesquisas. Desta forma, entende-se como positiva a compilação de todos esses processos nessa dissertação.

## 1.3 ORGANIZAÇÃO DO TRABALHO

O restante do trabalho está organizado da seguinte forma: no Capítulo 2, discute-se a fundamentação teórica, principais ferramentas, modelos e conceitos utilizados na dissertação. A metodologia é explicada no Capítulo 3. Os experimentos e resultados são descritos no Capítulo 4. Finalmente, os resultados são discutidos no Capítulo 5, juntamente com a conclusão.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, discute-se os principais modelos, ferramentas e conceitos utilizados e abordados neste estudo. Os acrônimos e termos são mantidos em inglês sempre que possível, a fim de facilitar e permitir uma melhor comparação com trabalhos internacionais. Além disso, muitos termos não possuem uma tradução oficial ou são incomuns e desconhecidos.

### 2.1 APRENDIZADO DE MÁQUINA

O aprendizado de máquina (*Machine Learning* - ML), é a ciência de programar computadores para que possam aprender a partir dos dados (GÉRON, 2019). Esse conceito busca construir sistemas que representem informações de maneiras não lineares e sejam capazes de resolver problemas, como classificar se um e-mail é spam ou não, prever se vai chover nos próximos dias e muitas outras aplicações, desde as mais simples até as mais difíceis.

#### 2.1.1 Modelos neurais

Os modelos neurais criam e utilizam representações de neurônios, inspirados pelo neurônio biológico, com o objetivo de obter representações complexas das informações contidas nas entradas. Esses sistemas podem fornecer bons resultados para diversos problemas, com soluções mais compactas em comparação com alternativas como o *Support Vector Machine* (SVM) (BISHOP, 2006).

#### 2.1.2 Neurônio

O neurônio é a base desses tipos de modelos e uma possível representação é mostrada na Figura 2. Para calcular sua saída, realiza-se uma combinação linear da entrada, que é então passada por uma função de ativação. A saída é definida por

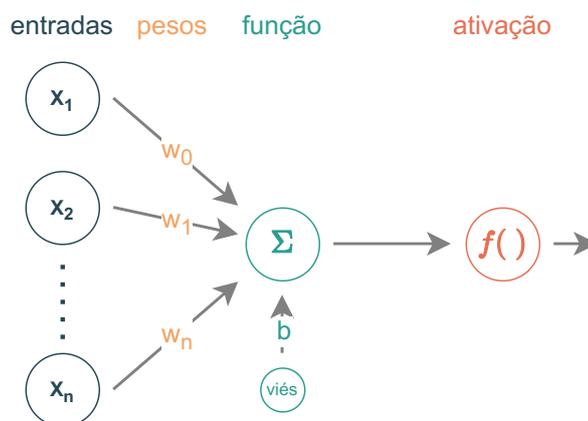
$$\hat{y} = f(\mathbf{w}\mathbf{x} + b) \quad (1)$$

onde  $\mathbf{x}$  representa a entrada, sendo um vetor com  $n$  elementos,  $\mathbf{w}$  é um conjunto de pesos que pondera cada um dos elementos da entrada,  $b$  é um fator de viés,  $f$  é a função de ativação e  $\hat{y}$  é a saída do neurônio (GÉRON, 2019).

#### 2.1.3 MLP

Agrupando mais neurônios, pode-se construir redes mais complexas. Por exemplo, a Figura 3 mostra uma rede com 2 entradas, 1 camada oculta com 2 neurônios e outro neurônio na saída. Nesse caso,  $\mathbf{x}$  são os dados de entrada aplicados à rede,  $\mathbf{w}$  são os pesos entre as entradas e os neurônios da camada oculta, ou entre a camada

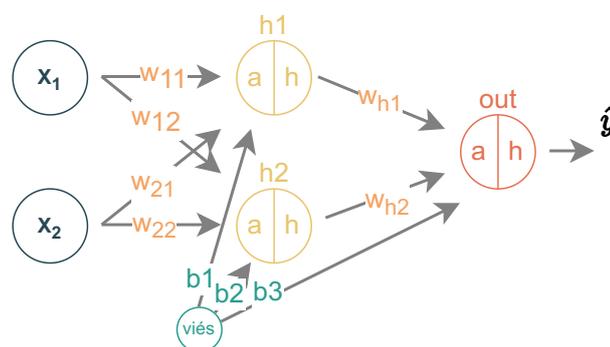
Figura 2 – Representação de um neurônio



Fonte: Autor.

oculta e o neurônio de saída,  $\mathbf{b}$  representa os termos de viés de cada neurônio, e  $\hat{y}$  é a saída da rede após seu processamento.

Figura 3 – Rede neural simples



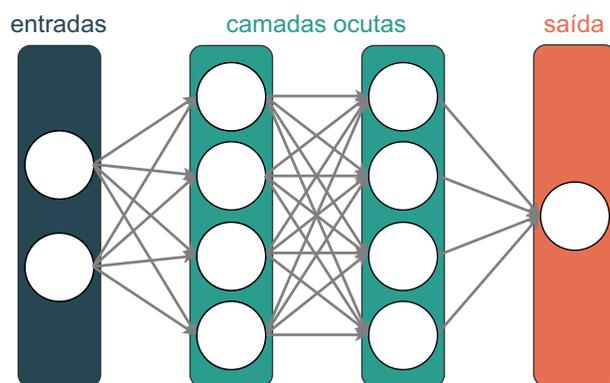
Fonte: Autor.

Essas redes com mais camadas são chamadas de *Multi-Layer Perceptron* (MLP) e consistem em um maior número de neurônios distribuídos em uma ou mais camadas ocultas (GÉRON, 2019). A Figura 4 mostra um exemplo com 2 entradas, 2 camadas ocultas com 4 neurônios cada e uma saída no final.

#### 2.1.4 Aprendizado profundo

Inicialmente, o aprendizado profundo (*Deep Learning* - DL) foi definido como redes com mais de duas camadas ocultas. No entanto, hoje em dia, é comum encontrar redes com dezenas ou centenas de camadas, de modo que a definição de aprendizado profundo pode variar entre diferentes trabalhos. Ainda assim, é comum chamar qualquer modelo de redes neurais de aprendizado profundo, mesmo que possuam poucas camadas (GÉRON, 2019). Essas redes mais profundas buscam aumentar a

Figura 4 – Exemplo de uma rede MLP



Fonte: Autor.

capacidade das redes, permitindo que elas resolvam problemas cada vez mais complexos.

### 2.1.5 Treinamento

O método mais utilizado para treinar essas redes é o *backpropagation*. Esse método utiliza a descida do gradiente com técnicas eficazes para calcular o gradiente em apenas duas passagens na rede: uma no sentido direto e outra no sentido reverso. Dessa forma, é possível calcular o gradiente em relação à todas as partes do modelo, incluindo os pesos e os vieses. Isso permite determinar quais parâmetros devem ser ajustados para reduzir o erro do modelo. Esse processo é repetido até atingir um limite máximo de iterações ou quando alguma condição de parada é satisfeita (GÉRON, 2019).

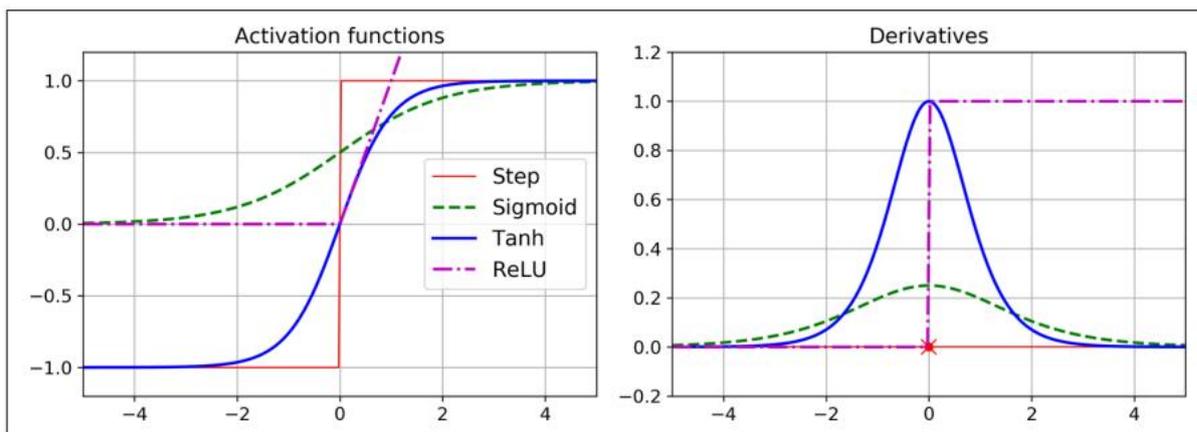
Outro ponto importante em relação ao treinamento é entender fatores relacionados à inicialização dos pesos e à aleatoriedade. O treinamento de uma rede neural envolve a otimização de uma função multidimensional não convexa, o que significa que, dependendo da inicialização, é possível cair em mínimos locais. Espera-se que os mínimos locais estejam próximos do mínimo global, mas isso nem sempre é o caso. Uma maneira de lidar com esse problema é inicializar os pesos usando estratégias aleatórias e realizar mais de um treinamento para a mesma rede ou conjunto de parâmetros. No entanto, isso também aumenta o custo computacional associado (NARKHEDE; BARTAKKE; SUTAONE, 2022).

### 2.1.6 Funções de ativação

A função de ativação tem o papel de introduzir não-linearidades nos modelos neurais, permitindo que eles sejam mais do que apenas combinações lineares das entradas. Existem diversas funções de ativação possíveis, como mostrado na Figura

5. É importante destacar que as funções de ativação devem ser diferenciáveis, para que o modelo possa aprender com base nos dados, seja utilizando o *Stochastic Gradient Descent* (SGD) ou outros otimizadores. A função de ativação mais amplamente utilizada é a *Rectified Linear Unit* (ReLU), que apresenta bons resultados na prática e possui uma computação rápida, embora não seja diferenciável em zero (GÉRON, 2019).

Figura 5 – Exemplos de funções de ativação

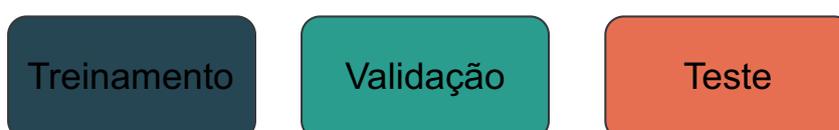


Fonte: Géron (2019).

### 2.1.7 Seleção de hiperparâmetros e divisão dos conjuntos de dados

Para selecionar os melhores parâmetros nos modelos de aprendizado de máquina, são testadas várias combinações de hiperparâmetros para determinar qual delas é capaz de prever o problema em questão de forma mais eficaz. O processo comum envolve a divisão do conjunto de dados em subconjuntos, cada um com uma finalidade específica. Um exemplo de divisão pode ser observado na Figura 6. Parte dos dados é usada para treinar os modelos, formando o conjunto de treinamento; outra parte é utilizada para avaliar quais combinações de hiperparâmetros apresentam os melhores resultados, formando o conjunto de validação; por fim, o conjunto de teste é usado apenas no final para avaliar o desempenho do modelo em um conjunto de dados ainda não visto.

Figura 6 – Conjuntos de dados

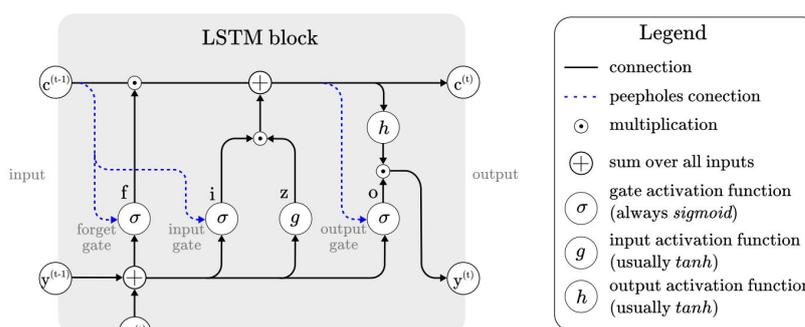


Fonte: Autor.

### 2.1.8 Redes neurais recorrentes

As Redes Neurais Recorrentes (*Recurrent Neural Network* - RNN) são redes que se concentram em sequências temporais ou sequências de informações. Aplicações comuns desse tipo de rede incluem previsão de preços de ações ou análise de sentimentos em textos, por exemplo (GÉRON, 2019). Para lidar com o problema do gradiente desaparecendo ou explodindo, que pode ocorrer em outras redes recorrentes, foi proposta a LSTM (HOCHREITER; SCHMIDHUBER, 1997). Essa rede ainda é utilizada para processamento de séries temporais (VAN HOUTD; MOSQUERA; NÁPOLES, 2020), mas vem sendo substituída por transformers desde sua introdução em 2017 (PATWARDHAN; MARRONE; SANSONE, 2023). Para o processamento de texto (*Natural Language Processing* - NLP) essa substituição praticamente já ocorreu e a LSTM já praticamente não é utilizada mais para esse propósito, a não ser para aplicações mais triviais. A estrutura de um bloco básico de LSTM é mostrada na Figura 7.

Figura 7 – Bloco LSTM



Fonte: Van Houdt, Mosquera e Nápoles (2020).

O funcionamento da LSTM é definido por todas essas subparcelas. O  $y$  representa a memória recente (*short-term*) e a saída, e o  $c$  a memória de longo prazo (*long-term*). O  $t$  e  $t-1$  estão relacionados aos valores atuais ou anteriores das variáveis. O processo principal pode ser dividido em algumas etapas:

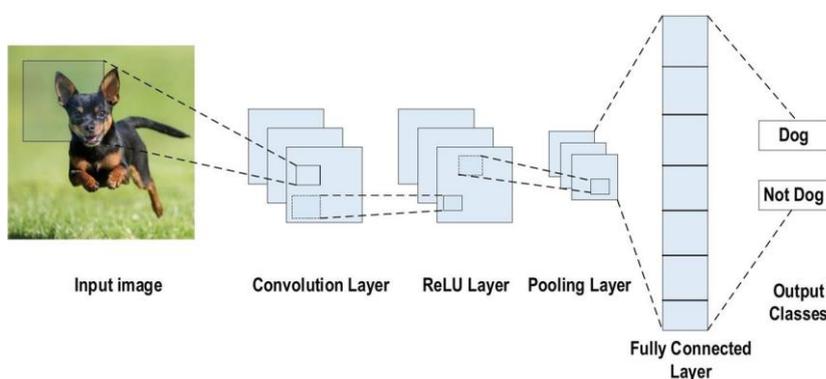
1. a porta do esquecimento controla qual parte da memória de longo prazo deve ser excluída;
2. a entrada controla como a memória de longo prazo deve ser mantida ou atualizada;
3. a atualização modifica o novo valor da memória de longo prazo;
4. a porta de saída controla a ativação da saída atual em função da memória de longo prazo e da entrada atual.

### 2.1.9 Redes convolucionais

As CNNs surgiram na década de 1980 a partir do estudo do processamento visual no cérebro, com o objetivo de replicar o funcionamento dos neurônios biológicos. As CNNs têm sido amplamente utilizadas para processamento de imagens, classificação de vídeos e várias outras aplicações (GÉRON, 2019). Nesta Subseção, é descrito o funcionamento das CNNs em relação ao processamento de imagens.

A ideia por trás das CNNs é reconhecer padrões em pequenas regiões das imagens e, por meio de várias camadas, processar esses padrões para identificar objetos e informações nas imagens. Embora isso também possa ser feito diretamente com redes profundas, torna-se computacionalmente proibitivo para imagens com resoluções maiores que alguns pixels, além de potencialmente dificultar o treinamento devido à alta capacidade da rede (GÉRON, 2019). Um exemplo desse funcionamento pode ser observado na Figura 8. É perceptível que a camada convolucional utiliza informações de apenas uma região da imagem e, em seguida, essas informações são processadas nas camadas subsequentes até chegar a uma ou mais camadas totalmente conectadas, que fazem a previsão ou classificação da rede. Neste exemplo, a rede realiza uma classificação binária para determinar se a imagem contém um cachorro ou não.

Figura 8 – Exemplo de rede convolucional



Fonte: Alzubaidi *et al.* (2021).

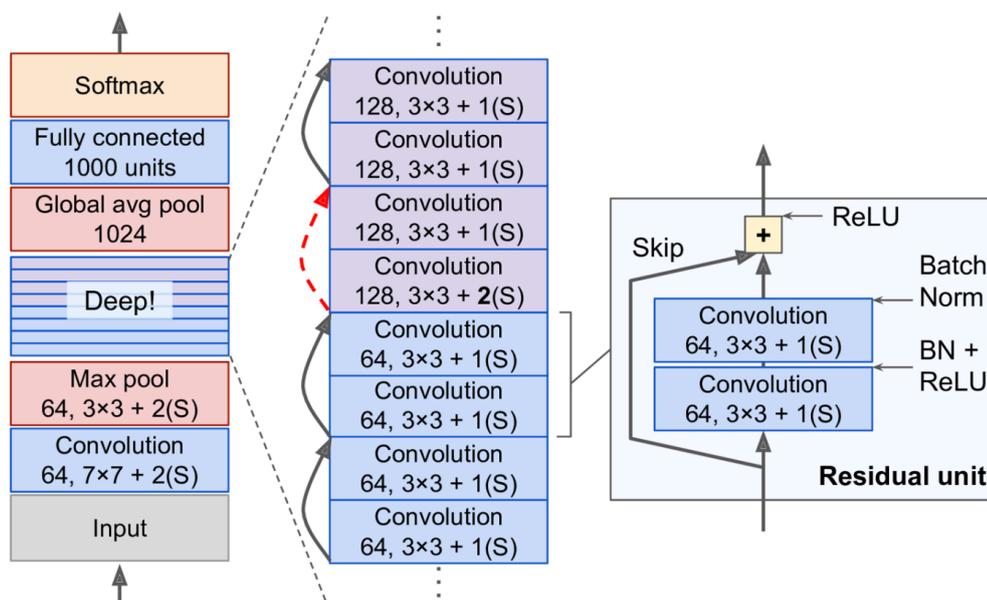
Essas redes geralmente possuem várias camadas nesse estilo, que reduzem progressivamente o tamanho das imagens e convertem as informações anteriores em atributos. No final, esses atributos são aplicados a uma camada densa ou a uma MLP, que combina todos esses dados para encontrar o resultado desejado da rede. No entanto, diferentes arquiteturas podem modificar parte desse funcionamento básico.

Um ponto importante a ser observado nas CNNs é que, diferentemente de outros tipos de redes, é muito complexo e computacionalmente custoso desenvolver novas arquiteturas que obtenham bons resultados. Portanto, para a maioria dos trabalhos, é recomendado utilizar arquiteturas desenvolvidas e pré-treinadas e aplicar

*transfer learning*. Esse processo envolve substituir a última ou as últimas camadas de uma dessas arquiteturas e ajustá-las para o novo problema, retreinando apenas essas camadas ou toda a rede, com foco na adaptação para o novo objetivo do modelo. Isso normalmente resulta em bons resultados e com um custo computacional consideravelmente menor (ALZUBAIDI *et al.*, 2021).

Uma das redes mais utilizadas atualmente é a *Residual Neural Network* (ResNet), ela foi desenvolvida por He *et al.* (2015), e sua arquitetura base é mostrada na Figura 9. Do lado esquerdo, mostra-se a entrada, todo o processo realizado e a saída no final. Na coluna do meio, demonstra-se algumas das convoluções em sequência que são feitas. E na coluna da direita, mostra-se somente um bloco residual, mostrando a passagem pelo bloco ou o pulo da entrada diretamente a saída com *skip* das conexões. Essa possibilidade de passar da entrada diretamente para a saída do bloco é o que permite a ResNet a criar redes mais profundas funcionais. Esse processo também é utilizado em outros tipos de redes, mas isso permitiu redes mais profundas serem treinadas e melhores resultados atingidos para esses modelos convolucionais.

Figura 9 – Arquitetura base da ResNet



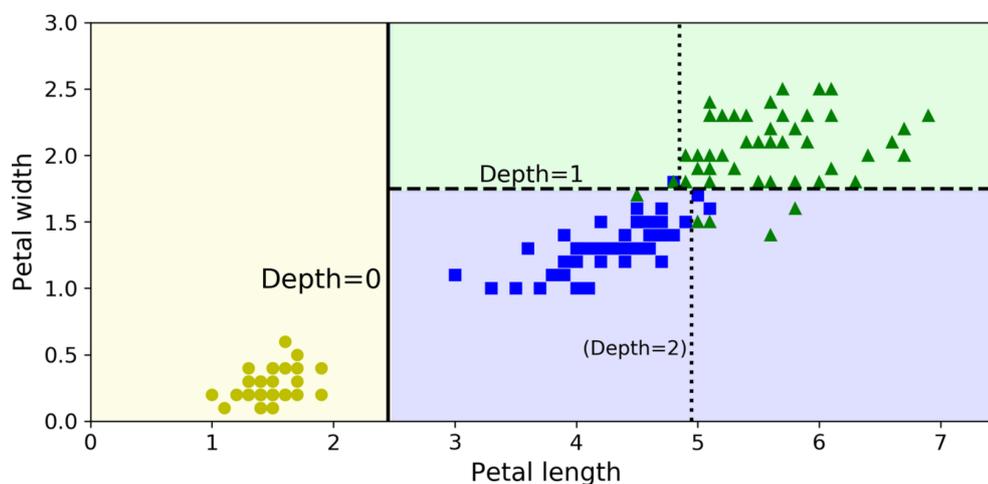
Fonte: Géron (2019).

### 2.1.10 Árvore de decisão

A árvore de decisão é outro tipo de modelo versátil que pode ser utilizado tanto para tarefas de classificação ou regressão. O seu funcionamento é baseado em separar o conjunto de dados em conjuntos menores a partir de divisões com base nos dados numéricos ou categóricos. Um dataset utilizado para comparação ou explicação de modelos de aprendizado de máquina é o dataset Iris. Nele existem 4

atributos e 3 categorias para o tipo de flor que o dado se refere. Esse dataset será utilizado como exemplo para demonstrar o funcionamento da árvore de decisão. Esse exemplo pode ser observado na Figura 10, que divide o dataset Iris por duas de suas variáveis: largura de pétala (eixo y) e comprimento de pétala (eixo x). As cores do fundo representam a classificação dada pelo modelo para uma amostra colocada naquele local. Por outro lado, as cores e formatos das instâncias (círculos, quadrados ou triângulos) representam a espécie real da flor.

Figura 10 – Exemplo de divisão do conjunto de dados a partir de um modelo de árvore de decisão



Fonte: Géron (2019).

Nesse caso, o primeiro nível da árvore de decisão é representada pela linha mais grossa ( $depth=0$ ) e divide o conjunto de dados entre aqueles que possuem larguras de pétalas maior ou menor que 2.45cm. Já o segundo nível, divide o conjunto entre comprimentos de pétalas menor ou maior que 1.75cm ( $depth=1$ ). Essa divisão pode continuar ser cada vez mais precisa aumentando a profundidade da árvore, como a linha pontilhada sugere ( $depth=2$ ). Aumentando a profundidade das árvores pode melhorar a separação dos conjuntos, mas após certo ponto, o modelo começa a apresentar *overfitting* em relação ao conjunto de dados e os resultados no conjunto de validação começam a piorar. Outro ponto interessante em relação à modelos de árvore é que dependendo do formato dos dados a divisão é muito difícil e recomenda-se transformar os dados para outro conjunto de parâmetros. Por exemplo, separar uma função polinomial de grau 1 utilizando esse processo necessita de uma profundidade elevada, enquanto, se esses dados fossem rotacionados (para ficarem paralelos ao eixo x ou y), uma árvore com profundidade 1 já conseguiria separar os conjuntos facilmente (GÉRON, 2019).

### 2.1.11 Métodos combinados

Também é possível combinar um ou mais desses métodos mencionados em um *ensemble*. Essas combinações, em muitos casos, obtêm resultados similares ou melhores do que os modelos individualmente. Alguns modelos já possuem um ensemble embutido, como os modelos de árvore, como Floresta Aleatória ou *Gradient Boosting*.

No caso particular da Floresta Aleatória, ela é definida como uma combinação de múltiplas árvores de decisão em paralelo. Sua previsão é gerada a partir da média de cada uma das árvores para a regressão ou do voto da maioria para classificação. Esse modelo, usualmente, é treinado de forma que cada árvore recebe uma parcela dos dados de treinamento. A intuição em relação a isso é que cada árvore irá trabalhar e tratar melhor uma parte dos dados e com isso a previsão do conjunto será melhorada. Essa proposta busca reduzir o *overfitting* do conjunto como um todo, já que as arvores possuem conjuntos de treinamento levemente diferentes é mais improvável que o *overfitting* continue acontecendo. Com isso, a Floresta Aleatória é ainda hoje um modelo muito capaz, no mínimo, utilizado como base de comparação com outros modelos ou soluções possíveis pra um problema. Além disso, ela também é utilizada como um dos possíveis métodos para calcular a importância dos atributos para um determinado problema (GÉRON, 2019).

Já no caso do *Gradient Boosting*, também são usualmente combinadas árvores de decisão, porém de uma forma sequencial. O processo consiste em adicionar preditores em um conjunto, sendo que cada um busca corrigir os erros dos predecessores. A função objetivo, ou valor a ser predito, por uma nova camada do modelo é a diferença entre o valor real e a previsão realizada pelo modelo ainda sem essa nova camada, ou seja, o erro residual do modelo anterior. Desta forma, cada camada nova vai corrigindo erros cada vez menores nas previsões. De forma similar ao caso da floresta aleatória, existe um ponto ideal para parar antes de ocorrer o *overfitting* nos dados de treinamento. O *Gradient Boosting* também é um modelo com bons resultados entre os modelos não neurais. Por outro lado, ele possui uma dificuldade maior em relação ao treinamento que é que cada nova camada do modelo deve ser treinada após a finalização da última e não paralelizado como no caso da floresta aleatória, aumentando significativamente seu tempo de treinamento (GÉRON, 2019).

## 2.2 GERAÇÃO FOTOVOLTAICA

Existem duas formas de gerar energia a partir da energia solar. Uma delas é através de grandes usinas solares que aquecem fluídos para acionar turbinas (geração elétrica heliotérmica). A outra forma é através do efeito fotovoltaico diretamente. Para tal, foram desenvolvidas células fotovoltaicas que convertem a radiação solar em eletricidade. Ao juntar essas células, é possível montar um painel fotovoltaico que gera

uma certa quantidade de energia elétrica quando exposto à energia solar. A energia gerada dessa forma é instantânea, ou seja, está diretamente relacionada à intensidade da radiação solar no momento, sem carregar uma estabilidade referente ao sistema como ocorre com a heliotérmica.

Os principais fatores relacionados à energia gerada por um painel fotovoltaico são: o nível de irradiância que atinge o painel, a temperatura e as características do próprio painel. Devido a isso e outros fatores que podem alterar a geração como degradação do painel, sujidade, perdas por conversão, é comum dividir a previsão da energia gerada em duas etapas: prever a GHI e, em seguida, prever a potência a partir da GHI e outros parâmetros. A incidência da radiação varia de acordo com o tempo e a localização. Um dos fatores dessa variação é à geometria Sol-Terra, na qual a posição do Sol no céu varia ao longo das estações, assim como a sua distância da Terra. Essa mudança é exemplificada na Figura 11.

Figura 11 – Posição dos hemisférios em relação ao Sol no decorrer do ano



Fonte: Pereira *et al.* (2017).

Além disso, quando a radiação solar atinge a atmosfera, ela é dispersada, podendo seguir diretamente, refletir ou refratar. A irradiância que segue direto do Sol até a superfície, sem desvios, é a Irradiância Direta Normal (*Direct Normal Irradiance* - DNI). A parcela que chega refletida ou retratada é a Irradiância Difusa Horizontal (*Diffuse Horizontal Irradiance* - DHI). Por fim, a Irradiância Global Horizontal (*Global Horizontal Irradiance* - GHI) é uma soma vetorial dessas duas e é normalmente utilizada como ponto de estudo para as previsões, pois a partir dela é possível estimar as irradiâncias em outros ângulos. Esse valor varia de acordo com a localização na Terra, o horário do dia, dia do ano e as condições meteorológicas, que são sistemas que possuem um certo grau de imprevisibilidade. Esses fatores causam a variação da GHI e fazem com que sua previsão não seja trivial.

### 2.3 MODELO PERSISTENTE INTELIGENTE

O modelo SP é o modelo mais comumente utilizado como ponto de referência para comparar os modelos de previsão de irradiância (LIU; JORDAN, 1960). Para realizar a previsão utilizando esse modelo, também é necessário um modelo de céu claro. Esse tipo de modelo busca estimar os valores de irradiância para um local específico considerando um cenário de céu limpo constante, geralmente é estimado com base na localização geográfica desse local e possivelmente outras variáveis auxiliares.

Primeiramente, para realizar a previsão, é necessário estimar o índice de céu claro atual ( $K_t$ ). Esse índice é calculado como a razão entre o valor da irradiância atual e o valor esperado pelo modelo de céu claro ( $K_t(t) = I(t)/I_{CS}(t)$ ).

O modelo SP assume que o índice de céu claro no momento da previsão será o mesmo do momento atual. Isso implica que as condições ambientais permaneçam constantes: se estiver nublado, continuará nublado com a "mesma intensidade"; se estiver com céu claro, essa condição será mantida. Além disso, o modelo persistente inteligente ajusta o valor futuro da GHI com base no próximo valor esperado da irradiância de céu claro. Isso é importante porque, durante a manhã, os valores de irradiância tendem a aumentar até atingir o pico, enquanto à tarde começam a diminuir. Devido à simplicidade do modelo e aos resultados razoáveis, especialmente para previsões de curto prazo de poucos minutos, ele é frequentemente usado como ponto de referência inicial para qualquer modelo que busque prever a irradiância futura. A GHI prevista é dada por

$$SP(t + \Delta t) = I_{CS}(t + \Delta t) \cdot K_t(t) \quad (2)$$

onde  $SP(t + \Delta t)$  é o valor da irradiância prevista pelo modelo persistente inteligente,  $I_{CS}(t + \Delta t)$  é a irradiância de céu claro do momento futuro e  $K_t(t)$  o índice de céu claro atual.

### 2.4 MÉTRICAS DE ERRO

Foram utilizadas para avaliar e comparar os resultados dos modelos diversas métricas comumente utilizadas na literatura (SINGLA; DUHAN; SAROHA, 2021). Primeiramente, o erro médio absoluto (*Mean Absolute Error* - MAE) pode ser calculado por

$$MAE = \frac{1}{n} \sum_{n=1}^n |\hat{y}_i - y_i| \quad (3)$$

sendo  $\hat{y}_i$  os valores preditos,  $y_i$  os valores reais, e  $n$  o número de predições realizadas. Já o RMSE, pode ser calculado por

$$RMSE = \sqrt{\frac{1}{n} \sum_{n=1}^n (\hat{y}_i - y_i)^2} \quad (4)$$

que é calculado a partir das mesmas entradas que o MAE. Além disso, é possível utilizar variações dessas métricas normalizadas, isso facilita para comparar resultados entre conjuntos de dados distintos. Por exemplo, o *Normalized Mean Absolute Error* é definido por  $nMAE = MAE/\bar{I}$  e o *Normalized Root Mean Squared Error* por  $nRMSE = RMSE/\bar{I}$ , onde MAE e RMSE são as métricas sem a normalização e  $\bar{I}$  é a média da irradiância real. Além disso, tem-se o FS, que é uma métrica que facilita observar a melhora ou piora de um resultado em relação a uma referência. Ela é calculada por

$$FS = 1 - \frac{M}{M_{ref}} \quad (5)$$

onde  $FS$  é o *forecast skill* em si e pode ser representado em percentual,  $M$  é a métrica que está sendo avaliada e  $M_{ref}$  é a métrica do modelo de referência ou comparação. Quando o resultado é melhor que a referência, o valor do  $FS$  é positivo, quando é pior, ele é negativo. Esse valor pode ser considerado um "grau de melhora" em relação à referência. O  $M$  e  $M_{ref}$  são erros já calculados a partir de outra métrica, como MAE ( $FS_{MAE}$ ) ou RMSE ( $FS_{RMSE}$ ), por exemplo. Outra métrica comumente utilizada é o erro percentual absoluto médio (*Mean Absolute Percentage Error* - MAPE). Ela pode ser calculada por

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{\hat{y}_i - y_i}{\hat{y}_i} \right| \quad (6)$$

a partir das variáveis já definidas. A última métrica utilizada como comparação nesse trabalho é o  $R^2$  ou coeficiente de determinação. Ele pode ser calculado por

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (7)$$

onde  $R^2$  é a própria métrica em si,  $SS_{res}$  é a soma dos quadrados dos resíduos e  $SS_{tot}$  a soma total dos quadrados.

Dessas métricas, deu-se prioridade à MAE para escolha dos melhores modelos e hiperparâmetros. Ela foi selecionada como referência, pois é menos sensível a desvios (previsões muito erradas) e é mais facilmente interpretada do que outras opções como o RMSE. Outro ponto importante, é que as redes foram treinadas com outras variáveis alvo além da GHI, porém as métricas são calculadas considerando os erros em relação à GHI, ou seja, a variável alvo foi reconvertida para a irradiância para calcular os erros relacionados.

## 2.5 NORMALIZAÇÃO DOS DADOS TABULARES

Na maioria dos casos, é importante realizar a normalização dos dados de entrada do modelo para evitar instabilidades durante o treinamento, devido a diferenças de escala entre os atributos (GÉRON, 2019). Uma forma comum de normalização é a

chamada normalização min-max, que dimensiona os dados de forma que o valor mínimo seja 0 e o valor máximo seja 1 para cada atributo. Essa transformação é realizada pela seguinte função

$$X_{norm} = \frac{X - \min(X)}{\max(X) - \min(X)} \quad (8)$$

onde  $X$  é um vetor com todas as entradas de um atributo,  $\min(X)$  e  $\max(X)$  são o menor e maior valor encontrados nesse vetor, respectivamente, e  $X_{norm}$  é esse vetor escalonado pela normalização min-max. Outra normalização possível é a *z-score*, que faz com que a média do conjunto seja 0 e com variância 1. Ela pode ser calculada pela equação

$$X_{norm} = \frac{X - \mu}{\sigma} \quad (9)$$

onde  $X$  continua sendo o vetor com as entradas e  $X_{norm}$  a saída após a transformação. Além disso,  $\mu$  é a média e  $\sigma$  o desvio padrão da entrada, respectivamente.

### 3 MATERIAIS E MÉTODOS

Nesse capítulo, são explicados os processos, dados e ferramentas utilizados na dissertação.

#### 3.1 ANÁLISE E PRÉ-PROCESSAMENTO DOS DADOS DE ENTRADA

Nessa seção discute-se o conjunto de dados utilizados, quais processamentos foram feitos e como esse conjunto foi computado para esse estudo.

##### 3.1.1 Conjunto de dados

Muitas publicações da área de energia solar possuem datasets próprios e fechados, desta forma a comparação entre trabalhos não é trivial. Fora disso ainda existem grandes instituições que disponibilizam abertamente os dados, nos Estados Unidos o NREL, e no Brasil o Fotovoltaica UFSC e INPE, por exemplo. Esses dados são muito interessantes, porém, muitas vezes ainda é necessário um grande pré-processamento e validação por parte do usuário para fazer a utilização correta desses dados. Entendo essas limitações, em 2019, foi criado o dataset de Folsom. A ideia dos criadores foi realizar boa parte desse pré-processamento e validação e já disponibilizar o dataset o mais próximo possível da utilização, tirando esse ônus para eles ao invés de recair sobre todos que estão utilizando os dados. Essa proposta também busca trazer um dataset onde diversos tipos de trabalhos de solar podem ser comparados, pois o dataset possui dados de uma estação solarimétrica, imagens *all-sky*, imagens de satélite e previsões NWP (PEDRO; LARSON; COIMBRA, 2019). Devido a esse contexto e mais de 80 publicações até a data de escrita desse trabalho, decidiu-se utilizar esse conjunto de dados para essa pesquisa e focar ao máximo em como utilizar e juntar esses dados e nos modelos de aprendizado de máquina, ao invés de trabalhar extensivamente no pré-processamento ou limpeza dos dados.

O conjunto de dados utilizado neste estudo foi coletado por uma estação solarimétrica localizada em Folsom, Califórnia, nas coordenadas 38,642° Norte, 121,148° Oeste, com uma altitude estimada de 100 metros (PEDRO; LARSON; COIMBRA, 2019)<sup>1</sup>. O conjunto de dados consiste em medidas radiométricas e ambientais, bem como imagens *all-sky* para os anos de 2014, 2015 e 2016, com a resolução minuto a minuto. Além disso, também estão disponíveis imagens de satélite e modelos de previsão do tempo para a mesma região. No entanto, para o escopo dessa dissertação, que se concentra na previsão de curtíssimo prazo, esses outros dados não foram utilizados. Foi adotada a divisão dos dados proposta pelos autores, em que o ano de

<sup>1</sup> A altitude foi estimada a partir do Google Earth, uma vez que não estava disponível no artigo original, mas é importante para os modelos de previsão de céu claro.

2016 foi reservado para teste. Além disso, para esse estudo, adotou-se os anos de 2014 e 2015 como conjunto de treinamento e validação, respectivamente.

Nos dados radiométricos estão disponíveis a GHI, DNI e DHI. Já nos dados ambientais estão disponíveis a temperatura, pressão atmosférica e umidade relativa do ar, além da velocidade média e máxima do vento, sua direção predominante e a precipitação. Finalmente, as imagens *all-sky* foram coletadas em RGB com resolução 1536x1536 pixels.

Um ponto importante a ser observado é sobre o instante que estão consideradas as medidas. Para os dados de sensores estavam todos com o tempo de referência exato, ou seja, 00 segundos para todas as instâncias, algo que não ocorre para o caso das imagens. Sobre isso, acredita-se que as medidas são feitas mais de uma vez por minuto e é tirada a média desses valores e colocada no instante 00 segundos. Ou seja, a irradiância para 10:00 é a média entre os valores 10:00:00 e 10:00:59 e mantida no instante 0 segundos. Não tem-se certeza sobre isso, pois não é explicado em detalhes em Pedro, Larson e Coimbra (2019), porém é o procedimento realizado pelo Instituto Nacional de Pesquisas Espaciais (INPE) e pelo Laboratório de Energia Solar da UTFPR (LABENS) em algumas das estações solarimétricas no Brasil. Também não está confirmado se os dados são jogados para a borda anterior ou posterior.

Essa questão de jogar a média dos dados para uma das bordas pode causar problemas de causalidade ou na análise dos dados, como a imagem não segue o mesmo parâmetro de tempo. Desta forma, parte dos erros observados ou discrepâncias podem ser relacionadas às duas séries temporais terem processos distintos e nem sempre estarem coerentes entre si. Esse entendimento sobre a coleta dos dados pode ser explorado em mais detalhes em um trabalho a parte, como a GHI pode variar significativamente em poucos instantes, esses parâmetros sobre o conjunto de dados podem ser muito relevantes para esse ou outros estudos de previsão de curto prazo. Mais detalhes sobre sincronismo não foram avaliados nesse trabalho, pois considerou-se os dados exatamente como estavam disponíveis no dataset de Folsom e que essas avaliações já teriam sido feitas, entretanto esse é um ponto que pode ser melhor explorado em outros trabalhos.

### 3.1.2 Limpeza inicial e imputação dos dados

Inicialmente, juntou-se os dados de irradiância e ambientais dos arquivos *Folsom\_irradiance* e *Folsom\_weather* no mesmo conjunto de dados. Nessa etapa, todos os horários referem-se ao horário na UTC-0.

Esses dados tabulares possuem 12 instâncias onde ao menos um minuto está faltando entre os dados disponíveis. Dessas, 6 instâncias estavam faltando até 6 minutos. Para esses momentos, o GHI e DNI estavam disponíveis e somente as outras variáveis não. Existem métodos para adicionar dados faltantes de solar até 5 dias, como

o *gap filling* e também métodos estatísticos mais complexos para adição de dados faltantes, entretanto, como o foco do trabalho não é na limpeza e pré-processamento dos dados e o número de instâncias faltantes é pequeno, decidiu-se por uma maneira mais trivial para resolver esse problema. Portanto, realizou-se a imputação desses momentos através da imputação linear. Essa operação quebra a causalidade, pois está sendo imputando com base no dado futuro, porém como são poucos instantes e isso não ocorre na variável alvo, decidiu-se fazê-lo para deixar o dataset com menos pontos de indisponibilidade e mais completo no geral. A Tabela 1 mostra os momentos onde os dados foram imputados. Outro ponto importante a ser observado, é que os dados faltantes são dados de 2014, isso é positivo pois não altera os dados em relação ao conjunto de teste. Após todos os filtros e seleção dos dados a serem utilizados, obteve-se 700.399 instâncias, ou seja, esses 14 dados imputados não são significativos no dataset como um todo a ponto de necessitar métodos muito complexos para sua imputação, dado o foco dessa dissertação.

Tabela 1 – Momentos de imputação dos dados de sensores

Início	Fim	#Instâncias
2014-01-16 07:58	2014-01-16 07:58	1
2014-02-19 17:38	2014-02-19 17:38	1
2014-06-04 19:13	2014-06-04 19:13	1
2014-06-04 20:47	2014-06-04 20:49	3
2014-07-23 21:28	2014-07-23 21:32	5
2014-07-24 18:08	2014-07-24 18:10	3

Fonte: Autor.

Os outros 6 momentos não foram imputados, pois um período maior encontrava-se indisponível. A Tabela 2 mostra o início e o final da indisponibilidade dos dados. É interessante observar que em 3 das ocorrências os dados faltantes são múltiplos de 1440 e muitos horários de início e fim são parecidos, o que sugere que esses dias foram retirados devido a análises de qualidade da estação ou algum processo similar. Nesse caso, a quantidade de dados não imputados representa cerca de 3% do conjunto de dados utilizado, mesmo assim, não é uma parcela tão relevante a ponto de focar nesse problema dado o objetivo desse trabalho. Além disso, esse percentual nem está considerando dados diurnos ou noturnos, então, decidiu-se simplificar essa limpeza e seleção de dados e trabalhar com mais detalhes em outras partes dessa pesquisa.

### 3.1.3 Escolha do melhor modelo de céu claro

Em Pedro, Larson e Coimbra (2019), é utilizado o modelo de céu claro de Ineichen e Perez (INEICHEN; PEREZ, 2002), sem confirmação se esse é o que melhor

Tabela 2 – Momentos não imputados

Início	Fim	#Instâncias
2014-01-16 03:18	2014-01-16 03:57	40
2014-01-30 03:18	2014-02-05 07:59	8922
2014-02-13 03:18	2014-02-19 07:59	8922
2015-08-23 08:00	2015-08-24 07:59	1440
2015-12-31 08:00	2016-01-01 07:59	1440
2016-07-13 08:00	2016-07-15 07:59	2880

Fonte: Autor.

se adapta ao conjunto de dados. Tendo isso em vista, utilizou-se a biblioteca PVLib para comparar os modelos de Ineichen e Perez, Haurwitz e *Simplified Solis* (sSolis).

Para fazer essa comparação, primeiramente, estimou-se os valores de céu claro para cada um dos modelos nas coordenadas da estação em todos os momentos em que os dados foram coletados. Utilizando a função *detect\_clear\_sky* também da PVLib, detectaram-se os momentos em que cada um desses modelos considerou os dados da GHI real como céu claro. Como os intervalos em que cada modelo considerou como céu claro eram diferentes, selecionou-se apenas os momentos em que todos os modelos consideraram como céu claro. Em seguida, calculou-se as métricas MAE e RMSE entre cada um dos modelos de céu claro e a irradiância medida de céu claro. Esses resultados podem ser observados na Tabela 3.

Tabela 3 – Comparação dos diferentes modelos de céu claro em relação à GHI de céu claro real

Modelo	#dados	MAE	RMSE
sSolis	375.573	20,9	26,5
Haurwitz	375.573	23,7	29,0
Ineichen e Perez	375.573	27,0	30,9

Fonte: Autor.

Pela Tabela 3, percebe-se que o modelo sSolis está mais próximo dos valores reais da GHI e, portanto, será utilizado como referência para os modelos de céu claro neste trabalho. Esse resultado corrobora com as observações de Antonanzas-Torres *et al.* (2019), que também identificaram o sSolis como um dos melhores modelos de céu claro. Outro ponto relevante a ser destacado é que, mesmo o melhor modelo, ainda não reproduz perfeitamente os dados da GHI. Portanto, aprimoramentos nos modelos de céu claro ou ajustes para melhor adaptar o modelo a esse conjunto de dados podem ser realizados, o que provavelmente resultará em previsões mais precisas como um todo. A significativa divergência entre os modelos também evidencia a importância de testar e avaliar cuidadosamente qual modelo de céu claro melhor se adapta a cada conjunto de dados, especialmente em trabalhos onde a GHI é convertida para o  $K_t$ .

### 3.1.4 Adição de variáveis de entrada

Além das variáveis disponibilizadas pelo conjunto de dados de Folsom, foram estimadas informações extras e adicionadas ao conjunto de entradas do problema. Entre essas informações, estão o zênite, azimute e equação do tempo, que foram estimados utilizando o algoritmo de posição solar (*Solar Position Algorithm - SPA*) do Laboratório Nacional de Energias Renováveis (*National Renewable Energy Laboratory - NREL*) (REDA; ANDREAS, 2008), utilizando as coordenadas geográficas como entrada. Uma vez que a irradiância é proporcional ao cosseno do zênite, o cosseno do zênite também foi incluído como entrada adicional.

Na sequência, foram estimados os valores da GHI, DNI e DHI de céu claro utilizando o modelo sSolis (INEICHEN, 2008). Além disso, foi adicionada uma variável booleana que indica se os dados são diurnos ou noturnos, sendo considerados diurnos quando o zênite é menor que  $85^\circ$ , o mesmo critério utilizado por Pedro, Larson e Coimbra (2019). Utilizando novamente a função *detect\_clear\_sky* da biblioteca PVLib, foi criada outra variável booleana para indicar se, no instante considerado, o céu está classificado como céu claro ou não.

O próximo passo foi estimar o índice de céu claro ( $K_t$ ) dividindo o valor do GHI real pelo valor do GHI de céu claro no respectivo instante. No entanto, no início ou no final do dia, o  $K_t$  pode apresentar valores muito elevados devido à divisão do GHI por um valor muito baixo do GHI de céu claro. Portanto, nos dados noturnos ( $zenite \geq 85^\circ$ ), substituiu-se o valor de  $K_t$  pelo último valor válido ou por 1 quando não estava disponível.

Na próxima etapa, ajustou-se o índice de tempo para o horário local (UTC-7) e criou-se uma variável para representar o dia do ano e outra para representar o minuto do dia. Além disso, foram adicionadas variáveis para os valores futuros de céu claro para GHI, DNI e DHI (15 minutos no futuro) e o valor previsto da GHI 15 minutos no futuro utilizando o modelo persistente inteligente (SP). Essas variáveis são mostradas junto a sua origem na Tabela 4.

### 3.1.5 Imagens

As imagens foram redimensionadas da resolução inicial de 1536x1536 para 256x256 com o objetivo de reduzir o custo computacional dos testes realizados. Esse processo foi realizado utilizando a biblioteca OpenCV e as imagens resultantes foram salvas em formato jpg com qualidade de encoding de 95, que é o padrão da biblioteca.

#### 3.1.5.1 Opções de recortes

Foram realizados testes relacionados aos recortes das imagens. Quatro configurações foram estudadas: sem recorte, recorte retangular e dois recortes elípticos.

Tabela 4 – Dados tabulares

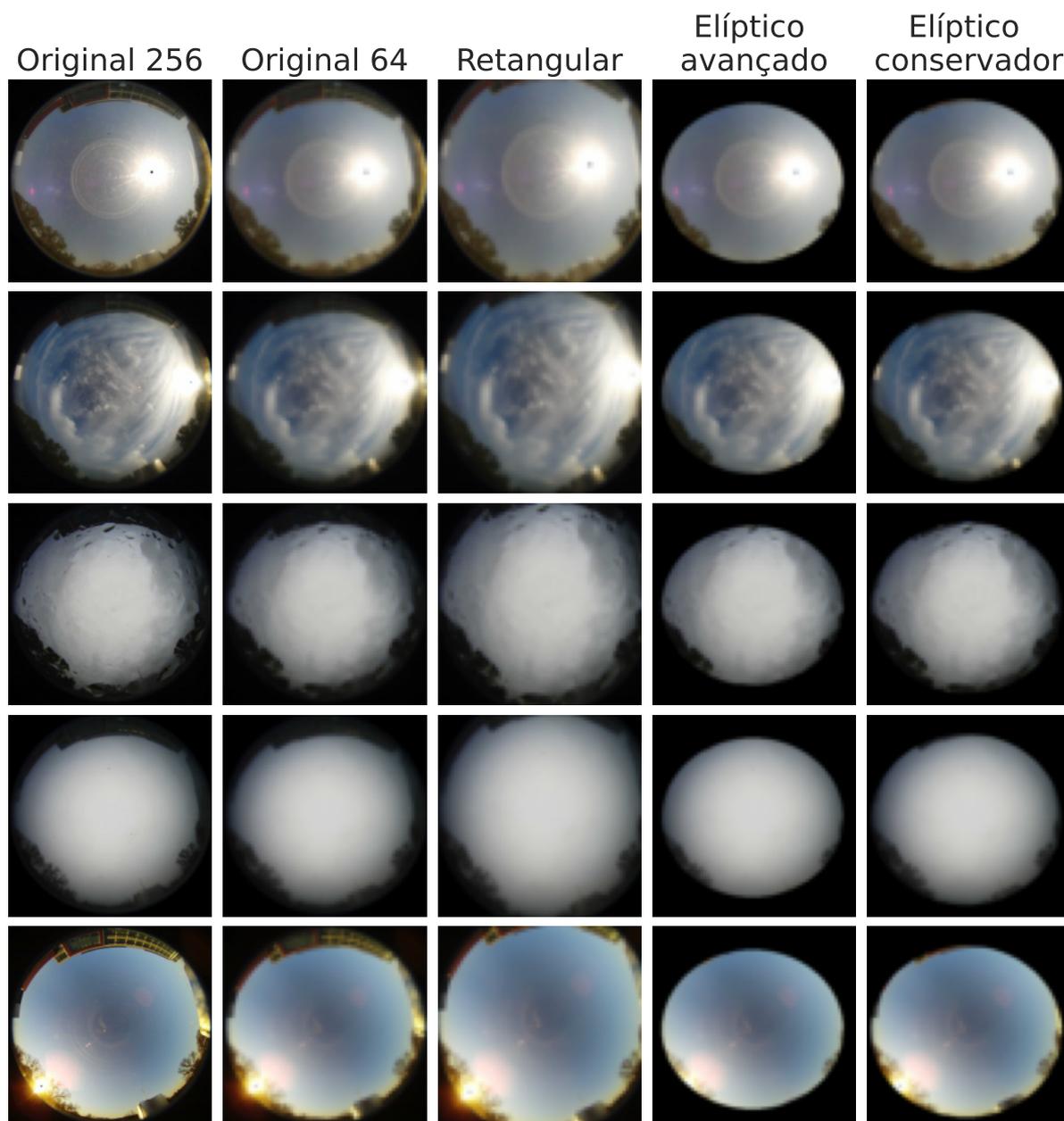
Dado	Origem
GHI	Folsom
DNI	Folsom
DHI	Folsom
Temperatura	Folsom
Umidade	Folsom
Pressão	Folsom
Vel. Vento	Folsom
Dir. Vento	Folsom
Vel. Max. Vento	Folsom
Precipitação	Folsom
Zênite	NREL SPA
cos(Zênite)	Calculado
Azimute	NREL SPA
Equação do tempo	NREL SPA
GHI Céu claro	sSolis
GHI Céu claro	sSolis
DHI Céu claro	sSolis
$K_t$	Calculado
É céu claro	detect_clear_sky
É diurno	Calculado
Dia do ano	Folsom
Minuto do dia	Folsom
Próx. GHI Céu claro	sSolis
Próx. GHI Céu claro	sSolis
Próx. DHI Céu claro	sSolis
SP	Calculado
<b>Total</b>	<b>26</b>

Fonte: Autor.

Todos esses recortes foram aplicados na imagem RGB com tamanho de 256x256 pixels. Após o recorte, a imagem resultante foi possivelmente redimensionada para tamanhos de 64x64 ou 128x128 pixels, dependendo do teste.

O recorte retangular foi realizado utilizando a função *crop* da transformação funcional da biblioteca TorchVision. Foram removidos 5 pixels da borda esquerda, 27 pixels da borda superior, 17 pixels da borda direita e 15 pixels da borda inferior. Além da opção de recorte retangular, também foi realizado o recorte elíptico utilizando funções da biblioteca PIL. Para isso, uma máscara foi desenhada usando a função *Image.new*, com o mesmo tamanho da imagem original de 256x256 pixels. Em seguida, foi desenhada uma elipse com a função *ellipse*, com valores 1 dentro da elipse e 0 fora dela. Para o parâmetro *xy* da função *ellipse*, foram utilizados os valores [(5, 25), (243, 240)] para o recorte conservador e [(12, 30), (238, 230)] para o recorte avançado. Algumas imagens e os possíveis recortes são mostrados na Figura 12, onde os recortes foram feitos em 256x256 pixels e as imagens foram posteriormente convertidas para 64x64 pixels.

Figura 12 – Possíveis recortes das imagens



Fonte: Autor.

### 3.1.5.2 Normalização da imagem

Para a maioria dos testes contendo imagens, foi utilizada a normalização das imagens sugeridas pela *Residual Neural Network* (ResNet). Para tal, foi utilizada a função *normalize* das transformações da biblioteca *TorchVision*. A sugestão é utilizar 0,485, 0,456, 0,406 para os valores médios e 0,229, 0,224, 0,225 para o desvio padrão, para cada canal respectivamente. Quando essa normalização não foi utilizada isso foi especificado no teste em questão.

### 3.1.6 Verificar disponibilidade dos dados

Inicialmente, considerou-se a disponibilidade de pelo menos 15 dados, incluindo o dado atual e os 14 dados anteriores, tanto para os sensores quanto para as imagens. No caso dos sensores, verificou-se se o dado atual e os 14 instantes anteriores estavam disponíveis. Nessas configurações, foram encontradas 1.551.618 instâncias de tempo que atenderam a esse critério de disponibilidade. Outro ponto importante é que apenas foram considerados como momentos de previsão aqueles em que a GHI futura é considerada diurna ( $zenite < 85^\circ$ ). Com essa filtragem, o número de instâncias válidas foi reduzido para 720.673. Para o caso das imagens, verificou-se apenas onde os dados dos sensores já eram válidos. Observou-se a disponibilidade das imagens a partir do instante atual (00 segundos) até 15 minutos e 59 segundos no passado. Considerou-se apenas os momentos em que duas condições foram atingidas:

1. Ao menos 14 imagens disponíveis
2. Tempo máximo entre 2 imagens consecutivas de 2 minutos e 30 segundos

Considerou-se como dados válidos quando ambos os critérios foram atingidos. Com base nesses parâmetros, foram consideradas 700.399 instâncias como válidas.

### 3.1.7 Seleção de imagens a serem utilizadas

Conforme explicado na subseção anterior, foi necessário garantir a disponibilidade de pelo menos 14 imagens. Nesta etapa, foi criado um arquivo adicional contendo o caminho das imagens a serem usadas em cada instante. Quando exatamente 15 imagens estavam disponíveis, o caminho para essas imagens foi salvo na lista de imagens a serem utilizadas. Quando mais de 15 imagens estavam disponíveis, foram selecionadas as 15 mais recentes para serem utilizadas. Por fim, quando menos de 15 imagens estavam disponíveis, a imagem mais próxima disponível (dentro do intervalo selecionado) foi selecionada para dois momentos de tempo.

### 3.1.8 Definição das variáveis alvo

Foram criadas as variáveis alvo com base nos instantes selecionados na Subseção 3.1.6, nos momentos onde as imagens e sensores estavam disponíveis e o dado (futuro) foi considerado diurno, ou seja, com zênite menor que  $85^\circ$ . O período de previsão selecionado foi 15 minutos. Três opções de variável alvo foram criadas: a GHI futura, o  $K_t$  futuro e a diferença entre a GHI futura e a previsão pelo modelo persistente inteligente, que foi definida por

$$SP_{diff} = I - SP \quad (10)$$

onde  $SP_{diff}$  é essa variável alvo,  $I$  é a GHI futura e  $SP$  é a previsão da GHI futura com base no modelo persistente inteligente. Além disso, disponibilizou-se uma variável

auxiliar booleana que marcava se o valor futuro é considerado de céu claro ou não, para complementar a análise dos resultados no conjunto de teste.

### 3.1.9 Normalização dos dados tabulares

Com relação aos dados tabulares, foram definidas três possibilidades. Uma delas é não realizar o escalonamento e manter os dados originais. As outras são, realizar o escalonamento com base na normalização min-max ou na *z-score*, conforme explicado na Subseção 2.5. Para selecionar o conjunto de dados e realizar a normalização, é recomendado considerar apenas os dados de treinamento. Portanto, seleciona-se o conjunto de treinamento e todos os dados que serão utilizados para a previsão (valor atual + N instâncias passadas) em todos os momentos nos quais a previsão será realizada. Com base nesse conjunto de dados selecionado, são encontrados os valores mínimos, máximos, média ou desvio padrão, e aplica-se a normalização coluna a coluna dos atributos, excluindo os dados binários.

### 3.1.10 Análise exploratória das variáveis alvo

Os dados disponíveis e válidos das variáveis alvo, juntamente com sua distribuição por ano, são apresentados na Tabela 5. O percentual de céu claro refere-se aos instantes em que a variável alvo foi considerada como sendo de céu claro. Percebe-se que a quantidade de dados disponíveis é similar entre os anos, com um valor um pouco menor para 2014 e o percentual de céu claro é bem próximo.

Tabela 5 – Disponibilidade dos dados

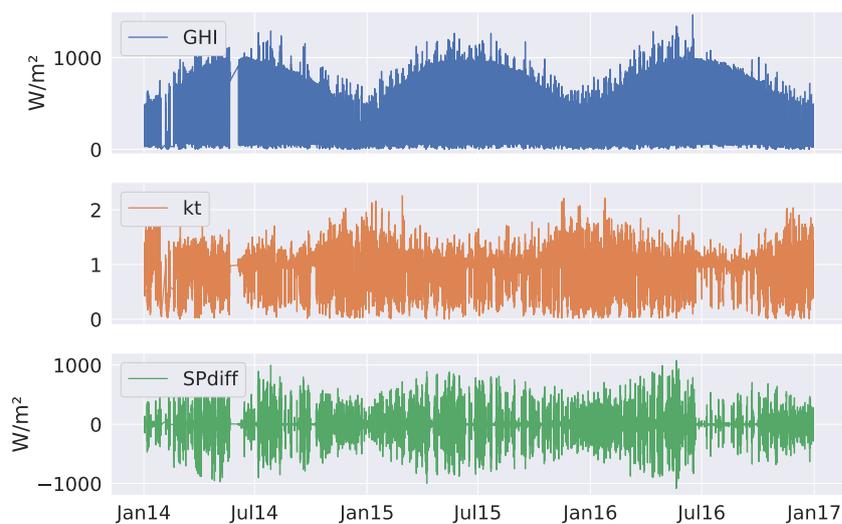
Período	#dados	céu claro
2014	222.032	51,5 %
2015	240.677	51,4 %
2016	237.690	52,2 %
<b>Total</b>	<b>700.399</b>	<b>51,7 %</b>

Fonte: Autor.

Através da Figura 13 pode-se observar que a GHI varia seu máximo no decorrer do ano, tendo os maiores valores no verão da Califórnia. Outro ponto interessante é que em diversas instâncias o maior valor do GHI em um momento muito supera os dias ou momentos próximos, esse efeito pode ser gerado por efeitos de reflexão ou lente causados pela geometria Sol/Nuvens/Terra observado por Moreira Branco e Tiepolo (2020) e descrito em Pereira *et al.* (2017). Já o  $K_t$  e  $SP_{diff}$  variam consideravelmente no decorrer de praticamente todo o período, sendo que as maiores variações do  $SP_{diff}$  parecem também ser no verão, enquanto para o  $K_t$  estão no inverno.

De forma complementar, a Figura 14 demonstra que a GHI está relativamente bem distribuída até cerca de 1000 W/m<sup>2</sup>. Além disso, o  $K_t$  está centralizado próximo

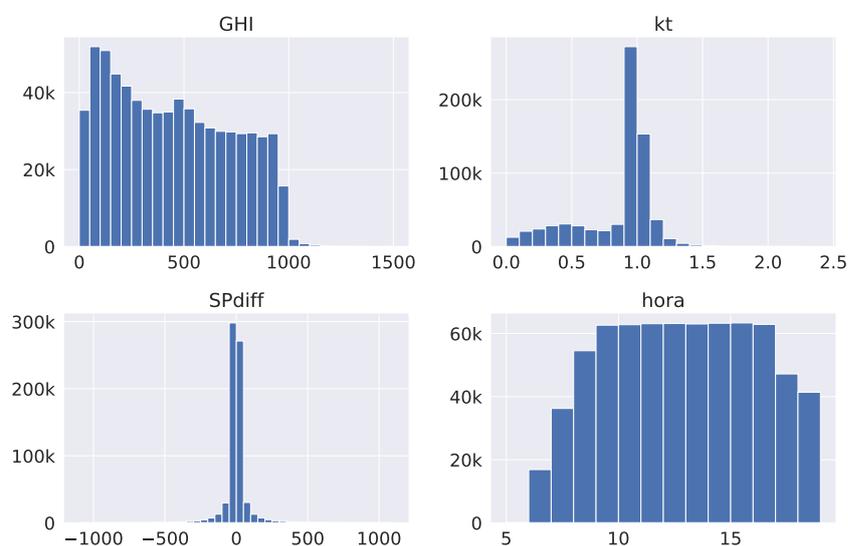
Figura 13 – Variação das variáveis alvo em todo o conjunto de dados



Fonte: Autor.

de 1 com uma cauda maior em valores menores e o  $SP_{diff}$  distribuído ao redor de zero. Por fim, o horário em que esses dados são usados para a previsão demonstra também uma distribuição ao decorrer do dia, com a maioria entre as 10 e 17.

Figura 14 – Histograma das variáveis alvo



Fonte: Autor.

Em adição às informações das figuras, a Tabela 6 descreve a quantidade de dados disponíveis, seus valores médios, mínimos, máximos, desvio padrão e quartis de 25, 50 e 75%. Percebe-se que o valor médio é maior no céu claro e menor no nublado. Além disso, o mínimo, quartis e máximo do céu claro estão divididos mais igualmente. Já no caso do nublado eles estão agrupados nos valores mais baixos,

com exceção do 75% ao máximo onde os valores aumentam muito.

Tabela 6 – Informações sobre a série de dados do GHI

<b>Parâmetro</b>	<b>Completo</b>	<b>Céu Claro</b>	<b>Nublado</b>
média	446,4	547,4	338,3
desvio padrão	283,7	259,7	268,2
min	0,3	23,1	0,3
25%	190,9	334,2	112,8
50%	425,0	552,4	264,5
75%	684,1	773,2	518,0
max	1.466,0	1.017,0	1.466,0

Fonte: Autor.

As informações a respeito do  $K_t$  podem ser observadas na Tabela 7. Percebe-se que para os dados de céu claro o desvio padrão é bem baixo e a média é praticamente 1. Também é observável que o nublado contém a menor média e maior desvio padrão, como esperado.

Tabela 7 – Informações sobre a série de dados do  $K_t$

<b>Parâmetro</b>	<b>Completo</b>	<b>Céu Claro</b>	<b>Nublado</b>
média	0,849	0,999	0,687
desvio padrão	0,291	0,065	0,347
min	0,004	0,264	0,004
25%	0,736	0,959	0,396
50%	0,963	0,985	0,710
75%	1,013	1,026	0,979
max	2,258	1,901	2,258

Fonte: Autor.

Por fim, os dados referentes ao  $SP_{diff}$  estão disponíveis na Tabela 8. Percebe-se que os dados de céu claro tem o menor desvio padrão e menor amplitude de variação, mesmo assim o valor médio fica muito próximo dos outros.

Tabela 8 – Informações sobre a série de dados do  $SP_{diff}$

<b>Parâmetro</b>	<b>Completo</b>	<b>Céu Claro</b>	<b>Nublado</b>
média	0,1	1,3	-1,3
desvio padrão	89,4	18,4	127,2
min	-1.083,3	-214,1	-1.083,3
25%	-5,6	-2,4	-32,8
50%	-0,3	-0,2	-0,8
75%	5,3	2,2	30,2
max	1.075,0	771,7	1.075,0

Fonte: Autor.

### 3.2 EXPERIMENTOS

Foram conduzidos testes com arquiteturas e parâmetros variados para avaliar qual abordagem oferece a melhor capacidade de previsão da irradiância. Por exemplo, os testes estão divididos em três categorias: somente com dados tabulares, somente com imagens *all-sky* ou utilizando uma combinação de ambos. Os testes foram executados sequencialmente, e os melhores parâmetros foram mantidos para os próximos experimentos, a menos que haja uma explicação específica no texto sobre a escolha de parâmetros diferentes.

O treinamento desses modelos tem características de variabilidade entre diferentes *runs* (realizações) e também entre épocas. Para amenizar a variabilidade entre épocas, aplicou-se uma média móvel com período 5, centrada na época em questão. Assim, o resultado para uma época específica foi calculada por

$$X_{MA}(n) = \sum_{k=n-2}^{n+2} X(k) \quad (11)$$

onde,  $X_{MA}(n)$  é a métrica após a média móvel,  $n$  é o número da época e  $X(k)$  é a métrica antes de passar pela transformação. Para as extremidades da série, repetiu-se o primeiro ou último valor, respectivamente, até 5 instâncias estarem disponíveis para o cálculo da média móvel.

Além disso, para reduzir a variabilidade entre *runs*, cada teste foi realizado três vezes para cada conjunto de parâmetros. Isso deve amenizar as aleatoriedades em relação às inicializações dos pesos e ao treinamento dos modelos. Para agregar esses resultados, calculou-se o valor médio, mínimo e máximo entre essas *runs* e essas métricas agregadas que serão reportadas para os experimentos.

Para evitar um custo computacional desnecessário, quando a melhor época (em relação à média móvel) não foi melhor que a fronteira superior do modelo de referência (aquele utilizado como parâmetro de comparação), o teste foi finalizado sem executar a segunda e terceira *run*. Além disso, não foram executadas as três *runs* nos testes dos modelos com dados somente de imagem e nos modelos com dados híbridos devido ao alto custo computacional envolvido. Em alguns testes, o número de *runs* apresentado foi maior do que o estipulado aqui, mas isso está relacionado ao fato de que alguns testes foram executados automaticamente, então esses resultados foram comparados com os demais normalmente.

Para compreender os testes, observou-se gráficos do MAE de treinamento e validação nos casos em que essas informações foram relevantes para a análise dos resultados do teste em questão. Esses gráficos possuem apenas a agregação min-max entre as *runs*, sem a aplicação da média móvel, para facilitar a visualização da variação entre as épocas. A linha contínua no gráfico representa o valor médio das *runs*,

enquanto a área sombreada está entre o valor mínimo e máximo. Esses processos podem ser visualizados exemplificados na Figura 16, apresentada na seção 4.1.1.

A inspeção visual é utilizada para compreender as variações entre os parâmetros. No entanto, para garantir melhor reprodutibilidade e clareza, os melhores modelos foram selecionados, preferencialmente, com base na agregação min-max após a aplicação da média móvel, levando em consideração a melhor época. Nos casos em que o tempo de treinamento difere significativamente entre os parâmetros testados, o tempo relativo em relação ao teste mais longo é mostrado, e isso também foi utilizado para escolher o melhor modelo em questão. Nos momentos onde isso não foi seguido, explica-se no texto os motivos.

Para o treinamento do modelo híbrido, selecionou-se os pesos da melhor época da melhor *run* dos modelos apenas com sensores e dos modelos apenas de imagem, duas vezes para o caso dos sensores, considerando todos os atributos disponíveis ou somente os 10 mais importantes. Também foi feita uma comparação entre a inicialização dos pesos novamente ou a importação desses melhores pesos encontrados nos testes anteriores.

Finalmente, foram selecionados as melhores arquiteturas e hiperparâmetros para cada conjunto de dados de entrada (sensores, imagens e ambos) e retreinadas, para avaliar suas previsões no conjunto de teste. Nessa etapa, esses modelos foram retreinados cinco vezes com o ano de 2014 como treinamento, e cinco vezes com os anos de 2014 e 2015 como treinamento. Ao contrário dos outros testes, como o número de *runs* é maior, utilizou-se a agregação por meio da média e desvio padrão, em vez do mínimo e máximo, como realizado nos experimentos anteriores. Nesse caso, computou-se métricas e análises relacionadas apenas às previsões da última época de treinamento somente, uma vez que os resultados do conjunto de validação não foram calculados nesse experimento. Nessa etapa, esses resultados foram comparados com o modelo persistente inteligente. Além disso, foram observadas suas previsões em 4 dias específicos, dois dias de céu claro e dois dias nublados, um de cada categoria no verão e também no inverno, para comparar a capacidade de previsão em momentos distintos do ano.

### 3.2.1 Ferramentas e bibliotecas utilizadas

O hardware utilizado para a maior parte dos processamentos e atividades deste trabalho foi um servidor com um processador Ryzen 9 5900X de 12 núcleos, 64 GB de RAM e uma placa de vídeo RTX 3090 com 24 GB de VRAM. Além disso, o Google Colab<sup>2</sup> foi utilizado para alguns experimentos e treinamentos.

A linguagem de programação utilizada foi o Python. Para os modelos neurais,

<sup>2</sup> <https://colab.research.google.com/>

foi utilizada a biblioteca PyTorch versão 1.13<sup>3</sup>. Para os modelos não neurais e para o cálculo de algumas das métricas de erro, foi utilizada a biblioteca SciKit Learn<sup>4</sup>. Para o processamento, análise e visualização de dados, foram utilizadas as bibliotecas Pandas<sup>5</sup>, NumPy<sup>6</sup>, Matplotlib<sup>7</sup> e Seaborn<sup>8</sup>. Outras bibliotecas também foram utilizadas neste trabalho, mas serão citadas nas seções específicas em que foram utilizadas. Para auxiliar na organização e processamento dos resultados dos modelos, também foi utilizado o Wandb<sup>9</sup> para registrar e analisar os resultados.

---

<sup>3</sup> <https://pytorch.org/>

<sup>4</sup> <https://scikit-learn.org/>

<sup>5</sup> <https://pandas.pydata.org/>

<sup>6</sup> <https://numpy.org/>

<sup>7</sup> <https://matplotlib.org/>

<sup>8</sup> <https://seaborn.pydata.org/>

<sup>9</sup> <https://wandb.ai/>

## 4 EXPERIMENTOS E RESULTADOS

Nesse capítulo são descritos os experimentos e resultados para cada conjunto de dados e para o retreinamento.

### 4.1 MODELOS UTILIZANDO DADOS DE SENSORES

Nessa etapa, foi trabalhado apenas com os dados tabulares e não foram consideradas as imagens. Foram comparados modelos recorrentes, modelos neurais não recorrentes e não neurais. Diferentes variáveis alvo foram comparadas, bem como distintas arquiteturas e hiperparâmetros, além de parâmetros relacionados ao treinamento, buscando um melhor entendimento da capacidade e limitações dos modelos e dados utilizados. A arquitetura e hiperparâmetros utilizados para o início dos testes dessa Subseção foram os seguintes:

- 50 neurônios ocultos e 1 camada para a LSTM
- 15 instâncias de tempo (atual e 14 passadas)
- otimizador Adam com taxa de aprendizado 0,001
- *batch\_size* 128
- treinamento até 50 épocas, mantendo *checkpoints* dos modelos referentes às melhores épocas em relação ao MAE no conjunto de validação
- embaralhamento do treinamento ativo
- função perda L1
- normalização dos dados de entrada min-max
- sem *dropout* ou regularização L2

#### 4.1.1 Variável alvo

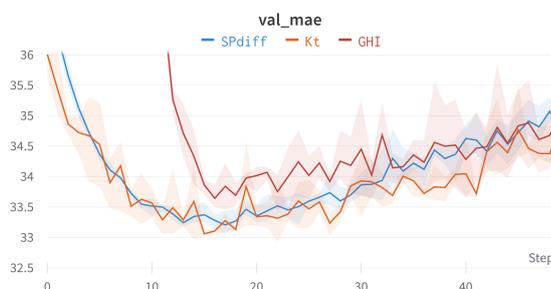
Para esse experimento foram consideradas as três possíveis variáveis alvo definidas anteriormente:  $GHI$ ,  $K_t$  ou  $SP_{diff}$ . As Figuras 15 e 16 mostram o MAE de treinamento e validação, respectivamente. Foram realizadas 3 *runs*, para cada conjunto de hiperparâmetros. Pode-se observar que o melhor valor para a validação é encontrado entre 10 e 20 épocas, após isso os modelos começam a ter *overfitting* e piorar o resultado na validação. Isso é confirmado pela Tabela 9, que mostra que o melhor resultado foi encontrado para a previsão do  $K_t$ , seguido pelo  $SP_{diff}$  e o  $GHI$  diretamente obteve o pior resultado. Com isso em vista, **o  $K_t$  foi mantido como variável alvo para os próximos testes.**

Figura 15 – MAE no conjunto de treinamento para possíveis variáveis alvo para modelos somente com dados de sensores



Fonte: Autor.

Figura 16 – MAE no conjunto de validação para possíveis variáveis alvo para modelos somente com dados de sensores



Fonte: Autor.

Tabela 9 – Métrica MAE para o conjunto de validação utilizando possíveis variáveis alvo para modelos somente com dados de sensores

Experimento	Média	Mínimo	Máximo	Época	#runs
$K_t$	33,23	33,14	33,29	16	3
$SP_{diff}$	33,29	33,25	33,32	15	3
$GHI$	33,80	33,76	33,87	17	3

Fonte: Autor.

#### 4.1.2 Batch size e taxa de aprendizado

Nessa etapa, foi comparado tamanhos de batch de 64, 128, 256 e 512 para as taxas de aprendizado 0,01, 0,001 e 0,0001. Os melhores resultados foram com a taxa de aprendizado  $1e-4$  então esses resultados serão mostrados somente para não poluir a tabela. A Tabela 10 mostra que os resultados entre praticamente todos os *batch\_sizes* é próximo, tendo isso em vista e o tempo relativo de processamento, decidiu-se **alterar os testes para *batch\_size* 256 e manter a taxa de aprendizado em 0,001.**

Tabela 10 – Métrica MAE para o conjunto de validação otimizando *batch\_size* e taxa de aprendizado para modelos somente com dados de sensores

Experimento	Média	Mínimo	Máximo	Época	#runs	% Tempo
<i>batch_size</i> 128 e lr 0,001	33,23	33,14	33,29	16	3	51%
<i>batch_size</i> 256 e lr 0,001	33,25	33,13	33,42	20	3	28%
<i>batch_size</i> 64 e lr 0,001	33,28	33,11	33,53	9	3	100%
<i>batch_size</i> 512 e lr 0,001	33,41	33,24	33,59	24	3	15%

Fonte: Autor.

### 4.1.3 Normalização dos dados tabulares

Nessa etapa, foi comparada a normalização dos dados: não aplicando-a, aplicando normalização min-max ou aplicando a normalização *z-score*. A Tabela 11 mostra melhores resultados para a normalização min-max e que as outras normalizações foram consideravelmente piores, então o teste não foi repetido até 3 *runs*. Com isso, **a normalização min-max foi mantida.**

Tabela 11 – Métrica MAE para o conjunto de validação com diferentes normalizações dos dados tabulares para modelos somente com dados de sensores

Experimento	Média	Mínimo	Máximo	Época	#runs
min-max	33,25	33,13	33,42	20	3
<i>z-score</i>	34,19	34,19	34,19	9	1
sem normalização	35,29	35,29	35,29	42	1

Fonte: Autor.

### 4.1.4 Função perda

Nessa etapa, comparou-se a função perda L1 (MAE) com L2 (MSE). A Tabela 12 demonstra melhores resultados para a perda L1. Esse resultado é coerente com o esperado, pois busca-se otimizar o MAE, portanto usá-la como perda trouxe resultados positivos, mesmo considerando a conversão do  $K_f$  para GHI. Com esses resultados, manteve-se a **função perda L1 para os próximos testes.**

Tabela 12 – Métrica MAE para o conjunto de validação com diferentes funções perda para modelos somente com dados de sensores

Experimento	Média	Mínimo	Máximo	Época	#runs
Perda L1	33,25	33,13	33,42	20	3
Perda L2	36,18	36,18	36,18	12	1

Fonte: Autor.

#### 4.1.5 Aumento da capacidade do modelo e aplicação de regularização L2 ou *dropout*

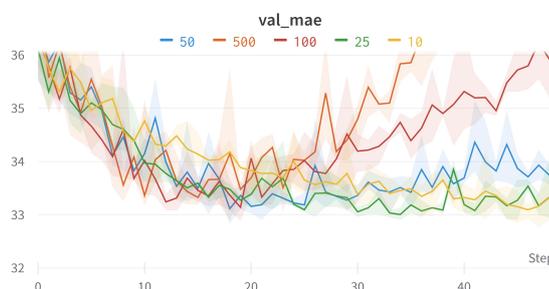
Nessa etapa foram testados modelos LSTM com maior ou menor capacidade com base no número de neurônios ocultos e comparado aplicar regularização L2 ou *dropout*. A Figura 17 mostra, como esperado, que a capacidade dos modelos vai aumentando com um número maior de neurônios ocultos. Entretanto, essa maior capacidade não traz melhoras no conjunto de validação, vide Figura 18 e Tabela 13, inclusive, são observados melhores resultados para 25 neurônios ocultos. Além disso, é interessante observar que os modelos com 500 e 100 neurônios tem um grau mais significativo de *overfitting*, enquanto nos outros isso não é tão observado. Com isso, os modelos com o melhores desempenhos e mais estáveis continuam sendo os mais simples.

Figura 17 – MAE no conjunto de treinamento para modelo somente com dados de sensores com maior capacidade



Fonte: Autor.

Figura 18 – MAE no conjunto de validação para modelo somente com dados de sensores com maior capacidade



Fonte: Autor.

Buscando amenizar esse *overfitting* e comparar um modelo com maior capacidade, testou-se valores diferentes valores de regularização do tipo L2 para o modelo

Tabela 13 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores com maior capacidade

Experimento	Média	Mínimo	Máximo	Época	#runs
25 neurônios	33,09	33,04	33,13	35	3
10 neurônios	33,18	32,87	33,34	47	3
50 neurônios	33,25	33,13	33,42	20	3
100 neurônios	33,39	33,29	33,49	17	3
500 neurônios	33,55	33,30	33,88	15	3

Fonte: Autor.

com 100 neurônios ocultos. Nas tabelas e figuras a seguir também mostrou-se o modelo com 100 neurônios ocultos sem a regularização e o modelo com 25 neurônios que foi o que obteve os melhores resultados.

Essa regularização adiciona mais um termo referente aos pesos do modelo à função perda. Desta forma, quando maior esse valor, mais é evitado o *overfitting*, em troca perde-se parte da capacidade do modelo. A Figura 19 mostra que essa regularização funciona para o treinamento, mesmo assim o modelo com regularização L2 igual a  $1e-5$  ainda obteve melhores resultados de MAE para o conjunto de treinamento que o modelo mais simples. Mesmo assim, a Figura 20 e Tabela 14 mostram que essa melhoria novamente não é repassada à validação.

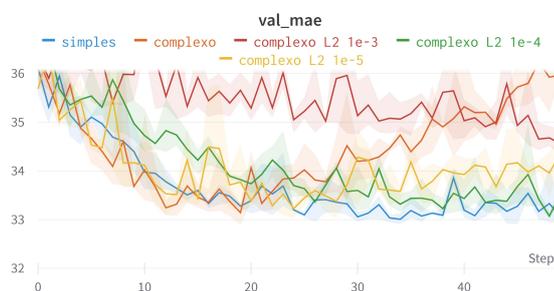
Figura 19 – MAE no conjunto de treinamento para modelo somente com dados de sensores com maior capacidade e com regularização L2



Fonte: Autor.

Outra ideia para reduzir o *overfitting* é adicionar uma camada de *dropout*. Essa camada foi adicionada entre a saída da LSTM e a camada final. A Figura 21 mostra a variação do MAE de treinamento em relação a diferentes valores de *dropout*. Porém, em relação à validação, ainda não há melhora em relação ao modelo mais simples, vide Figura 22 e Tabela 15. O modelo com *dropout* igual a 0,1 atinge o mesmo resultado que o mais simples, mas um pequeno grau de *overfitting* já é observado nas próximas épocas. **Desta forma, manteve-se o modelo simples (25 neurônios e sem regularização L2 ou *dropout*) para os próximos testes.**

Figura 20 – MAE no conjunto de validação para modelo somente com dados de sensores com maior capacidade e com regularização L2



Fonte: Autor.

Tabela 14 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores com maior capacidade e com regularização L2

Experimento	Média	Mínimo	Máximo	Época	#runs
simples	33,09	33,04	33,13	35	3
complexo L2 1e-4	33,36	33,30	33,42	36	3
complexo	33,39	33,29	33,49	17	3
complexo L2 1e-5	33,42	33,34	33,48	24	3
complexo L2 1e-3	34,60	34,43	34,90	49	3

Fonte: Autor.

Figura 21 – MAE no conjunto de treinamento para modelo somente com dados de sensores com maior capacidade e com dropout

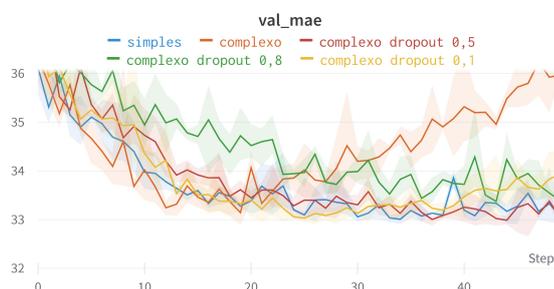


Fonte: Autor.

#### 4.1.6 Número de instâncias passadas

Nessa etapa, comparou-se modificar o número de instâncias de tempo utilizadas pelos modelos para a previsão. A Tabela 16 mostra que aumentar ou diminuir o número de instâncias passadas não melhorou o desempenho no conjunto de validação, inclusive acabou piorando em relação à 15 instâncias. **Portanto, manteve-se o número de instâncias utilizadas em 15 (atual mais 14 passadas) como entrada dos modelos.**

Figura 22 – MAE no conjunto de validação para modelo somente com dados de sensores com maior capacidade e com *dropout*



Fonte: Autor.

Tabela 15 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores com maior capacidade e com *dropout*

Experimento	Média	Mínimo	Máximo	Época	#runs
simples	33,09	33,04	33,13	35	3
complexo <i>dropout</i> 0,1	33,09	33,04	33,13	26	3
complexo <i>dropout</i> 0,5	33,12	32,97	33,28	49	3
complexo	33,39	33,29	33,49	17	3
complexo <i>dropout</i> 0,8	33,49	33,39	33,56	49	3

Fonte: Autor.

Tabela 16 – Métrica MAE para o conjunto de validação comparando modelos somente com dados de sensores variando o número de instâncias passadas

Experimento	Média	Mínimo	Máximo	Época	#runs
15 instâncias	33,09	33,04	33,13	35	3
25 instâncias	33,21	33,12	33,34	46	3
5 instâncias	33,29	33,20	33,35	49	3
100 instâncias	33,31	33,10	33,42	42	3
50 instâncias	33,35	33,16	33,47	34	3

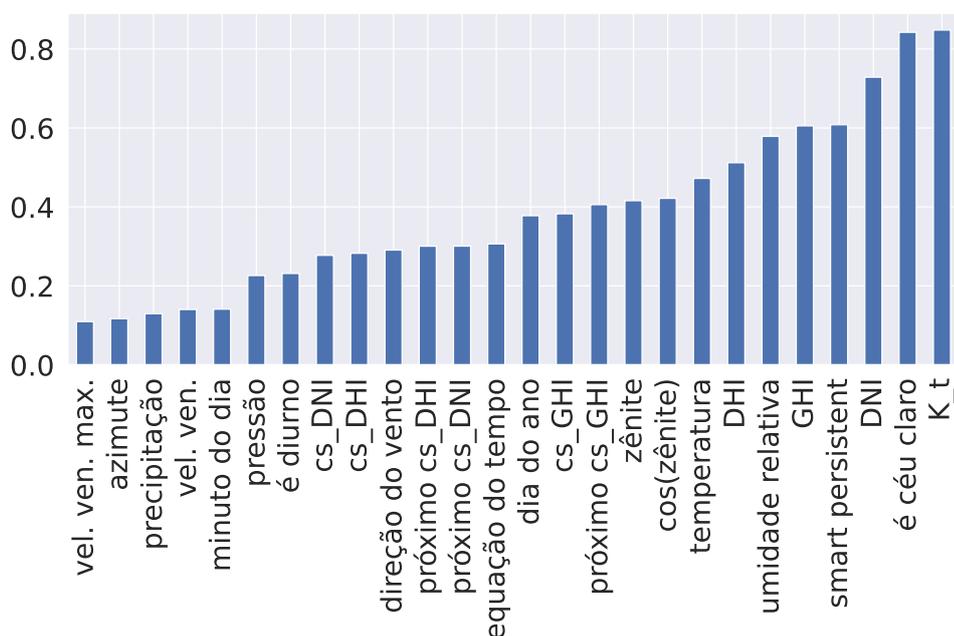
Fonte: Autor.

#### 4.1.7 Somente com atributos mais importantes

Para esse teste, foram comparados modelos contendo todos os atributos disponíveis ou somente os N mais importantes. Para escolher os atributos mais importantes, utilizou-se o coeficiente  $\phi_k$ <sup>1</sup>. Esse método consegue realizar correlações mais relevantes entre variáveis com variáveis categóricas e reverte para a correlação de Pearson em casos onde ela é mais recomendada. Para calcular as correlações não foram utilizados os dados de teste, somente de 2014 e 2015. Os resultados de correlação são mostrados na Figura 23.

A Tabela 17 mostra que somente o resultado para 1 atributo é consideravelmente

<sup>1</sup> <https://phik.readthedocs.io/>

Figura 23 – Correlação  $\phi_k$  entre atributos e a variável alvo  $K_t$ 

Fonte: Autor.

pior que os outros, e para 5, 10 e 15, percebe-se uma melhora em relação a todos os atributos. Desta forma, **manteve-se o modelo com os 10 atributos mais importantes para os próximos testes**, como esse obteve o melhor resultado. Esses atributos são os seguintes:  $K_t$ , é céu claro, DNI, *smart-persistent*, GHI, umidade relativa, DHI, temperatura, cos(zênite), zênite. É interessante de se observar que os próximos valores de céu claro, velocidade do vento, dia do ano e minuto do dia não parecem trazer muitas informações relevantes ao modelo.

Tabela 17 – Métrica MAE para o conjunto de validação utilizando somente os N atributos mais importantes para modelos somente com dados de sensores

Experimento	Média	Mínimo	Máximo	Época	#runs
10 atributos	32,42	32,37	32,50	48	3
5 atributos	32,64	32,50	32,72	49	3
15 atributos	32,80	32,70	32,87	40	3
todos os atributos	33,09	33,04	33,13	35	3
3 atributos	33,42	33,20	33,56	49	3
1 atributo	35,41	35,30	35,49	46	3

Fonte: Autor.

#### 4.1.8 MLP

Nas próximas etapas, buscou-se testar modelos mais simples e colocá-los em comparação com o LSTM, como ponto de referência somente. Deseja-se confirmar se

as previsões referentes às redes recorrentes estão coerentes com o esperado, sem erros ou problemas muito significativos. Para tal, foi montado um modelo MLP com uma camada e foi testado utilizando os 10 atributos mais importantes. Foram testados somente com os dados atuais (input 1), com os atuais e mais 4 passados (input 5) e com o mesmo número de instâncias temporais utilizados pelo modelo LSTM (input 15). Além disso, comparou-se utilizar 10, 50 ou 250 neurônios para o MLP e 250 obteve os melhores resultado então será mostrado nessa comparação.

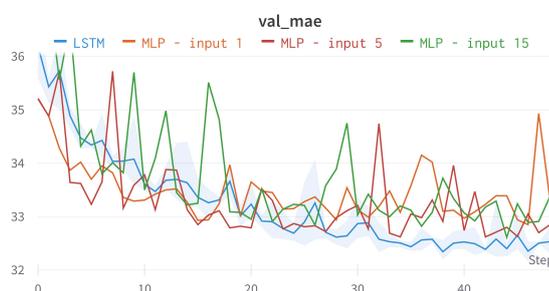
Para o treinamento, percebe-se que a capacidade do LSTM é maior, mas muito próxima em relação aos modelos com 5 e 15 instantes de tempo, vide Figura 24. Já para a validação, observa-se pela Figura 25 e Tabela 18 que o LSTM também tem melhores resultados, mas os modelos MLP estão muito próximos em algumas épocas, oscilando ao redor de 33 de MAE para 5 e 15 instâncias de tempo.

Figura 24 – MAE no conjunto de treinamento para modelos MLP e LSTM somente com dados de sensores



Fonte: Autor.

Figura 25 – MAE no conjunto de validação para modelos MLP e LSTM somente com dados de sensores



Fonte: Autor.

Tabela 18 – Métrica MAE para o conjunto de validação comprando com modelos MLP e LSTM somente com dados de sensores

Experimento	Média	Mínimo	Máximo	Época	#runs
LSTM	32,42	32,37	32,50	48	3
MLP input 5	32,76	32,76	32,76	44	1
MLP input 15	32,96	32,96	32,96	49	1
MLP input 1	33,10	33,10	33,10	40	1

Fonte: Autor.

#### 4.1.9 Modelos não neurais

Nesse etapa, buscou-se complementar os resultados anteriores comparando-os com modelos não neurais também. Novamente, nessa etapa não espera-se encontrar os melhores resultados possíveis com modelos não neurais, mas somente uma comparação para corroborar se as redes neurais estão tendo resultados, no mínimo, comparáveis com outros métodos de previsão. Mesmo se os resultados não forem estritamente melhores, é interessante trabalhar com redes neurais, pois elas possibilitam e concatenar os modelos de imagens com os de sensores.

Como ponto de comparação, decidiu-se testar os modelos de árvore, como Floresta aleatória e *Gradient Boosting*, além da regressão linear com regularização L2 (Ridge). Esses modelos foram testados para diversos conjuntos de hiperparâmetros através das distribuições explicada nas Tabelas 19 e 20. Cada teste foi realizado  $n$  vezes, vide cada tabela, inicialmente sem a normalização dos dados de entrada e depois foram repetidos normalizando os dados de entrada. Além disso, esse teste também foi rodado 3 vezes, para os mesmos inputs testados no experimento anterior (input 1, input 5 e input15).

Tabela 19 – Parâmetros dos modelo Ridge

Parâmetro	Tipo	Ridge
alpha	real	loguniform(1e-10, 1e10)
<b>número de testes</b>	-	<b>10</b>

Fonte: Autor.

Percebe-se pela Tabela 21 que os resultados dos modelos são bons, especialmente para os modelos de árvore. Entretanto, para uma comparação mais completa e justa entre esses modelos seria necessário observá-los no conjunto de testes. Esses testes foram realizados para conferir se os resultados do LSTM estão, ao menos, coerentes com o esperado e isso parece ser o caso.

Tabela 20 – Parâmetros dos modelos não neurais de árvore

Parâmetro	Tipo	Floresta Aleatória	Gradient Boosting
estimadores	int	100	100
max_depth	int	[10,100]	[2,8]
min_samples_leaf	int	[50,200]	[25,200]
max_features	categoria	log2	log2
max_samples	real	[0.25,1]	
loss	categoria	-	erro absoluto
taxa de aprendizado	real	-	loguniform(1e-3,1e1)
subsample	real	-	[0.5,1]
<b>número de testes</b>	-	<b>50</b>	<b>50</b>

Fonte: Autor.

Tabela 21 – Métrica MAE para os conjuntos de treinamento e validação utilizando modelos não neurais somente com dados de sensores

Experimento	Input	Normalização	Treinamento	Validação
Gradient Boosting	5	sem	30,33	33,19
Floresta aleatória	5	sem	30,56	34,15
Regressão Linear Ridge	15	min-max	38,63	39,73

Fonte: Autor.

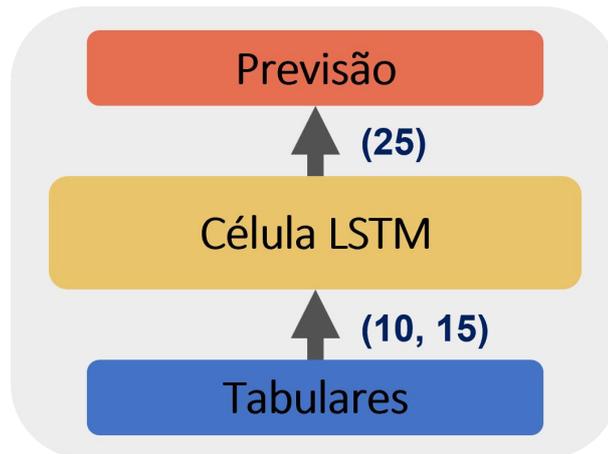
#### 4.1.10 Resumo dos modelos somente com dados de sensores

Nessa seção, foram testadas e otimizadas diversas arquiteturas e suas variações utilizando modelos LSTM. Além disso, utilizou-se modelos MLP e não neurais como uma comparação geral, e confirmou-se que os resultados do LSTM estão coerentes com o esperado. Para uma comparação mais justa entre esses modelos seria importante observar os resultados no conjunto de teste, entretanto como o LSTM é uma arquitetura mais flexível, ele será mantido para as próximas etapas. Pode-se perceber que os modelos com maiores capacidades, normalmente não trazem melhorias no conjunto de validação e, por vezes, observa-se o *overfitting*. Além disso, em muitos testes, os resultados aparentam chegar em um *plateau*, possivelmente relacionado ao limite de previsão a partir dos sensores ou dos modelos estudados aqui. A melhor arquitetura encontrada para modelos utilizando somente dados de sensores é mostrada na Figura 26 e tem os seguintes parâmetros:

- 25 neurônios ocultos e 1 camada para a LSTM
- 15 instâncias de tempo (atual e 14 passadas)
- otimizador Adam com taxa de aprendizado 0,001
- *batch\_size* 256
- treinamento até 50 épocas, mantendo *checkpoints* dos modelos referentes às melhores épocas em relação ao MAE no conjunto de validação

- embaralhamento do treinamento ativo
- função perda L1
- normalização dos dados de entrada min-max
- sem *dropout* ou regularização L2
- 10 atributos mais importantes

Figura 26 – Arquitetura do melhor modelo utilizando somente dados de sensores



Fonte: Autor.

## 4.2 MODELOS UTILIZANDO SOMENTE IMAGENS

Nessa etapa, comparou-se modelos somente com imagens e, assim como no caso anterior, foram comparadas arquiteturas distintas e buscou-se otimizar os parâmetros para obter os melhores modelos possíveis para as arquiteturas testadas e entender as limitações dos modelos e das imagens *all-sky*. Diferente do caso dos modelos somente utilizando dados de sensores, a maioria dos testes de imagens foram feitos somente com 1 *run* por parâmetros, devido ao elevado tempo de processamento. Para o início dos testes dessa Subseção, utilizou-se o seguinte modelo e hiperparâmetros:

- *batch\_size* 256
- ResNet18
- otimizador Adam com taxa de aprendizado 0,001
- treinamento até 50 épocas, mantendo *checkpoints* dos modelos referentes às melhores épocas em relação ao MAE no conjunto de validação
- função perda L1
- embaralhamento dos dados ativo
- imagem com recorte retangular em 256x256 e convertida para 64x64

- imagem normalizada a partir da sugestão da ResNet

### 4.2.1 Variável alvo

De forma, similar ao caso para os modelos utilizando somente dados de sensores, comparou-se os resultados em função das diferentes possíveis variáveis alvo. Entretanto, para esse caso, não utilizou-se o  $SP_{diff}$  como opção de variável alvo. Ela foi retirada, pois traz em si informações a respeito da irradiância atual para poder fazer a previsão do modelo persistente inteligente. Isso estaria disponibilizando de forma indireta essa informação para o modelo e é uma informação indisponível, somente com imagens.

A Figura 27 mostra que o modelo com a variável alvo GHI atinge melhores resultados no conjunto de treinamento. Mesmo assim, ambos atingem MAE de treinamento menor que 10. Já a Figura 28 e Tabela 22 mostram que o  $K_t$  obtém melhores resultados, isso é coerente com o esperado pois a transformação para o  $K_t$  ameniza a complexidade do modelo em focar na situação das nuvens instantâneas e futuras ao invés de trabalhar nisso e também na GHI especificamente para um momento ou outro, que pode variar no decorrer do ano com imagens muito parecidas. Tendo isso em vista, **manteve-se o  $K_t$  como variável alvo para o resto dos testes em relação às imagens.**

Figura 27 – MAE no conjunto de treinamento para possíveis variáveis alvo para modelos somente com o uso de imagens



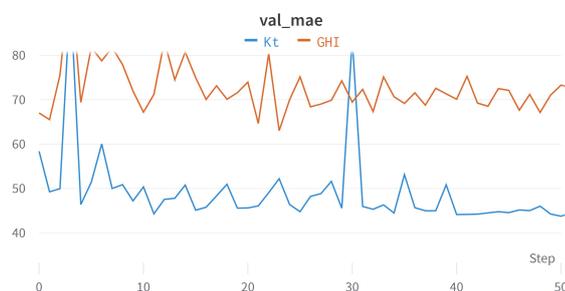
Fonte: Autor.

Tabela 22 – Métrica MAE para o conjunto de validação comparando possíveis variáveis alvo para modelos somente com o uso de imagens

Experimento	Média	Época	#runs
$K_t$	44,35	42	1
GHI	68,41	0	1

Fonte: Autor.

Figura 28 – MAE no conjunto de validação para possíveis variáveis alvo para modelos somente com o uso de imagens



Fonte: Autor.

#### 4.2.2 Batch size e taxa de aprendizado

Nessa etapa foram testados as taxas de aprendizado de 0,01, 0,001 e 0,0001 para os diferentes *batch\_size* de 128, 256 e 512. Os resultados para a taxa de aprendizado 0,0001 foram melhores e serão mostrados e comparados aqui. Primeiramente, a Figura 29 demonstra que reduzindo a taxa de aprendizado para 0,0001 é possível atingir resultados ainda melhores para o conjunto de treinamento.

Figura 29 – MAE no conjunto de treinamento otimizando *batch\_size* e taxa de aprendizado para modelos somente com o uso de imagens



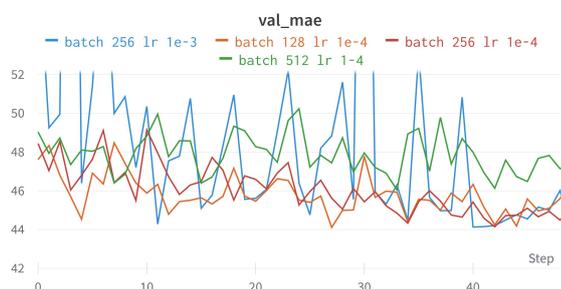
Fonte: Autor.

Com relação à validação, os valores não mudam muito, pela Figura 30 e Tabela 23 observa-se que os resultados são similares, mas para o *batch\_size* 128 e 256 com taxa 1e-4 os resultados são mais estáveis. Tendo isso em vista e o tempo de processamento relativo, **o *batch\_size* 256 e taxa de aprendizado 0,0001 foram utilizados para os próximos testes.**

#### 4.2.3 Função perda

Pode-se observar pela Tabela 24 que a perda L1 traz melhores resultados como função perda. Isso também foi observado para o treinamento dos modelos somente

Figura 30 – MAE no conjunto de validação otimizando *batch\_size* e taxa de aprendizado para modelos somente com o uso de imagens



Fonte: Autor.

Tabela 23 – Métrica MAE para o conjunto de validação otimizando o *batch\_size* e taxa de aprendizado para modelos somente com o uso de imagens

Experimento	Média	Época	#runs	% Tempo
<i>batch_size</i> 256 e lr 0,001	44,35	42	1	79%
<i>batch_size</i> 256 e lr 0,0001	44,66	43	1	84%
<i>batch_size</i> 128 e lr 0,0001	44,80	44	1	100%
<i>batch_size</i> 512 e lr 0,0001	46,77	43	1	75%

Fonte: Autor.

com dados de sensores, então **a perda L1 foi mantida para os próximos testes.**

Tabela 24 – Métrica MAE para o conjunto de validação com diferentes funções perda para modelos somente com o uso de imagens

Experimento	Média	Época	#runs
perda L1	44,66	43	1
perda L2	48,14	41	1

Fonte: Autor.

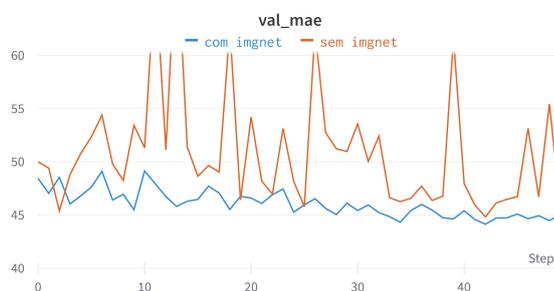
#### 4.2.4 Importar ou não os pesos da ImageNet

Nessa etapa foi testado inicializar ou não o modelo da ResNet18 com os pesos da ImageNet. A Figura 31 e Tabela 25 demonstram que os resultados utilizando os pesos da ImageNet são melhores, além de mais estáveis. **Desta forma, os modelos foram inicializados com os pesos da ImageNet para os próximos testes.**

#### 4.2.5 Recorte da imagem

Nessa etapa comparou-se as opções de recortes explicadas na Subseção 3.1.5.1. Até o momento o recorte retangular havia sido utilizado, nessa etapa este foi comparado com os elípticos e sem o recorte. Para manter a mesma resolução e os

Figura 31 – MAE no conjunto de validação utilizando ou não pesos da ImageNet para modelos somente com o uso de imagens



Fonte: Autor.

Tabela 25 – Métrica MAE para o conjunto de validação utilizando ou não pesos da ImageNet para modelos somente com o uso de imagens

Experimento	Média	Época	#runs
Pesos ImageNet	44,66	43	1
Pesos aleatórios	46,02	43	1

Fonte: Autor.

testes consistentes, os recortes foram feitos na imagem em 256x256 e ela foi então convertida para 64x64. A Tabela 26 demonstra melhores resultados para o recorte elíptico avançado, mesmo que não muito diferentes dos outros. **Com isso, o recorte elíptico mais completo foi mantido para os próximos testes.**

Tabela 26 – Métrica MAE para o conjunto de validação com diferentes recortes das imagens para modelos somente com o uso de imagens

Experimento	Média	Época	#runs
Recorte elíptico avançado	43,27	49	1
Recorte elíptico conservador	44,03	41	1
Sem recorte	44,42	40	1
Recorte retangular	44,66	43	1

Fonte: Autor.

#### 4.2.6 Tamanho da imagem

A Figura 32 mostra os resultados para o treinamento das imagens 64x64, 128x128 e 256x256. Percebe-se que todos os modelos obtêm valores baixos de MAE para o treinamento e o modelo com a imagem 64 apresenta menor capacidade que os outros. Em relação à validação, o caso é alterado, mostrando que o modelo com a menor imagem teve melhores resultados, além de resultados mais estáveis e com menor custo computacional, vide Figura 33 e Tabela 27. Isso demonstra que talvez

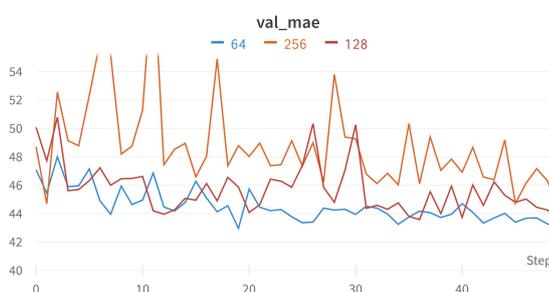
as informações a mais disponíveis não estão sendo significativas para a previsão dos modelos. **Com isso, manteve-se a imagem em 64x64.**

Figura 32 – MAE no conjunto de treinamento para diferentes tamanhos das imagens como entrada dos modelos somente com o uso de imagens



Fonte: Autor.

Figura 33 – MAE no conjunto de validação para diferentes tamanhos das imagens como entrada dos modelos somente com o uso de imagens



Fonte: Autor.

Tabela 27 – Métrica MAE para o conjunto de validação com diferentes tamanhos das imagens para modelos somente com o uso de imagens

Experimento	Média	Época	#runs	Tempo
64x64	43,27	49	1	22%
128x128	43,94	49	1	43%
256x256	45,43	49	1	100%

Fonte: Autor.

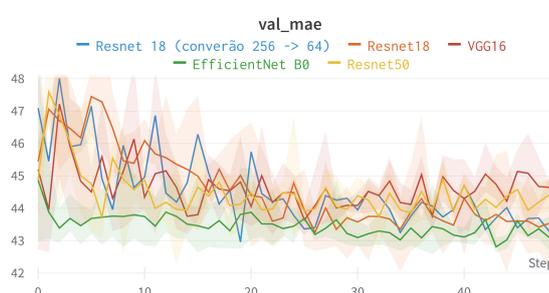
#### 4.2.7 Possíveis arquiteturas

Nessa etapa, comparou-se os resultados entre diferentes arquiteturas de CNNs. Nesse teste, não estava sendo trivial separar os resultados das diferentes arquiteturas, portanto, foi decido rodar 3 vezes para cada conjuntos de parâmetros para separar

melhor os resultados entre as arquiteturas. Existem duas ResNet18, pois a primeira era do teste comparando o melhor recorte e isso foi feito em 256x256 e depois convertido para 64x64, alterando muito o tempo de treinamento.

A Figura 34 e Tabela 28 demonstram que as diversas arquiteturas tem resultados muito próximos entre si para a validação, sendo parte dessa variação possivelmente variações *run-to-run* ou limitações da previsão a partir das imagens. Tendo em vista essa performance similar e o tempo relativo de processamento, **manteve-se a ResNet18** devido à sua performance comparável e menor custo computacional.

Figura 34 – MAE no conjunto de validação para modelos com diferentes arquiteturas CNN para modelos somente com o uso de imagens



Fonte: Autor.

Tabela 28 – Métrica MAE para o conjunto de validação para modelos com diferentes arquiteturas CNN para modelos somente com o uso de imagens

Experimento	Média	Mínimo	Máximo	Época	#runs	Tempo
EfficientNet B0	43,16	42,79	43,57	49	3	45%
ResNet 18 256	43,27	43,27	43,27	49	1	56%
ResNet 18	43,36	42,37	44,56	49	3	22%
ResNet 50	44,05	43,65	44,34	33	3	61%
VGG 16	44,09	43,66	44,87	27	3	100%

Fonte: Autor.

#### 4.2.8 Normalização das imagens

Nessa etapa, comparou-se utilizar ou não a normalização da imagem sugerida pela ResNet. A Tabela 29 demonstra que os resultados são muito próximos, mas melhores utilizando a normalização. Buscando manter a recomendação da ResNet, com resultados um pouco melhores e devido ao baixo custo computacional, a **normalização foi mantida para os próximos testes**.

Tabela 29 – Métrica MAE para o conjunto de validação com ou sem normalização da imagem para modelos somente com o uso de imagens

Experimento	Média	Mínimo	Máximo	Época	#runs
Com normalização	43,36	42,37	44,56	49	3
Sem normalização	43,46	43,46	43,46	32	1

Fonte: Autor.

#### 4.2.9 Diferença entre imagem atual e anterior

Nessa etapa, buscou-se explorar informações em relação à mudança das condições do céu. Utilizou-se a imagem atual subtraída pela imagem anterior como entrada do modelo. Outro processamento realizado foi ajustar os parâmetros da imagem para valores positivos. Isso foi feito após a conversão de imagem para Tensor. Nesse momento, cada pixel da imagem é representada por um número real no intervalo  $[0,1]$ . Cada imagem tem esse intervalo, portanto, o intervalo após a subtração é  $[-1,1]$ . Para manter os valores positivos, como espera a ResNet, foi feita a conversão somando 1 e dividindo por 2, desta forma o intervalo após a subtração será novamente  $[0,1]$ . Após essa conversão, a diferença das imagens foi aplicada a melhor rede e hiperparâmetros até o momento como entrada do modelo. Rodou-se o teste 2 vezes, uma vez utilizando a normalização da ResNet e outra sem ela.

Pode-se observar pela Tabela 30 que somente a imagem atual obteve melhores resultados que ambos os testes considerando a diferença das imagens. Isso demonstra que nesse caso essa variação não auxiliou o modelo a obter melhores informações para realizar a previsão. Desta forma, **manteve-se somente a imagem atual para os próximos testes como modelo de referência.**

Tabela 30 – Métrica MAE para o conjunto de validação comparando modelos que utilizam somente imagem atual versus diferença entre imagem atual e anterior

Experimento	Média	Mínimo	Máximo	Época	#runs
Somente imagem atual	43,36	42,37	44,56	49	3
Atual - Anterior não normalizada	45,13	45,13	45,13	6	1
Atual - Anterior normalizada	45,40	45,40	45,40	8	1

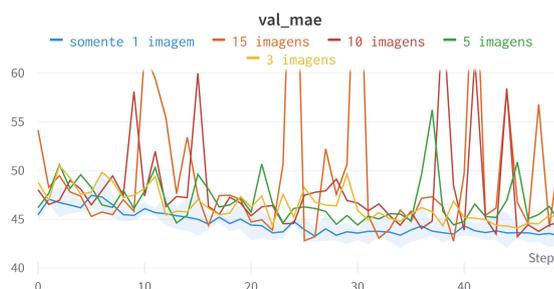
Fonte: Autor.

#### 4.2.10 Múltiplas imagens

Seguindo estudando possibilidades para adição de mais informações temporais aos dados da imagem, criou-se um modelo que passa N imagens na ResNet e concatena os *encodings* dessas imagens e as aplica na camada final. Nesse modelo, a mesma ResNet é aplicada a todas as imagens e tanto ela quanto a camada de saída estão liberadas para o treinamento. A Figura 35 e Tabela 31 demonstram que somente

1 imagem continua obtendo os melhores resultados, além obter respostas mais estáveis entre épocas. Desta forma, **manteve-se somente 1 imagem para os próximos testes.**

Figura 35 – MAE no conjunto de validação para modelos com somente uma imagem ou múltiplas imagens como entrada



Fonte: Autor.

Tabela 31 – Métrica MAE para o conjunto de validação comparando modelos com somente uma imagem ou múltiplas imagens como entrada

Experimento	Média	Mínimo	Máximo	Época	#runs
1 imagem (referência)	43,36	42,37	44,56	49	3
10 imagens	44,10	44,10	44,10	47	1
3 imagens	44,38	44,38	44,38	45	1
15 imagens	44,62	44,62	44,62	33	1
5 imagens	44,92	44,92	44,92	30	1

Fonte: Autor.

#### 4.2.11 Data augmentation

Observa-se que em muitos dos testes, tem-se uma grande capacidade do modelo, devido a sua resposta nos dados de treinamento, mas isso não reflete para os dados de validação. Nessa etapa, buscou-se comparar com diferentes aplicações de *data augmentation (DA)*.

Nessa etapa, foram estudadas algumas possíveis transformações nos dados e comparou-se os melhores resultados obtidos. As transformações estudadas foram variar parâmetros relacionados a cor, parâmetros de nitidez e aplicar diferentes níveis de *blur*. Outras transformações não foram aplicadas, pois entende-se que não fazem sentido para esse problema. Virar a imagem na horizontal ou vertical, rotações, zoom ou recortes aleatórios, irão alterar drasticamente a posição do sol e nuvens no céu e entende-se que essa é uma informação útil e não deve ser alterada. Os processamentos utilizaram transformações de imagem do Pytorch e PytorchVision.

Aqui comparou-se os melhores resultados obtidos para cada transformação e a combinação entre elas. Para a cor, variou-se separadamente o brilho, contraste, saturação e hue utilizando a função ColorJitter com os parâmetros 0,1, 0,3, 0,5, 0,9. O melhor resultado encontrado para o brilho e hue foi com 0,1, para a saturação 0,3 e com contraste 0,5. Então, testou-se com todos esses em simultâneo, porém não obteve-se melhores resultados, desta forma o melhor nessa etapa foi somente com a saturação em 0,3.

Na sequência, testou-se alterar a nitidez utilizando a função Sharpness com os valores 0, 0,5, 2, 5 e 10. A melhor variação foi com Sharpness 10. Também, testou-se diferentes valores de *blur* a partir da função GaussianBlur utilizando *kernel\_size* 3, 5 e 7 com *blur* de 0,1, 0,3, 0,5, 1, 3, 5. Para todos os Kernels, o blur 3 obteve o melhor resultado e entre eles o *kernel\_size* 5 foi o que obteve a melhor performance.

Os testes individuais foram realizados com 1 rodada por conjuntos de parâmetros, porém o combinado foi rodado 3 vezes para melhor comparar a variação entre aplicar ou não o DA. A Tabela 32 demonstra que a combinação dos DA obtiveram melhores resultados que cada transformação separadamente ou não utilizar o DA. Com isso, **manteve-se essa combinação de DA para os próximos testes** (ColorJitter com saturação 0,3, Sharpness com valor 10, GaussianBlur com *kernel\_size* 5 e valor 3).

Tabela 32 – Métrica MAE para o conjunto de validação utilizando ou não diferentes DA para o treinamento de modelos somente com o uso de imagens

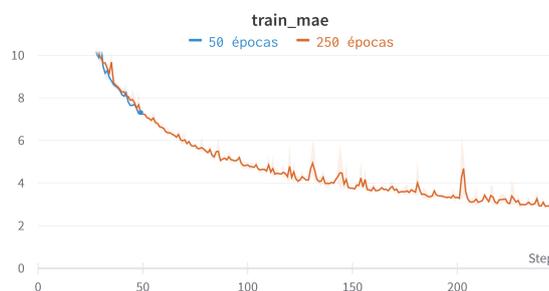
Experimento	Média	Mínimo	Máximo	Época	#runs
Todas as variações	41,80	41,73	41,89	43	3
Somente blur	42,62	42,62	42,62	45	1
Somente saturação	42,79	42,79	42,79	42	1
Somente shapness	42,88	42,88	42,88	36	1
Sem augmentação	43,36	42,37	44,56	49	3

Fonte: Autor.

#### 4.2.12 Treinar por mais épocas

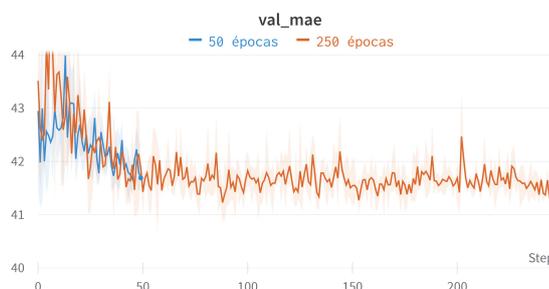
Nesse teste, buscou-se comparar se o modelo ainda pode obter melhores resultados treinando por mais épocas. Percebe-se pela Figura 36 que o resultado no conjunto de treinamento continua melhorando após 50 épocas. Por outro lado, através da Figura 37 e Tabela 33, observa-se que os resultados para o conjunto de validação não melhoram de uma forma consistente e significativa. Optou-se por **manter o número de épocas máximo como 50**, devido ao menor custo computacional e resultados similares.

Figura 36 – MAE no conjunto de treinamento para modelos somente com o uso de imagens treinados por um número maior de épocas



Fonte: Autor.

Figura 37 – MAE no conjunto de validação para modelos somente com o uso de imagens treinados por um número maior de épocas



Fonte: Autor.

Tabela 33 – Métrica MAE para o conjunto de validação para modelos somente utilizando imagens treinados por um número maior de épocas

Experimento	Média	Mínimo	Máximo	Época	#runs	Tempo
Mais épocas	41,43	41,04	41,79	88	3	100%
Menos épocas	41,80	41,73	41,89	43	3	19,7%

Fonte: Autor.

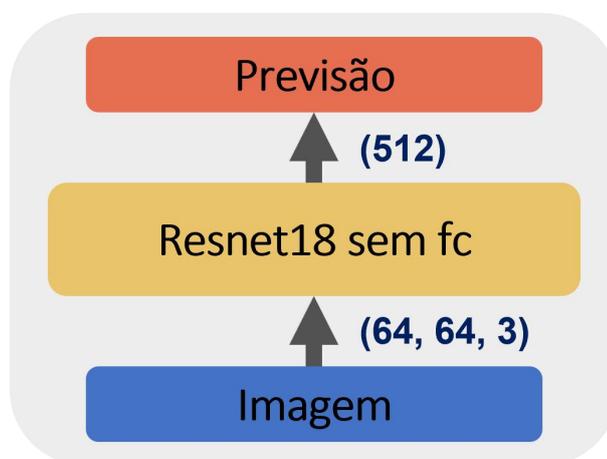
#### 4.2.13 Resumo dos modelos de imagem

De forma similar ao caso dos testes dos modelos somente utilizando dados de sensores, foram testadas diversas arquiteturas e hiperparâmetros. Nesse caso, em alguns testes observou-se melhoras mais significativas, como na escolha da variável alvo, na otimização do *batch\_size* e taxa de aprendizado e aplicando o *data augmentation*. Também observou-se novamente, que o L1 como perda obteve melhores resultados. O recorte também mostrou melhorias. O aumento do tamanho da imagem, trouxe piora nos resultados de validação, contrário ao esperado. As arquiteturas distintas demonstraram resultados distintos, mas nada com variações muito significativas em

desempenho, porém com piora de performance. Por fim, treinar mais épocas também mostrou uma diferença pequena em relação ao caso limitado de 50 épocas, porém com um custo mais elevado. A melhor arquitetura somente para imagens encontrada é mostrada na Figura 38 e os parâmetros utilizados foram:

- *batch\_size* 256
- ResNet18
- otimizador Adam com taxa de aprendizado 0,0001
- treinamento até 50 épocas, mantendo *checkpoints* dos modelos referentes às melhores épocas em relação ao MAE no conjunto de validação
- função perda L1
- embaralhamento dos dados ativo
- imagem recortada com recorte elíptico avançada em 256x256 e convertida para 64x64
- imagem normalizada a partir da sugestão da ResNet
- Utilizando DA

Figura 38 – Arquitetura do melhor modelo somente utilizando imagens



Fonte: Autor.

Percebe-se nesses testes, situações similares que para o caso dos modelos somente com dados de sensores. Em muitos casos a maior capacidade do modelo não traz melhoria dos resultados no dataset de validação. Isso, como no caso anterior, pode estar relacionado a um possível limite de performance atingido para os modelos somente com imagens ou a necessidade de modelos e sistemas mais complexos que os considerados nesse trabalho.

### 4.3 MODELOS COM DADOS HÍBRIDOS

Similar aos testes anteriores, foram comparadas diferentes arquiteturas e otimizados os parâmetros para encontrar o melhor modelo possível. Para essa etapa, utilizou-se os melhores modelos encontrados separadamente para modelos somente utilizando dados de sensores e imagens, concatenou-se esses modelos sem a camada de saída e considerou-se a adição ou não de uma camada Densa após a concatenação e antes da camada de saída. Para as análises dos resultados, também foram considerados os resultados dos melhores modelos utilizando somente imagem e somente sensores como comparação.

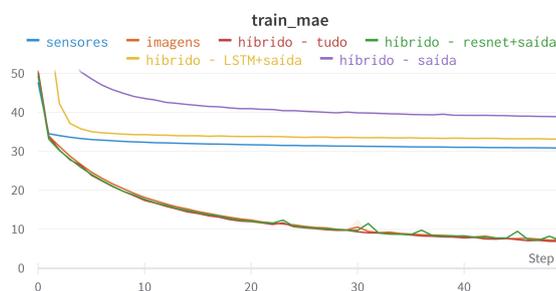
Os testes foram feitos considerando o treinamento somente da camada de saída, da camada de saída e modelo de imagem, da camada de saída e modelo somente com dados de sensores ou de tudo ao mesmo tempo. Além disso, similar ao caso das imagens, a maior parte dos testes foi rodada somente 1 vez por conjunto de hiperparâmetros, devido ao custo computacional. Para esses testes os parâmetros iniciais foram definidos como:

- *batch\_size* 256
- otimizador Adam com taxa de aprendizado 0,0001
- treinamento até 50 épocas, mantendo *checkpoints* dos modelos referentes às melhores épocas em relação ao MAE no conjunto de validação
- função perda L1
- embaralhamento dos dados ativo
- imagem recortada com recorte elíptico avançada em 256x256 e convertida para 64x64
- imagem normalizada a partir da sugestão da ResNet
- utilizando DA

#### 4.3.1 Sem importar pesos dos outros testes

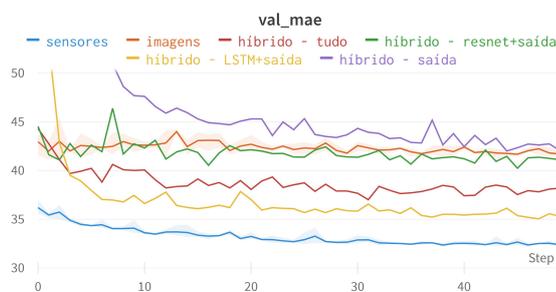
Nessa etapa, utilizou-se somente a melhor arquitetura encontrada, sem importar os pesos dos treinamentos anteriores. A Figura 39 demonstra as diferentes capacidades dos modelos, em todos onde a ResNet é treinada, apresentam valores de MAE de treinamento abaixo de 10. Percebe-se pela Figura 40 e Tabela 34 que os resultados somente do modelo somente com dados de sensores são os melhores. Isso pode ser causado por uma dificuldade em aprender treinando ambos os modelos juntos.

Figura 39 – MAE para o conjunto de treinamento para modelos híbridos sem importar pesos dos melhores modelos somente com uso de sensores ou imagens



Fonte: Autor.

Figura 40 – MAE para o conjunto de validação para modelos híbridos sem importar pesos dos melhores modelos somente com uso de sensores ou imagens



Fonte: Autor.

Tabela 34 – Métrica MAE para o conjunto de validação para modelos híbrido sem importar pesos dos melhores modelos somente com uso de sensores ou imagens

Experimento - treinável	Média	Mínimo	Máximo	Época	#runs
sensores	32,42	32,37	32,50	48	3
híbrido - LSTM+saída	35,27	35,27	35,27	48	1
híbrido - tudo	37,73	37,73	37,73	32	1
híbrido - ResNet+saída	41,04	41,04	41,04	47	1
imagens	41,80	41,73	41,89	43	3
híbrido - saída	42,28	42,28	42,28	49	1

Fonte: Autor.

### 4.3.2 Importando melhores pesos para ambos

Nessa etapa, buscou-se importar os pesos das melhores épocas das melhores runs encontradas nos modelos utilizando somente dados de sensores e imagens e iniciar o treinamento a partir dessa etapa. Pode-se observar pela Figura 41 que todos os modelos além do modelo somente com dados de sensores atingiram valores de MAE de treinamento abaixo de 10. Entretanto, a Figura 42 e Tabela 35 demonstram que isso

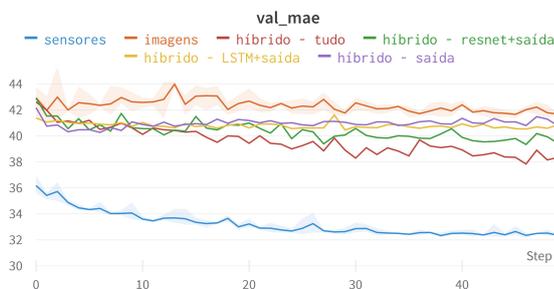
não é repassado para a validação, na verdade, todos os novos testes se aproximam dos resultados do modelo somente com imagem e não conseguem resultados nem próximos aos dos modelos somente com sensores. Acredita-se que os resultados referentes ao modelo somente com imagem já trazem um grau de *overfitting* e isso acaba dificultando o modelo aprender com os dados dos modelos somente com sensores ou uma combinação deles.

Figura 41 – MAE para o conjunto de treinamento para modelos híbridos importando pesos dos melhores modelos somente com uso de sensores e imagens



Fonte: Autor.

Figura 42 – MAE para o conjunto de validação para modelos híbridos importando pesos dos melhores modelos somente com uso de sensores e imagens



Fonte: Autor.

### 4.3.3 Importando somente pesos da LSTM

Buscando evitar a situação do caso anterior, importou-se somente os pesos do modelo LSTM e os pesos da ResNet foram inicializados com os da ImageNet. A Figura 43 e Tabela 36 demonstram resultados comparáveis com os do modelo somente com sensores para os modelos nos quais a ResNet não estava liberada para treinamento. Isso corrobora com o caso anterior, demonstrando uma dificuldade em treinar o conjunto todo para obter melhores resultados. Mesmo nesses casos, o melhor

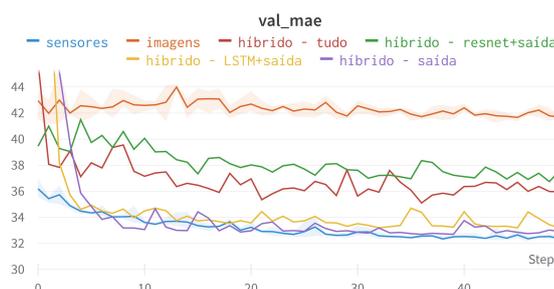
Tabela 35 – Métrica MAE para o conjunto de validação para modelos híbridos importando pesos dos melhores modelos somente com uso de sensores e imagens

Experimento - treinável	Média	Mínimo	Máximo	Época	#runs
sensores	32,42	32,37	32,50	48	3
híbrido - tudo	38,33	38,33	38,33	47	1
híbrido - ResNet+saída	39,59	39,59	39,59	44	1
híbrido - saída	40,45	40,45	40,45	5	1
híbrido - LSTM+saída	40,62	40,62	40,62	46	1
imagens	41,80	41,73	41,89	43	3

Fonte: Autor.

resultado ficou similar ao dos modelos somente com sensores, não foi observada melhora.

Figura 43 – MAE para o conjunto de validação para modelos híbridos importando pesos do melhor modelo somente com uso de sensores e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet



Fonte: Autor.

Tabela 36 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet

Experimento - treinável	Média	Mínimo	Máximo	Época	#runs
sensores	32,42	32,37	32,50	48	3
híbrido - saída	32,73	32,73	32,73	37	1
híbrido - LSTM+saída	33,33	33,33	33,33	43	1
híbrido - tudo	35,69	35,69	35,69	37	1
híbrido - ResNet+saída	37,08	37,08	37,08	46	1
imagens	41,80	41,73	41,89	43	3

Fonte: Autor.

#### 4.3.4 Adicionando uma camada densa

Nessa etapa, iterou-se nos melhores modelos dos casos anteriores, que seria não treinar a ResNet, iniciando-a com os pesos da ImageNet e variando treinar ou não a LSTM junto com a camada de saída. Entretanto, adicionou-se uma camada Densa com números diferentes de neurônios, com ativação ReLU, entre a concatenação e saída. A Tabela 37 demonstra, como nos casos anteriores, que no melhor dos casos os resultados ficam próximos aos dos modelos com sensores, mas não atingem valores melhores de forma consistente.

Tabela 37 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet, e adicionando uma MLP no depois da concatenação dos dados

Experimento - treinável	Média	Mínimo	Máximo	Época	#runs
sensores	32,42	32,37	32,50	48	3
híbrido+10 - LSTM+saída	32,70	32,70	32,70	48	1
híbrido+10 - saída	32,83	32,83	32,83	46	1
híbrido+50 - saída	33,44	33,44	33,44	10	1
híbrido+50 - LSTM+saída	33,59	33,59	33,59	31	1
híbrido+250 - saída	35,22	35,22	35,22	8	1
híbrido+250 - LSTM+saída	35,42	35,42	35,42	35	1
imagens	41,80	41,73	41,89	43	3

Fonte: Autor.

#### 4.3.5 Com todos os atributos como entrada

Nessa etapa buscou-se utilizar o melhor modelo somente com dados de sensores com todos os atributos, ao invés somente do com os 10 melhores. A intuição é trazer também informações sobre a direção e intensidade do vento e comparar se isso, junto aos dados de imagem, pode melhorar os resultados. Repetiu-se os testes do caso anterior com uma camada Densa de 10 neurônios e sem a camada Densa entre a saída da LSTM e a camada final.

Percebe-se pelas Tabelas 38 e 39 que os melhores resultados são encontrados treinando somente a saída. Esses resultados não são melhores que o modelo híbrido encontrado no teste anterior, portanto será mantido o modelo com somente os 10 atributos mais importantes.

#### 4.3.6 Resumo dos modelos híbridos

Percebe-se pelos experimentos, que para esse contexto estudado, a imagem não aparenta contribuir para os resultados. No melhor dos casos, os resultados são similares aos modelos utilizando somente dados os sensores, além de complexificar

Tabela 38 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores (com todos os atributos ao invés dos 10 melhores) e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet, e adicionando uma MLP no depois da concatenação dos dados

Experimento - treinável	Média	Mínimo	Máximo	Época	#runs
sensores	32,42	32,37	32,50	48	3
híbrido + 10 LSTM completo - saída	33,22	33,22	33,22	49	1
híbrido + 10 LSTM completo - LSTM+saída	33,55	33,55	33,55	48	1
híbrido + 10 LSTM completo - tudo	35,52	35,52	35,52	42	1
híbrido + 10 LSTM completo - ResNet+saída	37,62	37,62	37,62	49	1
imagens	41,80	41,73	41,89	43	3

Fonte: Autor.

Tabela 39 – Métrica MAE para o conjunto de validação dos modelos híbridos importando pesos do melhor modelo somente com uso de sensores (com todos os atributos ao invés dos 10 melhores) e inicializando os pesos do modelo de imagem a partir dos pesos da ImageNet, sem adição de uma MLP no depois da concatenação dos dados

Experimento - treinável	Média	Mínimo	Máximo	Época	#runs
sensores	32,42	32,37	32,50	48	3
híbrido LSTM completo - saída	33,24	33,24	33,24	29	1
híbrido LSTM completo - LSTM+saída	33,96	33,96	33,96	49	1
híbrido LSTM completo - tudo	35,21	35,21	35,21	46	1
híbrido LSTM completo - ResNet+saída	36,80	36,80	36,80	46	1
imagens	41,80	41,73	41,89	43	3

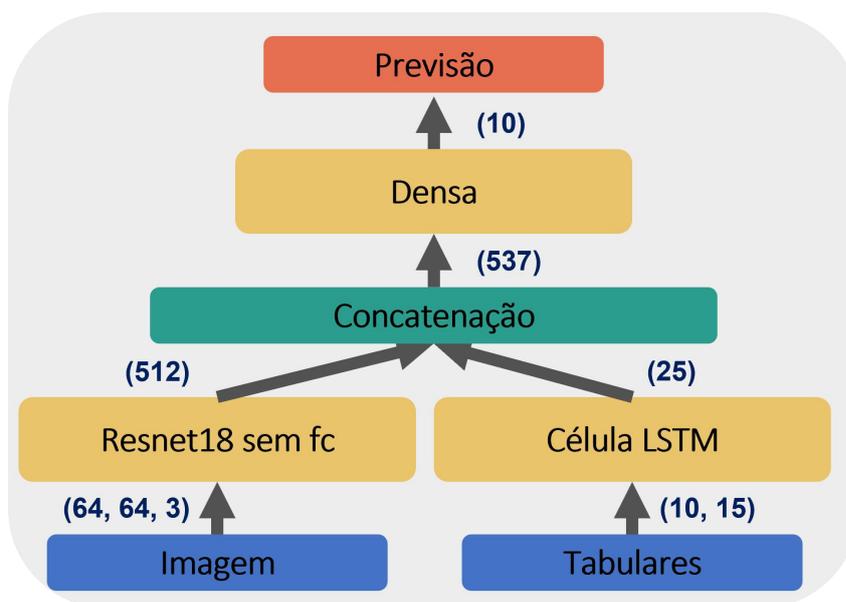
Fonte: Autor.

os modelos e aumentar a possibilidade de *overfitting*. Isso sugere que os dados dos sensores são suficientes para a melhor previsão possível nesse caso. Desta forma o melhor modelo encontrado até o momento segue sendo o modelo de sensores somente utilizando LSTM, seguido pelo híbrido e por último os modelos de imagem. A melhor arquitetura encontrada para o modelo híbrido é mostrada na Figura 44 e é baseada no modelo somente com dados de sensores com os 10 atributos mais importantes de entrada e a imagem atual.

#### 4.4 COMPILADOS DA SELEÇÃO DE HIPERPARÂMETROS

Nessa etapa resume-se a seleção de hiperparâmetros das seções 4.1, 4.2 e 4.3. A Tabela 40 demonstra os experimentos e principais informações a respeito de cada um deles.

Figura 44 – Arquitetura do melhor modelo híbrido



Fonte: Autor.

Tabela 40 – Resultados compilados referentes à etapa de seleção de hiperparâmetros (o S representa os experimentos somente com sensores, o I somente com imagens *all-sky* e o H os referentes ao modelo híbrido)

Exp.	Descrição	Melhor escolha/comentário
(S) 4.1.1	Variável Alvo	$K_t$
(S) 4.1.2	<i>Batch_size</i> e taxa de aprendizado	256, 0,001
(S) 4.1.3	Normalização	min-max
(S) 4.1.4	Função perda	L1
(S) 4.1.5	Capacidade do modelo	25 neurônios somente
(S) 4.1.6	Instâncias passadas	15 instâncias
(S) 4.1.7	Somente atributos importantes	10 atributos
(I) 4.2.1	Variável Alvo	$K_t$
(I) 4.2.2	<i>Batch_size</i> e taxa de aprendizado	256, 0,0001
(I) 4.2.3	Função perda	L1
(I) 4.2.4	Importar pesos ImageNet	Importar ImageNet
(I) 4.2.5	Recorte da imagem	Elíptico avançado
(I) 4.2.6	Tamanho da imagem	64x64x3
(I) 4.2.7	Possíveis arquiteturas	ResNet18
(I) 4.2.8	Normalização das imagens	Com normalização
(I) 4.2.9	Diferença entre atual e anterior	Somente atual
(I) 4.2.10	Múltiplas imagens	Somente atual
(I) 4.2.11	<i>Data augmentation</i>	Com <i>Data augmentation</i>
(I) 4.2.12	Treinar por mais épocas	Somente até época 50
(H) 4.3	Pesos LSTM	Importar melhores
(H) 4.3	Liberar pesos LSTM para treinamento?	Não
(H) 4.3	Pesos ResNet	Importar da ImageNet
(H) 4.3	Liberar pesos ResNet para treinamento?	Não
(H) 4.3.4	Camada densa	10 neurônios
(H) 4.3.5	Atributos	Somente os 10 principais

Fonte: Autor.

#### 4.5 RETREINAMENTO E RESULTADOS NO CONJUNTO DE TESTE

Nessa etapa, os melhores modelos para cada conjunto de dados foram retreinados 5 vezes e os resultados da última época analisados. Esse processo foi realizado duas vezes. A primeira, utilizando somente os dados de 2014 como treinamento e a segunda utilizando tanto os dados de 2014 e 2015 para treinamento dos modelos. Os resultados e métricas apresentados a seguir são todos do conjunto de teste e os valores apresentados são a média e desvio padrão dos resultados de cada uma das *runs*.

Primeiramente, a Tabela 41 demonstra os resultados para o conjunto de teste completo. Nessa tabela, pode-se observar que os resultados para o MAE do modelo somente com dados de sensores obtiveram os melhores resultados, seguidos pelo modelo com dados híbridos e por último os modelos com imagens, como observado anteriormente. É interessante de se observar, que tirando o híbrido treinado somente em 2014, que parece que teve uma run com piores resultados, os modelos de sensores e híbrido estão praticamente equivalentes. Isso, corrobora a hipótese que a imagem está sendo praticamente ignorada para conseguir esses resultados comparáveis. Para o RMSE, segue-se a mesma tendência, porém com o híbrido um pouco melhor que os outros.

Tabela 41 – Métricas de erro para o retreinamento - conjunto de teste completo

Modelo	Dados de treinamento	MAE	RMSE
Sensores	2014	30,75 ± 0,13	73,65 ± 0,45
Sensores	2014 e 2015	<b>30,47 ± 0,44</b>	72,97 ± 0,13
Imagens	2014	39,49 ± 0,42	80,67 ± 0,65
Imagens	2014 e 2015	37,89 ± 0,33	79,47 ± 0,78
Híbrido	2014	32,10 ± 1,16	73,22 ± 0,38
Híbrido	2014 e 2015	30,67 ± 0,17	<b>72,32 ± 0,27</b>
Persistente	-	35,53	87,45

Fonte: Autor.

Além dos resultados para o conjunto completo, buscou-se comparar separadamente onde a GHI futura foi considerada como sendo de céu claro ou nublado, essa separação dos dados é explicada nas Subseções 3.1.4 e 3.1.8. A Tabela 42 mostra os resultados compilados somente para os períodos ensolarados. Observa-se resultados similares aos encontrados no conjunto completo. Outro ponto importante a ser mencionado é a ordem de grandeza dos erros, que para os melhores modelos é abaixo de 5 para o MAE e abaixo de 10 para o RMSE. Isso é interessante, pois essa parcela do problema é praticamente resolvida, encontrando boas metodologias para realizar as previsões em momentos nublados pode melhorar significativamente os resultados do modelo como um todo.

Tabela 42 – Métricas de erro para o retreinamento - períodos ensolarados

Modelo	Dados de treinamento	MAE	RMSE
Sensores	2014	<b>3,61 ± 0,25</b>	8,40 ± 0,26
Sensores	2014 e 2015	3,79 ± 0,78	<b>7,85 ± 0,37</b>
Imagens	2014	12,86 ± 0,50	18,56 ± 1,09
Imagens	2014 e 2015	10,70 ± 0,31	16,29 ± 0,47
Híbrido	2014	5,67 ± 1,92	11,42 ± 1,91
Híbrido	2014 e 2015	3,96 ± 0,31	8,95 ± 0,58
Persistente	-	5,08	19,83

Fonte: Autor.

Já na Tabela 43, observa-se os erros para os períodos nublados. Todos possuem uma maior ordem de grandeza e essa etapa que é o cerne de maior dificuldade da previsão. Como nos casos anteriores, os melhores resultados são disputados entre os modelos somente com dados de sensores e modelo híbrido. Outro ponto interessante a ser observado, é que o modelo de imagem é similar ao persistente para os períodos nublados. Isso é um indício que o problema de previsão do modelo de imagens é o erro relacionado ao modelo de céu claro, já que a previsão é feita no  $K_t$  e depois reconvertida para o GHI. Esse é um resultado relevante, pois melhorando esse modelo de céu claro, possivelmente o modelo de imagens pode obter resultados melhores que o SP.

Tabela 43 – Métricas de erro para o retreinamento - períodos nublados

Modelo	Dados de treinamento	MAE	RMSE
Sensores	2014	60,45 ± 0,12	106,21 ± 0,66
Sensores	2014 e 2015	<b>59,67 ± 0,09</b>	105,26 ± 0,20
Imagens	2014	68,63 ± 0,48	115,10 ± 0,86
Imagens	2014 e 2015	67,64 ± 0,37	113,74 ± 1,10
Híbrido	2014	61,02 ± 0,40	105,26 ± 0,40
Híbrido	2014 e 2015	59,88 ± 0,09	<b>104,23 ± 0,40</b>
Persistente	-	68,85	124,83

Fonte: Autor.

Além disso, comparou-se os melhores modelos para cada conjunto de dados de entrada na Tabela 44 utilizando mais opções de métricas de erro. Esses resultados são o valor médio entre as 5 *runs* para cada conjunto de dados, e esses modelos foram os treinados em 2014 e 2015. Pode-se observar que nas métricas relacionadas ao MAE, o modelo com dados de sensores obteve o melhor resultado, atingindo  $FS_{MAE}$  igual a 14,2% em relação ao SP. Para as métricas relacionadas com o RMSE, o modelo híbrido obteve os melhores resultados. Como comparação, em Wen *et al.* (2022), para a previsão de 15 minutos a frente, foi obtido resultados com nMAE 8,47% e nRMSE 14,72%, ambos similares a ordem de grandeza encontrada nesse trabalho. É importante observar que a previsão de Wen *et al.* (2022) é a média de 5 instantes

de tempo e nessa dissertação é o valor instantâneo, então não necessariamente é uma comparação um para um, porém é o trabalho que mais se aproxima a esse na metodologia e conjuntos de dados utilizados.

Tabela 44 – Resultados no conjunto de teste dos melhores modelos treinados com função perda L1

Modelo	MAE	$FS_{MAE}$ %	nMAE %	RMSE	$FS_{RMSE}$ %	nRMSE %	MAPE %	R2 %
<i>SP</i>	35,5	-	8,0	87,4	-	19,6	16,3	90,6
Sensores	<b>30,5</b>	<b>14,2</b>	<b>6,8</b>	73,0	16,6	16,4	<b>14,4</b>	93,5
Imagens	37,9	-6,6	8,5	79,5	9,1	17,8	20,2	92,3
Híbrido	30,7	13,7	6,9	<b>72,3</b>	<b>17,3</b>	<b>16,2</b>	14,7	<b>93,6</b>

Fonte: Autor.

Os resultados do modelo híbrido acabaram ficando melhores nas métricas relacionadas ao MSE, quando foi utilizada a perda L1 para o treinamento e a métrica MAE para o ajuste de parâmetros. Para confirmar se os dados de imagem estão sendo relevantes ou não, fez-se um último teste repetindo o treinamento da Tabela 44, porém utilizando a perda L2 ao invés da L1. Esses novos resultados, apresentados na Tabela 45, demonstram que o modelo de sensores é, novamente, superior ao híbrido na métrica otimizada, seja MAE ou MSE. Desta forma, comprova-se que a imagem não ajudou em nada para esse contexto estudado. Outro ponto relevante, é que foi possível comparar os resultados desses modelos com outros da literatura otimizados para o RMSE, ainda que de uma forma sub-ótima, já que todos os hiperparâmetros desses modelos foram selecionados usando o MAE como métrica a ser otimizada. Por exemplo, o  $FS_{RMSE}$  atingiu 19,1% para o modelo com dados de sensores. Esse resultado, novamente, é similar aos da literatura: em Paletta, Arbod e Lasenby (2021) foi encontrado  $FS_{RMSE}$  entre 20 e 21% para previsões de 10 e 20 minutos a frente e em Wen *et al.* (2022) os resultados foram 18,3% para 15 minutos a frente.

Tabela 45 – Resultados no conjunto de teste dos melhores modelos treinados com função perda L2

Modelo	MAE	$FS_{MAE}$ %	nMAE %	RMSE	$FS_{RMSE}$ %	nRMSE %	MAPE %	R2 %
<i>SP</i>	35,5	-	8,0	87,4	-	19,6	16,3	90,6
Sensores	<b>32,6</b>	<b>8,4</b>	<b>7,3</b>	<b>70,8</b>	<b>19,1</b>	<b>15,9</b>	<b>15,7</b>	<b>93,9</b>
Imagens	37,9	-6,6	8,5	78,1	10,7	17,5	20,1	92,5
Híbrido	34,9	1,9	7,8	70,8	19,0	15,9	16,1	93,9

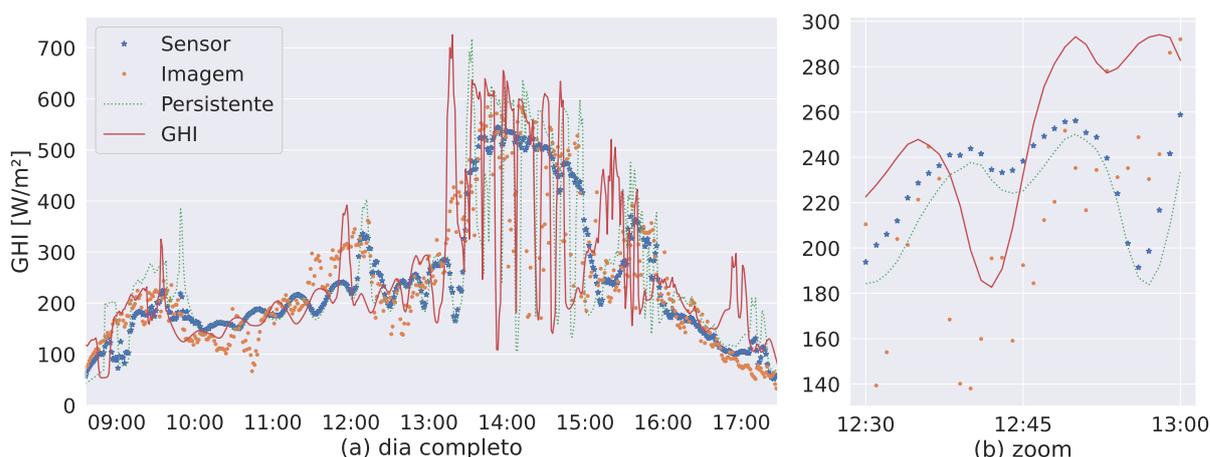
Fonte: Autor.

### 4.5.1 Previsões dos modelos para dias específicos

Nessa etapa, buscou-se comparar a previsão dos modelos para alguns dias em específico e buscar entender mais sobre essas previsões e possíveis pontos de melhoria. Como informações complementares, também serão mostradas a GHI futura (alvo da previsão) e a previsão da GHI pelo modelo SP. Para os modelos, foi escolhida para visualização a melhor *run* em relação ao MAE para o conjunto de teste completo. Como os resultados do modelo híbrido são muito similares ao do modelo somente com dados de sensores, para não poluir os gráficos com muitas informações, preferiu-se mostrar somente os resultados dos modelos somente com sensores e somente com imagens. Buscou-se escolher dois dias ensolarados e dois dias nublados, próximos do verão e inverno local, para comparar a resposta dos modelos ao decorrer do ano.

A Figura 45 mostra os resultados para um dia nublado em janeiro de 2016. Pode-se observar, que as variações do modelo persistente, como esperado, respondem após o tempo de previsão de 15 minutos. Por exemplo, vários picos oscilam e retornam a menores valores, antes da previsão do modelo persistente variar. Pelo zoom, pode-se observar que os resultados dos modelos somente com dados de sensores, seguem mais o valor médio do GHI futuro e corrigindo um pouco a previsão do modelo persistente inteligente, como esperado pela métrica otimizada. Por outro lado, a previsão do modelo de imagem é mais instável e não demonstram acompanhar tão bem o GHI, por vezes variando sem aparente mudança do GHI, como observado próximo das 11:00.

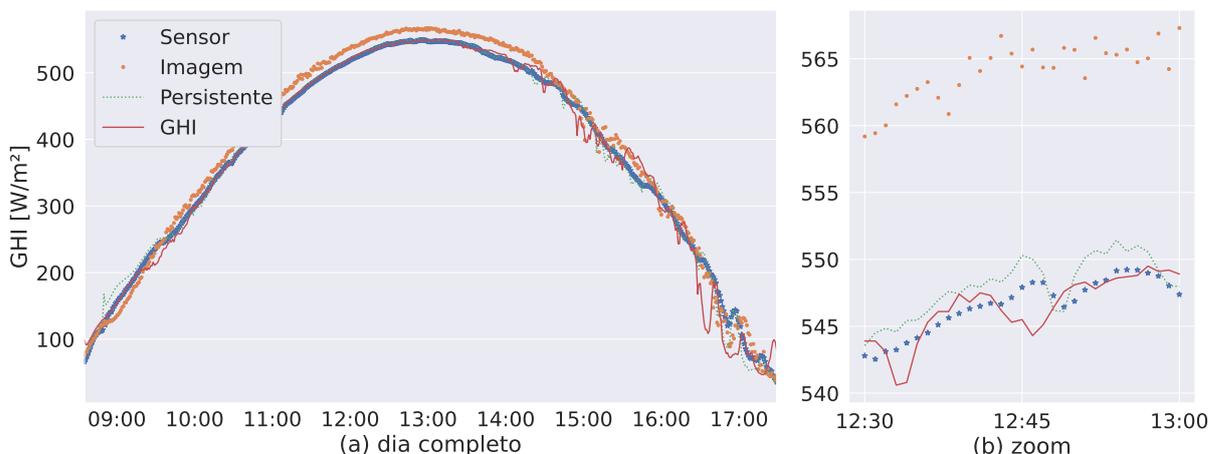
Figura 45 – Resultados dos modelos dia nublado de inverno - 2016-01-24



Fonte: Autor.

Já a Figura 47 mostra as previsões para um dia ensolarado de inverno. Nesse caso, tanto o modelo somente com dados de sensores quanto persistente oscilam junto com o GHI, e seguem sua tendência. Por outro lado, as imagens estão acima dos outros, possivelmente relacionados aos erros do modelo de céu claro utilizado e com a previsão sendo feita sobre o  $K_t$ .

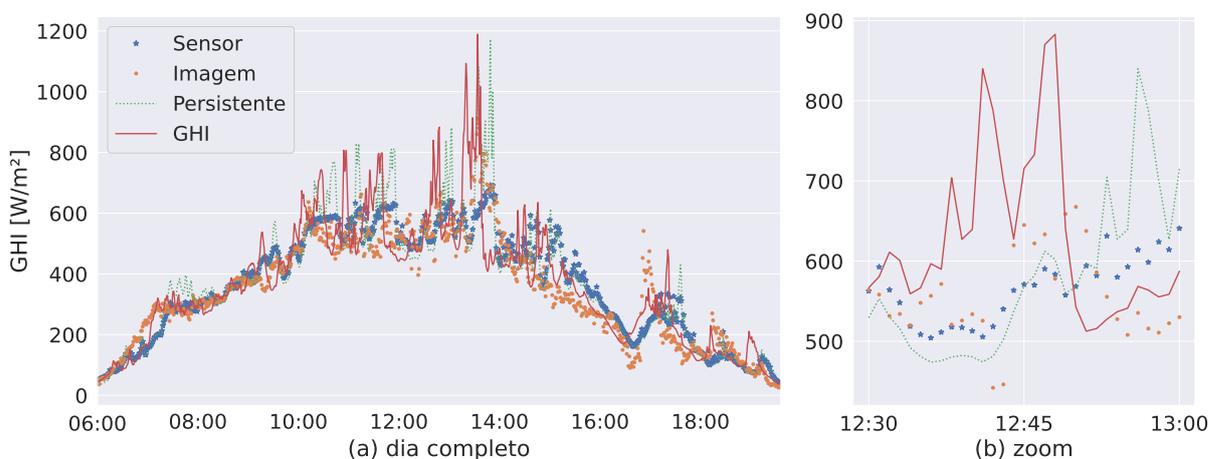
Figura 46 – Resultados dos modelos dia ensolarado de inverno - 2016-01-25



Fonte: Autor.

Em adição aos casos anteriores, a Figura 47 mostra um dia nublado no verão. Um primeiro ponto a se observar é a ordem de grandeza maior atingida pelo GHI, próximo a 1.200 W/m<sup>2</sup>. Além disso, observa-se uma situação similar ao primeiro dia nublado, com o persistente demorando a responder e o modelo somente com dados de sensores seguindo um valor médio da previsão. Nesse caso, ao menos no zoom, a previsão das imagens pareceu em média mais próximo ao GHI, mas com maior oscilação que as outras. Outro instante onde as imagens tiveram previsões acima do esperado foi próximo as 17:00 com um pico de irradiância prevista maior que a observada.

Figura 47 – Resultados dos modelos dia nublado de verão - 2016-06-04

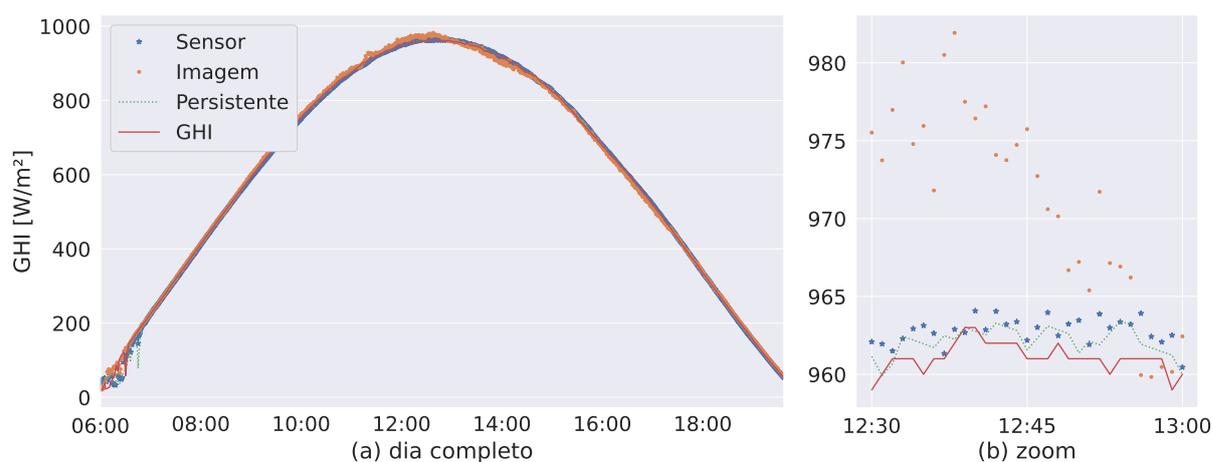


Fonte: Autor.

Finalmente, a Figura 48 mostra um dia de verão ensolarado. Um primeiro ponto que chama a atenção é que a irradiância não ultrapassa 1.000 W/m<sup>2</sup> e como é somente

um dia de diferença do caso anterior, pode ser que os picos observados no dia nublado referem-se momentos de *over irradiance* conforme observado por Moreira Branco e Tiepolo (2020). Voltando ao dia ensolarado, todas as previsões estão mais próximas da GHI, até mesmo o modelo de imagem, apesar de algumas instabilidades no zoom. Isso demonstra que o modelo de céu claro está melhor adaptado aos meses de verão que de inverno para essa localidade, demonstrando novamente um possível ponto de melhoria a ser seguido.

Figura 48 – Resultados dos modelos dia ensolarado de verão - 2016-06-05



Fonte: Autor.

## 5 CONCLUSÃO E DISCUSSÃO

Nesse trabalho, foram aplicados diversos métodos de aprendizado de máquina para 3 opções de conjuntos de dados de entrada: somente sensores, somente imagens *all-sky* e ambos. Sobre os resultados, um dos pontos mais relevantes observado foi que em várias instâncias, o aumento da capacidade dos modelos não trouxe melhores resultados no conjunto de validação e, por vezes, aumentou ou fez com que fosse observado o *overfitting*.

O melhor modelo utilizando somente dados de sensores encontrado utilizou a arquitetura LSTM somente com uma camada, 25 neurônios ocultos e 15 instâncias de tempo (atual e 14 passadas). A previsão foi feita para o  $K_t$  futuro, somente 10 atributos foram utilizados para a previsão e os dados foram normalizados com a normalização min-max. Já o melhor modelo somente com imagens encontrado foi utilizando a ResNet18 como base, também prevendo o  $K_t$  e com a imagem recortada a partir de um recorte elíptico em 256x256 e convertida para 64x64. O melhor modelo com dados híbridos, foi utilizando a combinação dos melhores modelos somente com dados de sensores e somente com imagens concatenados e com uma camada Densa com 10 neurônios, com ativação ReLU, e mantendo a ResNet com os pesos da ImageNet congelados para o treinamento.

Com esses parâmetros, realizou-se o retreinamento e comparou-se os resultados com o SP para o conjunto de teste completo, e separadamente para os dados nublados ou de céu claro. Percebeu-se que os melhores resultados estão relacionados aos modelos somente com dados de sensores ou híbridos, com os modelos somente com imagens piores que os outros e, tirando o caso para os dados nublados, também piores que o modelo persistente. Outro ponto importante encontrado é que o modelo com dados híbridos somente atingiu resultados comparáveis ao modelo somente com dados de sensores, demonstrando que a imagem não trouxe informações relevantes que pudessem melhorar os resultados de previsão, ao menos para esse contexto estudado. O melhor resultado para o conjunto completo foi utilizando o modelo com dados de sensores, cujo  $FS_{MAE}$  em relação ao SP foi de 14,2%. Esse resultado é comparável a literatura, sendo obtido em Paletta, Arbod e Lasenby (2021) 10,4% para a previsão da GHI e em Zhang *et al.* (2018) 12,1% para a previsão da potência 10 minutos a frente. Além disso, obteve-se valores próximos para o nMAE e nRMSE em relação aos obtidos em Wen *et al.* (2022) para a previsão 15 minutos a frente (embora nesse trabalho os autores utilizaram como variável alvo a média da irradiância em 5 minutos, ao invés da irradiância instantânea como nessa dissertação). Para concluir quais desses modelos de fato possuem melhores resultados é importante avaliá-los nos mesmos conjuntos de dados e com os mesmos procedimentos. Como os conjuntos e processos são distintos, não é possível afirmar nesse estudo quais são melhores; mesmo assim,

a ordem de grandeza demonstra, ao menos, resultados similares com os da literatura.

Um ponto inesperado encontrado, foi que a adição das imagens não trouxe melhorias às previsões. Pode-se levantar três hipóteses que explicariam esse resultado: (1) as informações das imagens para esse contexto sejam desnecessárias e a previsão dos sensores por si só já obtém os melhores resultados; (2) sejam necessários modelos e sistemas mais complexos e diferentes, para poder melhor extrair atributos das imagens e melhor agregar os dados de imagens e sensores; ou (3) o conjunto de dados tenha inconsistências na sua criação (por exemplo, relacionados ao sincronismo entre os dados de sensores e imagens ou outras limitações). Em relação a literatura, comparou-se os resultados encontrados da melhor forma possível, mas existem diversos datasets, metodologias e processos distintos e ainda não há um consenso sobre qual é a melhor forma de trabalhar para a previsão de curto prazo da GHI.

Mesmo tendo essas limitações e contexto, acredita-se que o estudo dessa dissertação é relevante, pela metodologia focada em melhorar os modelos de aprendizado de máquina, comparar diferentes conjuntos de dados verificar os resultados em relação ao modelo persistente inteligente. Desta forma, entende-se como positivo os resultados encontrados nesse trabalho e a disponibilização desses resultados e metodologia que, até o conhecimento do autor, não existe desta mesma forma na literatura.

## 5.1 PRÓXIMOS TRABALHOS

Para próximos trabalhos, pode-se considerar alguns caminhos de estudo e melhoria. Uma possibilidade seria estudar esse contexto de comparação entre modelos de aprendizado de máquina em outros períodos de previsão, e também outros conjuntos de dados, e confirmar se as mesmas conclusões são alcançadas.

Outra possibilidade seria focar na melhoria dos modelos individualmente, como no modelo somente com imagens. Percebeu-se que as previsões desse modelo variam significativamente de acordo com a época do ano e até mesmo em pequenos intervalos de tempo, mesmo sem uma variação significativa da GHI. Seria viável buscar ou desenvolver melhores modelos de céu claro, adaptados especificamente ao local ou estação em questão. Além disso, seria possível utilizar outros dados conhecidos em conjunto com a imagem, como o minuto do dia, o dia do ano e o GHI de céu claro, por exemplo. Outra abordagem seria focar na estabilização das previsões, considerando as previsões dos minutos anteriores, seja por meio de uma média móvel exponencial ou desenvolvendo modelos para agregar essas previsões.

Por fim, uma última sugestão seria focar no desenvolvimento de arquiteturas ainda mais complexas para a previsão da GHI diretamente. Portanto, pode ser necessário explorar arquiteturas diferentes ou até mesmo considerar a integração direta da parte física relacionada à energia solar.

## REFERÊNCIAS

- ALZUBAIDI, Laith *et al.* Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. en. **Journal of Big Data**, v. 8, n. 1, p. 53, mar. 2021. ISSN 2196-1115.
- ANTONANZAS-TORRES, F.; URRACA, R.; POLO, J.; PERPIÑÁN-LAMIGUEIRO, O.; ESCOBAR, R. Clear sky solar irradiance models: A review of seventy models. en. **Renewable and Sustainable Energy Reviews**, v. 107, p. 374–387, jun. 2019. ISSN 13640321.
- BISHOP, Christopher M. **Pattern recognition and machine learning**. New York: Springer, 2006. (Information science and statistics). ISBN 978-0-387-31073-2.
- CHU, Yinghao; PEDRO, Hugo T.C.; LI, Mengying; COIMBRA, Carlos F.M. Real-time forecasting of solar irradiance ramps with smart image processing. en. **Solar Energy**, v. 114, p. 91–104, abr. 2015. ISSN 0038092X.
- CORSINI, Iuri. **Energia elétrica aumentou mais do que o dobro da inflação nos últimos anos**. pt-BR. [S.l.: s.n.], 2022. Disponível em: <https://www.cnnbrasil.com.br/economia/energia-eletrica-aumentou-mais-do-que-o-dobro-da-inflacao-nos-ultimos-anos/>. Acesso em: 6 jun. 2023.
- DATA, Our World in. **Solar (photovoltaic) panel prices**. [S.l.: s.n.], 2023. Disponível em: <https://ourworldindata.org/grapher/solar-pv-prices>. Acesso em: 6 jun. 2023.
- EPE. **Balanco Energético Nacional 2022**. pt. [S.l.], 2022.
- GÉRON, Aurélien. **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow**. 2. ed. [S.l.: s.n.], 2019.
- HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. **Deep Residual Learning for Image Recognition**. en. [S.l.]: arXiv, dez. 2015. arXiv:1512.03385 [cs]. Disponível em: <http://arxiv.org/abs/1512.03385>. Acesso em: 8 mar. 2023.
- HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long Short-Term Memory. **Neural Computation**, v. 9, n. 8, p. 1735–1780, nov. 1997. Conference Name: Neural Computation. ISSN 0899-7667.

INEICHEN, Pierre. A broadband simplified version of the Solis clear sky model. en. **Solar Energy**, v. 82, n. 8, p. 758–762, ago. 2008. ISSN 0038092X.

INEICHEN, Pierre; PEREZ, Richard. A new airmass independent formulation for the Linke turbidity coefficient. en. **Solar Energy**, v. 73, n. 3, p. 151–157, set. 2002. ISSN 0038092X.

JUNCKLAUS MARTINS, Bruno; CERENTINI, Allan; MANTELLI NETO, Sylvio Luiz; LOUREIRO CHAVES, Thiago Zimmermann; MOREIRA BRANCO, Nicolas; WANGENHEIM, Aldo von; RÜTHER, Ricardo; MARIAN ARRAIS, Juliana. Systematic Review of Nowcasting Approaches for Solar Energy Production based upon Ground-Based Cloud Imaging. en. **Solar Energy Advances**, p. 100019, jul. 2022. ISSN 2667-1131.

KUMAR, Varun; PANDEY, A. S.; SINHA, S. K. Grid integration and power quality issues of wind and solar energy system: A review. en. *In: 2016 International Conference on Emerging Trends in Electrical Electronics & Sustainable Energy Systems (ICETEESES)*. Sultanpur, India: IEEE, mar. 2016. P. 71–80.

LIU, Benjamin Y.H.; JORDAN, Richard C. The interrelationship and characteristic distribution of direct, diffuse and total solar radiation. en. **Solar Energy**, v. 4, n. 3, p. 1–19, jul. 1960. ISSN 0038092X.

MACIEL, Joylan Nunes; WENTZ, Victor Hugo; LEDESMA, Jorge Javier Gimenez; ANDO JUNIOR, Oswaldo Hideo. Analysis of Artificial Neural Networks for Forecasting Photovoltaic Energy Generation with Solar Irradiance. en. **Brazilian Archives of Biology and Technology**, v. 64, spe, e21210131, 2021. ISSN 1678-4324, 1516-8913.

MOREIRA BRANCO, Nicolas; TIEPOLO, Gerson Máximo. Análise preliminar das irradiâncias na estação SONDA da UTFPR em Curitiba. pt. *In: SEMINÁRIO de Iniciação Científica e Tecnológica - SICITE*. [S.l.: s.n.], out. 2020. v. XXV.

NARKHEDE, Meenal V.; BARTAKKE, Prashant P.; SUTAONE, Mukul S. A review on weight initialization strategies for neural networks. en. **Artificial Intelligence Review**, v. 55, n. 1, p. 291–322, jan. 2022. ISSN 0269-2821, 1573-7462.

PALETTA, Quentin; ARBOD, Guillaume; LASENBY, Joan. Benchmarking of deep learning irradiance forecasting models from sky images – An in-depth analysis. en. **Solar Energy**, v. 224, p. 855–867, ago. 2021. ISSN 0038092X.

PATWARDHAN, Narendra; MARRONE, Stefano; SANSONE, Carlo. Transformers in the Real World: A Survey on NLP Applications. en. **Information**, v. 14, n. 4, p. 242, abr. 2023. ISSN 2078-2489.

PEDRO, Hugo T. C.; LARSON, David P.; COIMBRA, Carlos F. M. A comprehensive dataset for the accelerated development and benchmarking of solar forecasting methods. en. **Journal of Renewable and Sustainable Energy**, v. 11, n. 3, p. 036102, mai. 2019. ISSN 1941-7012.

PEREIRA, Enio *et al.* **Atlas brasileiro de energia solar**. 2. ed. São José dos Campos: INPE, 2017. ISBN 978-85-17-00089-8.

PI, Maozheng; JIN, Ning; CHEN, Dongxiao; LOU, Bing. Short-Term Solar Irradiance Prediction Based on Multichannel LSTM Neural Networks Using Edge-Based IoT System. en. Edição: Yingjie Wang. **Wireless Communications and Mobile Computing**, v. 2022, p. 1–11, jan. 2022. ISSN 1530-8677, 1530-8669.

RAJAGUKGUK, Rial A.; RAMADHAN, Raden A. A.; LEE, Hyun-Jin. A Review on Deep Learning Models for Forecasting Time Series Data of Solar Irradiance and Photovoltaic Power. en. **Energies**, v. 13, n. 24, p. 6623, dez. 2020. ISSN 1996-1073.

REDA, I; ANDREAS, A. **Solar Position Algorithm for Solar Radiation Applications (Revised)**. en. [S./], jan. 2008. nrel/tp-560-34302, 15003974. Disponível em: <http://www.osti.gov/servlets/purl/15003974/>. Acesso em: 13 jun. 2022.

SINGLA, Pardeep; DUHAN, Manoj; SAROHA, Sumit. Review of Different Error Metrics: A Case of Solar Forecasting. en. **AIUB Journal of Science and Engineering (AJSE)**, v. 20, n. 4, p. 158–165, dez. 2021. ISSN 2520-4890, 1608-3679.

STEFFEL, S. J.; CAROSELLI, P. R.; DINKEL, A. M.; LIU, J. Q.; SACKKEY, R. N.; VADHAR, N. R. Integrating Solar Generation on the Electric Distribution Grid. en. **IEEE Transactions on Smart Grid**, v. 3, n. 2, p. 878–886, jun. 2012. ISSN 1949-3053, 1949-3061.

VAN HOUDT, Greg; MOSQUERA, Carlos; NÁPOLES, Gonzalo. A review on the long short-term memory model. en. **Artificial Intelligence Review**, v. 53, n. 8, p. 5929–5955, dez. 2020. ISSN 0269-2821, 1573-7462.

WANG, Zhongju; WANG, Long; HUANG, Chao; LUO, Xiong. A Hybrid Ensemble Learning Model for Short-Term Solar Irradiance Forecasting Using Historical Observations and Sky Images. en. *In: 2021 IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia)*. Chengdu, China: IEEE, jul. 2021. P. 1404–1408.

WEN, Haoran; DU, Yang; LIM, Eng Gee; WEN, Huiqing; YAN, Ke; LI, Xingshuo; JIANG, Lin. A solar forecasting framework based on federated learning and distributed computing. en. **Building and Environment**, v. 225, p. 109556, nov. 2022. ISSN 03601323.

WENTZ, Victor Hugo; MACIEL, Joylan Nunes; GIMENEZ LEDESMA, Jorge Javier; ANDO JUNIOR, Oswaldo Hideo. Solar Irradiance Forecasting to Short-Term PV Power: Accuracy Comparison of ANN and LSTM Models. en. **Energies**, v. 15, n. 7, p. 2457, mar. 2022. ISSN 1996-1073.

YANG, Dazhi *et al.* A review of solar forecasting, its dependence on atmospheric sciences and implications for grid integration: Towards carbon neutrality. en. **Renewable and Sustainable Energy Reviews**, v. 161, p. 112348, jun. 2022. ISSN 13640321.

ZHANG, Jinsong; VERSCHAE, Rodrigo; NOBUHARA, Shohei; LALONDE, Jean-François. Deep photovoltaic nowcasting. en. **Solar Energy**, v. 176, p. 267–276, dez. 2018. ISSN 0038092X.