



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

LUCAS BARROS SCIANNI MORAIS

**PREVISÃO DE CARGA DE CURTO PRAZO
UTILIZANDO REDES NEURAS ARTIFICIAIS**

Itajubá, MG
Março de 2022



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

LUCAS BARROS SCIANNI MORAIS

PREVISÃO DE CARGA DE CURTO PRAZO UTILIZANDO REDES NEURAS ARTIFICIAIS

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

Área de Concentração: Sistemas Elétricos de Potência / Economia do Setor Eletro-Energético

Orientador: Prof. Dr. José Wanderley Marangon Lima

Co-Orientador: Prof. PhD. Anderson Rodrigo de Queiroz

Itajubá, MG
Março de 2022

DEDICATÓRIA

AOS MEUS PAIS E MEUS AVÓS

AGRADECIMENTOS

Aos meus pais, Christina e Paulo e ao meu irmão Gabriel, que me deram forças, me apoiaram nas decisões e me ajudaram a enxergar o melhor caminho a seguir. Aos meus avós Mario Ívo e Benita, que nunca me deixaram faltar.

À minha ex-namorada, atual noiva e futura esposa Sophie, pela amizade e companheirismo. Por me acalmar quando necessário e sempre me incentivar a correr atrás dos meus sonhos.

Aos amigos/colegas/incentivadores/professores Yago, Giancarlo e Victor pelo companheirismo e ajuda em todos os momentos que precisei.

A todos os colegas e professores com quem interagi durante esse período. Todos foram excepcionais e indispensáveis.

Agradeço também aos orientadores Rodrigo e Marangon, pela orientação, amizade, dedicação e confiança em meu trabalho e minha capacidade.

É importante reconhecer e agradecer a empresa Energisa pelo apoio e patrocínio do projeto de P&D PD-06585 “Redes neurais aplicadas à previsão de afliência e demanda para suporte à formação de preços de energia elétrica no curto prazo” que motivou o desenvolvimento deste trabalho. E a empresa MC&E pela oportunidade e confiança. Este P&D foi coordenado pela empresa MC&E e UNIFEI, as quais também merecem o meu agradecimento.

Por fim, agradeço também ao patrocínio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001

RESUMO

Em sistemas elétricos de potência, um dos principais desafios enfrentados pelos operadores independentes dos sistemas (ISO) é garantir um equilíbrio confiável entre oferta e demanda de energia. Para isso, é necessário um planejamento adequado da operação do sistema. A Previsão de Carga de Curto Prazo (PCCP) é essencial nesse processo, uma vez que ela auxilia no planejamento das operações diárias, incluindo demanda e oferta correspondentes, definição de preços futuros de eletricidade e definição de reservas de geração. À medida que as tecnologias de redes inteligentes e as fontes intermitentes de energia renovável vem aumentando significativamente nos mercados de eletricidade, a tarefa de prever a carga se torna mais complexa e representa um desafio para os ISOs. Modelos não lineares baseados em técnicas de aprendizado de máquina tornaram-se bastante populares nos últimos anos, bem como o uso de modelos híbridos focados em problemas específicos. Este trabalho foca no desenvolvimento de modelos de redes neurais artificiais para resolver o problema de PCCP, dentre eles as redes neurais MultiLayer Perceptron (MLP) com diferentes números de camadas e as redes neurais recorrentes Long-Short Term Memory (LSTM) e Gated Recurrent Unit (GRU) em suas versões unidirecionais e bidirecionais. Além dos dados disponibilizados pelo ISO, informações de temperatura prevista obtidas do modelo Global Ensemble Forecast System (GEFS) foram utilizadas para tentar gerar previsões de carga mais precisas. Em geral, os resultados mostram que os modelos de redes neurais recorrentes produziram maior acurácia e resultados mais confiáveis do que os demais modelos, incluindo os modelos utilizados pelo operador brasileiro. Isso é enfatizado com a aplicação do teste Diebold-Mariano que realiza uma comparação pareada entre modelos.

Palavras-chave: Previsão de Carga de Curto Prazo; Planejamento de Sistemas Elétricos; *Machine Learning*; Redes Neurais Artificiais; Modelo Climático Global.

ABSTRACT

In electrical power systems, one of the main challenges faced by system operators (ISO) is ensuring a reliable balance between energy supply and demand. For this, a proper planning of the system operation is necessary. Short-Term Load Forecasting (PCCP) is essential in this process, as it assists in planning daily operations, including matching demand and supply, setting future electricity prices and defining generation reserves. As smart grid technologies and intermittent renewable energy sources have increased significantly in electricity markets, the task of forecasting the load has increased in complexity and poses a challenge for ISOs. Nonlinear models based on machine learning techniques have become quite popular in recent years, as has the use of hybrid models focused on specific problems. This work focuses on the development of artificial neural network models to solve the PCCP problem, among them the MultiLayer Perceptron (MLP) networks with different numbers of layers and the Long-Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) recurrent networks in its unidirectional and bidirectional versions. In addition to the data provided by ISO, predicted temperature information obtained from the Global Ensemble Forecast System (GEFS) model was used to try to generate more accurate load forecasts. In general terms, the results show that the recurrent models produce greater accuracy and more reliable results than the other models, including the models used by the Brazilian ISO. This is emphasized with the application of the Diebold-Mariano in a paired comparison test between models.

Keywords: Short-Term Load Forecast; Electrical Systems Planning; Machine Learning; Artificial Neural Networks; Global Climate Model.

LISTA DE FIGURAS

Figura 1 - Modelos de Planejamento Elétrico utilizados pelo ONS.....	15
Figura 2 - Fluxograma do modelo de PCCP do ONS	16
Figura 3 - Fluxograma de previsão de carga média diária no PrevCargaDESSEM.....	17
Figura 4 – Exemplo geral de uma MLP rasa	22
Figura 5 – Exemplo de DNN.....	25
Figura 6 – Funcionamento de uma LSTM.....	26
Figura 7 – Arquitetura de uma GRU.	28
Figura 8 – RNN bidirecional	29
Figura 9 - Metodologia Dropout.....	31
Figura 10 - Validação Cruzada K-fold	32
Figura 11 - Comparação da Temperatura equivalente verificada e a temperatura prevista pelo GEFS	38
Figura 12 - Comparação da Temperatura equivalente verificada e a temperatura prevista pelo GEFS após aplicação de regressão	39
Figura 13 - Montagem do input para treinamento.....	40
Figura 14 - Exemplo de input para RNN.....	40
Figura 15 – Fluxograma da previsão de carga no curto prazo.....	42
Figura 16 - Comparação de Desempenho – Submercado SE/CO	45
Figura 17 - Comparação de Desempenho – Submercado S	45
Figura 18 - Comparação de Desempenho - Submercado NE.....	46
Figura 19- Comparação de Desempenho - Submercado N	46
Figura 20 – Performance da estrutura bi-GRU treinadas com dados do GEFS e dados do SINtegre.....	56

LISTA DE TABELAS

Tabela 1 – Performance da MLP rasa e das DNN.....	43
Tabela 2 – Performance das previsões por arquitetura por submercado.	44
Tabela 3 – Comparação par a par entre a performance dos modelos no SE/CO.....	47
Tabela 4 – Comparação par a par entre a performance dos modelos no S.....	48
Tabela 5 – Comparação par a par entre a performance dos modelos no NE.....	49
Tabela 6 – Comparação par a par entre a performance dos modelos no N.	50
Tabela 7 – Performance da MLP rasa e das DNN com dados do GEFS e SINtegre	51
Tabela 8 - Performance dos modelos para o submercado SE/CO utilizando dados previstos do SINtegre e do GEFS.	52
Tabela 9 – Comparação par a par entre a performance dos modelos com os dados do GEFS.	53
Tabela 10 – Comparação par a par entre a performance dos modelos com os dados do SINtegre.	54
Tabela 11 – Comparação entre os modelos treinados com dados do GEFS e do SINtegre	55

LISTA DE ABREVIATURAS E SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
ANN	Artificial Neural Network
Bi-GRU	Rede Neural <i>Gated Recurrent Unit</i> Bidirecional
Bi-LSTM	Rede Neural <i>Long Short-Term Memory</i> Bidirecional
Bi-RNN	Rede Recorrente Bidirecional
CCEE	Câmara de Comercialização de Energia Elétrica
CNN	Rede Neural Convolucional
DM	Teste de Diebold-Mariano
DNN	Deep Neural Networks
ESN	<i>Echo State Network</i>
GEFS	<i>Global Ensemble Forecast System</i>
GRU	<i>Gated Recurrent Unit</i>
IBGE	Instituto Brasileiro de Geografia e Estatística
ISO	Operador de Sistema Independente
LSTM	<i>Long Short Term Memory</i>
MAPE	Erro Percentual Absoluto Médio
MLP	<i>Multilayer Perceptron</i>
N	Submercado de Energia Norte
NCEP	<i>National Centers for Environmental Prediction</i>
NE	Submercado de Energia Nordeste
NSE	Coeficiente Nash-Sutcliffe Efficiency
ONS	Operador Nacional do Sistema
PCCP	Previsão de Carga de Curto Prazo
RBF	<i>Radial Basis Functions</i>
RMSE	Erro Quadrático Médio
RNN	Redes Neurais Recorrentes
S	Submercado de Energia Sul
SE/CO	Submercado de Energia Sudeste/Centro-Oeste
SIN	Sistema Interligado Nacional
SVM	Support Vector Machine

Uni-GRU Rede Neural *Gated Recurrent Unit* Unidirecional

Uni-LSTM Rede Neural *Long Short-Term Memory* Unidirecional

SUMÁRIO

LISTA DE FIGURAS	v
LISTA DE TABELAS	vi
LISTA DE ABREVIATURAS E SIGLAS	vii
CAPÍTULO 1 INTRODUÇÃO	11
1.1. Definição do Problema	11
1.2. Motivação	12
1.3. Objetivos.....	13
1.4. Estrutura da Dissertação.....	14
CAPÍTULO 2 PREVISÃO DE CARGA DE CURTO PRAZO: ESTADO DA ARTE	15
2.1 Aplicação de PCCP no Brasil.....	15
2.2 Trabalhos Relacionados no Brasil	18
2.3 Trabalhos Relacionados no Exterior	19
CAPÍTULO 3 MODELOS DE PREVISÃO	21
3.1 Introdução	21
3.2 MLP: Multilayer Perceptron	21
3.3 DNN: Deep Neural Networks.....	24
3.4 LSTM: Long Short-Term Memory	25
3.5 GRU: Gated Recurrent Unit.....	27
3.6 Modelos Bidirecionais: BI-LSTM E BI-GRU.....	29
3.7 Técnicas Utilizadas para Melhorar os Modelos	30
3.8 Métricas De Acurácia	32
3.8.1 <i>Mean Average Percentage Error</i>	32
3.8.2 <i>Coeficiente Nash-Sutcliffe Efficiency</i>	33
3.8.3 Teste de Diebold-Mariano.....	33
CAPÍTULO 4 METODOLOGIA	36
4.1 Dados Considerados Para Treinamento e Teste.....	36
4.2 Funcionamento e Parâmetros dos Modelos	41
CAPÍTULO 5 RESULTADOS E DISCUSSÃO	43

5.1	Simulações Utilizando Temperaturas Do Sintegre	43
5.2	Comparação de resultados: SINtegre vs GEFS.....	50
CAPÍTULO 6 CONCLUSÕES		57
6.1	Considerações Finais	57
6.2	Trabalhos Futuros	58
REFERÊNCIAS		59

CAPÍTULO 1

INTRODUÇÃO

1.1. Definição do Problema

Um dos principais desafios enfrentados na operação de redes elétricas realizado em geral pelos operadores independentes de sistemas (do inglês, *Independent System Operator* - ISO) é garantir um equilíbrio confiável entre oferta e demanda de energia [1]. Existem incertezas futuras associadas a recursos, tecnologias, condições climáticas, perfis de demanda e outros fatores que aumentam a dificuldade para os ISOs. Com o surgimento dos prosumidores, que através de fontes renováveis geram parte (ou toda) de sua demanda de eletricidade, novos investimentos em tecnologia e infraestrutura [2-3] têm sido feitos para acomodar esse montante de energia renovável. Esse processo contribui para o aumento na complexidade do perfil de demanda do sistema e consequentemente, para o planejamento da operação de sistemas elétricos. No entanto, para garantir um planejamento robusto e adequado para a operação do sistema, é de extrema importância realizar previsões precisas para a demanda de eletricidade em diferentes horizontes de tempo (do curtíssimo ao longo prazo) para definir os despachos dos geradores (estabelecer cronogramas antecipados), a expansão do sistema elétrico (linhas de transmissão e redes de distribuição), reduzir os custos dos sistemas e gerenciar com precisão a rede, garantindo confiabilidade ao consumidor final.

Em linhas gerais, a previsão de carga pode ser classificada em quatro categorias: longo prazo (um ou mais anos à frente); médio prazo (uma semana a meses à frente); curto prazo (uma hora a dias à frente) [4]; e curtíssimo prazo (5 a 30 minutos à frente). A previsão de carga de longo prazo é realizada para auxiliar no planejamento da expansão de capacidade em sistemas elétricos. A previsão de carga de médio prazo é necessária para programação de fornecimento de combustível, gestão hídrica, operações de manutenção e planejamento de importação/exportação de energia. A previsão de carga de curto prazo (PCCP) é principalmente focada no planejamento/programação das operações diárias do sistema, em tarefas como intercâmbio de energia elétrica e gerenciamento de demanda [5]. Por último, a previsão de carga de curtíssimo prazo é geralmente usada para manter as reservas de geração nos valores definidos pelo ISO [6] e programação em tempo real dos recursos de geração [7].

No Brasil, recentemente a agência reguladora de energia elétrica (ANEEL) estabeleceu o uso de um modelo de despacho hidrotérmico de curto prazo denominado DESSEM pelo ISO brasileiro (Operador Nacional do Sistema - ONS) e pela Câmara de Comercialização de Energia Elétrica (CCEE) para auxiliar a operação do sistema em relação a tomadas de decisão intra-horária para o dia seguinte. O objetivo é usar o modelo DESSEM para estabelecer diretrizes de despacho para as operadoras, bem como para definir preços horários para o mercado de eletricidade de curto prazo [8]. O DESSEM utiliza a PCCP para otimizar o sistema e definir estratégias operacionais, portanto o desenvolvimento de modelos de PCCP é essencial para esse processo que define a programação antecipada e a precificação no mercado de energia elétrica no Brasil, e o ONS está buscando continuamente desenvolver e melhorar os modelos de PCCP de forma a se alcançar previsões mais precisas.

1.2. Motivação

Modelos de PCCP vêm sendo usados por vários ISOs para planejar a operação elétrica para um horizonte de um dia a uma semana à frente e para precificação de eletricidade, como é o caso do Brasil com o modelo DESSEM e outros mercados como na Finlândia [9], Espanha [10], Austrália [11], Ontário [12] e Nova Iorque [13]. Estes modelos têm sido amplamente estudados, e os principais modelos desenvolvidos e aplicados para tal tarefa estão divididos entre lineares e não lineares. Embora os modelos lineares sejam aplicados há décadas em muitas áreas da ciência e sejam capazes de representar características físicas como variáveis climáticas e sociais, eles são limitados em termos de incorporar a não linearidade típica do comportamento das cargas elétricas de curto prazo [14-15].

Os modelos lineares clássicos como autorregressivo (AR), modelos de heterocedasticidade condicional autorregressiva generalizada (GARCH) e suavização exponencial usam apenas dados históricos de carga como recursos de entrada. Outras classes de modelos como os semiparamétricos, AR com entradas exógenas e Redes Neurais Artificiais (do inglês, Artificial Neural Network - ANN) consideram outros atributos como temperatura e variáveis de calendário para PCCP [16].

Para simplificar o problema de PCCP, modelos lineares clássicos decompõem a série temporal em componentes sazonais e irregulares ou aplicam a transformação *wavelet* [17]. Esses componentes são modelados de forma independente e apresentam menor complexidade do que a série original [18]. Modelos não lineares melhoram essa tarefa, uma vez que são capazes de identificar e reproduzir as não linearidades presentes na série analisada. Os modelos

não lineares clássicos são os que compõem o grupo das redes neurais artificiais como o *Multi-Layer Perceptrons* (MLP) e *Radial Basis Functions* (RBF) com uma camada oculta [19]. Com o tempo, avanços e melhorias de hardware e aprendizado de máquina permitiram a execução de modelos que podem trabalhar com um grande conjunto de dados e parâmetros, aumentando ainda mais a capacidade de abstração e melhorando a performance.

Essas evoluções viabilizaram uma maior aplicação das Máquina Vetor Suporte (do inglês, *Support Vector Machine* - SVM) e das Redes Neurais Profundas (do inglês, *Deep Neural Networks* - DNN), permitindo maior eficiência no processo de aprendizado de máquina para problemas não lineares [20]. Nesse contexto, muitos pesquisadores desenvolveram alternativas para melhorar o desempenho da PCCP, como o modelo ensemble (técnica que consiste em combinar os resultados de múltiplos modelos para se obter uma previsão mais precisa) e os modelos híbridos baseados em metodologias que incluem previsões meteorológicas [21]. Modelos DNN como redes com muitas camadas ocultas [22], e as Redes Neurais Recorrentes (RNN) que podem modelar sequências de séries temporais de maior complexidade são exemplos que podem beneficiar a precisão das previsões [23].

1.3. Objetivos

Este trabalho é focado no desenvolvimento e comparação entre modelos de PCCP, uma vez que estas previsões desempenham um papel fundamental no suporte às ISOs, em especial o ONS no caso brasileiro, para tomar decisões sobre a programação de geração de um dia a uma semana a frente. Estruturas de redes neurais do tipo *Multilayer Perceptron* (MLP), *Long Short Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU) são exploradas com o objetivo de se obter modelos com boa acurácia e estabilidade. Dados de temperatura prevista de um modelo climático global também serão avaliados, visando melhorar a acurácia dos modelos. Ao final, a partir do teste estatístico de Diebold-Mariano será avaliado se o uso dos dados do modelo climático ofereceram ganhos de performance nos modelos.

Os objetivos específicos deste trabalho são:

1. Maior entendimento sobre Sistema Interligado Nacional (SIN);
 2. Familiarização com o sistema integrado do ONS (SINtegre);
 3. Desenvolvimento de técnicas avançadas de programação e *machine learning*;
 4. Avaliação de diferentes modelos de *machine learning*;
 5. Familiarização com modelos climáticos globais.
-

1.4. Estrutura da Dissertação

Esta dissertação foi estruturada em seis capítulos, de tal forma que os conceitos sejam apresentados conforme a ordem em que são necessários.

CAPÍTULO 1 – INTRODUÇÃO: uma breve introdução a respeito da necessidade do desenvolvimento de modelos de previsão de carga, especialmente da previsão de carga de curto prazo e suas aplicações no Brasil.

CAPÍTULO 2 – PREVISÃO DE CARGA DE CURTO PRAZO: ESTADO DA ARTE: contextualização do processo de planejamento eletroenergético do Brasil e como a PCCP é feita e utilizada pelo operador do sistema brasileiro (ONS). Também é apresentada uma revisão da literatura com estudos relacionados a aplicação de *machine learning* para PCCP publicados no Brasil e no exterior.

CAPÍTULO 3 – MODELOS DE PREVISÃO: descrição e formulação matemática dos modelos de *machine learning* utilizados para realizar a PCCP e das métricas utilizadas para avaliar a qualidade das previsões e comparar os modelos.

CAPÍTULO 4 – METODOLOGIA: apresentação da metodologia utilizada no trabalho, desde a obtenção de dados, tratamentos e montagem dos inputs das redes até o treinamento e testes dos modelos.

CAPÍTULO 5 – RESULTADOS: análise e discussão dos resultados encontrados e comparação entre os modelos através de métricas de acurácia e do teste de hipótese.

CAPÍTULO 6 – CONCLUSÕES: conclusões do trabalho e propostas de trabalhos futuros.

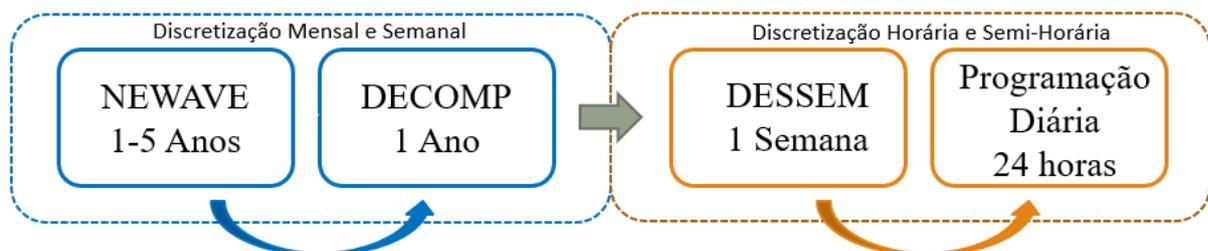
CAPÍTULO 2

PREVISÃO DE CARGA DE CURTO PRAZO: ESTADO DA ARTE

2.1 Aplicação de PCCP no Brasil

O operador nacional do sistema elétrico brasileiro (ONS) vem utilizando ao longo dos anos algoritmos computacionais para otimizar a programação operacional do sistema de geração hidrotérmica, visando o custo mínimo de operação, em diferentes horizontes de planejamento [25]. O modelo NEWAVE realiza a otimização energética ao longo de 5 anos com intervalos mensais, determinando o preço mensal da eletricidade. Por sua vez, o modelo DECOMP auxilia no planejamento da operação de dois a seis meses com intervalos semanais, determinando preços semanais. O DESSEM foi projetado para minimizar o custo operacional do despacho do sistema hidrotérmico para até duas semanas a frente com discretização semi-horária, determinando os preços horários para o dia seguinte e por patamar para os demais dias da semana. Para uma descrição detalhada sobre esses modelos ver Maceira et al. [26]. A Figura 1 ilustra o modelo de planejamento do ONS.

Figura 1 - Modelos de Planejamento Elétrico utilizados pelo ONS



Fonte: Autor

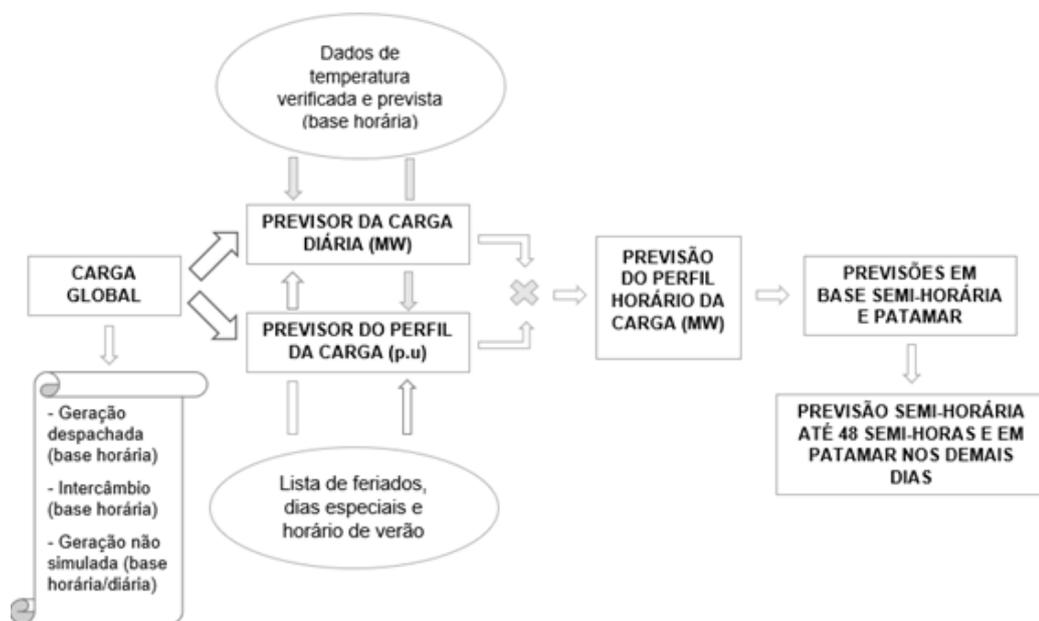
Estes modelos consideram a representação do sistema elétrico composto por: usinas hidrelétricas, usinas reversíveis, usinas termelétricas, e outras usinas renováveis como eólica, biomassa, solar etc. (cujo despacho não é otimizado pelo modelo). Para representar a rede, as usinas estão dispostas em submercados (subsistemas) interligados. Assim, as previsões de carga são realizadas separadamente para os quatro submercados brasileiros (Sudeste/Centro-Oeste – SE/CO, Sul - S, Nordeste - NE, Norte - N).

No modelo DESSEM a PCCP é uma informação de entrada essencial para a execução da programação diária da operação do sistema de geração de energia brasileiro. Para isso, o

ONS emprega modelos que utilizam algoritmos de aprendizado de máquina. O principal modelo de previsão de carga do operador brasileiro, denominado PrevCargaDESSEM, realiza previsões de carga até 192 horas à frente com resolução temporal semi-horária e por patamar.

Se o dia D da previsão é um dia normal, o PrevCargaDESSEM treina modelos SVM e ANN para a previsão da demanda média diária em MW e na sequência treina 24 SVM, um para cada hora do dia, para previsão do respectivo perfil de carga em p.u. da demanda média diária. A previsão horária é obtida pelo produto entre a previsão da demanda média diária (MW) e a previsão do perfil horário (em p.u.), como ilustra a Figura 2.

Figura 2 - Fluxograma do modelo de PCCP do ONS



Fonte: Autor

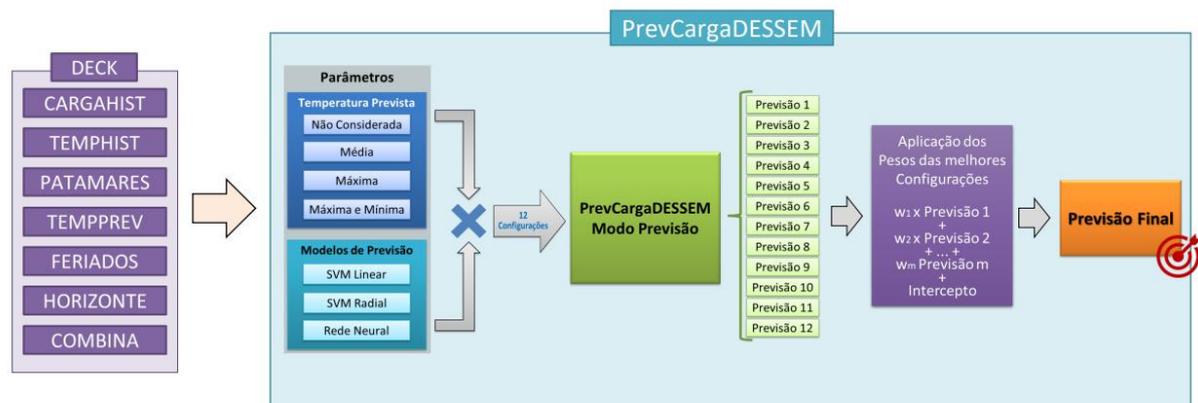
Os dados utilizados pelo ONS para realizar a PCCP são: série de dados de carga global em MW (para cada hora); histórico de temperatura horária verificada obtidos da Rede de Meteorologia do Comando da Aeronáutica (REDEMET); previsão de temperatura horária obtidas através de aeródromos e ponderadas por pesos calculados através de algoritmos de otimização (como algoritmo genético); variáveis de calendário (hora, dia, mês e ano); feriados e dias especiais; horário de verão (início e fim); horizonte de previsão (início e fim).

Para a previsão da demanda média diária (em MW) para um determinado dia D, o programa oferece três métodos: ANN (com dois neurônios em uma camada oculta), SVM com kernel linear e SVM com kernel radial. Nas três opções disponíveis as variáveis explicativas incluem as demandas médias dos dias D-1 e D-7 e a temperatura, sendo que para esta última

variável o programa oferece quatro especificações possíveis: sem temperatura, média da temperatura diária, máxima temperatura diária e valores mínimo e máximo da temperatura diária. Portanto, o PrevCargaDESSEM oferece 12 possibilidades para a previsão da demanda média diária, resultantes da combinação de 3 métodos de previsão (ANN, SVM com kernel linear e SVM com kernel radial) vezes 4 modelagens para a variável temperatura (sem temperatura, temperatura média, temperatura máxima e temperaturas mínima e máxima). O PrevCargaDESSEM ainda permite que o usuário execute cada uma das 12 modelagens disponíveis e obtenha a previsão por cada método (previsão individual) ou faça uma execução em lote dos 12 modelos e ao final calcule a média ponderada das previsões (previsão combinada), com ponderações calculadas previamente pelo ONS (disponíveis no arquivo denominado *combina*). A Figura 3 ilustra o funcionamento desta etapa.

Os arquivos do conjunto de entrada e a previsão diária estão disponíveis em um banco de dados do ONS que pode ser encontrado no endereço eletrônico da plataforma SINtegre [27].

Figura 3 - Fluxograma de previsão de carga média diária no PrevCargaDESSEM



Fonte: [28]

Já a previsão do perfil horário de carga para um dia D normal, em p.u. da demanda média, conta com 24 SVM, um para cada hora $h \forall h=1, \dots, 24$. Os padrões de entrada incluem as demandas horárias na hora h nos dias $D-1$ e $D-7$, bem como a temperatura prevista na hora h do dia D , caso a variável temperatura tenha sido incluída entre as variáveis explicativas.

Caso o dia D seja normal, porém os dias $D-1$ ou $D-7$ não sejam normais, uma variável explicativa é incluída contendo a informação sobre o tipo de feriado que assume valor 1 para feriado com perfil de sábado e 0 para feriado com perfil de domingo.

Se o dia D da previsão for um feriado/dia especial o PrevCargaDESSEM faz uma previsão por identificação de padrões, i.e., busca nos históricos de carga e temperatura um perfil de carga com características semelhantes às verificadas no momento da previsão.

No modelo de previsão do ONS é possível discretizar os dados de carga por meio de *splines* cúbicas em intervalos de meia hora, intervalos horários ou níveis variáveis. As previsões são feitas para programar a operação das usinas de geração para o dia seguinte, onde se adota a discretização de meia hora para obter um plano de operação de curtíssimo prazo e informações para estimar os preços horários de energia elétrica. A precisão dos modelos é avaliada pelas medidas de acurácia Erro Percentual Absoluto Médio (do inglês, *Mean Average Percentage Error* - MAPE) e Erro Quadrático Médio (do inglês, *Root Mean Square Error* - RMSE).

2.2 Trabalhos Relacionados no Brasil

No que diz respeito a literatura acadêmica relacionada ao tema de PCCP, é possível encontrar alguns estudos relacionados ao mercado brasileiro, que aplicam os modelos de aprendizagem de máquinas. Em Silva et al. [29] foi desenvolvida uma ANN do tipo *feedforward* para os próximos sete dias, em que os preditores são: mês; dias úteis; previsão de carga uma e duas horas atrás; variação de carga durante o dia; e uma métrica ponderada com as variáveis climáticas de temperatura, velocidade do vento e umidade relativa.

Silva et al. [30] compararam três modelos PCCP: um ARIMA e os modelos de Redes Neurais Recorrente (RNN) *Long Short Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU). Entre as informações de entrada consideradas no estudo estão: mês; dias de semana e feriados; crescimento do PIB; temperaturas regionais expressas em carga ponderada; fator de tendência para o crescimento populacional.

Ribeiro e al. [31] apresentaram uma RNN do tipo Echo State Network (ESN). É uma RNN com arquitetura simples, com uma camada escassamente oculta e com pesos sinápticos fixos e atribuídos aleatoriamente. A principal característica do ESN é que os únicos pesos fixos que são modificados no treinamento são as sinapses que conectam os neurônios da camada oculta com a camada de saída. No estudo, essa abordagem é utilizada para o PCCP na região sul do Brasil, utilizando os dados de carga para as primeiras quatro semanas.

Tanto na literatura quanto nos modelos do ONS, os modelos de *machine learning* estão consolidados. No entanto, os modelos RNN ainda não são utilizados pelo ONS, e as principais metodologias de processamento de dados são pouco exploradas. Apenas os algoritmos de otimização são usados para projetar o conjunto de dados de temperatura equivalente. Apesar de serem utilizados modelos de *machine learning*, ainda há oportunidades de uso de RNN e outras metodologias encontradas na literatura de PCCP.

2.3 Trabalhos Relacionados no Exterior

Diversos estudos relacionados a PCCP através de técnicas de *machine learning* também foram conduzidos em outros países. Kong et al [50] desenvolveram uma LSTM para previsão de carga residencial de curto prazo de um smart grid na Austrália, em que os dados de carga são inicialmente clusterizados para identificação dos padrões de carga. Em seguida, os dados pré-processados e sequenciados em diferentes passos temporais para o treinamento de diferentes configurações de LSTM.

Ghofrani et al. [53] propõem um ANN bayesiana para previsão de carga no curto prazo, e para a definição dos dados de entrada recorrem uma análise de correlação que calcula os coeficientes de correlação entre as entradas e saídas de treinamento, e ao cálculo da distância euclidiana pela norma ℓ_2 para os coeficientes de correlação. A distância euclidiana mínima é selecionada como a entrada mais correlacionada e decomposta por uma transformação wavelet para fornecer os dados de entrada para a previsão para as horas do dia seguinte.

Tian e Hao [54] apresentam um modelo de previsão de carga semi-horária para New South Wales também utilizando as SVM por regressão. Neste estudo os autores realizam o pré-processamento dos dados baseando-se em uma seleção longitudinal para eliminar ruídos e identificar as principais características dos dados de carga.

Zheng et al [51] propõem uma LSTM, em que antes do treinamento é aplicado um algoritmo de clusterização k-means baseado na ponderação para avaliação de dias similares, em seguida os autores aplicam o sequenciamento para aprendizagem da LSTM. Dentre as variáveis de entrada avaliadas estão temperatura, pico de carga no dia anterior, umidade, precipitação, velocidade de vento e as variáveis de calendário. As previsões são para a carga de curto prazo do sistema elétrico de New England para o intervalo de uma hora do dia seguinte.

Jin et al. [55] avaliam nos mercados de eletricidade da Espanha, Austrália e Nova Iorque (EUA) um modelo de previsão de carga no curto prazo para as horas do dia seguinte baseado em uma ANN do tipo MLP, no qual os dados de carga inicialmente são agrupados pelos mapas auto-organizáveis. Em seguida os agrupamentos são rotulados com base em janelas temporais, para posteriormente serem inseridos no treinamento da rede.

Jiao et al [52] apresentam uma LSTM para a previsão de carga não residencial na China, recorrendo a uma segmentação de padrões similares de carga bastante robusto. Inicialmente, o algoritmo de clusterização k-means é aplicado para classificar padrões de consumo de carga. Em seguida, o coeficiente de correlação de Spearman investiga a correlação entre as séries

temporais segmentadas, onde são identificadas múltiplas sequencias de séries temporais correlacionadas. Por fim, para utilizar sequencias com diferentes correlações e capturar as dependências temporais a longo prazo, os autores realizam as previsões pela LSTM com três passos temporais.

Kumar et al. [56] utilizam as LSTM e as GRU apenas utilizando os históricos de carga por aprendizado sequencial para previsão de carga de consumo doméstico para as horas seguintes. Bouktif et al. [58] também utilizam LSTM apoiadas por algoritmos genético para definir os parâmetros iniciais do modelo de previsão para a carga de meia em meia hora do dia seguinte com dados metropolitanos na França, dentre as variáveis de entrada analisadas estão carga, temperatura, variáveis de calendário, umidade e velocidade do vento.

CAPÍTULO 3

MODELOS DE PREVISÃO

3.1 Introdução

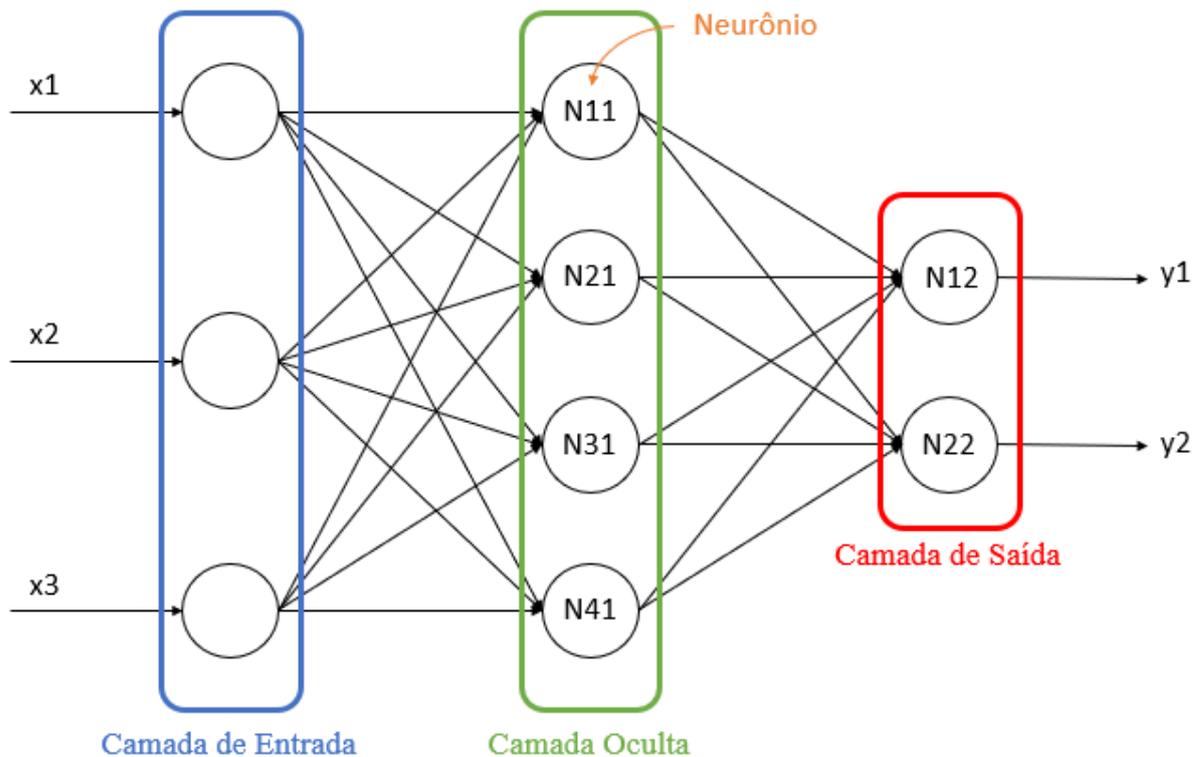
As ANNs têm sido utilizadas há décadas como modelos para realizar previsões de séries temporais, no entanto o desenvolvimento de novos métodos e hardwares trouxeram avanços significativos no estado da arte na literatura acadêmica nos últimos anos [32]. As ANNs são capazes de gerar bons resultados de previsão, sendo flexíveis em sua configuração e capazes de generalizar e capturar a não linearidade presente em problemas de previsão de carga [14]. Dada uma amostra com vetores de entrada e saída, as ANNs são capazes de mapear o relacionamento padrão entre os dados, assim o processo de aprendizado é baseado na otimização dos parâmetros das funções da rede.

Portanto, tornam-se bastante apropriados para grandes conjuntos de dados, mas com pouco conhecimento sobre as interações que produzem os resultados [24]. As redes neurais Multi-Layer Perceptron (MLP) baseadas em metodologias *feedforward* (onde não há *loops* da rede e as informações seguem uma mesma direção, partindo da camada de entrada rumo a camada de saída); Redes Neurais Recorrentes (RNN); e, em menor grau, a Rede Neural Convolucional (CNN) são usadas para realizar PCCP.

3.2 MLP: Multilayer Perceptron

A MLP é uma das arquiteturas mais importantes desenvolvidas na literatura de redes neurais. Ela está direta ou indiretamente associada a quase todas as outras arquiteturas e metodologias desenvolvidas nos tempos atuais. A Figura 4 mostra um exemplo de uma MLP com uma camada (*layer*) oculta.

Figura 4 – Exemplo geral de uma MLP rasa



Fonte: Autor

A arquitetura de uma MLP rasa contém basicamente três camadas: a camada de entrada (*input layer*); uma camada oculta (*hidden layer*) e uma camada de saída (*output layer*). A camada de entrada recebe as variáveis de entrada (inputs) do modelo, em seguida, esses inputs são processados pelos chamados neurônios existentes na camada oculta. Cada neurônio faz uma combinação linear da informação a qual ele tem acesso imediato.

No caso do exemplo da Figura 4, cada neurônio N presente na camada oculta faz uma combinação linear das variáveis de entrada X_1 , X_2 , e, X_3 então aplica uma função não linear ao resultado da sua combinação linear. Essa função não linear é chamada função de ativação (ϕ). Existem diferentes tipos de funções de ativação, tais como: sigmoid, tanh, ReLU, etc. Cada uma dessas funções tem um objetivo específico e normalmente influenciam significativamente o resultado do modelo. Na camada de saída são realizadas as combinações lineares a partir dos resultados dos neurônios da camada oculta. A função de ativação é então aplicada novamente em cada neurônio e a previsão do modelo é obtida. Os cálculos matemáticos referentes ao exemplo da Figura 4 são representados pelas equações (1) – (12).

Normalmente agrupam-se exemplos de conjuntos input e output de forma vetorizada como em (1)-(2), onde nas colunas tem-se os exemplos de 1 até m que são compostos pelas linhas dos inputs (vetor X) e dos outputs (vetor Y).

$$X = \begin{bmatrix} x_1^1 & \dots & x_1^m \\ x_2^1 & \dots & x_2^m \\ x_3^1 & \dots & x_3^m \end{bmatrix} \quad (1)$$

$$Y = \begin{bmatrix} y_1^1 & \dots & y_1^m \\ y_2^1 & \dots & y_2^m \end{bmatrix} \quad (2)$$

Para a camada oculta e a camada de saída atribui-se variáveis que são ajustadas no processo de treinamento da rede. Para a camada oculta tem-se (3)-(4) e para a camada de saída tem-se (5)-(6).

$$W^{[1]} = \begin{bmatrix} w_{11}^{[1]} & w_{12}^{[1]} & w_{13}^{[1]} & w_{14}^{[1]} \\ w_{21}^{[1]} & w_{22}^{[1]} & w_{23}^{[1]} & w_{24}^{[1]} \\ w_{31}^{[1]} & w_{32}^{[1]} & w_{33}^{[1]} & w_{34}^{[1]} \end{bmatrix} \quad (3)$$

$$b^{[1]} = \begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \\ b_3^{[1]} \\ b_4^{[1]} \end{bmatrix} \quad (4)$$

$$W^{[2]} = \begin{bmatrix} w_{11}^{[2]} & w_{12}^{[2]} \\ w_{21}^{[2]} & w_{22}^{[2]} \\ w_{31}^{[2]} & w_{32}^{[2]} \\ w_{41}^{[2]} & w_{42}^{[2]} \end{bmatrix} \quad (5)$$

$$b^{[2]} = \begin{bmatrix} b_1^{[2]} \\ b_2^{[2]} \end{bmatrix} \quad (6)$$

Finalmente, tem-se o cálculo da previsão \hat{Y} dado um vetor X , (7)-(12). Onde as variáveis $Z^{[l]}$ correspondem aos resultados das combinações lineares, e as variáveis $A^{[l]}$ correspondem a valores após a aplicação de uma função não linear.

$$Z^{[1]} = (W^{[1]})^T X + b^{[1]} \quad (7)$$

$$A^{[1]} = \phi(Z^{[1]}) = \phi\left[(W^{[1]})^T X + b^{[1]}\right] \quad (8)$$

$$Z^{[2]} = (W^{[2]})^T A^{[1]} + b^{[2]} \quad (9)$$

$$A^{[2]} = \phi(Z^{[2]}) = \phi\left[(W^{[2]})^T A^{[1]} + b^{[2]}\right] \quad (10)$$

$$\hat{Y} = A^{[2]} \quad (11)$$

$$\hat{Y} = \begin{bmatrix} \hat{y}_1^1 & \dots & \hat{y}_1^m \\ \hat{y}_2^1 & \dots & \hat{y}_2^m \end{bmatrix} \quad (12)$$

O processo de treinamento de uma MLP consiste na comparação entre os valores estimados pela rede (\hat{Y}) e os valores observados (Y) e atualização dos pesos (W) e viés (b). Esse processo de atualização é feito através do cálculo do gradiente descendente por meio de uma técnica chamada *backpropagation* [59]. Essa técnica consiste simplificada no cálculo de derivadas parciais, onde o erro do modelo ($\hat{Y} - Y$) é derivado em função de cada peso W e viés b . Esse cálculo é realizado a partir da camada de saída em direção a camada de entrada por meio da regra da cadeia.

Os resultados dessas derivadas são então multiplicados por uma taxa de aprendizagem (*learning rate*) definida arbitrariamente e os resultados são somados aos respectivos W e b . Esse processo se repete por um número específico de interações ou limitado por uma função objetivo (quando o erro da previsão do conjunto de teste atinge um valor mínimo pré-definido).

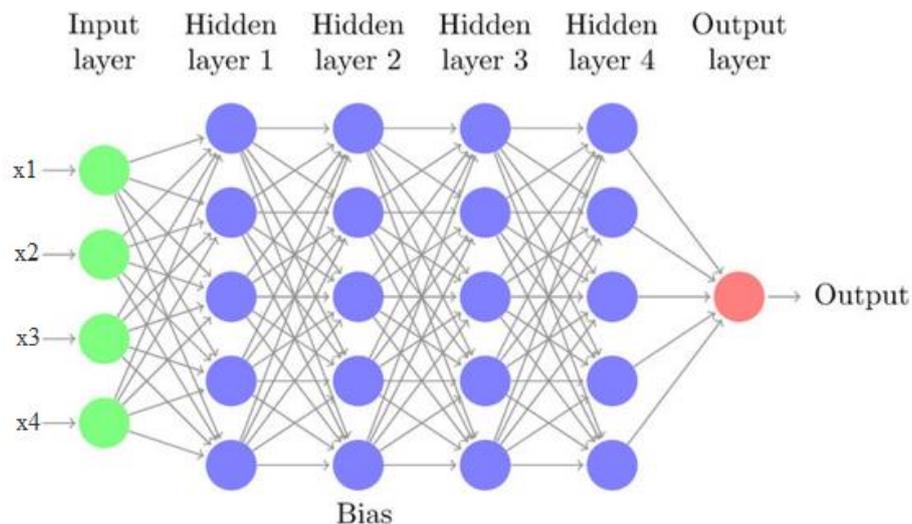
3.3 DNN: Deep Neural Networks

A disponibilidade de grandes conjuntos de dados (*Big Data*) e computadores com hardwares mais poderosos tornaram-se realidade desde os anos 2000. Assim, tornou-se possível aprimorar os modelos de ANN, que tem se tornado bem-sucedidos quando um grande e confiável conjunto de dados está disponível para o treinamento de modelos.

Uma das estruturas que surgiram com esta evolução se trata das Deep Neural Networks (DNN), ilustradas na Figura 5. As arquiteturas de DNN tem um grande número de camadas ocultas quando comparadas com as MLP rasas (com uma camada oculta). As estruturas do tipo MLP podem ser treinados como DNN, o que em muitos casos pode ser benéfico, pois proporciona uma melhor abstração dos parâmetros da rede e, portanto, uma melhor representação das relações não lineares entre entradas e saídas do problema. A grande

desvantagem da aplicação de DNN está no esforço computacional que deverá ser alocado para se trabalhar com esse tipo de rede.

Figura 5 – Exemplo de DNN.



Fonte: Adaptado de [44]

Assim como muitos softwares e sistemas de engenharia, as DNNs são complexas demais para serem totalmente otimizadas por seres humanos. Para elaborar essas estruturas, os humanos são responsáveis pelo design de alto nível, e os detalhes referentes a otimização dos parâmetros das DNNs são deixados para os sistemas computacionais.

A construção de uma DNN inclui alguns desafios, tais como: projetar os componentes da arquitetura, como colocá-los juntos em uma topologia de rede completa; a definição dos hiperparâmetros para os componentes; e o design global. Esses aspectos precisam ser otimizados separadamente à cada tarefa.

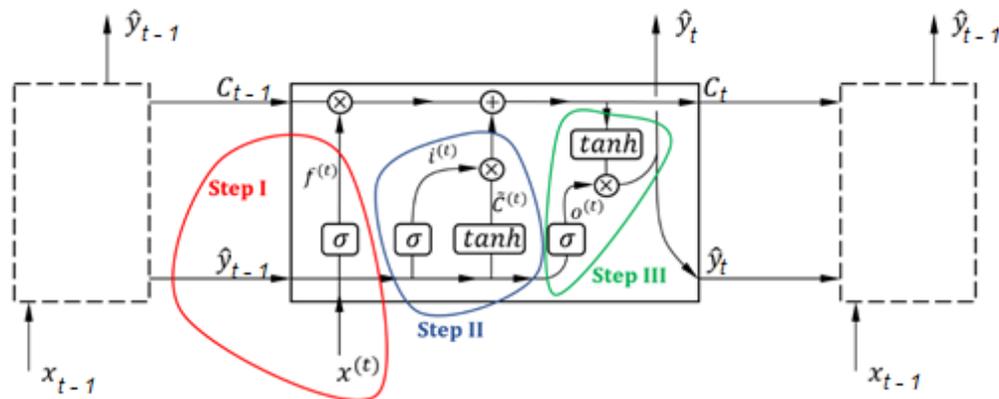
3.4 LSTM: Long Short-Term Memory

A *Long Short Term Memory* (LSTM) é um tipo de rede neural recorrente (RNN). Essas redes foram desenvolvidas especialmente para tarefas onde a dependência temporal se estende por vários estágios de tempo.

As RNNs possuem mecanismos de memorização, que permitem que o modelo entenda como os estágios de tempo observados anteriormente afetam os estágios seguintes. Nas LSTMs os neurônios são chamados de células e cada etapa de tempo passada é chamada de *time-step*.

A LSTM faz uso de três portões (*gates*): portão de esquecimento (*forget gate*), onde as informações julgadas como não úteis no estado da célula são removidas; portão de entrada (*input gate*), onde ocorre a adição de novas informações ao estado da célula; e o portão de saída (*output gate*), onde as informações úteis são extraídas do estado atual da célula e apresentadas como saída. A Figura 6 ilustra as principais componentes de uma célula LSTM.

Figura 6 – Funcionamento de uma LSTM.



A célula LSTM recebe como entrada o conjunto de variáveis referente ao estágio atual ($x^{(t)}$), a saída do último estágio oculto ($\hat{y}^{(t-1)}$) e o estado da célula no estágio anterior ($C^{(t-1)}$) (o estágio anterior se refere ao intervalo de tempo anterior). O estado da célula é responsável por carregar informação relevante durante todo o processamento, armazenando informações de muitos estágios atrás até o último estágio.

A primeira etapa (*Step I*) realizada nessa rede acontece na chamada camada de esquecimento. Ela é responsável por decidir qual informação do estágio anterior da rede ($C^{(t-1)}$) será mantida no estágio t e qual será eliminada. Na camada de esquecimento a operação (13) é realizada onde os parâmetros W_f e b_f são ajustados para definir a variável $f^{(t)}$; essa variável pode assumir valores entre 0 e 1, uma vez que passa por uma função sigmoide representada por σ , e desse modo, ao ser multiplicada pelo estado anterior da rede $C^{(t-1)}$, define o quanto de $C^{(t-1)}$ é mantido no estágio t . Quanto mais próximo a 1 mais informações do estágio anterior serão utilizadas no estágio atual.

$$f^{(t)} = \sigma (W_f \cdot [\hat{y}^{(t-1)}, x^{(t)}] + b_f) \quad (13)$$

A segunda etapa mostrada na Figura 6 (*Step II*) define qual nova informação ($\tilde{C}^{(t)}$) será armazenada no estado atual da célula $C^{(t)}$. Para tanto, essa etapa é dividida em duas operações

(14 e 15). A operação (14) é responsável por criar os candidatos ao estado atual $\tilde{C}^{(t)}$, onde operação W_c e b_c são parâmetros internos da rede que ponderam a combinação linear dos parâmetros $\hat{y}^{(t-1)}$ e $x^{(t)}$ e estabelece valores entre -1 e 1, passando por uma função tangente hiperbólica (\tanh). Em (15) é feito o cálculo de $i^{(t)}$, de maneira semelhante a (13), que assume valor entre 0 e 1, o qual será multiplicado por $\tilde{C}^{(t)}$, determinando quais inputs sugeridos em $\tilde{C}^{(t)}$ serão efetivamente considerados para definir $C^{(t)}$. Na terminologia de redes neurais essa camada de operação se chama portão de entrada.

$$\tilde{C}^{(t)} = \tanh(W_c \cdot [\hat{y}^{(t-1)}, x^{(t)}] + b_c) \quad (14)$$

$$i^{(t)} = \sigma(W_i \cdot [\hat{y}^{(t-1)}, x^{(t)}] + b_i) \quad (15)$$

Concluída a etapa II (*Step II*) podemos finalmente calcular $C^{(t)}$, como em (16) e prosseguir para o cálculo da variável de saída $\hat{y}^{(t)}$ na etapa III (portão de saída). Nessa última etapa, $C^{(t)}$ passa pela função \tanh para reassumir valores entre -1 e 1, intervalo originalmente observado em $C^{(t-1)}$. O valor resultante dessa operação é multiplicado pela variável $o^{(t)}$, calculada no portão de saída (17), responsável por definir qual informação do estado $C^{(t)}$ será efetivamente utilizado para compor a variável de saída $\hat{y}^{(t)}$ (18).

$$C^{(t)} = f^{(t)} \cdot C^{(t-1)} + i^{(t)} \cdot \tilde{C}^{(t)} \quad (16)$$

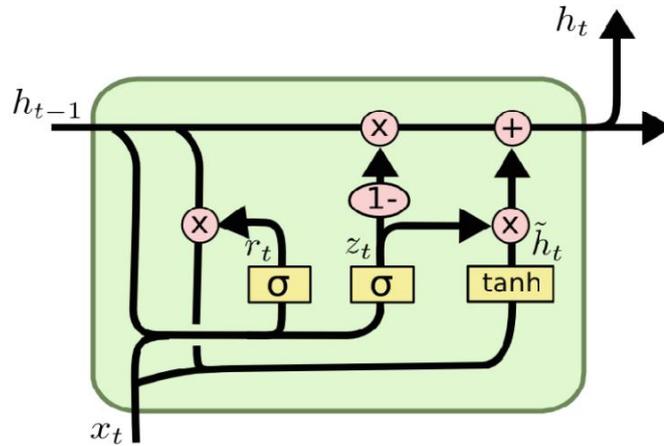
$$o^{(t)} = \sigma(W_o \cdot [\hat{y}^{(t-1)}, x^{(t)}] + b_o) \quad (17)$$

$$\hat{y}^{(t)} = o^{(t)} \cdot \tanh(C^{(t)}) \quad (18)$$

3.5 GRU: Gated Recurrent Unit

A rede GRU é um tipo de RNN bastante semelhante a LSTM, entretanto sem utilizar a camada de esquecimento. A arquitetura de uma GRU é constituída de apenas dois estágios, chamados de portão de atualização z_t (*update gate*) e portão de redefinição r_t (*reset gate*), que são dois vetores que decidem quais informações devem ser passadas para a saída. A Figura 7 ilustra a arquitetura de uma GRU.

Figura 7 – Arquitetura de uma GRU.



Fonte: [45]

Enquanto as LSTMs têm dois estados diferentes passados entre as células – o estado da célula ($C^{(t-1)}$) e o estado oculto ($\hat{y}^{(t-1)}$), que carregam a memória de longo e curto prazo, respectivamente – as GRUs têm apenas um estado oculto ($\hat{y}^{(t-1)} = h^{(t-1)}$) transferido entre as etapas de tempo. Este estado oculto é capaz de manter as dependências de longo e curto prazo ao mesmo tempo, uma vez que a GRU não possui a etapa descrita pela equação (13), onde a informação do estágio anterior é ponderada, podendo excluir informações de muitos estágios atrás. Portanto, ao contrário da LSTM, a GRU pode ser treinada para manter informações de muito tempo atrás sem dissipá-las com o tempo ou remover informações irrelevantes para a previsão.

Os cálculos realizados em uma célula GRU são divididos em duas etapas. Na primeira etapa são calculados r_t e z_t usando o estado oculto do estágio anterior e os dados de entrada do estágio atual, como mostram as equações (19 e 20).

$$z^t = \sigma(W_z \cdot [h^{(t-1)}, x^{(t)}] + b_z) \quad (19)$$

$$r^t = \sigma(W_r \cdot [h^{(t-1)}, x^{(t)}] + b_r) \quad (20)$$

onde: $x^{(t)}$, é o conjunto de variáveis referente ao estágio atual t ; $h^{(t-1)}$ é o estado oculto do estágio anterior; σ é a função sigmoide; W é a matriz de pesos e b o vetor de viés, sendo uma matriz e um vetor de viés diferentes para cada portão.

A segunda etapa consiste em calcular os candidatos a saída e os valores de saída da rede:

$$\tilde{h}^t = \tanh(W_h \cdot [r_t \cdot h^{(t-1)}, x^{(t)}] + b_h) \quad (21)$$

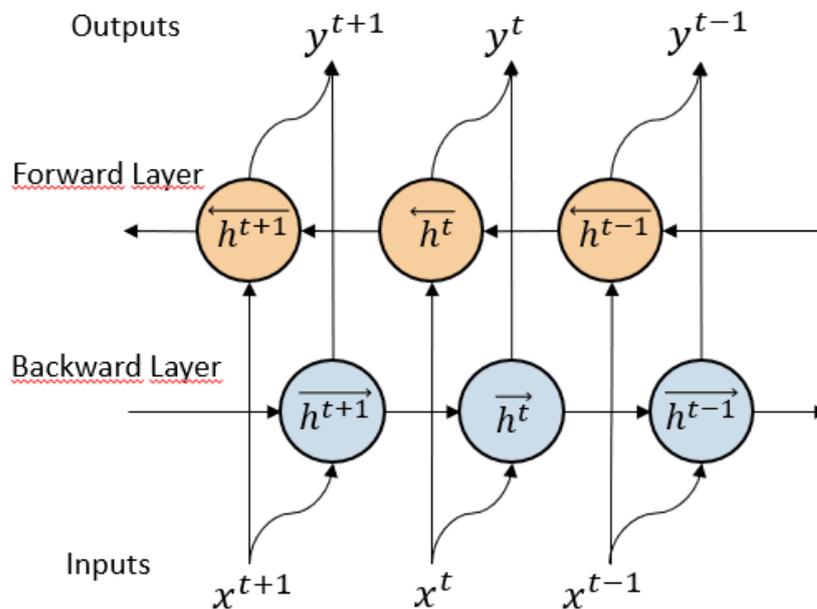
$$y^t = (1 - z^t) \cdot h^{(t-1)} + z^t \cdot \tilde{h}^t \quad (22)$$

onde y^t é o vetor de saída, \tilde{h}^t são os candidatos a vetor de saída.

3.6 Modelos Bidirecionais: BI-LSTM E BI-GRU

Além das estruturas de RNN que possuem uma estrutura de fluxo unidirecional, existem também os modelos que são caracterizados por uma estrutura de fluxo de dados bidirecional, também conhecidos como RNN bidirecionais (bi-RNN). Uma das limitações das redes com fluxo unidirecional, é fazer somente uso do contexto anterior dos dados. Já as estruturas com fluxo bidirecional processam os dados em duas direções, tanto em *forward*, como em *backward*, em diferentes estágios. Um caso geral de uma bi-RNN está ilustrado na Figura 8.

Figura 8 – RNN bidirecional



Fonte: [57]

Para um conjunto de entrada em que $1 \leq t \leq T$, a etapa em *forward* receberá os dados de $t = 1$ a $t = T$ e a saída da célula em *forward* é denotada por \overleftarrow{h}^t , enquanto a etapa em *backward*

receberá os dados de $t = T$ a $t = 1$ e a saída da célula em *backward* é denotada por \overleftarrow{h}^t (equações 23 e 24).

$$\overrightarrow{h}^t = \tanh(W_{\overrightarrow{h}} \cdot [\overrightarrow{h}^{(t-1)}, x^{(t)}] + b_{\overrightarrow{h}}) \quad (23)$$

onde: $W_{\overrightarrow{h}}$ é a matriz de pesos de saída em *forward* e $b_{\overrightarrow{h}}$ corresponde ao vetor de viés de saída em *forward*.

$$\overleftarrow{h}^t = \tanh(W_{\overleftarrow{h}} \cdot [\overleftarrow{h}^{(t-1)}, x^{(t)}] + b_{\overleftarrow{h}}) \quad (24)$$

onde: $W_{\overleftarrow{h}}$ é a matriz de pesos de saída em *backward* e $b_{\overleftarrow{h}}$ corresponde ao vetor de viés de saída em *backward*.

Com as saídas da célula para cada sentido de fluxo é possível então calcular a saída final y^t através de (25).

$$y^t = \sigma(W_y \cdot [\overrightarrow{h}^t, \overleftarrow{h}^t] + b_y) \quad (25)$$

onde: W_y é a matriz de pesos de saída e b_y é o vetor viés correspondente a saída.

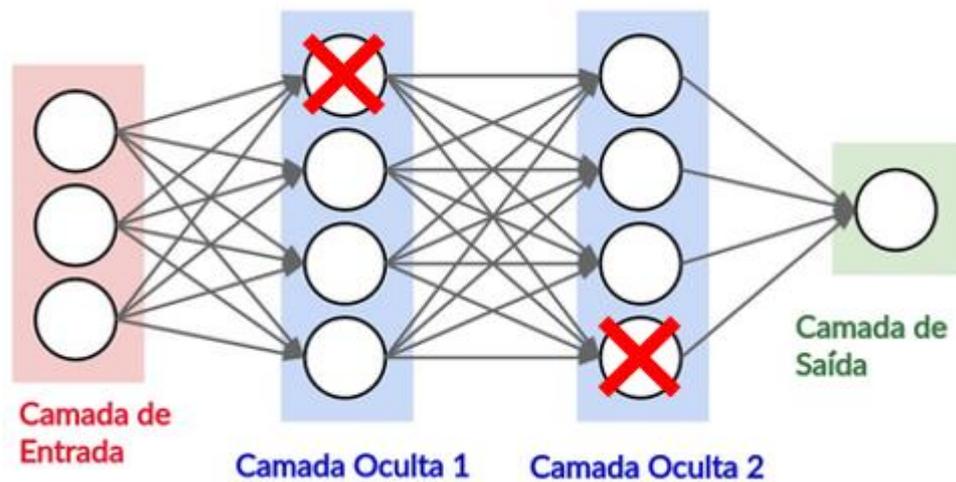
A combinação da estrutura de fluxo bidirecional com as redes LSTM e GRU geram os modelos bidirecionais bi-LSTM e bi-GRU.

3.7 Técnicas Utilizadas para Melhorar os Modelos

Um aspecto importante na definição de um modelo de rede neural diz respeito a técnica de regularização utilizada. Quando capacitada com um grande número de parâmetros (neurônios e camadas ocultas), as redes neurais têm normalmente alta capacidade de memorização, entretanto não é desejável que o modelo memorize totalmente o conjunto de treinamento. O ideal é que o modelo consiga generalizar o processo observado no treinamento e que seja capaz de não apenas explicar o que aconteceu no passado, mas também de criar boas extrapolações em face a eventos que ela não observou ainda. Dessa forma, técnicas de regularização foram criadas para aprimorar a capacidade de generalização do modelo.

Apesar da literatura propor múltiplas maneiras de se regularizar uma ANN, uma das técnicas mais utilizadas conhecida como *Dropout*, onde uma parte dos neurônios da rede são aleatoriamente desligados/ligados durante seu treinamento, conforme ilustrado na Figura 9. Esse processo induz o modelo a “acreditar menos” na saída de cada neurônio individualizado, e fazer com que a rede como um todo entenda melhor o processo observado.

Figura 9 - Metodologia Dropout



Fonte: Adaptado de [43]

O Monte Carlo *Dropout* proposto por Gal e Ghahramani [33] é uma percepção do uso do *Dropout* como uma aproximação de um modelo probabilístico caracterizado pelo processo gaussiano. Assim, podemos tratar muitas redes diferentes com neurônios descartados como amostras de Monte Carlo, fornecendo bases matemáticas para incorporar as incertezas do modelo, e melhorando seu desempenho.

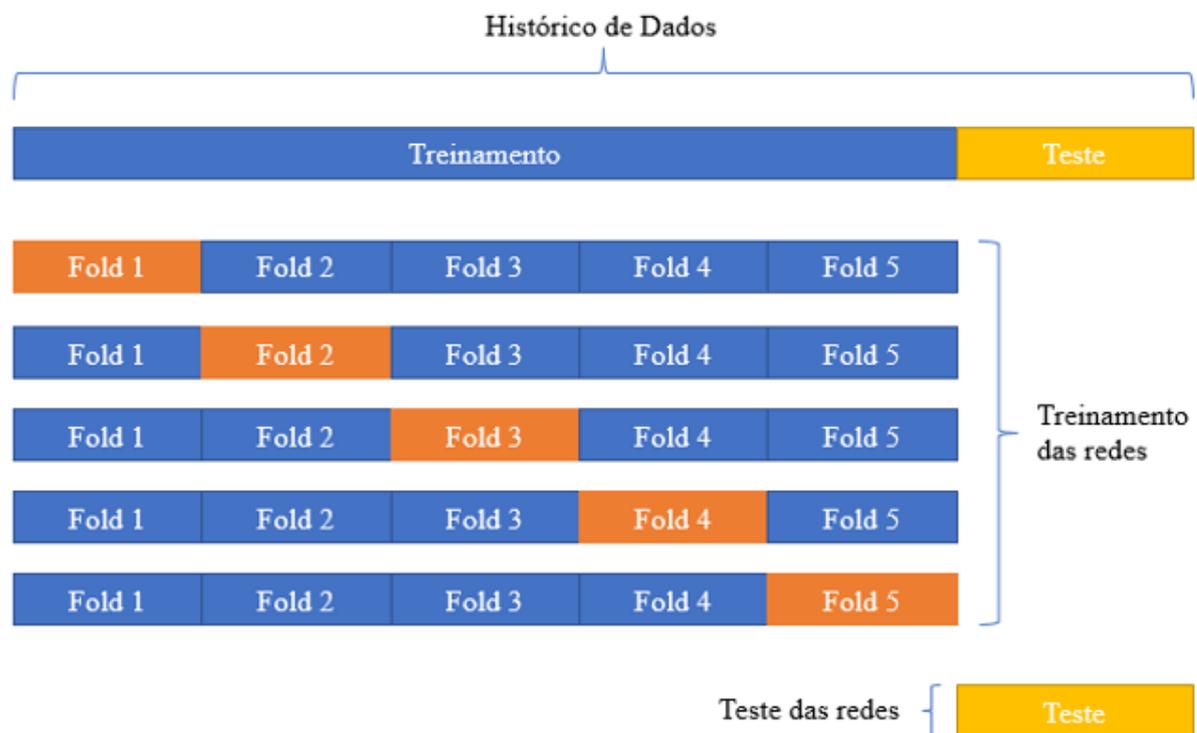
O funcionamento do Monte Carlo *Dropout* se baseia na aplicação do *Dropout* no momento do teste. Assim, em vez de uma previsão, obtém-se múltiplas previsões, cada uma com um conjunto diferente de neurônios “desligados”. Portanto, para um mesmo conjunto de variáveis de entrada, uma mesma rede treinada será capaz de fornecer diversas previsões diferentes. Analisando a variação nos valores dessas previsões é possível verificar a “certeza” da rede em relação a previsão. O valor final do modelo pode ser obtido calculando a média geométrica dos valores obtidos em cada previsão.

Outra técnica utilizada para melhorar a performance do modelo foi a validação cruzada *k-fold*. Essa técnica consiste em dividir o conjunto de dados utilizado para treinamento da ANN em k subconjuntos mutuamente exclusivos do mesmo tamanho e, a partir daí, um subconjunto é utilizado para teste e os $k-1$ restantes são utilizados para estimação dos parâmetros (treinamento do modelo). Este processo é realizado k vezes alternando os subconjuntos de treino e de teste. Ao final, k modelos diferentes foram treinadas e testadas com subconjuntos diferentes de dados de treino e teste. Uma estimativa de precisão confiável terá uma variância

relativamente pequena entre as previsões de cada modelo k , portanto, vários estudos sugeriram realizar repetidamente a validação cruzada k -fold ([46] e [47]).

A previsão é realizada por cada uma das redes e o resultado final é obtido calculando a média geométrica das previsões. A Figura 10 mostra o esquema realizado pela validação cruzada considerando 5 *folds*.

Figura 10 - Validação Cruzada k -fold



Fonte: Autor

3.8 Métricas De Acurácia

As métricas de acurácia são informações importantes para comparar o desempenho de modelos de previsão e decidir se representam um bom ajuste dos dados. Para avaliar os modelos de ANN desenvolvidos foram utilizadas as métricas *Mean Average Percentage Error* (MAPE), e o coeficiente *Nash-Sutcliffe Efficiency* (NSE), apresentados a seguir:

3.8.1 Mean Average Percentage Error

O *Mean Average Percentage Error* (MAPE) é uma métrica bastante popular de estimativa de desvios de previsão, caracterizada pela média percentual dos desvios absolutos

das previsões. A medida é representada por um percentual, utilizando o valor observado como divisor, conforme descrito na Equação 26. No entanto, têm a desvantagem de serem infinitas ou indefinidas se $y_i = 0$, ou se possuir tendência a valores extremos, quando qualquer y_i está próximo de zero.

$$MAPE = \frac{1}{N} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (26)$$

onde: y_i é o valor verificado no período i e \hat{y}_i é o valor da previsão para o período i

3.8.2 Coeficiente *Nash-Sutcliffe Efficiency*

O coeficiente de *Nash-Sutcliffe Efficiency* (NSE), proposto por Nash e Sutcliffe (1970), representa o somatório de erros quadráticos na forma padronizada. Assim, é comparado se a variância do modelo em relação ao valor observado é inferior a variância da média em relação ao observado. O resultado pode variar entre 1 (um) até menos infinito, sendo que o resultado mais próximo de 1, indica que o modelo é um bom preditor. A Equação 27 ilustra o cálculo para o coeficiente de NSE, sendo válido destacar que ao elevar os desvios ao quadrado, o coeficiente tende a penalizar os grandes desvios ocorridos.

$$NSE = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (27)$$

onde: y_i é o valor observado em cada período i ; \hat{y}_i é o valor previsto em cada período i ; \bar{y} é a média dos valores observados.

3.8.3 Teste de Diebold-Mariano

As medidas tradicionais de erro podem levar a conclusões equivocadas sobre o desempenho dos modelos, quando se tem a realização de apenas um processo estocástico. Nesses casos, pelo fato de um mesmo conjunto de dados ser usado mais de uma vez para as inferências e seleção de modelos, um modelo pode apresentar desempenho superior a outro muito mais por acaso, do que por melhor poder preditivo.

Esse problema é denominado como *data snooping* ([34] e [35]), e é bastante comum quando se tem apenas uma série histórica de dados ou a ausência de técnicas que realizam mais de um procedimento estocástico. Apesar de o *data snooping* já ser um problema conhecido na

literatura acadêmica, as métricas tradicionais de erro ainda são frequentemente utilizadas para a comparação de modelos.

Para contornar esse problema, Diebold e Mariano [36] apresentam um teste de acurácia capaz de validar estatisticamente o desempenho de um determinado modelo em relação a um modelo utilizado como benchmark. O detalhamento do teste de Diebold-Mariano é apresentado a seguir:

Assumindo que os valores reais observados são $\{y_i; i = 1, \dots, n + m\}$, e as previsões a partir dos modelos que serão comparados iguais a $\{\hat{y}_i^{(a)}; i = 1, \dots, n + m\}$ e $\{\hat{y}_i^{(b)}; i = 1, \dots, n + m\}$, os erros de previsão dos dois modelos serão:

$$e_{n+1}^{(a)} = y_{n+1} - \hat{y}_{n+1}^{(a)}, l = 1, 2, \dots, m \quad (28)$$

$$e_{n+1}^{(b)} = y_{n+1} - \hat{y}_{n+1}^{(b)}, l = 1, 2, \dots, m \quad (29)$$

Assim, uma função de perda $F(e_{n+1}^{(j)})$, $j = a, b$ é usada para comparar a acurácia de previsão entre os dois modelos (Equações 30 e 31):

$$F(e_{n+1}^{(j)}) = (e_{n+1}^{(j)})^2 \quad (30)$$

$$F(e_{n+1}^{(j)}) = |e_{n+1}^{(j)}|^2 \quad (31)$$

Por sua vez, o valor estatístico do teste de Diebold-Mariano é calculado por:

$$DM = \frac{\sum_{i=1}^m (F(e_{n+1}^{(a)}) - F(e_{n+1}^{(b)})) / m}{\sqrt{S^2 / m}} \quad (32)$$

onde: S^2 é o estimador consistente de variância assintótica de $d_1 = F(e_{n+1}^{(a)}) - F(e_{n+1}^{(b)})$.

O teste de hipótese de Diebold-Mariano pode ser descrito como:

- H_0 = a função de perda dos modelos gera previsões que não são estatisticamente diferentes ($E(d_1) = 0$).

Por sua vez, as hipóteses alternativas podem ser dadas de duas formas:

- $H_a = E(d_1) > 0$, onde o modelo a possui melhor performance de previsão do que b ;
- $H_a = E(d_1) < 0$, onde o modelo b possui melhor performance de previsão do que a .

As estatísticas do teste de Diebold-Mariano seguem uma distribuição normal padronizada. Portanto, a hipótese nula será rejeitada quando o valor estatístico do teste satisfizer o critério expresso pela Equação 33:

$$|DM| > z_{\alpha/2} \quad (33)$$

onde: $z_{\alpha/2}$ = expressa o valor de z para um nível de significância α .

Harvey, Leybourne e Newbold [37] sugerem que melhores propriedades para pequenas amostras podem ser obtidas, corrigindo o viés para a estatística de teste de Diebold-Mariano e comparando a estatística corrigida com uma distribuição *t-Student* com $(T - 1)$ graus de liberdade, ao invés da distribuição normal padronizada. Assim, o valor estatístico de Diebold-Mariano com a modificação de Harvey, Leybourne e Newbold (HLN), utilizada nas comparações entre os resultados dos modelos avaliados neste projeto, é dado por:

$$DM_{HNL} = DM \sqrt{\frac{T + 1 - 2h + h(h - 1)}{T}} \quad (34)$$

onde: DM_{HNL} é o valor estatístico do teste de Diebold-Mariano com a modificação HLN; h é o horizonte de previsão; T é o tamanho da amostra para a estimativa dos erros.

CAPÍTULO 4

METODOLOGIA

4.1 Dados Considerados Para Treinamento e Teste

Um requisito essencial para que se consiga um bom resultado de previsão utilizando modelos de *machine learning* é o tratamento do conjunto de dados. Estes dados devem ser passados para as redes neurais de forma normalizada, após realizar a correção de outliers e dados faltantes. Neste trabalho os modelos de ANNs foram treinados utilizando os dados de carga e temperatura (histórica e prevista) obtidos na plataforma SINtegre do ONS [27]. Posteriormente foram testados os dados de temperatura prevista do Global Ensemble Forecast System (GEFS). Além disso, fez-se uso da técnica *One Hot Encoding* para codificar os dados de calendário, uma vez que se tratam de variáveis categóricas.

4.1.1 Dados do SINtegre

O ONS disponibiliza diariamente os Decks de arquivos utilizados para previsão de carga de curto prazo realizada pelo modelo PrevCargaDESSEM. Esses decks são separados por submercado e contém informações de carga e temperatura verificada desde 01/01/2016, a temperatura prevista para até 7 dias a frente e informações de feriados e dias especiais.

Os arquivos de carga e temperatura são compostos pelas seguintes colunas: ‘ano’, ‘mês’, ‘dia’, ‘hora’, ‘minuto’ e, o valor de carga ou temperatura (histórica ou prevista). Por sua vez, o arquivo com as informações de feriados e dias especiais estão dispostos com quatro colunas, entre elas: ‘ano’, ‘mês’, ‘dia’, e o tipo de feriado (variável categórica).

Posteriormente, foi realizado o tratamento desses dados. O primeiro passo foi plotar graficamente cada variável, em busca de *outliers*. Os *outliers* observados foram substituídos pelos valores observados na hora anterior. Com os *outliers* tratados, o passo seguinte foi verificar a ocorrência de dados faltantes nos arquivos. Nas observações em que constaram dados faltantes foi feita uma interpolação linear para preencher lacunas. Por último, fez-se a normalização dos dados, subtraindo o cada valor pelo menor valor encontrado no período e dividindo o resultado pelo valor do desvio padrão.

Também são disponibilizados no SINtegre os arquivos contendo os resultados das previsões do modelo PrevCargaDESSEM para cada submercado. Estas previsões estão disponíveis a partir de março de 2021 e são utilizadas para comparação.

4.1.2 Dados do GEFS

O *Global Ensemble Forecast System* (GEFS) é um modelo climático global, desenvolvido pelo *National Centers for Environmental Prediction* (NCEP), que gera 21 previsões separadas a fim de contornar as incertezas e as limitações inerentes do próprio modelo. Estas incertezas são quantificadas para produzir uma gama de resultados potenciais. Os dados do GEFS estão disponíveis de duas formas: valores históricos de previsão para o período de 2000 a 2019 e previsões geradas diariamente que ficam armazenadas por um período de sete dias e depois são descartadas. As previsões do GEFS são realizadas para um horizonte de dez dias a frente, discretizadas de três em três horas.

Utilizando o conjunto histórico de dados, foram realizados testes para avaliar o desempenho das redes com as informações de temperatura prevista do GEFS, em substituição aos dados de temperatura prevista coletados no SINtegre. Os testes foram realizados para o submercado Sudeste/Centro-Oeste, que se trata do submercado com maior de demanda de energia do país.

Como as previsões do GEFS são realizadas por pontos de grade para todo o globo com resolução de 0,5 graus, para se obter uma temperatura equivalente para o submercado foi realizada uma ponderação da temperatura prevista a partir dos dados de população das vinte cidades mais populosas de cada estado que compõem o submercado. Essa abordagem é similar a apresentada em [38] e [39] para tratamento e composição de dados de temperatura para regiões agregadas buscando representar o efeito da população (variável importante para definição da carga de uma cidade específica) no cálculo da média ponderada desse preditor. Assim, os dados de informações de temperatura prevista de cada cidade foram coletados na base do GEFS informando as coordenadas geográficas de cada cidade, sendo que as populações de cada cidade até o ano de 2020 foram extraídas diretamente do website do Instituto Brasileiro de Geografia e Estatística (IBGE) [41].

Para efeito de avaliação das previsões de temperatura equivalente estimada a partir dos dados do GEFS, plotou-se um gráfico que compara os valores de temperatura verificada obtida no SINtegre e os valores de temperatura prevista do GEFS ponderados pelos dados de

população. Conforme observa-se na Onda T_{GEFS} é o valor previsto pelo modelo GEFS e $T_{GEFS.Final}$ é o valor obtido e considerado nos modelos após aplicação da regressão.

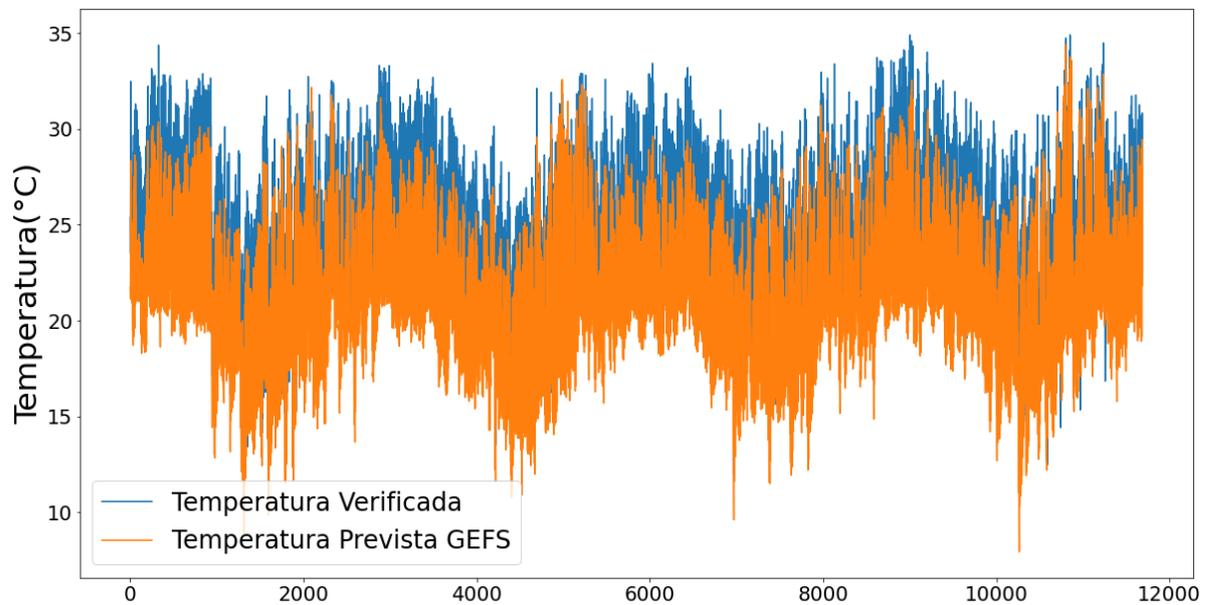
Figura 11, foi notada a presença de um viés entre as duas séries de dados. Com isso, recorreu-se ao uso de uma regressão linear com a finalidade de corrigir as previsões do modelo climático para um resultado mais próximo dos dados disponibilizados na plataforma do SINtegre. Essa abordagem havia sido aplicada anteriormente em [40] para correção de viés nas informações de modelos climáticos. Na Figura 12 é possível observar a comparação das temperaturas verificadas e a temperatura equivalente após a aplicação da regressão linear descrita pela Equação (35), na qual o coeficiente R2 encontrado foi de 0,874.

Assim, as temperaturas equivalentes para o submercado foram utilizadas na etapa de treinamento das ANNs no lugar da temperatura prevista obtida diretamente no portal do SINtegre.

$$T_{GEFS.Final} = 0,9717 * T_{GEFS} + 3,5416 \quad (35)$$

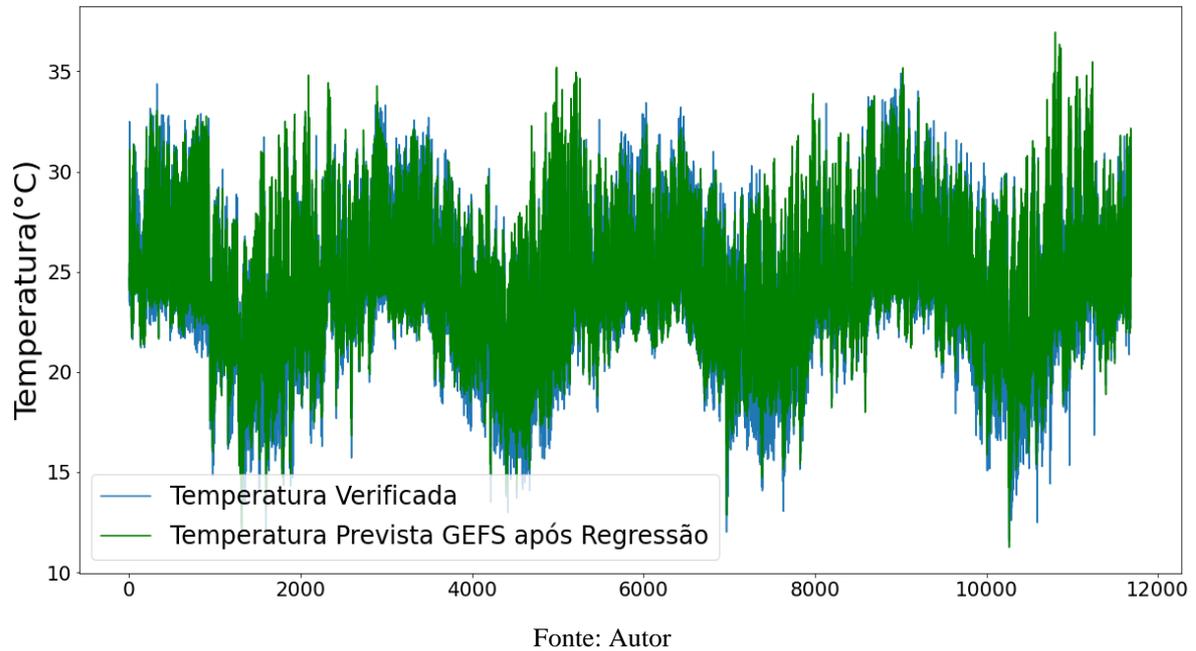
Onde T_{GEFS} é o valor previsto pelo modelo GEFS e $T_{GEFS.Final}$ é o valor obtido e considerado nos modelos após aplicação da regressão.

Figura 11 - Comparação da Temperatura equivalente verificada e a temperatura prevista pelo GEFS



Fonte: Autor

Figura 12 - Comparação da Temperatura equivalente verificada e a temperatura prevista pelo GEFS após aplicação de regressão



4.1.3 Montagem Dos Inputs Para as ANNs e Simulações

A técnica *one hot encoding* se baseia na transformação de uma variável categórica para uma forma binária. As variáveis de calendário utilizadas como input para as ANNs foram o dia da semana, dia do mês e o mês. Cada variável é representada por um vetor contendo zeros e uns. O tamanho do vetor é a quantidade de possibilidades que a variável por atingir. A variável dia da semana, por exemplo, possui 7 possibilidades, portanto será representada por um vetor com 7 posições. Se o dia em questão for uma segunda feira, a primeira posição do vetor assumirá o valor 1 e as outras 0. Caso seja uma terça feira, a segunda posição será igual a 1 e todas as outras 0 e assim por diante. As variáveis mês e dia do mês serão descritas por vetores com 12 e 31 posições respectivamente, totalizando 50 variáveis tipo *dummy* (binárias) representando a informação de calendário. Caso o dia de previsão seja um feriado, a variável de calendário será passada como sendo um domingo (onde apenas a sétima posição do vetor de dia de semana será igual a 1).

Quando é feita a codificação dos dados de calendário, os valores codificados do dia de interesse em se realizar a previsão (D+1) são passados como input para as redes. Além disso, também são passados os valores verificados de carga e temperatura nos dois dias anteriores ao

dia corrente D (D-1 e D-2), a temperatura prevista do dia corrente e a temperatura prevista para o dia D+1. A Figura 13 mostra a montagem de um exemplo de entrada das ANNs.

Figura 13 - Montagem do input para treinamento

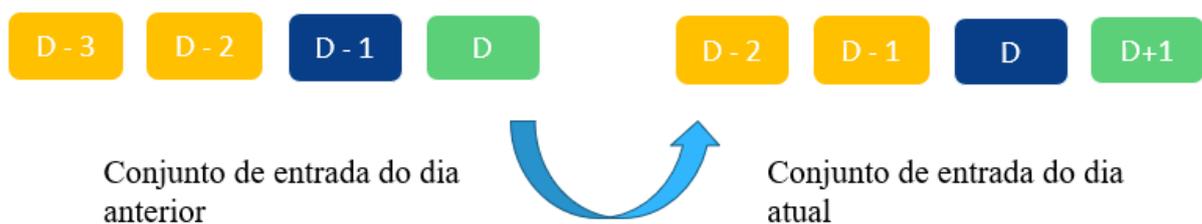


Fonte: Autor

Utilizando os dados do SINtegre foram obtidos valores de temperatura prevista em base horária, portanto 48 valores previstos foram passados como informação de entrada para o treinamento das ANNs. Já no caso do GEFS, a previsão é feita de 3 em 3 horas (devido a característica das informações disponíveis), sendo um total de 16 valores de temperatura prevista (contando o dia corrente, em que se faz a previsão, e o dia previsto). Portanto, para os dados do SINtegre cada conjunto de entrada da ANN será composto por 194 variáveis (50 para os dados de calendário e 144 para informações de carga e temperatura, sendo 96 para dados históricos e 48 para temperatura prevista). No caso do GEFS o conjunto de entrada será formado por 162 variáveis (50 para os dados de calendário e 112 para informações de carga e temperatura, sendo 96 para dados históricos e 16 para temperatura prevista).

Para as RNNs (LSTM e GRU), que são capazes de realizar agregação temporal, é necessário passar os conjuntos de entrada de cada período em que a ANN deverá levar em consideração (*time-steps*). A Figura 14 alustra um exemplo de input para RNN onde 2 *time-steps* são considerados. Para prever a carga do dia D+1, leva-se em consideração o que ocorreu no dia D. Portanto a RNN irá realizar uma previsão para o primeiro *time-step* (dia D), e utilizará o resultado para prever para o dia seguinte (procedimento descrito nos capítulos 3.4 e 3.5).

Figura 14 - Exemplo de input para RNN



Fonte: Autor

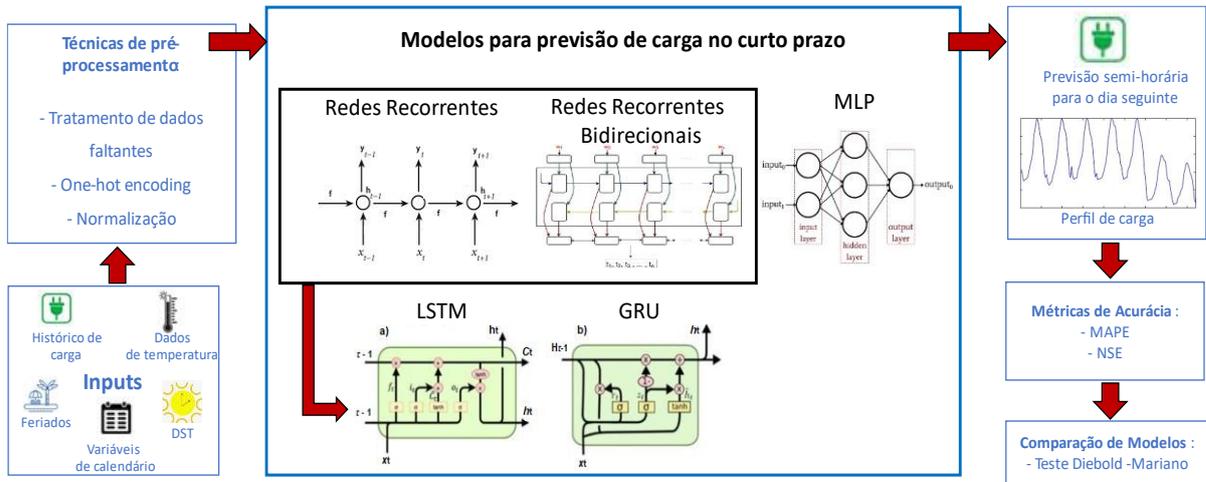
Como o histórico de dados disponível para as duas fontes de dados são diferentes, as análises foram separadas em duas partes. Na primeira parte analisa-se o desempenho dos modelos como os dados retirados do SINtegre. O conjunto de dados do SINtegre se inicia em 01/01/2016 até 19/09/2021 (excluindo os meses de abril a setembro de 2020, onde o perfil de carga é bastante influenciado pela pandemia do COVID-19 e acaba não representando fielmente a carga típica do sistema), sendo os últimos 90 dias reservados para o conjunto de teste (22/06/2021 a 19/09/2021).

Na segunda parte das análises foi feita uma comparação entre as previsões dos modelos desenvolvidos utilizando os dados de temperatura prevista do SINtegre e os dados do GEFS. Como o histórico de dados do GEFS vai até 2019, considerou-se o mesmo período com os dados do SINtegre, utilizando as informações de 01/01/2016 até 31/12/2019 para ambas as fontes de dados, sendo os últimos 90 dias deixados para teste (03/10/2019 a 31/12/2019). As comparações entre as performances dos modelos com os dados do GEFS e do SINtegre foram avaliadas considerando o mercado Sudeste/Centro-Oeste (SE/CO), que se trata do submercado com a maior demanda. Assim, foi possível avaliar o ganho de performance com os dados de temperatura prevista do GEFS.

4.2 Funcionamento e Parâmetros dos Modelos

Em linhas gerais, os dados são coletados e pré-processados antes de serem executadas as redes neurais desenvolvidas. Neste estudo, as execuções são feitas a partir de arquiteturas de MLP, e dos modelos unidirecionais e bidirecionais das redes recorrentes LSTM e GRU. A partir delas, obtêm-se as previsões em base horária, que posteriormente passam por uma *spline* monotônica cúbica [48] para gerar valores semi-horários. *Splines* cúbicos interpolados são populares para ajustar dados porque usam polinômios de baixa ordem e têm continuidade $C=\text{sup}2$, uma propriedade que permite que eles satisfaçam uma restrição de suavidade desejável [49]. Por fim as previsões passam por avaliações de desempenho através das métricas de acurácia. Para a comparação do desempenho entre modelos é utilizado o teste Diebold-Mariano. Na Figura 15 é ilustrado o fluxograma para a previsão de carga no curto prazo desenvolvida.

Figura 15 – Fluxograma da previsão de carga no curto prazo.



Fonte: Autor

Os modelos MLP desenvolvidos foram treinados considerando 1 a 5 camadas ocultas, número de neurônios igual 50% do número total de inputs (97 neurônios para o conjunto de dados obtidos no SINTegre e 81 para os dados do GEFS), taxa de *dropout* de 10%, 10 *folds* na validação cruzada e 1000 previsões usando o Monte Carlo *Dropout*, resultando em 10000 previsões diferentes (10 redes diferentes realizando 1000 previsões diferentes). Para as RNN, os modelos para os submercados SE/CO e S foram treinados com 5 *time-steps*, enquanto nos submercados NE e N foram utilizados 2 *time-steps*. Estes valores foram obtidos através de testes, que buscaram o melhor desempenho das redes.

CAPÍTULO 5

RESULTADOS E DISCUSSÃO

5.1 Simulações Utilizando Temperaturas Do SINtegre

Inicialmente foram executadas as previsões de carga utilizando os dados do SINtegre considerando o período de 01/01/2016 até 19/09/2021. Foram testadas estruturas MLP com diferentes quantidades de camadas ocultas, com a finalidade de verificar se modelos de DNN forneceriam previsões superiores a uma MLP rasa (com apenas uma camada oculta) para o conjunto de dados em questão. Na Tabela 1 estão descritos os resultados médios obtidos pela MLP rasa, e das MLPs com duas, três, quatro e cinco camadas ocultas, onde os melhores resultados estão sinalizados em verde.

Tabela 1 – Performance da MLP rasa e das DNN.

	SE/CO	S	NE	N
MLP 1 camada oculta (rasa)	MAPE: 1,978%	MAPE: 2,881%	MAPE: 2,575%	MAPE: 2,235%
MLP 2 camadas ocultas	MAPE: 2,131%	MAPE: 2,871%	MAPE: 2,905%	MAPE: 2,016%
MLP 3 camadas ocultas	MAPE: 2,035%	MAPE: 3,141%	MAPE: 3,541%	MAPE: 2,225%
MLP 4 camadas ocultas	MAPE: 1,994%	MAPE: 2,793%	MAPE: 3,524%	MAPE: 2,530%
MLP 5 camadas ocultas	MAPE: 2,305%	MAPE: 3,102%	MAPE: 3,922%	MAPE: 2,413%

Observa-se que as MLPs com maior número de camadas não apresentaram ganhos elevados de performance em relação a MLP rasa para o conjunto de dados utilizado. Dessa forma, optou-se em considerar apenas a MLP rasa para a continuidade das comparações com as demais estruturas investigadas e com o modelo do ONS, uma vez que quanto maior o número de camadas, mais complexa a rede se torna, aumentando o esforço computacional e o tempo de treinamento.

Os resultados médios dos modelos desenvolvidos para cada submercado foram então comparados e estão dispostos na Tabela 2 onde os melhores resultados estão sinalizados em verde. Para os submercados SE/CO e S a rede uni-GRU apresentou o melhor desempenho,

porém muito próximo a rede bi-LSTM (com diferença de apenas 0,009%). Para os submercados NE e N os modelos bi-GRU e MLP respectivamente obtiveram maior acurácia.

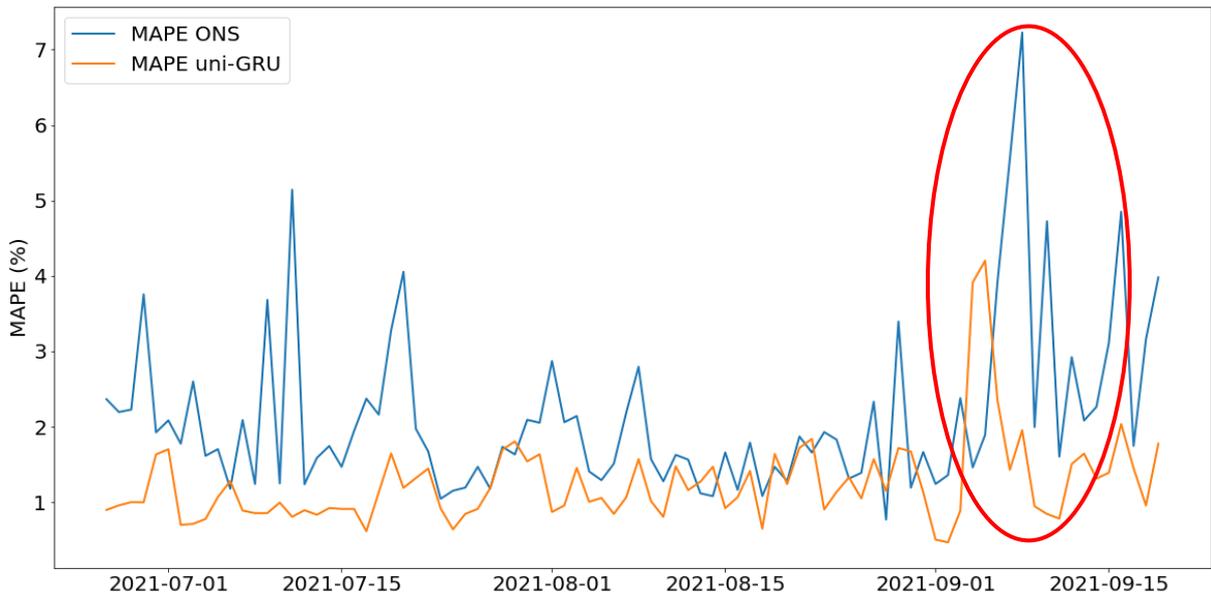
Tabela 2 – Performance das previsões por arquitetura por submercado.

	SE/CO	S	NE	N
ONS	MAPE: 2,135%	MAPE: 2,995%	MAPE: 2,103%	MAPE: 2,656%
	NSE: 0,933	NSE: 0,924	NSE: 0,803	NSE: 0,845
MLP	MAPE: 2,068%	MAPE: 2,873%	MAPE: 2,566%	MAPE: 2,245%
	NSE: 0,936	NSE: 0,926	NSE: 0,675	NSE: 0,744
uni-LSTM	MAPE: 1,342%	MAPE: 2,351%	MAPE: 2,124%	MAPE: 2,501%
	NSE: 0,973	NSE: 0,939	NSE: 0,803	NSE: 0,686
uni-GRU	MAPE: 1,300%	MAPE: 2,220%	MAPE: 1,987%	MAPE: 2,601%
	NSE: 0,975	NSE: 0,945	NSE: 0,821	NSE: 0,658
bi-LSTM	MAPE: 1,309%	MAPE: 2,251%	MAPE: 1,819%	MAPE: 2,539%
	NSE: 0,974	NSE: 0,939	NSE: 0,845	NSE: 0,673
bi-GRU	MAPE: 1,334%	MAPE: 2,293%	MAPE: 1,806%	MAPE: 2,457%
	NSE: 0,973	NSE: 0,941	NSE: 0,843	NSE: 0,692

Considerando o conjunto de teste com os últimos 90 dias do histórico de dados, foram plotados gráficos comparando o desempenho médio por dia (MAPE médio das 48 semi horas previstas por dia) das previsões dos melhores modelos com as previsões realizadas pelo ONS para cada submercado. Os gráficos estão apresentados nas Figura 16 a Figura 19. Observa-se que o período em que os modelos apresentaram piora na performance foi no feriado de 07/09 e nos dias adjacentes. Isso se deve ao fato de que esses dias possuem um perfil de carga que difere do perfil típico de dia útil e de finais de semana, e como possuem poucos dias de feriado no histórico (comparado ao número de dias úteis e finais de semana), os modelos não conseguiram alcançar um bom desempenho como fazem para dias normais. Uma alternativa para tentar

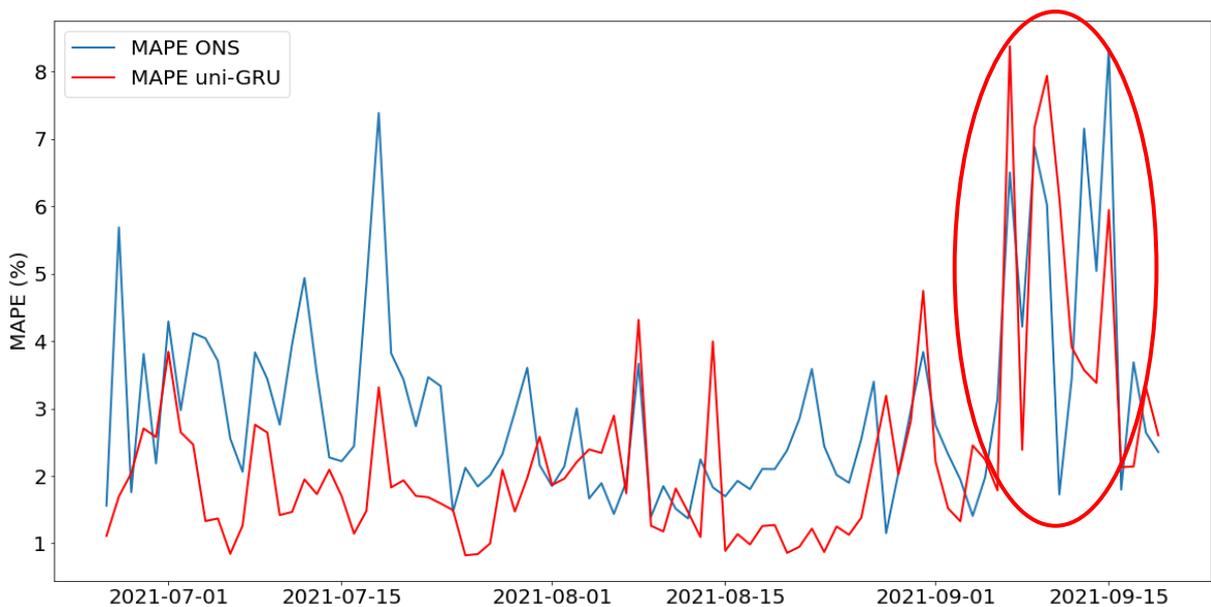
corrigir o problema seria treinar uma rede especializada apenas em prever demanda em feriados, algo a ser testado futuramente. Nota-se que a literatura aponta para os feriados como um dos principais desafios da PCCP [42].

Figura 16 - Comparação de Desempenho – Submercado SE/CO



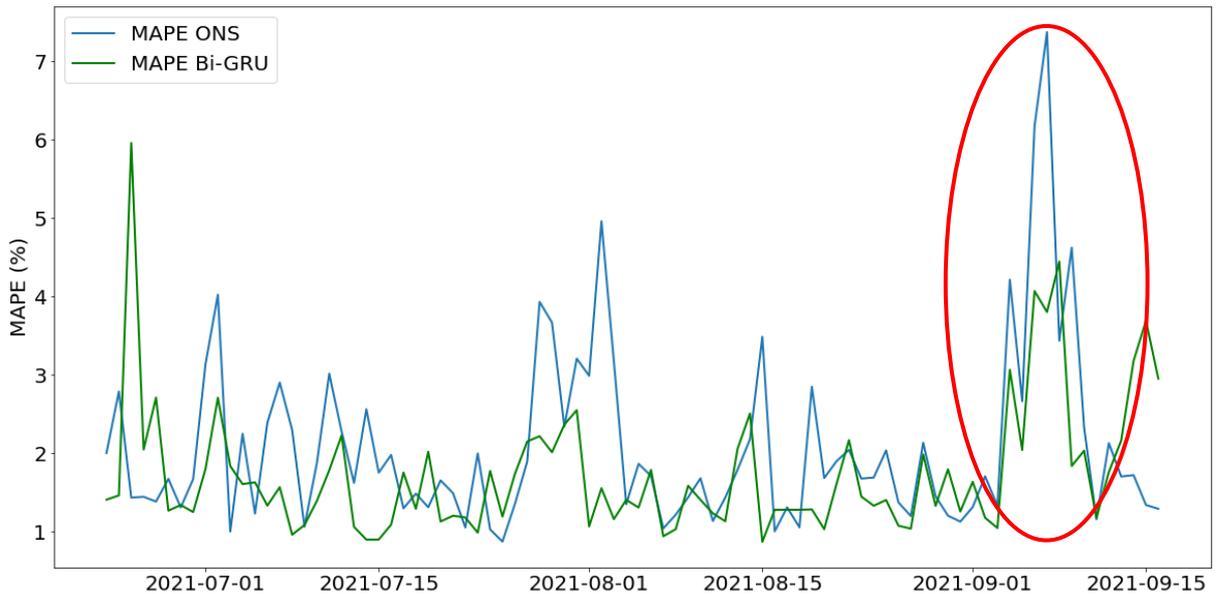
Fonte: Autor

Figura 17 - Comparação de Desempenho – Submercado S



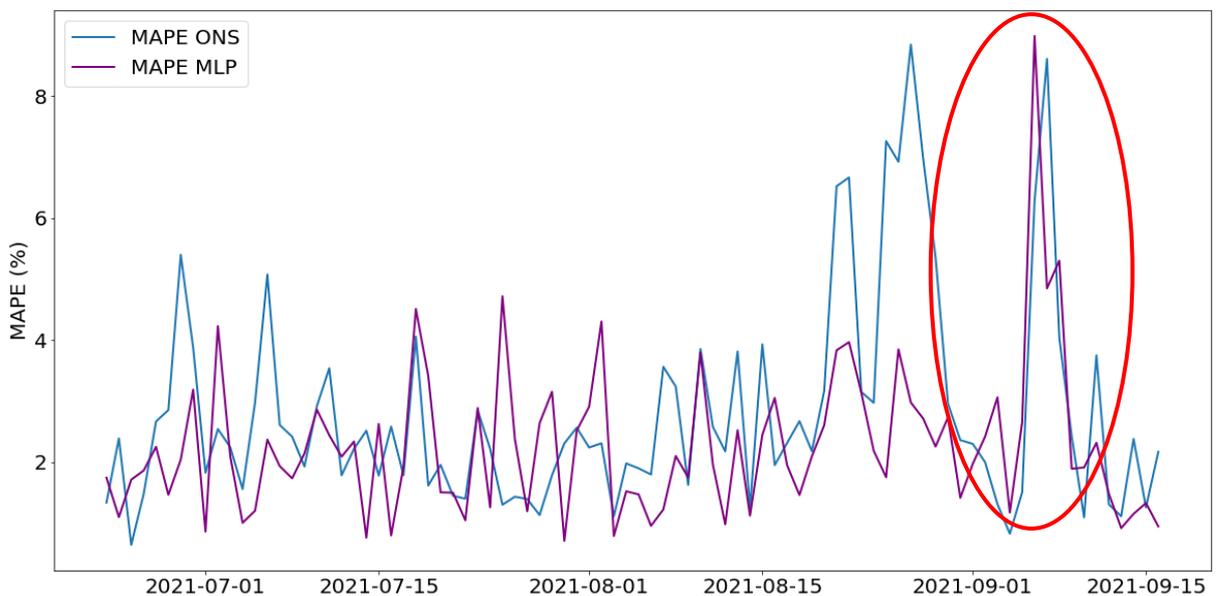
Fonte: Autor

Figura 18 - Comparação de Desempenho - Submercado NE



Fonte: Autor

Figura 19- Comparação de Desempenho - Submercado N



Fonte: Autor

Nesta etapa também foram comparadas as performances entre os modelos desenvolvidos e entre o modelo de previsão utilizado pelo ONS para cada submercado através do teste Diebold-Mariano (resultados dispostos nas Tabela 3 a 6). É válido relembrar que, um *p-value* abaixo de 0,05 ou um valor de DM fora do intervalo $-1,96 < DM < 1,96$ indica que

existe um modelo estatisticamente superior ao outro na comparação par a par, sendo que o modelo com o menor MAPE, ou maior NSE é o modelo com performance superior. Os resultados destacados em verde indicam que o melhor modelo é aquele presente na coluna, enquanto os resultados destacados em amarelo indicam que o melhor modelo está na linha. Os resultados destacados em vermelho não apontaram superioridade em algum modelo.

A comparação par a par entre a performance dos modelos, mostra que as arquiteturas de RNN demonstraram uma maior acurácia em relação aos resultados obtidos pela MLP e pelo ONS para o submercado SE/CO (Tabela 3). Entretanto o teste DM não apontou diferença estatística entre as RNN, ou seja, não é possível afirmar que estatisticamente existe um modelo superior nas comparações entre as estruturas unidirecionais e bidirecionais da LSTM e GRU.

Tabela 3 – Comparação par a par entre a performance dos modelos no SE/CO.

	MLP	Uni-LSTM	Uni-GRU	Bi-LSTM	Bi-GRU
ONS	p-value: 0,696 DM: 0,390	p-value: 0,000 DM: 5,863	p-value: 0,000 DM: 6,540	p-value: 0,000 DM: 6,261	p-value: 0,000 DM: 6,262
MLP	-	p-value: 0,000 DM: 6,093	p-value: 0,000 DM: 6,877	p-value: 0,000 DM: 6,419	p-value: 0,000 DM: 6,545
uni-LSTM	-	-	p-value: 0,257 DM: 1,140	p-value: 0,401 DM: 0,842	p-value: 0,860 DM: 0,176
uni-GRU	-	-	-	p-value: 0,857 DM: 0,180	p-value: 0,389 DM: 0,865
bi-LSTM	-	-	-	-	p-value: 0,520 DM: 0,645

No submercado S, observa-se que novamente todas as RNN apresentaram performance significativamente melhores que a MLP e que o ONS. Diferente do que foi observado no submercado SE/CO, houve uma superioridade estatística no desempenho da uni-GRU e bi-LSTM em relação a uni-LSTM.

Tabela 4 – Comparação par a par entre a performance dos modelos no S.

	MLP	Uni-LSTM	Uni-GRU	Bi-LSTM	Bi-GRU
ONS	p-value: 0,594 DM: 0,533	p-value: 0,000 DM: 4,133	p-value: 0,000 DM: 3,443	p-value: 0,000 DM: 4,946	p-value: 0,005 DM: 2,872
MLP	-	p-value: 0,005 DM: 2,856	p-value: 0,000 DM: 3,443	p-value: 0,000 DM: 3,233	p-value: 0,005 DM: 2,872
uni-LSTM	-	-	p-value: 0,023 DM: 2,304	p-value: 0,037 DM: 2,115	p-value: 0,331 DM: 0,977
uni-GRU	-	-	-	p-value: 0,536 DM: 0,627	p-value: 0,160 DM: -1,417
bi-LSTM	-	-	-	-	p-value: 0,288 DM: -1,067

Para o submercado NE observa-se que o modelo desenvolvido pelo ONS alcançou desempenho superior a MLP, sendo que somente os modelos bidirecionais da LSTM e GRU obtiveram resultados melhores (modelos que alcançaram maior acurácia). O teste DM também apontou uma superioridade da rede uni-GRU em relação a rede uni-LSTM.

Tabela 5 – Comparação par a par entre a performance dos modelos no NE.

	MLP	Uni-LSTM	Uni-GRU	Bi-LSTM	Bi-GRU
ONS	p-value: 0,000 DM: -3,686	p-value: 0,850 DM: -0,189	p-value: 0,349 DM: 0,939	p-value: 0,023 DM: 2,312	p-value: 0,023 DM: 2,306
MLP	-	p-value: 0,000 DM: 6,246	p-value: 0,000 DM: 7,047	p-value: 0,000 DM: 9,265	p-value: 0,000 DM: 8,884
uni-LSTM	-	-	p-value: 0,000 DM: 3,774	p-value: 0,000 DM: 9,943	p-value: 0,000 DM: 7,918
uni-GRU	-	-	-	p-value: 0,000 DM: -4,073	p-value: 0,000 DM: 4,702
bi-LSTM	-	-	-	-	p-value: 0,575 DM: 0,562

Para o submercado N apenas os modelos MLP e bi-GRU apresentaram performance superior ao modelo do ONS. Diferentemente dos outros submercados a MLP também se mostrou superior aos modelos uni-GRU e bi-LSTM. O teste DM também apontou uma superioridade dos modelos LSTM em relação aos modelos GRU, sendo a uni-LSTM superior a uni-GRU e a bi-LSTM superior a bi-GRU.

Tabela 6 – Comparação par a par entre a performance dos modelos no N.

	MLP	Uni-LSTM	Uni-GRU	Bi-LSTM	Bi-GRU
ONS	p-value: 0,001 DM: 3,200	p-value: 0,083 DM: 1,747	p-value: 0,247 DM: 1,163	p-value: 0,140 DM: 1,488	p-value: 0,049 DM: 1,993
MLP	-	p-value: 0,060 DM: -1,889	p-value: 0,014 DM: -2,492	p-value: 0,027 DM: -2,245	p-value: 0,103 DM: -1,643
uni-LSTM	-	-	p-value: 0,009 DM: 2,649	p-value: 0,321 DM: -0,996	p-value: 0,172 DM: 1,376
uni-GRU	-	-	-	p-value: 0,188 DM: -1,326	p-value: 0,000 DM: 3,678
bi-LSTM	-	-	-	-	p-value: 0,037 DM: 2,117

5.2 Comparação de resultados: SINtegre vs GEFS

Na segunda etapa das investigações, buscou-se avaliar a performance dos modelos de previsão utilizando os dados de temperatura prevista do GEFS e comparando com os resultados obtidos através dos dados do SINtegre. É válido destacar que nesta etapa do trabalho foram simulados modelos apenas para o submercado SE/CO. Nestas avaliações, novamente o primeiro passo foi avaliar o desempenho das MLPs com diferentes quantidades de camadas ocultas.

Na Tabela 7 estão descritos os resultados das MLP rasa (com apenas uma camada oculta), e das MLPs com duas, três, quatro e cinco camadas ocultas, que foram avaliadas com os dados do GEFS.

Tabela 7 – Performance da MLP rasa e das DNN com dados do GEFS e SINtegre

	GEFS	SINtegre
MLP 1 camada oculta (rasa)	MAPE: 2,256%	MAPE: 2,486%
MLP 2 camadas ocultas	MAPE: 2,460%	MAPE: 2,352%
MLP 3 camadas ocultas	MAPE: 2,565%	MAPE: 2,591%
MLP 4 camadas ocultas	MAPE: 2,552%	MAPE: 2,414%
MLP 5 camadas ocultas	MAPE: 2,618%	MAPE: 2,581%

Para os modelos treinados com os dados do GEFS a MLP com apenas uma camada oculta obteve o melhor resultado. Já para com os dados do SINtegre, os resultados foram melhores utilizando duas camadas ocultas. Porém observa-se ainda que as DNNs não fornecem grandes incrementos de performance em relação a MLP rasa (com ganho de apenas 0,13% na MLP com duas camadas para os dados do SINtegre). Dessa forma, novamente optou-se em considerar apenas a MLP rasa para a continuidade das comparações de performance com as estruturas de RNN.

Em seguida foram avaliadas as previsões com as demais arquiteturas utilizando os dados de temperatura prevista do GEFS. Nesta etapa não foram comparados os desempenhos das arquiteturas com as previsões publicadas do ONS, visto que o intuito de utilizar os dados do GEFS é buscar aprimoramentos na performance dos modelos desenvolvidos. Novamente, foram avaliadas as performances das arquiteturas unidirecionais e bidirecionais da LSTM e da GRU (uni-LSTM, uni-GRU, bi-LSTM e bi-GRU) para o submercado SE/CO. A acurácia obtida por cada arquitetura está descrita na Tabela 8.

Analisando os valores de MAPE e NSE médios para o conjunto de teste de três meses, é possível observar que o modelo bi-GRU obteve melhores resultados para ambas as fontes de dados utilizadas. Porém, é possível constatar que a utilização dos dados do GEFS não apresentou um aprimoramento significativo na performance dos modelos comparado os dados do SINtegre. Analisando o desempenho do modelo bi-GRU, o MAPE alcançado utilizando os dados do GEFS foi apenas 0,01% melhor enquanto o NSE foi 0,07 melhor com os dados do SINtegre.

Tabela 8 - Performance dos modelos para o submercado SE/CO utilizando dados previstos do SINtegre e do GEFS.

	GEFS	SINtegre
MLP	MAPE: 2,256% NSE: 0,898	MAPE: 2,486% NSE: 0,889
uni-LSTM	MAPE: 2,055% NSE: 0,882	MAPE: 1,973% NSE: 0,900
uni-GRU	MAPE: 1,894% NSE: 0,907	MAPE: 1,978% NSE: 0,907
bi-LSTM	MAPE: 2,038% NSE: 0,893	MAPE: 1,875% NSE: 0,921
bi-GRU	MAPE: 1,855% NSE: 0,915	MAPE: 1,865% NSE: 0,922

Para efeito de comparação de performance entre os modelos com cada uma das fontes de dados, foram realizados os testes Diebold-Mariano, os quais os resultados estão resumidos nas Tabela 9 e Tabela 10.

Para os modelos treinados com a temperatura prevista pelo GEFS, pode-se concluir pela Tabela 9 que a estrutura recorrente GRU mostrou-se superior as demais, sendo que não houve diferença estatística entre suas versões uni e bidirecionais. O teste DM também não acusou diferença estatística entre os modelos LSTM e MLP.

Aplicando o teste DM nas previsões dos modelos treinados com os dados SINtegre (Tabela 10), é possível observar que não houve diferença estatística entre os modelos RNN, os quais se mostraram superiores ao modelo MLP.

Tabela 9 – Comparação par a par entre a performance dos modelos com os dados do GEFS.

	MLP	Uni-LSTM	Uni-GRU	Bi-LSTM	Bi-GRU
MLP	-	p-value: 0,141 DM: 1,486	p-value: 0,014 DM: 2,510	p-value: 0,108 DM: 1,626	p-value: 0,021 DM: 2,340
uni-LSTM	-	-	p-value: 0,034 DM: 2,154	p-value: 0,776 DM: 0,285	p-value: 0,038 DM: 2,106
uni-GRU	-	-	-	p-value: 0,020 DM: 2,362	p-value: 0,525 DM: 0,638
bi-LSTM	-	-	-	-	p-value: 0,010 DM: 2,614
bi-GRU	-	-	-	-	-

Tabela 10 – Comparação par a par entre a performance dos modelos com os dados do SINtegre.

	MLP	Uni-LSTM	Uni-GRU	Bi-LSTM	Bi-GRU
MLP	-	p-value: 0,005 DM: 2,883	p-value: 0,001 DM: 3,316	p-value: 0,000 DM: 3,975	p-value: 0,000 DM: 4,206
uni-LSTM	-	-	p-value: 0,955 DM: 0,056	p-value: 0,205 DM: 1,277	p-value: 0,205 DM: 1,276
uni-GRU	-	-	-	p-value: 0,252 DM: 1,154	p-value: 0,178 DM: 1,358
bi-LSTM	-	-	-	-	p-value: 0,853 DM: 0,185
bi-GRU	-	-	-	-	-

Uma terceira aplicação do teste DM foi realizada. Para uma mesma estrutura, aplicou-se o teste entre os modelos treinados com os dados do SINtegre e com os dados do GEFS. Os resultados estão dispostos na Tabela 11. Através desses resultados pode-se afirmar que não existe diferença estatística entre os modelos treinados com dados de temperatura prevista disponíveis na plataforma SINtegre do ONS e dados previsto pelo sistema GEFS. Ou seja, o uso dos dados de temperatura prevista do GEFS não trouxe ganhos de performance comparado ao uso dos dados do SINtegre.

Tabela 11 – Comparação entre os modelos treinados com dados do GEFS e do SINtegre

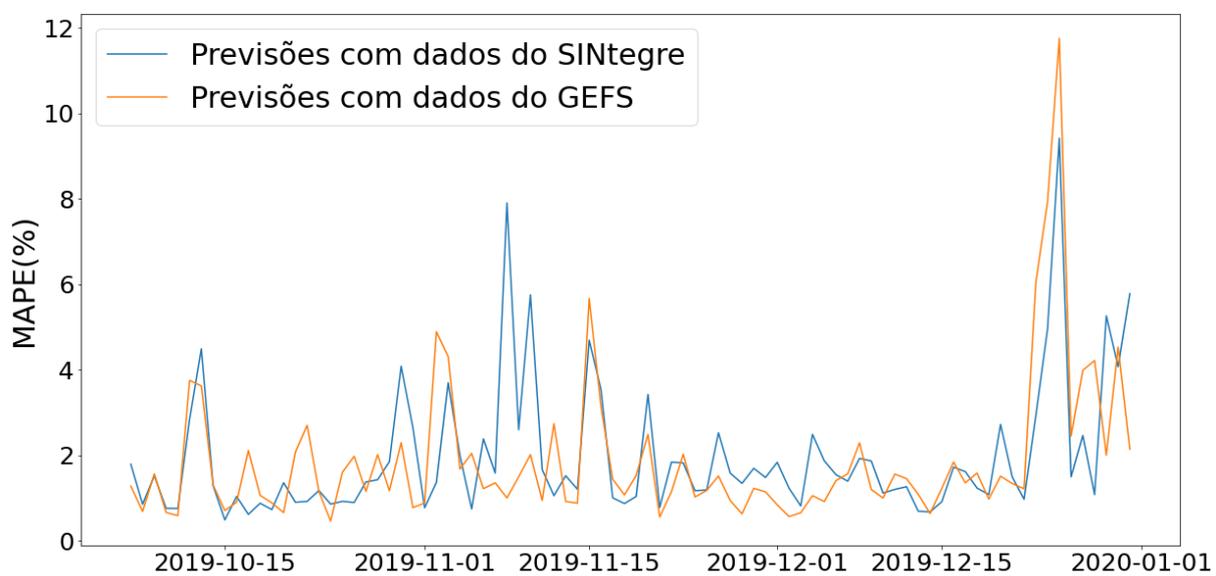
	SINtegre - MLP	SINtegre - Uni-LSTM	SINtegre - Uni-GRU	SINtegre - Bi-LSTM	SINtegre - Bi-GRU
GEFS - MLP	p-value: 0,255 DM: 1,146	-	-	-	-
GEFS - uni-LSTM	-	p-value: 0,521 DM: 0,644	-	-	-
GEFS - uni-GRU	-	-	p-value: 0,590 DM: 0,543	-	-
GEFS - bi-LSTM	-	-	-	p-value: 0,271 DM: 1,107	-
GEFS - bi-GRU	-	-	-	-	p-value: 0,940 DM: 0,076

Apesar do teste não ter resultado em uma diferença estatística entre os modelos, plotou-se um gráfico com os valores de MAPE médio por dia para os 90 dias do conjunto de teste para os modelos que obtiveram melhores resultados com relação as métricas de acurácia, mostradas na Tabela 8.

A Figura 20 ilustra a curva do MAPE obtida com a estrutura bi-GRU treinada com cada uma das fontes de dados. Nesse gráfico é possível observar que apesar de existir diferenças relevantes entre os modelos em alguns pontos, num geral os modelos possuem um comportamento bastante semelhante. Isso mostra que em caso os dados providos pelo ONS não estejam disponíveis, outras fontes podem ser utilizadas sem que haja perda de qualidade e acuraria nas previsões.

Outra vantagem no uso dos dados do GEFS está no horizonte de previsão de temperatura fornecido, uma vez que o GEFS fornece previsões até 10 dias a frente, enquanto o ONS realiza previsões apenas para 7 dias a frente.

Figura 20 – Performance da estrutura bi-GRU treinadas com dados do GEFS e dados do SINtegre.



Fonte - Autor

CAPÍTULO 6

CONCLUSÕES

6.1 Considerações Finais

Este trabalho apresentou uma avaliação de diferentes arquiteturas de ANNs para a modelagem de PCCP. Os resultados do presente estudo evidenciam o potencial dos modelos RNN, onde, na maioria dos casos analisados, as redes LSTM e GRU apresentaram desempenho superior às MLPs e ao modelo de previsão do ONS, obtendo resultados bastante satisfatórios para os diferentes conjuntos de testes, com valores de MAPE abaixo de 2,3% para todos os submercados de energia. A superioridade desses modelos pôde ser comprovada também através da aplicação do teste de hipótese proposto por Diebold e Mariano.

Também foi analisado o uso de informações do modelo climático global GEFS em combinação com modelos ANN para realizar PCCP. Os modelos de ANN obtiveram uma boa acurácia ao utilizar essas informações, como pode ser observado nos resultados da seção 5.2. Apesar dos resultados não terem apontado para um ganho de desempenho comparado aos modelos treinados com os dados disponibilizados pelo ONS (apontado pelo teste DM), o estudo mostrou que outras fontes podem ser utilizadas sem que haja perda de qualidade e acuraria nas previsões, além de possibilitar um horizonte maior nas previsões de carga.

Foi observado que o desempenho dos modelos piora em feriados e dias adjacentes, uma vez que a carga observada nesses dias possui comportamento diferente comparado a carga em dias normais, o que é muito verificado e discutido na literatura. Uma proposta de para tentar contornar esse problema seria treinar ANNs especializadas apenas em feriados. É importante mencionar que dados como feriado combinado ou não com final de semana alteram também o comportamento das pessoas e conseqüentemente da carga.

Apesar das RNN terem se mostrado bastante eficazes para PCCP, pôde-se observar que diferentes modelos se destacaram para cada submercado. Isso comprova a peculiaridade da demanda de cada submercado e como o fator climático afeta em seu comportamento.

6.2 Trabalhos Futuros

Estudos futuros devem ser direcionados para avaliar o desempenho da PCCP com novas arquiteturas de ANNs, como as CNN e as Redes Bayesianas, que podem explorar melhor um grande número de camadas ocultas e conjuntos de dados maiores, e de outros modelos de previsão, como SVMs e modelos estatísticos.

Novos estudos avaliando o uso de outros modelos climáticos, como o Centro Europeu de Previsões Meteorológicas de Médio Prazo (ECMWF), são pertinentes, bem como outras variáveis meteorológicas/climáticas, como umidade, cobertura de nuvens, vento e outras.

Outra direção futura deve ser direcionada para um treinamento/monitoramento dinâmico de ANNs à medida que novos dados se tornem disponíveis, isso criaria maneiras rápidas para as ANNs se adaptarem em relação aos comportamentos recentes da série temporal (preditores e variáveis de interesse). Para isso, a construção de gráficos de controle seria um caminho a se seguir.

O treinamento de outros modelos de ANN especializadas apenas em prever a carga de feriados também deve ser avaliado. Para isso, a obtenção de um histórico maior de dados é de grande valia.

REFERÊNCIAS

- [1] Jetcheva, J. G., Majidpour, M., & Chen, W. P. *Neural network model ensembles for building-level electricity load forecasts*. *Energy & Buildings*, 84, 214-223, 2014.
- [2] NIST – National Institute of Standards and Technology. *Framework and Roadmap for Smart Grid Interoperability Standards*. Release 1.0. 2010.
- [3] Leiva, J.; Palacios, A.; Aguado, J.A. *Smart metering trends, implications and necessities: A policy review*. *Renew. Sustain. Energy Rev.*, 55, 227–233, 2016.
- [4] Mocanu E, Nguyen PH, Gibescu M, Kling WL, *Deep learning for estimating building energy consumption*, *Sustainable Energy Grids Networks*. 6, 91–99, 2016.
- [5] Rahman, S. *Formulation and Analysis of a Rule-Based Short-Term Load Forecasting Algorithm*, *Proceedings of IEEE*, 78, 5, 1990.
- [6] Charytoniuk, W., & Chen, M. S. (2000). *Very short-term load forecasting using artificial neural networks*. *IEEE transactions on Power Systems*, 15(1), 263-268.
- [7] Taylor, J. W. (2008). *An evaluation of methods for very short-term load forecasting using minute-by-minute British data*. *International journal of forecasting*, 24(4), 645-658.
- [8] Santos, T. N., Diniz, A. L., Saboia, C. H., Cabral, R. N., & Cerqueira, L. F. (2020). *Hourly pricing and day-ahead dispatch setting in Brazil: The dessem model*. *Electric Power Systems Research*, 189, 106709.
- [9] Koponen, P., Ikäheimo, J., Koskela, J., Brester, C., & Niska, H. (2020). *Assessing and comparing short term load forecasting performance*. *Energies*, 13(8), 2054.
- [10] López, M., Sans, C., Valero, S., & Senabre, C. (2018). *Empirical comparison of neural network and autoregressive models in short-term load forecasting*. *Energies*, 11(8), 2080.
- [11] Wang, J., Wang, J., Li, Y., Zhu, S., & Zhao, J. (2014). *Techniques of applying wavelet de-noising into a combined model for short-term load forecasting*. *International Journal of Electrical Power & Energy Systems*, 62, 816-824.
- [12] El-Hendawi, M., & Wang, Z. (2020). *An ensemble method of full wavelet packet transform and neural network for short term electrical load forecasting*. *Electric Power Systems Research*, 182, 106265.
- [13] Henselmeyer, S., & Grzegorzec, M. (2021). *Short-Term Load Forecasting Using an Attended Sequential Encoder-Stacked Decoder Model with Online Training*. *Applied Sciences*, 11(11), 4927.
- [14] Kwon BS., Park RJ., Song KB. *Short-Term Load Forecasting Based on Deep Neural Networks Using LSTM Layer*. *Journal of Electrical Engineering and Technology*, 2020.
- [15] Zheng, J., Xu, C., Zhang, Z., Li, X. *Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network*. *Information Sciences and Systems*. IEEE, 2017.
- [16] Lago, J., De Ridder, F., & De Schutter, B. (2018). *Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms*. *Applied Energy*, 221, 386-405.
- [17] Conejo A, Plazas M, Espinola R, Molina A. *Day-ahead electricity price forecasting using the wavelet transform and ARIMA models*. *IEEE Trans Power Syst* 2005;20(2):1035–42.
- [18] J. Nowicka-Zagrajek, R. Weron. *Modeling electricity loads in California: ARMA models with hyperbolic noise*, *Signal Process.*, 82 (2002), pp. 1903-1915
- [19] Dudek, G. (2016). *Neural networks for pattern-based short-term load forecasting: A comparative study*. *Neurocomputing*, 205, 64-74.
- [20] Bengio Y., Courville A., Vincent P. *Representation Learning: A Review and New Perspectives*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35, 1798-1828, 2013.
-

-
- [21] Nowotarski J., Liu B., Weron R., Hong T. *Improving short term load forecast accuracy via combining sister forecasts*, 98, 40-49, 2016.
- [22] Goodfellow I, Bengio Y, Courville A. *Deep learning*. MIT Press; 2016 Available at: <<http://www.deeplearningbook.org/>> .
- [23] Sutskever I, Vinyals O, Le QV. *Sequence to sequence learning with neural networks*. In: Proceedings of the 27th international conference on neural information processing systems, NIPS'14. MIT Press; 2014. p. 3104–12.
- [24] Hippert H.S, Pedreira C.E, Souza R.C, *Neural networks for short-term load forecasting: A review and evaluation*, IEEE Trans. On Power Systems, 16 (1) 44–55, 2001.
- 2014.
- [25] de Queiroz, A. R. (2016). *Stochastic hydro-thermal scheduling optimization: An overview*. Renewable and Sustainable Energy Reviews, 62, 382-395.
- [26] Maceira, M. P., Terry, L. A., Costa, F. S., Damázio, J. M., & Melo, A. C. G. (2002, June). *Chain of optimization models for setting the energy dispatch and spot price in the Brazilian system*. In Proceedings of the power system computation conference-PSCC (Vol. 2, pp. 24-28).
- [27] ONS – Operador Nacional do Sistema. Plataforma SINtegre. Disponível em:<<http://www.ons.org.br/topo/aceso-restrito>>
- [28] ABRACEEL – Associação Brasileira dos Comercializadores de Energia. Apresentação 6ª Reunião FT PrevCargaDessem. Disponível em:< <https://abraceel.com.br/biblioteca/apresentacoes/2021/04/apresentacao-6a-reuniao-ft-prevcargadessem/>>
- [29] Silva, L. N., Abaide, A. R., Figueiró, I. C., Silva, J. O., Rigodanzo, J., Sausen, J. P. *Development of Brazilian multi region short-term load forecasting model considering climate variables weighting in ANN model*. In: 2017 52nd International Universities Power Engineering Conference (UPEC), 1-6. IEEE, 2017.
- [30] Silva, M., Santos, P., Forero, L., Mello Junior, H., Previsão de carga horária como parâmetro do modelo de precificação e despacho hidrotérmico de curto prazo (DESSEM). In.: Congresso Brasileiro de Inteligência Computacional (Belém, 2019), 2019.
- [31] F.M. Bianchi, E. Maiorino, M.C. Kampffmeyer, A.Rizzi, R. Jenssen. *An overview and comparative analysis of recurrent neural networks for short-term load forecasting*, 2017.
- [32] de Faria, V. A. D., de Queiroz, A. R., Lima, L. M., Lima, J. W. M., & da Silva, B. C. (2021). *An assessment of multi-layer perceptron networks for streamflow forecasting in large-scale interconnected hydrosystems*. International Journal of Environmental Science and Technology, 1-20.
- [33] Gal, Y.; Ghahramani, Z. *A Theoretically Grounded Application Of Dropout In Recurrent Neural Networks*. In: Lee, D.D.; Sugiyama, M.; Luxburg, U.V.; Guyon, I.; Garnett, R. Advances in Neural Information Processing Systems 29 (NIPS 2016), pp. 1019–1027. Curran Associates Inc., Barcelona, 2016.
- [34] Leamer E. *Specification studies: ad hoc inference with non-experimental data*. New York: Wiley, 1978.
- [35] Lovell, M.C. *Data mining*. Review of Economic Studies, 45, 1-12, 1983.
- [36] Diebold, F.X.; Mariano, R.S. *Comparing predictive accuracy*. Journal of Business & Economic Statistics, 13, 253-263, 1995.
- [37] Harvey, D.; Leybourne, S., Newbold, P. *Testing the Equality of Prediction Mean Squared Errors*. International Journal of Forecasting, 13, 281-91, 1997.
- [38] Eshraghi, H.; Queiroz, A. R., Sankarasubramanian, A., De Carolis, J. F. *Quantification of climate-induced interannual variability in residential US electricity demand*. Energy, 236, 121273, 2021.
- [39] Cawthorne, D.; Queiroz, A. R., Eshraghi, H., Arumugam, S., De Carolis, J. F. *The Role of Temperature Variability on Seasonal Electricity Demand in the Southern US*. Frontiers in Sustainable Cities, 3, 43, 2021a.
-

-
- [40] Cawthorne, D.; Queiroz, A. R.; Eshraghi, H.; Arumugam, S.; De Carolis, J.F. *The Role of Temperature Variability on Seasonal Electricity Demand in the Southern US*. *Frontiers in Sustainable Cities*, 3, 43, 2021b.
- [41] Instituto Brasileiro De Geografia E Estatística – IBGE. IBGE Cidades, 2021. Disponível em: <<https://cidades.ibge.gov.br>>.
- [42] Dahl, M.; Brun, A.; Kirsebom, O.S.; Andresen, G.B. *Improving Short-Term Heat Load Forecasts with Calendar and Holiday Data*. *Energies*, 11, 1678, 2018.
- [43] Medium, Analytics Vidhya. *Multi Layer Perceptron Using Keras on MNIST dataset for Digital Classification*. Available from: <<https://medium.com/analytics-vidhya/multi-layer-perceptron-using-keras-on-mnist-dataset-for-digit-classification-problem-relu-a276cbf05e97>>
- [44] *A Guide for Using Deep Learning for Complex Trait Genomic Prediction* - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Multi-Layer-Perceptron-MLP-diagram-with-four-hidden-layers-and-a-collection-of-single_fig1_334609713 [accessed 8 Feb, 2022]
- [45] Technopremium. RNN, *Talking about Gated Recurrent Unit*. Available from: <<https://technopremium.com/blog/rnn-talking-about-gated-recurrent-unit/>>
- [46] T. Wong and P. Yeh, "Reliable Accuracy Estimates from k -Fold Cross Validation," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1586-1594, 1 Aug. 2020, doi: 10.1109/TKDE.2019.2912815.
- [47] SAUD, Sheikh et al. *Performance improvement of empirical models for estimation of global solar radiation in India: A k -fold cross-validation approach*. *Sustainable Energy Technologies and Assessments*, v. 40, p. 100768, 2020
- [48] Dubinsky, A.; Elperin, T. *A method for calculating a load curve using average values of load over time intervals*, *International Journal of Electrical Power & Energy Systems*, v. 19, n. 6, pp. 393-395, August, 1997
- [49] Wolberg and Alfy, "Monotonic cubic spline interpolation," *1999 Proceedings Computer Graphics International*, 1999, pp. 188-195, doi: 10.1109/CGI.1999.777953.
- [50] Kong W., Dong ZY., Jia Y., Hill DJ., Xu Y., Zhang Y. *Short-Term Residential Load Forecasting based on LSTM Recurrent Neural Network*. *IEEE Transactions on Smart Grid*, 10 (1), 841-851, 2017.
- [51] Zheng, J., Xu, C., Zhang, Z., Li, X. *Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network*. *Information Sciences and Systems*. IEEE, 2017.
- [52] Jiao R., Zhang T., Jiang Y., He H. *Short-Term Non-residential Load Forecasting based on Multiple Sequences LSTM Recurrent Neural Network*, *IEEE Access*, 6, 59438-59448, 2018.
- [53] Ghofrani, M.; Ghayekhloo, M.; Arabali, A.; Ghayekhloo, A. *A hybrid short-term load forecasting with a new input selection framework*. *Energy*, 81, 777–786, 2015.
- [54] Tian C., Hao Y. *A Novel Nonlinear Combined Forecasting System for Short-Term Load Forecasting*. *Energies*, 11, 712, 2018.
- [55] Jin, C.H.; Pok, G.; Lee, Y.; Park, H.-W.; Kim, K.D.; Yun, U.; Ryu, K.H. *A SOM clustering pattern sequence-based next symbol prediction method for day-ahead direct electricity load and price forecasting*. *Energy Convers. Manag.*, 90, 84–92, 2015.
- [56] Kumar S., Hussain L., Banarjee S., Reza M. *Energy Load Forecasting using Deep Learning Approach- LSTM and GRU in Spark Cluster*. In: 2018 Fifth International Conference on Emerging Applications of Information Technology (EAIT). IEEE, 1-4, 2018.
- [57] *Improving the DBLSTM for on-line Arabic handwriting recognition* - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Bidirectional-Recurrent-Neural-Network-BRNN_fig5_339490382 [accessed 19 Feb, 2022]
-

[58] Bouktif S., Fiaz A., Ouni A. Serhani MA. *Optimal Deep Learning LSTM Model for Electric Load Forecasting using Feature Selection and Genetic Algorithm: Comparison with Machine Learning Approaches*. *Energies*, 11, 1636, 2018.

[59] WYTHOFF, Barry J. *Backpropagation neural networks: a tutorial*. *Chemometrics and Intelligent Laboratory Systems*, v. 18, n. 2, p. 115-155, 1993.
