

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
ELÉTRICA

PREVISÃO DE GERAÇÃO FOTOVOLTAICA ATRAVÉS DE
MÉTODOS COMPUTACIONAIS PARAMETRIZADOS POR
PLANEJAMENTO DE EXPERIMENTOS

Max Olinto Moreira

Itajubá

Julho de 2022

Max Olinto Moreira

**PREVISÃO DE GERAÇÃO FOTOVOLTAICA ATRAVÉS
DE MÉTODOS COMPUTACIONAIS
PARAMETRIZADOS POR PLANEJAMENTO DE
EXPERIMENTOS**

Tese submetida ao Programa de Pós-Graduação em Engenharia Elétrica como parte dos requisitos para a obtenção do título de Doutor em Ciências em Engenharia Elétrica.

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Orientador: Prof. Dr. Pedro Paulo Balestrassi
Coorientador: Prof. Dr. Benedito Donizeti Bonatto

Itajubá
Julho de 2022

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
ELÉTRICA

Max Olinto Moreira

PREVISÃO DE GERAÇÃO FOTOVOLTAICA ATRAVÉS DE
MÉTODOS COMPUTACIONAIS PARAMETRIZADOS POR
PLANEJAMENTO DE EXPERIMENTOS

Tese submetida ao Programa de Pós-Graduação em Engenharia Elétrica como parte dos requisitos para obtenção do título de Doutor em Ciências em Engenharia Elétrica.

Banca Examinadora:

Prof. Dr. Antônio Carlos Zambroni de Souza (Unifei)

Prof. Dr. Paulo Fernando Ribeiro (Unifei)

Prof. Dr. Takaaki Ohishi (Unicamp)

Prof. Dr. Wesley Vieira da Silva (UFAL)

Prof. Dr. Pedro Paulo Balestrassi (Orientador)

Prof. Dr. Benedito Donizeti Bonatto (Coorientador)

Itajubá
Julho de 2022

Dedico este trabalho especialmente à minha esposa Natália e aos familiares que me acompanharam nesta jornada.

AGRADECIMENTOS

A Deus, que me permite percebê-Lo em meu caminho. Por cuidar de cada detalhe, fazendo-me sentir que, mesmo quando as coisas aparentam não irem da maneira como eu gostaria que fosse, Ele arquiteta situações e momentos para que tudo ocorra da melhor maneira.

Ao meu orientador, Prof. Pedro Paulo Balestrassi, por ter acreditado em mim desde o começo desta jornada, não condicionando minha entrada no programa de doutorado à publicação de um artigo mas, antes, conduzindo-me e apontando os caminhos que eu deveria seguir. Tenho admiração pelo Prof. Pedro não apenas pela competência acadêmica, a qual exerce com maestria, mas essencialmente pela pessoa que ele é, com olhar voltado à formação humana de seus orientados e não apenas técnica. Prof. Pedro, obrigado pela confiança e por me desafiar a buscar novos conhecimentos.

Aos colegas do GEPE Qualidade pelos momentos de conversa, de aprendizado e de dificuldade que compartilhamos.

Aos professores do programa de Pós-Graduação em Engenharia Elétrica Antônio Zambroni, Paulo Ribeiro e Benedito Bonatto pelos ensinamentos.

Ao IFSULDEMINAS, em especial ao Campus Av. Carmo de Minas, pelo apoio, incentivo à qualificação e por proporcionarem as condições necessárias para me dedicar ao doutorado.

Àqueles que me receberam e acolheram de maneira tão carinhosa durante essa jornada: meus queridos familiares Betânia, Hemekson e Luiza.

A todos os amigos e familiares que, de alguma forma, participaram comigo nessa jornada, seja com orações e palavras de consolo nos momentos de dificuldades, seja com mensagens de alegria e comemorações nos momentos de conquistas.

Por último e nunca menos importante, à minha esposa Natália, pelo companheirismo, incentivo, compreensão e amor incondicional.

*“..visto que ninguém conhece o futuro,
quem lhe poderá dizer o que vai acontecer?”
(Eclesiastes 8:7)*

RESUMO

Nos últimos anos, as fontes de energia renováveis e sustentáveis atraíram a atenção de vários investidores e partes interessadas, como agentes do setor de energia e consumidores. Os sistemas de energia elétrica têm experimentado a rápida inserção de fontes geradoras renováveis distribuídas e, como resultado, enfrentam desafios de planejamento e operação à medida que novas conexões são feitas à rede. É de grande dificuldade observar e antecipar os níveis exigidos de geração fotovoltaica, que são tarefas consideradas inerentes a uma rápida inserção na rede elétrica. Essa geração distribuída/renovável deve ser integrada de forma coordenada, de modo que não haja impacto negativo no desempenho elétrico da rede, aumentando a complexidade do gerenciamento de energia. Neste trabalho, uma estratégia multivariada, baseada em planejamento de experimentos (*DOE*), é endereçada para a previsão de geração fotovoltaica usando uma nova abordagem para parametrização e combinação de um conjunto de redes neurais artificiais (RNA). Duas questões principais serão exploradas: como selecionar as RNAs e como combiná-las no *ensemble*. Como complemento dessa metodologia, também é apresentada a redução de dimensionalidade dos dados climáticos através de Análise de Componentes Principais (*PCA*). A abordagem de planejamento de experimentos (*DOE*) é aplicada aos fatores da série temporal de geração fotovoltaica e aos fatores da RNA. Em seguida, é realizada uma análise de *cluster* para selecionar as redes que obtiveram os melhores resultados. A partir deste ponto, uma análise de mistura (*MDE*) é empregada para determinar os pesos ideais para a formação da previsão por conjunto *ensemble*. A metodologia é detalhada ao longo do trabalho e, com base na combinação de previsões, foi estimada a geração fotovoltaica para um conjunto de painéis específicos, localizado no sul do Estado de Minas Gerais. Por conseguinte, um estudo mais abrangente, que considerou um conjunto de dados de dezessete plantas de geração, com características sazonais, também foi examinado. A versatilidade do método proposto permitiu a alteração do número de fatores a serem utilizados no arranjo experimental, no modelo de previsão e no horizonte de previsão desejado e, conseqüentemente, aprimorou a determinação da previsão para os cenários estudados.

Palavras-chave: Previsão Fotovoltaica; Planejamento de Experimentos; Redes Neurais Artificiais; *Ensemble*.

ABSTRACT

In recent years, renewable and sustainable energy sources have attracted the attention of various investors and stakeholders, such as energy sector players and consumers. Electric power systems have experienced the rapid insertion of distributed renewable generating sources and, as a result, face planning and operational challenges as new connections are made to the grid. It is very difficult to observe and anticipate the required levels of photovoltaic generation, which are tasks considered inherent to a quick insertion into the electrical grid. This distributed/renewable generation must be integrated in a coordinated way, so that there is no negative impact on the electrical performance of the grid, increasing the complexity of energy management. In this work, a multivariate strategy, based on design of experiments (DOE), is addressed for the prediction of photovoltaic generation using a new approach for parameterization and combination of a set of artificial neural networks (ANN). Two main questions will be explored: how to select the ANNs and how to combine them in the forecast by sets (ensemble). As a complement to this methodology, the reduction of dimensionality of climate data through Principal Component Analysis (PCA) is also presented. The design of experiments (DOE) approach is applied to the PV generation time series factors and to the ANN factors. Then, a cluster analysis is performed to select the networks that obtained the best results. From this point, a mixture analysis (MDE) is used to determine the ideal weights for the formation of the ensemble. The methodology is detailed throughout the work and, based on the combination of forecasts, the photovoltaic generation was estimated for a set of specific panels, located in the south of the State of Minas Gerais. Therefore, a more comprehensive study, which considered a dataset of seventeen generation plants, with seasonal characteristics, was also examined. The versatility of the proposed method allowed changing the number of factors to be used in the experimental arrangement, in the forecasting model and in the desired forecasting horizon and, consequently, improving the determination of the forecast for the studied scenarios.

Keywords: *Photovoltaic Forecast; Design of Experiments; Artificial Neural Networks; Ensemble*

LISTA DE ILUSTRAÇÕES

Figura 2.1 – Exemplo esquemático de agrupamento de seis pontos de dados e sua representação no dendrograma.	16
Figura 2.2 – Analogia da sala: O analisador sabe que há uma restrição de recursos e tempo para varrer toda a sala para a possível correspondência paramétrica ideal. Mas, com base em experiências anteriores (ou pesquisas anteriores), ele sabe que existe uma região de busca reduzida que pode levar a uma boa solução para o problema (não necessariamente a ótima).	20
Figura 2.3 – Desenhos fatoriais e suas respectivas resoluções. Baseado no <i>software Minitab</i>	21
Figura 2.4 – <i>Design simplex</i> com três componentes(a) e com quatro componentes(b). Adaptado de (MONTGOMERY, 2017).	24
Figura 3.1 – Metodologia proposta para previsão de geração fotovoltaica (<i>PV</i>) usando rede neural artificial (RNA), planejamento de experimentos (<i>DOE</i>), análise de <i>cluster</i> e <i>mixture DOE</i> (MDE).	26
Figura 3.2 – Estrutura do conjunto de dados: sequencial ou desempilhado por dia da semana.	28
Figura 3.3 – Iterações do algoritmo de treinamento da RNA usando estrutura de dados sequencial ou desempilhada. Três entradas para uma saída.	29
Figura 3.4 – Pseudocódigo da metodologia proposta, exibindo cada instância anteriormente apresentada. A implementação original pode ser observada no Apêndice A deste documento.	33
Figura 3.5 – Metodologia proposta para previsão de geração fotovoltaica (<i>PV</i>) usando análise de cluster, análise de componentes principais (<i>PCA</i>), rede neural artificial (RNA), planejamento de experimentos (<i>DOE</i>) e <i>mixture DOE</i> (MDE).	34
Figura 3.6 – Quatro subdivisões dos períodos do ano para o território brasileiro: verões quentes e chuvosos, invernos secos e outonos com predominância de vento.	35
Figura 3.7 – Séries temporais consideradas: geração fotovoltaica (kWh) e meteorológicas (colunas de 2 a 17).	36
Figura 3.8 – Reorganizando os dados de geração para treinamento: empilhando dados para treinamento com base nas informações dos 6 dias anteriores.	38
Figura 3.9 – Pseudocódigo proposto para automatizar o processo de previsão fotovoltaica.	40

Figura 4.1 – Painéis fotovoltaicos instalados sobre o Laboratório de Alimentos do Instituto Federal do Sul de Minas, Campus Carmo de Minas	41
Figura 4.2 – Gráfico de contorno para visualização semanal de geração de energia fotovoltaica de uma planta localizada em Minas Gerais, Brasil.	42
Figura 4.3 – Efeitos principais para o <i>MAPE</i>	46
Figura 4.4 – Dendrograma usando ligação de <i>Ward</i> com distância euclidiana.	47
Figura 4.5 – Análise variância usando o procedimento de comparação de <i>Tukey</i>	47
Figura 4.6 – Tetraedro representando uma configuração de pesos que compõe a combinação $(w_1, w_2, w_3, w_4) = (0,625, 0,125, 0,125, 0,125)$	49
Figura 4.7 – Elipsóide de confiança para a semana de previsão desejada.	50
Figura 4.8 – Visão geral do percentual de representação da variância das variáveis climáticas, relacionadas a cada usina de geração fotovoltaica, considerando dois componentes principais.	51
Figura 4.9 – Visão geral do percentual de representação da variância das variáveis climáticas, relacionadas a cada usina de geração fotovoltaica, considerando três componentes principais.	51
Figura 4.10–Gráfico de efeitos principais para cada estação do ano, considerando o erro médio de todas as dezessete plantas de geração.	57
Figura 4.11–Análise variância usando o procedimento de comparação de <i>Tukey</i> para cada estação do ano.	58
Figura 4.12–Combinação de pesos ajustada que configura os menores erros de previsão para as dezessete usinas de geração, separadas por estação do ano: (A) Outono - $(0,33; 0,33; 0,33)$; (B) Inverno - $(0,0; 0,0; 0,5; 0,5)$; (C) Verão - $(0,0; 0,5; 0,5)$ e Primavera, que não é plotado porque os elementos de combinação continham 5 fatores (difícil de ver): $(0,0; 0,0; 0,33; 0,33; 0,33)$	60
Figura 4.13–Mapa de intensidade dos erros por estação do ano: os resultados apresentados são os <i>MAPEs</i> mínimos encontrados para cada usina de geração, após o processo de combinação dos resultados. Os mapas foram gerados utilizando a linguagem <i>JavaScript</i> , através da biblioteca open source <i>jQuery MAPAEL</i> (BROUTÉ, 2013).	62

LISTA DE TABELAS

Tabela 2.1 – Sumário de trabalhos relacionados (Parte 1).	8
Tabela 2.2 – Sumário de trabalhos relacionados (Parte 2 - continuação).	9
Tabela 3.1 – Representação matemática e gráfica das funções de transferência, de acordo com o catálogo do <i>Matlab</i>	29
Tabela 3.2 – Fatores do <i>DOE</i> para a RNA com dois níveis cada.	30
Tabela 3.3 – Fatores do <i>DOE</i> para a série temporal com dois níveis cada.	31
Tabela 3.4 – Fatores do <i>DOE</i> para a série temporal com dois níveis cada.	39
Tabela 3.5 – Fatores do <i>DOE</i> para a RNA com dois níveis cada.	39
Tabela 4.1 – Análise do coeficiente de correlação de Pearson em variáveis exógenas. Os resultados são estatisticamente significativos com um valor $p < 0,05$	42
Tabela 4.2 – DOE fatorial fracionário	44
Tabela 4.3 – Erros percentuais médios absolutos (MAPEs) associados a cada dia da semana e MAPE médio.	45
Tabela 4.4 – <i>DOE</i> de misturas para a definição dos pesos do <i>ensemble</i>	49
Tabela 4.5 – Detalhamento das plantas de geração fotovoltaica utilizadas.	53
Tabela 4.6 – Estrutura da Matriz Experimental do DOE.	54
Tabela 4.7 – Resultados da previsão em termos de seus erros médios (<i>MAPEs</i>) após cada execução experimental, separados por estação do ano.	56
Tabela 4.8 – Arranjo <i>DOE</i> de misturas considerando três elementos a serem combinados. Os resultados do outono e inverno estão listados.	59
Tabela 4.9 – Arranjo <i>DOE</i> de misturas considerando quatro elementos a serem combinados. Os resultados do inverno estão listados.	60
Tabela 4.10 – Arranjo <i>DOE</i> de mistura considerando quatro elementos a serem combinados. Os resultados da primavera estão listados.	61

LISTA DE ABREVIATURAS E SIGLAS

DOE	Planejamento de Experimentos
INMET	Instituto Nacional de Meteorologia
MAPE	Erro Percentual Médio Absoluto
MDE	Planejamento de Experimentos do Tipo Mistura
MLP	<i>Perceptron</i> de Múltiplas Camadas
NWP	Predição Numérica Climática
PCA	Análise de Componentes Principais
PV	Painel Fotovoltaico
RBF	Função de Base Radial
RNA	Rede Neural Artificial
SARIMA	Modelo Auto-Regressivo Integrado de Médias Móveis
SCG	Gradiente Conjugado Escalonado
SVM	Máquinas de Vetores Suporte

SUMÁRIO

	Agradecimentos	v
	Resumo	vii
	Abstract	viii
	Lista de Ilustrações	viii
	Lista de Tabelas	x
	Lista de Abreviaturas e Siglas	xii
1	INTRODUÇÃO	1
1.1	Justificativa e Relevância	1
1.2	Objetivo Geral	4
1.2.1	Objetivos Específicos	4
1.3	Limitações do Trabalho	4
1.4	Originalidade do Tema e Contribuições	5
1.5	Estrutura do Documento	5
2	FUNDAMENTAÇÃO TEÓRICA	7
2.1	Classificação dos Métodos de Previsão	14
2.2	Pré-Processamento dos Dados	15
2.3	Análise Hierárquica de <i>Cluster</i>	15
2.4	<i>Ensemble</i> de Redes Neurais Artificiais	16
2.5	Planejamento de Experimentos Fatorial	19
2.6	Análise de Componentes Principais - <i>PCA</i>	21
2.7	<i>Mixture DOE</i> para a composição dos pesos do <i>ensemble</i>	22
3	METODOLOGIA	25
3.1	Variáveis Climáticas Escolhidas por Correlação	25
3.1.1	Fatores da Série Temporal	25
3.1.2	Fatores da Rede Neural Artificial (RNA)	28
3.1.3	Planejamento de Experimentos Fatorial	30
3.1.4	Análise de <i>cluster</i> para a seleção das RNA's	31
3.1.5	<i>Mixture DOE</i> para a composição dos pesos do <i>ensemble</i>	31
3.1.6	Elipsóide de confiança	31
3.2	Variáveis Climáticas Considerando Redução de Dimensionalidade	34

3.2.1	Preparação dos Dados Sazonais	35
3.2.2	Agrupamento de Dias Semelhantes	37
3.2.3	Redução de Dimensionalidade usando <i>PCA</i>	37
3.2.4	Parametrização da RNA	38
3.2.5	<i>DOE</i> Fatorial	39
3.2.6	<i>Mixture DOE</i> para definição dos pesos do <i>Ensemble</i>	40
4	ESTUDO DE CASO	41
4.1	Usina de Geração Fotovoltaica em Carmo de Minas/MG	41
4.1.1	Previsão Fotovoltaica Através de RNA e <i>DOE</i>	43
4.1.2	Combinação dos Resultados	48
4.1.3	Região de Confiança	48
4.2	Conjunto de Usinas Localizadas em Diferentes Regiões do País	50
4.2.1	Estágio de Preparação do Experimento	52
4.2.2	Previsão Sazonal de Um Dia à Frente	55
4.2.3	Previsão Combinada - <i>Ensemble</i>	58
5	CONCLUSÕES	64
5.1	Conclusões gerais	64
	REFERÊNCIAS	66
	APÊNDICES	78
	APÊNDICE A – IMPLEMENTAÇÕES DE CÓDIGO USANDO MA- TLAB	79
	APÊNDICE B – ARTIGO APROVADO	101

1 INTRODUÇÃO

1.1 Justificativa e Relevância

Por muitas décadas, o fluxo unidirecional de energia elétrica dos geradores para a transmissão, distribuição e consumo no usuário final permaneceu praticamente inalterado. O sistema clássico e tradicional das grandes centrais que levam a energia elétrica até os centros consumidores, através das linhas de transmissão e distribuição, poderá ser alterado, potencialmente e gradativamente, uma vez que a produção está passando a existir próximo ao local do consumo. O aumento da participação das energias renováveis na matriz elétrica em todo o mundo é uma demanda de interesse econômico, social e ambiental (DANESHI, 2018; ZEREN; AKKUŞ, 2020). As amplas perspectivas para a substituição do uso de combustíveis fósseis reforçam o potencial desses recursos renováveis para o fornecimento de energia elétrica (DU *et al.*, 2019).

Nessa perspectiva, as redes de distribuição não foram projetadas para suportar a inserção de unidades de geração (CUK *et al.*, 2011), mas baseiam-se em um fluxo de potência unidirecional que transportam a energia de uma subestação para o consumidor final (MONTEIRO *et al.*, 2017). Portanto, é essencial que haja uma mudança significativa no paradigma do sistema de suprimento de energia pelos seguintes motivos: (a) a escassez de combustíveis fósseis, que leva a problemas ambientais e energéticos e (b) a energia renovável atraiu o interesse de muitos investidores em diferentes regiões nos últimos anos (SUN *et al.*, 2018).

Alguns autores (WANG *et al.*, 2022b) citam a energia solar como o principal foco dos investidores nos tempos recentes. Com o advento das redes inteligentes, e também para lidar com mudanças inesperadas nas condições climáticas, as previsões de geração fotovoltaica (PV) são fundamentais para gerenciar redes de distribuição, micro-redes ou residências inteligentes (AGOUA *et al.*, 2018). O desempenho do sistema, bem como as decisões operacionais, podem ser aprimorados com base nas previsões de carga e de geração (MICHAELSON *et al.*, 2017). A eficácia desses modelos impacta na eficiência e segurança do sistema, e as medições fornecem informações confiáveis para clientes e fornecedores de energia (YOUSEFI *et al.*, 2019).

Infelizmente, incertezas relacionadas à natureza intermitente da energia solar têm um impacto negativo na operação eficiente, confiável e segura dos sistemas de energia elétrica (BRACALE *et al.*, 2017), de maneira que diversos autores apontam que os fatores meteorológicos e as condições de infraestrutura das redes de distribuição são aspectos que influenciam fortemente no uso eficiente da energia solar como fonte alternativa (SOBRI *et*

al., 2018; TOLLEFSON, 2019). A influência de alguns fatores meteorológicos foi testada em diversos estudos, o que reforçou a necessidade urgente de propor métodos de monitoramento de fenômenos e modelos robustos de previsão (LI *et al.*, 2020; WANG *et al.*, 2016). Uma maneira de lidar com esse problema é definir previsões precisas (HUANG *et al.*, 2022). Os principais benefícios de uma previsão mais precisa destacam as chances de evitar elevações de tensão local (LIN *et al.*, 2022a), observadas quando a geração fotovoltaica é maior que a demanda (DALL'ANESE *et al.*, 2015).

Em comparação aos erros de previsão de carga, que geralmente são de aproximadamente 1% a 3% (ZHANG *et al.*, 2015), os erros de geração (solar e eólica) são significativamente mais substanciais, atingindo valores médios de 15% a 20% (ZHANG *et al.*, 2015). Alguns países, como exemplo a Espanha, fornecem incentivos para melhores previsões do nível de geração solar no dia seguinte (LORENZ *et al.*, 2009). Sabendo que a geração fotovoltaica é altamente incerta e difícil de prever (HUANG *et al.*, 2015), é de extrema importância classificar os horizontes de previsão (YANG; HUANG, 2018): previsões de longo prazo para períodos superiores a 1 mês; previsões de médio prazo para um intervalo de 1 a 7 dias; e previsões de curto prazo para algumas horas à frente. Os horizontes de curto e médio prazo são os mais estudados, devido às incertezas relacionadas à dependência do clima e à consequente diminuição da confiabilidade à medida que o horizonte desejado aumenta.

A importância de cada horizonte pode ser quantificada de acordo com as ações que devem ser tomadas (ou evitadas) em diferentes momentos do tempo. As previsões de curto prazo são úteis para lidar com sobretensões (ZHANG *et al.*, 2015), que podem não apenas comprometer a segurança da rede em termos de sobrecarga de equipamentos, mas também podem causar danos permanentes a motores, equipamentos eletrônicos e eletroeletrônicos (MONTEIRO *et al.*, 2017). As previsões de médio prazo, que são o escopo deste trabalho, são usadas pelos operadores para determinar requisitos de reserva, despachos de armazenamento e qualidade de energia (ZHANG *et al.*, 2019). As previsões de longo prazo são úteis para decisões de mercado (LEE *et al.*, 2018) e investimentos setoriais (DAS *et al.*, 2018).

Existem várias metodologias para a previsão da irradiância solar, sendo as mais comuns as analíticas, estocásticas, empíricas, modelos estatísticos e redes neurais artificiais (Fontes Rodrigues *et al.*, 2022). Os modelos analíticos baseiam-se na observação de fenômenos físicos e da interação entre eles, como por exemplo a análise da radiação solar e seus impactos na atmosfera terrestre. Os modelos empíricos são aplicáveis quando não há possibilidade de medição direta da radiação solar. Desse modo, outros parâmetros climáticos são usados como base, como duração da luz solar, temperatura e nebulosidade. Os modelos estocásticos presumem processos lineares de previsão, a exemplo dos métodos com componentes auto-regressivos e médias móveis (VOYANT *et al.*, 2018).

Contudo, a não-linearidade dos dados e fenômenos meteorológicos podem comprometer a precisão dos modelos de previsão. Sendo assim, o uso das redes neurais artificiais e de técnicas de aprendizagem de máquina têm se mostrado eficazes em estudos de previsão. As redes neurais artificiais (RNAs) foram escolhidas como modelo de previsão para serem utilizadas neste trabalho devido à sua importância na literatura, pois mesmo estudos que não aplicam diretamente as RNAs mencionam que estas desempenham um papel importante na previsão de geração de PV (DAS *et al.*, 2018; VISCONDI; ALVES-SOUZA, 2019; SOBRI *et al.*, 2018; ACKERMANN *et al.*, 2013).

As redes neurais artificiais têm sido amplamente utilizadas para prever a geração de energia fotovoltaica (HOSSAIN *et al.*, 2017; TEO *et al.*, 2015). Algumas pesquisas apontaram as vantagens da análise de *Big Data* para previsão de energia renovável (CHAI *et al.*, 2020; ZHEN *et al.*, 2020; GENSLER *et al.*, 2016). Alguns trabalhos (CUI *et al.*, 2019; GÓMEZ-QUILES *et al.*, 2019; ZHENG *et al.*, 2019) apresentaram soluções em previsão baseadas em aprendizado de máquina supervisionado ou não supervisionado, como o modelo de média móvel integrado autoregressivo, a previsão em sequência de padrões, o modelo de RNA.

Duas questões chamam a atenção neste cenário: a) Os processos simulados muitas vezes requerem um número elevado de combinações para testar todas as soluções possíveis; e b) os modelos de previsão podem gerar mais de uma solução potencial para o problema e não se sabe, com certeza, qual é a melhor. A originalidade da metodologia proposta nesta tese, aplicada à previsão de geração fotovoltaica, visa cobrir essas lacunas e é sustentada por dois pilares principais. O primeiro é, sabendo que a arquitetura de processamento dos computadores atuais é limitada, a aplicação do *DOE* para reduzir o número de combinações relacionadas à parametrização da RNA é atrativa, mantendo a confiabilidade estatística. A segunda está relacionada a como escolher ou combinar os resultados da previsão. Neste caso, propõe-se aplicar a análise hierárquica de *clusters* para selecionar as melhores redes e então utilizar *Mixture DOE* para realizar uma previsão por conjuntos ou *ensemble*.

Este trabalho apresenta uma proposta metodológica que promove avanços nos estudos e na prática de uso de modelos de previsão baseados em aprendizado de máquina para previsões. Diante do desafio de garantir maior precisão, o modelo proposto traz contribuições na medida em que reduz o tempo de treinamento e os custos computacionais, bem como otimiza hiperparâmetros dos algoritmos e modela as características temporais complexas.

Como a previsão deve fornecer clareza e confiabilidade (BRACALE *et al.*, 2017), o método de previsão proposto neste trabalho é baseado em uma abordagem de projeto de experimentos (*DOE*), para auxiliar na escolha de fatores (da série e do modelo de previsão) que podem resultar em estimativas mais precisas. O *DOE* é uma ferramenta estatística na

qual cada execução experimental é um teste e permite ao investigador descobrir algumas informações sobre um processo ou sistema (MONTGOMERY, 2017).

Na sequência, as melhores configurações observadas na abordagem *DOE* são mantidas, através de uma análise de *cluster*, para formar uma previsão combinada. Uma previsão por conjunto (*ensemble*) tende a melhorar os resultados dos modelos individuais (GALICIA *et al.*, 2019). A combinação proposta considera que a definição dos pesos do conjunto é calculada por uma análise de mistura.

A metodologia é subdividida em dois processos, de maneira que o primeiro considera o uso da variável climática através da análise de correlação e o segundo considera dados meteorológicos por meio da redução de dimensionalidade através da análise de componentes principais (PCA).

1.2 Objetivo Geral

Desenvolver uma metodologia para previsão de geração fotovoltaica através da combinação de redes neurais artificiais parametrizadas por planejamento de experimentos.

1.2.1 Objetivos Específicos

Diante do objeto de estudo, os objetivos específicos deste trabalho incluem:

- Elencar os níveis mais importantes dos fatores do DOE, tendo como base trabalhos relacionados da comunidade científica.
- Automatizar a execução da previsão através da implementação de algoritmo que receba a matriz DOE e execute cada corrida experimental.
- Promover a Análise Hierárquica de *Cluster* a fim de selecionar as redes com melhores resultados de previsão.
- Combinar os resultados por meio da Análise de Misturas para a definição dos pesos.
- Construir região de confiança para os resultados obtidos, quando aplicável.
- Reduzir a dimensionalidade dos dados, quando aplicável, usando Análise de Componentes Principais.

1.3 Limitações do Trabalho

A implementação dessa metodologia deve levar em conta algumas considerações empiricamente relacionadas a fatores não controláveis, que devem ser observadas com atenção:

- (a) Depósito de resíduos nos painéis, como poeira; sensores climáticos danificados (ou mesmo quando não há proximidade com a usina de geração); indisponibilidade de informações ou dados (valores nulos - isso compromete o aprendizado e a precisão dos modelos de previsão); intensidade e frequência de cobertura de nuvens (há indícios de que alguns tipos de painéis fotovoltaicos produzem mais energia quando a nuvem se afasta devido ao resfriamento temporário das células durante o período de sombreamento); velocidade do vento (a intensidade do vento pode dissipar o calor, o que aumenta a eficiência dos painéis). Esses fatores podem interferir no resultado da previsão, naturalmente, pois leva a representar níveis de geração fotovoltaica que não condizem com o esperado;
- (b) Uma vez que o DOE leva a uma redução do esforço computacional, não há garantia de sempre obter a melhor solução de previsão (assim como nenhum outro método de previsão, apresentado na literatura, é capaz de fazê-lo).

1.4 Originalidade do Tema e Contribuições

As contribuições desta pesquisa para a comunidade científica e para o setor elétrico, no que tange à previsão de geração fotovoltaica, incluem:

- (a) Redução do esforço computacional em termos do número de simulações, sem comprometer a confiabilidade estatística, através da modelagem baseada em *Design of Experiments*;
- (b) Aplicação da Análise Hierárquica de *Cluster* para selecionar os modelos que obtiveram melhor performance nas corridas experimentais;
- (c) Aplicação da Análise Hierárquica de *Cluster* para separar os dias de medição com características climáticas semelhantes;
- (d) Combinação dos resultados através da definição dos pesos por meio de *Mixture Analysis*;
- (e) Verificação da similaridade estatística dos resultados em relação ao conjunto de dados por meio da construção de um elipsóide de confiança, quando aplicável;
- (f) Redução da dimensionalidade do conjunto de dados usando Análise de Componentes Principais.

1.5 Estrutura do Documento

Esta tese está organizada em cinco capítulos:

- **Capítulo 2:** apresenta um levantamento de trabalhos publicados na comunidade científica, relacionados ao tema da pesquisa, elencando também alguns dos principais métodos de previsão que são aplicados neste contexto.
- **Capítulo 3:** descreve a originalidade da metodologia para previsão de geração de energia fotovoltaica e detalha cada etapa do processo.
- **Capítulo 4:** apresenta dois estudos de caso para a previsão de geração fotovoltaica a partir de dados de geração recentes. As séries de dados foram obtidas considerando medições de plantas de geração de várias regiões do país.
- **Capítulo 5:** contempla as conclusões do trabalho bem como as expectativas de experimentação para novos estudos de caso usando a metodologia de previsão proposta.
- **Apêndice A:** apresenta as implementações de código da metodologia apresentada usando a linguagem de programação do software *MATLAB*.
- **Apêndice B:** folha de aprovação do *journal* no qual esta pesquisa foi publicada.

2 FUNDAMENTAÇÃO TEÓRICA

Esta Seção investigou alguns dos trabalhos relacionados mais relevantes, bem como as lacunas a serem observadas:

- Muitos desses trabalhos não exploram o potencial da previsão combinada;
- Na maioria dos casos, os parâmetros do modelo de previsão não são definidos formalmente, mas escolhidos de forma empírica, fixa, aleatória ou mesmo por varredura de uma faixa de valores. Quando isso acontece, não é possível identificar quais parâmetros influenciam o resultado, bem como pode exigir um esforço computacional considerável ao processar inúmeras possibilidades. Observou-se também que alguns estudos focam principalmente na previsão de radiação solar e não na previsão de geração fotovoltaica, pois pode haver bastante diferença na aderência dos modelos de previsão em termos de fatores incontroláveis, como poeira nos painéis, sensores danificados, eficiência do painel, etc.

As Tabelas 2.1 e 2.2 resumem os principais trabalhos, apresentando as características predominantes de cada um, além de um comentário crítico sobre questões que não foram abordadas ou não foram evidenciadas.

ZHEN *et al.* (2020) propuseram um modelo híbrido, baseado em redes neurais convolucionais (CNN), *Long Short-Term Memory (LSTM)* e *Artificial Neural Network (ANN)*, para previsão de energia fotovoltaica em tempo real. O método de normalização não foi discutido pelos autores. Primeiro, a etapa de pré-processamento da imagem do céu requer alto esforço computacional para extrair características. Em seguida, alguns parâmetros dos modelos de previsão são definidos pela varredura de uma faixa de valores. Por ser uma operação em tempo real (15 min), pode não ser uma operação interessante se o modelo tiver que ser recalibrado. Os parâmetros ajustados para a rede neural foram as camadas ocultas e o número de neurônios. Não foi possível ver como a interação entre os parâmetros do modelo afeta o resultado.

THEOCHARIDES *et al.* (2020) apresentaram a previsão de energia fotovoltaica para um dia de antecedência, com resolução horária. O modelo é baseado em Redes Neurais Artificiais e usa o método de correção regressiva linear para ajustar os resultados de previsão usando irradiância solar. Nenhum método de normalização de dados é discutido pelos autores. A arquitetura do modelo de previsão foi configurada com base nos dados de entrada, o que significa que foi realizada uma varredura empírica para encontrar os melhores valores paramétricos em um intervalo de dados. O pós-processamento combina agrupamento K-means e um modelo linear regressivo.

Tabela 2.1 – Sumário de trabalhos relacionados (Parte 1).

Autor e Ano	Método de Previsão	Horizonte de Previsão	Variáveis Exógenas	Parametrização	Comentário
(ZHEN <i>et al.</i> , 2020)	Modelo híbrido baseado em ANN, CNN e LSTM	Minuto	Imagem do céu e irradiância solar	Número de camadas e neurônios	Alguns parâmetros dos modelos de previsão não foram especificados. Alguns deles são definidos pela varredura de um intervalo de valores.
(THEOCHARIDES <i>et al.</i> , 2020)	RNA	Dia à frente, horário	Irradiância global incidente, temperatura ambiente, umidade relativa, direção e velocidade do vento, azimute solar e ângulos de elevação	Neurônios, camadas e épocas.	Os parâmetros foram definidos pela varredura de um intervalo de valores.
(SANGRODY <i>et al.</i> , 2020)	Modelos de previsão baseados em similaridade (SBFMs)	Dia à frente, minuto	Dados de temperatura, umidade, ponto de orvalho, velocidade do vento, irradiância e cobertura do céu	Baseado em k-vizinhos mais próximos (KNN). O método <i>Grid Search</i> é aplicado para encontrar k ótimo.	Em um certo ponto, a combinação rendeu erros maiores. A configuração paramétrica do modelo não é detalhada.
(PAN; TAN, 2019)	Baseado em análise de <i>cluster</i> e regressão de <i>ensemble</i> .	Dia à frente, horário	Doze variáveis climáticas como pressão de superfície, umidade relativa, cobertura de nuvens, componente do vento, temperatura, irradiação solar, precipitação, etc.	Hiperparâmetros selecionados usando o método <i>Grid Search</i> .	Clusters baseados em variáveis climáticas exigem maior esforço computacional.
(WEN <i>et al.</i> , 2019)	Rede neural recorrente profunda com unidades de memória de longo prazo (DRNN-LSTM)	Mês à frente, horário	Radiação horizontal global e radiação horizontal difusa	Otimização de hiperparâmetros bayesianos.	A parametrização automatizada é eficiente, mas ainda requer considerável esforço de processamento quando cada previsão é realizada. A previsão combinada não é explorada.
(OZEGWU, 2019)	RNA	Mês a dois anos à frente	Sol, temperatura, nebulosidade, precipitação, umidade relativa, ponto de orvalho, temperatura, temperatura do solo, evaporação e pressão.	Fixo: Uma camada escondida com 20 neurônios. Algoritmo de treinamento de Levenberg-Marquardt. Funções de transferência tangente sigmóide e linear.	A combinação é realizada entre os modelos e não entre os resultados de diferentes preditores. Aqui, um novo modelo híbrido é gerado.

Tabela 2.2 – Sumário de trabalhos relacionados (Parte 2 - continuação).

Autor e Ano	Método de Previsão	Horizonte de Previsão	Variáveis Exógenas	Parametrização	Comentário
(QING; NIU, 2018)	Redes <i>Long Short-term Memory (LSTM)</i> e RNA usando o clássico algoritmo backpropagation (BPNN)	Dia à frente, horário	Temperatura, ponto de orvalho, umidade, visibilidade, velocidade do vento e resumo descritivo do clima	Vária, dependendo do benchmark.	Alguns parâmetros são definidos aleatoriamente e outros são encontrados combinando muitas execuções, o que aumenta o processamento. A previsão combinada não é explorada.
(BUGALA et al., 2018)	RNA	Dia à frente, horário	Horas de sol, duração do dia, pressão atmosférica, temperatura máxima do ar, insolação diária e nebulosidade	Os parâmetros foram estimados usando um algoritmo heurístico.	Para cada rodada de previsão, os parâmetros devem ser estimados por meio de heurísticas, o que aumenta o esforço computacional.
(DEO; ŞAHIN, 2017)	RNA	Sazonal e mensal	Dados derivados de satélite e temperatura da superfície terrestre.	Vários parâmetros foram testados. A arquitetura do modelo pode variar (55 para previsão mensal e 9 para previsão sazonal).	Embora o modelo seja parcimonioso, há uma variação na arquitetura do modelo que enfatiza uma variedade de parâmetros, como algoritmo de treinamento e função de ativação. Não é indicado como a relação entre os parâmetros influencia no resultado.
(SIVANEASAN et al., 2017)	RNA e lógica fuzzy para pré-processamento de dados meteorológicos	Mensal	Cobertura de nuvens, temperatura, velocidade do vento e direção do vento com valor de irradiância.	Fixo: algoritmo de treinamento Levenberg-Marquardt, 25 neurônios escondidos e tangente sigmóide como função de ativação.	Não há indicação se a alteração de um determinado parâmetro melhora ou piora os resultados. A previsão combinada não é explorada.
(CERVONE et al., 2017)	RNA e <i>Ensemble</i> análogo	Três dias à frente, horário	Irradiância horizontal global, cobertura de nuvens, temperatura do ar, azimute solar e elevação.	De 4 a 20 neurônios ocultos.	A RNA é inicializada várias vezes para a mesma estação e os parâmetros são definidos pela busca de um intervalo de valores.
(LIMA et al., 2016)	RNA	Dia à frente, horário	Umidade relativa, temperatura, velocidade do vento, cobertura de nuvens, água precipitável.	Uma camada oculta, tangente hiperbólica como função de ativação, Levenberg-Marquardt como algoritmo de treinamento.	Os parâmetros foram definidos pela varredura de um intervalo de valores. Alguns deles não são discutidos ou apresentados.

SANGRODY *et al.* (2020) propuseram a previsão da geração fotovoltaica usando métodos de previsão semelhantes escolhidos através do método dos k-vizinhos mais próximos (KNN). A normalização dos dados leva em consideração o método de dimensionamento de recursos, que considera os valores máximo e mínimo da série temporal. A configuração paramétrica do modelo de previsão não é detalhada e é definida pelo método *grid search*. Em determinado momento, os autores relatam que a precisão da previsão está estritamente relacionada à escolha das variáveis climáticas e que o processo de combinação de alguns resultados não promoveu melhorias na previsão.

O método de previsão proposto por PAN; TAN (2019) faz uso da análise de *cluster* para classificar as características meteorológicas e, em seguida, usa a regressão de cristas para determinar os pesos do *ensemble*. Foi abordado o horizonte de previsão para um dia à frente, com resolução horária. O método de normalização utilizou os valores máximo e mínimo da série temporal. Discutiu-se que, na primeira etapa (análise de *cluster*), tanto os dados de geração quanto apenas os dados climáticos podem ser utilizados. No entanto, para este último, o custo computacional em termos de processamento pode ser consideravelmente aumentado. Os hiperparâmetros foram selecionados usando o método de busca em grade.

WEN *et al.* (2019) realizaram uma previsão integrada envolvendo carga e geração fotovoltaica. Os dados foram normalizados utilizando valores máximos e mínimos. Os experimentos foram implementados na linguagem de programação Python e, para definir os parâmetros do modelo de previsão, é utilizado um método automatizado que investiga o espaço de busca utilizando otimização Bayesiana. É claro que esse tipo de pesquisa é mais eficiente que uma varredura manual ou aleatória, mas ainda consome processamento para cada previsão realizada. A previsão fotovoltaica é estimada de hora em hora para o período de um mês à frente.

O autor OZOE GWU (2019) desenvolveu uma metodologia com o objetivo de prever a radiação solar para um horizonte de um ano à frente da média mensal diária. Esta previsão é útil para o dimensionamento de energia fotovoltaica e também para atividades agrícolas. Percebe-se que o estudo de caso, realizado na Nigéria, envolveu a comparação de diversas arquiteturas de redes neurais, implementadas pelos autores utilizando o software Matlab, com parametrização fixa. A ideia dos autores centra-se na combinação dos modelos de previsão e não nos resultados. Este modelo híbrido usa o número do mês como parte das entradas no processo de previsão solar de longo prazo.

QING; NIU (2018) realizaram uma comparação entre as redes de *Long Short-term Memory (LSTM)* e os modelos de RNA usando o algoritmo clássico de retropropagação do erro (BPNN) para previsão de irradiância solar. Dependendo do *benchmark*, houve variação no número de épocas, no número de camadas ocultas e no número de neurônios por camada. Assim, houve uma varredura para testar uma gama de parâmetros em cada

um, o que levou a um aumento no processamento para encontrar os valores ideais. O método de normalização de escala linear foi utilizado no pré-processamento das entradas. As variáveis exógenas aqui são previsões meteorológicas horárias para um dia específico e serão usadas para geração de previsão fotovoltaica. A configuração completa dos modelos de previsão não foi apresentada e alguns parâmetros foram definidos através da execução empírica para atingir um determinado valor.

DEO; ŞAHIN (2017) propuseram uma metodologia de previsão de radiação solar para territórios que possuem cobertura de dados de satélite. A normalização dos dados de entrada ocorreu por meio de valores máximos e mínimos da série temporal. A implementação ocorreu através do *software* Matlab e vários parâmetros foram testados, como algoritmo de treinamento e função de ativação. A arquitetura de rede pode variar dependendo do contexto de previsão (mensal ou sazonal). Embora o modelo de rede neural seja parcimonioso e explique os dados com um número mínimo de parâmetros ou variáveis preditoras, houve uma intensa variação paramétrica para cada arquitetura que é gerada e, conseqüentemente, aumenta o esforço computacional em termos de processamento a cada vez que a previsão é realizada. Não é indicado como a relação entre os parâmetros influencia no resultado.

O trabalho de SIVANEASAN *et al.* (2017) aplicou a RNA para avaliar a previsão diária para um período de um mês à frente. A precisão nos resultados deveu-se à aplicação da lógica *fuzzy* no pré-processamento das entradas. Os autores definiram a parametrização do modelo neural de forma fixa, com 8 neurônios na camada de entrada, 25 neurônios ocultos, tangente sigmóide como função de ativação e Levenberg-Marquardt como algoritmo de treinamento. Um período de três meses (janeiro a março de 2017) de dados climáticos foi escolhido para treinar o modelo. Não foi possível identificar no trabalho se esses parâmetros foram definidos aleatoriamente ou se foram encontrados a partir do processamento intensivo de várias combinações de execuções. Não é discutido no artigo se a mudança em determinados parâmetros do modelo pode alterar os resultados ou não.

Com base em dados meteorológicos reais e dados de geração fotovoltaica de três usinas de energia na Itália, CERVONE *et al.* (2017) propuseram uma previsão para 72 horas à frente usando uma RNA e um *ensemble* análogo, calculado a partir de uma métrica que avalia a similaridade de previsões entre diferentes janelas de tempo no mesmo conjunto de dados. Os autores consideraram a solução escalável e produziram resultados mais confiáveis quando os métodos foram combinados. No entanto, a definição dos pesos dessa combinação foi feita iterativamente, gerando aproximadamente 1002 e 3004 combinações, o que exigiu recursos computacionais extras com processamento paralelo para a obtenção dos resultados. A RNA foi inicializada várias vezes para a mesma estação e os parâmetros foram definidos pela busca de um intervalo de valores.

LIMA *et al.* (2016), sabendo que o Brasil tem um enorme potencial para produzir

energia fotovoltaica, realizaram um estudo para prever a irradiância solar, considerando um horizonte de 24 horas à frente. Os autores usaram análise de *cluster* para identificar regiões do mapa regional com características climáticas semelhantes. Vários parâmetros foram testados para cada uma das 110 estações a fim de estabelecer coerência entre os resultados e os valores de irradiação observados, o que demanda alto custo computacional. O estudo não considerou uma análise estatística da influência da variação dos parâmetros da rede neural na previsão, mas revelou que o processamento realizado com esse modelo aumenta a confiabilidade da previsão e gera resultados consistentes.

A seleção de variáveis externas ao conjunto de dados de geração fotovoltaica foi observada no trabalho de BUGALA *et al.* (2018), através do coeficiente de correlação de Pearson. Os autores analisaram sete variáveis: número de horas de sol, temperatura máxima do ar, intensidade diária de iluminação, nebulosidade, duração da incidência solar e pressão do ar. Os dois últimos foram considerados estatisticamente irrelevantes para o modelo de regressão linear e não foram utilizados. Para a rede neural, apenas a variável pressão do ar foi desconsiderada e uma rede neural do tipo função de base radial (*RBF*) foi gerada para a previsão, com seis neurônios na camada de entrada, cinco neurônios na camada oculta e um neurônio na camada de saída. O erro obtido pela rede *RBF* 6:6-5-1:1 foi consideravelmente baixo, levando os autores a recomendar sua aplicação para uma previsão para o dia seguinte. No entanto, o estudo não abordou os impactos preditivos de uma variação paramétrica na rede neural.

THOREY *et al.* (2018) usaram técnicas de previsão probabilística e conjuntos ponderados para gerar um horizonte temporal de 6 dias para 219 usinas de geração fotovoltaica na França. Segundo os autores, as previsões do estudo de caso, que consideraram uma combinação ponderada, foram menos sujeitas a superestimação e alta dispersão.

Através da previsão usando modelos de seis famílias diferentes, como a média móvel integrada autoregressiva sazonal (*SARIMA*) e a RNA, o trabalho YANG; DONG (2018) sugeriu oito maneiras de combinar os resultados das melhores previsões individuais para cada família. A análise empírica dos autores encontrou combinações por meio de regressão através de mínimos quadrados comuns e um laço de regressão. Os autores consideraram o modelo de *perceptron* multicamadas (*MLP*) em sua configuração paramétrica padrão e afirmaram que o desempenho da rede do tipo *MLP* foi menor que o dos outros modelos em estudo, com uma indicação potencial de excesso de ajuste. No entanto, os autores também explicaram que a exploração paramétrica para calibrar o modelo não era o foco do trabalho.

Alguns autores (AMROUCHE; PIVERT, 2014) consideraram prever a irradiação de uma região com base nas previsões de irradiação para locais vizinhos, emitidas por entidades meteorológicas. Nesse caso, a solução proposta por AMROUCHE; PIVERT (2014), baseada em redes neurais, pode ser considerada evolutiva, pois o modelo é re-

alimentado com previsões anteriores à medida que avança cronologicamente no tempo. Os autores consideraram apenas variar o número de entradas da rede e fixaram os valores de parâmetros como o número de camadas ocultas, função de ativação e número de neurônios.

Em VISCONDI; ALVES-SOUZA (2019), os autores consideraram uma análise sistematicamente selecionada de 38 artigos. As pesquisas foram guiadas por palavras-chave como "*Big Data*", "*Data Mining*" e "*Machine Learning*", ou seja, todas relacionadas à previsão de geração de energia fotovoltaica. Em resumo, eles constataram que as redes neurais têm algoritmos de previsão mais precisos do que outros modelos.

O modelo de previsão de potência de saída fotovoltaica desenvolvido por HOS-SAIN *et al.* (2017) foi aplicado para previsão em curto prazo, especificamente, uma hora à frente. Nesse caso, os autores utilizaram o algoritmo *Extreme Learning Machine (ELM)*. Para previsão de potência de saída fotovoltaica do dia seguinte, os pesquisadores testaram o modelo usando radiação solar média diária (W/m^2), velocidade do vento (m/s), temperatura ambiente e do módulo ($^{\circ}C$). O modelo baseado em *ELM* foi comparado com outros dois modelos, um de regressão por vetores de suporte (*SVR*) e outro por RNA. Os resultados mostraram que o *ELM* apresentou maior precisão e menor tempo computacional na previsão em curto prazo da potência de saída fotovoltaica diária e horária.

No intuito de aprimorar o *ELM*, MISHRA *et al.* (2020) implantaram um novo modelo denominado *ELM Expandido (EELM)* para previsão de energia fotovoltaica. O *EELM* inova ao permitir a seleção automática do número da camada oculta e pesos de entrada aleatórios. Contudo, as capacidades de extrapolação mais altas do *EELM* só foram demonstradas para um horizonte de previsão de menos de 1 hora. Portanto, com base nos trabalhos de pesquisa elencados, a eficácia de um modelo de previsão de geração de energia fotovoltaica pode ser ainda mais precisa por meio da experimentação de cenários viáveis. Segundo YAO *et al.* (2022), o uso de métodos de aprendizado de máquina são muito eficazes para previsão de geração de energia fotovoltaica, dada a natureza não linear das variáveis. Entretanto, os autores indicam que sejam adotados métodos combinados a fim de capturar a característica estocástica da irradiância solar e a alta variabilidade das medições.

Uma revisão abrangente da literatura foi conduzida por DAS *et al.* (2018), onde os autores exploraram a previsão de geração de energia fotovoltaica a partir de diferentes perspectivas: (a) o pré-processamento das entradas do modelo através da normalização, (b) a correlação entre as variáveis de entrada e os dados de geração, (c) um curto horizonte de previsão, (d) análise do desempenho dos métodos, (e) detalhes dos critérios de avaliação de erros, (f) análise dos métodos de previsão e (g) trabalhos recentes a partir das perspectivas das técnicas utilizadas e a precisão dos resultados. Os autores concluíram que redes e modelos neurais baseados em máquinas de vetores de suporte (*SVM*) promo-

vem boa execução e têm boa aderência aos dados. A métrica mais frequentemente usada para avaliação de erros é o erro médio quadrático (*RMSE*), e o horizonte mais explorado para previsão é o curto prazo.

A revisão de literatura examinada por [SOBRI *et al.* \(2018\)](#) avaliou as diferentes técnicas de previsão de geração de energia fotovoltaica e os recentes progressos alcançados nessa área. A análise realizada pelos autores considerou fatores como o método de previsão, horizonte temporal e métrica de erro. Além de constatar que a aplicação dos métodos de RNA e Máquina de Vetores Suporte (*SVM*) é vantajosa para a solução de problemas não lineares, eles também mostraram que os modelos baseados em conjuntos tentam extrair a precisão e a robustez dos métodos individuais.

Por fim, a revisão de literatura apresentada por [BARBIERI *et al.* \(2017\)](#) teve como objetivo listar os métodos utilizados para prever energia fotovoltaica, bem como revisar os métodos estatísticos utilizados para esse fim. Além disso, os autores compararam os diferentes horizontes temporais em termos de desempenho e classificação. A configuração da hora é definida de acordo com a operação do sistema. Para horizontes de longo prazo, a sugestão é usar modelos baseados na previsão numérica do tempo (*NWP-Numerical Weather Prediction*).

A maioria dos estudos não abrange um horizonte de estudo superior a 48 horas ([ANTONANZAS *et al.*, 2016](#)) devido às incertezas climáticas, mas a importância das previsões para períodos de médio prazo está estritamente relacionada às operações de equilíbrio da rede, como gerenciamento de transmissão, despacho econômico e planejamento de manutenção de usinas ([AHMED *et al.*, 2020](#)).

2.1 Classificação dos Métodos de Previsão

Os métodos de previsão de geração de energia fotovoltaica são geralmente classificados como modelos físicos ou estatísticos ([MEER *et al.*, 2018](#)). Os modelos físicos descrevem o estado físico e as características da usina de geração, como localização, diferentes variáveis meteorológicas ([YANG; HUANG, 2018](#); [DAS *et al.*, 2018](#)), efeitos de sombra, tipo de módulo e ângulo de azimute / inclinação ([ZHANG *et al.*, 2019](#)). Os modelos estatísticos consideram o histórico dos dados e tentam extrair conhecimento do passado para prever uma série temporal ([ZHANG *et al.*, 2019](#); [SOBRI *et al.*, 2018](#)).

Segundo [SOBRI *et al.* \(2018\)](#), os modelos estatísticos incluem RNAs, *SVMs*, modelos de Cadeias de Markov, modelos autorregressivos e modelos de regressão. As RNAs têm sido extensivamente aplicada em muitos estudos sobre problemas de séries temporais não lineares (intrinsecamente associadas à previsão de geração de energia fotovoltaica) ([LEE *et al.*, 2018](#); [WANG; SHEN, 2019](#); [LIU *et al.*, 2015](#); [CERVONE *et al.*, 2017](#)). Mesmo os estudos que não utilizam redes neurais em suas análises mencionam que as RNAs são

uma importante técnica de previsão (VISCONDI; ALVES-SOUZA, 2019).

2.2 Pré-Processamento dos Dados

Basicamente, a coleta e preparação de séries temporais antecede a análise de dados. Esses dados muitas vezes são difíceis de serem obtidos devido à política de proteção de dados das usinas de geração locais (SRIVASTAVA; LESSMANN, 2018), o que pode comprometer os avanços no processo de previsão de geração fotovoltaica se esta etapa não for considerada seriamente (AKSU *et al.*, 2019). Esta etapa abrange a correção de dados ausentes, normalização de dados, ajuste de resolução de dados e agrupamento de dados (MASSAOUDI *et al.*, 2021). Uma etapa comumente abordada no pré-processamento de dados brutos é o método de normalização.

O processo de normalização é uma etapa importante para algoritmos de aprendizado de máquina. Em geral, o resultado desse processo visa redimensionar ou converter o conjunto de dados original para um novo padrão. Não há consenso na literatura quanto ao melhor método de normalização, mas alguns deles envolvem a transformação dos dados utilizando média, desvio padrão, valores máximos e mínimos. Alguns autores consideram que, além de melhorar a precisão dos algoritmos de previsão, o processo de normalização também contribui positivamente para uma melhoria no desempenho (SINGH; SINGH, 2019). Nesse caso, o objetivo principal é reduzir a magnitude e manter os dados em uma escala próxima à das funções de transferência da rede neural.

2.3 Análise Hierárquica de *Cluster*

A técnica de agrupamento hierárquico é utilizada para agrupar os dados com determinados níveis de similaridade. Esses métodos de agrupamento consideram inicialmente cada ponto de dados (ou objeto) como um grupo (De Almeida *et al.*, 2021). Inicialmente, cada ponto é considerado como um *cluster* e a soma dos quadrados dos desvios é zero. Então, objetos semelhantes começam a se unir para formar grupos.

À medida que os agrupamentos se cruzam, a soma dos quadrados dos desvios aumenta. Isso permite construir uma estrutura em árvore que interliga informações em grupos através de um método de ligação, denominado dendrograma. O dendrograma aglomerativo é comumente projetado com uma abordagem de baixo para cima, onde *clusters* menores são agrupados em *clusters* maiores. A Figura 2.1, de forma simplificada, esquematiza a separação dos seis pontos de dados em grupos e estrutura a representação minimalista do respectivo dendrograma. A distância entre os grupos que se formam é calculada pelo método de ligação, que neste trabalho foram considerados dois: Completo e *WARD*.

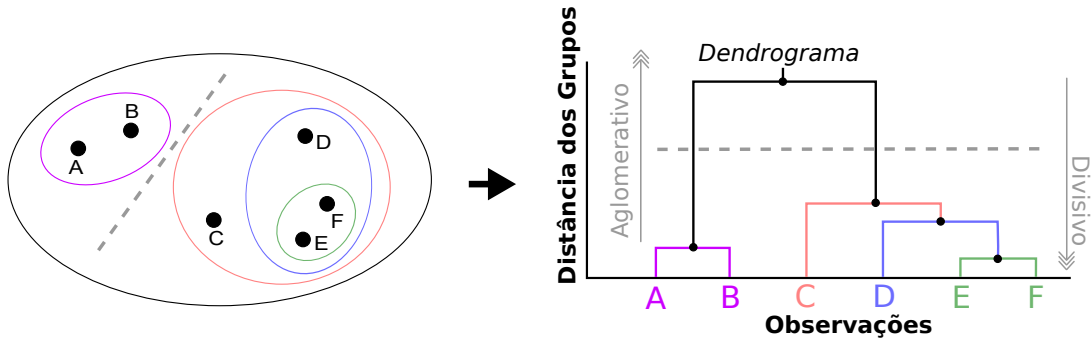


Figura 2.1 – Exemplo esquemático de agrupamento de seis pontos de dados e sua representação no dendrograma.

O método de ligação Completo, também conhecido como vizinho mais distante, calcula a distância máxima entre um objeto em um determinado *cluster* e outro objeto pertencente a outro *cluster*. Em geral, o diâmetro dos grupos formados tende a ter tamanhos semelhantes. O método Completo foi escolhido por apresentar bom desempenho em determinados casos (De Luca; ZUCCOLOTTO, 2021) e é representado pela Equação 2.1, onde “ $D(x, y)$ ” é a distância entre os *clusters* “ x ” e “ y ” e “ $x[i]$ ” simboliza um objeto “ i ” no *cluster* “ x ” (TOKUDA *et al.*, 2022).

$$D(x, y) = \max_{i,j} \{D(x[i], y[j])\} \tag{2.1}$$

O método de ligação de Ward procura minimizar internamente a soma dos desvios quadrados em cada *cluster* (LIMA *et al.*, 2016), e a distância entre esses *clusters* é calculada pela soma dos desvios quadrados dos pontos aos centróides (e, portanto, a variância entre os elementos). Neste caso, cada grupo tende a ter o mesmo número de objetos. A escolha do método de Ward para compor os experimentos deste trabalho se deu por demonstrar boa separabilidade entre grupos e consistência (De Almeida *et al.*, 2021). A Equação 2.2 calcula a distância do Ward, onde “ $|y|$ ” representa o número de objetos presentes no *cluster* “ y ”, e assim por diante.

$$D(x, y) = \sqrt{\frac{|y| + |z|}{|y| + |z| + |w|} D(y, z)^2 + \frac{|y| + |w|}{|y| + |z| + |w|} D(y, w)^2 - \frac{|y|}{|y| + |z| + |w|} D(z, w)^2} \tag{2.2}$$

2.4 Ensemble de Redes Neurais Artificiais

Redes neurais artificiais são algoritmos cujo princípio de funcionamento é inspirado no funcionamento do cérebro humano (SOBRI *et al.*, 2018; VISCONDI; ALVES-SOUZA,

2019; CERVONE *et al.*, 2017). Existem inúmeras situações em que o uso de redes neurais artificiais é satisfatório (Kara Mostefa Khelil *et al.*, 2021), como reconhecimento de padrões, classificação, detecção de falhas e previsão de geração fotovoltaica (DHIMISH *et al.*, 2018). Uma vez que o problema de previsão da geração fotovoltaica possui, em essência, características não lineares, modelos de aprendizado de máquina tentam capturar essas variações de forma eficiente e apresentá-las na saída (WANG *et al.*, 2022a), mas com a premissa de que não há modelo na literatura que tenha um bom desempenho em todos os casos (ABBASIMEHR *et al.*, 2020). As RNAs foram escolhidas neste trabalho devido ao seu desempenho superior em comparação com outros modelos de aprendizado de máquina (KIM *et al.*, 2022).

Basicamente, a estrutura é composta por três camadas em sua arquitetura mínima (LIN *et al.*, 2022b): camada de entrada, camada oculta (ou escondida) e camada de saída (MUBIRU; BANDA, 2008). Cada camada possui um número específico de nós (ou neurônios) que interconectam uma camada à outra. A primeira camada é conhecida como entrada de dados. Esta camada pode conter um ou mais neurônios. A segunda camada, conhecida como camada intermediária (ou oculta), pode não ser única e possui vários neurônios definidos pelo analisador, independente do número escolhido para a primeira camada. Por fim, existe a última camada, ou camada de saída, onde os resultados são obtidos após o processo de treinamento e teste. Com base em BALESTRASSI *et al.* (2009), o número de camadas ocultas escolhidas varia entre uma ou duas, sendo que zero ou três raramente são utilizadas.

A conexão entre os neurônios é definida por pesos, calculados iterativamente em um estágio de treinamento (CERVONE *et al.*, 2017). Neurônios estão presentes em todas as camadas e constituem a arquitetura da rede, podendo ser adicionados (ou removidos) de cada camada à medida que se ajusta bem (ou mal) ao problema em questão. A anatomia de um neurônio mostra que ele recebe uma entrada, calcula os pesos relativos a essa entrada e retorna o resultado por meio de uma função de ativação (YAHYA *et al.*, 2022).

Conforme discutido em XIAO *et al.* (2017), o número de neurônios nas camadas intermediárias desempenha um papel importante no desempenho da previsão. Alguns autores consideram definir o número de neurônios aleatoriamente (YEON *et al.*, 2019), enquanto outros o consideram de forma sistemática, como KURKOVA (1992).

As redes têm uma inicialização aleatória no processo de treinamento (LÓPEZ *et al.*, 2018) e o aprendizado é adquirido com base na otimização do ajuste de pesos, transferindo informações de uma camada para outra, até que um critério estabelecido seja alcançado (MONTEIRO *et al.*, 2017), ou seja, em um critério para mapear a não-linearidade entre as entradas e as saídas (VAGROPOULOS *et al.*, 2016).

Basicamente, cada época representa um processo de treinamento do modelo com

um conjunto de dados. O número de vezes que o algoritmo itera sobre os mesmos dados para aumentar a precisão da saída corresponde ao número de épocas. Evidentemente, aumentar esse número tende a aumentar a precisão da resposta e, conseqüentemente, tende a aumentar o custo computacional para o processamento do modelo.

Uma rede neural do tipo *MLP* interconecta totalmente as camadas e é regulada com base no aprendizado supervisionado (AKHTER *et al.*, 2019). A modelagem de uma rede neural com k entradas, apenas uma saída y e h neurônios ocultos é mapeada pela seguinte Equação 2.3 (BARBIERI *et al.*, 2017):

$$y = (y; w) = \sum_{j=0}^h \left[w_{jo} f \left(\sum_{i=0}^k w_{ij} \cdot x_i \right) \right] \quad (2.3)$$

Os pesos e o *bias* (usado para calibrar o grau de liberdade dos ajustes dos pesos) são representados por i e j , que interconectam as camadas, os parâmetros de rede são representados pelo vetor w e a função de transferência representada por f . Considerando que na maioria dos casos a rede neural tem uma inicialização aleatória de pesos e também possui diversos parâmetros com uma variedade de opções, podemos obter resultados diferentes para cada configuração mapeada. O desafio é determinar qual configuração (ou conjunto delas) promove bons ajustes de modelo para os dados.

A definição de parâmetros que otimizam o funcionamento da RNA não é imediata, e muitas vezes não há consenso em relação a determinadas escolhas, como o número de camadas e o número de neurônios em cada camada (KOTU; DESHPANDE, 2019). Alguns autores consideram a escolha dos parâmetros por tentativa e erro (MAHFOUD *et al.*, 2022) e não de forma sistemática. Os parâmetros da RNA considerados neste trabalho foram baseados nas pesquisas de ZEKIĆ-SUŠAC *et al.* (2021) e BALESTRASSI *et al.* (2009) e são abordados na próxima Seção (3), que apresenta detalhadamente a metodologia estimada nesta pesquisa, que usa *DOE* como uma ferramenta estatística para reduzir o espaço de busca paramétrico.

A taxa de aprendizagem pode ser considerada como o tamanho do passo para encontrar a solução do problema e deve ser escolhida com cuidado, pois valores altos podem levar a uma rápida convergência do algoritmo para pontos de solução subótimos, e valores baixos podem levar à estagnação do processo no espaço de busca.

Um *ensemble*, proposto inicialmente por HANSEN; SALAMON (1990), busca extrair características de um modelo que, quando combinado, descreve melhores resultados. Neste trabalho, a técnica de combinação é discutida nas Seções 2.3 e 2.7.

2.5 Planejamento de Experimentos Fatorial

Percebe-se na literatura um vasto registro do uso de *DOE*, como para calibração paramétrica de modelos de previsão (PONTES *et al.*, 2016), para escolha do conjunto de treinamento (CHU *et al.*, 2021) e também para otimização de parâmetros aplicada em simulações de fabricação (SBAYTI *et al.*, 2022). O DOE, por meio da composição de suas ferramentas que utilizam recursos matemáticos e estatísticos, permite ao analisador identificar sistematicamente a relação entre causa e efeito, o que pode levar a uma solução que otimize o processo. Em geral, há uma escolha de fatores e níveis, variáveis de resposta, a estrutura do desenho experimental e a própria execução (MONTGOMERY, 2017). A abordagem DOE permite que um pequeno número de experimentos seja realizado no processo de simulação; portanto, sua aplicação torna-se vantajosa (PONTES *et al.*, 2016). A lógica da escolha está intrinsecamente ligada ao tipo de estudo.

Desenhos fatoriais completos ou fracionados, geralmente com dois níveis, são bem aceitos pela indústria (ANTONY, 2014). Projetos fatoriais completos consideram todas as combinações possíveis, o que gera um espaço de busca com dimensão de 2^k , onde k é o número de fatores. Entende-se que, ao aumentar o número de fatores (até mesmo seus respectivos níveis), o planejamento fatorial completo leva a um número extremamente alto de execuções experimentais, o que pode gerar altos custos e alta demanda de tempo (JANKOVIC *et al.*, 2021). Assim, este estudo considera um planejamento fatorial fracionário de 2 níveis devido às limitações naturais de um experimento simulado, que são a escassez de recursos computacionais e tempo.

A Figura 2.2 mostra uma representação esquemática para esclarecer o potencial do DOE, que permite ao analisador restringir o espaço de busca paramétrica a fatores que potencialmente levam à solução do problema. A varredura de todo o espaço de busca implica um alto custo computacional e de tempo. Assim, com base em referências (da literatura, por exemplo), consegue reduzir esse espaço de busca a um conjunto específico de parâmetros, o que naturalmente não garante a solução ótima, mas permite ter uma ideia desse ajuste e como os fatores interagem um com o outro.

O número de execuções (ou corridas) experimentais aumenta exponencialmente à medida que o número de fatores aumenta. Isso pode ser um problema quando há falta de recursos ou tempo para examinar todas as combinações possíveis. Muitos dados, tanto da geração fotovoltaica quanto climáticos, bem como o número de parâmetros dos modelos de aprendizado de máquina que podem ser combinados, desafiam o poder de processamento dos computadores atuais, que é limitado (MASSAOUDI *et al.*, 2021). Quando se trata de pesquisas envolvendo simulação computacional, geralmente há muitos dados e/ou parâmetros envolvidos. A fim de mitigar o custo computacional dos experimentos deste trabalho, o DOE foi considerado para reduzir o espaço de busca paramétrica por ser uma ferramenta eficaz para este fim (KHAN *et al.*, 2020).

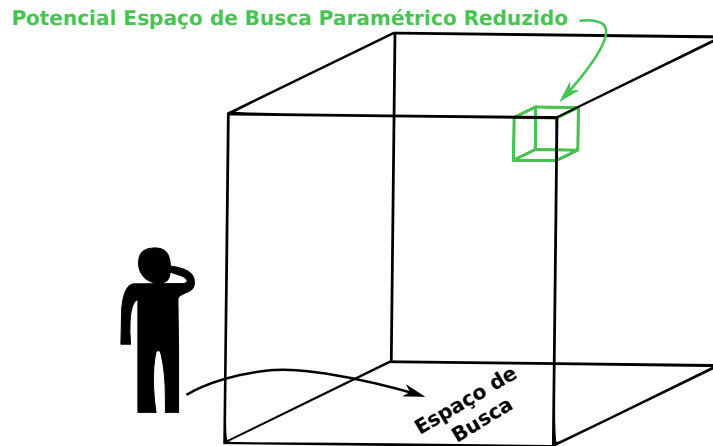


Figura 2.2 – Analogia da sala: O analisador sabe que há uma restrição de recursos e tempo para varrer toda a sala para a possível correspondência paramétrica ideal. Mas, com base em experiências anteriores (ou pesquisas anteriores), ele sabe que existe uma região de busca reduzida que pode levar a uma boa solução para o problema (não necessariamente a ótima).

A matriz experimental deve ser definida com base nos fatores e seus respectivos níveis. A qualidade de reduzir (ou aumentar) a profundidade dessa pesquisa usando DOE é medida em termos de confundimentos e é resumida na resolução do experimento. Quando há escassez de recursos para realizar os experimentos, além de escolher os níveis dos fatores, o DOE permite a redução de execuções experimentais, mantendo a confiabilidade estatística (JANKOVIC *et al.*, 2021) dessas execuções.

Assim, a resolução pode ser entendida como a quantidade de esforço a ser empregada no experimento e é definida por um projeto predeterminado. A redução de esforço refere-se a uma fração do planejamento fatorial completo, nos casos em que não é possível executar todas as combinações. Em geral, pode-se fazer inferências com um alto nível de confiança estatística ao realizar experimentos fracionários. Na Figura 2.3, as resoluções em vermelho (identificadas por III) devem ser cuidadosamente consideradas, pois o número de execuções pode ser significativamente muito baixo. Conforme mostrado na Figura 2.3, este trabalho considerou a resolução de nível IV, com redução de esforço 2^{k-p} , de modo que $k = 11$ e $p = 6$, pois neste nível os principais efeitos são considerados sem confusão com as interações de dois fatores.

Neste processo simulado, tem-se 11 fatores com dois níveis cada. Assim, gastaria-se 2^k tentativas para testar todas as combinações, onde k corresponde ao número de fatores, atingindo um total de 2048 execuções experimentais para esta análise. Esse número é considerável para análises adicionais e, principalmente, se a parametrização (e a execução) for implementada manualmente para cada teste.

Para reduzir o número de execuções (ou corridas experimentais), o *DOE* permite projetar o experimento com redução de esforço sem comprometer a análise de inferência

Corridas Experimentais	Fatores														
	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
4	Full	III													
8		Full	IV	III	III	III									
16			Full	V	IV	IV	IV	III	III	III	III	III	III	III	
32				Full	VI	IV	IV	IV	IV	IV	IV	IV	IV	IV	
64					Full	VII	V	IV	IV	IV	IV	IV	IV	IV	
128						Full	VIII	VI	V	V	IV	IV	IV	IV	

Figura 2.3 – Desenhos fatoriais e suas respectivas resoluções. Baseado no *software Mini-tab*.

do pesquisador, através da fórmula $2^{(k-p)}$, em que p representa a redução de esforço.

2.6 Análise de Componentes Principais - PCA

A Análise de Componentes Principais (*Principal Components Analysis - PCA*) é uma ferramenta multivariada amplamente utilizada na literatura (XIA *et al.*, 2022). Basicamente, o processo reduz a dimensionalidade do conjunto de dados, a um conjunto não correlacionado, conhecido como componentes principais (que pode explicar a variância do conjunto original). Ele pode separar informações redundantes e aleatórias. A representação da variância dos dados tende a estar nos primeiros componentes (onde o primeiro componente tem a explicação máxima em relação aos outros componentes (AHMADI *et al.*, 2020) e assim por diante). O ruído tende a estar nos últimos componentes, ou seja, os componentes principais são combinações lineares não correlacionadas (JOHNSON; WICHERN, 2007) das variáveis originais ponderadas pelos autovalores.

Segundo (RENCHEER, 2003), pode ser descrita brevemente considerando n vetores de observação y_1, y_2, \dots, y_n e o respectivo vetor médio \bar{y} (onde será a origem do eixo elipsóide). A mudança para a origem \bar{y} é descrita $y_i - \bar{y}$. Girar o eixo centrado na média resulta em componentes principais, que não são correlacionados. O movimento de rotação multiplica cada y_i por uma matriz ortogonal A , conforme a Equação 2.4:

$$z_i = Ay_i \tag{2.4}$$

Se A é ortogonal, então $A'A = I$, e a distância até a origem permanece a mesma, conforme observado na Equação 2.5:

$$z'_i z_i = (Ay_i)'(Ay_i) = y'_i A' A y_i = y'_i y_i \tag{2.5}$$

A rotação transforma y_i em um ponto z_i , mantendo a mesma distância da origem. O cálculo da matriz A permite descobrir os eixos do elipsóide, tornando $z = Ay$ não

correlacionado. Desta forma, deseja-se que a matriz de covariância amostral de z , $S_z = ASA'$ seja diagonal, como na Equação 2.6:

$$S_z = ASA' = \begin{pmatrix} S_{z_1}^2 & 0 & \dots & 0 \\ 0 & S_{z_2}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & S_{z_p}^2 \end{pmatrix} \quad (2.6)$$

onde S é a matriz de covariância de y_1, y_2, \dots, y_n . Como λ_i 's são os autovalores de S e C uma matriz ortogonal na qual as colunas são os autovetores normalizados de S , $diag(\lambda_1, \lambda_2, \dots, \lambda_p) = C'SC = D$. A transposta da matriz C é a matriz ortogonal A que diagonaliza S , conforme mostrado na Equação 2.7:

$$A = C' = \begin{pmatrix} a'_1 \\ a'_2 \\ \vdots \\ a'_p \end{pmatrix} \quad (2.7)$$

de modo que a_i é o autovetor i -ésimo normalizado de S . Os componentes principais são representados pelas variáveis $z_1 = a'_1 y$, $z_2 = a'_2 y, \dots, z_p = a'_p y$ em $z = Ay$. Os elementos diagonais de ASA' são autovalores de S . Isso torna os autovalores $\lambda_1, \lambda_2, \dots, \lambda_p$ de S as variâncias dos componentes principais $z_i = a'_i y$, conforme descrito na Equação 2.8:

$$S_{z_i}^2 = \lambda_i \quad (2.8)$$

Como os autovalores são as variâncias dos componentes principais, a expressão de porcentagem de explicação da variância (EV) pelos primeiros k componentes é usada, conforme definido na Equação 2.9:

$$EV = \frac{\lambda_1 + \lambda_2 + \dots + \lambda_k}{\lambda_1 + \lambda_2 + \dots + \lambda_p} \quad (2.9)$$

Reduzir a dimensionalidade dos dados meteorológicos, para treinamento de modelos de aprendizado de máquina, evita *overfitting* e permite que os dados originais sejam substituídos por esse novo conjunto de dados, reduzido, mas retém a maior parte das informações originais (GE *et al.*, 2020).

2.7 Mixture DOE para a composição dos pesos do ensemble

Combinar previsões é tentar obter melhor desempenho em relação aos previsores quando considerados individualmente (KOURENTZES *et al.*, 2019). A literatura relata

um benefício empírico dessa combinação na melhoria dos resultados de previsão (QIAN *et al.*, 2022), de maneira que a previsão combinada tende a produzir resultados mais precisos. Nesse sentido, esta proposta busca alcançar essa combinação por meio de um experimento de mistura.

Assim, este trabalho utiliza *Mixture* DOE para combinar os resultados de previsão. Uma análise de mistura considera os fatores como “ingredientes”, e denota que as proporções devem ser iguais a 1 (um) (MONTGOMERY, 2017). Especificamente, um experimento de mistura considera encontrar as proporções ótimas para cada ingrediente, ou seja, no problema de previsão, essa proporção é identificada pelos pesos w_i e, neste estudo, os fatores representam os ingredientes dessa analogia. O objetivo da mistura é encontrar os pesos que proporcionem um erro de previsão menor que o erro obtido pelo melhor componente de previsão, considerado individualmente.

Por exemplo, w_1, w_2, \dots, w_N , são os pesos que compõem o *ensemble*, onde N é a quantidade de componentes a serem combinados (podendo ser previamente selecionados por meio da análise de *cluster*).

Aqui, o valor combinado (que é tomado como resposta) depende apenas dos pesos (proporção dos ingredientes) e não apenas dos próprios fatores. Segundo (CORNELL, 2011), os pesos w_i são não negativos, expressos como frações das misturas, cuja soma de todos os N fatores (ingredientes) deve ser unitária. Assim, $w_1 + w_2 + \dots + w_N = 1$.

A Figura 2.4 (adaptado de MONTGOMERY (2017)) mostra dois projetos centróides *simplex*. O primeiro (a) considera um exemplo com três fatores, ou ingredientes, tendo uma representação gráfica deste arranjo como um triângulo e analisa a combinação usando três componentes: $w_1 + w_2 + w_3 = 1$. O segundo (b) considera quatro componentes, e $w_1 + w_2 + w_3 + w_4 = 1$. À medida que o número de fatores aumenta, a representação geométrica também muda. Por exemplo, ao considerar quatro fatores, a representação é dada por um tetraedro. Vários fatores maiores ou iguais a cinco são viáveis, mas não há mais possibilidade de representação visual.

Cada vértice do triângulo ou tetraedro é considerado uma "mistura pura", ou seja, a proporção dos outros componentes para esse vértice é nula. Por exemplo (considerando o triângulo), quando $w_2 = 1, w_1$ e w_3 são automaticamente zero. O número de pontos está relacionado ao número de componentes considerados no *design simplex* do centróide e é generalizado pela fórmula $(2^c - 1)$, em que c é o número total de componentes.

O *ensemble* proposto utiliza pesos definidos pela análise de mistura, aplicados aos resultados selecionados pela análise de *cluster*. Assim, a seguinte Equação 2.10 modela matematicamente esta combinação, na qual w representa os pesos e y os valores previstos, escolhidos pela análise de *cluster*:

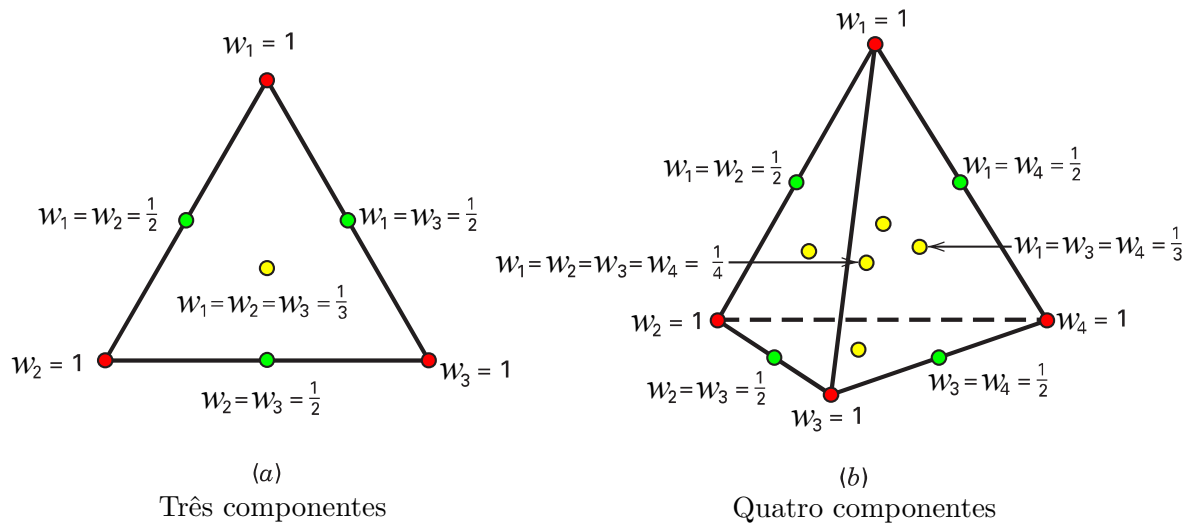


Figura 2.4 – *Design simplex* com três componentes(a) e com quatro componentes(b). Adaptado de (MONTGOMERY, 2017).

$$\hat{y}_i = \sum_{i=0}^n w_i \cdot y_i \tag{2.10}$$

A métrica para avaliação e definição dos pesos é baseada no Erro Percentual Absoluto Médio (*MAPE*), que já vem sendo utilizado em trabalhos de previsão recentes (NIU *et al.*, 2022). O cálculo do erro é obtido conforme mostrado na Equação 2.11:

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \tag{2.11}$$

de modo que y_i é o valor real de medição da geração fotovoltaica, \hat{y}_i é o valor previsto e N corresponde ao número de pontos previstos.

3 METODOLOGIA

Tendo em vista a versatilidade da metodologia proposta, será apresentada nesta Seção levando em conta dois aspectos que foram observados nesse processo:

- (a) considerando o uso da variável climática através da análise de correlação de *Pearson*;
- (b) considerando dados meteorológicos por meio da redução de dimensionalidade, usando análise de componentes principais (*PCA*);

Em ambos os casos, o processo metodológico de previsão usa, essencialmente, o planejamento de experimentos (*DOE*), que permite ao analisador identificar informações que são relevantes a esse processo (outroa definidas empiricamente ou aleatoriamente) bem como mapear a interação entre os fatores e como a relação entre as variáveis afeta a precisão do resultado (LIRA *et al.*, 2020).

Ainda assim, o método proposto é flexível ao ponto de permitir que vários cenários sejam considerados, a depender das características de cada planta (ou conjunto de usinas) de geração fotovoltaica. Fatores relacionados à série temporal também podem ser mapeados a depender das características e disponibilidade do conjunto de dados. Neste sentido, há flexibilidade na escolha de parâmetros.

3.1 Variáveis Climáticas Escolhidas por Correlação

A Figura 3.1 resume os passos para a previsão de geração fotovoltaica para o horizonte de uma semana à frente. Aqui, os fatores do *DOE* são divididos em dois grupos: fatores da RNA e fatores da série temporal. A análise de *cluster* seleciona hierarquicamente grupos de previsão com potencial para compor uma previsão combinada. A análise de mistura aplicada ao grupo de *cluster* tende a melhorar os resultados e, conseqüentemente, a diminuição do erro de previsão.

Sendo assim, as próximas Seções descrevem cada etapa da Figura 3.1, iniciando com os fatores da série histórica de geração até a etapa de combinar e validar o resultado de previsão.

3.1.1 Fatores da Série Temporal

Inicialmente, serão apresentados os métodos de normalização de dados utilizados nesta etapa. A importância dos métodos de normalização é discutida na Seção 2.2. O primeiro método, geralmente chamado de padronização (ou *standardization*), foi aplicado

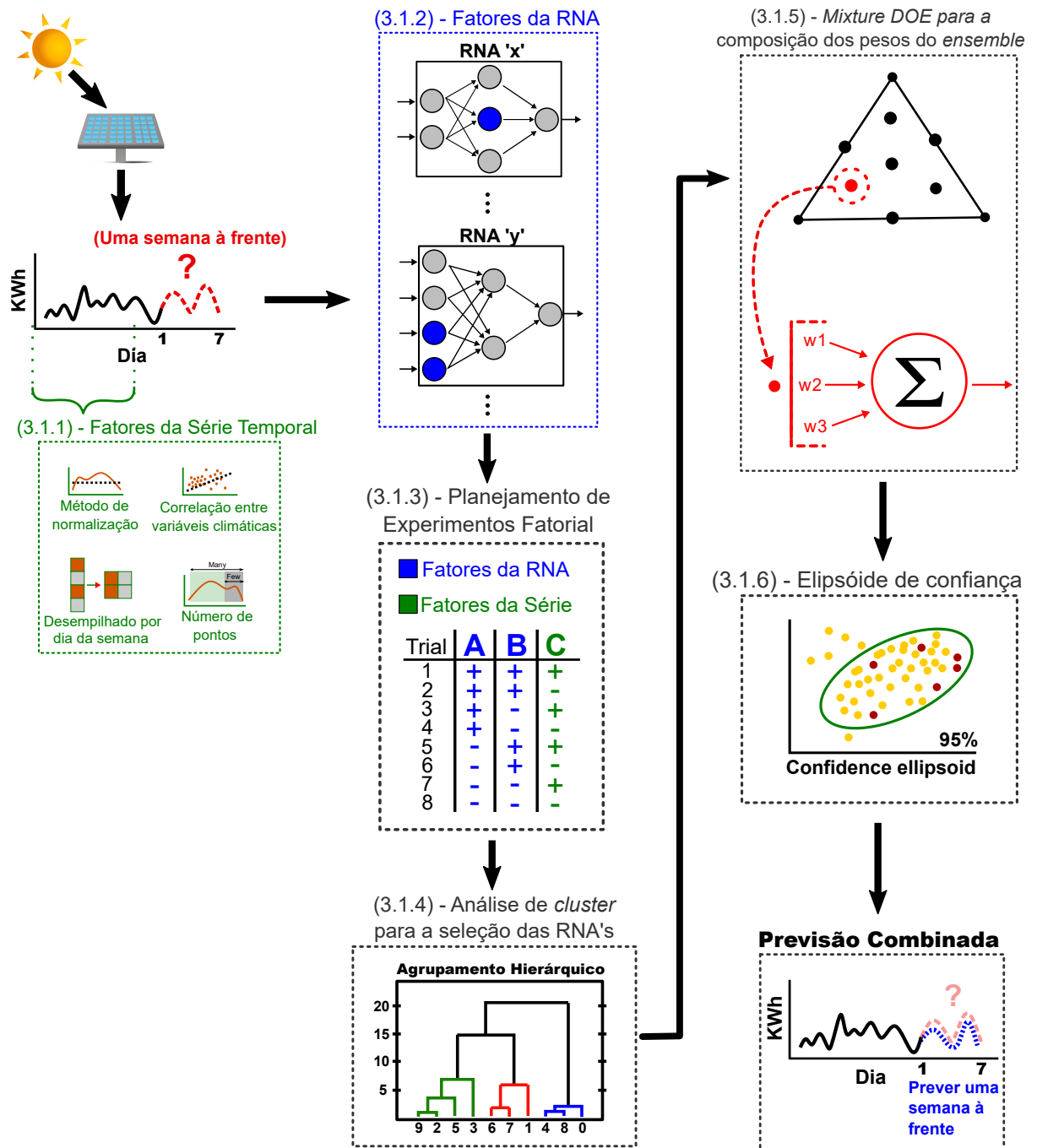


Figura 3.1 – Metodologia proposta para previsão de geração fotovoltaica (PV) usando rede neural artificial (RNA), planejamento de experimentos (DOE), análise de cluster e mixture DOE (MDE).

em alguns estudos com foco na previsão de geração de energia fotovoltaica (AMROUCHE; PIVERT, 2014; APPIAH *et al.*, 2019; MALVONI *et al.*, 2016) e dimensiona os dados em um intervalo de 0-1, dividindo cada observação y_i pelo valor máximo da série temporal $max(Y)$. Para cada valor de geração i , o correspondente normalizado \hat{y}_i , é calculado usando a Equação 3.1:

$$\hat{y}_i = \frac{y_i}{\max(Y)} \quad (3.1)$$

O segundo método, geralmente chamado de *Feature Scaling*, é frequentemente aplicado para pré-processamento de dados de entrada (DAS *et al.*, 2018; LIU *et al.*, 2015; DU *et al.*, 2018). Ele consiste, assim como no método anterior, manter os dados entre 0 e 1 através da divisão do valor observado y_i menos o valor mínimo da série $\min(Y)$, dividido pela subtração entre o valor máximo $\max(Y)$ e o valor mínimo. A fórmula é observada na Equação 3.2:

$$\hat{y}_i = \frac{y_i - \min(Y)}{\max(Y) - \min(Y)} \quad (3.2)$$

O uso de variáveis externas é uma abordagem comum para melhorar o processo de previsão e deve ser considerada quando houver uma forte correlação (DAS *et al.*, 2018). Dados meteorológicos, como temperatura, irradiância, umidade, velocidade do vento, horas de sol, cobertura de nuvens e precipitação, são exemplos desses tipos de dados. Esta medida pode ser positiva ou negativamente correlacionada. O primeiro indica que, a série pode aumentar ou diminuir proporcionalmente e, o segundo caso indica que uma série pode aumentar enquanto outra diminui (proporcionalmente). Esse comportamento pode ser calculado usando a fórmula de correlação de *Pearson* (Equação 3.3), sendo que muitos estudos (YANG; HUANG, 2018; SOBRI *et al.*, 2018; LIMA *et al.*, 2016; BUGALA *et al.*, 2018; LIU *et al.*, 2015; DU *et al.*, 2018) adotaram esta mesma abordagem:

$$\rho_{x,y} = \frac{\text{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (3.3)$$

Originalmente, as séries de dados são obtidas com as observações ordenadas no tempo. Para alguns autores (YANG; HUANG, 2018; MELLIT; PAVAN, 2010), o uso de dias semelhantes pode contribuir para melhorias no resultado da previsão. Para investigar se há alguma relação entre a geração de energia fotovoltaica quando se considera os mesmos dias da semana, propusemos desempilhar os dados considerando cada dia da semana, resultando em sete séries de dados (separadas). Quando a previsão é realizada, se os dados não forem empilhados, cada série temporal é usada como entrada para o modelo, considerando o dia específico da semana. O processo usado para desempilhar os dados originais (sequenciais) está resumido na Figura 3.2:

O número de observações nas séries temporais pode variar dependendo do local de medição, das condições do equipamento usado, dos dispositivos de armazenamento e da periodicidade das informações coletadas. Considerando essas informações, a metodologia deste trabalho propõe (inicialmente) o uso de todos os dados disponíveis para o treinamento do modelo e, em um segundo momento, o uso de uma parte reduzida desse

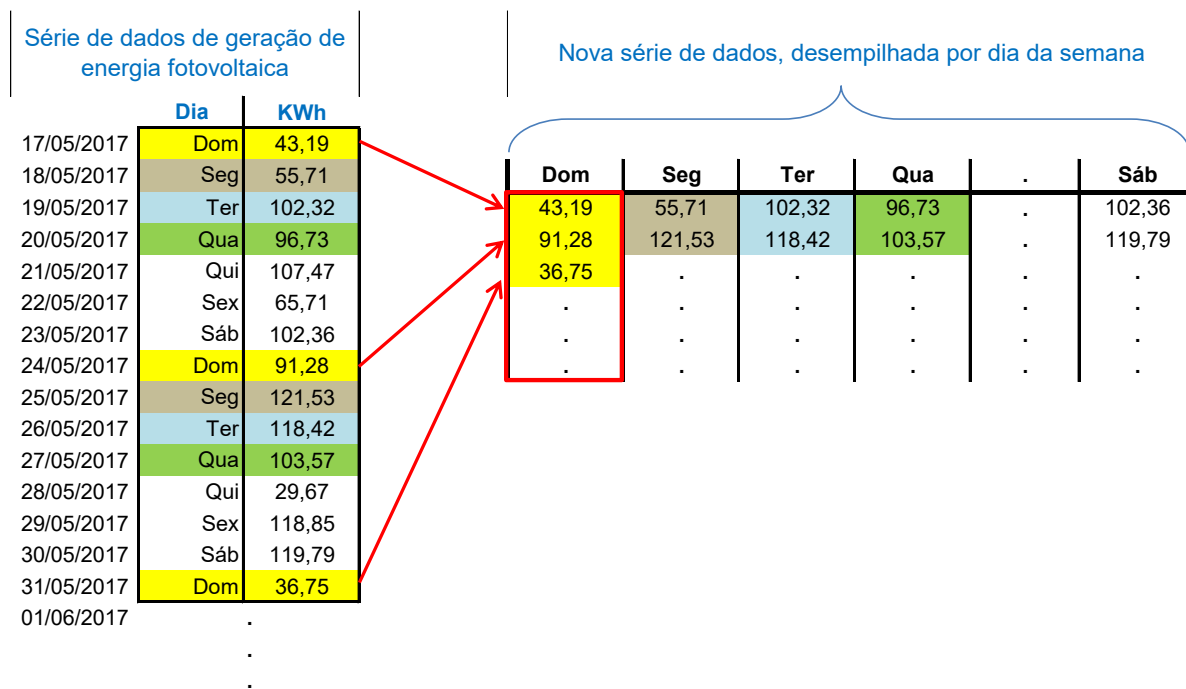


Figura 3.2 – Estrutura do conjunto de dados: sequencial ou desempilhado por dia da semana.

histórico de dados para avaliar os impactos nos resultados previstos. Em outras palavras, a quantidade de dados a ser usada no processo de previsão é um fator do *DOE*.

3.1.2 Fatores da Rede Neural Artificial (RNA)

Aqui são elencados os fatores e níveis relacionados ao modelo de previsão escolhido: Redes Neurais Artificiais. Com base no trabalho de [BALESTRASSI et al. \(2009\)](#), o número de camadas ocultas escolhidas para compor um dos fatores do *DOE* foi uma e duas.

A depender da estrutura de dados (sequencial ou desempilhado por um determinado dia da semana), entradas e saídas de treinamento podem ser organizadas sob duas perspectivas diferentes: a primeira considera os dados desempilhados por dia da semana, enquanto a outra utiliza as séries de dados em seu formato original. Esse fator está relacionado à quantidade de neurônios na camada de entrada do modelo.

Por exemplo, se considerarmos um número de entradas igual a três, tem-se uma configuração de treinamento para o modelo, conforme resumido na Figura 3.3. Em todos os casos, tem-se apenas uma saída como o valor desejado.

Mesmo sabendo que não há consenso na literatura para a definição da arquitetura das redes neurais artificiais e que depende da natureza do problema que está sendo considerado ([MITTAL et al., 2018](#)), este trabalho considerou o estudo das séries temporais não-lineares apresentadas por [BALESTRASSI et al. \(2009\)](#), o qual utilizou a seguinte for-

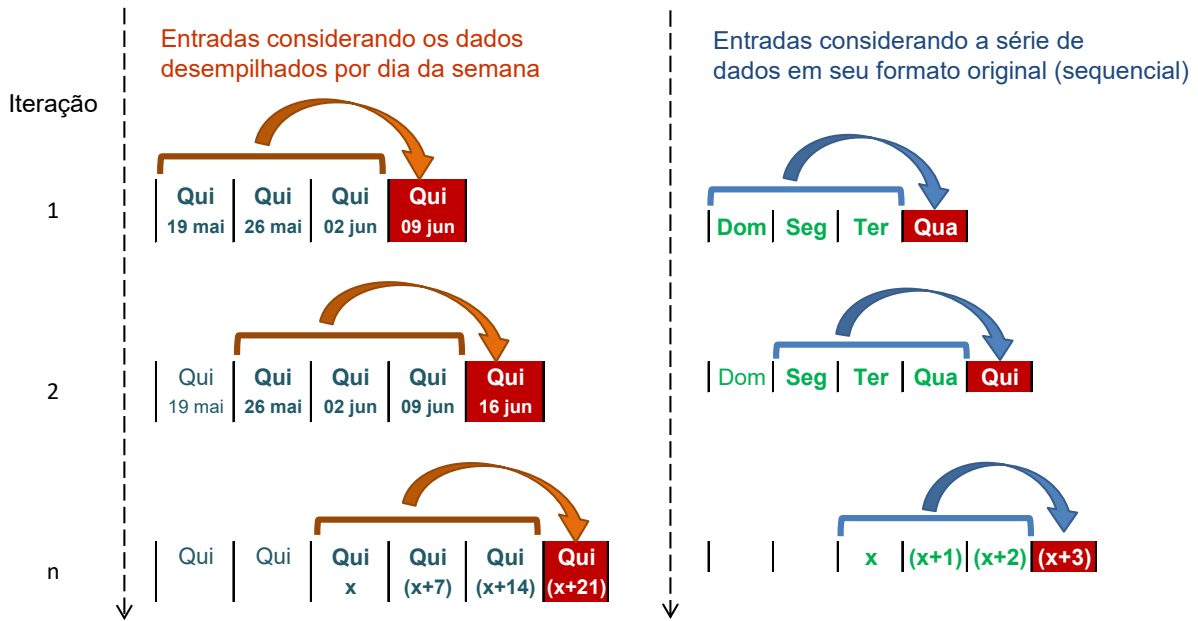


Figura 3.3 – Iterações do algoritmo de treinamento da RNA usando estrutura de dados sequencial ou desempilhada. Três entradas para uma saída.

mulação para o cálculo do número de neurônios nas camadas intermediárias: $K \times (N + 1)$, em que N é o número de entradas e $K = (1, 5; 2)$.

Conforme discutido em [BALESTRASSI et al. \(2009\)](#), a taxa de aprendizagem considerada neste trabalho foi 0,1 ou 0,9.

Dois algoritmos foram escolhidos para formar a base do treinamento da rede neural nos experimentos realizados. Os algoritmos de treinamento são responsáveis pela atualização dos pesos e valores de *bias* da rede ([KHEIRKHAH et al., 2013](#)). O primeiro algoritmo, o gradiente conjugado escalonado (*SCG*) ([MOLLER, 1993](#)), é frequentemente aplicado ao processo de treinamento ([AMARAL et al., 2019](#)). O segundo algoritmo, *Levenberg-Marquardt* ([MARQUARDT, 1963](#)), foi selecionado porque foi considerado na literatura como tendo desempenho superior no treinamento de redes neurais artificiais do tipo *feedforward* ([HAGAN; MENHAJ, 1994](#)).

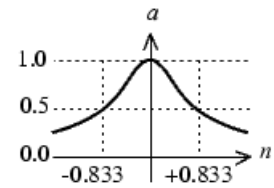
As funções de transferência (ou ativação) visam calcular a saída de uma camada a partir dos dados que chegam da camada imediatamente anterior. Nesta etapa metodológica, foram consideradas duas funções: base radial ([MITTAL et al., 2018](#)) e sigmóide tangencial hiperbólica ([XIAO et al., 2017](#)). A Tabela 3.1 permite visualizar as representações matemáticas e gráficas para cada função.

Tabela 3.1 – Representação matemática e gráfica das funções de transferência, de acordo com o catálogo do *Matlab*

Tipo da Função	Representação Matemática	Representação Gráfica
----------------	--------------------------	-----------------------

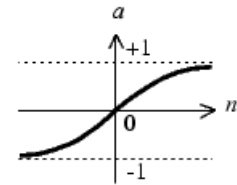
Base Radial

$$f(x) = e^{-(\delta x)^2}$$



Sigmóide Tangente Hiperbólica

$$f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$



3.1.3 Planejamento de Experimentos Fatorial

Como existem 11 fatores, divididos em 7 fatores da rede neural artificial e 4 fatores da série temporal, o experimento foi preparado com uma redução de esforço de 2^{11-6} . Isso significa que apenas 32 execuções experimentais são necessárias para resolver esse *DOE* fatorial fracionado, cuja resolução é *IV*.

A RNA foi parametrizada e implementada no *software Matlab*. Os fatores da RNA estão listados na Tabela 3.2, juntamente com seus respectivos níveis.

Tabela 3.2 – Fatores do *DOE* para a RNA com dois níveis cada.

FATORES	Nível 1	Nível 2
(A) Camadas Escondidas	1	2
(B) Épocas	100	400
(C) Número de Entradas	3	5
(D) Neurônios por Camada	1,5	2
(E) Taxa de Aprendizagem	0,1	0,9
(F) Algoritmo de Treinamento	(scg) <i>Gradiente Conjugado Escalonado</i>	(lm) Levenberg-Marquadt
(G) Função de Transferência	(htg) <i>Sigmóide Tangente Hiperbólica</i>	(rbf) Base Radial

Os fatores da série temporal foram considerados em função da estrutura dos dados (desempilhados ou sequenciais), tipo de normalização, uso de variáveis correlacionadas (horas de insolação) e número de pontos da série (poucos ou muitos). Esses níveis estão resumidos na Tabela 3.3.

Tabela 3.3 – Fatores do *DOE* para a série temporal com dois níveis cada.

Fatores	Nível 1	Nível 2
(H) Desempilhado por Dia da Semana	(v) <i>Verdadeiro</i>	(f) <i>Falso</i>
(I) Método de Normalização	(fs) <i>Feature Scaling</i>	(mv) <i>Max Value</i>
(J) Variável Correlacionada	(n) <i>Não</i>	(s) <i>Sim</i>
(K) Número de Pontos	98	679

3.1.4 Análise de *cluster* para a seleção das RNA's

A análise de *cluster* foi aplicada a fim de separar os resultados de previsão (obtidos pelas RNA's nas corridas experimentais do *DOE*) com características semelhantes, ou seja, separando os resultados em grupos equivalentes para posterior combinação. Esse procedimento possibilita a construção de uma estrutura em árvore que interconecta informações em grupos por meio de um método de ligação, denominado “dendograma”. O dendograma é aglomerativo e geralmente é projetado com uma abordagem de baixo para cima, onde agrupamentos menores são agrupados em agrupamentos maiores.

O agrupamento é baseado no método de ligação de *Ward* e em uma medida de distância euclidiana. O objeto de interesse é o grupo com os menores valores de erro médio percentual absoluto (*MAPE*) que possuam semelhança. A significância das diferenças entre pares de grupos é estatisticamente identificada por uma análise de variância unidirecional, usando a comparação de *Tukey* como procedimento primário.

3.1.5 *Mixture DOE* para a composição dos pesos do *ensemble*

A quantidade de fatores do arranjo de misturas depende, exclusivamente, dos agrupamentos formados na etapa da análise de *cluster*. Sendo assim, pode haver diferenças na quantidade de fatores a depender do tipo de dados e da performance do modelo de previsão (que pode levar à formação de poucos ou muitos grupos).

3.1.6 Elipsóide de confiança

Quando aplicável, o resultado combinado obtido foi analisado a fim de descobrir se possui equivalência com o conjunto de dados original. Esta análise foi realizada através da elipse de confiança para os dados a um nível de 95% de confiança.

Uma região de confiança permite avaliar se os pontos na elipse seguem as mesmas características da maioria dos dados no conjunto original. Nesse caso, uma região de confiança é construída com base em dados de insolação e dados de geração de energia

fotovoltaica. Suponha que se tenha uma matriz de variância-covariância *sigma* (S). A equação a seguir define uma elipse (quando o número de variáveis $p = 2$) centrada na média (\bar{x}) com uma distância constante c :

$$(x - \bar{x})' S^{-1} (x - \bar{x}) = c^2 \quad (3.4)$$

Uma decomposição espectral da elipse rotacionada pode ser escrita como:

$$(x - \bar{x})' P \Lambda^{-1} P' (x - \bar{x}) = c^2 \quad (3.5)$$

Em 3.5, a matriz Λ é representada pelos valores próprios de S , denotados por:

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (3.6)$$

A matriz P é definida pelos autovetores de S :

$$P = \begin{bmatrix} \cos \theta & -\text{sen} \theta \\ \text{sen} \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \quad (3.7)$$

Assim, tem-se:

$$[P' (x - \bar{x})]' \Lambda^{-1/2} \Lambda^{-1/2} [P' (x - \bar{x})] = c^2 \quad (3.8)$$

$$\sqrt{[P' (x - \bar{x})]' \Lambda^{-1/2} \Lambda^{-1/2} [P' (x - \bar{x})]} = \sqrt{c^2} \quad (3.9)$$

$$\Lambda^{-1/2} [P' (x - \bar{x})] = \sqrt{\chi^2} \quad (3.10)$$

$$[P' (x - \bar{x})] = \sqrt{\chi^2} \Lambda^{1/2} \quad (3.11)$$

Assumindo que P é ortonormal, $P^{-1} = P'$:

$$[P^{-1} (x - \bar{x})] = \sqrt{\chi^2} \Lambda^{1/2} \quad (3.12)$$

$$(x - \bar{x}) = P [\sqrt{\chi^2} \Lambda^{1/2}] \quad (3.13)$$

$$x = \bar{x} + P [\sqrt{\chi^2} \Lambda^{1/2}] \quad (3.14)$$

A equação geral pode ser escrita como:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} + \begin{bmatrix} c\sqrt{\lambda_1}h_{11} \cos \alpha - c\sqrt{\lambda_2}h_{12}sen\alpha \\ c\sqrt{\lambda_1}h_{21} \cos \alpha + c\sqrt{\lambda_2}h_{22}sen\alpha \end{bmatrix} \quad c = \sqrt{\chi^2_{(p,\alpha/2)}}; \quad 0 < \alpha < 2\pi \quad (3.15)$$

Por fim, o pseudocódigo que resume esta etapa metodológica pode ser visto na Figura 3.4. As previsões usando *DOE* foram implementadas e automatizadas usando a linguagem de programação do *software Matlab* e podem ser encontradas no Apêndice A deste documento. Neste caso, a matriz *DOE* é lida a partir de um arquivo de texto e seus valores funcionam como índices vetoriais que representam os fatores do experimento. Cada linha da matriz *DOE* representa uma execução (ou corrida) experimental.

```

1. BEGIN
2.   doeMatrix ← carregar a matriz do DOE Fatorial Fracionado;
3.   Criar um vetor para cada fator do DOE;
4.   Preencher cada vetor com seus respectivos valores de 2 níveis;
5.   kwh ← carregar dados de geração fotovoltaica;
6.   insolation ← carregar dados com as horas de insolação;
7.   FOR run ← 1 TO numeroDeLinhas( doeMatrix )
8.     Baseado na matriz DOE, verificar se a série é sequencial ou desempilhada
9.     Normalizar o dado com base em doeMatrix;
10.    IF a variável externa correlacionada deve ser aplicada DO
11.      | trainingData ← insolation;
12.    ELSE
13.      | trainingData ← kwh;
14.    END IF
15.    FOR day ← 1 TO 7
16.      prepararNeuroniosDaRede( hiddenLayers, inputsNumber, unitsPerLayer );
17.      Criar a Rede Neural;
18.      Atribuir taxa de aprendizagem, número de épocas e função de transferência;
19.      treinarRNA( trainingData );
20.      Adicionar o valor previsto ao trainingData;
21.      Desnormalizar o valor previsto com base no método de normalização;
22.      Calcular MAPE;
23.    END FOR
24.  END FOR
25.  Agrupar os resultados baseando-se no método de ligação de Ward e distância Euclideana;
26.  Criar dendograma;
27.  Separar o grupo com os menores MAPEs e que possuem similaridade;
28.  Criar o DOE de misturas usando o arranjo Simplex Centroid;
29.  Selecionar a melhor combinação de pesos que reduz o MAPE total;
30.  Criar a elipse de confiança;
31.  Verificar se os resultados combinados pertencem ao elipsóide de confiança;
32. END

```

Figura 3.4 – Pseudocódigo da metodologia proposta, exibindo cada instância anteriormente apresentada. A implementação original pode ser observada no Apêndice A deste documento.

Os fatores mencionados anteriormente são processados para prever uma semana à frente e o *MAPE* é calculado. Aqui usa-se previsão sobre previsão, ou seja, a previsão dos dias subsequentes ao primeiro dia são calculadas com base na previsão do dia anterior: o valor previsto é usado como entrada do modelo para o próximo passo. Em seguida, usando o *software Minitab*, são realizadas a análise de *cluster* e a análise de mistura

(*Mixture DOE*). O elipsóide de confiança é criado usando uma formulação estruturada no *software Excel*.

3.2 Variáveis Climáticas Considerando Redução de Dimensionalidade

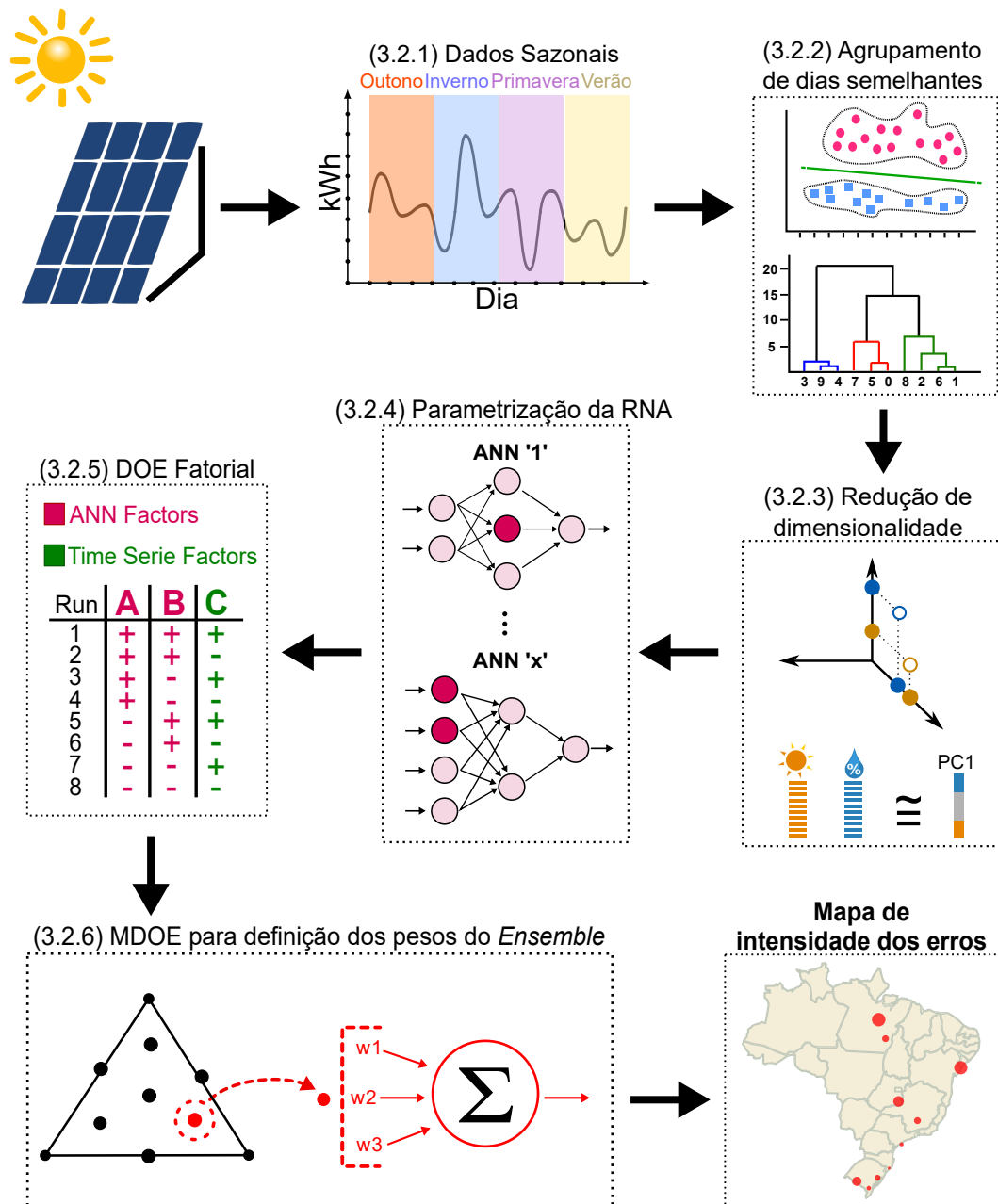


Figura 3.5 – Metodologia proposta para previsão de geração fotovoltaica (*PV*) usando análise de cluster, análise de componentes principais (*PCA*), rede neural artificial (*RNA*), planejamento de experimentos (*DOE*) e *mixture DOE* (*MDE*).

Muitas vezes, a escolha de uma variável climática usando o critério de correlação

pode descartar outras variáveis que poderiam contribuir para o treinamento do modelo de previsão. A fim de resolver esse impasse, é proposta esta segunda etapa metodológica, que usa análise de componentes principais para extrair as principais características de todo o conjunto de dados climáticos, sem descartar qualquer variável.

A redução de dimensionalidade permite explicar a variância do conjunto total, com pouca perda de informação. A vantagem dessa estratégia é utilizar o máximo de informação das variáveis exógenas para contribuir com o treinamento do modelo de previsão.

A Figura 3.5 resume os passos para a previsão de geração fotovoltaica para o horizonte de um dia à frente, considerando várias plantas de geração localizadas em diferentes regiões do território brasileiro. Em um primeiro momento os dados são separados por estação do ano. Aqui, os fatores do *DOE* são divididos em dois grupos: fatores da RNA e fatores da série temporal. A análise de *cluster* seleciona hierarquicamente grupos de previsão com potencial para compor uma previsão combinada. A análise de mistura aplicada ao grupo de *cluster* tende a melhorar os resultados e, conseqüentemente, a diminuição do erro de previsão.

A aplicação desta metodologia auxilia na operação de redes de distribuição ativas e sistemas de transmissão emergentes, uma vez que o operador é informado sobre a disponibilidade real de geração na próxima janela de tempo. Assim, são possíveis ajustes de geração e configuração do sistema, permitindo que as concessionárias prestem um serviço confiável.

Sendo assim, as próximas Seções descrevem cada etapa da Figura 3.5, iniciando com a preparação dos dados sazonais.

3.2.1 Preparação dos Dados Sazonais

A série de dados foi dividida em estações do ano. Uma vez que cada estação apresenta características diferentes, essa separação pode ser questão essencial para o sucesso de uma previsão mais precisa. Sabe-se que o aumento ou diminuição da eficiência dos painéis pode ser influenciado por fatores ambientais da região, como velocidade do vento, umidade, poeira, temperatura, entre outros (DONALDSON *et al.*, 2021), justificando a segmentação por estação. No Brasil, os verões são quentes e úmidos, com predomínio de chuvas em várias regiões, enquanto o inverno provoca seca e frio. A partir da Figura 3.6, é possível identificar esses períodos ao longo dos meses do ano.

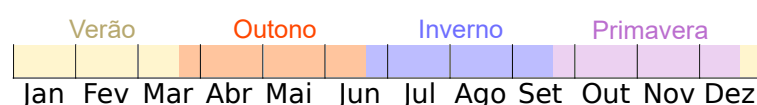


Figura 3.6 – Quatro subdivisões dos períodos do ano para o território brasileiro: verões quentes e chuvosos, invernos secos e outonos com predominância de vento.

A Figura 3.7 exemplifica a estrutura dos dados coletados, com cada linha representando um dia de medição e cada coluna representando uma variável observada. A primeira coluna consiste nos dados de geração fotovoltaica. As demais (2 a 17) compreendem as variáveis climáticas.

	kWh ↓	Temp. Inst. ↓	Temp. Max. ↓	Temp. Min. ↓	Umid. Inst. ↓	***	Veloc. Vento. ↓	Dire. Vento. ↓	Rajad. Vento. ↓	Radiação ↓
	1	2	3	4	5	***	14	15	16	17
1	31612	28.3800	28.9400	27.8000	60.7100	***	2.7200	89.5800	5.8600	591.8300
2	32000	28.6000	29.3600	27.9800	62.5400	***	1.8100	139.0800	4.5100	761.2300
3	41000	29.6200	30.2300	29.1400	61.3300	***	2.2100	91.7500	4.9900	861.8900
4	26000	27.4200	28.0300	27.0400	65.2500	***	2.3400	168.5400	5.8100	551.9500
5	34000	27.2000	27.6100	26.7600	64	***	2.7600	223.8300	6.0300	631.1200
6	35000	26.9800	27.4700	26.4800	66.5400	***	3.4200	203.4600	7.6500	733.5700
7	38000	27.8300	28.2700	27.1700	69.5800	***	1.5300	186.2500	4.6900	692.9800
8	31000	27.9600	28.6800	27.3200	63.7500	***	1.5900	170.2500	5.4500	741.8000
9	15000	25.6200	26.2600	24.9800	73	***	1.8000	153.9200	6.3700	276.2600
10	40000	26.9800	27.4700	26.4500	67.7500	***	1.6000	196.5800	4.4200	767.5200
11	33000	27.2500	27.7200	26.7500	67.6300	***	1.4100	177.4200	3.9700	670.9200
12	41000	28.4300	28.9000	27.8300	59.2500	***	1.2000	153.9200	4.5300	802.7800
13	38000	28.0900	28.6000	27.4500	59.5800	***	1.3000	160	3.9800	780.6700
14	34000	28.5200	29.0500	28.0100	59.1300	***	1.8200	114.8300	4.4500	776.7000
15	40000	28.7300	29.0800	28.3800	64.9200	***	3.1700	51	6.1100	833.8800
16	40000	29.2100	29.5800	28.7900	64.8800	***	3.0700	54.2100	6.2800	868.9000
17	40000	29.1000	29.5800	28.6300	61.6300	***	2.2200	88.0400	5.1700	851.3600

Dados Meteorológicos

Figura 3.7 – Séries temporais consideradas: geração fotovoltaica (kWh) e meteorológicas (colunas de 2 a 17).

Portanto, quando se trata de geração fotovoltaica, os painéis podem ganhar ou perder eficiência devido a inúmeros fatores incontrolláveis, relacionados à estação do ano (ZHANG *et al.*, 2019), como acúmulo de poeira, predominância de nuvens sobre a área de geração, resfriamento de células solares, etc. A separação dos dados em estações visa mitigar esses efeitos para que o modelo de previsão não sofra com as inconsistências que podem ser geradas no processo de treinamento.

Como cada usina de geração fotovoltaica possui capacidade de geração diferente, e os dados climáticos possuem unidades de medida diferentes, foram consideradas duas formas de normalizar os dados, colocando-os em uma escala viável para o processo de otimização por meio de algoritmos de aprendizado de máquina.

A primeira, (também utilizada na etapa 3.1) utiliza os valores máximo e mínimo das séries temporais, ajustando a escala dos dados dentro do intervalo entre 0 e 1. Pode ser observado na Equação 3.16, onde ' y_i ' é o valor observado, ' $min(Y)$ ' o mínimo valor da série temporal e ' $max(Y)$ ' o valor mais alto (ALARAJ *et al.*, 2021):

$$\hat{y}_i = \frac{y_i - \min(Y)}{\max(Y) - \min(Y)} \quad (3.16)$$

A outra técnica de normalização, conhecida como padronização ou método Z-Score, utiliza a média e o desvio padrão da própria série, tornando o valor normalizado centrado em torno da média com desvio padrão unitário (UROLAGIN *et al.*, 2021). O cálculo da padronização é realizado de acordo com a Equação 3.17, onde ' y_i ' é o valor observado, ' μ ' é a média e ' σ ' o desvio padrão.

$$\hat{y}_i = \frac{y_i - \mu}{\sigma} \quad (3.17)$$

É desafiador quando não há histórico de dados suficiente ou, até mesmo, quando existem períodos faltosos em algumas séries de dados. Quando isso acontece e os dados são obtidos de fontes distintas, a preparação requer considerável esforço manual para que se alinhem os períodos de medição da geração com os das variáveis meteorológicas.

3.2.2 Agrupamento de Dias Semelhantes

Após a divisão da série de dados em estações do ano, a técnica de agrupamento hierárquico (ou análise de *cluster*) foi utilizada para agrupar os dias com níveis de similaridade em relação aos dados meteorológicos. Neste caso, consideram-se os valores fornecidos pela instituição meteorológica para o dia desejado.

O número de agrupamentos é um fator do *DOE* e os níveis foram definidos para 3 ou 4. A escolha desses valores aconteceu em função das características climáticas observadas: (a) ensolarado, (b) nublado/úmido, (c) chuvoso ou (d) ventoso/poeira.

3.2.3 Redução de Dimensionalidade usando *PCA*

A motivação para reduzir a dimensionalidade dos dados fundamenta-se pelo fato de a quantidade de dados aumentar no espaço cronológico e, também, à medida que novas unidades de geração são inseridas no sistema. O detalhamento teórico dessa ferramenta multivariada pode ser observado na Seção 2.6, específica sobre *PCA*.

É realizada a redução da dimensionalidade dos dados em dois casos específicos, dependendo do processo metodológico, definido pela consideração, ou não, das variáveis meteorológicas. Quando as variáveis climáticas são consideradas na corrida experimental, o *PCA* é aplicado. Como exemplo, citam-se as colunas 2 a 17 da Figura 3.7.

Em alguns casos, o uso de variáveis climáticas pode não ser interessante e deve ser testado. Isso geralmente se deve a vários fatores, como a distância da estação meteorológica à usina de geração, poeira nos painéis, etc. Quando o processo experimental

não considera as variáveis climáticas, mas apenas as variáveis de geração fotovoltaica, é necessária uma reestruturação dos dados. Nesse caso, o processo de empilhamento de dados para treinamento do modelo é exemplificado na Figura 3.8.

Aqui, são escolhidos 6 dias de geração antes do dia de medição observado. Esses seis dias irão compor os dados de treinamento referentes àquele dia observado, conforme visto na Figura 3.8 com as marcações "A"(verde) e "B"(amarelo). À medida que se percorre a estrutura de dados de geração, a janela deslizante forma novos dados de treinamento para as medições dos dias subsequentes. Finalmente, o *PCA* é aplicado a este conjunto de dados (colunas 2 a 7). A região destacada em vermelho é desconsiderada nesta situação, pois possui muitas células nulas, o que representa ruído para o modelo de previsão.

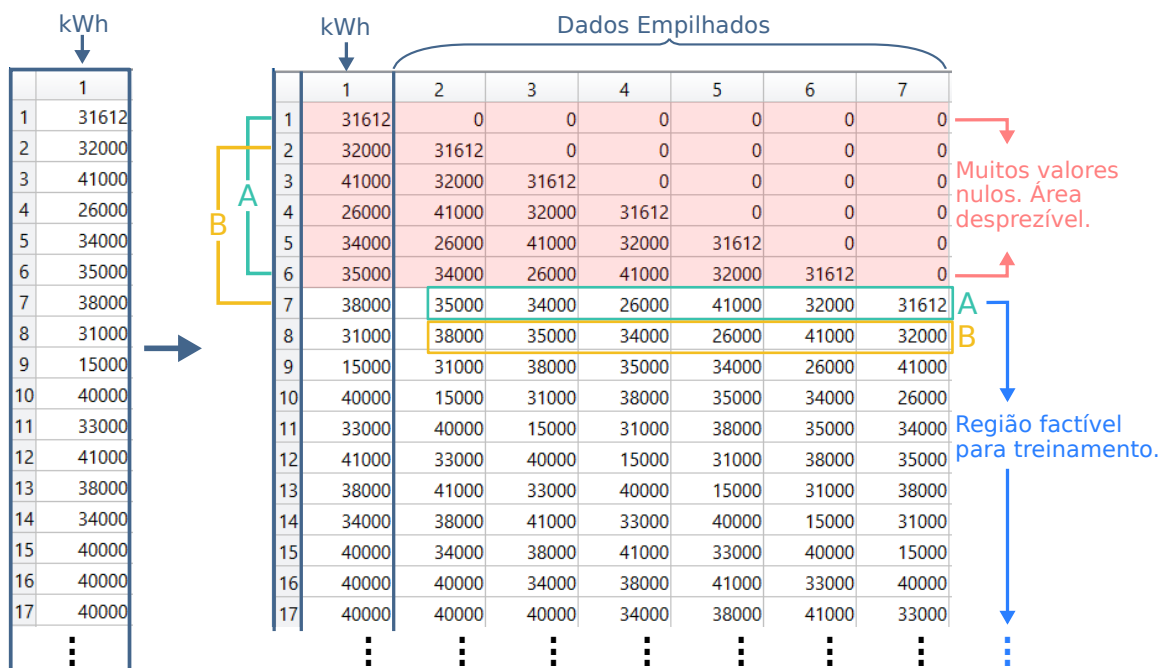


Figura 3.8 – Reorganizando os dados de geração para treinamento: empilhando dados para treinamento com base nas informações dos 6 dias anteriores.

Assim, quando consideradas as variáveis climáticas, há uma redução de dezesseis observações. Quando se considera apenas a geração fotovoltaica, há a formação de seis variáveis para redução de dimensionalidade.

Neste trabalho, a partir do segundo componente, observou-se uma representatividade de mais de 80% da variância dos dados originais. Assim, o uso dos primeiros 2 ou 3 componentes principais é testado em cada corrida experimental.

3.2.4 Parametrização da RNA

A parametrização da RNA seguiu as mesmas definições discutidas anteriormente, na Seção 3.1.2. Foram considerados os parâmetros: número de camadas escondidas, nú-

mero de neurônios em cada camada, taxa de aprendizagem, número de épocas, algoritmo de treinamento e função de transferência

3.2.5 DOE Fatorial

Tabela 3.4 – Fatores do *DOE* para a série temporal com dois níveis cada.

Fatores	Nível 1	Nível 2
(A) Número de Componentes Principais	2	3
(B) Uso de Variável Climática	(s) <i>Sim</i>	(n) <i>Não</i>
(C) Método de Ligação (<i>Clusters</i>)	(wd) 'Ward'	(cp) <i>Completo</i>
(D) Número de <i>Clusters</i>	3	4
(E) Método de Normalização	(zs) 'Padronização'	(mx) 'Min-Max'

Tabela 3.5 – Fatores do *DOE* para a RNA com dois níveis cada.

FATORES	Nível 1	Nível 2
(F) Camadas Escondidas	1	2
(G) Neurônios por Camada	1,5	2
(H) Taxa de Aprendizagem	0,1	0,9
(I) Épocas	100	400
(J) Algoritmo de Treinamento	(scg) <i>Gradiente Conjugado Escalonado</i>	(lm) <i>Levenberg-Marquadt</i>
(K) Função de Transferência	(htg) <i>Sigmóide Tangente Hiperbólica</i>	(rbf) <i>Base Radial</i>

Conforme mostrado na Figura 2.3, este trabalho considerou a resolução de nível IV, pois neste nível os principais efeitos são considerados sem confusão com as interações de dois fatores. Foram considerados 11 fatores na arquitetura experimental, sendo separados em cinco fatores relacionados à série temporal e seis fatores associados à rede neural artificial.

A versatilidade do desenho experimental, relacionada a essência do problema de previsão da geração fotovoltaica, permitiu que a escolha dos fatores e seus respectivos níveis se baseassem em trabalhos anteriores (BALESTRASSI *et al.*, 2009), e um desses trabalhos também considera essa temática como objeto de estudo (MOREIRA *et al.*, 2021).

Sabendo que existem inúmeras combinações de fatores e que cada fator possui inúmeros níveis, esse espaço de busca fica reduzido ao utilizar o DOE e, dessa forma, o analisador pode fazer alterações nos fatores ou níveis e entender o impacto que essa alteração tem na qualidade dos resultados. A Tabela 3.4 detalha os fatores e respectivos níveis relacionados à série temporal e a Tabela 3.5 resume cada fator considerado para a RNA.

3.2.6 *Mixture DOE* para definição dos pesos do *Ensemble*

A definição dos pesos para a combinação dos resultados segue os fundamentos da Seção 2.7, mantendo a mesma essência de cálculo da etapa metodológica 3.1.5. No caso deste estudo, especificamente, há uma tentativa de reduzir o MAPE médio de um conjunto de plantas fotovoltaicas. Assim, a otimização do resultado pode penalizar os resultados de alguma unidade de geração pontualmente pelo fato de o interesse estar voltado para a melhora dos resultados considerados conjuntamente.

A Figura 3.9 detalha o pseudocódigo que automatiza o processo de previsão descrito nos tópicos anteriores. A implementação deste algoritmo ocorreu através do software *Matlab*.

```

1. BEGIN
2.   PVDataSet ← carregar dados de geração fotovoltaica;
3.   ClimDataSet ← carregar dados climáticos;
4.   Separar cada conjunto de dados por estação do ano;
5.   Remover valores nulos de cada conjunto de dados;
6.   doeMatrix ← Gerar a matrix do DOE Fatorial Fracionado com fatores e níveis;
7.   Seasons ← { 'Outono', 'Inverno', 'Primavera', 'Verão' };
8.   FOR s ← 1 TO numeroDeElementos( Seasons )
9.     PVDataSet ← carregarDadosPVdoPeriodo( s );
10.    ClimDataSet ← carregarDadosClimaticosdoPeriodo( s );
11.    FOR run ← 1 TO numeroDeLinhas( doeMatrix )
12.      Verificar se as variáveis climáticas serão utilizadas ou apenas a série de geração;
13.      Normalizar os dados de acordo com a corrida experimental do doeMatrix;
14.      Agrupar dias semelhantes baseado nas características climáticas;
15.      Reduzir a dimensionalidade dos dados usando PCA;
16.      Escolher o número de componentes principais baseado na matrix do DOE;
17.      IF variáveis climáticas devem se consideradas DO
18.        | trainingData ← variáveis climáticas após agrupamento e redução de dimensão;
19.      ELSE
20.        | trainingData ← empilha, agrupa e reduz dimensão dos dados de geração PV;
21.      END IF
22.      FOR counter ← 1 TO numeroDePlantasPV()
23.        prepareAnnNeurons( hiddenLayers, inputsNumber, unitsPerLayer );
24.        Create FeedForwardNet;
25.        Set learningRate, numberOfEpochs and transferFunction;
26.        trainANN( trainingData );
27.        Desnormalizar forecastedValue usando método apropriado;
28.        Calcular MAPE;
29.      END FOR
30.    END FOR
31.  END FOR
32.  Para cada corrida experimental, extrai a média das réplicas (quando aplicável);
33.  Agrupar os resultados usando o método de ligação de WARD e distância Euclideana;
34.  Avaliar se os grupos com menores MAPEs possuem similiaridade estatística;
35.  Criar Arranjo de Misturas usando Simplex Centroid Design;
36.  Escolher a combinação de pesos que reduz o MAPE;
37. END

```

Figura 3.9 – Pseudocódigo proposto para automatizar o processo de previsão fotovoltaica.

4 ESTUDO DE CASO

Este capítulo apresenta simulações que foram realizadas utilizando dados reais (de geração fotovoltaica e meteorológicos). Foram levados em consideração os aspectos metodológicos apresentados nas Seções 3.1 e 3.2

4.1 Usina de Geração Fotovoltaica em Carmo de Minas/MG

O objetivo deste estudo é, através da metodologia 3.1 previamente apresentada, prever a geração de energia em uma planta fotovoltaica. Para isso, foi considerado o horizonte de uma semana à frente, usando dados discretizados diariamente. Há 686 observações na série de dados, cobrindo o período de 21 de maio de 2017 a 6 de abril de 2019. A semana de previsão desejada é de 31 de março a 6 de abril de 2019.

Esses dados foram coletados de uma usina solar com capacidade instalada de aproximadamente 35 kWp, localizada no Instituto Federal do Sul de Minas Gerais, cidade de Carmo de Minas, Brasil. Ao total, são 108 placas instaladas no prédio do Laboratório de Alimentos que podem produzir até 30kWh (IFSULDEMINAS, 2017).



Figura 4.1 – Painéis fotovoltaicos instalados sobre o Laboratório de Alimentos do Instituto Federal do Sul de Minas, Campus Carmo de Minas

Os dados exógenos das séries temporais disponíveis para esta análise incluíram: nebulosidade, horas de insolação, temperatura, precipitação e umidade, e foram obtidos através do Instituto Nacional de Meteorologia - Brasil (INMET). A escolha de uma variável exógena para a preparação do experimento foi selecionada avaliando o coeficiente de correlação de *Pearson*.

A Tabela 4.1 resume as correlações encontradas entre as variáveis e aponta as horas de insolação como a variável com o índice mais alto, em 0,811. Essa variável (horas

de insolação) será utilizada na composição de um fator experimental, cujos dois níveis indicam a presença de uma variável correlacionada ou não (apenas os dados de geração são utilizados neste último caso).

Tabela 4.1 – Análise do coeficiente de correlação de Pearson em variáveis exógenas. Os resultados são estatisticamente significativos com um valor $p < 0,05$.

CORRELAÇÕES	kWh	Nebulosidade	Insolação	Temperatura	Precipitação
Nebulosidade	-0,529				
Insolação	0,811	-0,759			
Temperatura	0,723	-0,225	0,509		
Precipitação	-0,220	0,409	-0,372	-0,135	
Umidade	-0,683	0,639	-0,750	-0,424	0,331

Para esse problema específico, o gráfico de contorno, mostrado na Figura 4.2 permite uma maneira diferente de analisar os dados de geração e de ver tendências mais claramente do que no gráfico de séries temporais (comumente apresentado em linhas). Este gráfico de contorno funciona como um mapa de frequência, onde a cor indica o nível de geração de eletricidade. Os extremos são azuis e amarelos, onde o primeiro representa a geração mínima e o segundo representa a geração máxima. Nos eixos horizontais, tem-se o dia da semana e, nos eixos verticais, o índice da semana.

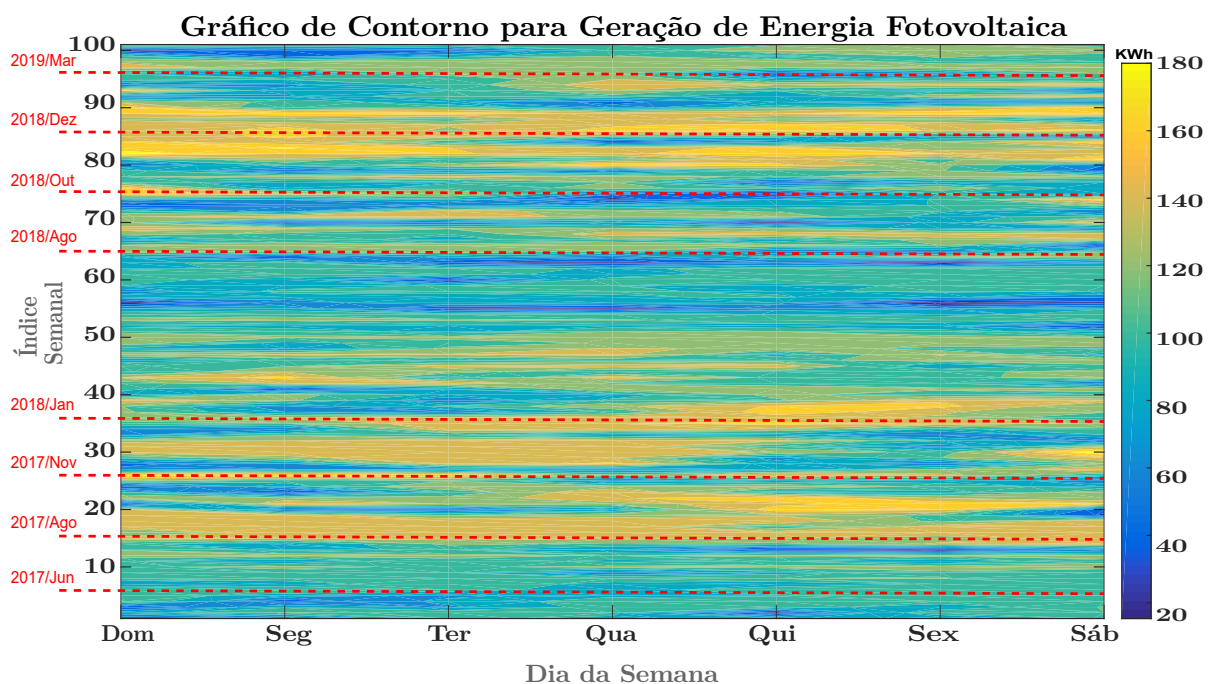


Figura 4.2 – Gráfico de contorno para visualização semanal de geração de energia fotovoltaica de uma planta localizada em Minas Gerais, Brasil.

Por exemplo, pode-se ver na vigésima semana que existe uma zona amarela, o que significa que estava ensolarado nesse período e que a geração estava alta. Na 60^a semana, pode-se ver que houve baixa geração, identificada visualmente pela zona de cor azul. As linhas pontilhadas, na cor vermelha, indicam o ano/mês aproximado dessa semana. Por meio dessas linhas, é possível identificar as épocas do ano em que a geração foi maior ou menor.

4.1.1 Previsão Fotovoltaica Através de RNA e DOE

A condução dos experimentos pode ser observada nas Tabelas 4.2 (matriz *DOE* com seus respectivos fatores e níveis) e 4.3 (resultados de previsão para uma semana à frente). Aqui, as identificações alfabéticas e os respectivos níveis de cada corrida experimental são mapeados nas duas tabelas anteriores (Tabela 3.2 e Tabela 3.3). Em cada execução experimental, é possível observar os *MAPEs* de cada dia da semana, seguidos pela média semanal. O valor médio mínimo encontrado para a semana alvo foi de 4,99% e o valor máximo foi de 46,23%. A última coluna apresenta o desvio padrão para cada execução.

Como se trata de uma série de dados cujo comportamento é altamente volátil (devido à natureza das incertezas climáticas associadas ao processo de geração), pode-se ter erros no processo de previsão que variam de semana para semana e também de região para região.

Nessa perspectiva numérica, pode-se observar que a configuração que contribuiu para a redução do *MAPE* foi: duas camadas ocultas; 100 épocas; três neurônios na camada de entrada; seis neurônios nas camadas intermediárias (correspondendo ao cálculo de $2 \times 3 = 6$); taxa de aprendizado igual a 0,1; algoritmo de treinamento *Levenberg-Marquadt*; *RBF*; série de dados sequenciais (desconsiderando os desempilhados); método de normalização *feature scaling*; uso apenas da série de geração fotovoltaica (desconsiderando a série horas de insolação como variável exógena); e o uso de todas as observações disponíveis (679 medições) para o treinamento da RNA.

Em contrapartida e ainda considerando uma perspectiva numérica, a configuração que contribuiu negativamente (isto é, para o aumento do *MAPE*) foi a seguinte: duas camadas ocultas; 400 épocas; cinco neurônios na camada de entrada; oito neurônios nas camadas intermediárias ($1,5 \times 5 = 7,5$, arredondado para 8); taxa de aprendizado igual a 0,1; o algoritmo de treinamento *Levenberg-Marquadt*; função de transferência sigmóide tangente hiperbólica; série de dados sequencial (desconsiderando os desempilhados); método de normalização *feature scaling*; uso de horas de insolação como uma variável exógena para as entradas da RNA; e o uso de todas as observações disponíveis (679 medições).

É importante destacar que somente a avaliação numérica desses resultados pode

Tabela 4.2 – DOE fatorial fracionário

Run	Fatores da Rede Neural Artificial							Fatores da Série Temporal			
	A	B	C	D	E	F	G	H	I	J	K
1	1	100	3	1,5	0,1	scg	hts	v	fs	n	98
2	2	100	3	1,5	0,1	lm	hts	v	mv	s	98
3	1	400	3	1,5	0,1	lm	rbf	v	fs	n	679
4	2	400	3	1,5	0,1	scg	rbf	v	mv	s	679
5	1	100	5	1,5	0,1	lm	rbf	f	mv	n	98
6	2	100	5	1,5	0,1	scg	rbf	f	fs	s	98
7	1	400	5	1,5	0,1	scg	hts	f	mv	n	679
8	2	400	5	1,5	0,1	lm	hts	f	fs	s	679
9	1	100	3	2	0,1	scg	rbf	f	mv	s	679
10	2	100	3	2	0,1	lm	rbf	f	fs	n	679
11	1	400	3	2	0,1	lm	hts	f	mv	s	98
12	2	400	3	2	0,1	scg	hts	f	fs	n	98
13	1	100	5	2	0,1	lm	hts	v	fs	s	679
14	2	100	5	2	0,1	scg	hts	v	mv	n	679
15	1	400	5	2	0,1	scg	rbf	v	fs	s	98
16	2	400	5	2	0,1	lm	rbf	v	mv	n	98
17	1	100	3	1,5	0,9	scg	hts	f	fs	s	679
18	2	100	3	1,5	0,9	lm	hts	f	mv	n	679
19	1	400	3	1,5	0,9	lm	rbf	f	fs	s	98
20	2	400	3	1,5	0,9	scg	rbf	f	mv	n	98
21	1	100	5	1,5	0,9	lm	rbf	v	mv	s	679
22	2	100	5	1,5	0,9	scg	rbf	v	fs	n	679
23	1	400	5	1,5	0,9	scg	hts	v	mv	s	98
24	2	400	5	1,5	0,9	lm	hts	v	fs	n	98
25	1	100	3	2	0,9	scg	rbf	v	mv	n	98
26	2	100	3	2	0,9	lm	rbf	v	fs	s	98
27	1	400	3	2	0,9	lm	hts	v	mv	n	679
28	2	400	3	2	0,9	scg	hts	v	fs	s	679
29	1	100	5	2	0,9	lm	hts	f	fs	n	98
30	2	100	5	2	0,9	scg	hts	f	mv	s	98
31	1	400	5	2	0,9	scg	rbf	f	fs	n	679
32	2	400	5	2	0,9	lm	rbf	f	mv	s	679

Tabela 4.3 – Erros percentuais médios absolutos (MAPEs) associados a cada dia da semana e MAPE médio.

Run	MAPEs							Média	Desv. Pad.
	Dom	Seg	Ter	Qua	Qui	Sex	Sáb		
1	35,43	35,88	6,96	13,89	35,64	22,51	46,41	28,1	14,06
2	26,04	26,52	22,85	27,95	16,42	67,13	42,83	32,82	17,1
3	10,52	32,9	15,73	20,21	24,44	20,48	4,51	18,4	9,28
4	21,09	44,03	13,57	8,28	17,43	28,85	47,19	25,78	14,99
5	11,94	12,38	17,76	11,04	18,72	2,81	3,62	11,18	6,18
6	13,22	19,4	12,67	21,79	11,08	40,82	41,96	22,99	13,14
7	9,4	9,48	18,41	7,96	16,69	5,16	4,7	10,26	5,35
8	3,96	46,86	38,57	49,79	36,61	73,2	74,6	46,23	24,09
9	4	5,81	0,08	8,18	1,34	25,09	26,1	10,08	10,93
10	0,5	0,78	8,1	4,55	17,7	1,18	2,09	4,99	6,22
11	20,1	43,31	35,23	46,15	33,32	69,02	70,39	45,36	18,61
12	4,23	0,87	7,81	1,53	12,1	10,42	10,14	6,73	4,53
13	21,08	24,36	17,1	24,86	6,13	31,87	41,64	23,86	11,17
14	1,47	16,27	21,31	42,11	29,68	24,04	22,1	22,42	12,39
15	26,5	18,87	10,58	17,9	4,66	37,76	44,31	22,94	14,25
16	16,65	1,45	5,12	21,58	16,72	3,79	29,81	13,59	10,5
17	3,53	8,45	35,65	46,85	33,94	69,65	39,52	33,94	22,55
18	3,6	1,57	12,33	11,29	23,88	7,34	8,86	9,84	7,3
19	21,07	19,43	12,69	21,81	11,1	40,85	41,99	24,14	12,49
20	10,79	8,29	7,72	8,07	1,91	30,4	5,24	10,34	9,27
21	15,61	26,62	19,12	29,46	3,5	60,79	55,17	30,04	20,9
22	21,06	16,15	24,55	20,2	25,12	36,45	3,22	20,97	10,07
23	26,55	24,59	26,94	35,87	14,82	61,94	64,27	36,43	19,24
24	2,08	18,23	8,26	51,24	14,78	12,52	1,45	15,51	16,95
25	0,69	1,04	0,07	0,43	22,73	6,75	10,41	6,02	8,36
26	23,57	23,11	15,59	25,07	12,57	40,4	40,82	25,87	11,03
27	8,47	12,84	11,88	6,79	14,05	7,8	10,26	10,3	2,74
28	3,15	10,35	13,69	51,31	5,32	39,4	35,71	22,71	19,08
29	9,72	13,27	13,85	12,62	19,72	7,79	6,25	11,89	4,49
30	18,79	43,25	35,17	46,11	33,26	68,95	70,31	45,12	18,89
31	22,17	17,78	22,42	16,13	23,51	3,03	2,24	15,33	9,07
32	17,25	11,01	4,7	13,41	3,52	30,78	29,66	15,76	10,96

induzir alguém a interpretar erroneamente os fatores que impactam significativamente a qualidade da previsão. Portanto, a metodologia proposta, baseada no *DOE*, permite identificar quais fatores são estatisticamente significativos em termos de promoção de melhorias na redução do *MAPE*. Essa inferência é importante para orientar previsões para diferentes períodos.

Através de um gráfico de efeitos principais, pode-se observar que dois fatores obtiveram valores de p inferiores a 5%: a função de transferência e o uso de variáveis correlacionadas (neste caso, horas de insolação).

A partir da Figura 4.3, pode-se ver isso acontecendo e a interpretação desse gráfico é fácil. Cada caixa representa um fator e sua variação, do nível 1 ao nível 2. Quanto mais horizontal a linha, maior a indicação de que, se houver uma variação no nível do fator, haverá pouca ou nenhuma variabilidade na resposta (que neste caso é o *MAPE*). Uma inclinação mais acentuada da linha indica que esse fator tende a impactar a resposta quando há variações em seus níveis.

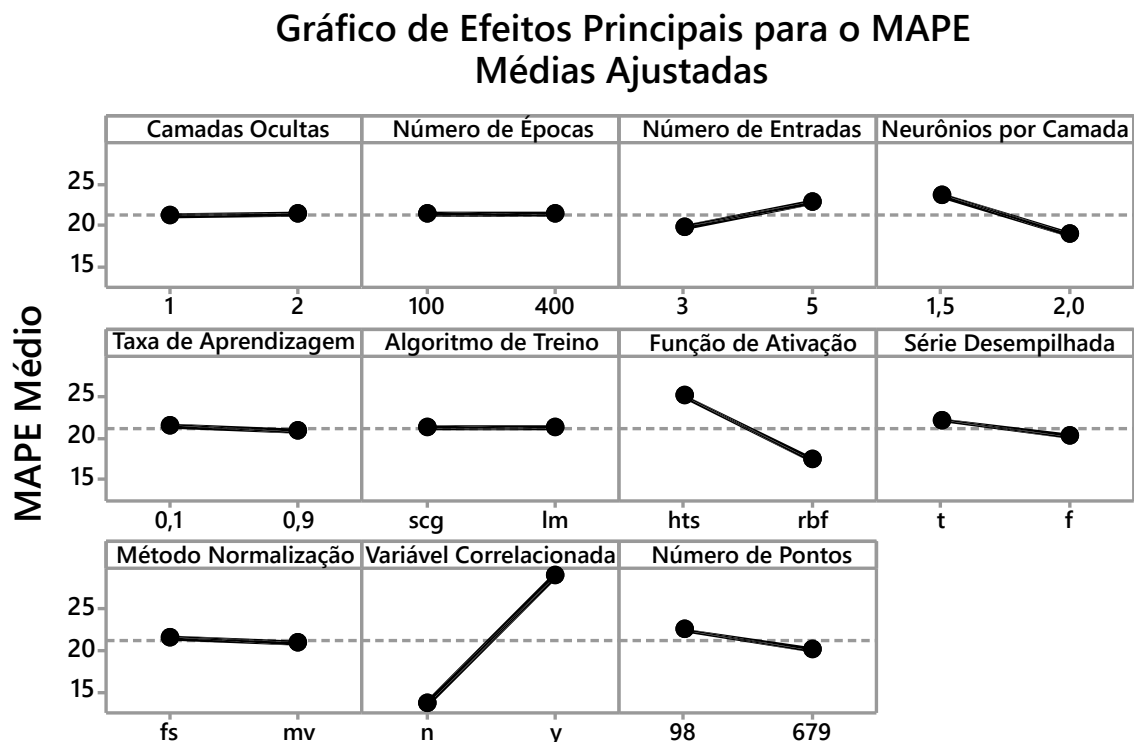


Figura 4.3 – Efeitos principais para o *MAPE*.

Neste ponto, calcula-se as previsões e seus respectivos erros. Objetiva-se usar as melhores previsões para promover uma combinação que reduz o *MAPE* médio para a semana desejada. Naturalmente, a escolha dos componentes para compor essa combinação, conhecida como *ensemble*, deve ser realizada sistematicamente. A partir do dendograma (Figura 4.4), pode-se identificar um grupo de interesse, composto por quatro elementos

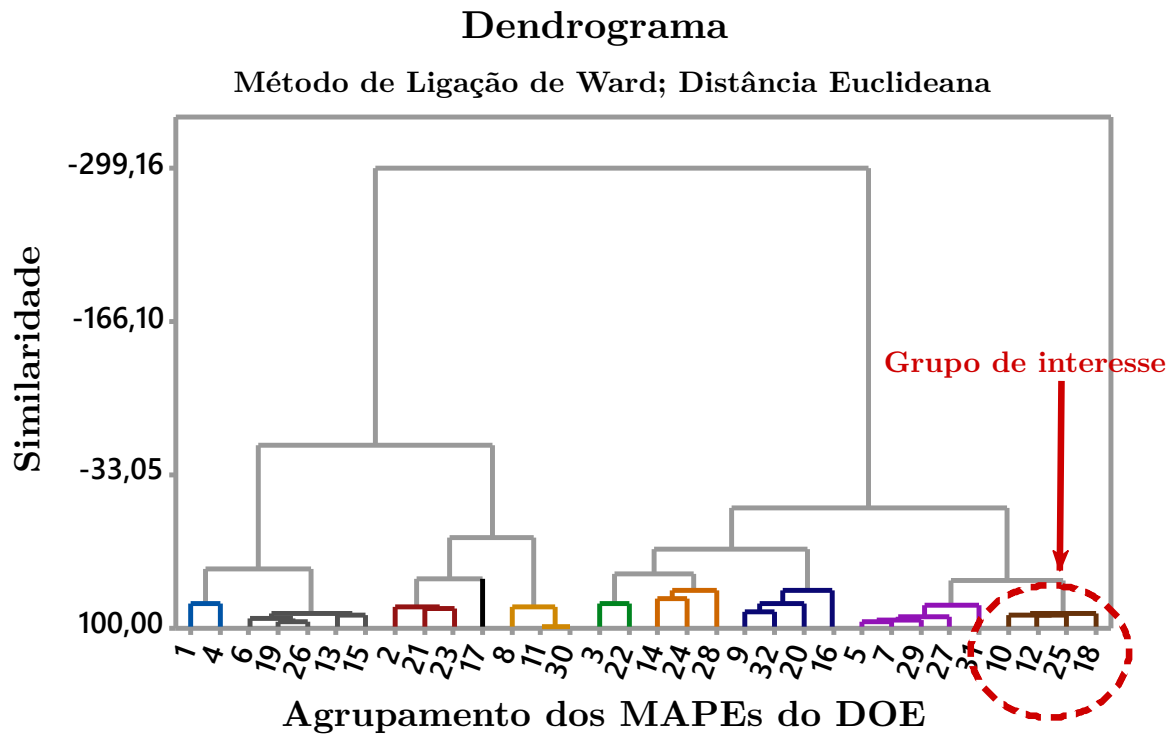


Figura 4.4 – Dendrograma usando ligação de *Ward* com distância euclidiana.

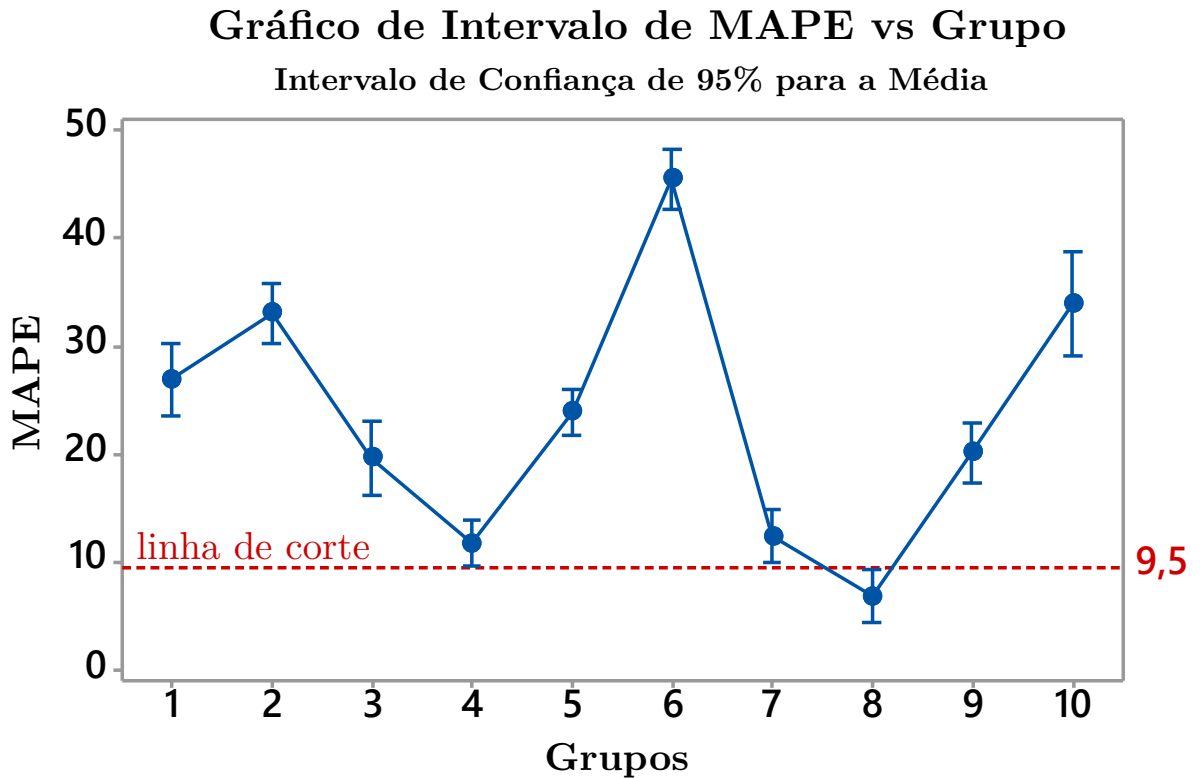


Figura 4.5 – Análise variância usando o procedimento de comparação de *Tukey*.

das execuções experimentais anteriores: linhas 10, 12, 18 e 25.

A separação estatística foi realizada com base no método de ligação de *Ward*, com distância de medida euclidiana e 10 grupos (ou *clusters*), definidos empiricamente. Uma análise de variância unidirecional usando o procedimento de comparação de *Tukey* (Figura 4.5) foi realizada para garantir que os grupos fossem estatisticamente diferentes entre si e que apresentassem o menor *MAPE* médio (numericamente). Na Figura 4.5, pode-se ver que a linha de corte aparece com um *MAPE* médio de aproximadamente 9,5%. O grupo 8 da Figura 4.5 corresponde ao grupo de interesse da Figura 4.4.

Tendo escolhido o grupo com as melhores previsões, consideradas individualmente, espera-se que a combinação desses resultados leve a uma redução do *MAPE* semanal. Diante desta ótica, a próxima Seção discute a estratégia baseada em *MDE* aplicada para a formação de conjuntos.

4.1.2 Combinação dos Resultados

Como discutido anteriormente, a principal motivação deste estudo é construir uma combinação sistemática dos valores previstos para diminuir o erro de previsão. A técnica de combinação foi baseada na definição dos pesos que forneceriam o *MAPE* mais baixo, usando o *MDE*. Para este caso específico, um grupo contendo quatro elementos foi selecionado, representando os valores previstos das execuções experimentais 10, 12, 18 e 25, que são w_1 , w_2 , w_3 e w_4 , respectivamente. A Tabela 4.4 mostra os dados da mistura.

Cada linha assume uma combinação com um peso predeterminado. Nesse caso, deve-se notar que as quatro primeiras linhas desse projeto de mistura consideram apenas a previsão individual, com peso 1 e zero para as demais.

A partir da Tabela 4.4, verifica-se que a configuração de pesos que reduz o *MAPE* médio semanal tem um efeito maior para o primeiro componente da previsão, indicado por w_1 . Assim, a representação geométrica dessa distribuição de peso é mostrada na Figura 4.6, onde há a formação de um tetraedro. As coordenadas de posicionamento do ponto ideal do *ensemble* transcrevem como $(w_1, w_2, w_3, w_4) = (0,625, 0,125, 0,125, 0,125)$. Nesse caso, o *MAPE* médio caiu para 4,70%.

4.1.3 Região de Confiança

A região de confiança permite verificar se os dados são estatisticamente semelhantes entre si no mesmo intervalo. Os pontos encontrados fora da região delimitada pela elipse não possuem as mesmas características e podem ser considerados estatisticamente diferentes. Nesse sentido, quer-se verificar se os resultados da previsão pertencem ao elipsóide. Na Figura 4.7, vê-se que os pontos destacados em vermelho correspondem à previsão combinada e estão dentro dos limites do elipsóide.

Para construção elipsóide, usa-se a seguinte matriz S de variância-covariância, com

Tabela 4.4 – DOE de misturas para a definição dos pesos do *ensemble*

Pontos de Mistura				MAPEs Individuais							Média	Desv. Pad.
10 (w_1)	12 (w_2)	18 (w_3)	25 (w_4)	Dom	Seg	Ter	Qua	Qui	Sex	Sáb		
1,000	0,000	0,000	0,000	0,5	0,78	8,1	4,55	17,7	1,18	2,09	4,99	6,22
0,000	1,000	0,000	0,000	4,23	0,87	7,81	1,53	12,1	10,42	10,14	6,73	4,53
0,000	0,000	1,000	0,000	3,6	1,57	12,33	11,29	23,88	7,34	8,86	9,84	7,3
0,000	0,000	0,000	1,000	0,69	1,04	0,07	0,43	22,73	6,75	10,41	6,02	8,36
0,500	0,500	0,000	0,000	2,36	0,83	7,95	3,04	14,9	4,62	4,02	5,39	4,74
0,500	0,000	0,500	0,000	2,05	0,4	10,21	7,92	20,79	4,26	5,48	7,3	6,81
0,500	0,000	0,000	0,500	0,59	0,91	4,01	2,06	20,22	2,78	4,16	4,96	6,87
0,000	0,500	0,500	0,000	3,91	0,35	10,07	6,41	17,99	1,54	0,64	5,84	6,39
0,000	0,500	0,000	0,500	2,46	0,96	3,87	0,55	17,42	8,58	10,28	6,3	6,15
0,000	0,000	0,500	0,500	2,14	0,27	6,13	5,43	23,3	0,29	0,77	5,48	8,22
0,333	0,333	0,333	0,000	2,78	0,03	9,41	5,79	17,89	0,63	0,27	5,26	6,54
0,333	0,333	0,000	0,333	1,8	0,9	5,28	1,88	17,51	5,33	6,15	5,55	5,67
0,333	0,000	0,333	0,333	1,59	0,08	6,78	5,14	21,44	0,59	0,18	5,11	7,66
0,000	0,333	0,333	0,333	2,84	0,11	6,69	4,13	19,57	3,28	3,9	5,79	6,38
0,250	0,250	0,250	0,250	2,25	0,28	7,04	4,23	19,1	2,16	2,4	5,35	6,42
0,625	0,125	0,125	0,125	1,37	0,53	7,57	4,39	18,4	0,49	0,15	4,7	6,62
0,125	0,625	0,125	0,125	3,24	0,58	7,42	2,88	15,6	6,29	6,27	6,04	4,85
0,125	0,125	0,625	0,125	2,93	0,65	9,68	7,76	21,49	2,59	3,23	6,9	7,17
0,125	0,125	0,125	0,625	1,47	0,66	3,48	1,9	20,92	4,46	6,4	5,61	7,03

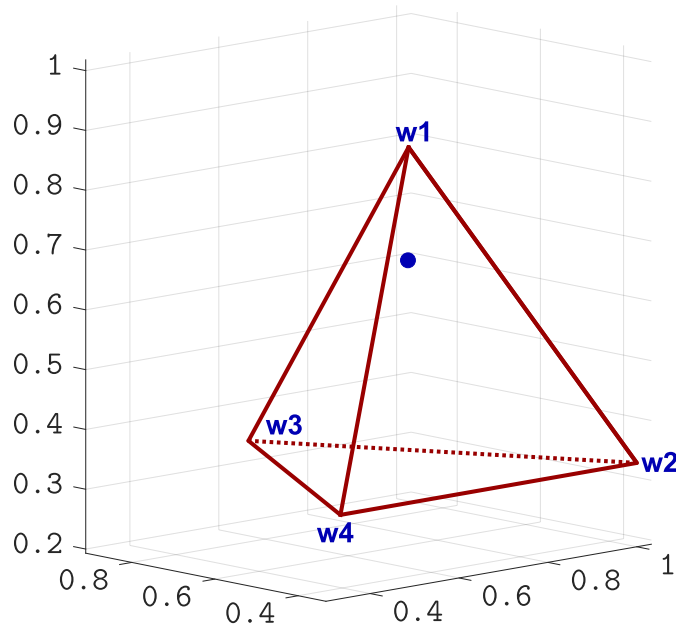


Figura 4.6 – Tetraedro representando uma configuração de pesos que compõe a combinação $(w_1, w_2, w_3, w_4) = (0,625, 0,125, 0,125, 0,125)$.

base em dados de geração e horas de insolação, a qual é centrada nas médias de 113,8 (kWh) e 6,7 (horas de insolação):

$$S = \begin{bmatrix} 1167,35 & 79,50 \\ 79,50 & 9,21 \end{bmatrix} \quad (4.1)$$

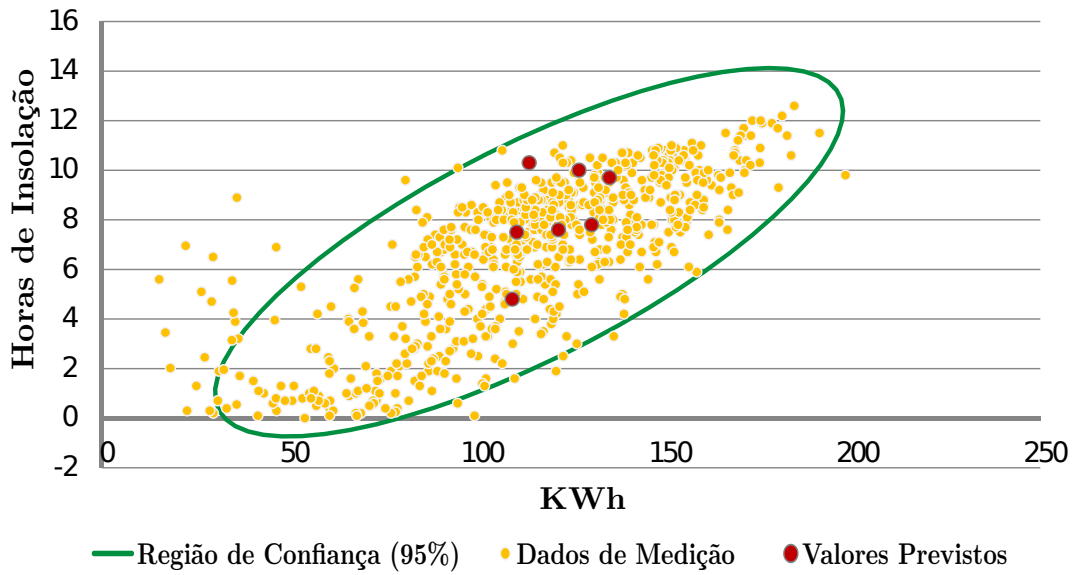


Figura 4.7 – Elipsóide de confiança para a semana de previsão desejada.

Os autovalores foram calculados para $\lambda_1 = 1172,78$ e para $\lambda_2 = 3,78$. Os autovetores associados de S foram armazenados na seguinte matriz:

$$P = \begin{bmatrix} e_1 & e_2 \\ 0,998 & -0,068 \\ 0,068 & 0,998 \end{bmatrix} \quad (4.2)$$

Assim, considera-se que este estudo de caso teve sucesso em seus objetivos.

4.2 Conjunto de Usinas Localizadas em Diferentes Regiões do País

Uma vez que no Brasil a geração fotovoltaica representa a maior parcela do crescimento das energias renováveis (RIGO *et al.*, 2022), este estudo de caso considerou dezessete usinas de geração fotovoltaica localizadas em diferentes regiões do Brasil. Aqui será considerado o aspecto metodológico discutido na Seção 3.2.

A previsão é realizada com um dia de antecedência, considerando cada estação do ano, ou seja, ao final da execução dos experimentos, haverá quatro previsões para cada usina de geração (uma para cada estação do ano). O dia de previsão foi escolhido aleatoriamente, visto que o intervalo de tempo de cada usina de geração nem sempre coincidia (pela falta de dados, por exemplo).

O número de componentes principais, como fator *DOE*, pode ser considerado um dos itens-chave desta simulação, pois leva à redução da dimensionalidade das variáveis climáticas. Os níveis variam entre dois e três, o que significa que às vezes são usados dois componentes para treinar o modelo e às vezes são usados três componentes.

A redução na dimensionalidade implica uma pequena perda de informação. Assim, é interessante apresentar o percentual acumulado da variância em cada nível, para cada cidade (usina de geração).

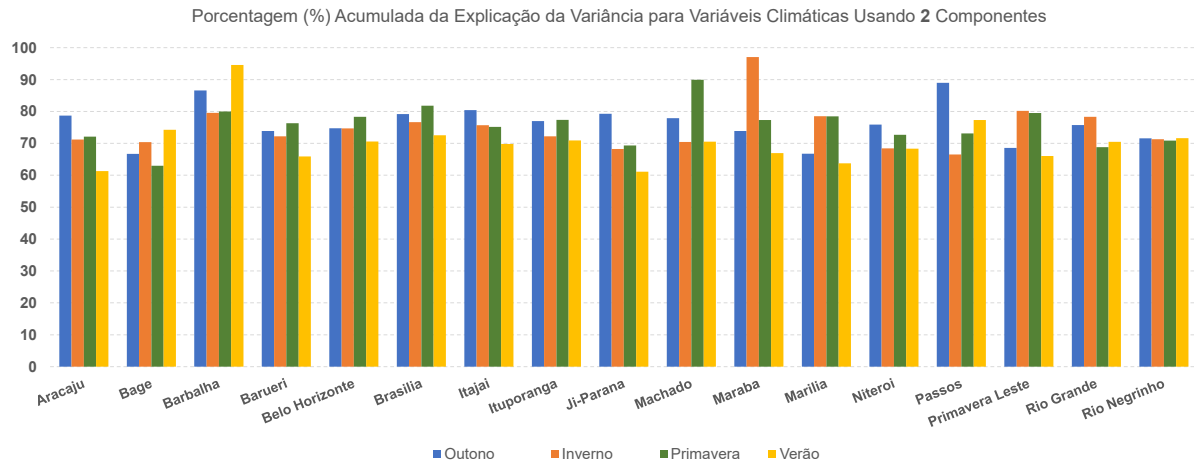


Figura 4.8 – Visão geral do percentual de representação da variância das variáveis climáticas, relacionadas a cada usina de geração fotovoltaica, considerando dois componentes principais.

A partir da Figura 4.8, observa-se que a explicação da variância das variáveis climáticas, para cada unidade de geração e considerando cada estação do ano, é superior a 60%, com uma média aproximada de 75% da explicação total. Para esta demonstração gráfica, foi utilizado o processo de normalização 'MaxMin'.

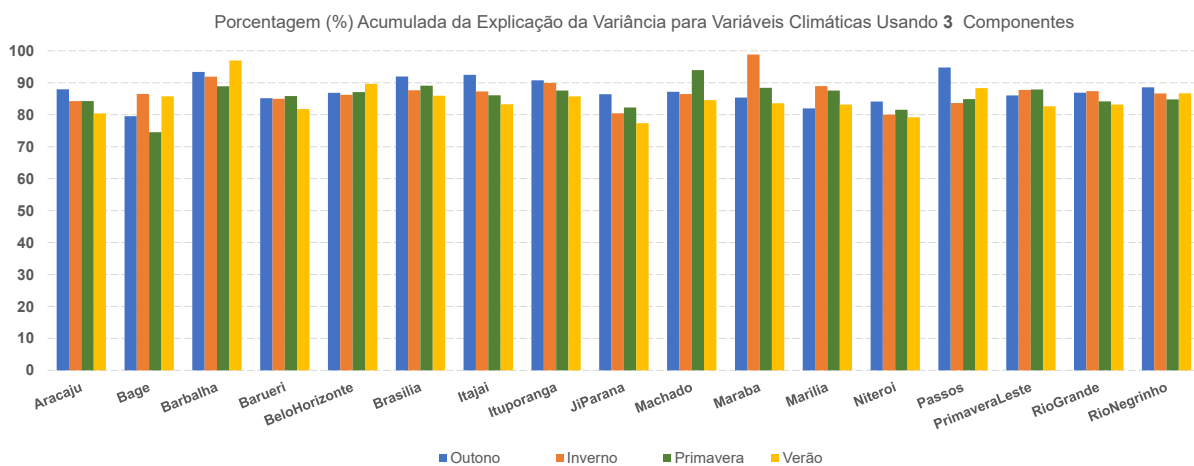


Figura 4.9 – Visão geral do percentual de representação da variância das variáveis climáticas, relacionadas a cada usina de geração fotovoltaica, considerando três componentes principais.

Ao considerar três componentes, percebe-se um aumento natural na explicação da variância das variáveis climáticas, o que é mostrado na Figura 4.9. Nesse caso, o percentual de explicação acumulado é superior a 75% para todas as usinas e estações de geração, com uma média de 85% de explicação para todas as estações. Isso significa que

se forem utilizados três componentes, há uma maior representação do conjunto de dados, o que implica em mais informações para ajuste e treinamento do modelo de previsão.

A ideia principal desta escolha é justamente testar, usando a ferramenta estatística *DOE*, se há interferência nos resultados da previsão quando um componente extra é considerado (ou não).

4.2.1 Estágio de Preparação do Experimento

Foram utilizados dados reais de geração fotovoltaica do repositório *PVOutput.org* ([PVOUTPUT, 2022](#)), com resolução diária, exceto os dados das usinas de geração das cidades de Machado e Passos, que foram adquiridos do Instituto Federal do Sul de Minas Gerais - IFSULDEMINAS. Dezesete unidades geradoras são consideradas neste estudo, cada uma com uma capacidade de geração diferente e geograficamente separadas em todo o território brasileiro. O motivo da escolha dessas unidades se deu pela disponibilidade e qualidade dos dados no horizonte temporal do estudo. Dados ausentes ou nulos foram desconsiderados.

Para facilitar a coleta desses dados, de forma automatizada, foi implementado um *script* na linguagem de programação *Java* que faz uma solicitação ao repositório e baixa as séries de dados com discretização diária, organizando-as por planta de geração. É importante destacar uma limitação que foi observada em relação à disponibilidade de dados: há mais de 17 usinas de geração disponíveis no repositório, mas muitas delas não possuem as respectivas informações climáticas, que foram adquiridas de outro banco de dados, o Instituto Nacional de Meteorologia – INMET ([INMET, 2022](#)).

A etapa de coleta de dados é desafiadora, pois muitas dessas informações possuem dados ausentes, com ruídos e muitas vezes sem acesso público. Assim, como as estações meteorológicas estão dispersas por todo o território brasileiro, nem sempre coincidem com a proximidade de uma determinada usina de geração fotovoltaica ou mesmo com a disponibilidade de dados climáticos, o que reduz o número de casos experimentais.

Além do fato dos dados climáticos não estarem disponíveis para todas as usinas de geração (e vice-versa), há também o desafio de sincronizar os períodos de medição: muitas vezes há dados de geração para um intervalo de datas, mas não há informações de previsão do tempo para o mesmo período. Quando isso acontece, esse intervalo de tempo deve ser descartado.

Esta é a razão pela qual não foi possível considerar o mesmo dia de previsão (respeitando a estação do ano) para todas as unidades geradoras. Após eliminar os dados ausentes e sincronizar os períodos de data, os dados foram separados sazonalmente para que o treinamento do modelo de previsão ocorresse apenas com os dados específicos daquela estação do ano.

Tabela 4.5 – Detalhamento das plantas de geração fotovoltaica utilizadas.

Usina (nome da cidade)	Estado	Tamanho do Sistema (kW)	Período dos Dados
Bagé	Rio Grande do Sul	10.600	2017 - 2018
Barbalha	Ceará	1.100	2019 - 2021
Barueri	São Paulo	12.100	2018 - 2021
Belo Horizonte	Minas Gerais	7.200	2019 - 2021
Brasília	Distrito Federal	4.950	2018 - 2021
Itajaí	Santa Catarina	2.200	2016 - 2019
Ituporanga	Santa Catarina	27.00	2020 - 2021
Ji-Paraná	Rondônia	2.295	2019 - 2021
Machado	Minas Gerais	365	2018 - 2020
Marabá	Pará	5.940	2019 - 2021
Marília	São Paulo	5.500	2017 - 2021
Niterói	Rio de Janeiro	5.775	2019 - 2021
Passos	Minas Gerais	180	2019 - 2021
Primavera do Leste	Mato Grosso	16.640	2019 - 2021
Rio Grande	Rio Grande do Sul	3.640	2017 - 2021
Rio Negrinho	Santa Catarina	4.960	2017 - 2021

Esse processo foi considerado porque se acredita que cada estação do ano possui características próprias e que pode afetar a geração de energia. Por exemplo, o excesso de poeira devido ao tempo seco, ou mesmo a passagem de nuvens em períodos de chuva, podem alterar o comportamento e a correlação dos dados.

Os dados climáticos foram obtidos através do Banco de Dados Meteorológicos do Instituto Nacional de Meteorologia (INMET) (INMET, 2022), considerando as estações meteorológicas mais próximas das usinas de geração fotovoltaica previamente selecionadas, contemplando dezesseis parâmetros: temperatura instantânea ($^{\circ}\text{C}$), temperatura máxima ($^{\circ}\text{C}$), temperatura mínima ($^{\circ}\text{C}$), umidade instantânea (%), umidade máxima (%), umidade mínima (%), precipitação instantânea ($^{\circ}\text{C}$), precipitação máxima ($^{\circ}\text{C}$), precipitação mínima ($^{\circ}\text{C}$), pressão instantânea (hPa), pressão máxima (hPa), pressão mínima (hPa), velocidade do vento (m/s), direção do vento ($^{\circ}$), rajada de vento (m/s) e radiação (KJ/m^2).

Para facilitar a identificação de cada usina de geração fotovoltaica, e seus respectivos dados climáticos, foi considerada a cidade mais próxima daquele ponto de medição e essas características estão listadas na Tabela 4.5.

A Tabela 4.6 elenca as 32 corridas experimentais baseadas no arranjo do DOE fatorial fracionário em sua resolução IV. A matriz experimental é dividida em fatores da série e fatores da RNA. Os níveis de cada fator estão descritos em detalhes nas Tabelas 3.4 e 3.5.

Tabela 4.6 – Estrutura da Matriz Experimental do DOE.

Run	Fatores da Série Temporal					Fatores da RNA					
	A	B	C	D	E	F	G	H	I	J	K
1	2	s	wd	3	zs	1	1.5	0.1	100	scg	htg
2	3	s	wd	3	zs	2	1.5	0.1	400	lm	htg
3	2	f	wd	3	zs	2	2	0.1	100	scg	rbf
4	3	f	wd	3	zs	1	2	0.1	400	lm	rbf
5	2	s	cp	3	zs	2	2	0.9	400	scg	htg
6	3	s	cp	3	zs	1	2	0.9	100	lm	htg
7	2	f	cp	3	zs	1	1.5	0.9	400	scg	rbf
8	3	f	cp	3	zs	2	1.5	0.9	100	lm	rbf
9	2	s	wd	4	zs	1	2	0.9	400	lm	rbf
10	3	s	wd	4	zs	2	2	0.9	100	scg	rbf
11	2	f	wd	4	zs	2	1.5	0.9	400	lm	htg
12	3	f	wd	4	zs	1	1.5	0.9	100	scg	htg
13	2	s	cp	4	zs	2	1.5	0.1	100	lm	rbf
14	3	s	cp	4	zs	1	1.5	0.1	400	scg	rbf
15	2	f	cp	4	zs	1	2	0.1	100	lm	htg
16	3	f	cp	4	zs	2	2	0.1	400	scg	htg
17	2	s	wd	3	mx	1	1.5	0.9	100	lm	rbf
18	3	s	wd	3	mx	2	1.5	0.9	400	scg	rbf
19	2	f	wd	3	mx	2	2	0.9	100	lm	htg
20	3	f	wd	3	mx	1	2	0.9	400	scg	htg
21	2	s	cp	3	mx	2	2	0.1	400	lm	rbf
22	3	s	cp	3	mx	1	2	0.1	100	scg	rbf
23	2	f	cp	3	mx	1	1.5	0.1	400	lm	htg
24	3	f	cp	3	mx	2	1.5	0.1	100	scg	htg
25	2	s	wd	4	mx	1	2	0.1	400	scg	htg
26	3	s	wd	4	mx	2	2	0.1	100	lm	htg
27	2	f	wd	4	mx	2	1.5	0.1	400	scg	rbf
28	3	f	wd	4	mx	1	1.5	0.1	100	lm	rbf
29	2	s	cp	4	mx	2	1.5	0.9	100	scg	htg
30	3	s	cp	4	mx	1	1.5	0.9	400	lm	htg
31	2	f	cp	4	mx	1	2	0.9	100	scg	rbf
32	3	f	cp	4	mx	2	2	0.9	400	lm	rbf

4.2.2 Previsão Sazonal de Um Dia à Frente

A previsão é realizada para cada usina de geração almejando um dia para cada estação do ano. A partir da matriz *DOE*, todas as 32 execuções experimentais devem ser executadas. Esse processo traduz, essencialmente, a variação paramétrica no espaço de busca. Para cada execução experimental a Rede Neural Artificial é reinicializada a fim de evitar interferências nos resultados de uma execução experimental para outra.

Os experimentos foram realizados de forma automatizada, cujo algoritmo foi implementado na linguagem *Matlab*®. Assim, o *MAPE* médio de cada estação do ano está incluído na Tabela 4.7, para todas as corridas experimentais definidas anteriormente.

Os gráficos de efeitos principais permitem ao analisador visualizar os parâmetros que podem influenciar positivamente (ou negativamente) a previsão. Assim, destaca-se como uma das vantagens desta metodologia a flexibilidade de análise das previsões por época, em termos dos parâmetros que mais influenciam todo o processo.

Considerando o *MAPE* médio de cada estação do ano como resposta do *DOE*, os principais efeitos dessa execução podem ser identificados. Aqui, os eixos verticais do gráfico mostram a variação média do erro *MAPE*, enquanto os níveis de cada fator são distribuídos nos eixos horizontais, isto é, no caso da previsão fotovoltaica, quanto menor o erro melhor é o nível do parâmetro. A Figura 4.10 agrupou quatro gráficos, sendo 4.10(A) os principais efeitos do Outono, 4.10(B) os principais efeitos do Inverno, 4.10(C) os principais efeitos da Primavera e 4.10(D) os principais efeitos do Verão.

O outono, representado pela Figura 4.10(A), apresentou características interessantes quanto à escolha de alguns parâmetros, destacando-se o número de componentes principais, sendo três bem ajustados; o número de camadas ocultas foi definido como dois; e o algoritmo de treinamento foi '*Levenberg-Marquardt*'. Os demais parâmetros, com menor variação no erro, como o método de normalização, verifica-se que '*minMax*' se ajusta bem; o uso de variáveis climáticas, neste caso, teve pouco efeito nos resultados; o método de ligação para agrupamentos de dias foi '*completo*'; o número de grupos para formação de *clusters* foi 4; o número de unidades por camada 2; taxa de aprendizado com pouca variação de um parâmetro para outro; o número de épocas permaneceu em 100; e a função de transferência com pouca variação no erro. O outono foi a única estação do ano que se diferenciou das expectativas de trabalho quanto ao uso de variáveis climáticas.

Já o inverno, representado pela Figura 4.10(B), enfatiza o uso de variáveis climáticas na previsão com interferência significativa na variação do erro. O método de normalização que melhor se ajustou para esta estação também foi '*minMax*'; o método de ligação representou pouca variação no erro; o número de *clusters* foi 4; o número de componentes principais foi 3; o número de camadas ocultas 2; o número de elementos por camada dificilmente altera o erro, assim como o número de épocas e a função de

Tabela 4.7 – Resultados da previsão em termos de seus erros médios (*MAPEs*) após cada execução experimental, separados por estação do ano.

<i>Run</i>	Outono		Inverno		Primavera		Verão	
	<i>MAPE</i>	<i>STD</i>	<i>MAPE</i>	<i>STD</i>	<i>MAPE</i>	<i>STD</i>	<i>MAPE</i>	<i>STD</i>
1	17,62	3,79	15,63	3,11	15,19	3,70	14,22	1,68
2	15,72	2,23	15,50	3,19	9,71	1,46	13,76	0,86
3	15,30	1,68	16,21	0,83	17,35	3,06	20,21	2,12
4	19,14	2,22	15,40	1,34	16,26	2,09	14,24	2,97
5	14,99	1,61	13,30	1,18	11,42	0,78	15,36	1,06
6	12,90	2,14	14,70	3,54	10,25	0,57	16,66	3,15
7	17,89	1,35	19,15	2,43	20,20	2,64	19,79	3,46
8	16,56	1,77	15,86	1,42	17,11	2,94	17,43	2,14
9	17,35	1,72	14,86	1,59	11,50	1,25	16,75	0,99
10	17,23	1,89	15,15	1,32	10,35	1,48	13,51	1,96
11	19,28	3,74	14,99	2,03	17,50	2,55	19,69	5,34
12	20,26	2,30	16,62	3,25	16,81	1,35	16,17	1,50
13	15,95	1,21	15,19	1,03	9,21	1,46	14,54	1,96
14	18,10	2,09	12,97	2,13	10,78	2,36	14,49	1,14
15	21,06	1,60	18,55	1,06	16,59	2,83	20,05	5,16
16	14,18	2,00	15,84	2,22	16,77	1,67	17,89	2,35
17	17,94	2,31	15,32	0,67	15,43	1,86	13,91	1,17
18	18,55	1,76	18,51	3,33	13,38	3,30	14,08	1,97
19	18,90	3,78	15,57	1,49	15,13	2,48	17,57	3,51
20	16,21	2,96	13,79	2,98	16,35	4,79	15,13	3,02
21	16,57	4,10	12,94	1,32	11,91	2,36	14,09	0,96
22	18,68	2,56	13,92	2,23	12,20	1,35	18,00	2,10
23	18,24	3,41	18,19	2,10	14,94	2,50	17,13	1,69
24	15,44	2,85	14,64	1,73	15,40	2,46	16,59	1,09
25	19,52	4,93	15,84	2,45	13,02	4,98	17,09	1,83
26	17,31	3,24	15,54	1,96	13,40	2,02	14,55	1,96
27	14,47	2,82	14,63	1,90	16,69	1,38	18,41	3,67
28	15,67	1,99	12,18	1,66	16,31	2,46	16,65	1,73
29	15,73	2,75	13,90	2,19	10,66	3,64	13,23	1,54
30	18,65	1,82	13,93	2,73	9,41	2,19	12,72	1,88
31	13,03	1,32	18,45	1,16	19,38	2,03	19,63	2,57
32	12,34	1,66	15,11	1,03	14,63	3,41	20,75	3,45

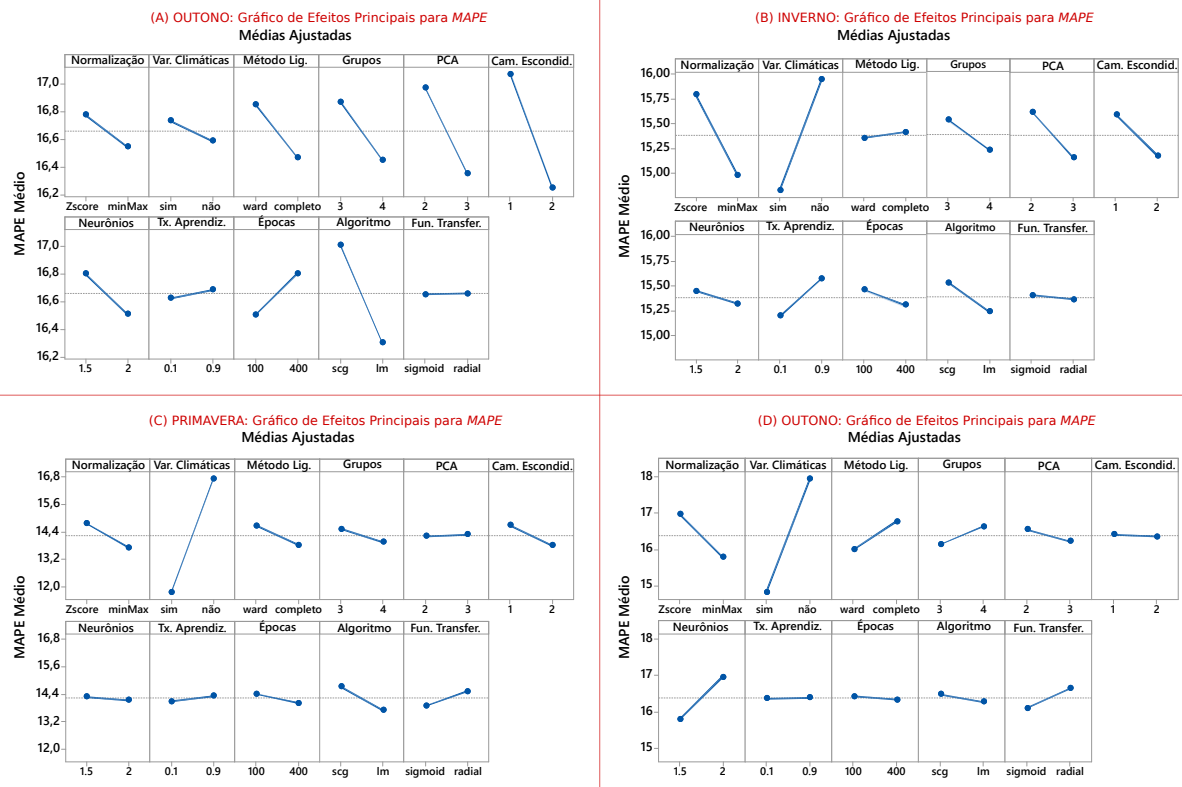


Figura 4.10 – Gráfico de efeitos principais para cada estação do ano, considerando o erro médio de todas as dezessete plantas de geração.

transferência; o algoritmo de treinamento foi '*Levenberg-Marquardt*'.

Da mesma forma, a Primavera, representada pela Figura 4.10(C), promove maior ênfase no uso de variáveis climáticas no processo de previsão, tendo boa representação do erro. Em geral, parâmetros como método de normalização apontam para 'minMax', assim como os anteriores; camadas ocultas 2; e algoritmo de treinamento '*Levenberg-Marquardt*'. Os demais parâmetros exercem pouca influência no erro de previsão, considerando esta estação do ano.

Por último, mas não menos importante, o verão, representado pela Figura 4.10(D), também destaca o uso de variáveis climáticas no processo de previsão, contribuindo para a redução do erro. Assim como nas demais estações do ano, o método de normalização foi mantido como 'minMax' conforme indicado para reduzir o erro. O método de ligação para formação dos grupos de dias semelhantes foi '*Ward*'; O número de componentes principais 3; o número de neurônios por camada 1,5 (de acordo com a equação de multiplicação explicada na subseção 3.1.2); e função de transferência '*Simétrica Sigmóide*'. Os demais parâmetros não mencionados têm pouca influência no erro quando variados.

Como esperado nesta investigação, a maioria dos resultados indica que o uso de variáveis climáticas, com pelo menos 3 componentes principais na redução da dimensionalidade, contribui para a redução do erro de previsão.

4.2.3 Previsão Combinada - *Ensemble*

A motivação da combinação é produzir resultados mais precisos do que os melhores componentes de previsão, considerados individualmente. Assim, a combinação dos resultados de previsão utilizou a ferramenta estatística *Mixture DOE* para encontrar os pesos ideais para que o *ensemble* pudesse ser formado.

Antes, é preciso definir quantos fatores (ingredientes) participarão dessa mistura. Para isso, os resultados previstos de cada corrida experimental, inicialmente processados, são classificados de acordo com o *MAPE*. Oito grupos foram escolhidos por meio da análise de *cluster*, que utiliza o método de ligação de *Ward* com distância euclidiana. A partir desses oito grupos de previsão, foi realizada a análise de variância unidirecional utilizando o procedimento de comparação de *Tukey*, de forma que apenas o) grupos com os menores *MAPEs*, estatisticamente diferentes dos demais, foram escolhidos.

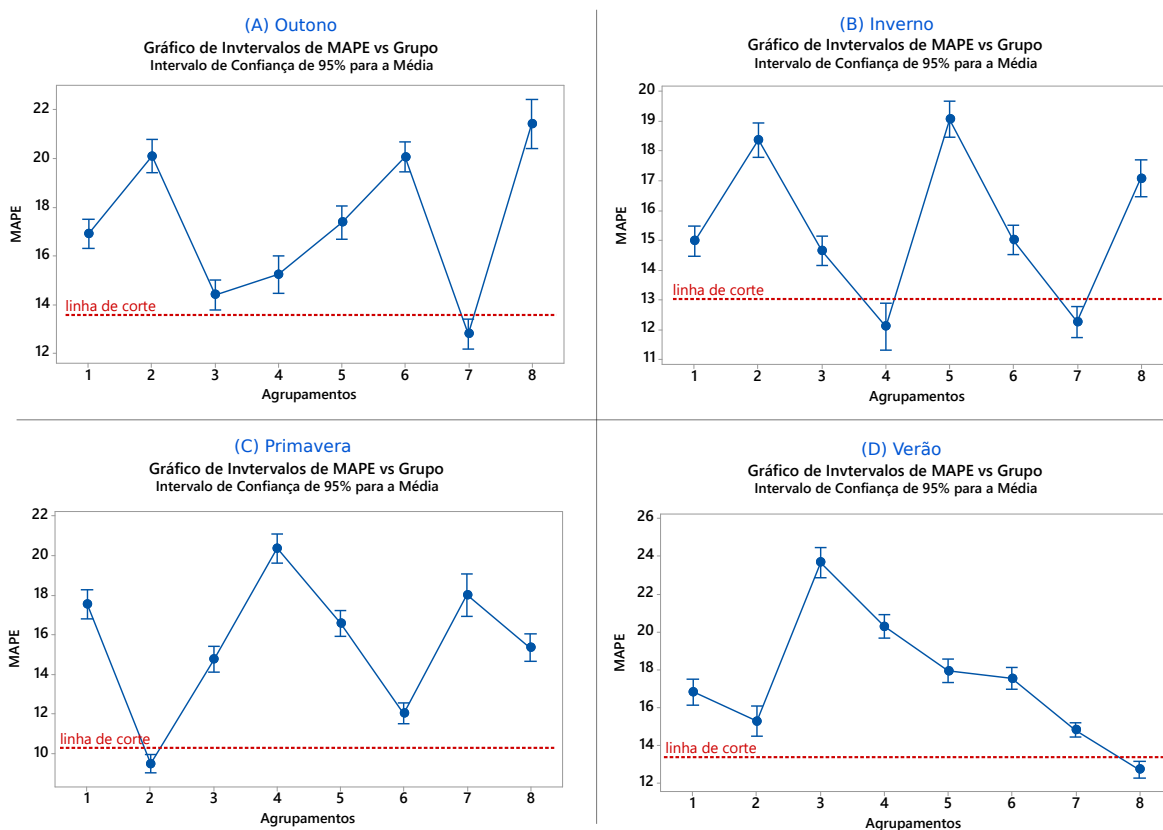


Figura 4.11 – Análise variância usando o procedimento de comparação de *Tukey* para cada estação do ano.

A Figura 4.11 apresenta o gráfico de intervalos que relaciona os *MAPEs* pelos grupos. Foi escolhido o grupo que difere estatisticamente dos demais e que possui o *MAPE* mais baixo. No caso da previsão referente ao outono, apresentada na Figura 4.11(A), foram escolhidos os resultados pertencentes ao grupo '7', que são três elementos. A estação do inverno, indicada pela Figura 4.11(B), revelou dois grupos estatisticamente iguais com *MAPE* inferior, '4' e '7', sendo que o total desses dois grupos possui quatro

Tabela 4.8 – Arranjo *DOE* de misturas considerando três elementos a serem combinados. Os resultados do outono e inverno estão listados.

Definição dos Pesos			OUTONO		VERÃO	
w_1	w_2	w_3	Média	Desv.Pad.	Média	Desv.Pad.
1,000	0,000	0,000	11,81	14,69	11,56	6,57
0,000	1,000	0,000	10,68	7,02	10,85	8,52
0,000	0,000	1,000	10,26	8,57	10,99	7,15
0,500	0,500	0,000	9,68	7,96	10,79	6,30
0,500	0,000	0,500	10,18	8,42	11,08	5,67
0,000	0,500	0,500	10,15	7,10	10,45	7,34
0,333	0,333	0,333	9,29	7,23	10,73	5,96
0,667	0,167	0,167	10,18	10,09	11,14	5,59
0,167	0,667	0,167	9,50	6,67	10,62	7,33
0,167	0,167	0,667	9,46	7,51	10,75	6,36

elementos. A primavera identificou o grupo numerado '2' na Figura 4.11(C), que possui cinco elementos favoráveis à combinação, e o verão classificou o grupo numerado '8' na Figura 4.11(D) como o grupo de menor *MAPE*, possuindo três elementos. Assim, a formação da Mistura *DOE* é função do número de elementos (fatores ou ingredientes) a serem combinados. Neste caso, cada fator representa o peso que otimiza a combinação e visa reduzir o *MAPE* total.

A Tabela 4.8 resume as combinações de peso para três elementos. Aqui são apresentados os *MAPEs* médios para as estações de outono e verão. Especificamente para o outono, os pesos que melhor se ajustaram às previsões foram (0,333; 0,333; 0,333) com *MAPE* médio de 9,29% e desvio padrão de 7,23. Para o verão, a combinação de pesos que melhor se ajustou às previsões foi (0,00; 0,50; 0,50), com *MAPE* média de 10,45% e desvio padrão de 7,34.

A Tabela 4.9 lista as combinações de pesos para a previsão do dia cuja estação é inverno. Esta tabela apresenta quatro elementos que participam dessa previsão combinada. Os pesos que tornam essa mistura ideal são dados por (0,00; 0,00; 0,50; 0,50) com um *MAPE* médio de 9,11% e desvio padrão de 5,55.

A Tabela 4.10 lista os pesos para a previsão do dia de primavera. Como agora há cinco elementos para combinar, a tabela cresce naturalmente em quantidade de linhas. A combinação ideal desses elementos é dada pelos pesos (0,00; 0,00; 0,333; 0,333; 0,333) com *MAPE* médio de 6,75% e desvio padrão de 6,47.

Quando um dos pesos é nulo, indica que aquele respectivo elemento não contribui para a formação do *ensemble* e pode ser descartado, pois a multiplicação por zero é zero. Uma representação geométrica de cada combinação pode ser visualizada usando o triângulo (para três elementos) e o tetraedro (para quatro elementos).

Cinco ou mais elementos são viáveis, mas a representação geométrica é mais difícil

Tabela 4.9 – Arranjo *DOE* de misturas considerando quatro elementos a serem combinados. Os resultados do inverno estão listados.

Definição dos pesos				INVERNO	
w_1	w_2	w_3	w_4	Média	Desv.Pad.
1,000	0,000	0,000	0,000	12,69	9,57
0,000	1,000	0,000	0,000	12,14	8,56
0,000	0,000	1,000	0,000	10,18	7,19
0,000	0,000	0,000	1,000	10,62	8,16
0,500	0,500	0,000	0,000	10,13	7,16
0,500	0,000	0,500	0,000	10,27	7,85
0,500	0,000	0,000	0,500	10,24	6,11
0,000	0,500	0,500	0,000	10,20	6,89
0,000	0,500	0,000	0,500	9,77	5,79
0,000	0,000	0,500	0,500	9,11	5,55
0,333	0,333	0,333	0,000	9,92	6,43
0,333	0,333	0,000	0,333	9,55	5,01
0,333	0,000	0,333	0,333	9,48	5,73
0,000	0,333	0,333	0,333	9,60	4,59
0,250	0,250	0,250	0,250	9,49	4,78
0,625	0,125	0,125	0,125	10,43	6,92
0,125	0,625	0,125	0,125	10,29	5,95
0,125	0,125	0,625	0,125	9,82	5,28
0,125	0,125	0,125	0,625	9,11	6,01

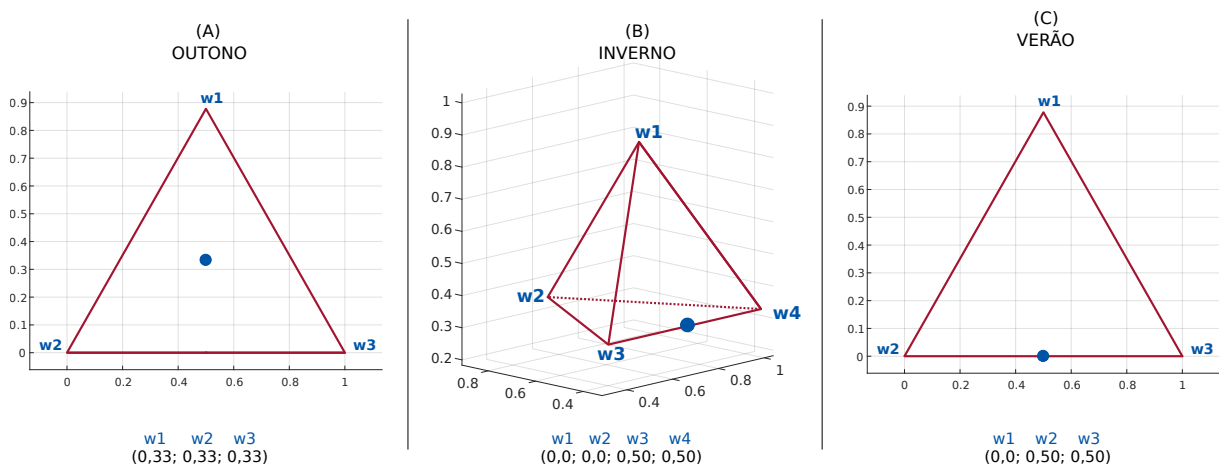


Figura 4.12 – Combinação de pesos ajustada que configura os menores erros de previsão para as dezessete usinas de geração, separadas por estação do ano: (A) Outono - (0,33; 0,33; 0,33); (B) Inverno - (0,0; 0,0; 0,5; 0,5); (C) Verão - (0,0; 0,5; 0,5) e Primavera, que não é plotado porque os elementos de combinação continham 5 fatores (difícil de ver): (0,0; 0,0; 0,33; 0,33; 0,33).

Tabela 4.10 – Arranjo *DOE* de mistura considerando quatro elementos a serem combinados. Os resultados da primavera estão listados.

Definição dos Pesos					PRIMAVERA	
$w1$	$w2$	$w3$	$w4$	$w5$	Média	Desv.Pad.
1,000	0,000	0,000	0,000	0,000	9,09	5,97
0,000	1,000	0,000	0,000	0,000	9,96	6,21
0,000	0,000	1,000	0,000	0,000	7,79	7,95
0,000	0,000	0,000	1,000	0,000	7,47	6,41
0,000	0,000	0,000	0,000	1,000	9,11	6,19
0,500	0,500	0,000	0,000	0,000	9,08	6,01
0,500	0,000	0,500	0,000	0,000	8,02	6,36
0,500	0,000	0,000	0,500	0,000	7,18	6,31
0,500	0,000	0,000	0,000	0,500	8,40	6,03
0,000	0,500	0,500	0,000	0,000	7,70	6,20
0,000	0,500	0,000	0,500	0,000	8,33	5,76
0,000	0,500	0,000	0,000	0,500	9,52	5,44
0,000	0,000	0,500	0,500	0,000	6,96	6,76
0,000	0,000	0,500	0,000	0,500	6,86	6,53
0,000	0,000	0,000	0,500	0,500	7,91	5,94
0,333	0,333	0,333	0,000	0,000	8,05	5,97
0,333	0,333	0,000	0,333	0,000	7,94	6,06
0,333	0,333	0,000	0,000	0,333	8,76	5,92
0,333	0,000	0,333	0,333	0,000	6,89	6,64
0,333	0,000	0,333	0,000	0,333	7,51	6,10
0,333	0,000	0,000	0,333	0,333	7,40	6,30
0,000	0,333	0,333	0,333	0,000	7,30	6,15
0,000	0,333	0,333	0,000	0,333	7,75	5,85
0,000	0,333	0,000	0,333	0,333	8,59	5,40
0,000	0,000	0,333	0,333	0,333	6,75	6,47
0,250	0,250	0,250	0,250	0,000	7,35	6,23
0,250	0,250	0,250	0,000	0,250	7,97	5,77
0,250	0,250	0,000	0,250	0,250	8,03	5,97
0,250	0,000	0,250	0,250	0,250	7,00	6,31
0,000	0,250	0,250	0,250	0,250	7,47	5,91
0,200	0,200	0,200	0,200	0,200	7,45	6,06
0,600	0,100	0,100	0,100	0,100	8,11	5,95
0,100	0,600	0,100	0,100	0,100	8,64	5,82
0,100	0,100	0,600	0,100	0,100	7,02	6,78
0,100	0,100	0,100	0,600	0,100	7,09	6,29
0,100	0,100	0,100	0,100	0,600	8,11	5,89

de ver. Nessa perspectiva, a Figura 4.12 apresenta a representação do ponto ideal de uma combinação de previsões para cada estação do ano, exceto a primavera (que possui cinco pesos). Outono e verão aparecem nas Figuras 4.12(A) e 4.12(C), respectivamente, através do triângulo e inverno em 4.12(B), através do tetraedro.

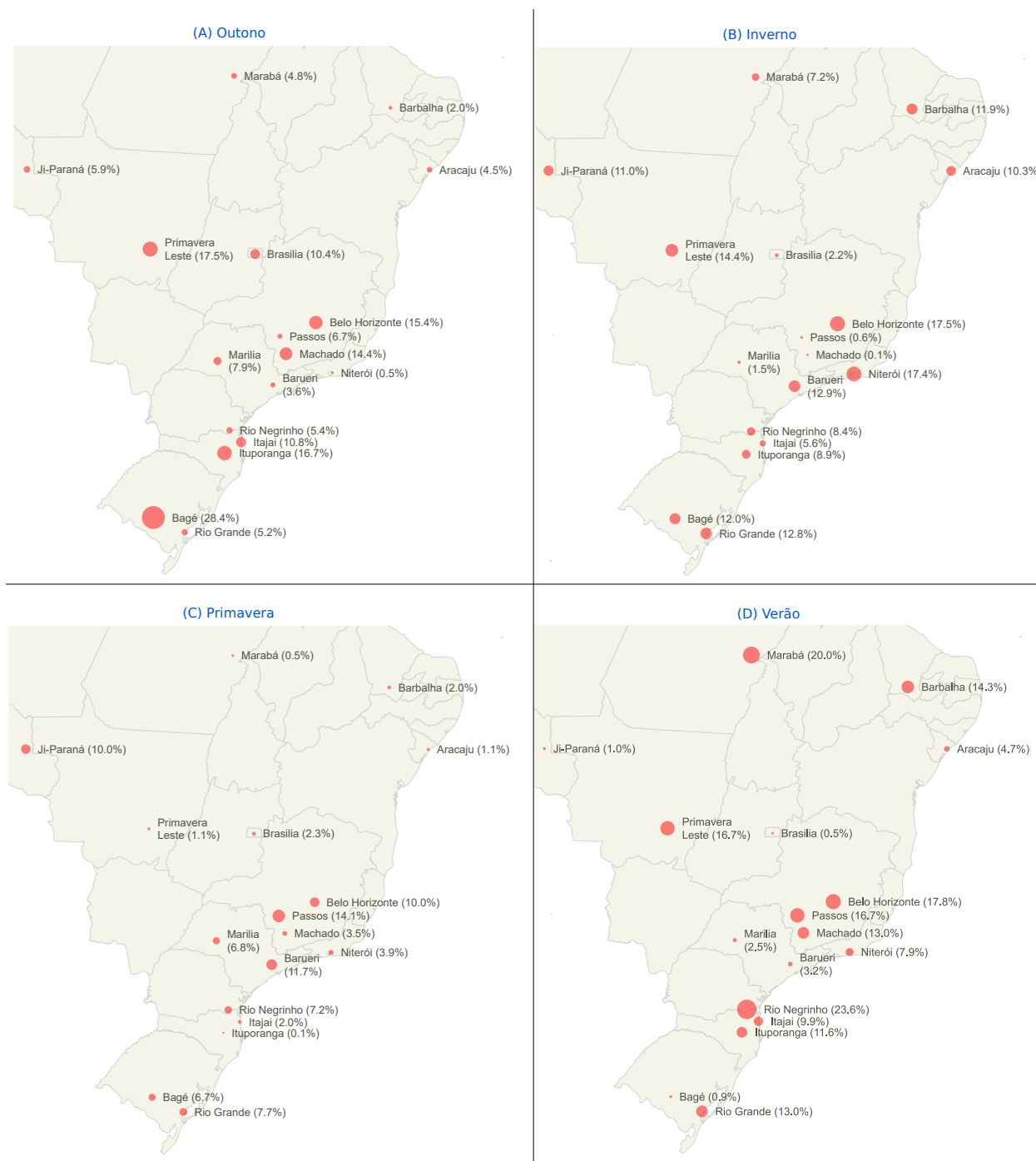


Figura 4.13 – Mapa de intensidade dos erros por estação do ano: os resultados apresentados são os *MAPEs* mínimos encontrados para cada usina de geração, após o processo de combinação dos resultados. Os mapas foram gerados utilizando a linguagem *JavaScript*, através da biblioteca open source *jQuery MAPAEL* (BROUTÉ, 2013).

O mapa da Figura 4.13 mostra a previsão final combinada para cada usina de

geração fotovoltaica e seus respectivos *MAPEs*. Os maiores valores numericamente observados dos *MAPEs* foram 28,4% (Outono-Bagé) e 23,6% (Verão-Rio Negrinho). Ao contrário, os valores numericamente menores para cada estação foram 0,1% (Inverno-Machado e Primavera-Ituporanga), 0,5% (Outono-Niterói e Primavera-Niterói), 0,6% (Inverno-Passos) e 0,9% (Verão-Bagé). A estação com menor média geral foi a primavera, possivelmente por estar menos sujeita a fatores incontroláveis, como movimento de nuvens, poeira, etc.

A próxima Seção discute as conclusões desta pesquisa.

5 CONCLUSÕES

5.1 Conclusões gerais

Em termos práticos, não existe na comunidade científica um modelo de previsão que tenha bom desempenho em todas as situações e que seja adaptável aos mais diversos cenários. Embora a literatura revele uma variação nas técnicas de previsão e não haja consenso sobre um método genérico que atenda a todos os casos, a área de previsão é um alvo constante de estudos.

Assim, diante dessa perspectiva, este trabalho apresentou uma metodologia de previsão de geração fotovoltaica usando redes neurais artificiais parametrizadas pelo método de planejamento de experimentos, cuja versatilidade permite ao analisador identificar os parâmetros que mais interferem nos resultados.

A originalidade da metodologia proposta reduz o número de execuções experimentais simuladas através do *DOE* fatorial fracionado e permite ao analista inferir sobre as decisões a serem tomadas com um alto nível de confiança. A versatilidade do método proposto permite alterar o número de fatores a serem usados no arranjo experimental, no modelo de previsão e no horizonte de previsão desejado.

A inserção da geração fotovoltaica nas redes de energia se intensificou e a acurácia da previsão promove a confiabilidade do sistema, além de permitir a eficiência do gerenciamento de energia. A técnica preditiva escolhida, a RNA, é amplamente utilizada na literatura para esse fim. A RNA teve sua configuração paramétrica sistematicamente definida com base no *DOE*, com o desenho experimental 2^{11-6} . A seleção dos resultados obtidos pelas redes com os menores *MAPEs* é agrupada hierarquicamente através da análise de *cluster*. Este agrupamento se mostrou vantajoso pelo fato de providenciar as entradas para uma próxima etapa, que é realizada através do *DOE* de misturas. Encontrados os pesos ideais, o *ensemble* é formado para constituir a previsão combinada.

Destaca-se ainda a redução da dimensionalidade dos dados meteorológicos neste processo. Sabe-se que, à medida que novas usinas de geração fotovoltaica são inseridas no sistema elétrico e à medida que o tempo evolui, a quantidade de dados armazenados para análise aumenta. Mantendo os níveis de explicação das variáveis, a redução do conjunto de dados é promissora e a técnica de Análise de Componentes Principais (*PCA*) foi utilizada para explicar a variância de dezesseis variáveis climáticas, reduzindo-as a duas ou três variáveis, com grau satisfatório de explicação média em torno de 75% - 85%.

Além da redução de dados, o esforço computacional para previsão também deve

ser considerado. Como os parâmetros de predição não são explorados de forma aleatória ou em sua totalidade, a ferramenta que combina técnicas matemáticas e estatísticas para o desenvolvimento de arranjos experimentais eficientes, equilibrados e econômicos, Design of Experiments (*DOE*), permite reduzir a busca paramétrica e o custo desta execução diminui.

Desta forma, o analisador pode inferir com alto nível de confiança, mantendo a confiabilidade estatística. Naturalmente, como em qualquer outra metodologia baseada em heurística para previsão de geração fotovoltaica, não há garantia de obtenção da solução ótima.

Foram realizados dois estudos de caso usando a metodologia aqui apresentada. A partir do primeiro estudo de caso, realizado para uma planta de geração fotovoltaica no sul de Minas Gerais, foi possível perceber que a metodologia proposta reduziu o erro percentual absoluto médio de uma semana à frente. A fim de validar os resultados alcançados, foi construída a elipse de confiança dos dados com o intuito de verificar se as previsões estavam dentro do padrão esperado para o conjunto.

O segundo estudo de caso, que realizou a previsão de um dia à frente, contou com a organização sazonal dos dados, partindo da premissa de que dias com características semelhantes tendem a produzir resultados mais assertivos. Dessa forma, a análise de *clusters* permitiu escolher os dias com características mais próximas ao dia de previsão desejado, reduzindo a quantidade de informações inseridas no modelo de previsão e evitando o superaprendizado. Por fim, verificou-se experimentalmente que a previsão combinada produziu melhores resultados quando comparada aos melhores previsores, considerados individualmente.

Este estudo de caso abrangeu 17 usinas de geração localizadas em diferentes regiões do território brasileiro, e a avaliação paramétrica considerou todas essas usinas juntas. Trabalhos futuros podem avaliar diferentes horizontes de previsão, diferentes fatores e diferentes modelos de previsão.

REFERÊNCIAS

- ABBASIMEHR, H.; SHABANI, M.; YOUSEFI, M. An optimized model using LSTM network for demand forecasting. *Computers Industrial Engineering*, v. 143, p. 106435, 2020. ISSN 0360-8352. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0360835220301698>>. Citado na página 17.
- ACKERMANN, T.; CHEREVATSKIY, S.; BROWN, T.; ERIKSSON, R.; SAMADI, A.; GHANDHARI, M.; SODER, L.; LINDENBERGER, D.; JAGEMANN, C.; HAGSPIEL, S.; CUK, V.; RIBEIRO, P. F.; COBBEN, S.; BINDNER, H.; ISLEIFSSON, F. R.; MIHET-POPA, L. Smart modeling of optimal integration of high penetration of pv-smooth pv. *Final Report for Smooth PV Project under PV ERA NET Call*, 2013. Citado na página 3.
- AGOUA, X. G.; GIRARD, R.; KARINIOTAKIS, G. Short-term spatio-temporal forecasting of photovoltaic power production. *IEEE Transactions on Sustainable Energy*, v. 9, n. 2, p. 538–546, 2018. ISSN 1949-3029. Citado na página 1.
- AHMADI, M.; SAMET, H.; GHANBARI, T. A New Method for Detecting Series Arc Fault in Photovoltaic Systems Based on the Blind-Source Separation. *IEEE Transactions on Industrial Electronics*, v. 67, n. 6, p. 5041–5049, 2020. Citado na página 21.
- AHMED, R.; SREERAM, V.; MISHRA, Y.; ARIF, M. A review and evaluation of the state-of-the-art in pv solar power forecasting: Techniques and optimization. *Renewable and Sustainable Energy Reviews*, v. 124, p. 109792, 2020. ISSN 1364-0321. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1364032120300885>>. Citado na página 14.
- AKHTER, M. N.; MEKHILEF, S.; MOKHLIS, H.; Mohamed Shah, N. Review on forecasting of photovoltaic power generation based on machine learning and metaheuristic techniques. *IET Renewable Power Generation*, v. 13, n. 7, p. 1009–1023, 2019. Citado na página 18.
- AKSU, G.; GÜZELLER, C.; ESER, T. The Effect of the Normalization Method Used in Different Sample Sizes on the Success of Artificial Neural Network Model. *International Journal of Assessment Tools in Education*, v. 6, p. 170–192, 2019. Citado na página 15.
- ALARAJ, M.; KUMAR, A.; ALSAIDAN, I.; RIZWAN, M.; JAMIL, M. Energy Production Forecasting From Solar Photovoltaic Plants Based on Meteorological Parameters for Qassim Region, Saudi Arabia. *IEEE Access*, v. 9, p. 83241–83251, 2021. Citado na página 36.
- AMARAL, R. P. F.; RIBEIRO, V. M.; AGUIAR, E. P. de. Type-1 and singleton fuzzy logic system trained by a fast scaled conjugate gradient methods for dealing with binary classification problems. *Neurocomputing*, v. 355, p. 57–70, 2019. ISSN 0925-2312. Citado na página 29.
- AMROUCHE, B.; PIVERT, L. X. Artificial neural network based daily local forecasting for global solar radiation. *Applied Energy*, v. 130, p. 333–341, 2014. ISSN 0306-2619. Citado 2 vezes nas páginas 12 e 26.

- ANTONANZAS, J.; OSORIO, N.; ESCOBAR, R.; URRACA, R.; PISON, F. J. Martinez-de; ANTONANZAS-TORRES, F. Review of photovoltaic power forecasting. *Solar Energy*, v. 136, p. 78–111, 2016. ISSN 0038-092X. Citado na página 14.
- ANTONY, J. 6 - Full Factorial Designs. In: ANTONY, J. (Ed.). *Design of Experiments for Engineers and Scientists (Second Edition)*. Second edi. Oxford: Elsevier, 2014. p. 63–85. ISBN 978-0-08-099417-8. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780080994178000067>>. Citado na página 19.
- APPIAH, A. Y.; ZHANG, X.; AYAWLI, B. B. K.; KYEREMEH, F. Long short-term memory networks based automatic feature extraction for photovoltaic array fault diagnosis. *IEEE Access*, v. 7, p. 30089–30101, 2019. ISSN 2169-3536. Citado na página 26.
- BALESTRASSI, P. P.; POPOVA, E.; PAIVA, A. P.; LIMA, J. W. M. Design of experiments on neural network's training for nonlinear time series forecasting. *Neurocomputing*, v. 72, n. 4, p. 1160–1178, 2009. ISSN 0925-2312. Brain Inspired Cognitive Systems (BICS 2006) / Interplay Between Natural and Artificial Computation (IWINAC 2007). Citado 5 vezes nas páginas 17, 18, 28, 29 e 39.
- BARBIERI, F.; RAJAKARUNA, S.; GHOSH, A. Very short-term photovoltaic power forecasting with cloud modeling: A review. *Renewable and Sustainable Energy Reviews*, v. 75, p. 242–263, 2017. ISSN 1364-0321. Citado 2 vezes nas páginas 14 e 18.
- BRACALE, A.; CARPINELLI, G.; FALCO, P. D. A probabilistic competitive ensemble method for short-term photovoltaic power forecasting. *IEEE Transactions on Sustainable Energy*, v. 8, n. 2, p. 551–560, 2017. ISSN 1949-3029. Citado 2 vezes nas páginas 1 e 3.
- BROUTÉ, V. *jQuery Mapael - Dynamic vector maps*. 2013. (Acessado: Dezembro 2021). Disponível em: <<https://github.com/neveldo/jquery-Mapael>>. Citado 2 vezes nas páginas ix e 62.
- BUGALA, A.; ZABOROWICZ, M.; BONIECKI, P.; JANCZAK, D.; KOSZELA, K.; CZEKALA, W.; LEWICKI, A. Short-term forecast of generation of electric energy in photovoltaic systems. *Renewable and Sustainable Energy Reviews*, v. 81, p. 306–312, 2018. ISSN 1364-0321. Citado 3 vezes nas páginas 9, 12 e 27.
- CERVONE, G.; CLEMENTE-HARDING, L.; ALESSANDRINI, S.; MONACHE, L. D. Short-term photovoltaic power forecasting using artificial neural networks and an analog ensemble. *Renewable Energy*, v. 108, p. 274–286, 2017. ISSN 0960-1481. Citado 5 vezes nas páginas 9, 11, 14, 16 e 17.
- CHAI, S.; XU, Z.; JIA, Y.; WONG, W. K. A Robust Spatiotemporal Forecasting Framework for Photovoltaic Generation. *IEEE Transactions on Smart Grid*, v. 11, n. 6, p. 5370–5382, 2020. Citado na página 3.
- CHU, X.; LUO, Y.; WANG, J.; GUO, W.; WU, F.; WANG, W.; LI, W. Fast Multiple Edge Response Method Based on the Design of Experiment and Machine Learning. *IEEE Microwave and Wireless Components Letters*, v. 31, n. 5, p. 521–524, 2021. Citado na página 19.

- CORNELL, J. A. *Experiments with Mixtures: Designs, Models, and the Analysis of Mixture Data*. [S.l.]: Wiley, 2011. (Wiley Series in Probability and Statistics). ISBN 9781118150498. Citado na página 23.
- CUI, J.; LIU, S.; YANG, J.; GE, W.; ZHOU, X.; WANG, A. A Load Combination Prediction Algorithm Considering Flexible Charge and Discharge of Electric Vehicles. In: *2019 IEEE 10th International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*. [S.l.: s.n.], 2019. p. 711–716. Citado na página 3.
- CUK, V.; RIBEIRO, P.; COBBEN, J. F. G.; KLING, W. L.; ISLEIFSSON, F. R.; BINDNER, H. W.; MARTENSEN, N.; SAMADI, A.; SDER, L. Considerations on the modeling of photovoltaic systems for grid impact studies. *Mathematics of Computation - Math. Comput.*, 2011. Citado na página 1.
- DALL'ANESE, E.; DHOPE, V. S.; JOHNSON, B. B.; GIANNAKIS, G. B. Optimal dispatch of residential photovoltaic inverters under forecasting uncertainties. *IEEE Journal of Photovoltaics*, v. 5, n. 1, p. 350–359, 1 2015. ISSN 2156-3381. Citado na página 2.
- DANESHI, H. Overview of Renewable Energy Portfolio in CAISO - Operational and Market Challenges. In: *2018 IEEE Power Energy Society General Meeting (PESGM)*. [S.l.: s.n.], 2018. p. 1–5. Citado na página 1.
- DAS, U. K.; TEY, K. S.; SEYEDMAHMOUDIAN, M.; MEKHILEF, S.; IDRIS, M. Y. I.; DEVENTER, V. W.; HORAN, B.; STOJCEVSKI, A. Forecasting of photovoltaic power generation and model optimization: A review. *Renewable and Sustainable Energy Reviews*, v. 81, p. 912–928, 2018. ISSN 1364-0321. Citado 5 vezes nas páginas 2, 3, 13, 14 e 27.
- De Almeida, F. A.; MELLO, L. G. D.; ROMÃO, E. L.; GOMES, G. F.; GOMES, J. H. D. F.; De Paiva, A. P.; FILHO, J. M.; BALESTRASSI, P. P. A PCA-Based Consistency and Sensitivity Approach for Assessing Linkage Methods in Voltage Sag Studies. *IEEE Access*, v. 9, p. 84871–84885, 2021. Citado 2 vezes nas páginas 15 e 16.
- De Luca, G.; ZUCCOLOTTO, P. Hierarchical time series clustering on tail dependence with linkage based on a multivariate copula approach. *International Journal of Approximate Reasoning*, v. 139, p. 88–103, 2021. ISSN 0888-613X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0888613X21001390>>. Citado na página 16.
- DEO, R. C.; ŞAHIN, M. Forecasting long-term global solar radiation with an ANN algorithm coupled with satellite-derived (MODIS) land surface temperature (LST) for regional locations in Queensland. *Renewable and Sustainable Energy Reviews*, v. 72, p. 828–848, 2017. ISSN 1364-0321. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1364032117301247>>. Citado 2 vezes nas páginas 9 e 11.
- DHIMISH, M.; HOLMES, V.; MEHRDADI, B.; DALES, M. Comparing Mamdani Sugeno fuzzy logic and RBF ANN network for PV fault detection. *Renewable Energy*, v. 117, p. 257–274, 2018. ISSN 0960-1481. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0960148117310273>>. Citado na página 17.

- DONALDSON, D. L.; PIPER, D. M.; JAYAWEERA, D. Temporal Solar Photovoltaic Generation Capacity Reduction From Wildfire Smoke. *IEEE Access*, v. 9, p. 79841–79852, 2021. Citado na página 35.
- DU, E.; ZHANG, N.; HODGE, B.-M.; WANG, Q.; LU, Z.; KANG, C.; KROPOSKI, B.; XIA, Q. Operation of a High Renewable Penetrated Power System With CSP Plants: A Look-Ahead Stochastic Unit Commitment Model. *IEEE Transactions on Power Systems*, v. 34, n. 1, p. 140–151, 2019. Citado na página 1.
- DU, L.; ZHANG, L.; TIAN, X. Deep power forecasting model for building attached photovoltaic system. *IEEE Access*, v. 6, p. 52639–52651, 2018. ISSN 2169-3536. Citado na página 27.
- Fontes Rodrigues, B. K.; GOMES, M.; Oliveira Santana, Â. M.; BARBOSA, D.; MARTINEZ, L. Modelling and forecasting for solar irradiance from solarimetric station. *IEEE Latin America Transactions*, v. 20, n. 2, p. 250–258, 2022. Citado na página 2.
- GALICIA, A.; TALAVERA-LLAMES, R.; TRONCOSO, A.; KOPRINSKA, I.; MARTINEZ-ALVAREZ, F. Multi-step forecasting for big data time series based on ensemble learning. *Knowledge-Based Systems*, v. 163, p. 830–841, 2019. ISSN 0950-7051. Citado na página 4.
- GE, L.; XIAN, Y.; YAN, J.; WANG, B.; WANG, Z. A Hybrid Model for Short-term PV Output Forecasting Based on PCA-GWO-GRNN. *Journal of Modern Power Systems and Clean Energy*, v. 8, n. 6, p. 1268–1275, 2020. Citado na página 22.
- GENSLER, A.; HENZE, J.; SICK, B.; RAABE, N. Deep Learning for solar power forecasting — An approach using AutoEncoder and LSTM Neural Networks. In: *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.: s.n.], 2016. p. 2858–2865. Citado na página 3.
- GÓMEZ-QUILES, C.; ASECIO-CORTÉS, G.; GASTALVER-RUBIO, A.; MARTÍNEZ-ÁLVAREZ, F.; TRONCOSO, A.; MANRESA, J.; RIQUELME, J. C.; RIQUELME-SANTOS, J. M. A Novel Ensemble Method for Electric Vehicle Power Consumption Forecasting: Application to the Spanish System. *IEEE Access*, v. 7, p. 120840–120856, 2019. Citado na página 3.
- HAGAN, M. T.; MENHAJ, M. B. Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, v. 5, n. 6, p. 989–993, 11 1994. ISSN 1045-9227. Citado na página 29.
- HANSEN, L. K.; SALAMON, P. Neural network ensembles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, v. 12, p. 993–1001, 1990. Citado na página 18.
- HOSSAIN, M.; MEKHILEF, S.; DANESH, M.; OLATOMIWA, L.; SHAMSHIRBAND, S. Application of extreme learning machine for short term output power forecasting of three grid-connected PV systems. *Journal of Cleaner Production*, v. 167, p. 395–405, 2017. ISSN 0959-6526. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0959652617317973>>. Citado 2 vezes nas páginas 3 e 13.

- HUANG, C.; CHEN, S.; YANG, S.; KUO, C. One-day-ahead hourly forecasting for photovoltaic power generation using an intelligent method with weather-based forecasting models. *IET Generation, Transmission Distribution*, v. 9, n. 14, p. 1874–1882, 2015. ISSN 1751-8687. Citado na página 2.
- HUANG, X.; LI, Q.; TAI, Y.; CHEN, Z.; LIU, J.; SHI, J.; LIU, W. Time series forecasting for hourly photovoltaic power using conditional generative adversarial network and Bi-LSTM. *Energy*, v. 246, p. 123403, 2022. ISSN 0360-5442. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0360544222003061>>. Citado na página 2.
- IFSULDEMINAS. *IF Solar - Implantação do projeto de geração fotovoltaica nos Campi do Ifsuldeminas*. 2017. Disponível em: <<https://portal.ifsuldeminas.edu.br/index.php/ultimas-noticias-ifsuldeminas/78-noticias-da-prodi/1327-projeto-ifsolar>>. Citado na página 41.
- INMET. *INMET Base de Dados Meteorológicos*. 2022. (Acessado: Agosto 2021). Disponível em: <<https://bdmep.inmet.gov.br/>>. Citado 2 vezes nas páginas 52 e 53.
- JANKOVIC, A.; CHAUDHARY, G.; GOIA, F. Designing the design of experiments (DOE) – An investigation on the influence of different factorial designs on the characterization of complex systems. *Energy and Buildings*, v. 250, p. 111298, 2021. ISSN 0378-7788. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S037877882100582X>>. Citado 2 vezes nas páginas 19 e 20.
- JOHNSON, R. A.; WICHERN, D. W. *Applied Multivariate Statistical Analysis*. [S.l.]: Pearson Prentice Hall, 2007. (Applied Multivariate Statistical Analysis). ISBN 9780131877153. Citado na página 21.
- Kara Mostefa Khelil, C.; AMROUCHE, B.; KARA, K.; CHOUDER, A. The impact of the ANN's choice on PV systems diagnosis quality. *Energy Conversion and Management*, v. 240, p. 114278, 2021. ISSN 0196-8904. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0196890421004544>>. Citado na página 17.
- KHAN, K.; SHUKLA, S.; SINGH, B. Improved Performance Design Realization of a Fractional Kilowatt Induction Motor With Predictive Current Control for Water Pumping. *IEEE Transactions on Industry Applications*, v. 56, n. 4, p. 4575–4587, 2020. Citado na página 19.
- KHEIRKHAH, A.; AZADEH, A.; SABERI, M.; AZARON, A.; SHAKOURI, H. Improved estimation of electricity demand function by using of artificial neural network, principal component analysis and data envelopment analysis. *Computers Industrial Engineering*, v. 64, n. 1, p. 425–441, 2013. ISSN 0360-8352. Citado na página 29.
- KIM, J.; OH, H.; CHOI, J. K. Learning based cost optimal energy management model for campus microgrid systems. *Applied Energy*, v. 311, p. 118630, 2022. ISSN 0306-2619. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0306261922001015>>. Citado na página 17.
- KOTU, V.; DESHPANDE, B. Chapter 4 - Classification. In: KOTU, V.; DESHPANDE, B. (Ed.). *Data Science (Second Edition)*. Second edi. Morgan Kaufmann, 2019. p.

65–163. ISBN 978-0-12-814761-0. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780128147610000046>>. Citado na página 18.

KOURENTZES, N.; BARROW, D.; PETROPOULOS, F. Another look at forecast selection and combination: Evidence from forecast pooling. *International Journal of Production Economics*, v. 209, p. 226–235, 2019. ISSN 0925-5273. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0925527318302196>>. Citado na página 22.

KURKOVA, V. Kolmogorov's theorem and multilayer neural networks. *Neural Networks*, v. 5, n. 3, p. 501–506, 1992. ISSN 0893-6080. Citado na página 17.

LEE, W.; KIM, K.; PARK, J.; KIM, J.; KIM, Y. Forecasting solar power using long-short term memory and convolutional neural networks. *IEEE Access*, v. 6, p. 73068–73080, 2018. ISSN 2169-3536. Citado 2 vezes nas páginas 2 e 14.

LI, P.; ZHOU, K.; LU, X.; YANG, S. A hybrid deep learning model for short-term PV power forecasting. *Applied Energy*, v. 259, p. 114216, 2020. ISSN 0306-2619. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0306261919319038>>. Citado na página 2.

LIMA, F. J. L.; MARTINS, F. R.; PEREIRA, E. B.; LORENZ, E.; HEINEMANN, D. Forecast for surface solar irradiance at the brazilian northeastern region using nwp model and artificial neural networks. *Renewable Energy*, v. 87, p. 807–818, 2016. ISSN 0960-1481. Citado 4 vezes nas páginas 9, 11, 16 e 27.

LIN, J.; MA, J.; ZHU, J. A Privacy-Preserving Federated Learning Method for Probabilistic Community-Level Behind-the-Meter Solar Generation Disaggregation. *IEEE Transactions on Smart Grid*, v. 13, n. 1, p. 268–279, 2022. Citado na página 2.

LIN, Y.; LI, B.; MOISER, T. M.; GRIFFEL, L. M.; MAHALIK, M. R.; KWON, J.; ALAM, S. M. S. Revenue prediction for integrated renewable energy and energy storage system using machine learning techniques. *Journal of Energy Storage*, v. 50, p. 104123, 2022. ISSN 2352-152X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2352152X22001591>>. Citado na página 17.

LIRA, J. O. B.; RIELLA, H. G.; PADOIN, N.; SOARES, C. CFD + DoE optimization of a flat plate photocatalytic reactor applied to NO_x abatement. *Chemical Engineering and Processing - Process Intensification*, v. 154, p. 107998, 2020. ISSN 0255-2701. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0255270120304554>>. Citado na página 25.

LIU, J.; FANG, W.; ZHANG, X.; YANG, C. An improved photovoltaic power forecasting model with the assistance of aerosol index data. *IEEE Transactions on Sustainable Energy*, v. 6, n. 2, p. 434–442, 2015. ISSN 1949-3029. Citado 2 vezes nas páginas 14 e 27.

LORENZ, E.; HURKA, J.; HEINEMANN, D.; BEYER, H. G. Irradiance forecasting for the power prediction of grid-connected photovoltaic systems. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, v. 2, n. 1, p. 2–10, 2009. ISSN 1939-1404. Citado na página 2.

- LÓPEZ, M.; VALERO, S.; RODRIGUEZ, A.; VEIRAS, I.; SENABRE, C. New online load forecasting system for the spanish transport system operator. *Electric Power Systems Research*, v. 154, p. 401–412, 2018. ISSN 0378-7796. Citado na página 17.
- MAHFOUD, S.; DEROUICH, A.; OUANJLI, N. E. L.; MAHFOUD, M. E. L. Enhancement of the Direct Torque Control by using Artificial Neuron Network for a Doubly Fed Induction Motor. *Intelligent Systems with Applications*, v. 13, p. 200060, 2022. ISSN 2667-3053. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2667305322000011>>. Citado na página 18.
- MALVONI, M.; GIORGI, D. M. G.; CONGEDO, P. M. Photovoltaic forecast based on hybrid pca-lssvm using dimensionality reduced data. *Neurocomputing*, v. 211, p. 72–83, 2016. ISSN 0925-2312. SI: Recent Advances in SVM. Citado na página 26.
- MARQUARDT, D. W. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, Society for Industrial and Applied Mathematics, v. 11, n. 2, p. 431–441, 1963. ISSN 03684245. Citado na página 29.
- MASSAOUDI, M.; CHIH, I.; ABU-RUB, H.; REFAAT, S. S.; OUESLATI, F. S. Convergence of Photovoltaic Power Forecasting and Deep Learning: State-of-Art Review. *IEEE Access*, v. 9, p. 136593–136615, 2021. Citado 2 vezes nas páginas 15 e 19.
- MEER, D. van der; MOULI, G. R. C.; MOULI, G. M.-E.; ELIZONDO, L. R.; BAUER, P. Energy management system with pv power forecast to optimally charge evs at the workplace. *IEEE Transactions on Industrial Informatics*, v. 14, n. 1, p. 311–320, 1 2018. ISSN 1551-3203. Citado na página 14.
- MELLIT, A.; PAVAN, A. M. A 24-h forecast of solar irradiance using artificial neural network: Application for performance prediction of a grid-connected pv plant at trieste, italy. *Solar Energy*, v. 84, n. 5, p. 807–821, 2010. ISSN 0038-092X. Citado na página 27.
- MICHAELSON, D.; MAHMOOD, H.; JIANG, J. A predictive energy management system using pre-emptive load shedding for islanded photovoltaic microgrids. *IEEE Transactions on Industrial Electronics*, v. 64, n. 7, p. 5440–5448, 2017. ISSN 0278-0046. Citado na página 1.
- MISHRA, S.; TRIPATHY, L.; SATAPATHY, P.; DASH, P. K.; SAHANI, N. An Efficient Machine Learning Approach for Accurate Short Term Solar Power Prediction. In: *2020 International Conference on Computational Intelligence for Smart Power System and Sustainable Energy (CISPSSE)*. [S.l.: s.n.], 2020. p. 1–6. Citado na página 13.
- MITTAL, M.; BORA, B.; SAXENA, S.; GAUR, A. M. Performance prediction of pv module using electrical equivalent model and artificial neural network. *Solar Energy*, v. 176, p. 104–117, 2018. ISSN 0038-092X. Citado 2 vezes nas páginas 28 e 29.
- MOLLER, M. F. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, v. 6, n. 4, p. 525–533, 1993. Citado na página 29.
- MONTEIRO, R. V. A.; GUIMARÃES, G. C.; MOURA, F. A. M.; ALBERTINI, M. R. M. C.; ALBERTINI, M. K. Estimating photovoltaic power generation: Performance analysis of artificial neural networks, support vector machine and kalman filter. *Electric*

Power Systems Research, v. 143, p. 643–656, 2017. ISSN 0378-7796. Citado 3 vezes nas páginas 1, 2 e 17.

MONTGOMERY, D. C. *Design and analysis of experiments*. [S.l.]: John Wiley sons, 2017. Citado 5 vezes nas páginas viii, 4, 19, 23 e 24.

MOREIRA, M. O.; BALESTRASSI, P. P.; PAIVA, A. P.; RIBEIRO, P. F.; BONATTO, B. D. Design of experiments using artificial neural network ensemble for photovoltaic generation forecasting. *Renewable and Sustainable Energy Reviews*, v. 135, p. 110450, 2021. ISSN 1364-0321. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1364032120307371>>. Citado na página 39.

MUBIRU, J.; BANDA, E. Estimation of monthly average daily global solar irradiation using artificial neural networks. *Solar Energy*, v. 82, n. 2, p. 181–187, 2008. ISSN 0038-092X. Citado na página 17.

NIU, D.; YU, M.; SUN, L.; GAO, T.; WANG, K. Short-term multi-energy load forecasting for integrated energy systems based on CNN-BiGRU optimized by attention mechanism. *Applied Energy*, v. 313, p. 118801, 2022. ISSN 0306-2619. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0306261922002483>>. Citado na página 24.

OZOE GWU, C. G. Artificial neural network forecast of monthly mean daily global solar radiation of selected locations based on time series and month number. *Journal of Cleaner Production*, v. 216, p. 1–13, 2019. ISSN 0959-6526. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0959652619301076>>. Citado 2 vezes nas páginas 8 e 10.

PAN, C.; TAN, J. Day-Ahead Hourly Forecasting of Solar Generation Based on Cluster Analysis and Ensemble Model. *IEEE Access*, v. 7, p. 112921–112930, 2019. Citado 2 vezes nas páginas 8 e 10.

PONTES, F. J.; AMORIM, G. F.; BALESTRASSI, P. P.; PAIVA, A. P.; FERREIRA, J. R. Design of experiments and focused grid search for neural network parameter optimization. *Neurocomputing*, v. 186, p. 22–34, 2016. ISSN 0925-2312. Citado na página 19.

PVOUTPUT. *PVOutput. Open-access database*. 2022. (Acessado: Setembro 2021). Disponível em: <<https://www.pvoutput.org/>>. Citado na página 52.

QIAN, W.; ROLLING, C. A.; CHENG, G.; YANG, Y. Combining forecasts for universally optimal performance. *International Journal of Forecasting*, v. 38, n. 1, p. 193–208, 2022. ISSN 0169-2070. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0169207021000765>>. Citado na página 23.

QING, X.; NIU, Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy*, v. 148, p. 461–468, 2018. ISSN 0360-5442. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0360544218302056>>. Citado 2 vezes nas páginas 9 e 10.

RENCHER, A. C. *Methods of Multivariate Analysis*. Wiley, 2003. (Wiley Series in Probability and Statistics). ISBN 9780471461722. Disponível em: <<https://books.google.com.br/books?id=SpvBd7IUCxkC>>. Citado na página 21.

- RIGO, P. D.; SILUK, J. C. M.; LACERDA, D. P.; SPELLMEIER, J. P. Competitive business model of photovoltaic solar energy installers in Brazil. *Renewable Energy*, v. 181, p. 39–50, 2022. ISSN 0960-1481. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S096014812101329X>>. Citado na página 50.
- SANGRODY, H.; ZHOU, N.; ZHANG, Z. Similarity-Based Models for Day-Ahead Solar PV Generation Forecasting. *IEEE Access*, v. 8, p. 104469–104478, 2020. ISSN 2169-3536. Citado 2 vezes nas páginas 8 e 10.
- SBAYTI, M.; GHIOTTI, A.; BAHLOUL, R.; BELHADJSALAH, H.; BRUSCHI, S. Effective Strategies of Metamodeling and Optimization of Hot Incremental Sheet Forming Process of Ti6Al4V Artificial Hip Joint Component. *Journal of Computational Science*, p. 101595, 2022. ISSN 1877-7503. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1877750322000308>>. Citado na página 19.
- SINGH, D.; SINGH, B. Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, p. 105524, 2019. ISSN 1568-4946. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1568494619302947>>. Citado na página 15.
- SIVANEASAN, B.; YU, C. Y.; GOH, K. P. Solar Forecasting using ANN with Fuzzy Logic Pre-processing. *Energy Procedia*, v. 143, p. 727–732, 2017. ISSN 1876-6102. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1876610217365177>>. Citado 2 vezes nas páginas 9 e 11.
- SOBRI, S.; KOOHI-KAMALI, S.; RAHIM, N. A. Solar photovoltaic generation forecasting methods: A review. *Energy Conversion and Management*, v. 156, p. 459–497, 2018. ISSN 0196-8904. Citado 6 vezes nas páginas 2, 3, 14, 16, 17 e 27.
- SRIVASTAVA, S.; LESSMANN, S. A comparative study of LSTM neural networks in forecasting day-ahead global horizontal irradiance with satellite data. *Solar Energy*, v. 162, p. 232–247, 2018. ISSN 0038-092X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0038092X18300173>>. Citado na página 15.
- SUN, S.; WANG, S.; ZHANG, G.; ZHENG, J. A decomposition-clustering-ensemble learning approach for solar radiation forecasting. *Solar Energy*, v. 163, p. 189–199, 2018. ISSN 0038-092X. Citado na página 1.
- TEO, T. T.; LOGENTHIRAN, T.; WOO, W. L. Forecasting of photovoltaic power using extreme learning machine. In: *2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*. [S.l.: s.n.], 2015. p. 1–6. Citado na página 3.
- THEOCHARIDES, S.; MAKRIDES, G.; LIVERA, A.; THERISTIS, M.; KAIMAKIS, P.; GEORGHIU, G. E. Day-ahead photovoltaic power production forecasting methodology based on machine learning and statistical post-processing. *Applied Energy*, v. 268, p. 115023, 2020. ISSN 0306-2619. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0306261920305353>>. Citado 2 vezes nas páginas 7 e 8.

- THOREY, J.; CHAUSSIN, C.; MALLET, V. Ensemble forecast of photovoltaic power with online crps learning. *International Journal of Forecasting*, v. 34, n. 4, p. 762–773, 2018. ISSN 0169-2070. Citado na página 12.
- TOKUDA, E. K.; COMIN, C. H.; da F. Costa, L. Revisiting agglomerative clustering. *Physica A: Statistical Mechanics and its Applications*, v. 585, p. 126433, 2022. ISSN 0378-4371. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0378437121007068>>. Citado na página 16.
- TOLLEFSON, J. California wildfires and power outages cause disruptions for scientists. *Nature*, v. 575, 2019. Citado na página 2.
- UROLAGIN, S.; SHARMA, N.; DATTA, T. K. A combined architecture of multivariate LSTM with Mahalanobis and Z-Score transformations for oil price forecasting. *Energy*, v. 231, p. 120963, 2021. ISSN 0360-5442. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0360544221012111>>. Citado na página 37.
- VAGROPOULOS, S. I.; KARDAKOS, E. G.; SIMOGLU, C. K.; BAKIRTZIS, A. G.; CATALÃO, J. P. S. Ann-based scenario generation methodology for stochastic variables of electric power systems. *Electric Power Systems Research*, v. 134, p. 9–18, 2016. ISSN 0378-7796. Citado na página 17.
- VISCONDI, G. de F.; ALVES-SOUZA, S. N. A systematic literature review on big data for solar photovoltaic electricity generation forecasting. *Sustainable Energy Technologies and Assessments*, v. 31, p. 54–63, 2019. ISSN 2213-1388. Citado 5 vezes nas páginas 3, 13, 15, 16 e 17.
- VOYANT, C.; GOOIJER, J. G. D.; NOTTON, G. Periodic autoregressive forecasting of global solar irradiation without knowledge-based model implementation. *Solar Energy*, v. 174, 08 2018. Citado na página 2.
- WANG, G.; SU, Y.; SHU, L. One-day-ahead daily power forecasting of photovoltaic systems based on partial functional linear regression models. *Renewable Energy*, v. 96, p. 469–478, 2016. ISSN 0960-1481. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0960148116303913>>. Citado na página 2.
- WANG, H.; SHEN, J. An improved model combining evolutionary algorithm and neural networks for pv maximum power point tracking. *IEEE Access*, v. 7, p. 2823–2827, 2019. ISSN 2169-3536. Citado na página 14.
- WANG, J.; ZHOU, Y.; LI, Z. Hour-ahead photovoltaic generation forecasting method based on machine learning and multi objective optimization algorithm. *Applied Energy*, v. 312, p. 118725, 2022. ISSN 0306-2619. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0306261922001830>>. Citado na página 17.
- WANG, Q.; HOBBS, W. B.; TUOHY, A.; BELLO, M.; AULT, D. J. Evaluating Potential Benefits of Flexible Solar Power Generation in the Southern Company System. *IEEE Journal of Photovoltaics*, v. 12, n. 1, p. 152–160, 2022. Citado na página 1.

- WEN, L.; ZHOU, K.; YANG, S.; LU, X. Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting. *Energy*, v. 171, p. 1053–1065, 2019. ISSN 0360-5442. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0360544219300775>>. Citado 2 vezes nas páginas 8 e 10.
- XIA, J.; ZHANG, Y.; SONG, J.; CHEN, Y.; WANG, Y.; LIU, S. Revisiting Dimensionality Reduction Techniques for Visual Cluster Analysis: An Empirical Study. *IEEE Transactions on Visualization and Computer Graphics*, v. 28, n. 1, p. 529–539, 2022. Citado na página 21.
- XIAO, W.; NAZARIO, G.; WU, H.; ZHANG, H.; CHENG, F. A neural network based computational model to predict the output power of different types of photovoltaic cells. *PLOS ONE*, Public Library of Science, v. 12, n. 9, p. 1–8, 2017. Citado 2 vezes nas páginas 17 e 29.
- YAHYA, Z.; IMANE, S.; HICHAM, H.; GHASSANE, A.; Bouchini-Idrissi Safia, E. Applied imagery pattern recognition for photovoltaic modules' inspection: A review on methods, challenges and future development. *Sustainable Energy Technologies and Assessments*, v. 52, p. 102071, 2022. ISSN 2213-1388. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2213138822001230>>. Citado na página 17.
- YANG, D.; DONG, Z. Operational photovoltaics power forecasting using seasonal time series ensemble. *Solar Energy*, v. 166, p. 529–541, 2018. ISSN 0038-092X. Citado na página 12.
- YANG, M.; HUANG, X. Ultra-short-term prediction of photovoltaic power based on periodic extraction of pv energy and lsh algorithm. *IEEE Access*, v. 6, p. 51200–51205, 2018. ISSN 2169-3536. Citado 3 vezes nas páginas 2, 14 e 27.
- YAO, T.; WANG, J.; WU, H.; ZHANG, P.; LI, S.; XU, K.; LIU, X.; CHI, X. Intra-Hour Photovoltaic Generation Forecasting Based on Multi-Source Data and Deep Learning Methods. *IEEE Transactions on Sustainable Energy*, v. 13, n. 1, p. 607–618, 2022. Citado na página 13.
- YEON, S.; YU, B.; SEO, B.; YOON, Y.; LEE, K. H. Ann based automatic slat angle control of venetian blind for minimized total load in an office building. *Solar Energy*, v. 180, p. 133–145, 2019. ISSN 0038-092X. Citado na página 17.
- YOUSEFI, M.; HAJIZADEH, A.; SOLTANI, M. N. A Comparison Study on Stochastic Modeling Methods for Home Energy Management Systems. *IEEE Transactions on Industrial Informatics*, v. 15, n. 8, p. 4799–4808, 2019. Citado na página 1.
- ZEKIĆ-SUŠAC, M.; HAS, A.; KNEŽEVIĆ, M. Predicting energy cost of public buildings by artificial neural networks, CART, and random forest. *Neurocomputing*, v. 439, p. 223–233, 2021. ISSN 0925-2312. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0925231221001430>>. Citado na página 18.
- ZEREN, F.; AKKUŞ, H. T. The relationship between renewable energy consumption and trade openness: New evidence from emerging economies. *Renewable Energy*, v. 147, p. 322–329, 2020. ISSN 0960-1481. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0960148119313369>>. Citado na página 1.

ZHANG, X.; LI, Y.; LU, S.; HAMANN, H. F.; HODGE, B.; LEHMAN, B. A solar time based analog ensemble method for regional solar power forecasting. *IEEE Transactions on Sustainable Energy*, v. 10, n. 1, p. 268–279, 1 2019. ISSN 1949-3029. Citado 3 vezes nas páginas 2, 14 e 36.

ZHANG, Y.; BEAUDIN, M.; TAHERI, R.; ZAREIPOUR, H.; WOOD, D. Day-ahead power output forecasting for small-scale solar photovoltaic electricity generators. *IEEE Transactions on Smart Grid*, v. 6, n. 5, p. 2253–2262, 2015. ISSN 1949-3053. Citado na página 2.

ZHEN, Z.; LIU, J.; ZHANG, Z.; WANG, F.; CHAI, H.; YU, Y.; LU, X.; WANG, T.; LIN, Y. Deep Learning Based Surface Irradiance Mapping Model for Solar PV Power Forecasting Using Sky Image. *IEEE Transactions on Industry Applications*, v. 56, n. 4, p. 3385–3396, 2020. Citado 3 vezes nas páginas 3, 7 e 8.

ZHENG, Y.; SONG, Y.; HILL, D. J.; MENG, K. Online Distributed MPC-Based Optimal Scheduling for EV Charging Stations in Distribution Systems. *IEEE Transactions on Industrial Informatics*, v. 15, n. 2, p. 638–649, 2019. Citado na página 3.

Apêndices

APÊNDICE A – IMPLEMENTAÇÕES DE CÓDIGO USANDO *MATLAB*

doe-on-ann-forecast.m

```

1 % -----
2 % | Author: Max Moreira |
3 % -----
4
5 % DOE using ANN Ensemble for PV Gen. Forecasting
6 % -----
7
8 tic % starts timer to measure performance
9 clc % Clear comand line
10 clear % Make sure environment variables will be cleaned
11
12 .....
13 ... ANN FACTORS ...
14 .....
15
16 % Number of Hidden Layers: One or Two
17 hiddenLayers = [1 2];
18
19 % Number of Epochs
20 epochs = [100 400];
21
22 % Array containing Number of inputs selected
23 inputsNumber = [3 5];
24
25 % Array of of multipliers that defines the number of neurons per layer:
26 %  $k*(N+1)$ , where N is the number of input and  $k = 1.5, 2$ 
27 unitsPerLayer = [1.5 2];
28
29 % Learning rate values
30 learningRate = [0.1 0.9];
31
32 % Training alghorithm options:
33 % Scaled Conjugate Gradient | Levenberg-Marquardt
34 trainFcnOptions = {'trainscg' 'trainlm'};
35
36 % This function defines the transfer function used to calculate the ...
    ith layer's output

```

```
37 % tansig - Symmetric sigmoid transfer function.
38 % radbas - Radial basis transfer function.
39 %     example: net.layers{i}.transferFcn = 'radbas';
40 transferFcn = {'tansig' 'radbas'};
41
42     .....
43     ... TIME SERIE FACTORS ...
44     .....
45
46 % Consider variable unstacked by day of week or not
47 unstacked = [true false];
48
49 % Normalization method: (1) Feature Scaling | (2) Max Value
50 % > for more info, please check files "normalizeData.m" and ...
51 %     "denormalizeData.m"
52 normalizationMethod = {'featureScaling', 'maxValue'};
53
54 % Use of external variables: (1) None | (2) Strong correlation
55 correlatedVariables = [1 2];
56
57 % Number of time serie points to consider. Should be divisible by 7
58 numOfPoints = [98 679];
59
60     .....
61     ... DOE MATRIX ...
62     .....
63
64 % Load DOE matrix, where each line represents an experimental run.
65 % Values of this matrix will access the previous arrays indexes, ...
66 %     respectively
67 doeMatrix = load('data/doe_factorial.txt');
68
69 % Number of experiments (Total amount of rows)
70 doeRows = size(doeMatrix,1);
71
72     .....
73     ... DATA ...
74     .....
75
76 % Load generation data - UNSTACKED by day of week
77 kwhUnstack = load('data/kwh-unstacked.txt');
78
79 % Divide into two parts: training and test data - UNSTACKED
80 kwhTrainUnstack = kwhUnstack(1:end-1, :); %train
81 kwhTestUnstack = kwhUnstack(end, :); %test
82
83 % Load generation data - original time serie
```

```
82 kwh = load('data/kwh-normal.txt');
83
84 % Divide into two parts: training and test data - UNSTACKED
85 kwhTrain = kwh(1:end-7); %train
86 kwhTest = kwh(end-6:end); %test
87
88 % Load Hours of Insolation (UNSTACKED) in order to train ANN using ...
    an external variable
89 insolationUnstack = load('data/insolation-unstacked.txt');
90
91 % Load Hours of Insolation in order to train ANN using an external ...
    variable
92 insolation = load('data/insolation-normal.txt');
93
94 % Load Temperature (UNSTACKED) in order to train ANN using an ...
    external variable
95 temperatureUnstack = load('data/temperature-unstacked.txt');
96
97 % Load Temperature in order to train ANN using an external variable
98 temperature = load('data/temperature-normal.txt');
99
100 .....
101 ... AUTOMATE DOE ...
102 .....
103
104 % For each row of DOE matrix:
105 for run = 1:doeRows
106
107     % Get the array indexes corresponding to each Factor:
108     f1 = doeMatrix(run,1); % hiddenLayers
109     f2 = doeMatrix(run,2); % epochs
110     f3 = doeMatrix(run,3); % inputsNumber
111     f4 = doeMatrix(run,4); % unitsPerLayer
112     f5 = doeMatrix(run,5); % learningRate
113     f6 = doeMatrix(run,6); % trainFcnOptions
114     f7 = doeMatrix(run,7); % transferFcn
115     f8 = doeMatrix(run,8); % unstacked
116     f9 = doeMatrix(run,9); % normalizationMethod
117     f10 = doeMatrix(run,10); % correlatedVariables
118     f11 = doeMatrix(run,11); % numOfPoints
119
120 .....
121 ..... PREPARING EXPERIMENT .....
122 .....
123
124     if unstacked(f8) == true % ...[UNSTACKED]... by day of week
125
```

```
126     % normalizeData('trainArray', 'ArrayToNormalize', ...
127         'normalizationMethod')
128     training = normalizeData( kwhTrainUnstack, kwhTrainUnstack, ...
129         char(normalizationMethod(f9)) );
130     %test = normalizeData( kwhTrainUnstack, kwhTestUnstack, ...
131         char(normalizationMethod(f9)) );
132
133     if correlatedVariables(f10) == 2 % Weak correlation
134         training = normalizeData( temperatureUnstack, ...
135             temperatureUnstack, char(normalizationMethod(f9)) );
136     elseif correlatedVariables(f10) == 3 % Strong correlation
137         training = normalizeData( insolationUnstack, ...
138             insolationUnstack, char(normalizationMethod(f9)) );
139     end
140
141     % for each day of week, forecast each target day
142     for i = 1:7
143
144         % extractUnstackedData('dayArray', 'numberOfInputs', ...
145             'numberOfPointsToUse')
146         [in out] = extractUnstackedData( training(:,i), ...
147             inputsNumber(f3), numOfPoints(f11) );
148
149         % prepareAnnNeurons('hiddenLayers', 'numOfInputs', ...
150             'unitsPerLayer')
151         annNeurons = prepareAnnNeurons( hiddenLayers(f1), ...
152             inputsNumber(f3), unitsPerLayer(f4) );
153
154         % Create FEEDFOWARD NETWORK
155         clear net;
156         net = feedforwardnet( annNeurons, ...
157             char(trainFcnOptions(f6)) );
158
159         % Set learning rate
160         net.trainParam.lr = learningRate(f5);
161
162         % Set Number of Epochs
163         net.trainParam.epochs = epochs(f2);
164
165         % Set transfer function
166         net.layers{ hiddenLayers(f1) }.transferFcn = ...
167             char(transferFcn(f7));
168         net.layers{ hiddenLayers(f1)+1 }.transferFcn = ...
169             char(transferFcn(f7));
170
171         % TRAIN ANN
172         net = train(net, in, out);
```



```
161
162     % Extract Inputs for test case..
163     [testInput testOut] = extractUnstackedData( ...
        training(:,i), inputsNumber(f3), ...
        (inputsNumber(f3)*7+7) );
164     testInput(end+1, :) = testOut(1,:);
165     testInput(1,:) = [];
166
167     % Here we have the forecasted value for the 'i' day of ...
        desired week
168     forecasted = net(testInput);
169
170     % Denormalize forecasted value
171     forecasted = denormalizeData(kwhTrainUnstack, ...
        forecasted, normalizationMethod(f9));
172
173     % Store the forecasted value into doeMatrix.
174     doeMatrix( run, (11+i) ) = abs(forecasted);
175
176     end % end 'for i=1:7'
177
178     else % ...[ORIGINAL]... data
179
180         training = normalizeData( kwhTrain, kwhTrain, ...
            char(normalizationMethod(f9)) );
181         %test = normalizeData( kwhTrain, kwhTest, ...
            char(normalizationMethod(f9)) );
182
183         if correlatedVariables(f10) == 2 % Weak correlation
184             training = temperature(1:end-7); % the last 7 days are ...
                related to the desired week to predict
185
186         elseif correlatedVariables(f10) == 3 % Strong correlation
187             training = insolation(1:end-7); % the last 7 days are ...
                related to the desired week to predict
188
189         end
190
191         % Normalize data according to "normalizationMethod"
192         training = normalizeData( training, training, ...
            char(normalizationMethod(f9)) );
193
194         % extractUnstackedData('dayArray', 'numberOfInputs', ...
            'numberOfPointsToUse')
195         [in out] = extractData( training, inputsNumber(f3), ...
            numOfPoints(f11) );
196
```



```

235         testInput = insulation((end-(7-i)-inputsNumber(f3)): ...
                                (end-(7-i+1)));
236
237     end
238
239     % Denormalize forecasted value
240     forecasted = denormalizeData(kwhTrainUnstack, ...
                                forecasted, normalizationMethod(f9));
241
242     % Store the forecasted value into doeMatrix.
243     doeMatrix( run, (11+i) ) = abs(forecasted);
244
245     end % end 'for i=1:7'
246
247     end % end 'if unstacked'
248
249 end
250
251     .....
252     ..... MAPE .....
253     .....
254 ERROR_mape = abs((doeMatrix(:,12:end) - kwhTestUnstack) ./ ...
                kwhTestUnstack * 100);
255
256 for i=1:doeRows
257     ERROR_mape(i,8) = mean(ERROR_mape(i,1:7));
258 end
259
260 % bar chart for MAPE's
261 bar(ERROR_mape(:,8) ', 'r');
262 %z = linkage(ERROR_mape, 'ward');
263 %dendrogram(z);
264 %anova1(z)
265
266     .....
267     ..... RMSE .....
268     .....
269 ERROR_rmse = (doeMatrix(:,12:end) - kwhTestUnstack).^2;
270
271 for i=1:doeRows
272     ERROR_rmse(i,9) = (mean(ERROR_rmse(i,1:7)))^1/2;
273 end
274
275     .....
276     ..... MAE .....
277     .....
278 ERROR_mae = abs((doeMatrix(:,12:end) - kwhTestUnstack));

```

```

279
280 for i=1:doeRows
281     ERROR_mae(i,9) = (mean(ERROR_mae(i,1:7)));
282 end
283
284 toc % stop and display the timer

```

seasonal_pv_forecasting_using_PCA.m

```

1
2
3 % -----
4 % | Author: Max Moreira |
5 % -----
6 % Seasonal P.V. Generation Forecasting Using PCA
7 % -----
8
9 tic % Starts timer
10 clc % Clear comand line
11 clear % Make sure environment variables will be cleaned
12 close % Close one or more figures
13
14 .....
15 ... TIME SERIE FACTORS ...
16 .....
17
18 % Number of Principal Components
19 pcaComponents = [2 3];
20
21 % Use climate variables?
22 useClimateVars = [true false];
23
24 % Linkage method, which describes how to measure the distance ...
    between clusters
25 linkageMethod = {'ward', 'complete'};
26
27 % Defines the number of clusters to be used
28 clusters = [3 4];
29
30 % Normalization method: (1) Standardization | (2) MinMax Scaling
31 % > for more info, please check files "normalizeData.m" and ...
    "denormalizeData.m"
32 normalizationMethod = {'standardization', 'minMax'};
33
34 .....
35 ... ANN FACTORS ...

```

```
36 .....
37
38 % Number of Hidden Layers: One or Two
39 hiddenLayers = [1 2];
40
41 % Array of multipliers that defines the number of neurons per layer:
42 %  $k*(N+1)$ , where N is the number of input and  $k = 1.5, 2$ 
43 unitsPerLayer = [1.5 2];
44
45 % Learning rate values
46 learningRate = [0.1 0.9];
47
48 % Number of Epochs
49 epochs = [100 400];
50
51 % Training algorithm options:
52 % Scaled Conjugate Gradient | Levenberg-Marquardt
53 trainFcnOptions = {'trainscg' 'trainlm'};
54
55 % This function defines the transfer function used to calculate the ...
56 % ith layer's output
57 % tansig - Symmetric sigmoid transfer function.
58 % radbas - Radial basis transfer function.
59 % example: net.layers{i}.transferFcn = 'radbas';
60 transferFcn = {'tansig' 'radbas'};
61
62 .....
63 ... DOE MATRIX ...
64 .....
65 % Load DOE matrix, where each line represents an experimental run.
66 % Values of this matrix will access the previous arrays indexes, ...
67 % respectively
68 doeMatrix = load('data/doe_fractional_factorial_NEW2.txt');
69
70 % Number of experiments (Total amount of rows)
71 doeRows = size(doeMatrix,1);
72
73 mainPath = 'data/Seasonal/';
74
75 Seasons = {'1Autumn', '2Winter', '3Spring', '4Summer'};
76 %Seasons = {'2Winter'};
77
78 % Matrices to store MAPE and Standard Deviation
79 MapeAutum = zeros(32,17);
80 StdAutum = zeros(32,17);
```

```
81 MapeWinter = zeros(32,17);
82 StdWinter = zeros(32,17);
83 MapeSpring = zeros(32,17);
84 StdSpring = zeros(32,17);
85 MapeSummer = zeros(32,17);
86 StdSummer = zeros(32,17);
87 DoeMapeAutum = zeros(160,17);
88 DoeMapeSpring = zeros(160,17);
89 DoeMapeSummer = zeros(160,17);
90 DoeMapeWinter = zeros(160,17);
91 ForecastAutum = zeros(161,17);
92 ForecastWinter = zeros(161,17);
93 ForecastSpring = zeros(161,17);
94 ForecastSummer = zeros(161,17);
95
96 % For each SEASONAL path, DO
97 for s=1:length(Seasons)
98
99     % Get next Season
100    season = Seasons{s};
101
102    % Scan directory to get file list
103    Files = dir( strcat(mainPath, season, '/*.txt') );    %| ...
104    length(Files) | Files.name
105
106    % For each file (PV_Plant) into directory
107    for k=1:length(Files)
108
109        % Get file full path
110        path = strcat(mainPath, season, '/', Files(k).name)
111
112        .....
113        ... AUTOMATE DOE ...
114        .....
115
116        for run = 1:doeRows
117
118            % Comment this line to avoid extra processing
119            fprintf('%s %s %d\n', season, path, run)
120
121            % Reads and converts data file to Table format type
122            Data = readtable(path);
123
124            % Get only number of climate columns (excluding Date and ...
125            PV Generation data)
126            nCols = size(Data,2) - 2;
```

```

126     % Create an array
127     originalData = table2array(Data(:,2:end)); % Includes ...
        all of them
128     climateData = table2array(Data(:,3:end)); % Climate ...
        data only
129
130     % Get the array indexes corresponding to each Factor:
131     f1 = doeMatrix(run,1); % normalization
132     f2 = doeMatrix(run,2); % useClimateVars
133     f3 = doeMatrix(run,3); % linkageMethod
134     f4 = doeMatrix(run,4); % groups
135     f5 = doeMatrix(run,5); % pcaComponents
136     f6 = doeMatrix(run,6); % hiddenLayers
137     f7 = doeMatrix(run,7); % unitsPerLayer
138     f8 = doeMatrix(run,8); % learningRate
139     f9 = doeMatrix(run,9); % epochs
140     f10 = doeMatrix(run,10); % training
141     f11 = doeMatrix(run,11); % transfer
142
143     .....
144     ..... IS CLIMATE VARIABLES BEING USED ?? .....
145     .....
146
147     if not( useClimateVars(f2) )
148
149         climateData = originalData(:,1);
150
151         for i=2:7
152             aux = zeros( size(climateData(:,1),1), 1);
153             aux(i:end, 1) = climateData(1:(end-i+1), 1);
154             climateData(:,i) = aux(:,1);
155         end
156         %climateData = climateData(i:end,2:end);
157
158         % Work around: if climate data should be unconsidered,
159         % update nCols
160         nCols = 6;
161     end
162
163     .....
164     ..... DATA NORMALIZATION PROCESS .....
165     .....
166
167     if strcmpi( char(normalizationMethod(f1)), ...
        'standardization')
168
169         % Z = (x-mean(x)) / std(x)

```

```
170         Data.kWh(1:end-1) = ...
           (Data.kWh(1:end-1)-mean(Data.kWh(1:end-1))) / ...
           std(Data.kWh(1:end-1));
171         %Data.kWh = (Data.kWh-mean(Data.kWh))/std(Data.kWh);
172         Data.TempInst = (Data.TempInst-mean(Data.TempInst)) ...
           / std(Data.TempInst);
173         Data.TempMax = (Data.TempMax-mean(Data.TempMax)) / ...
           std(Data.TempMax);
174         Data.TempMin = (Data.TempMin-mean(Data.TempMin)) / ...
           std(Data.TempMin);
175         Data.HumiInst = (Data.HumiInst-mean(Data.HumiInst)) ...
           / std(Data.HumiInst);
176         Data.HumiMax = (Data.HumiMax-mean(Data.HumiMax)) / ...
           std(Data.HumiMax);
177         Data.HumiMin = (Data.HumiMin-mean(Data.HumiMin)) / ...
           std(Data.HumiMin);
178         Data.DewpInst = (Data.DewpInst-mean(Data.DewpInst)) ...
           / std(Data.DewpInst);
179         Data.DewpMax = (Data.DewpMax-mean(Data.DewpMax)) / ...
           std(Data.DewpMax);
180         Data.DewpMin = (Data.DewpMin-mean(Data.DewpMin)) / ...
           std(Data.DewpMin);
181         Data.PressInst = ...
           (Data.PressInst-mean(Data.PressInst)) / ...
           std(Data.PressInst);
182         Data.PressMax = (Data.PressMax-mean(Data.PressMax)) ...
           / std(Data.PressMax);
183         Data.PressMin = (Data.PressMin-mean(Data.PressMin)) ...
           / std(Data.PressMin);
184         Data.WindSpeed = ...
           (Data.WindSpeed-mean(Data.WindSpeed)) / ...
           std(Data.WindSpeed);
185         Data.WindDir = (Data.WindDir-mean(Data.WindDir)) / ...
           std(Data.WindDir);
186         Data.GustWind = (Data.GustWind-mean(Data.GustWind)) ...
           / std(Data.GustWind);
187         Data.Radiation = ...
           (Data.Radiation-mean(Data.Radiation)) / ...
           std(Data.Radiation);
188         climateData = (climateData - mean(climateData(:))) / ...
           std(climateData(:));
189     else
190         % Z = (x-min(x)) / (max(x)-min(x))
191         Data.kWh(1:end-1) = ...
           (Data.kWh(1:end-1)-min(Data.kWh(1:end-1))) / ...
           (max(Data.kWh(1:end-1))-min(Data.kWh(1:end-1)));
192         %Data.kWh = (Data.kWh-min(Data.kWh)) / ...
```



```

    (max(Data.kWh)-min(Data.kWh));
193     Data.TempInst = (Data.TempInst-min(Data.TempInst)) / ...
        (max(Data.TempInst)-min(Data.TempInst));
194     Data.TempMax = (Data.TempMax-min(Data.TempMax)) / ...
        (max(Data.TempMax)-min(Data.TempMax));
195     Data.TempMin = (Data.TempMin-min(Data.TempMin)) / ...
        (max(Data.TempMin)-min(Data.TempMin));
196     Data.HumiInst = (Data.HumiInst-min(Data.HumiInst)) / ...
        (max(Data.HumiInst)-min(Data.HumiInst));
197     Data.HumiMax = (Data.HumiMax-min(Data.HumiMax)) / ...
        (max(Data.HumiMax)-min(Data.HumiMax));
198     Data.HumiMin = (Data.HumiMin-min(Data.HumiMin)) / ...
        (max(Data.HumiMin)-min(Data.HumiMin));
199     Data.DewpInst = (Data.DewpInst-min(Data.DewpInst)) / ...
        (max(Data.DewpInst)-min(Data.DewpInst));
200     Data.DewpMax = (Data.DewpMax-min(Data.DewpMax)) / ...
        (max(Data.DewpMax)-min(Data.DewpMax));
201     Data.DewpMin = (Data.DewpMin-min(Data.DewpMin)) / ...
        (max(Data.DewpMin)-min(Data.DewpMin));
202     Data.PressInst = ...
        (Data.PressInst-min(Data.PressInst)) / ...
        (max(Data.PressInst)-min(Data.PressInst));
203     Data.PressMax = (Data.PressMax-min(Data.PressMax)) / ...
        (max(Data.PressMax)-min(Data.PressMax));
204     Data.PressMin = (Data.PressMin-min(Data.PressMin)) / ...
        (max(Data.PressMin)-min(Data.PressMin));
205     Data.WindSpeed = ...
        (Data.WindSpeed-min(Data.WindSpeed)) / ...
        (max(Data.WindSpeed)-min(Data.WindSpeed));
206     Data.WindDir = (Data.WindDir-min(Data.WindDir)) / ...
        (max(Data.WindDir)-min(Data.WindDir));
207     Data.GustWind = (Data.GustWind-min(Data.GustWind)) / ...
        (max(Data.GustWind)-min(Data.GustWind));
208     Data.Radiation = ...
        (Data.Radiation-min(Data.Radiation)) / ...
        (max(Data.Radiation)-min(Data.Radiation));
209     climateData = (climateData-min(climateData(:))) / ...
        (max(climateData(:))-min(climateData(:)));
210
211     end %if normalization
212
213     .....
214     ... CLUSTER ...
215     .....
216     Z = linkage(climateData, char(linkageMethod(f3)), ...
        'euclidean');
217
```

```
218     % Create an array of climate variables only, already ...
        normalized!
219     climateData = table2array(Data(:,3:end));
220
221     % cluster group indicators stored in 19th column
222     groupNumbers = cluster(Z, 'maxclust', clusters(f4));
223     climateData(:,19) = groupNumbers;
224
225     % Add to Data table cluster groups (numbers)
226     Data(:,19) = num2cell(groupNumbers);
227
228     % Create an array with PV generation only, normalized!
229     pvData = table2array(Data(:,2));
230     pvData(:,2) = groupNumbers;
231
232     % (FILTER PV DATA) Get all rows with the same cluster ...
        label, previous calculated
233     pvData = pvData( pvData(:, 2) == pvData(end,2), 1 );
234
235     % (FILTER CLIMATE DATA) Get all rows with the same ...
        cluster label, previous calculated
236     climateData = climateData( ...
        climateData(:,19)==climateData(end,19), 1:nCols );
237
238     .....
239     .... PCA ....
240     .....
241     [coeff,score,latent,tsquared,explained,mu] = ...
        pca(climateData);
242
243     % Prepare INPUTS and OUTPUTS for training.
244     % PCA Components (scores) are used
245     in = score(1:end-1, 1:pcaComponents(f5));
246     out = pvData(1:end-1,1);
247
248     .....
249     .... ANN ....
250     .....
251
252     % prepareAnnNeurons('hiddenLayers', 'numOfInputs', ...
        'unitsPerLayer')
253     annNeurons = prepareAnnNeurons( hiddenLayers(f6), ...
        pcaComponents(f5), unitsPerLayer(f7) );
254
255     % Create FEEDFOWARD NETWORK
256     clear net;
257     net = feedforwardnet( annNeurons, ...
```

```

        char(trainFcnOptions(f10)) );
258
259     % Set learning rate
260     net.trainParam.lr = learningRate(f8);
261
262     % Set Number of Epochs
263     net.trainParam.epochs = epochs(f9);
264
265     % Set transfer function
266     net.layers{ hiddenLayers(f6) }.transferFcn = ...
        char(transferFcn(f11));
267     net.layers{ hiddenLayers(f6)+1 }.transferFcn = ...
        char(transferFcn(f11));
268
269     % TRAIN ANN
270     net = train(net, in', out');
271
272     % Extract inputs for test case.. based on PCA Scores
273     testInput = score( end, 1:pcaComponents(f5) );
274
275     % Here we have the forecasted value for the 'i' day of ...
        the desired week
276     forecasted = net(testInput');
277
278     .....
279     .... DENORMALIZE ....
280     .....
281
282     % retrieve original data and set it to pvGen variable
283     pvGen = originalData(1:end-1,1);
284     generation = originalData(end,1);
285
286     % denormalize forecasted value
287     if strcmpi( char(normalizationMethod(f1)), ...
        'standardization')
288         denormalizedForecasted = std(pvGen) * forecasted + ...
            mean(pvGen);
289     else
290         denormalizedForecasted = (max(pvGen) - min(pvGen)) * ...
            forecasted + min(pvGen);
291     end
292
293     %format long g
294
295     .....
296     .... ERROR ....
297     .....

```

```
298
299     % MAPE
300     mape = (abs( denormalizedForecasted - generation ) / ...
301             generation) * 100;
302
303     % Store MAPE into Doe Matrix, column 12
304     doeMatrix(run, 12) = mape;
305
306     % Store Forecasted Value into Doe Matrix, column 13
307     doeMatrix(run,13) = denormalizedForecasted;
308 %
309 %     % Get file names, excluding extension '.txt'
310 %     FileNames{s,k} = Files(k).name(1:end-4);
311 %
312 %     % Acumulated PCA explanation for 2 components
313 %     AcuExp2(s,k) = explained(1,1) + explained(2,1);
314 %     % Acumulated PCA explanation for 3 components
315 %     AcuExp3(s,k) = explained(1,1) + explained(2,1) + ...
316 %     explained(3,1);
317
318     end %doeRows
319
320     % Calculate MAPE and STD for each season, considering all ...
321     % files (plants)
322     switch season
323     case '1Autumn'
324         % Calculates MAPE for all experimental run
325         DoeMapeAutum(:,k) = doeMatrix(:,12);
326         ForecastAutum(1:end-1,k) = doeMatrix(:,13);
327         ForecastAutum(161,k) = originalData(end,1); ...
328         %Generation value
329         % Calculates the mean MAPE considering each run with ...
330         % same factors
331         for i=1:32
332             MapeAutum(i,k) = mean(doeMatrix(i:32:end,12));
333             StdAutum(i,k) = std(doeMatrix(i:32:end,12));
334         end
335     case '2Winter'
336         % Calculates MAPE for all experimental run
337         DoeMapeWinter(:,k) = doeMatrix(:,12);
338         ForecastWinter(1:end-1,k) = doeMatrix(:,13);
339         ForecastWinter(161,k) = originalData(end,1); ...
340         %Generation value
341         % Calculates the mean MAPE considering each run with ...
```

```
        same factors
339     for i=1:32
340         MapeWinter(i,k) = mean(doeMatrix(i:32:end,12));
341         StdWinter(i,k) = std(doeMatrix(i:32:end,12));
342     end
343
344     case '3Spring'
345         % Calculates MAPE for all experimental run
346         DoeMapeSpring(:,k) = doeMatrix(:,12);
347         ForecastSpring(1:end-1,k) = doeMatrix(:,13);
348         ForecastSpring(161,k) = originalData(end,1); ...
            %Generation value
349         % Calculates the mean MAPE considering each run with ...
            same factors
350     for i=1:32
351         MapeSpring(i,k) = mean(doeMatrix(i:32:end,12));
352         StdSpring(i,k) = std(doeMatrix(i:32:end,12));
353     end
354
355     case '4Summer'
356         % Calculates MAPE for all experimental run
357         DoeMapeSummer(:,k) = doeMatrix(:,12);
358         ForecastSummer(1:end-1,k) = doeMatrix(:,13);
359         ForecastSummer(161,k) = originalData(end,1); ...
            %Generation value
360         % Calculates the mean MAPE considering each run with ...
            same factors
361     for i=1:32
362         MapeSummer(i,k) = mean(doeMatrix(i:32:end,12));
363         StdSummer(i,k) = std(doeMatrix(i:32:end,12));
364     end
365     end %switch
366
367 %     if k == 2
368 %         return
369 %     end
370
371 end %files
372
373 %     figure(1)
374 %     subplot(2,2,s)
375 %     %figure(s)
376 %     bar(AcuExp2(s,:))
377 %     title('PCA2','FontSize',9)
378 %     xticklabels(FileNames)
379 %     xtickangle(90)
380 %     xlabel(season)
```

```

381 %     ylabel('Explanation for 2 components')
382 %     legend off
383 %     %hold off
384 %
385 %     figure(2)
386 %     subplot(2,2,s)
387 %     %figure(s)
388 %     bar(AcuExp3(s,:))
389 %     title('PCA3','FontSize',9)
390 %     xticklabels(FileNames(s,:))
391 %     xtickangle(90)
392 %     xlabel(season)
393 %     ylabel('Explanation for 3 components')
394 %     legend off
395
396 end %season
397
398 %fn = string(Files.name);
399 %fn = categorical(Files.name);
400
401
402
403 % Plotting bar chart with explained PCA for each file
404 % figure
405 % bar(Expl)
406 % xticklabels(FileNames)
407 % xtickangle(90)
408 % xlabel('PV stations')
409 % ylabel('Explanation for 2 components')
410
411 % save('PCA.mat','variables');
412 %save('PCA.mat','AcuExp2','AcuExp3','FileNames');
413
414 save('PCA.mat');
415
416 toc

```

prepareAnnNeurons.m

```

1
2 function [ neuronsArray ] = prepareAnnNeurons( hiddenLayers, ...
        numOfInputs, unitsPerLayer )
3 %PREPAREANN Summary of this function goes here
4 %   Detailed explanation goes here
5
6     neuronsArray = 1:hiddenLayers;

```

```
7
8     for i=1:hiddenLayers
9
10         % k*(N+1), where N is the number of input and k = 1, 1.5, 2
11         neuronsArray(i) = unitsPerLayer * (numOfInputs + 1);
12
13     end
14
15 end
```

normalizeData.m

```
1
2 function [ normalizedMatrix ] = normalizeData( originalMatrix, ...
3     matrixToNormalize, normalizationMethod )
4     %NORMALIZEDATA: normalizes data according to normalization ...
5     method below
6     % -> 'maxValue'
7     %     n = x[i] / max(X)
8     % -> 'featureScaling'
9     %     n = (x[i] - min(X)) / (max(X)-min(X))
10
11     maxVal = max(originalMatrix(:));
12     minVal = min(originalMatrix(:));
13     desvPd = std(originalMatrix(:));
14     oRows  = length(originalMatrix);
15
16     if strcmpi(normalizationMethod, 'maxValue')
17         for i=1:oRows
18             normalizedMatrix(i,:) = matrixToNormalize(i,:) / maxVal;
19         end
20     end
21
22     if strcmpi(normalizationMethod, 'featureScaling')
23         for i=1:oRows
24             normalizedMatrix(i,:) = ...
25                 (matrixToNormalize(i,:)-minVal)/(maxVal-minVal);
26         end
27     end
28
29     if strcmpi(normalizationMethod, 'none')
30         normalizedMatrix = matrixToNormalize;
31     end
32 end
```

denormalizeData.m

```

1
2 function [ denormalizedArray ] = denormalizeData( originalMatrix, ...
   matrixToDenormalize, normalizationMethod )
3   %DENORMALIZEDATA: denormalizes data according to normalization ...
   method
4   % chosen before
5   %   -> 'maxValue'
6   %       x[i] = n * max(X)
7   %   -> 'featureScaling'
8   %       x[i]= ((max(X)-min(X)) * n) + min(X)
9
10  maxVal = max(originalMatrix(:));
11  minVal = min(originalMatrix(:));
12  desvPd = std(originalMatrix(:));
13  oRows  = length(matrixToDenormalize);
14
15  if strcmpi(normalizationMethod, 'maxValue')
16      for i=1:oRows
17          denormalizedArray(i,:) = matrixToDenormalize(i,:) * maxVal;
18      end
19  end
20
21  if strcmpi(normalizationMethod, 'featureScaling')
22      for i=1:oRows
23          denormalizedArray(i,:) = ...
                matrixToDenormalize(i,:) * (maxVal-minVal)+minVal;
24      end
25  end
26
27  if strcmpi(normalizationMethod, 'none')
28      denormalizedArray = matrixToDenormalize;
29  end
30
31 end

```

extractData.m

```

1
2 function [ inputMatrix, outputMatrix ] = extractData( trainMatrix, ...
   numOfInputs, numOfPoints )
3 %EXTRACTDATA extracts inputs and outputs for training, considering ...
   ORIGINAL TIME SERIE
4 % This function reorganizes the data obeying the set of parameters

```



```

5
6     rows = length( trainMatrix );
7     startIndex = rows - numOfPoints + 1;
8
9     %     inputMatrix = zeros(numOfInputs, numOfWeeks-numOfInputs+1)
10
11     c = 1;
12
13     for i=startIndex:( rows - numOfInputs )
14
15         inputMatrix(:, c) = trainMatrix(i:i+numOfInputs-1);
16
17         outputMatrix(1, c) = trainMatrix(i+numOfInputs);
18
19         c = c + 1;
20
21     end
22
23 end

```

extractUnstackedData.m

```

1
2 function [inputMatrix, outputMatrix] = ...
3     extractUnstackedData(trainMatrix, numOfInputs, numOfPoints)
4 %EXTRACTUNSTACKEDDATA extracts inputs and outputs for training,
5 % considering UNSTACKED TIME SERIE BY DAY OF WEEK
6 % This function reorganizes the data obeying the set of parameters
7
8     % if (~exist('numOfWeeks', 'var')) -- use this in case of an ...
9     optional
10    % parameter..
11    numOfInputs = numOfInputs - 1;
12    numOfWeeks = fix(numOfPoints/7);
13    rows = length( trainMatrix );
14    startIndex = rows - numOfWeeks + 1;
15
16    %     inputMatrix = zeros(numOfInputs, numOfWeeks-numOfInputs+1)
17
18    c = 1;
19
20    for i=startIndex:( startIndex + numOfWeeks - numOfInputs - 2 )
21
22        inputMatrix(:, c) = trainMatrix(i:i+numOfInputs);
23
24        outputMatrix(1, c) = trainMatrix(i+numOfInputs+1);

```

```
23
24     c = c + 1;
25
26     end
27
28 %     inputMatrix = originalMatrix(fromIndex:(fromIndex+inputs-1), :);
29 %
30 %     outMatrix = originalMatrix((fromIndex+inputs), :);
31
32 end
```

APÊNDICE B – ARTIGO APROVADO

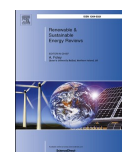
Renewable and Sustainable Energy Reviews 135 (2021) 110450



Contents lists available at ScienceDirect

Renewable and Sustainable Energy Reviews

journal homepage: <http://www.elsevier.com/locate/rser>



Design of experiments using artificial neural network ensemble for photovoltaic generation forecasting

M.O. Moreira^{a,c}, P.P. Balestrassi^{b,c,*}, A.P. Paiva^b, P.F. Ribeiro^c, B.D. Bonatto^c

^a Federal Institute of Education, Science and Technology – South of Minas Gerais, Carmo de Minas, MG, Brazil

^b Institute of Production Engineering and Management, Federal University of Itajubá, Itajubá, MG, Brazil

^c Institute of Electrical Systems and Energy, Federal University of Itajubá, Itajubá, MG, Brazil

ARTICLE INFO

Keywords

Photovoltaic forecast
Design of experiments
Artificial neural networks
Ensemble

ABSTRACT

In recent years, renewable and sustainable energy sources have attracted the attention of various investors and stakeholders, such as energy sector agents and even consumers. It is perplexing to observe and anticipate the required levels of photovoltaic generation, which are inherent tasks for such rapid insertion into the electric grid. This distributed/ renewable generation must be integrated in a coordinated way such that there is no negative impact on the electric performance of the grid, increasing in the complexity of energy management. In this article, a methodology for photovoltaic generation forecasting is addressed for a horizon of one week ahead, using a new approach based on an artificial neural network (ANN) ensemble. Two main questions will be explored with this approach: how to select the ANNs, and how to combine them in the ensemble. The design of experiments (DOE) approach is applied to the photovoltaic time series factors and ANN factors. Then, a cluster analysis is performed to select the best networks. From this point on, a mixture (MDE) is employed to determine the ideal weights for the ensemble formation. The methodology is detailed throughout the paper and, based on the combination of forecasts, the photovoltaic generation was estimated for a specific panel set located in the state of Minas Gerais, Brazil, reaching the value of 4.7% for the weekly mean absolute percentage error. The versatility of the proposed method allowed the change of the number of factors to be used in the experimental arrangement, the forecast model, and the desired forecast horizon, and consequently enhancing the forecasting determination.

* Corresponding author. Institute of Production Engineering and Management, Federal University of Itajubá, Itajubá, MG, Brazil.
E-mail address: pedro@unifei.edu.br (P.P. Balestrassi).

<https://doi.org/10.1016/j.rser.2020.110450>

Received 3 December 2019; Received in revised form 8 September 2020; Accepted 5 October 2020

Available online 14 October 2020

1364-0321/ © 2020 Elsevier Ltd. All rights reserved.