Instituto Politécnico de Leiria
Escola Superior de Tecnologia e Gestão
Departamento de Engenharia Informática e Departamento de Matemática
Mestrado em Ciência de Dados

PREVISÃO DE CURTO PRAZO PARA CONSUMO DE ENERGIA EM *CAMPI* UNIVERSITÁRIOS

PAULO ROBERTO DA SILVA OLIVEIRA

Leiria, Março de 2024

Esta página foi deixada em branco intencionalmente.

Instituto Politécnico de Leiria
Escola Superior de Tecnologia e Gestão
Departamento de Engenharia Informática e Departamento de Matemática
Mestrado em Ciência de Dados

PREVISÃO DE CURTO PRAZO PARA CONSUMO DE ENERGIA EM *CAMPI* UNIVERSITÁRIOS

PAULO ROBERTO DA SILVA OLIVEIRA

Relatório de Projeto realizado sob orientação e supervisão dos Professores Carlos Fernando de Almeida Grilo, João Sousa, Luís Távora e Pedro Marques.

Leiria, Março de 2024

Esta página foi deixada em branco intencionalmente.

AGRADECIMENTOS

"*Sem conselhos os projetos fracassam, mas com muitos conselheiros há sucesso.*" (Prov. 15:22 NAA)

Ao longo deste Projeto pude contar com o apoio de diversas pessoas, que me ajudaram a prosseguir na jornada, e é com satisfação que agradeço:

À minha família por todo o apoio, pela renúncia de tempo e presença, e por estar ao meu lado em todas as decisões.

Ao Prof. Carlos Grilo pelo seu tempo, paciência, conselhos, e profissionalismo que foram fundamentais no desenvolvimento deste Trabalho.

Ao Prof. João Sousa com suas observações precisas e sugestões perspicazes, e ao Prof. Pedro Marques por compartilhamento do conjunto de dados utilizado neste Projeto.

Ao Prof. Luís Távora preciso sublinhar o significante contributo para a conclusão atempada deste Projeto.

Finalmente, ao Instituto de Telecomunicações (IT) pelo apoio dado na realização deste trabalho (o IT é financiado a nível nacional pela Fundação para a Ciência e a Tecnologia e, onde aplicável, por fundos europeus através dos projetos UIDB/EEA/50008/2020 and LA/P/0109/2020).

Esta página foi deixada em branco intencionalmente.

RESUMO

Diversas instituições de ensino têm vindo a instalar medidores inteligentes em diferentes edifícios dos seus *campi*, permitindo detalhar o consumo quase em tempo real, dotando essas organizações de significativos volumes de dados com valiosa informação do ponto de vista estratégico. O consumo de energia em *campus* universitário é impulsionado principalmente por vários fatores, como: ocupação, horário de funcionamento, tipo da edificação, idade da edificação, tipologia de equipamento instalado e condições climatéricas. Há ainda categorias diferentes: edifícios académicos, administrativos e edifícios residenciais. Nesse contexto, modelos estatísticos e modelos de aprendizagem computacional supervisionados desempenham um papel essencial, uma vez que permitem aplicar técnicas de previsão baseadas em dados históricos.

Uma boa previsão do consumo de energia elétrica e de gás poderá viabilizar: **a**) O dimensionamento mais rigoroso de sistemas de produção fotovoltaica em regime de autoconsumo, procurando compatibilizar o consumo com a disponibilidade de produção fotovoltaica; **b**) Uma estimativa de encargos futuros com a energia elétrica; **c**) A adoção de planos de gestão de procura de energia, tentando induzir uma maior flexibilidade da procura em períodos mais críticos ou com maiores penalizações tarifárias.

Portanto, a proposta deste trabalho assenta na modelação com base no comportamento de dados históricos e na otimização de parâmetros de redes neuronais para obter o mínimo de erro possível na previsão do consumo de energia elétrica do dia seguinte para o *Campus* 2 da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria, incluindo a análise do desempenho dos modelos explorados.

Foram testados diversos modelos estatísticos SARIMA/SARIMAX com validação cruzada, e modelos de Inteligência Artificial (IA), nomeadamente, *k-Nearest Neighbors* (KNN), *Extreme Gradient Boosting* (XGBoost), redes neuronais, em particular, *Multilayer Perceptron* (MLP), redes *Long Short-Term Memory* (LSTM) e redes *Gated Recurrent Unit* (GRU) com diversas parametrizações e obtidos resultados com cada tipo de modelo, sendo visível que num significativo número deles foi possível obter um *Mean Absolute Percentage Error* (MAPE) abaixo dos 8%.

Palavras-chave: Previsão de Consumo de Energia, Séries Temporais, Modelos Estatísticos de Previsão, Aprendizagem Computacional, Redes Neuronais Artificiais.

ABSTRACT

Several educational institutions have been installing smart meters in different buildings on their *campi*, allowing consumption to be detailed almost in real time, providing these organizations with significant volumes of data with valuable information from a strategic point of view. Energy consumption in university *campus* is driven mainly by several factors, such as: occupancy, opening hours, type of building, age of the building, type of installed equipment and weather conditions. There are also different categories: academic buildings, administrative buildings and residential buildings. In this context, statistical models and supervised computational learning models play an essential role, as they allow the application of forecasting techniques based on historical data.

A good forecast of electricity and gas consumption could enable: **a**) More rigorous sizing of photovoltaic production systems under self-consumption, seeking to make consumption compatible with the availability of photovoltaic production; **b**) An estimate of future electricity charges; **c**) The adoption of energy demand management plans, trying to induce greater demand flexibility in more critical periods or with greater tariff penalties.

Therefore, the proposal of this work is based on modeling based on the behavior of historical data and the optimization of neural network parameters to obtain the minimum possible error in the prediction. of electricity consumption the following day for the *Campus* 2 of the Higher School of Technology and Management of the Polytechnic Institute of Leiria, including analysis of model performance explored.

Several SARIMA/SARIMAX statistical models were tested with cross-validation, and Artificial Intelligence (AI) models, namely, k-Nearest Neighbors (KNN), Extreme Gradient Boosting (XGBoost), neural networks, in particular, Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Unit (GRU) with different parameterizations and obtained results with each type of model, and it is clear that in a significant number of them it was possible to obtain a Mean Absolute Percentage Error (MAPE) below 8%.

Keywords: Energy Consumption Forecast, Time Series, Statistical Forecasting Models, Computational Learning, Artificial Neural Networks.

Esta página foi deixada em branco intencionalmente.

ÍNDICE

Apêndices

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

ACF	Função de Autocorrelação.
ADF	Teste de *Dickey-Fuller* Aumentado.
ADs	Árvores de Decisão.
AIC	Critério de Informação de Akaike.
AnS	Aprendizagem Não Supervisionada.
AR	Modelo Autorregressivo.
ARIMA	Modelo Autorregressivo Integrado de Média Móvel.
AS	Aprendizagem Supervisionada.
COLAB	*Google Colaboratory*.
DL	*Deep Learning*.
DST	Horário de Verão.
ESTG	Escola Superior de Tecnologia e Gestão.
EW	*Expanding Window*.
FFN	*Redes Feed-forward*.
GPU	*Graphics Processing Unit*.
GRU	*Gated Recurrent Unit*.
HW	Modelo de Suavização Exponencial *Holt-Winters*.
I	Modelo Integrado.
IA	Inteligência Artificial.

IF	*Isolation Forest*.
IPLeiria	Instituto Politécnico de Leiria.
IPMA	Instituto Português do Mar e da Atmosfera.
IQR	Amplitude do Intervalo Interquartil.
IT	Instituto de Telecomunicações.
KNN	*k-Nearest Neighbors*.
KPSS	Teste de *Kwiatkowski-Phillips-Schmidt-Shin*.
kVA	Potência Reactiva Indutiva.
kVAr	Potência Reactiva Capacitiva.
kW	Quilowatt.
kWh	Quilowatt-hora.
LOESS	*Locally Estimated Scatterplot Smoothing*.
LSTM	*Long Short-Term Memory*.
MA	Modelo de Média Móvel.
MAE	Erro Absoluto Médio.
MAPE	Erro Percentual Médio Absoluto.
ML	*Machine Learning*.
MLP	*Multilayer Perceptron*.
MLR	Regressão Linear Múltipla.
MSE	Erro Médio Quadrático.
MSTL	*Multiple STL Decomposition*.
NaN	*Not a Number*.
NARX	Rede Neural exógena auto-regressiva não linear.
NUMPY	*The NUMerical PYthon Package*.
PACF	Função de Autocorrelação Parcial.

PANDAS	*Python Data Analysis Library*.
QQ	Gráfico Quanti-Quantil.
RF	*Random Forest*.
RMSE	Raiz do Erro Médio Quadrático.
RNA	Rede Neuronal Artificial.
RNN	Rede Neural Recorrente.
RW	*Rolling Window*.
SARIMA	Modelo ARIMA Sazonal.
SARIMAX	Modelo SARIMA com Variável Exógena.
SHW	Modelo Sazonal de Suavização Exponencial *Holt-Winters*.
STL	*Seasonal and Trend decomposition using Loess*.
SVR	Support Vector Regression.
TEPCO	Tokyo Electric Power Company.
TPU	*Tensor Processing Unit*.
TS	Série Temporal.
TSC	*Time-Series Clustering*.
XGBoost	*Extreme Gradient Boosting*.
XLSX	*Microsoft Excel Worksheet*.

Esta página foi deixada em branco intencionalmente.

1

INTRODUÇÃO

A energia é fundamental para o desenvolvimento sustentável: ela tem um enorme impacto ambiental, social e económicos, entre os quais sua influência nas mudanças climáticas, esforços de redução da pobreza, produtividade industrial e agrícola e saúde ambiental e humana [1]. A sociedade depende de energia para obter calor e luz, para funcionamento das indústrias, das universidades, para acionar a maioria dos eletrodomésticos nas casas e tantas outras atividades [2].

Análises do consumo de energia são cruciais porque permitem ter uma perspetiva de padrões de distribuição social e formas de apropriação do uso de recursos. O desenvolvimento de estratégias de uma transição energética sustentável é um dos mais importantes desafios mundiais do século XXI, e as escolhas que faremos nos próximos anos sobre energia irão determinar que tipo de mundo as gerações futuras herdarão [1].

Uma tendência crescente na procura de energia elétrica implica [3]: **i**) Desafios para manter um equilíbrio entre consumo e produção; **ii**) A necessidade de desenvolvimento de estímulos à mudança de comportamentos na utilização de energia elétrica, buscando um uso eficiente e evitando o desperdício; **iii**) A criação de estratégias de controlo; **iv**) A necessidade de previsões confiáveis e robustas, baseadas nas séries temporais do histórico do consumo.

Uma forma simples de levar a cabo um procedimento de previsão é usar o valor do período anterior. Porém, usar esse procedimento simples para prever vários períodos adiante geralmente não funciona bem, pois os erros em relação aos valores reais tendem a aumentar. Previsões mais apropriadas e adequadas podem ser obtidas se adotarmos um procedimento mais elaborado que analise dados passados e presentes em busca de padrões ou relações existentes nos dados.

No caso de previsão de produção e consumo de energia, é necessário adotar as devidas precauções analíticas, uma vez que este tipo específico de previsão pode ser ainda influenciado por fatores meteorológicos, como temperatura, humidade e velocidade do vento [4], entre outros aspetos, por exemplo: ocupação, horário de funcionamento, tipo da edificação, idade da edificação, e tipologia de equipamento instalado [5].

1.1 MOTIVAÇÃO

O Instituto Politécnico de Leiria (IPLeiria) conta atualmente para suas atividades com aproximadamente 1.600 pessoas e ao redor de 14.500 estudantes, e está sediado na cidade de Leiria em Portugal. Cabe ainda destacar que o IPLeiria é uma instituição pública de ensino superior, que iniciou a sua atividade em 1980 e está presente na região de Leiria e Oeste através das suas cinco escolas superiores, localizadas nas cidades de Leiria (Escola Superior de Educação e Ciências Sociais, Escola Superior de Tecnologia e Gestão (ESTG), e Escola Superior de Saúde), em Caldas da Rainha (Escola Superior de Artes e Design) e na cidade de Peniche (Escola Superior de Turismo e Tecnologia do Mar).

Dentro desse contexto, a previsão de consumo de energia elétrica poderá auxiliar o processo de tomada de decisões de gestão ao permitir o planeamento dos custos com energia elétrica e o desenvolvimento de estratégias para minimizá-los.

Diversas instituições de ensino têm vindo a instalar medidores inteligentes em diferentes edifícios dos seus *campi*, permitindo detalhar o consumo quase em tempo real, dotando essas organizações de significativos volumes de dados com valiosa informação do ponto de vista estratégico. Entretanto, as diferentes características de edifícios poderão ser um facilitador para a coleta de dados e previsão do consumo ou criar alguma dificuldade quanto a esta previsão. Há ainda categorias diferentes: edifícios académicos, administrativos e edifícios residenciais [5].

A idade de prédios e obras arquitetónicas de condicionamento podem não atender aos padrões reais de eficiência energética no que se refere a ar condicionado ou aquecimento do ambiente e iluminação [6]. Nesse contexto, os métodos de aprendizagem computacional, bem como métodos estatísticos, desempenham um papel essencial [7], uma vez que permitem aplicar técnicas de previsão baseadas em dados históricos. Portanto, ao refletir na sustentabilidade do *Campus*, é necessário incluir uma boa previsão do consumo de energia elétrica e de gás, já que poderá viabilizar [7]: **a**) O dimensionamento mais rigoroso de sistemas de produção fotovoltaica em regime de autoconsumo, procurando compatibilizar o consumo com a disponibilidade de produção fotovoltaica; **b**) Uma estimativa de encargos futuros com a energia elétrica; **c**) A adoção de planos de gestão de procura de energia, tentando induzir uma maior flexibilidade da procura em períodos mais críticos ou com maiores penalizações tarifárias.

1.2 OBJETIVOS

Neste trabalho, teve-se como objetivo principal de estudo o desenvolvimento de modelos de previsão de procura de energia elétrica para o dia seguinte, baseados em séries temporais com registos históricos de consumo de energia do *Campus 2* do IPLeiria. Mas sabe-se que existe uma relação entre procura de energia elétrica e condições meteorológicas, em particular da temperatura, podendo este aspeto, entre outros, ser também considerado na previsão de consumo de energia no curto prazo. Assim, neste trabalho foram também desenvolvidos modelos que têm em conta registos históricos da temperatura, obtidos junto do Instituto Português do Mar e da Atmosfera (IPMA), bem como uma caracterização da tipologia de dias (semana / fim de semana) a considerar nesse histórico. Mais concretamente, teve-se também em conta se cada dia considerado no histórico correspondia a um domingo e se correspondia a um feriado.

Tendo em conta este objetivo, as principais contribuições deste trabalho são as seguintes: **1**. Revisão do estado da arte para o cenário acima proposto, incluindo os trabalhos mais relevantes e técnicas para lidar com previsão de consumo de energia para o curto prazo; **2**. Desenvolvimento de modelos estatísticos e de Aprendizagem Computacional (ML, do inglês *Machine Learning*) para previsão de consumo de energia elétrica para o curto prazo.

A abordagem geral adotada neste trabalho, consistiu em: **a**) Examinar diversos modelos estatísticos e de ML e verificar a adequação de tais modelos e técnicas para previsão de curto prazo do consumo; **b**) Comparar o Erro Percentual Médio Absoluto (MAPE, do inglês *Mean Absolute Percentage Error*) e da Raiz do Erro Médio Quadrático (RMSE, do inglês *Root Mean Square Error*) para avaliação da performance dos modelos de previsão de consumo de energia do *Campus* 2 do IPLeiria.

Deste trabalho resultou a publicação "*Previsão de Consumo de Energia para Campi Universitários baseada em IA*" e respetiva apresentação na 5ª Conferencia Campus Sustentável CCS2023, na ESTG do Instituto Politécnico de Viana do Castelo cujo livro de resumos pode ser encontrado em https://prometheus.ipvc.pt/conferences/ccs2023/livro-de-resumos/

1.3 ORGANIZAÇÃO DO RELATÓRIO

Os restantes capítulos do documento estão estruturados da seguinte forma:

No Capítulo 2 é apresentada uma explicação sobre o problema de previsão, contendo: séries temporais, métodos estatísticos, métodos de ML e análise de previsão, e apresenta-se uma revisão da literatura referente ao tema do projeto. O Capítulo 3 descreve a metodologia

adotada e as métricas utilizadas. O Capítulo 4 relata a análise exploratória e a preparação dos dados. Os modelos estatísticos de previsão de consumo são desenvolvidos no Capítulo 5, enquanto que os de aprendizagem computacional se encontram no 6. No Capítulo 7 são apresentados, numa ótica comparativa, os resultados de desempenho dos modelos construídos. Por fim, o Capítulo 8 apresenta a conclusão e perspetivas de trabalho futuro.

2

MATÉRIA RELACIONADA

O propósito deste capítulo é o de abordar as matérias relacionadas a respeito de séries temporais e análise preditiva, bem como os algoritmos usados neste projeto, nomeadamente métodos, ferramentas e desafios relativos à modelação preditiva com séries temporais. Embora haja uma enormidade de outras ferramentas que podem ser utilizadas para tal propósito, que estão amplamente relatadas na literatura, a abordagem se restringe aos algoritmos utilizados para melhor adequação de foco aos objetivos deste projeto, porém, com a necessária informação para melhor entendimento e compreensão do documento.

2.1 SÉRIES TEMPORAIS

Uma série temporal consiste num conjunto de medições, ordenadas ao longo do tempo, em uma determinada quantidade de interesse [8]. Sabe-se que o tempo é um fator importante em vários processos naturais, e é uma variável que tem sua importância nos negócios, na indústria e na medicina. A título de exemplo, temos a quantidade de veículos vendidos nos últimos seis meses, os valores pagos com o consumo de energia no ano anterior, ou a evolução dos valores de diabetes de um paciente nos últimos 12 meses, representam informações que compõem uma determinada série temporal, e o tempo é um elemento natural que está sempre presente quando os dados são coletados [9].

Uma série temporal pode ser representada graficamente como na Figura 1, que regista o desempenho da produção industrial indiana no período de abril de 2005 a novembro de 2014 [10].[1]

1 *Dataset* disponível em https://cran.r-project.org/web/packages/seasonal/index.html

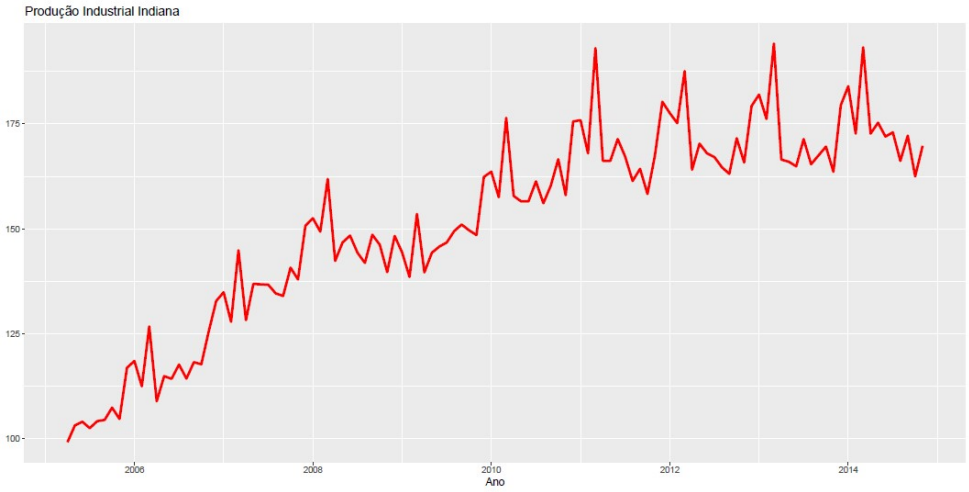


Figura 1: Produção Industrial Indiana (abr/05 a nov/14).

Como exemplos de dados de série temporal, pode-se ainda incluir a volatilidade do preço de uma ação na Euronext, os lucros corporativos trimestrais de uma empresa, e o consumo de energia por pessoa de uma região ou país, como no gráfico apresentado na Figura 2.

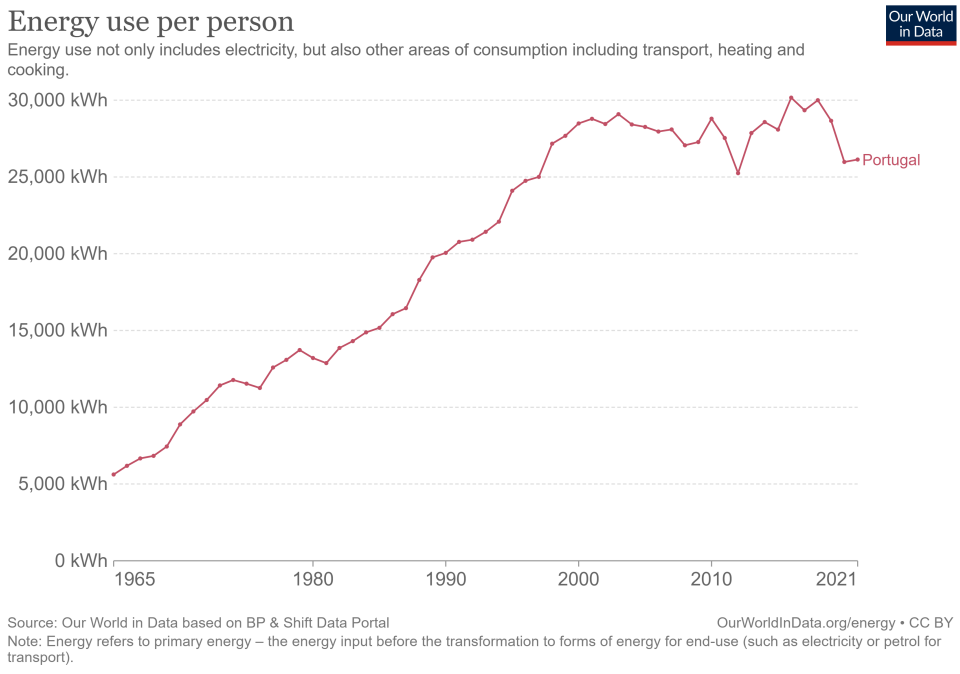


Figura 2: Consumo de Energia em Portugal (per capita) [11].

A monitorização contínua e a coleta de dados têm-se tornado cada vez mais comuns, uma vez que sensores e dispositivos de rastreamento estão em todos os lugares, gerando

uma imensa quantidade de dados sequenciais. Uma vez que séries temporais são excecionalmente significativas porque permite abordar as questões de causalidade, tendências e a probabilidade de resultados futuros, a necessidade de sua análise com técnicas estatísticas e de ML tem experimentado um expressivo aumento, com novos modelos mais promissores que combinam essas metodologias [12].

2.1.1 *Análise de Séries Temporais*

As séries temporais são compostas de padrões e o ponto de partida é realizar a sua decomposição para identificar os seus componentes, a saber: **a**) *Tendência* - comportamento de longo prazo da série; **b**) *Sazonalidade* - movimento sistemático, não necessariamente regular [13]; **c**) *Variações cíclicas ou ciclos* - um ciclo ocorre quando os dados exibem aumentos e quedas que não são de uma frequência fixa. Se as flutuações não tiverem uma frequência fixa, elas são cíclicas; se a frequência for imutável e associada a algum aspeto do calendário, o padrão é sazonal; **d**) Flutuações inexplicáveis, resultado de fatos fortuitos e inesperados como catástrofes naturais, atentados terroristas, decisões intempestivas de governos, pandemias, entre outros [4, 8].

Pode verificar-se na série temporal da produção industrial indiana [10] na Figura 1, a existência de tendência, sazonalidade e alguma irregularidade e, destaca-se na Figura 3, a representação gráfica da sazonalidade, acentuada nos meses de março, ao longo dos anos.

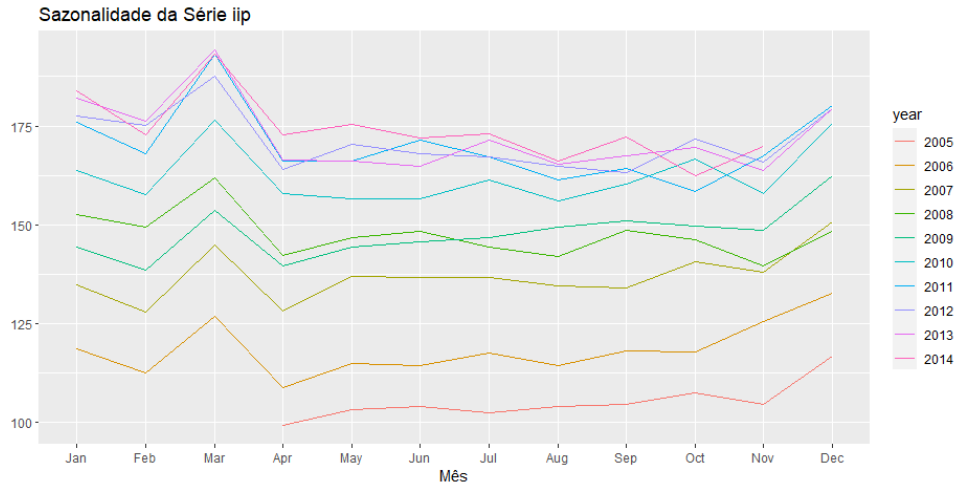


Figura 3: Sazonalidade da produção industrial indiana (abr/05 a nov/14).

A análise de séries temporais trabalha com o tempo e é baseada em dados para fazer previsões sobre o futuro, medindo o período de tempo em anos, estações, meses, dias, horas, minutos, segundos ou qualquer outra unidade de tempo adequada [9]. Essa análise

tenta extrair um resumo significativo e informações estatísticas a fim de diagnosticar comportamentos passados e predizer comportamentos futuros [12].

2.1.2 *Métodos de Previsão*

Os métodos de *previsão*, como por exemplo do consumo de energia elétrica, são identificados como clássicos e modernos da seguinte forma: *1*. Os métodos clássicos de previsão baseiam-se na teoria da regressão e na análise estatística (regressão, modelos autoregressivos) e métodos de previsão probabilística; *2*. Os métodos modernos de previsão englobam algoritmos de ML e, dependendo do horizonte temporal, diferentes abordagens podem ser utilizadas [2].

2.2 MODELOS ESTATÍSTICOS

Os modelos estatísticos baseiam-se em métodos de regressão linear, mas respondem pelas correlações que surgem entre pontos de dados, podendo revelar a dinâmica de uma série temporal e as estatísticas que descrevem o ruído e a incerteza de seu comportamento [12].

2.2.1 *ARIMA-SARIMA-SARIMAX*

ARIMA (*Autoregressive Integrated Moving Average*), SARIMA (*Seasonal Autoregressive Integrated Moving Average*) ou SARIMAX (*Seasonal Autoregressive Integrated Moving Average with Exogenous Regressors*) são modelos de séries temporais que podem ser usados para prever valores futuros de uma série temporal [14, 15].

Os modelos SARIMA e SARIMAX usam: Modelos autorregressivos (AR) para capturar a tendência da série temporal, com a notação **p** e **P**, respetivamente; Modelos integrados (I) para remover a sazonalidade, cuja notação é **d** e **D**, respetivamente; Modelos de média móvel (MA) para capturar o ruído, com a notação **q** e **Q**, baseados na suposição de que os valores atuais da série temporal são influenciados pelos valores anteriores; Um modelo unitário e outro sazonal ***m***, respetivamente; O modelo I é baseado na suposição de que a série temporal é estacionária, ou seja, que as características estatísticas da série temporal não mudam com o tempo. O modelo MA é baseado na suposição de que os valores atuais da série temporal são influenciados por erros de previsão anteriores [16].

ARIMA

O processo ARIMA(p, d, q) assume que o valor presente depende de valores passados, vindos da componente autorregressiva (AR(p)), e erros passados vindos da componente de médias móveis (MA(q)). No entanto, em vez de usar a série original, indicada como y_t, o processo ARIMA usa a série diferenciada, indicada como y'_t [14].

No ajustamento de um modelo ARIMA a um conjunto de dados de séries temporais, pode-se adotar o procedimento como na Figura 4:[2]

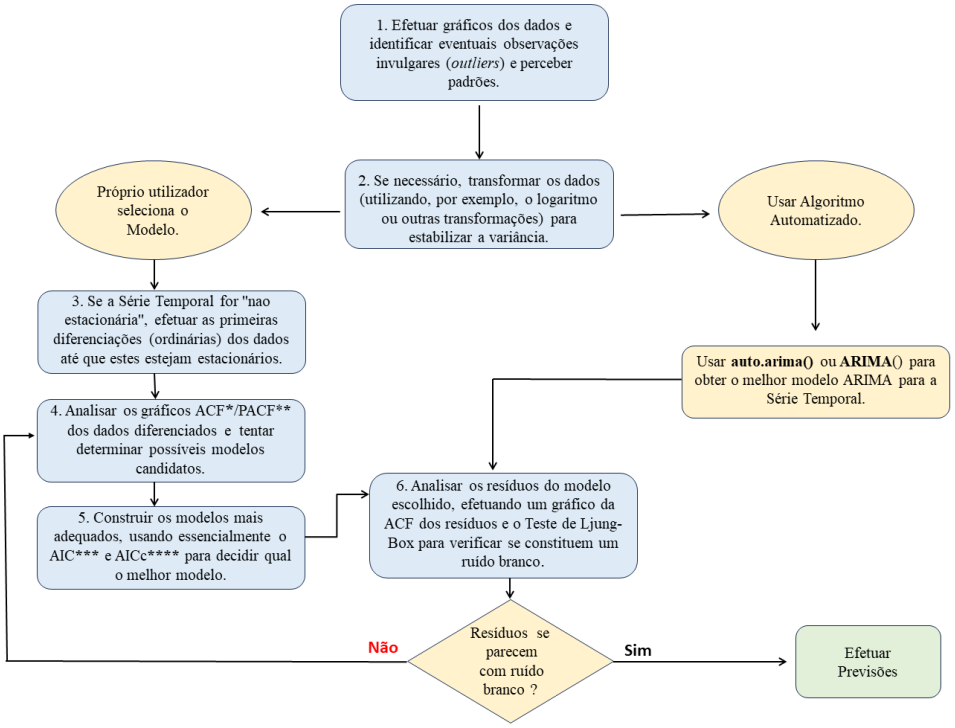


Figura 4: *Pipeline* para modelação ARIMA, baseado em [4, 14, 17].

O procedimento acima ilustrado, envolve três importantes etapas [18]:

1. Examinar a série temporal para entender suas características e identificar a tendência, a sazonalidade e o ruído (variações aleatórias) e observar se é necessário fazer uma diferenciação[3] para tornar a série estacionária, o que é fundamental para aplicar um modelo ARIMA [19].

2

1. * Função de Autocorrelação
2. ** Função de Autocorrelação Parcial
3. *** Critério de Informação de *Akaike*
4. **** Critério de Informação de *Akaike* corrigido

3 Diferenciação é converter uma série temporal de valores em uma série temporal de mudanças nos valores ao longo do tempo. Na maioria das vezes isso é feito calculando diferenças entre pares de pontos adjacentes no

2. Encontrar os coeficientes que minimizam o erro entre as previsões do modelo e os valores reais da série temporal. A estimação dos parâmetros é geralmente realizada usando técnicas de otimização [19].

3. Efetuar uma análise dos resíduos do modelo. Os resíduos devem ser aleatórios, não apresentar padrões e não mostrar autocorrelações significativas [20].

SARIMA

Por sua vez, o modelo SARIMA é formado pela inclusão de novos termos no modelo ARIMA(p, d, q) de um conjunto de parâmetros[4] $(P, D, Q)_m$, que nos permite levar em conta padrões periódicos ao prever uma série temporal, o que nem sempre é possível com um modelo ARIMA(p, d, q). Há quatro novos parâmetros no modelo: P, D, Q e ***m***. Os três primeiros têm o mesmo significado que no modelo ARIMA(p, d, q), sendo dessa forma suas contrapartes sazonais [14]. O parâmetro ***m*** representa a frequência dos dados. No contexto de uma série temporal, a frequência é definida como o número de observações por ciclo, e a duração do ciclo depende do conjunto de dados [22].

SARIMAX

Por fim, o modelo SARIMAX tem dois tipos de variáveis: a variável alvo (*endógena*) e variáveis externas (*exógenas*), pois é possível que variáveis externas também tenham impacto em séries temporais e podem, portanto, ser bons preditores de valores futuros. Um modelo SARIMAX estende ainda mais o modelo SARIMA(p, d, q)$(P, D, Q)_m$ ao qual se pode adicionar qualquer número de variáveis exógenas X_t [14].

2.2.2 *Otimização de Parâmetros*

A obtenção de parâmetros para modelos ARIMA-SARIMA-SARIMAX, também pode ser baseada em uma ferramenta automatizada de seleção de parâmetros por meio da função *auto_arima*() [23] da biblioteca *pmdarima* para modelos SARIMA e da função *sarimax*() [24] da biblioteca *statsmodels* tanto para modelos SARIMA quanto modelos SARIMAX.

O processo de seleção automática de hiperparâmetros do modelo ARIMA envolve a busca por diferentes combinações e a avaliação do desempenho do modelo em dados históricos, com a utilização desses parâmetros [25]. O objetivo é diminuir a complexidade e aumentar a qualidade do ajuste, e o modelo com o valor mínimo do AIC geralmente é o melhor para previsão [12]. Entretanto, o uso da função *auto_arima*() não garante a seleção

tempo, de modo que o valor da diferença série no tempo t é o valor no tempo t menos o valor no tempo t–1 [12].

4 O modelo **SARIMA** faz uso da notação em minúsculas para as partes não sazonais do modelo e a notação maiúscula para as partes sazonais do modelo [21].

do melhor modelo em todos os casos. É sempre recomendável examinar e validar o modelo selecionado e ajustá-lo conforme necessário para garantir que ele capture adequadamente as características da série temporal em questão [25].

2.3 SUPOSIÇÕES NA CONSTRUÇÃO DE MODELOS

2.3.1 *Autocorrelação*

Autocorrelação significa a correlação de uma variável com valores desfasados (com diferenças no tempo) dela mesmo. Se a variável x_t (t medido em anos) tem correlação sistematicamente com seu valor no ano anterior (a correlação entre x_t e x_{t-1} não é nula), diz-se então, que x_t é uma variável autocorrelacionada [26].

Quando se identifica a presença de autocorrelação, tem-se em primeiro lugar de verificar qual é a causa da autocorrelação. Se o problema é de especificação do modelo, ele pode ser corrigido com a inclusão de mais variáveis ou com a alteração da forma funcional (parâmetros). Se não é este o caso, ou seja, a autocorrelação é uma "parte integrante" do modelo estimado, a correção passa pelo conhecimento prévio de como é a estrutura da autocorrelação [26], e a presença de autocorrelação pode ser identificada traçando-se os valores observados da função de autocorrelação (ACF) para uma determinada série temporal [22].

2.3.2 *Normalidade*

Uma distribuição é uma função estatística que representa a probabilidade para todos os valores possíveis de um determinado valor ser gerado por um processo. A distribuição normal, também chamada de Gaussiana, é comummente usada porque descreve muitos fenómenos, sendo caracterizada por dois parâmetros: média μ e desvio padrão σ [27], desempenha um papel importante nas estatísticas, e muitos procedimentos práticos baseiam-se na suposição de que os dados da amostra são normalmente distribuídos [8].

Nem todos os conjuntos de dados ou séries temporais seguem uma distribuição normal. Isso pode ter implicações nas previsões de modelos estatísticos ou de ML, e os resultados desses modelos podem estar enviesados ou as estimativas dos parâmetros podem ser imprecisas [12], e em alguns casos, podem-se aplicar transformações nos dados para torná-los mais próximos de uma distribuição normal [8].

2.3.3 *Estacionariedade*

Séries temporais com tendências, ou com sazonalidade, não são estacionárias - a tendência e a sazonalidade afetarão o valor da série temporal em momentos diferentes, ou seja: o comportamento passado da série temporal, reflete ou afeta o comportamento futuro da série temporal. Por outro lado, uma série de ruído branco[5] é estacionária – não importa quando seja observada, ela deve parecer a mesma em qualquer momento. Uma série temporal estacionária não terá padrões previsíveis no longo prazo [4].

Um modelo de série temporal não estacionária sofrerá variações em relação ao seu desempenho ao mesmo tempo que as estatísticas da série variam. Ou seja, uma série temporal com uma média e variância não estacionárias, o viés e o erro no modelo variarão ao longo do tempo, tornando o modelo questionável [12]. Um processo não estacionário é transformado em estacionário usando o método de diferenciação, e a ordem d ou D desse processo será o número de vezes que o processo foi diferenciado antes de atingir a estacionariedade [28].

2.3.4 *Transformações*

Muitas vezes, as transformações também são usadas para melhorar a aproximação da normalidade ou para melhorar a linearidade na previsão do valor de uma série temporal [29]. As transformações são usadas para tornar uma série estacionária. A diferenciação pode estabilizar a tendência e a sazonalidade, enquanto os logaritmos estabilizam a variância [14].

Ajustar os dados históricos muitas vezes pode levar a uma tarefa de previsão mais simples. Há quatro tipos de ajustes que podem ser utilizados: ajustes de calendário, ajustes populacionais, ajustes de inflação e transformações matemáticas. O objetivo destes ajustes e transformações é simplificar os padrões nos dados históricos, removendo fontes conhecidas de variação ou tornando o padrão mais consistente em todo o conjunto de dados. Padrões mais simples geralmente levam para previsões mais precisas [4].

Outra possível transformação para resolver um problema difícil de previsão é o *resampling* (reamostragem). Há situações em que a reamostragem dos dados fará sentido e permitirá que se criem melhores modelos de previsão, pois ao se alterar a frequência de amostragem dos dados poderá ser benéfico para o armazenamento e o processamento. Há dois tipos de reamostragem: **a**) *Downsampling* que vem a ser um subconjunto de dados

5 As séries temporais que não apresentam autocorrelação são chamadas de ruído branco [4].

com a frequência mais baixa do que a série original; **b**) *Upsampling* é a representação de dados como se eles fossem coletados com uma frequência mais alta do que se verifica na realidade [12].

2.3.5 *Análise de Resíduos*

Os "resíduos" em um modelo de série temporal são o que resta após o ajuste, e calculam-se através da diferença entre as observações e os valores ajustados correspondentes [21].

A Equação (1) mostra os termos do resíduos:

$$\varepsilon_t = y_t - \hat{y}_t \tag{1}$$

onde,

- t é a unidade de tempo,
- ε_t são os resíduos,
- y_t é a observação no tempo t,
- $\hat{y}_t$ é a previsão no tempo t.

Os resíduos são úteis para verificar se um modelo capturou adequadamente as informações nos dados. Um bom método de previsão produzirá resíduos com as seguintes propriedades: **a**) Não são correlacionados. Se houver correlações entre os resíduos, haverá informações deixadas nos resíduos que devem ser usadas no cálculo das previsões; **b**) Têm média nula. Se os resíduos tiverem uma média não nula, as previsões serão enviesadas [21].

Erros de especificação do modelo, omissão de variável explicativa relevante e dessazonalização de séries, podem levar à autocorrelação dos resíduos, que gera estimadores dos parâmetros não enviesados, porém ineficientes, e erros-padrão dos parâmetros subestimados, o que acarreta problemas com os testes de hipótese das estatísticas t [30].

Idealmente, os resíduos de um modelo terão características semelhantes ao ruído branco, o que significaria que qualquer diferença entre os valores previstos e reais é devida à aleatoriedade. Portanto, os resíduos devem ser não correlacionados e distribuídos independentemente [14].

Há dois aspetos na análise residual: uma análise qualitativa e uma análise quantitativa. A análise qualitativa se concentra no estudo do gráfico Q-Q[6], enquanto a análise quantitativa determina se os resíduos não estão correlacionados [14].

Análise Qualitativa dos Resíduos:

Antes de efetuar uma previsão, é necessário que o primeiro passo seja uma análise de resíduos através do estudo do gráfico *quantil-quantil* (gráfico Q-Q) pela sua visualização gráfica, sendo uma ferramenta útil para verificar a hipótese de que os resíduos do modelo são normalmente distribuídos.

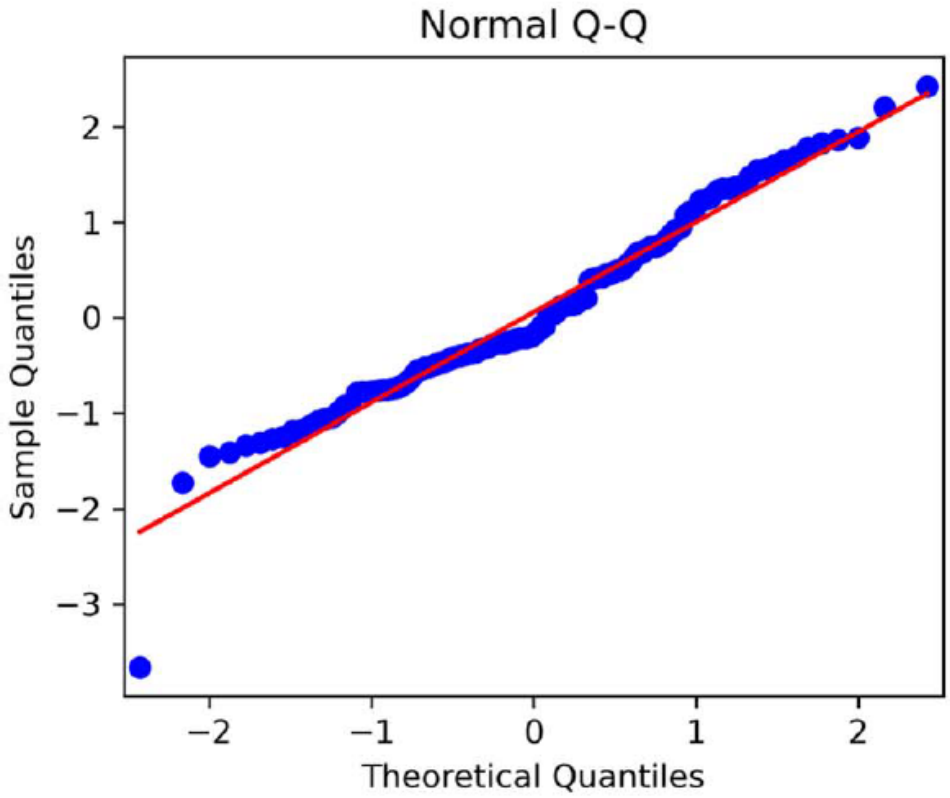


Figura 5: Q-Q *Plot*.

Na análise procura-se comparar a distribuição dos resíduos com uma distribuição normal e verificar se os resíduos são uma coleção de variáveis não correlacionadas [16]. Se ambas as distribuições forem semelhantes, o gráfico Q-Q exibirá uma linha reta na cor azul como na Figura 5, indicando que, aproximadamente, $y = x$. Isso, por sua vez, significa que o modelo tem um bom ajuste para os dados [14].

6 O gráfico Q-Q é construído desenhando os quantis dos resíduos no eixo y contra os quantis de uma distribuição teórica, neste caso a distribuição normal, no eixo x. Isso resulta em um gráfico de dispersão.

Análise Quantitativa dos Resíduos:

Após análise do gráfico Q-Q e determinar que os resíduos são aproximadamente normalmente distribuídos, é indicado aplicar o teste *Ljung-Box* para investigar se os resíduos estão correlacionados [14].

2.3.6 *Testes Estatísticos*

Shapiro Wilk

O teste de *Shapiro-Wilk* analisa se o modelo segue distribuição normal ajustando-se às observações e, sendo um teste não paramétrico, não faz nenhuma suposição sobre a forma da distribuição da população. Embora possa ser considerado um teste poderoso para a normalidade, pode não o ser totalmente para séries temporais, já que estas podem ser afetadas por tendências, sazonalidades e outros fatores que podem distorcer a distribuição dos dados e as estatísticas podem não refletir a distribuição real dos dados. Entretanto, é necessário usar outros testes para verificar a normalidade de séries temporais [16, 31].

Jarque-Bera

O teste de *Jarque-Bera* avalia a hipótese nula de que os dados seguem uma distribuição normal, contra a hipótese alternativa de que os dados não seguem uma distribuição normal. Para uma distribuição normal, a assimetria da amostra deve estar próxima de zero e a curtose da amostra deve estar próxima de três. O teste de *Jarque-Bera* determina se a assimetria e a curtose da amostra são incomummentes diferentes dos seus valores esperados [32].

Dickey-Fuller e KPSS

O teste *Dickey-Fuller* Aumentado (ADF) e o teste *Kwiatkowski-Phillips-Schmidt-Shin* (KPSS) são as métricas comummente usadas para avaliar uma série temporal quando se trata de problemas de estacionariedade [4]. Em Python, através do pacote *statsmodels* se pode testar a estacionariedade de uma série temporal [33].

Ljung-Box

Na previsão de séries temporais, aplica-se o teste *Ljung-Box* nos resíduos do modelo para testar se eles são semelhantes ao ruído branco. A hipótese nula do teste *Ljung-Box* afirma que os dados são distribuídos independentemente, o que significa que não há autocorrelação. Se o *p-value* for maior que o nível de significância, normalmente 0.01 ou 0.05, não se rejeita a hipótese nula, o que significa que os resíduos não evidenciam

serem dependentes. Portanto, não há autocorrelação, os resíduos são semelhantes ao ruído branco e o modelo pode ser usado para previsão [16].

2.4 MODELOS DE APRENDIZAGEM COMPUTACIONAL

O uso de aprendizagem computacional (ML, do inglês *machine learning*) tornou-se nos últimos anos omnipresente no dia a dia da sociedade contemporânea, através das recomendações automáticas de quais filmes assistir, que comida pedir, que produtos comprar, que música ouvir, etc. Muitos *sites* e dispositivos modernos têm algoritmos de ML em seu *core business* [34]. Há algum tempo que o uso de ML na previsão do consumo de energia, bem como na análise de procura por energia, é reportado na literatura [35]. Convém mencionar que os métodos de ML são divididos em aprendizagem supervisionada, não supervisionada e aprendizagem por reforço [36]. Neste trabalho foram utilizados apenas métodos de aprendizagem supervisionada, que serão descritos na próxima secção.

2.4.1 *Aprendizagem Supervisionada*

A Aprendizagem Supervisionada (AS) é utilizada para prever um determinado resultado, como exemplos de pares de entrada e saída. Em um modelo de ML o objetivo é fazer previsões para dados novos e nunca antes vistos. A AS é um dos tipos de aprendizagem mais usados e bem-sucedidos de ML [34].

Existem dois tipos principais de problemas de AS, denominados de classificação e regressão. Na classificação, o objetivo é classificar um rótulo de classe, que é uma escolha de um conjunto predefinido de uma lista de possibilidades. Um exemplo comum de AS é a classificação de imagens. Enquanto que na regressão, o objetivo é prever um valor numérico a partir de uma ou mais variáveis de entrada. Por exemplo, um algoritmo de regressão pode ser usado para prever o preço de uma casa com base em características como o número de quartos, a localização, a idade da casa, etc [37].

Em resumo, a AS é uma abordagem importante em ML, pois permite que os algoritmos aprendam a partir de exemplos rotulados e possam ser usados para prever rótulos em novos dados nunca antes vistos [37].

2.4.2 *Aprendizagem não Supervisionada (AnS)*

Este ramo da aprendizagem de máquina consiste em encontrar padrões nos dados de entrada sem a ajuda de quaisquer alvos, para fins de visualização de dados, agrupamento, deteção de valores discrepantes, ou para entender melhor as correlações presentes nos dados [17].

Uma aplicação comum de AnS é a redução de dimensionalidade, que transforma uma representação com maior dimensionalidade, consistindo em muitas características, e encontra uma nova forma de representar esses dados que resume as características essenciais. Nesse caso, o objetivo do algoritmo de AnS consiste em criar um modelo que usa um vetor de características como dados de entrada e o transforma em outro vetor que tem menos características do que os dados iniciais. O agrupamento é útil para localizar grupos de objetos semelhantes em uma grande coleção de objetos, como imagens ou documentos de texto. Na deteção de valores discrepantes, a saída é um número real que indica como o vetor de recursos de entrada é diferente de um exemplo típico no conjunto de dados [38].

2.4.3 *Redes Neuronais Artificiais*

As Redes Neuronais Artificiais (RNA) são um grupo interconectado de unidades de processamento inspirados em neurónios biológicos. Estas unidades, também designadas por neurónios, são frequentemente dispostas em camadas consecutivas [17].

2.4.4 *Multilayer Perceptron (MLP)*

Uma das arquiteturas de RNA mais disseminadas é conhecida por *Perceptrons* Multicamada (MLP, do inglês *Multilayer Perceptron*). Os MLP são redes neuronais *feed-forward network* (FFN) porque as entradas são processadas apenas na direção direta através de vários nós de entrada, até que cheguem ao nó de saída. A rede pode ou não ter camadas de nós ocultas. Esse tipo de rede é uma das variantes mais simples das redes neuronais [39]. A Figura 6 ilustra arquitetura típica de uma MLP.

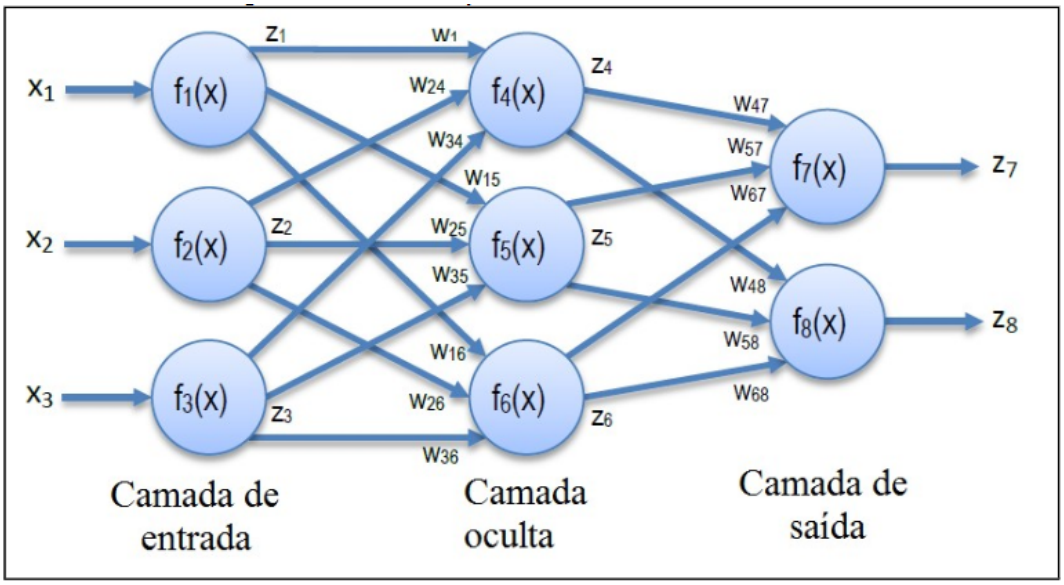


Figura 6: Modelo Simplicado de RNA [40].

Na Figura 7, pode observa-se a estrutura típica de um neurónio [41].

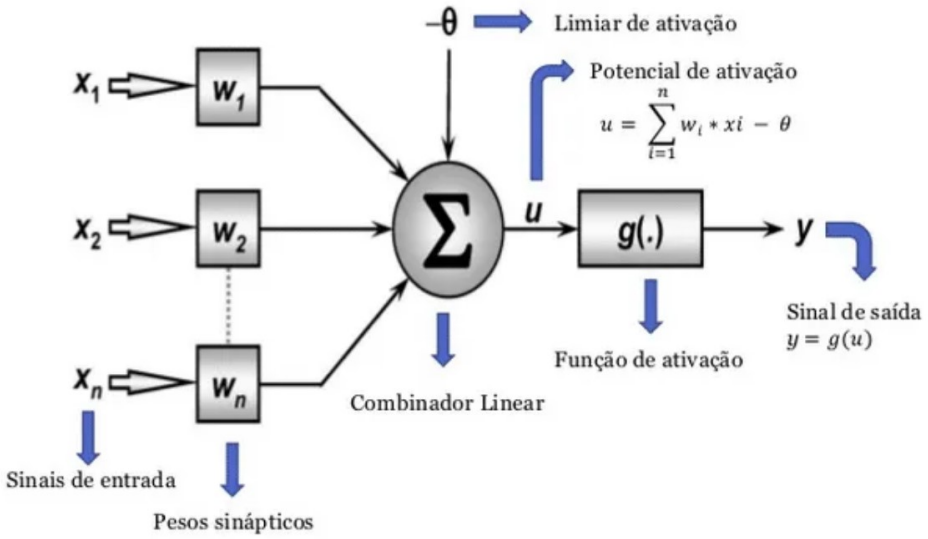


Figura 7: Rede neuronal artificial com um neurónio [42].

A Figura 7 mostra, simplificadamente [42]:

1. *Inputs* $X_1, X_2, ..., X_n$ - correspondem aos dados de entrada;
2. *Pesos sinápticos* $W_1, W_2, ..., W_n$ - são os valores para ponderar os sinais e dados de cada entrada da rede;

3. *Combinador Linear* $\sum$ - efetua a agregação de todos os sinais de entradas que foram ponderados pelos respetivos pesos sinápticos a fim de produzir um potencial de ativação;
4. *Limiar de ativação* Θ - especifica qual será o patamar apropriado para que o resultado produzido pelo combinador linear possa gerar um valor de disparo de ativação;
5. *Potencial de ativação* **u** - resultado obtido pela diferença do valor produzido entre o Combinador Linear e o limiar de ativação;
6. *Função de ativação* **g** - tem por objetivo limitar a saída de um neurónio em um intervalo de valores. Se o valor for positivo, ou seja, se **u** for maior ou igual a zero, então o neurónio produz um potencial excitatório; caso contrário, o potencial será inibitório;
7. *Sinal de saída* **y** - é o valor final de saída (podendo ser usado como entrada de outros neurónios que estão sequencialmente interligados).

O treino das redes neuronais é realizado de forma iterativa. Em cada iteração, é calculada a saída da rede para as amostras do conjunto de treino. Com base nesta saída, é calculado o gradiente da função de erro tendo em conta a saída desejada (isto é, o rótulo associado à amostra) e a saída verificada. Os gradientes nas camadas intermédias são calculados retropropagando o gradiente na camada de saída utilizando o algoritmo *Backpropagation*. Este algoritmo é uma solução usada para treinar rede neuronais com camadas ocultas de neurónios que envolve o conceito de gradiente de erro e descida de gradiente [43].

O método de descida de gradiente procura uma direção de descida usando uma pesquisa de linha exata ou uma pesquisa de linha de retrocesso. Seu objetivo é encontrar um mínimo de uma função. No caso das redes neuronais, pretende-se encontrar a configuração de pesos da rede que minimiza a função de erro [44].[7]

2.4.5 *Redes Neuronais Recorrentes*

Uma Rede Neuronal Recorrente (RNN) é utilizada para resolver problemas de ML que envolvem sequências de entradas. As RNNs têm conexões que possuem *loops*, adicionando *feedback* e memória às redes, ao longo do tempo. A memória permite que esse tipo de rede aprenda e generalize através de sequências de entradas, em vez de padrões individuais [35].

7 A função de erro é utilizada para avaliar ou medir a discrepância entre uma previsão e o valor real.

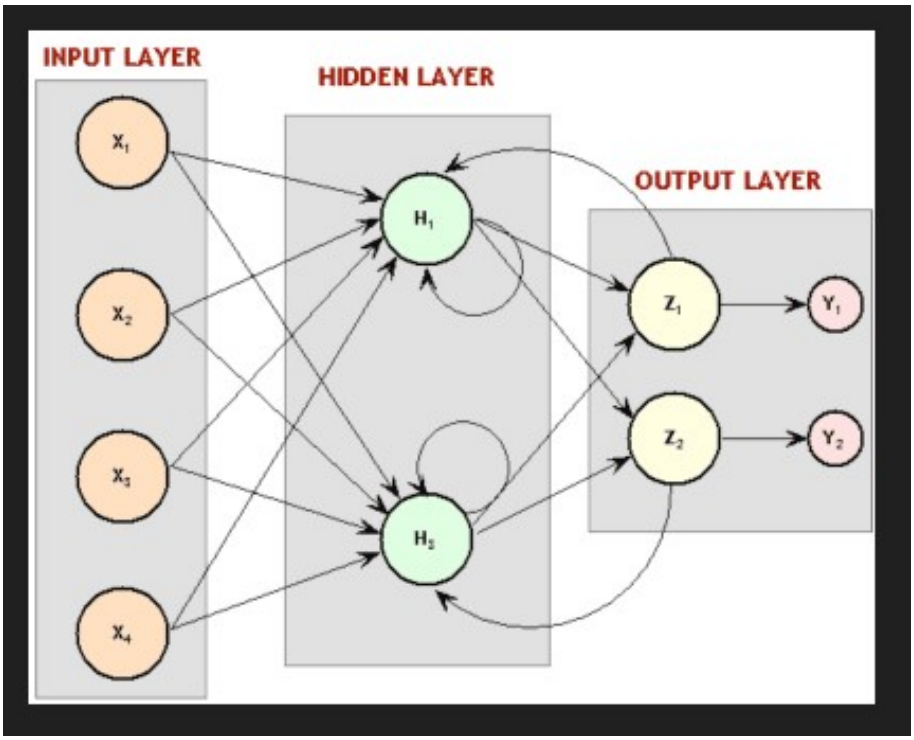


Figura 8: Modelo Simplificado de RNN [45].

Cada neurónio pode passar o seu sinal de saída para posterior utilização como uma das entradas, além de encaminhar para a próxima camada. A saída da rede pode realimentar como uma entrada para a rede com a próxima entrada, e assim por diante. As conexões recorrentes adicionam estado ou memória à rede e permitem que ela aprenda eventuais padrões existentes nas sequências de entrada [35]. Uma RNN se lembra de todas as informações ao longo do tempo e é, por isso, utilizada em problemas de previsão que envolvam séries temporais.

2.4.6 *Long Short-Term Memory (LSTM)*

As redes *Long Short-Term Memory* (LSTM) [45] são redes recorrentes capazes de associar memórias e entradas remotas no tempo. Esse tipo de arquitetura de RNN é, por isso, particularmente útil em tarefas que envolvem dependências de longo prazo.

Uma rede LSTM contém células de memória que podem reter informações por longos períodos de tempo, bem como portas que controlam o fluxo de informações dentro e fora das células. Isso permite que a rede esqueça ou lembre-se seletivamente de informações conforme necessário, o que é crucial para o processamento preciso de dados sequenciais. A abordagem da memória de curto e longo prazo inclui um portão de esquecimento

(*forget-gate*) que permite que se treinem os neurónios individuais sobre o que é importante e quanto tempo ele permanecerá importante.

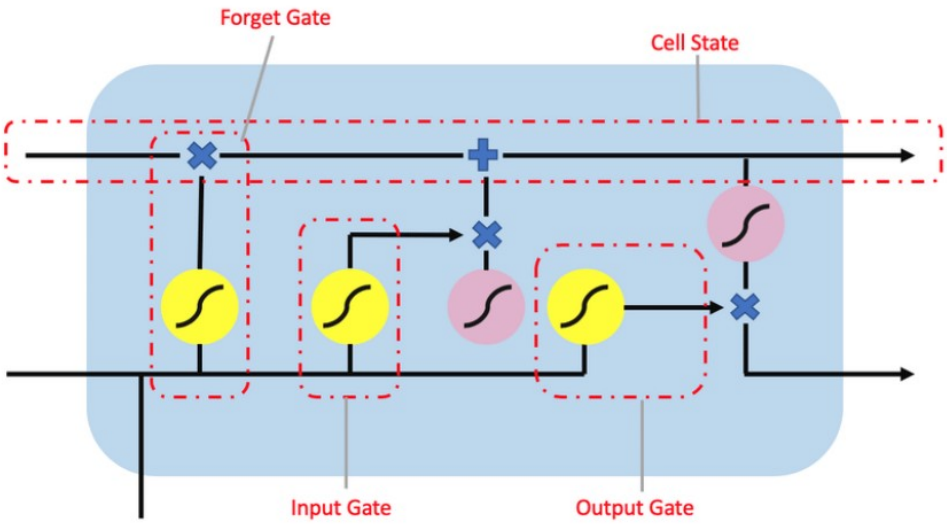


Figura 9: Modelo Simplificado de LSTM [46].

O diagrama na Figura 9, ilustra como os dados fluem através de uma célula de memória e são controlados por seus *gates* /portões. Esses portões atuam sobre os sinais que recebem e, de forma semelhante aos nós da rede neuronal, bloqueiam ou transmitem informações com base em sua força e importância, que filtram com seus próprios conjuntos de pesos.

Esses pesos são ajustados através do processo de aprendizagem das redes recorrentes. Ou seja, as células aprendem quando permitir que os dados entrem, saiam ou sejam excluídos através de um processo iterativo, calculando o erro durante o processo de aprendizagem e ajustando os pesos através da descida do gradiente [45].

2.4.7 *Gated Recurrent Unit (GRU)*

As redes *Gated Recurrent Unit* (GRU) são uma geração mais recente de redes neuronais recorrentes e são bastante semelhantes a uma rede LSTM [46].

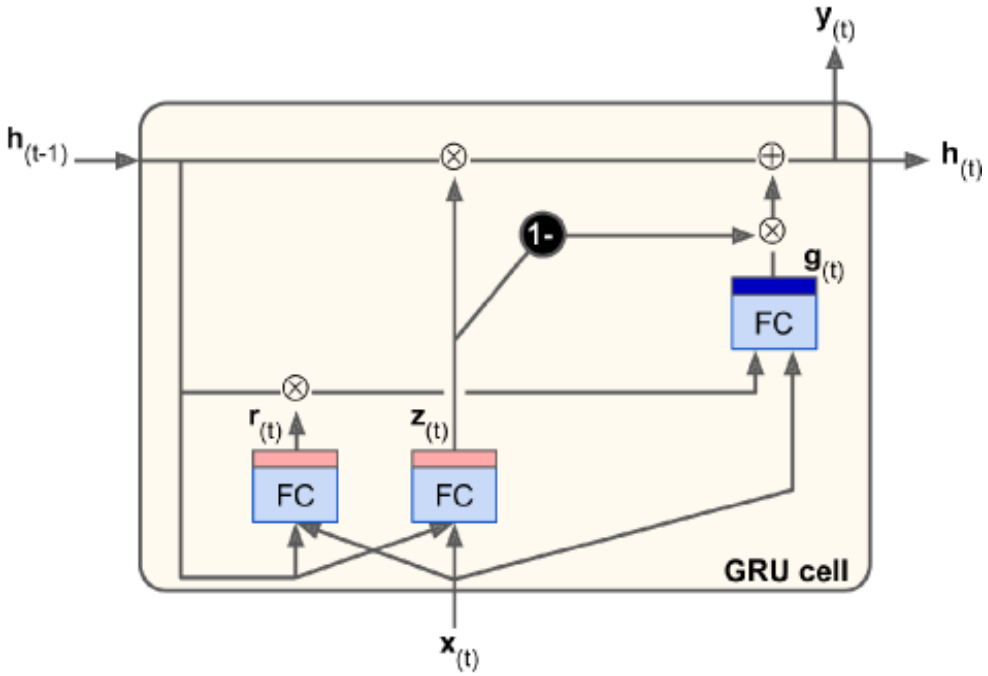


Figura 10: Modelo Simplificado de GRU [36].

A GRU[8] tem apenas dois portões, um portão de *reset* e um portão de atualização. Estas são as principais características: **a**) Um único controlador de porta $z_{(t)}$ controla tanto a porta de esquecimento quanto a porta de entrada. Se o controlador de portão gera um 1, o portão de esquecimento está aberto (= 1) e o portão de entrada é fechado (1–1 = 0). Se a saída for 0, ocorre o oposto. Em outras palavras, sempre que uma memória deve ser armazenada, o local onde ela será armazenada é apagado primeiro. Na verdade, essa é uma variante frequente da célula LSTM por si só. **b**) Não há porta de saída; o vetor de estado completo é gerado a cada passo de tempo. No entanto, há um novo controlador de portão $r_{(t)}$ que controla qual parte do anterior estado será apresentado à camada principal $g_{(t)}$ [36].

2.4.8 *k-Nearest Neighbors (KNN)*

O KNN é um algoritmo de aprendizagem não paramétrica, isto é, não faz suposições sobre a forma da relação ou padrão de comportamento dos dados. Ao contrário de outros algoritmos de aprendizagem que permitem descartar os dados de treino após a construção do modelo, o KNN mantém todos os exemplos de treino na memória. Uma vez que um novo exemplo x inédito chega, o algoritmo KNN encontra os k exemplos de treino mais

8 FC = *fully connected layer*

próximos de x e devolve o rótulo maioritário (caso se trate de um problema de classificação) ou o rótulo médio (no caso de regressão) [38].

2.4.9 *Árvores de Decisão*

As *Árvores de Decisão* (ADs) são modelos amplamente utilizados para tarefas de classificação e regressão, aprendendo uma hierarquia de perguntas que levam a uma decisão [34]. Nesse algoritmo, um modelo em forma de árvore é construído, onde cada nó representa uma decisão baseada em uma característica, e cada ramo representa o resultado dessa decisão.

O algoritmo funciona particionando recursivamente os dados em subconjuntos cada vez menores com base nos valores dos recursos, até que um critério de paragem seja atendido. Se a árvore for muito complexa pode gerar *overfitting* pelo facto de ser sensível a ruído nos dados [22, 47].

A Figura 11 mostra uma *árvore de decisão* onde cada nó contém um teste para um atributo.

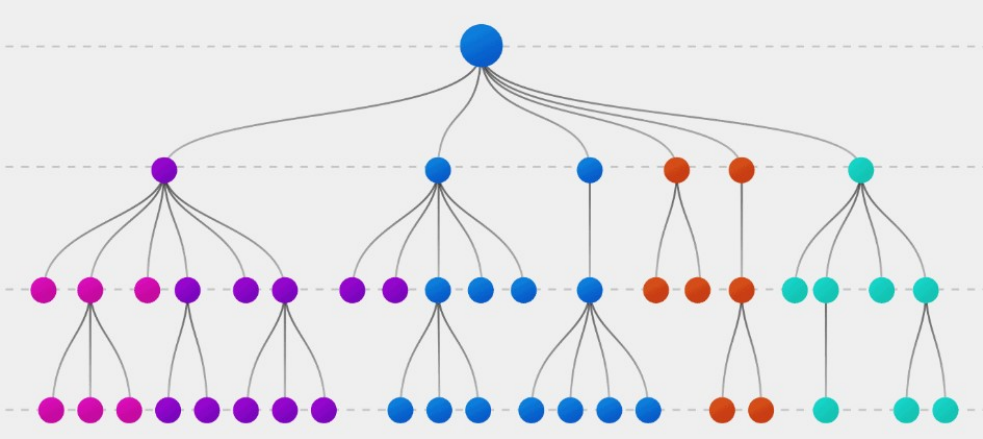


Figura 11: Diagrama de Árvores de Decisão [48].

Overfitting pode ocorrer quando o modelo se torna tão adaptado ao conjunto de treino que começa a memorizar os dados de treino em vez de aprender a identificar padrões e tendências subjacentes que podem ser generalizadas para novos dados. O resultado é que o modelo pode apresentar um desempenho excelente nos dados de treino, mas falhar em novos dados, levando a resultados imprecisos e prejudicando sua utilidade prática [22].

2.4.10 *Random Forest (RF)*

Ao se agrupar um conjunto de ADs tem-se o que é denominado de *Random Forest* (RF), que faz uso do método *Ensemble*. As previsões de todas as árvores individuais são obtidas, para em seguida, prever a classe que obtém a maioria dos votos, conforme ilustrado na Figura 12. Apesar de sua simplicidade, este é um poderoso algoritmo de ML [36].

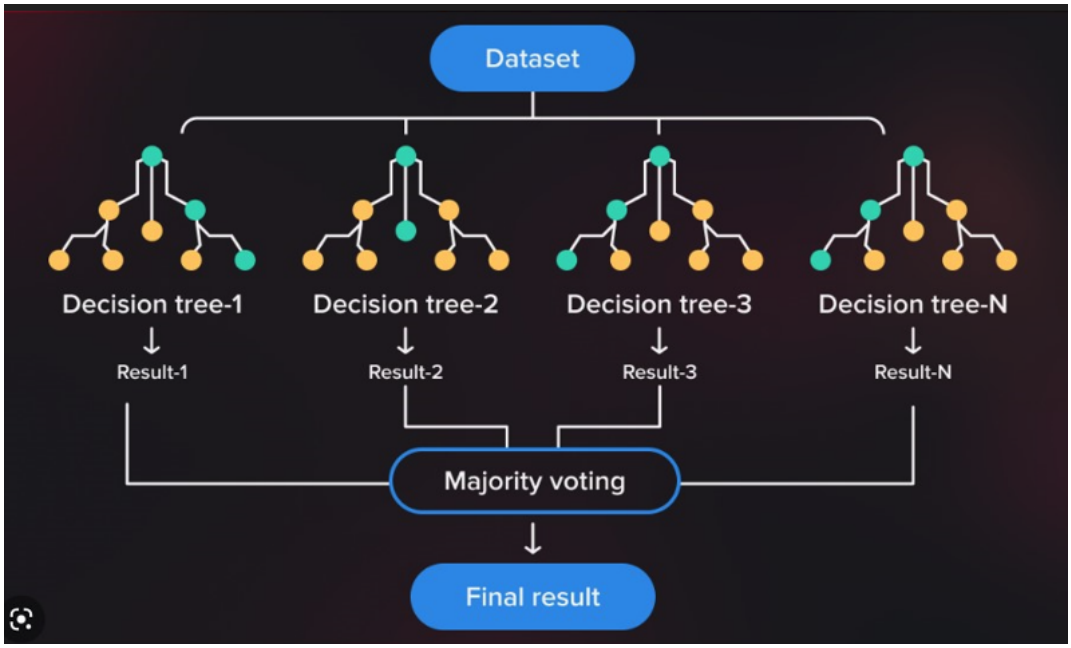


Figura 12: Diagrama Simplificado *Random Forest* [48].

Existem dois métodos principais para criar *Ensembles*: *bagging* e o *boosting* [22].

Bagging de preditores é um método para gerar várias versões de um preditor e usá-las para obter um preditor agregado. A agregação calcula a média das versões ao prever um resultado numérico e faz um voto de pluralidade ao prever uma classe [49].

Boosting (originalmente denominado de *boosting* de hipóteses) é outra maneira de construir um conjunto de preditores. *Boosting* cria modelos sequencialmente com a ideia de que os posteriores devem corrigir os erros dos anteriores e que os dados desajustados pelos anteriores devem ser mais fortemente ponderados por modelos posteriores [12].

2.4.11 *Extreme Gradient Boosting (XGBoost)*

O XGBoost é uma biblioteca otimizada de *boosting*, fazendo uso de algoritmos de ML sob a estrutura *Gradient Boosting*. Tal como outros algoritmos de *boosting*, o XGBoost usa

ADs para seu modelo de *ensemble*, criando um conjunto de modelos e combinando suas previsões para produzir uma previsão final [50].

Um exemplo de fluxograma do *XGBoost* pode ser visto na Figura 13.

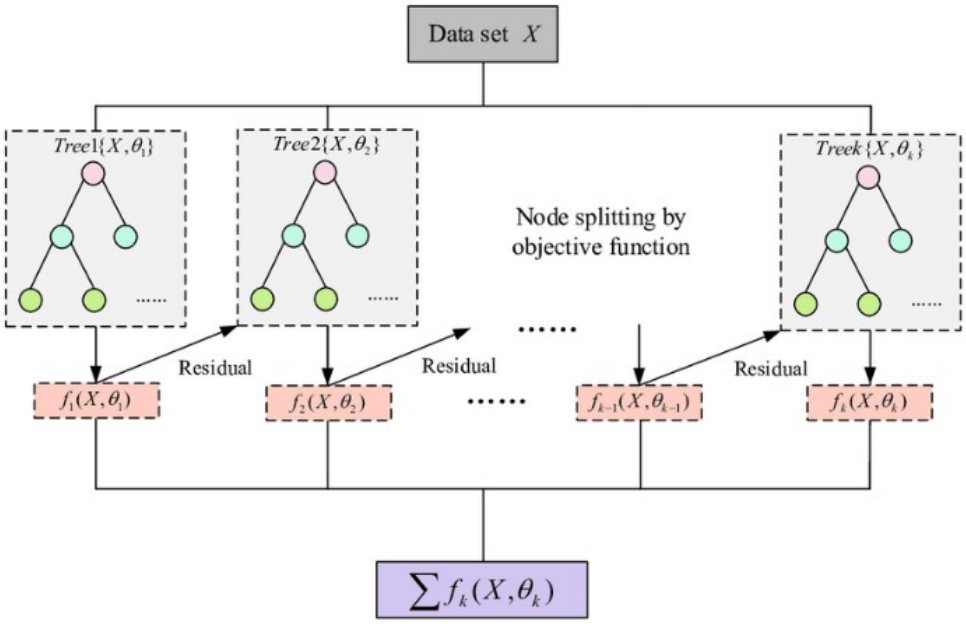


Figura 13: Construção Sequencial XGBoost [51].

O algoritmo XGBoost minimiza uma função de perda que também inclui um termo de penalidade para a complexidade do modelo, e esse termo de penalidade limita o número de árvores que são geradas [12]. Inclui ainda diferentes penalidades de regularização para evitar *overfitting*. Estas regularizações de penalidade permitem ajustar o modelo para que este possa generalizar adequadamente [50].

2.4.12 *Otimização de Hiperparâmetros*

Optuna

O pacote de *software Optuna*[9] foi desenvolvido especificamente para aplicações de ML, implementando novos critérios de *design* para otimização de hiperparâmetros. Os detalhes do registo do trilho são usados para determinar a área de pesquisa mais promissora e uma estratégia de remoção é usada para identificar e encerrar automaticamente os testes pouco promissores; com o algoritmo devidamente "focado", o número de combinações a avaliar

9 https://optuna.readthedocs.io/en/stable/installation.html

pode ser reduzido, com ganhos de eficiência. A cada execução individual (*trial*) de um modelo o Optuna avalia uma combinação específica de valores de hiperparâmetros [52].

A plataforma também é descrita como leve e muito versátil, pois os utilizadores podem construir seus próprios espaços de parâmetros e diferentes algoritmos de amostragem são fornecidos. Além disso, é uma arquitetura independente de plataforma inteiramente escrita em Python (com poucas dependências), permite paralelização e inclui ferramentas de visualização rápida [53].

2.5 REVISÃO DE LITERATURA

Nesta secção é realizada uma revisão da literatura relacionada com a previsão de consumo de energia, com a descrição das abordagens utilizadas para a previsão de consumo bem como dos métodos de previsão e do critérios de avaliação.

É necessário mencionar que há na literatura diversas publicações sobre "*load forecasting*" e "*electricity consumption forecasting*". A previsão de potência elétrica (*load forecasting*) refere-se à estimativa da potência elétrica que será requerida numa determinada área, região ou sistema elétrico em determinado instante. Esta previsão pode ser particularmente útil como indicador para representar um diagrama de carga (evolução da potência ao longo do tempo).

Já a previsão de consumo de eletricidade (*electricity consumption forecasting*) concentra-se na estimativa de consumo por parte dos consumidores individuais ou grupos de consumidores. Essa previsão é útil para empresas de energia, fornecedores de serviços públicos e operadores de rede ao planear a oferta, desenvolver estratégias de definição de preços, implementar programas de conservação de energia e gerir a procura. A previsão de consumo de eletricidade pode ser realizada em diferentes escalas de tempo, desde previsões de curto prazo até previsões de longo prazo, dependendo das necessidades e objetivos das partes interessadas [28].

A previsão de potência elétrica e a previsão de consumo de eletricidade são dois conceitos distintos, embora estejam relacionados com a procura de energia elétrica. Contudo, convém destacar que, mesmo que se possa distinguir os conceitos de previsão de consumo de eletricidade e de previsão de potência elétrica, os métodos e modelos utilizados para tal, são essencialmente os mesmos. Dessa forma, neste capítulo, são mencionadas publicações de literatura indistintamente de previsão de potência elétrica ou previsão de consumo de energia.

2.5.1 *Previsão de Consumo de Energia*

A previsão de consumo é principalmente o exercício de estimativa da procura de energia no futuro para um período específico de antecipação, em três categorias principais : **i**) Previsão de longo prazo, que é aplicada para previsão de consumo de energia até 50 anos à frente para facilitar o planeamento da expansão da produção e das redes elétricas; **ii**) Previsão de médio prazo, que é explorada para prever consumos semanais, mensais e anuais para realizar um planeamento operacional eficiente; **iii**) Previsão de consumo de curto prazo, que é usada para prever o consumo de energia até uma semana para minimizar os custos diários de execução e distribuição [28].

O uso de diferentes algoritmos para a previsão de consumo futuro de energia elétrica vem sendo utilizado concretamente em diversos sistemas de energia em diferentes tipologias de edifícios, tais como os cenários modelados sobre dados históricos nas localizações: **a**) Polo industrial na Espanha de 2014 a 2017 [54]; **b**) Medições coletadas em uma casa localizada em Sceaux, em França entre dezembro de 2006 e novembro de 2010 [55]; **c**) Edifícios na Universidade de Granada em Espanha [6]; e, **d**) Em casas residenciais na Holanda durante os anos de 2018 e 2019 [56].

2.5.2 *Métodos e Abordagens de Previsão*

Há vários métodos e abordagens, tanto estatísticas quanto de aprendizagem computacional, para prever o consumo de energia em edifícios e em determinadas regiões geográficas.

Amber et al. em estudo de 2017, utilizaram uma Regressão Múltipla para prever o uso de eletricidade em edifícios universitários localizados no *campus* Southwark da London South Bank University em Londres.

Utilizando um conjunto de dados compreendendo cinco anos de dados históricos diários de consumo de energia de janeiro de 2007 a dezembro de 2011 para a variável dependente (o consumo diário) e seis variáveis explicativas (temperatura ambiente, radiação solar, humidade relativa, velocidade do vento, índice de dias da semana e tipo de edifício), obtiveram um MAPE[10] de 8,58% e 9,76% para prédios administrativos e académicos, respetivamente [5].

O uso de técnicas estatísticas foi utilizado em 2018 por Elamin et.al. para prever a potência elétrica horária em determinada região no Japão. Foi utilizada uma modelação SARIMAX com os principais efeitos, que incluem variáveis climáticas (temperatura e

10 Erro Percentual Médio Absoluto (MAPE, do inglês *Mean Absolute Percentage Error*)

humidade) e variáveis *dummies* para as sazonalidades diária, semanal e anual [57]. O desempenho do modelo foi comparado com o de outro modelo SARIMAX que contém efeitos cruzados, ou seja, efeitos de interação de multiplicações das variáveis meteorológicas e das variáveis *dummies* sazonais, além das interações entre as diferentes classes das variáveis *dummies*. Foi utilizado um conjunto de dados horários de eletricidade coletados de janeiro de 2012 a dezembro de 2015, e obtido pela Tokyo Electric Power Company (TEPCO). Sendo uma grande empresa distribuidora no Japão, tem associado um volume de consumo muito considerável, permitindo obter-se um MAPE de 0.7% registado no modelo SARIMAX com os efeitos cruzados [57].

Fazendo uso de um modelo híbrido, Sheng et.al. construiram em 2020, um modelo de previsão de série temporal de carga baseado em SARIMAX-LSTM com o propósito de melhorar o desempenho da previsão. O conjunto de dados de carga horária, entre 22 de novembro de 2016 a 10 de janeiro de 2017, de uma cidade[11] foi usado como dados experimentais, alcançando um MAPE de 7.18% [58].

Numa investigação de 2021, apresentando uma metodologia de modelagem multivariada para a previsão de consumo de eletricidade de Espanha, Hakob Grigoryan, utilizou SARIMAX em dados históricos cobrindo o período de janeiro de 2008 a dezembro de 2019, com periodicidade mensal, para construir um modelo de previsão de consumo, combinando com o algoritmo RF, e uma remoção de ruído baseada na transformada *wavelet*. O modelo desenvolvido conseguiu produzir RMSE[12] igual a 0.125 e um MAPE igual a 2,85% [59].

Ernesto Madrid, considerou em seu Projeto de Mestrado no ano de 2021, a construção de modelos de previsão de potência elétrica de curto prazo - horizonte de previsão de uma semana, de hora em hora, totalizando 168 horas - com ML utilizando: Regressão Linear Múltipla (MLR), KNN, SVR, RF e XGBoost, em dados históricos de carga horária no Panamá, entre janeiro de 2015 e junho de 2020, ajustando hiperparâmetros via otimização com a *framework* Optuna, e obteve na modelação com o XGBoost o melhor desempenho, proporcionando um MAPE médio igual a 3,84% e com o KNN um MAPE médio igual 4.02% [60].

Também em 2021, Navid et al. organizaram um estudo para prever a procura de carga elétrica com base em ML incluindo SVM e RF usando métodos como rede neural exógena auto-regressiva não linear (NARX) e LSTM. Utilizando um conjunto de dados com reamostragem horária com nove anos (2010-2019), no Condado de Bruce, Ontário, Canadá, fundida com as informações climáticas (temperatura e velocidade do vento), obtiveram um MAPE de 10.21% no modelo com LSTM [3].

11 O nome da cidade não é mencionado no artigo.
12 Raiz Quadrada do Erro Médio Quadrático (RMSE, do inglês *Root Mean Square Error*)

Ainda em 2021, em uma investigação usando o conjunto de dados "*SmartMeter Energy Consumption Data in London Households*", Pooniwala et al. implementaram uma combinação de um modelo ARIMA sazonal (SARIMAX) com uma rede LSTM para uma melhor previsão de potência elétrica, obtendo um MAPE igual a 3.06% em um modelo hibrido [61].

Em 2022, Atabay et al. fizeram uso de ARIMAX, SARIMAX e RNNs para construir modelos para prever o consumo de eletricidade, usando dados históricos entre janeiro de 2016 a agosto de 2020, de uma casa de dois andares localizada em Houston, Texas-EUA, conseguindo um RMSE de 0.44 com a modelação via GRU [62].

Neste capítulo, foram apresentados conhecimentos iniciais sobre modelação para previsão de consumo de energia, diversos tipos de ferramentas, técnicas e métodos para desenvolvimento de modelos estatísticos e de ML. Pretendeu-se, desta forma, dotar o leitor do conhecimento necessário à compreensão do trabalho desenvolvido no âmbito deste projeto. Foi também apresentada uma breve revisão da literatura sobre previsão de consumo de energia e abordagens de previsão e trabalhos referentes a métodos estatísticos e de redes neuronais.

3

METODOLOGIA E AVALIAÇÃO DE DESEMPENHO

Este trabalho envolveu estudar modelos de previsão baseados nos algoritmos KNN, XGBoost e de Redes Neuronais - MLP, LSTM e GRU, com a plataforma TensorFlow[1] e tendo-se utilizado a biblioteca Optuna [52] para otimização de hiperparâmetros. Foram também construídos diversos modelos estatísticos SARIMA-SARIMAX fazendo uso da função *auto_arima*() [23] para modelos SARIMA e da função *sarimax*() [24] tanto para modelos SARIMA quanto modelos SARIMAX, para seleção automática dos parâmetros para os modelos estatísticos.

Todos os *scripts* de código em Python[2] dos modelos desenvolvidos neste projeto foram escritos e executados no Google Colab[3] PRO, via GPUs[4] e TPUs[5]. Fez-se uso das bibliotecas *Pandas*[6], *Seaborn*[7] [63] e *Numpy*[8] para avaliação de distribuições de frequência, técnicas de visualização de dados e medidas estatísticas.

Este capítulo tem como objetivo descrever a metodologia aplicada no desenvolvimento de modelos de predição de consumo de energia no curto prazo, bem como as métricas utilizadas para avaliar o desempenho dos modelos desenvolvidos.

1 https://colab.research.google.com/notebooks/gpu.ipynb
2 https://docs.python.org/3/library/index.html
3 https://colab.research.google.com/
4 https://www.geeksforgeeks.org/how-to-use-gpu-in-google-colab/
5 https://colab.research.google.com/notebooks/tpu.ipynb
6 https://pandas.pydata.org/docs/
7 https://seaborn.pydata.org/
8 https://numpy.org/doc/stable/reference/

3.1 METODOLOGIA

A Figura 14 mostra um diagrama do procedimento metodológico [4, 14, 16] adotado neste projeto, cujos passos são descritos abaixo.

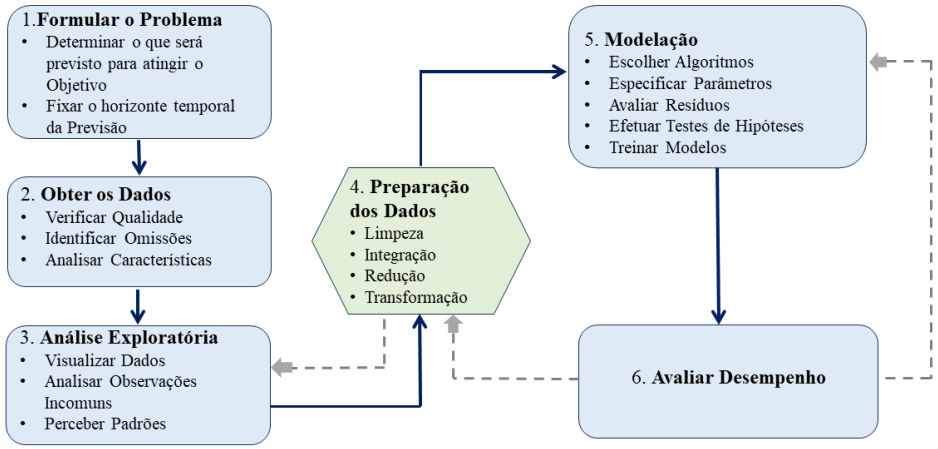


Figura 14: Procedimento Metodológico Adotado no Projeto.

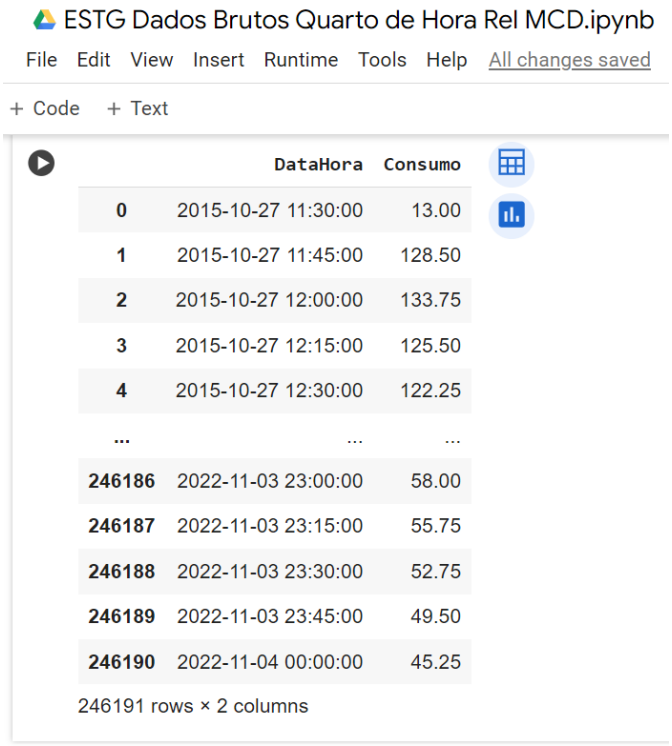
Convém observar que há um relacionamento dinâmico entre diversas etapas da metodologia, com ajustes acontecendo ao longo de todo o desenvolvimento do projeto, como evidenciado na Figura 14.

Formulação do Problema

O problema principal a resolver consistiu no desenvolvimento de modelos capazes de prever o consumo de energia do dia seguinte com base nos consumos realizados nos $\boldsymbol{n}$ dias anteriores. Neste trabalho, foram utilizados valores de $\boldsymbol{n}$ iguais a 7 e 14 (uma vez que as séries temporais associadas a consumo de energia apresentam habitualmente uma periodicidade semanal). Foram também desenvolvidos modelos que, além do consumo dos $\boldsymbol{n}$ dias anteriores, consideram outras variáveis como, a *temperatura*, ou variáveis binárias para caracterizar se o dia em causa é feriado ou um domingo.

Recolha e Compreensão dos Dados

Os dados em análise neste projeto, são dados brutos de consumo de energia elétrica do *Campus 2* do IPLeiria entre 27 de outubro de 2015 e 4 de novembro de 2022, coletados a cada 15 minutos no contador da E-Redes (empresa distribuidora de eletricidade em Portugal Continental). Estes dados foram disponibilizados, pelos professores orientadores, em ficheiros XLSX, contendo os registos de data e hora, da Potência Ativa (em kW), da Potência Reactiva Indutiva (em kVA) e da Potência Reactiva Capacitiva (em kVAr). Os dados encontravam-se distribuídos por seis ficheiros, em que cada um representa um determinado ano do período em estudo. Para cada um dos ficheiros, foi efetuado um processo de análise descritiva de dados, sendo efetuada na sequência uma consolidação conforme mostrado na Figura 15.


ESTG Dados Brutos Quarto de Hora Rel MCD.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

	DataHora	Consumo
0	2015-10-27 11:30:00	13.00
1	2015-10-27 11:45:00	128.50
2	2015-10-27 12:00:00	133.75
3	2015-10-27 12:15:00	125.50
4	2015-10-27 12:30:00	122.25
...	...	...
246186	2022-11-03 23:00:00	58.00
246187	2022-11-03 23:15:00	55.75
246188	2022-11-03 23:30:00	52.75
246189	2022-11-03 23:45:00	49.50
246190	2022-11-04 00:00:00	45.25

246191 rows × 2 columns

Figura 15: Consolidação dos Dados Brutos de Consumo em quarto de hora.

Também estão incluídos na análise os dados históricos brutos referente à meteorologia. O conjunto de dados, no período entre 01 de outubro de 2015 a 31 de outubro de 2022, com granularidade diária, foi obtida junto do Instituto Português do Mar e da Atmosfera (IPMA) [64], em ficheiro *Microsoft Excel Worksheet (XLSX)* com as colunas, conforme Figura 16, a seguir:

ESTG Dados Bruto Clima Rel MCD.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

```
[ ]  1 dfclima_bruto
```

	ESTACAO	ANO	MES	DIA	T_MED	T_MAX	T_MIN	HR_MED	HR_MAX	HR_MIN	DD_MED	DD_FFX	FF_MED	FF_MAX
0	1210718	2015	10	1	17.9	27.7	9.9	73.0	96.0	38.0	72.0	332.0	1.3	8.3
1	1210718	2015	10	2	17.2	23.5	13.1	78.0	96.0	46.0	15.0	315.0	1.3	8.6
2	1210718	2015	10	3	18.2	23.9	11.4	72.0	95.0	35.0	92.0	152.0	1.0	6.7
3	1210718	2015	10	4	23.6	27.2	20.5	73.0	89.0	57.0	189.0	146.0	6.5	14.5
4	1210718	2015	10	5	21.4	24.0	19.3	82.0	94.0	67.0	215.0	231.0	5.6	18.5
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2583	1210718	2022	10	27	21.4	24.4	19.0	77.0	88.0	65.0	164.0	189.0	4.6	13.1
2584	1210718	2022	10	28	20.7	23.5	18.8	75.0	93.0	55.0	146.0	173.0	3.6	11.9
2585	1210718	2022	10	29	17.8	20.6	14.9	85.0	97.0	61.0	151.0	246.0	2.0	10.4
2586	1210718	2022	10	30	16.4	23.4	11.5	84.0	99.0	50.0	106.0	104.0	1.1	4.7
2587	1210718	2022	10	31	17.0	20.4	11.4	89.0	98.0	75.0	215.0	254.0	2.0	7.2

2588 rows × 14 columns

Figura 16: Dados Brutos de Clima em base diária.

O significado de cada coluna é o seguinte:

- T_MED: Temperatura média do ar a 1,5m (ºC)
- T_MAX: Temperatura máxima do ar a 1,5m (ºC)
- T_MIN: Temperatura mínima do ar a 1,5m (ºC)
- HR_MED: Humidade relativa média (%)
- HR_MAX: Humidade relativa máxima (%)
- HR_MIN: Humidade relativa mínima (%)
- DD_MED: Rumo médio do vento (º)
- DD_FFX: Rumo do vento máximo (º)
- FF_MED: Intensidade média do vento (m/s)
- FF_MAX: Intensidade máxima instantânea do vento (m/s)

Os dados meteorológicos foram recolhidos na estação meteorológica 1210718 que está localizada no Aeródromo de Leiria.

Análise Exploratória

Nesta fase foi feita uma análise das características dos dados, tentando identificar anomalias ou problemas de incompletude, inconsistência e ruído. A informação obtida nesta etapa pode ajudar na seleção das técnicas mais adequadas para efetuar o pré-processamento de dados.

Preparação dos Dados

Utilizando as bibliotecas *Pandas* e *NumPy*, as tarefas efetuadas nesta etapa foram: **a**) Limpeza: para remoção de ruído[9], correção de inconsistências, imputação de valores ausentes; **b**) Integração: foi necessário unir dados de vários ficheiros em XLSX; **c**) *Resampling*: efetuou-se a mudança de frequência dos dados de consumo de energia de quarto de hora para uma reamostragem diária; **d**) Junção: foi feita a concatenação dos *datasets* de consumo e clima.

Modelação e Treino

Neste projeto foram utilizados os modelos estatísticos SARIMA e SARIMAX, modelos de ML KNN e XGBoost e também as redes neuronais MLP, LSTM e GRU. Na etapa de treino dos modelos de redes neuronais, os dados foram normalizados para o intervalo [0, 1]. Utilizou-se o Optuna para obter o número ideal de neurónios para cada camada intermédia oculta *unit1* e *unit2*, *dropout*[10] e *batch-size*[11], para o treino das redes neuronais, as funções *auto_arima*() e *sarimax*() para os modelos estatísticos e as técnicas de *expanding window*[12] [4] e *rolling window*[13] com os métodos estatísticos para se poderem comparar os resultados obtidos com estes modelos com os resultados obtidos com os modelos de ML.

9 O ruído pode incluir erros de medição e variabilidade natural imprevisível [65].

10 Técnica de regularização para redes neuronais artificiais que consiste em, durante o treino, desativar aleatoriamente uma determinada fração de neurónios [66].

11 O *batch size* é um hiperparâmetro de treino de redes neuronais que especifica o número de amostras dos dados que serão usadas para tentar melhorar a precisão de um modelo e reduzir seu tempo de treinamento [22].

12 Técnica em que o conjunto de treino é expandido a cada iteração do algoritmo, enquanto o conjunto de teste permanece um passo à frente.

13 Técnica na qual o modelo é reajustado regularmente com uma janela deslizante de dados históricos à medida que novos dados se tornam disponíveis.

3.2 AVALIAÇÃO DE DESEMPENHO

Esta etapa incluiu a realização de uma análise da magnitude do Erro Percentual Médio Absoluto (MAPE), da Raiz do Erro Médio Quadrático (RMSE), obtidos com os diferentes modelos utilizados, do Critério de Informação de Akaike (AIC) e do Erro Médio Quadrático (MSE, do inglês *Mean Square Error*) no ajuste dos modelos.

3.2.1 *MAPE*

A métrica de avaliação MAPE oferece uma maneira sistemática de quantificar o quão próximo os valores previstos estão dos valores reais. Ao fornecer uma medida numérica da precisão das previsões, o MAPE permite uma comparação objetiva entre diferentes modelos e abordagens. Além de proporcionar uma avaliação quantitativa, o MAPE também pode ajudar a identificar áreas de melhoria nos modelos [4, 22].

Esta métrica fornece uma medida de avaliação do desempenho do modelo em termos percentuais, o que é importante em muitas aplicações práticas. Contudo, medidas baseadas em erros percentuais têm a desvantagem de o valor calculado ser indefinido ou infinito se algum A_t for igual a zero ou próximo de zero [4]. A Equação (2) mostra que é calculado como a média das percentagens de erros absolutos entre os valores observados e previstos:

$$\text{MAPE} = 100 \times \frac{1}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \tag{2}$$

onde:

- A_t representa o valor real,
- F_t representa o valor previsto pelo modelo, e
- n é o número total de observações.

3.2.2 *MSE*

O Mean Squared Error (MSE) é uma das métricas mais utilizadas em aprendizado de máquina, calculado como a média dos erros quadrados e usada na área de análise de dados para avaliar o ajuste de modelos, e a Equação (3) mostra seus termos:

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(A_t - F_t)^2 \tag{3}$$

onde:

- A_t representa o valor real,
- F_t representa o valor previsto pelo modelo, e
- n é o número total de observações.

O MSE leva em consideração o conjunto de dados que está sendo previsto, uma vez que depende fundamentalmente da variância dos dados previstos. Outra vantagem dessa métrica é excluir valores negativos, uma vez que os erros são elevadas ao quadrado e depois obtida a média [8].

3.2.3 *RMSE*

RMSE é a raiz quadrada do erro médio quadrático e há uma vantagem em analisar o RMSE em vez do MSE. A razão para obter a raiz quadrada do MSE é que a escala do RMSE é a mesma escala da variável original [22].

A Equação (4) do RMSE é dada por:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(A_t - F_t)^2} \tag{4}$$

onde:

- A_t representa o valor real,
- F_t representa o valor previsto pelo modelo, e
- n é o número total de observações.

Quanto menor o valor do RMSE, melhor é o desempenho do modelo, já que indica uma menor diferença entre os valores previstos e os valores reais, sendo muito usada para comparar diferentes modelos de regressão.

3.2.4 *AIC*

O Critério de Informação de Akaike (AIC) é uma métrica de avaliação de ajuste muito utilizada em modelação da família de modelos ARIMA.

A Equação (5) mostra a fórmula:

$$AIC = 2k - 2\ln(L) \tag{5}$$

onde:

- k é o número de parâmetros no modelo,
- L é a verossimilhança do modelo.

Diversos *softwares* estatísticos usam definições ligeiramente diferentes para o AIC, embora todas devam conduzir à seleção do modelo mais preciso e fiável.

O AIC estima a qualidade de um modelo em relação a outros modelos. Dado que haverá alguma perda de informação quando um modelo for ajustado aos dados, o AIC estima a quantidade relativa de informações perdidas pelo modelo. Quanto menos informação for perdida, menor o valor AIC e melhor o modelo [14].

Neste capítulo, foram descritas a metodologia e as métricas para avaliação de desempenho dos modelos construídos neste projeto. A informação traz um conhecimento adequado para o desenvolvimento dos capítulos seguintes neste trabalho.

4

ANÁLISE EXPLORATÓRIA E PREPARAÇÃO DOS DADOS

Este capítulo trata das etapas desenvolvidas para análise e tratamento dos dados, quando cada um dos períodos anuais anteriormente mencionados teve os seus dados analisados em separado quanto à seleção, limpeza, qualidade, integridade, e sua distribuição, considerando o planeamento preliminar projetado para alcançar os objetivos propostos.

4.1 ANÁLISE EXPLORATÓRIA E LIMPEZA DOS DADOS

Numa primeira fase, foi realizada uma análise das características dos dados, tentando identificar anomalias ou problemas de incompletude, inconsistência e ruído, existentes no conjunto de dados original. A informação obtida nesta etapa pode ajudar na seleção das técnicas mais adequadas para efetuar o pré processamento de dados, no que se refere à identificação de duplicidade, dados omissos ou com valores iguais a zero.

4.1.1 *Dados de Consumo*

Em todos os *datasets* referentes aos registos quarto-horários da potência ativa ao longo dos períodos anuais provenientes dos ficheiros XLSX, e uma vez que o propósito do projeto é a previsão do consumo de energia no curto prazo (e não de potência), foi efetuada a conversão da coluna "Activa kW", com o uso do fator 0.25, para consumo e, concomitantemente, alterado o nome da coluna para "Consumo", a fim de representar a conversão mencionada. A coluna "Data hora" foi renomeada para "DataHora" e eliminadas as demais colunas. Para cada período anual foi de imediato criado um novo ficheiro .csv com as colunas "DataHora" e "Consumo".

Relativamente a **dados em falta**, percebeu-se a inexistência de registos na madrugada dos dias em que ocorria o início (março) do horário de verão - *daylight saving time* (DST) - quando os relógios são adiantados em 60 minutos, provocando dessa forma a perda de quatro registos. O procedimento adotado para suprir a ausência de dados devido a este problema foi o seguinte:

i) Identificar a ausência de registos do consumo da própria madrugada do dia de inicio do DST;

ii) Obter os valores de registo do consumo da madrugada referente à madrugada de sete dias antes, na mesma faixa horária;

iii) Obter os valores de registo do consumo da madrugada referente à madrugada de sete dias após, na mesma faixa horária;

iv) Efetuar uma consolidação dos dados estatísticos de sete dias antes e sete dias depois, relativo aos itens ii e iii, acima;

v) Utilizar a média simples obtida na consolidação estatística, no item iv, para substituir os quatro registos ausentes na faixa horária entre 01h00 e 01h45.

A imputação de dados ausentes pode ser um processo imperfeito. No entanto, precisa ser efetuado a fim de evitar redução da precisão do modelo posto que os modelos de previsão aprendem com os padrões presentes nos dados históricos e quando há valores ausentes, esses padrões ficam distorcidos, o que pode levar a previsões menos precisas, criando dificuldades na avaliação do modelo.

Ao adotar o procedimento acima descrito, espera-se que a lacuna de dados na madrugada do início do DST seja preenchida de forma satisfatória, minimizando o impacto da perda de informações na análise do consumo de energia. A média simples dos valores de consumo das madrugadas anteriores e posteriores fornece uma estimativa razoável do consumo real na madrugada com ausências de registo, considerando a potencial sazonalidade e os padrões de consumo típicos do período noturno.

Também foi identificada a ausência de dados, não causado pelo efeito DST, na faixa horária entre 01h15 a 05h45 do dia 20 de maio no ficheiro do ano de 2016 e como solução, optou-se por verificar os registos de valores na mesma faixa horária dos dados ausentes da semana anterior (7 dias antes) e de 7 dias após (semana posterior) e usar a média simples desses registos em substituição dos dados ausentes.

No ano de 2018 há ausência de 4 registos (06h45, 07h00, 07h15 e 07h30), também não causado pelo efeito DST, e o valor das 07h45 igual a zero, ou seja, perfazendo 5 posições de registos a serem corrigidas. Após verificações e comparações, decidiu-se usar a mediana entre a faixa horária de 05h15 as 09h00 do próprio dia 13 de setembro, como valor substituto aos registos problemáticos. O racional dessa decisão baseia-se no facto de que o valor da mediana comparado com o valor da média ficou mais adequado para ser utilizado garantindo que os dados substituídos fossem representativos do comportamento normal naquele período. A seleção de dados do mesmo dia e janela horária alargada também ajudou a manter qualquer possível sazonalidade e também os padrões intradiários da série.

Relativamente a **dados duplicados** foi verificado, em todo o período anual a partir de 2016, duplicidade de registos nos dias que se refere ao fim do DST, provocados pelo atrasar dos relógios em 60 minutos, gerando assim quatro registos a mais, totalizando oito registos. Decidiu-se obter a média dessas oito observações e utilizá-las como substitutas, em quatro posições entre 01h00 e 01h45, uma vez que os registos do consumo de energia têm resolução quarto-horária.

4.1.2 *Dados Meteorológicos*

Os arquivos com os dados meteorológicos brutos recebidos do IPMA contêm dados médios registados com resolução diária, e com as informações referentes a "Ano", "Mês" e "Dia", em colunas separadas. Foi efetuada a concatenação dessas colunas para uma única coluna "DIA", seguida da remoção das colunas "Ano" e "Mês". A análise das informações estatísticas dos dados meteorológicos permitiu verificar a existência de registos com valores negativos iguais a -990, como se identifica na Tabela 1.

Tabela 1: Resumo estatístico dos dados brutos de Clima.

Variável	Count	Mean	Std	Min	25%	50%	75%	Max
T_MED	2588.0	-27.91	203.75	-990.0	11.4	15.1	18.7	28.7
T_MAX	2588.0	-26.40	214.44	-990.0	16.5	20.5	24.6	44.1
T_MIN	2588.0	-29.05	193.76	-990.0	5.9	10.3	14.0	22.9
HR_MED	2588.0	65.58	116.59	-990.0	73.0	79.0	84.0	99.0
HR_MAX	2588.0	78.01	135.59	-990.0	94.0	96.0	98.0	100.0
HR_MIN	2588.0	35.66	130.97	-990.0	43.0	52.0	61.0	98.0
DD_MED	2588.0	188.65	225.13	-990.0	121.0	239.0	322.0	360.0
DD_FFX	2588.0	221.64	226.25	-990.0	174.0	303.0	326.0	359.0
FF_MED	2588.0	-22.60	155.31	-990.0	1.6	2.1	2.8	7.4
FF_MAX	2588.0	-18.36	163.24	-990.0	7.1	8.5	10.5	28.4

Na Tabela 1, observa-se que há diversas variáveis com valores mínimos negativos iguais a -990 e foi necessário buscar uma solução, sem remover as datas da série temporal, uma vez que as datas dos dados meteorológicos devem ser concatenadas com as datas do

dataset de consumo. Felizmente, todos esses valores não estão em sequência ordenada e sim espalhados pelo conjunto de dados de clima, nos seguintes intervalos de datas:

- entre 28 de agosto e 22 setembro de 2016
- entre 19 de abril e 3 maio de 2017
- em 14 de maio de 2018 e em 22 de maio 2018
- em 07 de julho de 2018 e em 23 de agosto de 2018
- entre 14 de novembro e 10 de dezembro de 2018
- em 04 de junho de 2019 e em 06 de julho de 2019
- em 23 de julho de 2019 e em 31 de julho de 2019
- em 03 de outubro de 2019 e em 11 de novembro de 2019
- em 17 e 18 de novembro de 2019
- entre 29 de janeiro e 04 de fevereiro de 2020
- em 16 de agosto de 2020
- entre 20 de março e 31 de março de 2021
- em 12 de maio de 2021.

Como alternativa de solução, optou-se pela interpolação linear, e uma nova análise das informações estatísticas dos dados meteorológicos permitiu verificar que a solução adotada foi adequada, como se identifica na Tabela 2.

Tabela 2: Novo resumo estatístico dos dados brutos de Clima.

Variável	Count	Mean	Std	Min	25%	50%	75%	Max
T_MED	2588.0	15.20	4.53	0.6	11.9	15.50	18.77	28.7
T_MAX	2588.0	21.28	5.37	8.10	17.0	20.90	24.90	44.1
T_MIN	2588.0	9.96	5.07	-5.7	6.4	10.60	14.00	22.9
HR_MED	2588.0	78.41	8.91	27.0	74.0	79.0	85.00	99.0
HR_MAX	2588.0	95.23	4.10	37.0	94.0	96.0	98.00	100.0
HR_MIN	2588.0	52.22	14.65	3.0	44.0	53.0	61.00	98.0
DD_MED	2588.0	221.89	107.89	0.0	125.0	250.18	322.00	360.0
DD_FFX	2588.0	255.12	94.40	0.0	183.0	303.00	326.00	359.0
FF_MED	2588.0	2.32	0.91	0.5	1.7	2.10	2.80	7.4
FF_MAX	2588.0	9.04	2.73	3.0	7.2	8.60	10.50	28.4

Convém destacar na Tabela 2, o valor de −5.7. Uma pesquisa nos dados brutos de Clima mostra esse valor como registo da temperatura miníma no dia 19 de janeiro de 2017. Conforme se pode constatar no site do IPMA[1], relativamente ao mês de janeiro de 2017, ocorreu um fluxo de massa fria oriundo do norte, determinado por um anticiclone que se estendia desde a Islândia às Canárias e um vale depressionário estendendo-se desde a Escandinávia ao Mediterrâneo, no início da semana, dias 15 e 16, transportou ar extremamente frio da região polar para a Europa Central e do Sul. Nos dias seguintes, com a mudança do fluxo para nordeste a massa de ar frio e seco veio a atingir, a partir do dia 18, o território do Continente, originando valores muito baixos da temperatura do ar.

Decidiu-se ainda, selecionar a variável T_MED, correspondendo à temperatura média do ar a 1,5m (°C), para ser utilizada nos modelos deste trabalho, alterando seu título para "Temp". As restantes colunas não foram utilizadas.

1 https://www.ipma.pt/pt/oclima/monitorizacao/index.jsp?selTipo=m&selVar=tn&selAna=an&selAno=2017

4.2 REDUÇÃO E INTEGRAÇÃO DE DADOS

Redução

Os *datasets* com os dados brutos de Consumo, coletados a cada quinze minutos, entre as 11h30 do dia 27 de outubro de 2015 às 00h00 do dia 04 de novembro de 2022, foram então consolidados, como mostra a Tabela 3, e esse novo *dataset* foi utilizado para início da análise.

Tabela 3: Dados consolidados em quarto de hora.

Índice	**DataHora**	**Consumo** (kWh)
0	2015-10-27 11:30:00	13.00
1	2015-10-27 11:45:00	128.50
2	2015-10-27 12:00:00	133.75
3	2015-10-27 12:15:00	125.50
4	2015-10-27 12:30:00	122.25
...	...	...
246186	2022-11-03 23:00:00	58.00
246187	2022-11-03 23:15:00	55.75
246188	2022-11-03 23:30:00	52.75
246189	2022-11-03 23:45:00	49.50
246190	2022-11-04 00:00:00	45.25

Os dados originais do consumo estão registados em intervalos de 15 minutos, e o objetivo é criar um modelo capaz de prever a procura de energia elétrica com resolução diária. Executou-se o agrupamento, convertendo os dados de quarto de hora para diário. Esse tipo de transformação foi executado com a biblioteca *Pandas*, tendo-se tido especial cuidado no uso correto dos argumentos das funções utilizadas para evitar a introdução de informações futuras no conjunto de treino (*leakage*), o que poderia gerar previsões irrealistas. Nesta etapa foi acrescentada a coluna "DiaSemana" com o nome do dia da semana a que se refere a coluna "DataHora".

Para o *dataset* final, que foi utilizado para a modelação, por questões de *daily resampling*, a data inicial ficou sendo 28 de outubro (quarta-feira) de 2015 e, para melhor ajuste de

período final, decidiu-se o término ser no dia 31 de outubro de 2022, já que esta corresponde à data de término dos dados meteorológicos obtidos junto do IPMA. A Tabela 4 condensa as informações sobre os dados após o agrupamento para frequência diária, contendo as colunas "DataHora", "Consumo" e "DiaSemana".

Tabela 4: Agrupamento dos dados de consumo para frequência diária.

Índice	**DataHora**	**Consumo** (kWh)	**DiaSemana**
0	2015-10-28	7692.25	Quarta-feira
1	2015-10-29	8556.00	Quinta-feira
2	2015-10-30	8012.25	Sexta-feira
3	2015-10-31	5273.50	Sábado
4	2015-11-01	4565.50	Domingo
...	...	...	
2556	2022-10-27	7156.75	Quinta-feira
2557	2022-10-28	6789.25	Sexta-feira
2558	2022-10-29	3691.25	Sábado
2559	2022-10-30	3019.50	Domingo
2560	2022-10-31	6744.50	Segunda-feira

Integração

Os dois *datasets*, com dados de consumo e temperatura, foram concatenados em somente um *dataset*, conforme Tabela 5, para ser utilizado na modelação.

Tabela 5: *Dataset* a ser utilizado para modelos de previsão.

Índice	**DataHora**	**Consumo** (kWh)	**DiaSemana**	**Temp** (°C)
0	2015-10-28	7692.25	Quarta-feira	15.8
1	2015-10-29	8556.00	Quinta-feira	18.1
2	2015-10-30	8012.25	Sexta-feira	17.8
3	2015-10-31	5273.50	Sábado	15.6
4	2015-11-01	4565.50	Domingo	15.4
...	...	...	...	...
2556	2022-10-27	7156.75	Quinta-feira	21.4
2557	2022-10-28	6789.25	Sexta-feira	20.7
2558	2022-10-29	3691.25	Sábado	17.8
2559	2022-10-30	3019.50	Domingo	16.4
2560	2022-10-31	6744.50	Segunda-feira	17.0

Feriados e Domingos

Decidiu-se também pesquisar o efeito de feriados e domingos no consumo de energia do *Campus 2*. A opção pelo domingo deu-se porque os edifícios do *Campus 2* estão encerrados, não havendo qualquer atividade. Para tal, foi adicionada uma coluna do tipo binária ao *dataset* que indicava se uma observação corresponde a um domingo. Foi ainda adicionada uma outra coluna binária com os feriados em Portugal. Essas variáveis foram utilizadas no desenvolvimento de modelos como variável exógena com a utilização da função SARIMAX e em modelos de ML, conforme Tabela 6.

Tabela 6: *Dataset* com inclusão de Feriados e Domingos.

Índice	**DataHora**	**Consumo** (kWh)	**DiaSemana**	**Temp** (°C)	**Feriado**	**Domingo**
0	2015-10-28	7692.25	Quarta-feira	15.8	0	0
1	2015-10-29	8556.00	Quinta-feira	18.1	0	0
2	2015-10-30	8012.25	Sexta-feira	17.8	0	0
3	2015-10-31	5273.50	Sábado	15.6	0	0
4	2015-11-01	4565.50	Domingo	15.4	0	1
...	...	...	...	...	...	...
2556	2022-10-27	7156.75	Quinta-feira	21.4	0	0
2557	2022-10-28	6789.25	Sexta-feira	20.7	0	0
2558	2022-10-29	3691.25	Sábado	17.8	0	0
2559	2022-10-30	3019.50	Domingo	16.4	0	1
2560	2022-10-31	6744.50	Segunda-feira	17.0	0	0

4.3 ANÁLISE DA SÉRIE TEMPORAL DO CONSUMO DE ENERGIA

Nesta secção são apresentados os resultados dos testes de *Shapiro-Wilk* e *Jarque-Bera* relativos à normalidade e os testes de *Dickey-Fuller* Aumentado (ADF) e o teste *Kwiatkowski-Phillips-Schmidt-Shin* (KPSS) de estacionariedade da série temporal do Consumo. Também nesta secção se faz uma análise visual da série temporal do Consumo, já que os gráficos permitem a visualização de muitas características dos dados, incluindo padrões sazonais, relações entre variáveis, mudanças ao longo do tempo e observações incomuns.

Pode-se notar na Figura 17, a existência de uma notória redução do consumo em parte do ano de 2020 e início do ano de 2021 devido ao efeito da pandemia do Covid19.

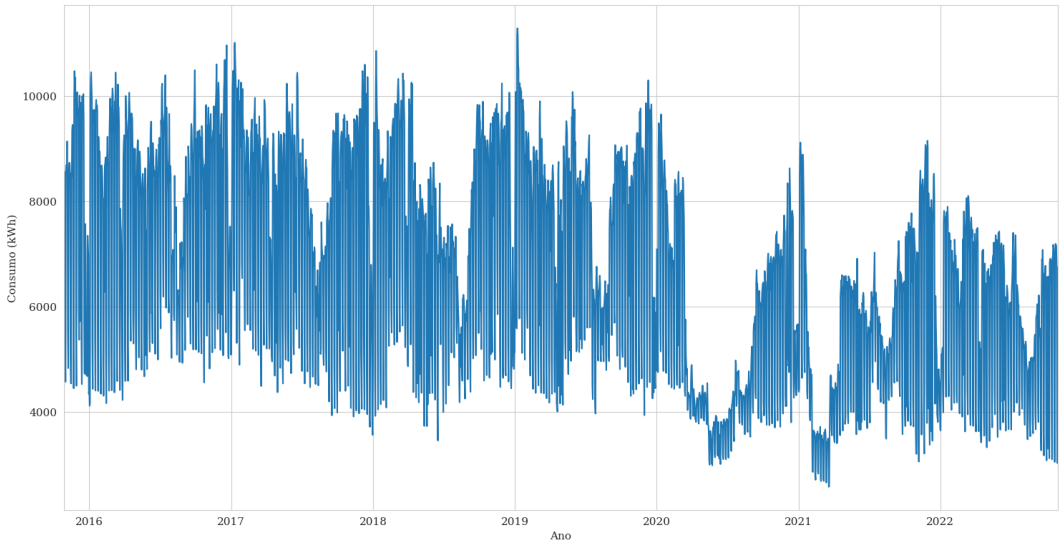


Figura 17: Evolução da série temporal do consumo.

As características que são vistas nos gráficos dos dados devem ser incorporadas, tanto quanto possível, nos métodos de previsão a serem utilizados. Assim como o tipo de dados determina qual o método de previsão a usar, também determina que gráficos são apropriados [4].

É facilmente percecionado que o consumo de energia em *campi* universitários no fim de semana é menor que durante a semana, como mostra a Figura 18.

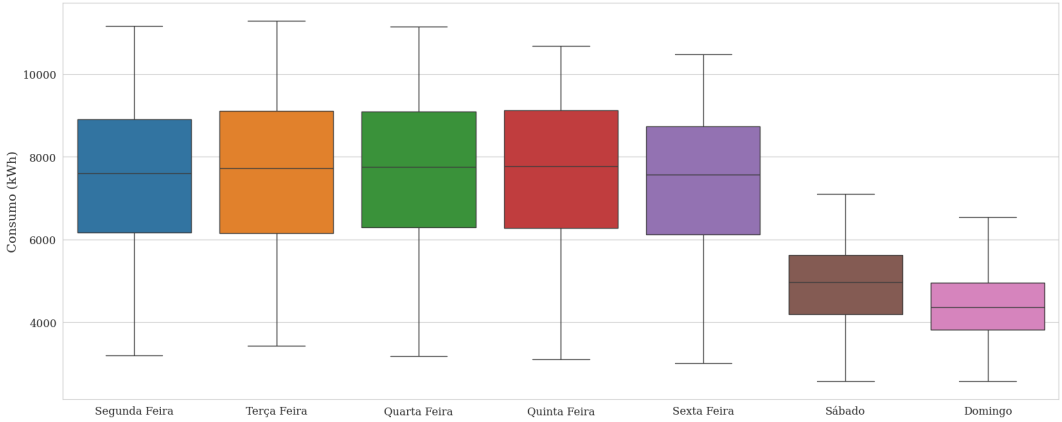


Figura 18: Evolução do consumo por dia da semana.

A Figura 18, revela ainda que a função *resample*(), utilizada para converter os dados para uma reamostragem horária e também diária, ocorreu de forma adequada, uma vez que não se visualizam alterações significativas referentes ao consumo dos sábados e domingos, confirmando um adequado reposicionamento dos dados, para garantir um agrupamento coerente por blocos quarto-horários. Também é possível constatar este facto, através da

visualização da evolução horária do consumo em dia próximo e após o final de semana, como revela a Figura 19.

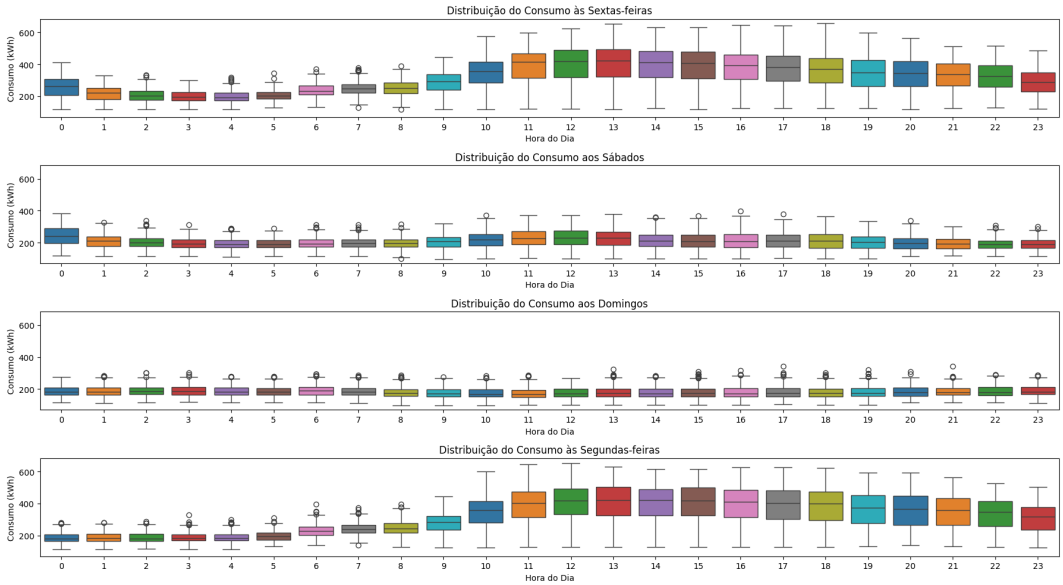


Figura 19: Distribuição do consumo de energia às Sextas-feiras, Sábados, Domingos e Segundas-feiras.

Além de permitir verificar a amplitude e variabilidade dos valores para cada hora desses dias, a Figura 19 revela que os dados se comportam como esperado, confirmando que os procedimentos adotados durante as etapas de preparação, pré-processamento e reagrupamento estão adequados.

4.3.1 *Outliers*

O *boxplot* é uma ferramenta gráfica que permite visualizar a distribuição e *outliers* (se houver). Pode também ser um meio complementar de entendimento sobre as características dos dados. Nas figuras 20 e 21 há uma série de *boxplots* que auxiliam no entendimento a respeito da série temporal do consumo de energia.

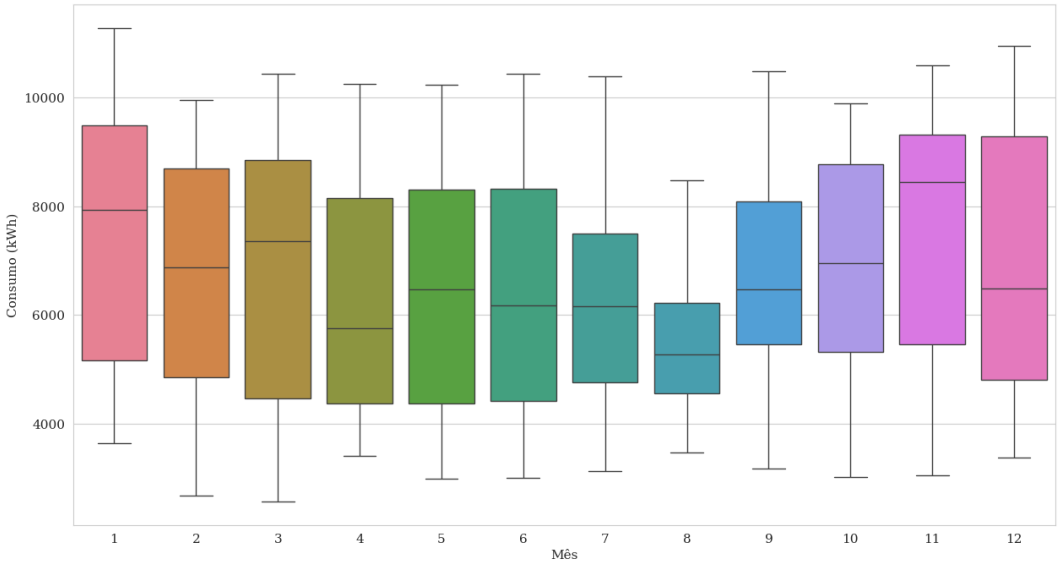


Figura 20: Variabilidade do consumo mensal de energia.

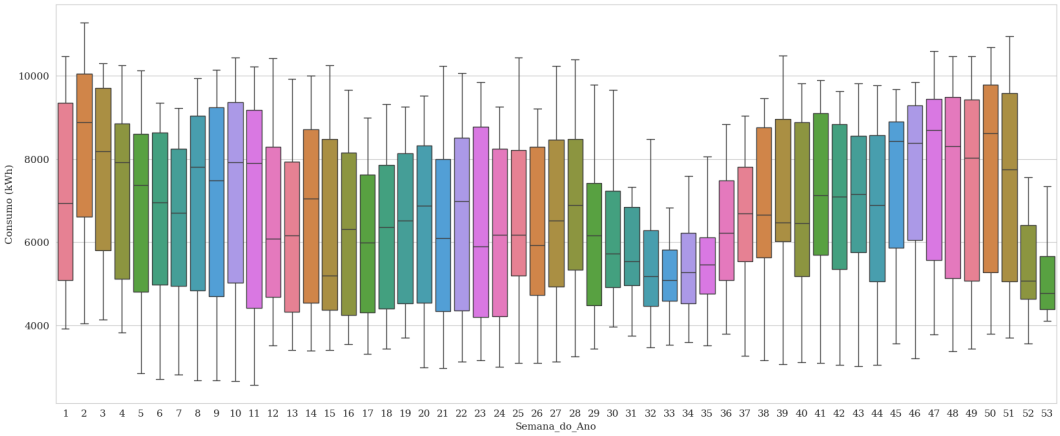


Figura 21: Variabilidade do consumo semanal de energia.

Os *boxplots* com frequências de consumo mensal e semanal[2], Figuras 20 e 21 parecem indicar a não existência de *outliers* nos dados.

Apesar disso, decidiu-se utilizar também o algoritmo *Isolation Forest* (IF)[3] como auxílio na identificação de possíveis *outliers* . O IF é um algoritmo de deteção de anomalias que funciona criando árvores de decisão de forma aleatória e isolando os pontos de dados que requerem menos partições para serem separados. Pontos de dados que são isolados mais rapidamente são mais propensos a serem considerados *outliers* [68].

2 De tempos em tempos, há um ano que tem 53 semanas em vez de 52 - não é um erro, mas sim uma parte do sistema de calendário gregoriano [12, 67] .

3 https://scikitlearn.org/stable/auto_examples/ensemble/plot_isolation_forest.html

Arbitrou-se que 5% das observações poderiam ser *outliers* e o resultado da verificação de possíveis *outliers* pelo IF pode ser visualizado na Figura 22.

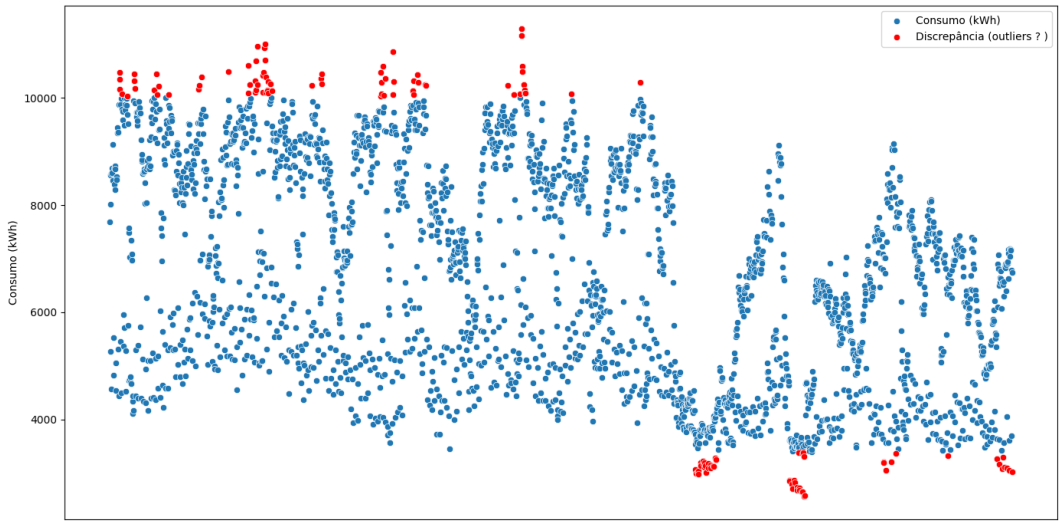


Figura 22: Verificação de valores discrepantes na série temporal do consumo.

Atribuiu-se que valores, identificados como discrepantes na Figura 22, menores que $Q_1 - 1.5 \cdot IQR$ e maiores que $Q_3 + 1.5 \cdot IQR$ seriam classificados como *outliers*,[4] já que está é uma forma simples e eficaz de identificar anomalias, e nenhuma observação da série temporal do consumo foi considerada como *outlier* segundo esse critério.

4.3.2 *Sazonalidade*

Ao analisar a sazonalidade de uma série temporal, pode observar-se a Função de Autocorrelação (ACF) para verificar se há padrões significativos. Esta etapa é importante para analisar o padrão de comportamento dos dados. Se houver autocorrelações significativas em múltiplo período sazonal, isso indica que a série temporal é sazonal.

Correlograma da Série Temporal do Consumo de Energia

A visualização dos gráficos ACF permite identificar uma estrutura de autocorrelação de forma rápida e intuitiva.

4 Três quartis: Q_1, Q_2 e Q_3, dividem um conjunto de dados ordenados em quatro partes iguais . O IQR é a amplitude do intervalo interquartil (IQR = $Q_3 - Q_1$), e é uma medida robusta de dispersão, o que significa que ela é menos afetada por *outliers* [69].

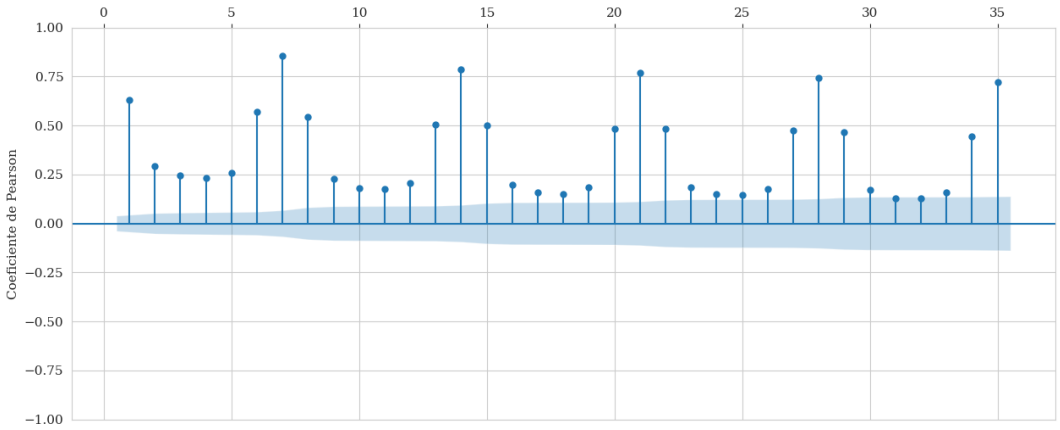


Figura 23: Função Autocorrelação da série temporal do consumo.

No gráfico de autocorrelação na Figura 23, já se percebe visualmente que o ACF da série temporal do consumo mostra que há autocorrelações significativas em múltiplos de 7, o que indica que a série temporal possui sazonalidade a cada 7 *lags*.[5]

Decomposição da Série Temporal do Consumo de Energia

Com a decomposição pode-se visualizar cada componente da série temporal em separado, o que pode ajudar a identificar a tendência e o padrão sazonal nos dados, sendo esta uma tarefa importante de auxílio na análise e no entendimento das séries temporais, permitindo uma melhor previsão.

A decomposição clássica composta é um método de análise de séries temporais que divide a série em quatro componentes principais como na Figura 24, revelando a direção geral da série ao longo do tempo, as oscilações de longo prazo e os padrões repetitivos, além do resíduo que são as flutuações aleatórias que não podem ser explicadas pelos outros componentes.

5 No contexto deste Capítulo 5 e do Capítulo 6, *lag* representa o intervalo de tempo, ou o número de períodos, ou o atraso, ou o desfasamento entre dois valores (observação) de uma série temporal (https://www.spestatistica.pt/en/glossary).

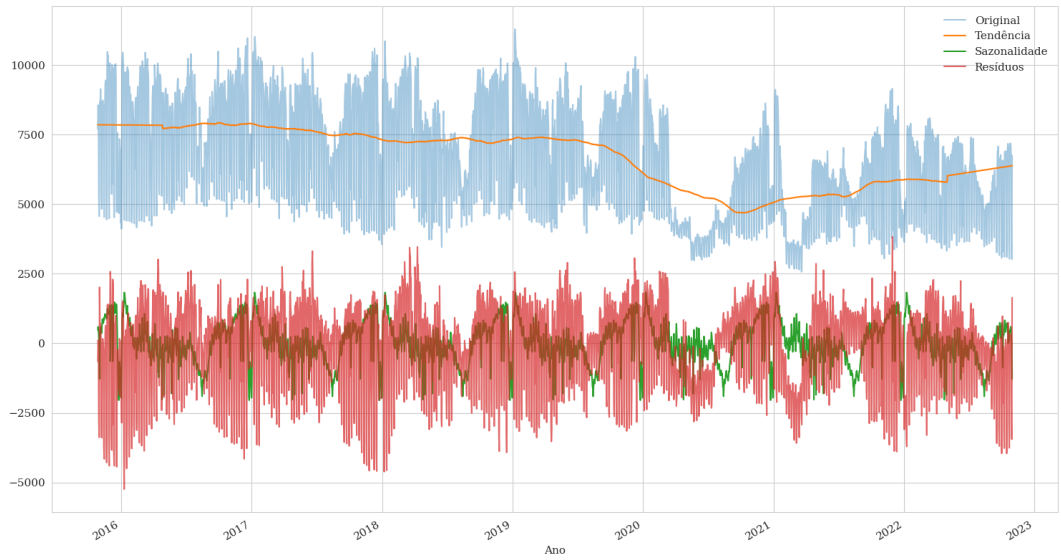


Figura 24: Decomposição Clássica Composta da série temporal do consumo.

A visualização e a análise dos *boxplots* nas figuras, 18, 20, 21, e do correlograma, na Figura 23, permitiram identificar que a série temporal do consumo possui diferentes sazonalidades.

Embora a decomposição clássica composta seja uma ferramenta útil para analisar séries temporais, não proporciona uma visualização adequada dessas diversas sazonalidades. Dessa forma, optou-se por efetuar a decomposição sazonal múltipla (MSTL, do inglês *Multiplicative Seasonal-Trend-Residual Decomposition using Loess*)[6], que permite a decomposição de séries temporais com múltiplos padrões sazonais [71].

Para um melhor entendimento das sazonalidades, decidiu-se ser mais adequado efetuar a decomposição MSTL em um intervalo menor para visualizar a sazonalidade semanal e mensal do ano de 2017, como observado na Figura 25.

6 A implementação do MSTL está disponível em [70], sendo um processo totalmente automatizado, com algoritmo de decomposição de séries temporais fazendo uso do método aditivo para lidar com séries temporais com vários padrões sazonais [71].

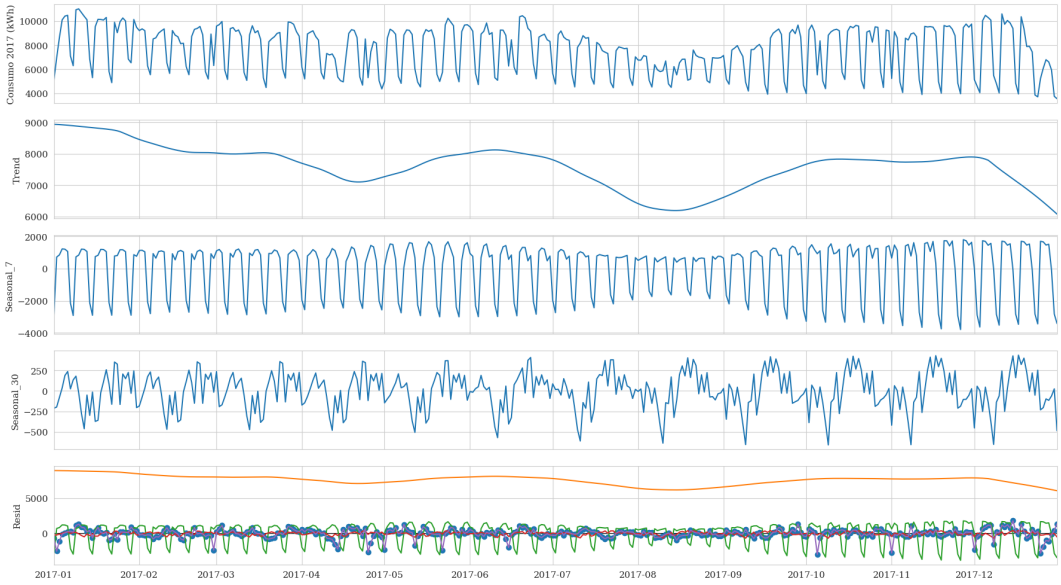


Figura 25: Decomposição multi sazonal da série temporal do consumo (2017).

A componente de tendência mostra a direção geral do comportamento da série temporal ao longo do tempo. No caso da série temporal do consumo percebe-se um comportamento diverso ao longo do tempo do intervalo em análise neste trabalho.

Já a componente sazonal, que representa os padrões que se repetem em intervalos fixos de tempo, ajuda a identificar de facto as flutuações semanais e mensais na série temporal. Por fim, nos resíduos estão as flutuações não explicadas pela tendência e pela sazonalidade, podendo conter ruído aleatório.

Ao concluir essa secção sobre sazonalidade, achou-se interessante efetuar uma decomposição MSTL do ano de 2020 tentando visualizar algum impacto da pandemia do Covid-19 no consumo de energia do *Campus 2* e, encontra-se na Figura 26, a seguir, o gráfico com a decomposição pretendida.

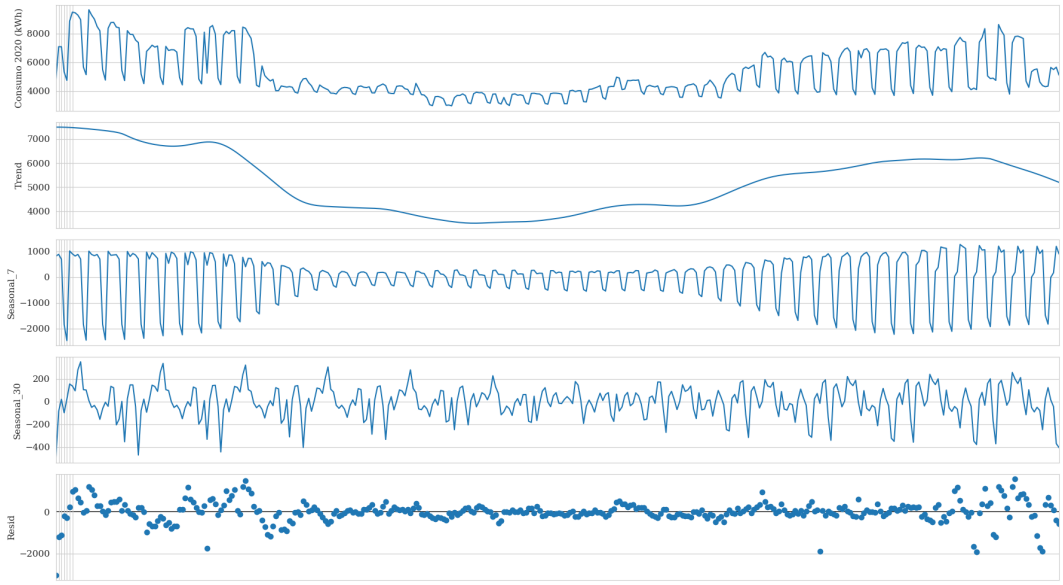


Figura 26: Decomposição multi sazonal da série temporal do consumo (2020).

Nota-se que no ano de 2020 o padrão de comportamento da série temporal do consumo sofreu alteração em todos os seus componentes. E ao se comparar a decomposição referente ao ano de 2020 com o ano de 2017, na Figura 25, que mostra a decomposição referente ao ano de 2017, e com o período de janeiro a outubro de 2022, na Figura 27, percebe-se claramente a diferença no consumo de energia, ocorrida em função da pandemia em 2020.

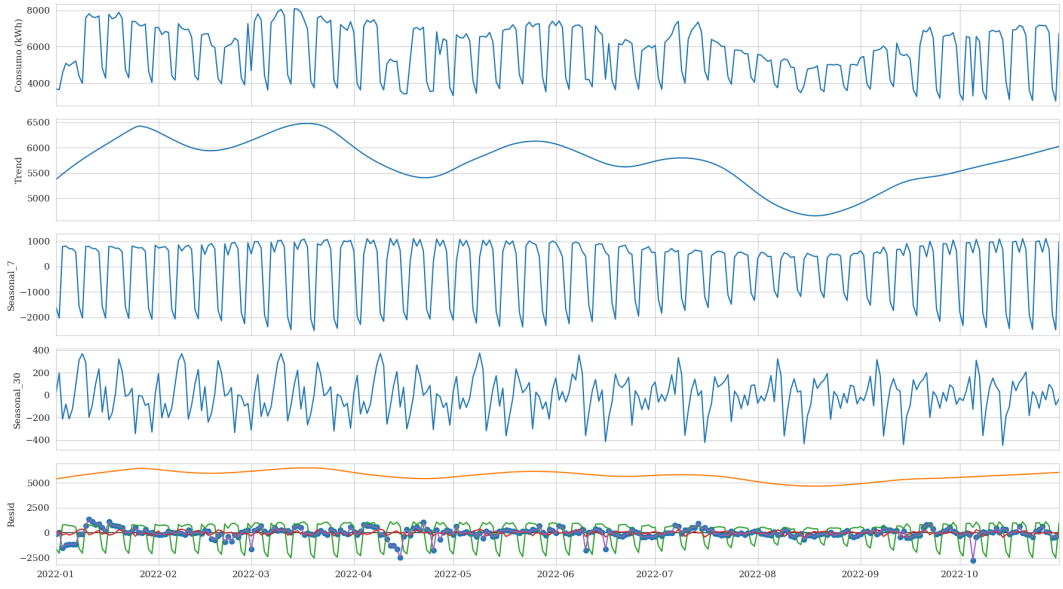


Figura 27: Decomposição multi sazonal da série temporal do consumo (jan-out/2022).

Por último, convém mencionar que a análise de um gráfico de decomposição MSTL é importante pois verifica-se como cada componente pode contribuir para a identificação e

entendimento de padrões e flutuações, além de auxiliar na escolha de modelos, parâmetros e algoritmos que serão utilizados na modelação da previsão, além do que o ajuste de hiperparâmetros pode-se tornar mais complexo com a presença de múltiplas sazonalidades [71].

4.3.3 *Normalidade*

Com o tipo de gráfico da Figura 28, pode-se dar sequência à análise dos dados, embora, é claro, não se possa obter uma informação sobre tendência com essa visualização inicial.

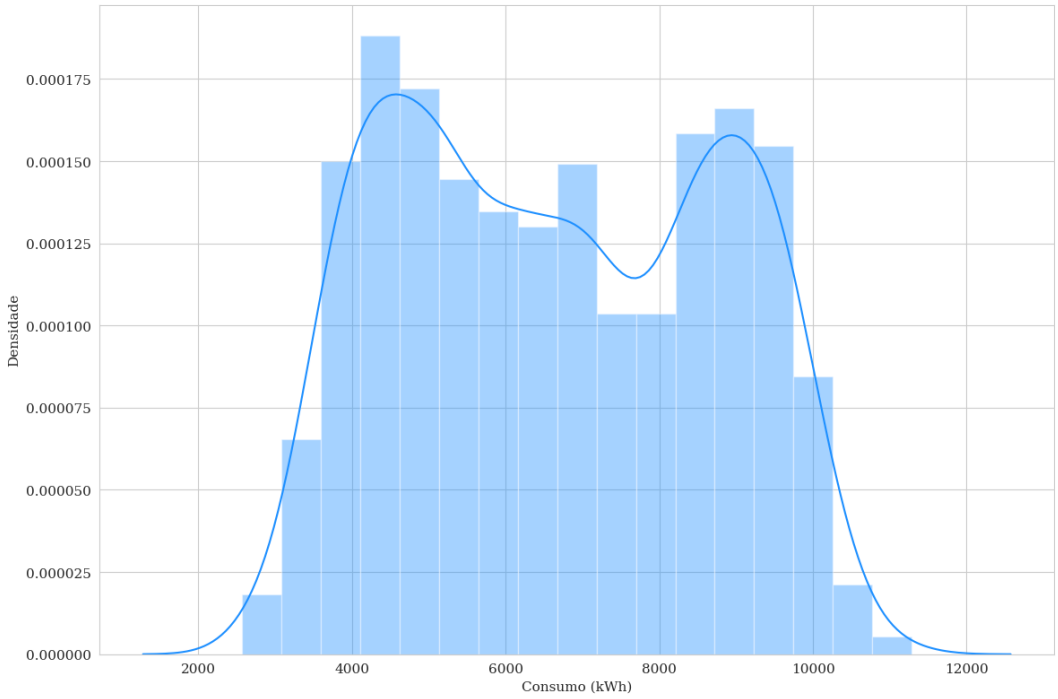


Figura 28: Histograma da série temporal do consumo.

A visualização do histograma revela que a série não segue uma distribuição normal e, ao ser inserida uma linha de densidade junto ao histograma, procurou-se perceber se a distribuição segue uma curva gaussiana ou não, ou se valores extremos são comuns. Necessário é incluir dois testes para confirmar a não normalidade dos dados, os quais foram efetuados mais adiante.

Pode ainda verificar-se se a série temporal do consumo de energia segue uma distribuição normal através do uso de um QQ-plot.

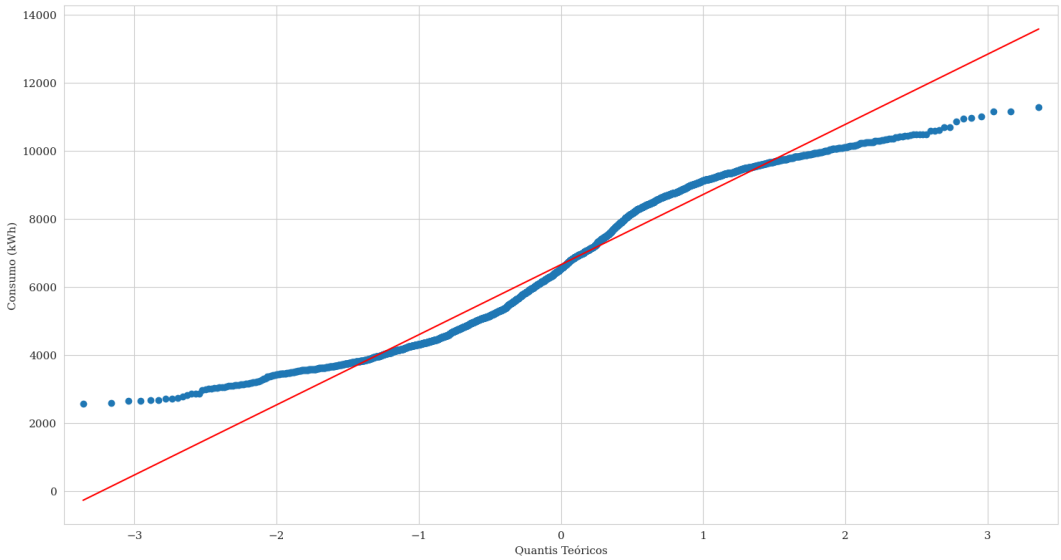


Figura 29: QQ *Plot* da série temporal do consumo.

Se os pontos do QQ-plot estiverem alinhados em uma linha reta, significa que a série temporal segue uma distribuição normal [14]. Contudo, nota-se na Figura 29 que não parece ser o caso da série temporal em análise neste trabalho.

Teste de Shapiro-Wilk sobre Normalidade para a Série Temporal do Consumo de Energia

O teste de Shapiro é baseado na comparação da distribuição dos dados com a distribuição normal. As hipóteses nula e alternativa deste teste são apresentadas na Tabela 7.

Tabela 7: Teste de *Shapiro-Wilk*.

Hipóteses para o Teste de Shapiro-Wilk	
Hipótese Nula (H_0):	A série temporal segue uma distribuição normal
Hipótese Alternativa (H_1):	A série temporal não segue uma distribuição normal
Alfa (nível de significância):	5%

Teste de Jarque-Bera sobre Normalidade para a Série Temporal do Consumo de Energia

O teste de Jarque-Bera é baseado na comparação da assimetria e da curtose dos dados com os valores esperados para uma distribuição normal, sendo suas hipóteses as que constam na Tabela 8. A estatística do teste de Jarque-Bera é sempre um número positivo e se estiver longe de zero, indica que os dados da amostra não apresentam uma distribuição normal [72].

Tabela 8: Teste de *Jarque-Bera*.

Hipóteses para o Teste de Jarque-Bera	
Hipótese Nula (H_0):	A série temporal segue uma distribuição normal
Hipótese Alternativa (H_1):	A série temporal não segue uma distribuição normal
Alfa (nível de significância):	5%

Resultados dos Testes sobre Normalidade para a Série Temporal do Consumo de Energia

Os resultados dos testes de normalidade de *Shapiro-Wilk* e *Jarque-Bera* relativos à série temporal do Consumo são apresentados na Tabela 9.

Tabela 9: Resultados dos testes de *Shapiro-Wilk* e de *Jarque-Bera* para a série temporal do consumo.

Teste	Estatística	*p-value*
Shapiro-Wilk	0.9547	1.7344×10^{-27}
Jarque-Bera	164.1949	2.2157×10^{-36}

Para ambos os testes, o valor do *p-value*, sendo menor que 5%, conduz à rejeição da hipótese nula, pois há evidência estatística de que a série temporal do consumo não segue uma distribuição normal. Esta conclusão é reforçada pela inspeção visual do histograma dos dados e da análise do Q-Q plot, apresentados nas figuras 28 e 29.

A constatação de que a série temporal do consumo não segue uma distribuição normal revela que poderá haver implicações nas previsões de modelos estatísticos ou de ML, afetando os parâmetros de previsão, portanto já se identifica uma possível necessidade de transformação, ou uma normalização dos dados para melhorar o desempenho de algoritmos sensíveis à distribuição dos dados [14].

4.3.4 *Estacionariedade*

É fundamental avaliar a estacionariedade (nível de estabilidade ou regularidade) de uma série temporal para saber de que modo o comportamento passado impacta o comportamento futuro dos dados [12]. Uma série temporal estacionária é aquela cujas propriedades estatísticas não dependem do momento em que a série é observada [4].

Teste ADF sobre Estacionariedade para a Série Temporal do Consumo de Energia

O Teste ADF verifica se uma série temporal é estacionária ou não. Quanto mais negativo o valor da estatística ADF, mais forte é a evidência de rejeitar a hipótese nula [73].

Tabela 10: Teste ADF.

Hipóteses para o ADF Teste	
Hipótese Nula (H_0):	A série temporal não é estacionária
Hipótese Alternativa (H_1):	A série temporal é estacionária
Alfa (nível de significância):	5%

Teste de Kwiatkowski-Phillips-Schmidt-Shin (KPSS) sobre Estacionariedade para a Série Temporal do Consumo

Quanto mais alto o valor da estatística KPSS, mais forte é a evidência de rejeitar a hipótese nula. Neste teste, a hipótese nula é que a série temporal é estacionária e procura-se evidências de que a hipótese nula é falsa. Consequentemente, *p-value* menor que o nível de significância estabelecido sugere que a diferenciação é obrigatória [4, 12].

Tabela 11: Teste KPSS.

Hipóteses para o KPSS Teste	
Hipótese Nula (H_0):	A série temporal é estacionária
Hipótese Alternativa (H_1):	A série temporal não é estacionária
Alfa (nível de significância):	5%

Resultados dos Testes de Estacionariedade para a Série Temporal do Consumo de Energia

Uma série temporal é estacionária quando suas propriedades estatísticas, como média, variância e autocorrelação, não dependem do tempo. Os resultados dos testes de estacionariedade do Teste *ADF* e *KPSS* relativos à série temporal do Consumo são apresentados na Tabela 12.

Tabela 12: Resultados dos testes *ADF* e *KPSS* para a série temporal do consumo.

Teste	Estatística	*p-value*
Augmented Dickey-Fuller (ADF)	-3.1483	0.0232
Kwiatkowski-Phillips-Schmidt-Shin (KPSS)	5.4960	0.01

Ambos os testes apresentam um *p-value* inferior a 5%. No caso do teste ADF, este resultado faz com que a hipótese nula seja rejeitada, indicando, portanto, a evidência estatística de que a série é estacionária. No entanto, no caso do teste KPSS a rejeição da hipótese nula indica que há evidência estatística de que a série não é estacionária. Há, por isso, a necessidade de transformá-la em uma série estacionária por meio de diferenciação, pois existe uma divergência entre os resultados obtidos com os dois testes.

Para saber quantas diferenciações seriam necessárias, recorreu-se à biblioteca *pmdarima*[7] e obteve-se como resultado que uma diferenciação ordinária na série temporal é necessária para estabilizar a tendência e a sazonalidade para que a série temporal seja estacionária. Após a diferenciação, efetuou-se uma comparação do histograma da série original e da série diferenciada para visualização e nova análise gráfica, conforme se pode ver na Figura 30.

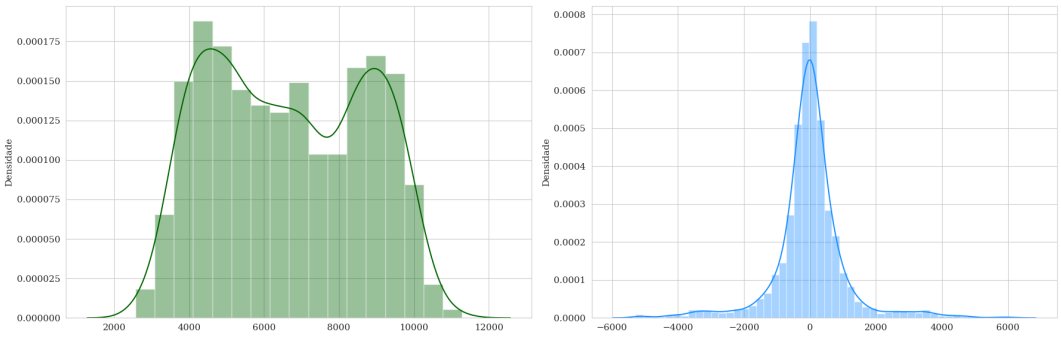


Figura 30: Histogramas da série original e diferenciada do consumo.

Finalmente, realizaram-se novamente os testes ADF e KPSS para a série diferenciada, cujos resultados são apresentados na Tabela 13.

Tabela 13: Resultados dos testes *ADF* e *KPSS* para a série temporal do consumo diferenciada.

Teste	Estatística	*p-value*
Augmented Dickey-Fuller (ADF)	-13.1850	1.1747×10^{-24}
Kwiatkowski-Phillips-Schmidt-Shin (KPSS)	0.01139	0.10

7 https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.utils.diff.html

O resultado do *p-value* inferior a 5% para o teste ADF faz com que a hipótese nula do ADF seja rejeitada. Portanto, há evidência estatística de que a série é estacionária. No caso do teste KPSS, sendo o resultado do *p-value* superior a 5%, a hipótese nula não é rejeitada, não havendo, por isso, evidência estatística de que a série não seja estacionária."

4.4 ANÁLISE DA SÉRIE TEMPORAL DA TEMPERATURA

Nesta secção faz-se uma análise visual da série temporal da Temperatura, pois é importante tentar identificar padrões, observações incomuns, mudanças ao longo do tempo e relações entre variáveis. Na Figura 31, tem-se a evolução da série temporal da temperatura utilizada neste trabalho.

Essas características devem ser incorporadas nos métodos de previsão, sempre que possível. Tal como na série temporal do Consumo, o tipo de dados determina o método de previsão a ser usado, bem como o tipo de gráfico mais adequado para a análise visual.

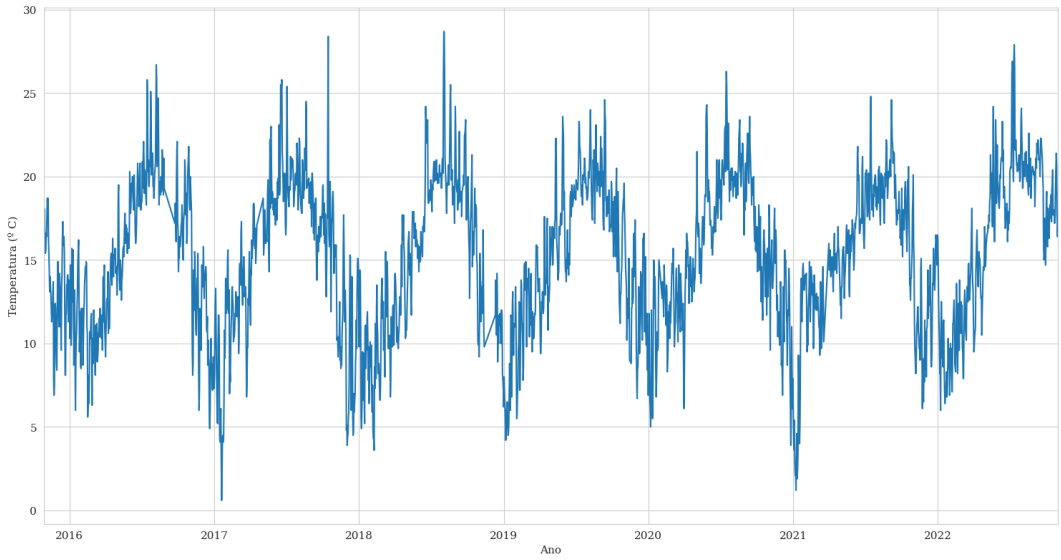


Figura 31: Evolução da série temporal da temperatura.

Ainda nesta secção são também efetuados os testes de *Shapiro-Wilk* e J*arque Bera* quanto à normalidade e os testes de *Dickey-Fuller Aumentado* (ADF) e o teste *Kwiatkowski-Phillips-Schmidt- Shin* (KPSS) de estacionariedade da série temporal da temperatura.

4.4.1 *Outliers*

Nas figuras 32, 33 e 34, a seguir, encontram-se os *boxplots* onde se pode visualizar informações sobre o padrão de comportamento da série temporal da Temperatura referentes à sua evolução mensal, semanal e diária.

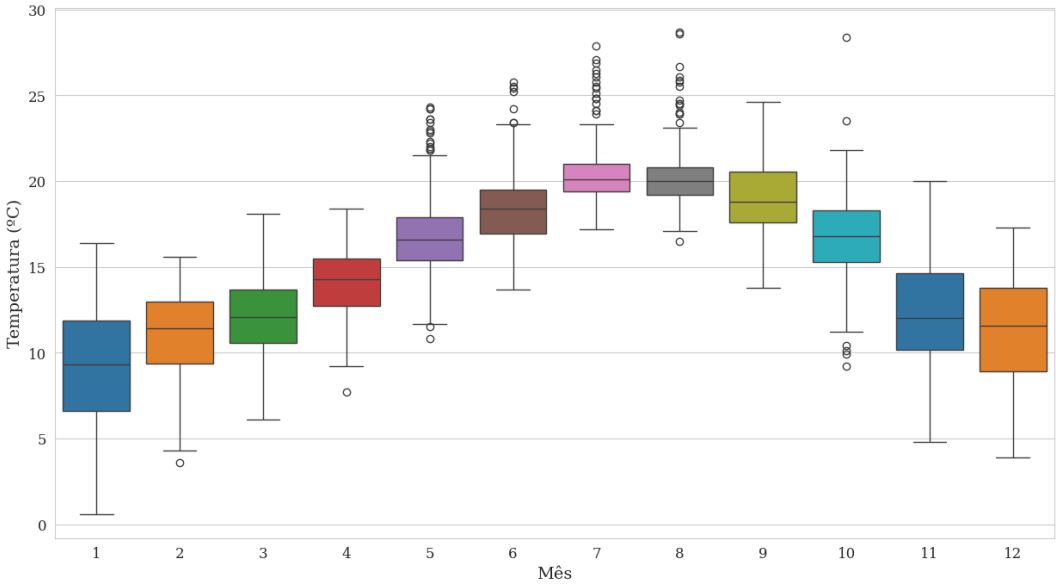


Figura 32: Evolução mensal da série temporal da temperatura.

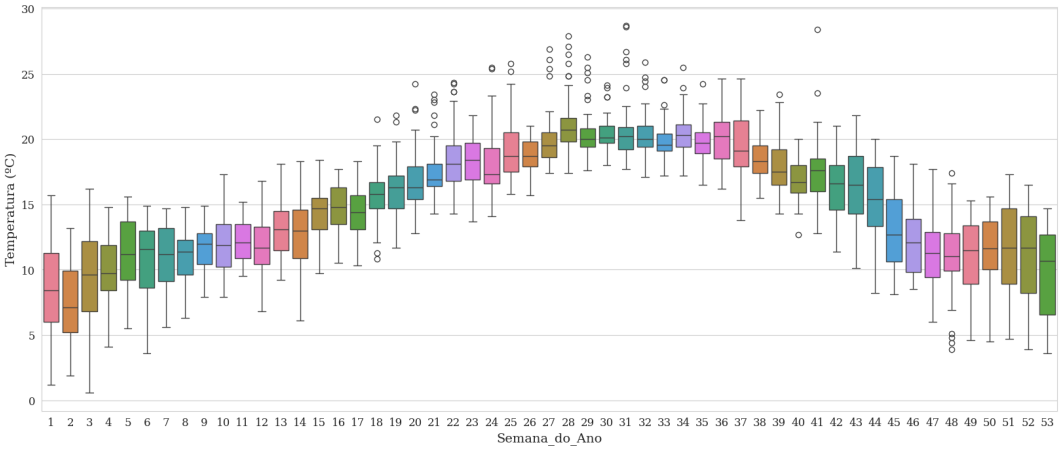


Figura 33: Evolução semanal da série temporal da temperatura.

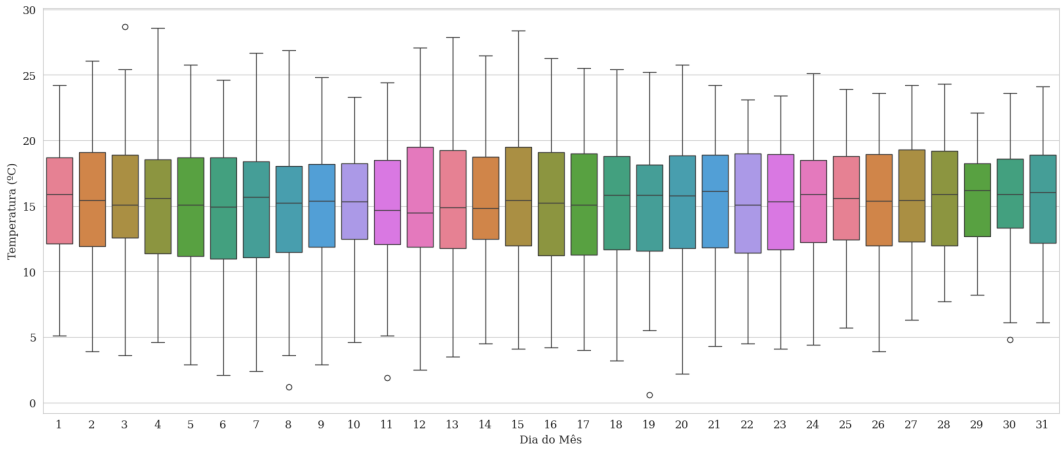


Figura 34: Evolução diária da série temporal da temperatura.

Destaca-se que nos três *boxpolts* são visualizadas observações na série temporal da temperatura que poderiam ser classificadas como *outliers*. No entanto, nota-se que à medida que a resolução dos *boxplots* passa de mensal para diária, há uma redução de observações que ficam fora da faixa padrão do *boxplot*.

De imediato fez-se um novo *boxplot* para verificação de *outliers*, cuja resultado se encontra na Figura 35.

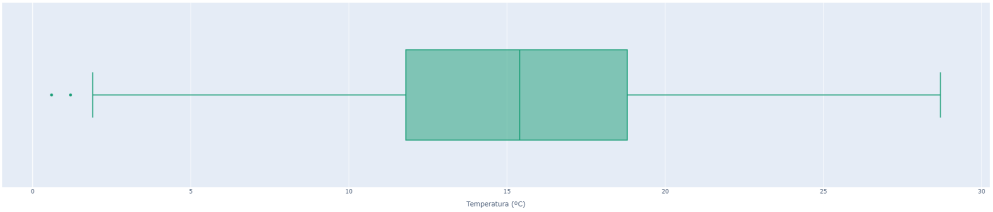


Figura 35: 1ª Verificação de valores discrepantes na série temporal da temperatura.

Nota-se, no *boxplot* da Figura 35, a identificação de duas observações da série temporal da temperatura como possíveis *outliers*, que devem ser investigados.

Da mesma maneira que foi utilizado o IF e o IQR[8] na série temporal do consumo, também aqui foi adotado o mesmo procedimento anteriormente descrito, e o resultado se encontra na Figura 36.

8 Três quartis: Q_1, Q_2 e Q_3, dividem um conjunto de dados ordenados em quatro partes iguais . O IQR é a amplitude do intervalo interquartil (IQR = $Q_3 - Q_1$).

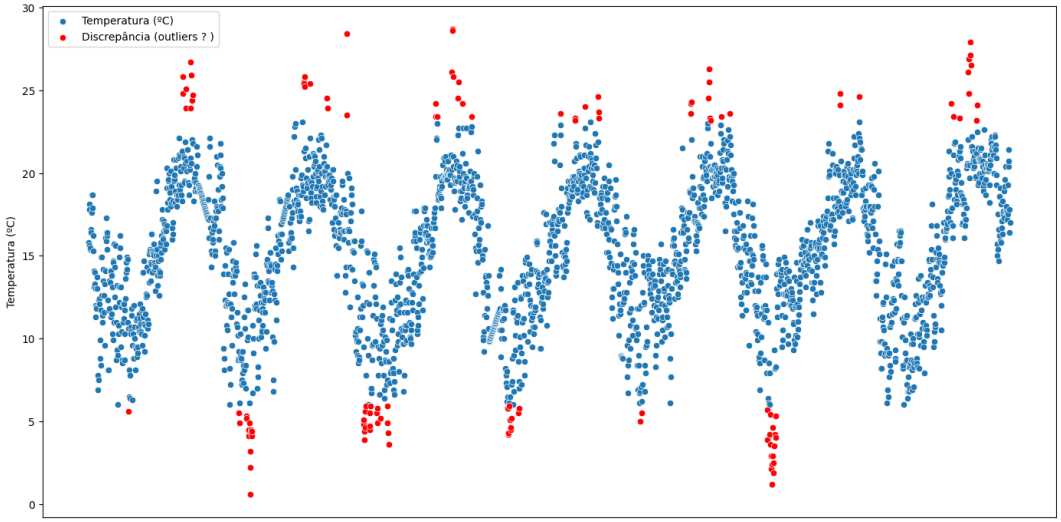


Figura 36: 2ª Verificação de valores discrepantes na série temporal da temperatura.

Dessa forma, o procedimento utilizando o IF e IQR, confirmou a existência de duas observações menores que $Q_1 - 1.5 \cdot \text{IQR}$. Por se tratar de somente duas observações, dentro de um conjunto de dados com 2561 observações, e também conforme destacado acima relativamente a dados da Tabela 2, decidiu-se não efetuar nenhuma remoção ou modificação nos valores originais daquelas observações, como relatado na literatura consultada. *Outliers* também ocorrem quando algumas observações são simplesmente diferentes. Neste caso, pode não ser sensato remover estas observações. A decisão de remover ou reter uma observação pode ser desafiadora [4].

4.4.2 *Sazonalidade*

Correlograma da Série Temporal da Temperatura

Analisar o padrão de comportamento dos dados verificando a autocorrelação da série temporal é um importante passo.

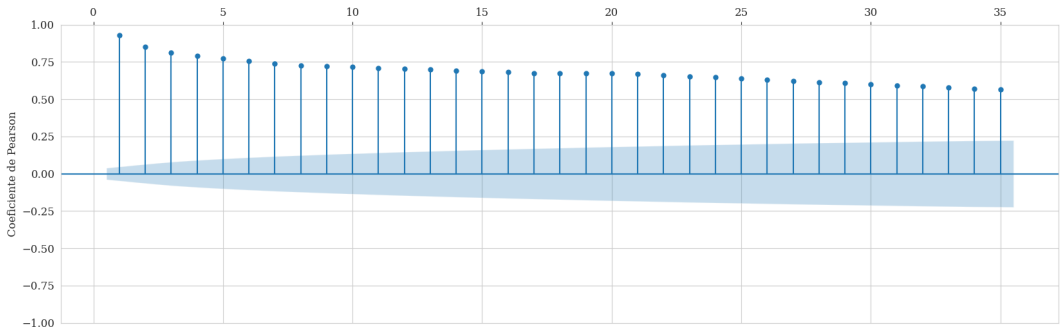


Figura 37: Função de Autocorrelação da série temporal da temperatura.

De forma distinta do gráfico de autocorrelação (ACF) da série temporal do consumo, onde se pode visualisar a cada 7 *lags* alterações sazonais, no gráfico de autocorrelação da série temporal da temperatura, percebe-se um decaimento suave à medida que o número de *lags* aumenta, indicando que os valores atuais estão correlacionados com os valores passados da série temporal.

Decomposição da Série Temporal da Temperatura

Com a decomposição pode-se visualizar cada componente individual, e tentar identificar a tendência e o padrão sazonal nos dados, como o exemplo na Figura 38, onde se percebe na série temporal da temperatura a variação sazonal ocorrendo em intervalos anuais.

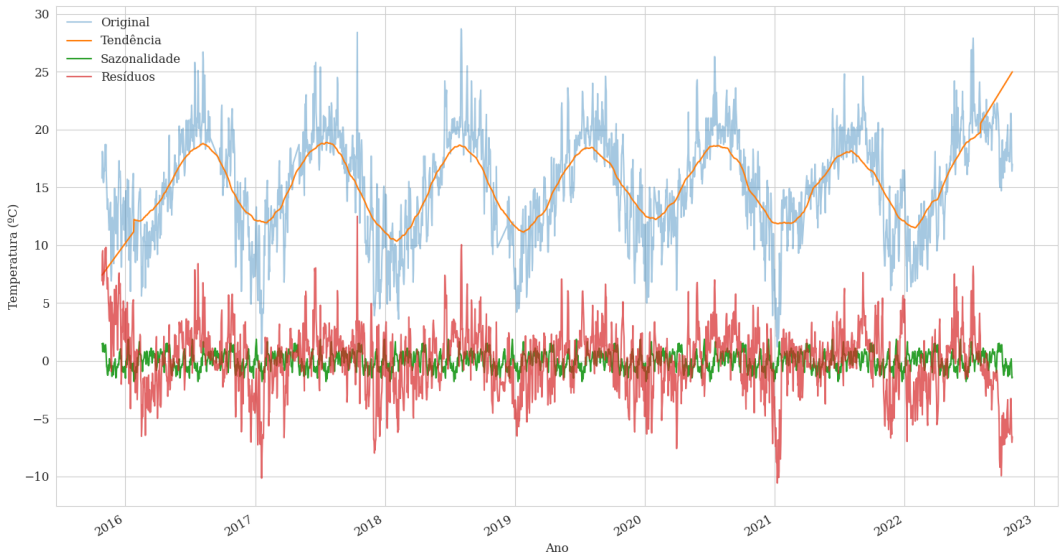


Figura 38: Decomposição Clássica Composta da série temporal da temperatura.

Na Figura 38 percebe-se a tendência ao redor dos 15 graus Celsius e a sazonalidade anual (em cor verde) da série temporal da temperatura.

4.4.3 *Normalidade*

A Figura 39 , mostra o histograma da série temporal da temperatura.

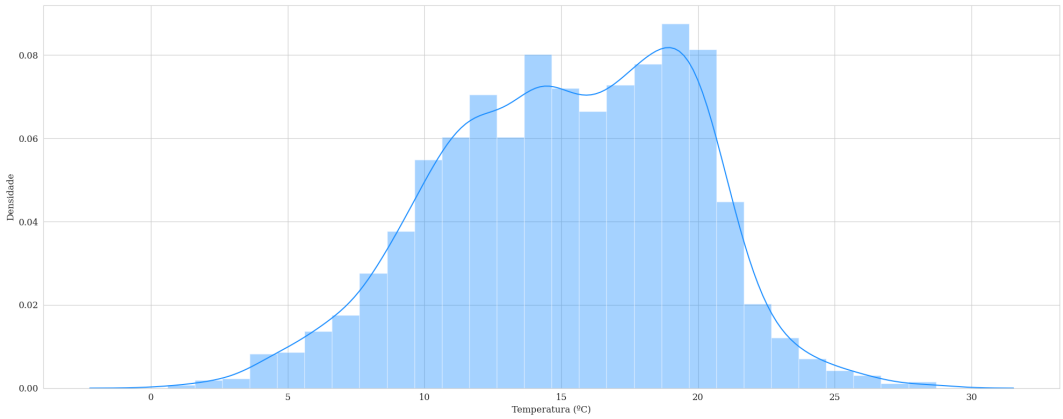


Figura 39: Histograma da série temporal da temperatura.

A visualização do histograma acima, deixa dúvidas quanto à normalidade da série temporal. O teste de *Shapiro-Wilk* e o teste de *Jarque-Bera* podem confirmar ou não a normalidade dos dados.

Convém ainda visualisar, através do QQ-plot, sobre a normalidade ou não da série temporal da temperatura, como se nota na Figura 40, e comparar com a Figura 29, referente ao consumo, para perceber diferenças no padrão de comportamento das duas séries temporais.

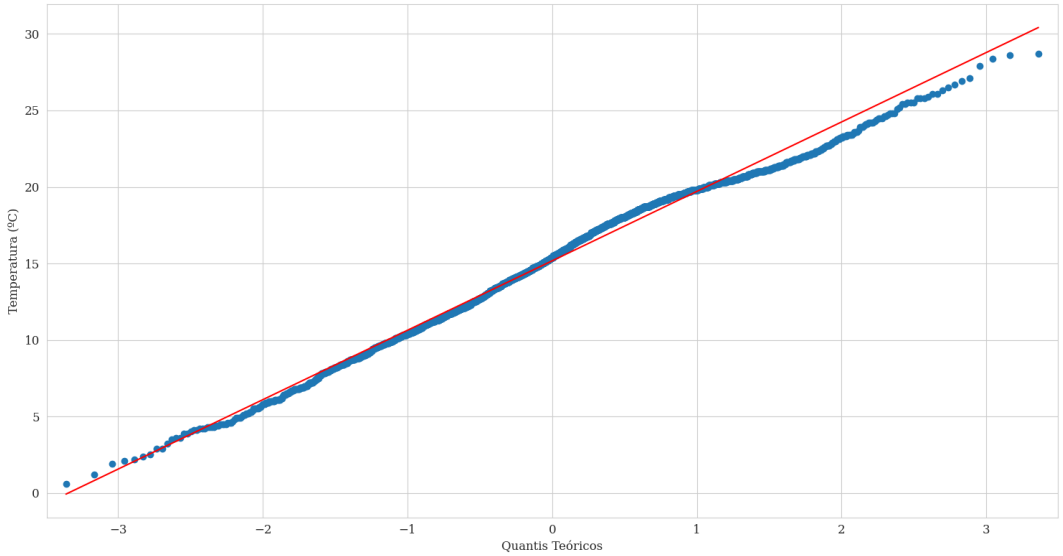


Figura 40: QQ *Plot* da série temporal da temperatura.

Pode-se notar que os extremos do QQ-plot se afastam da linha reta vermelha, indicando parecer que a série temporal da Temperatura não segue distribuição normal. Contudo, o QQ-plot é apenas uma ferramenta visual, e a confirmação relativa à normalidade da série temporal da Temperatura pode ser feita através dos testes de *Shapiro-Wilk* e de *Jarque-Bera*.

Os resultados dos testes de *Shapiro-Wilk* e de *Jarque-Bera* são apresentados na Tabela 14:

Tabela 14: Resultados dos testes de *Shapiro-Wilk* e de *Jarque-Bera* para a série temporal da temperatura.

Teste	Estatística	*p-value*
Shapiro-Wilk	0.9901	2.6680×10^{-12}
Jarque-Bera	36.9840	9.3113×10^{-09}

O valor do *p-value* em ambos os testes leva à rejeição da hipótese nula, significando que há evidência estatística de que a série temporal de temperatura não segue uma distribuição normal.

Séries temporais que não seguem uma distribuição normal podem afetar as estimativas dos parâmetros dos modelos desenvolvidos, trazendo implicações nas previsões de modelos estatísticos ou de machine learning [4, 14].

4.4.4 *Estacionariedade*

Para verificar se a série temporal de temperatura é estacionária, foram também realizados os testes *Augmented Dickey-Fuller Test* (ADF) e *Kwiatkowski-Phillips-Schmidt-Shin* (KPSS), cujos resultados se mostram na Tabela 15. Os resultados indicam a evidência estatística de que a série é estacionária, não havendo assim necessidade de efetuar uma diferenciação como aconteceu com a série temporal de consumo.

Tabela 15: Resultados dos testes *ADF* e *KPSS* para a série temporal da temperatura.

Teste	Estatística	*p-value*
Augmented Dickey-Fuller (ADF)	-3.2697	0.0163
Kwiatkowski-Phillips-Schmidt-Shin (KPSS)	0.1683	0.10

4.5 VISUALIZAÇÃO CONJUNTA DAS SÉRIES TEMPORAIS

Nesta secção faz-se uma breve análise visual conjunta da série temporal da Temperatura, com a série temporal do Consumo para perceção do padrão do relacionamento entre elas.

De imediato convém visualizar num mesmo gráfico, como se verifica na Figura 41, as duas séries temporais para tentar identificar alguma similaridade. Tendo em conta que a escala da série temporal da Temperatura é diferente da escala da série temporal do Consumo, para obtenção dos gráficos abaixo, os dados foram transformados para a mesma escala através da normalização `min_max`, que permite redimensionar os valores das variáveis para que fiquem dentro do intervalo [0, 1].

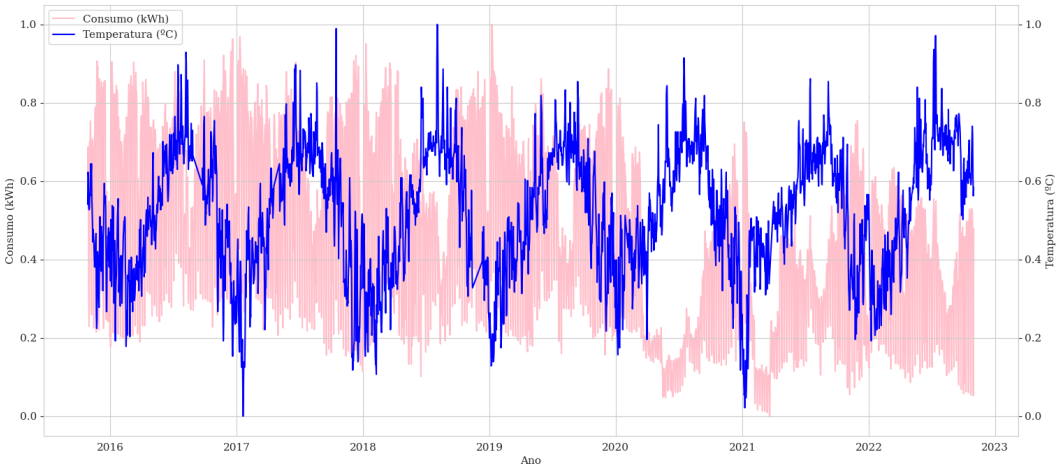


Figura 41: Evolução conjunta das séries temporais do consumo e da temperatura.

Visualmente já se percebe no gráfico dos dados da série temporal da Temperatura (com cor azul) uma sazonalidade anual, ficando nítido que há padrões de comportamento diferentes quando se compara com a série temporal do Consumo (com cor rosa). Porém, a análise visual de todo o período não permite outras conclusões.

Para efeitos de comparação, pode-se ainda verificar a diferença no padrão de comportamento semanal da série temporal do Consumo e da série temporal da Temperatura, visualizando as Figuras 42 e 43, dispostas abaixo.

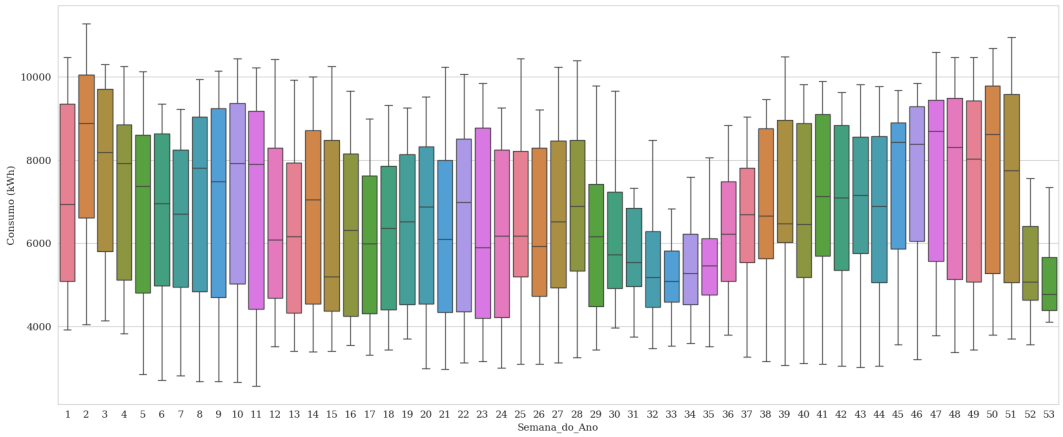


Figura 42: Evolução semanal da série temporal do consumo.

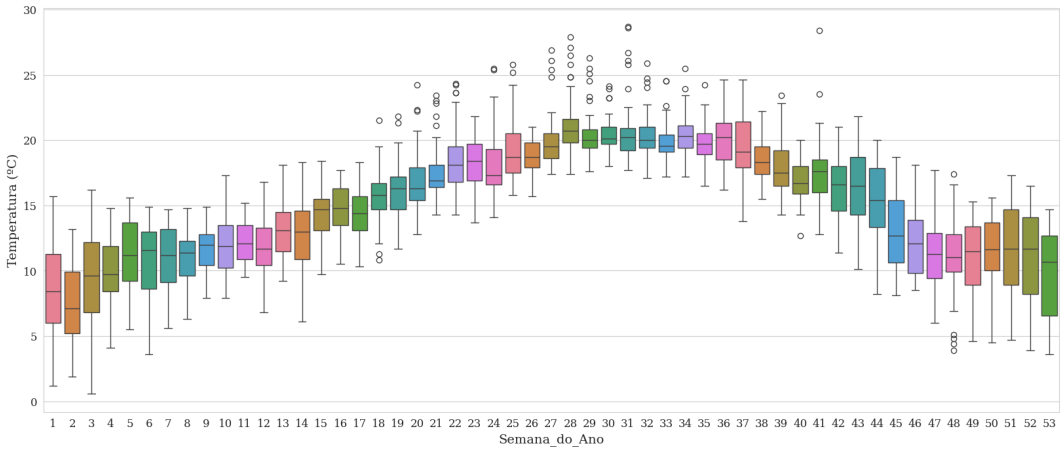


Figura 43: Evolução semanal da série temporal da temperatura.

Como complemento, pode-se ver que o correlograma das duas séries temporais, na Figura 44, mostra claramente uma autocorrelação da série temporal da Temperatura e consegue-se identificar uma clara sazonalidade semanal da série temporal do Consumo.

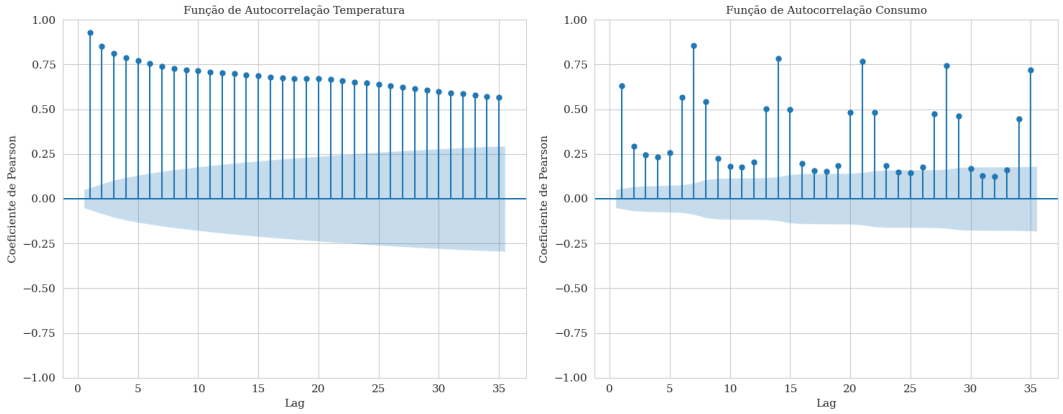


Figura 44: Autocorrelação das séries temporais da temperatura e do consumo.

Este capítulo apresentou as fases de análise e preparação dos dados efetuadas neste projeto e os problemas de previsão de consumo de energia, descrevendo as etapas executadas desde o tratamento e redução dos dados de consumo de energia e sua integração aos dados de temperatura. Neste capitulo também foram feitos os testes estatísticos adequados para desenvolvimento de modelos SARIMA-SARIMAX, e uma breve visualização das séries temporais em conjunto.

5

MODELAÇÃO ESTATÍSTICA

Neste capítulo é apresentado o processo de modelação realizado com os modelos SARIMA-SARIMAX, a utilização de validação cruzada em alguns modelos e os resultados obtidos. Dada a possibilidade de difícil ajuste dos parâmetros a utilizar com os modelos SARIMA e SARIMAX, optou-se pela seleção automática desses parâmetros com a utilização da função *auto_arima*() [23] para modelos SARIMA e da função *sarimax*() [24] tanto para modelos SARIMA quanto modelos SARIMAX.

5.1 CONJUNTO DE TREINO E TESTE

Previamente ao desenvolvimento dos modelos deste capítulo, foram efetuados testes de normalidade e estacionariedade nas variáveis *consumo* e *temperatura*, bem como utilizada uma diferenciação, para melhor entendimento e utilização dos dados. E para a construção destes modelos, o conjunto de dados foi dividido em treino e teste. Ao conjunto de teste foram atribuídos os últimos 365 dias do conjunto de dados, e no conjunto de treino foram considerados os primeiros 2196 dias.

5.2 MODELOS SARIMA

Após as análises gráficas, as verificações e os testes anteriormente descritos, optou-se pelo uso de ajuste de modelo automatizado construindo três modelos com a função *auto_arima*()[1] e três modelos com a função *sarimax*()[2]. Estas funções são utilizadas para estimar a melhor combinação de parâmetros do modelo. Foi também utilizada uma frequência $\boldsymbol{m}$ de sazonalidade igual a 7.

1 https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.arima.AutoARIMA
2 https://www.statsmodels.org/dev/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html

5.2.1 *Função Auto_Arima*

A análise gráfica e os testes estatísticos efetuados, foram considerados aquando da escolha da gama de parâmetros[74] a serem utilizados na função *auto.arima*(), sendo utilizado o conjunto de treino para estimação dos parâmetros, conforme mostra a Figura 45.

```
modelo2=pmd.auto_arima(train_df1,
                       start_p=0,
                       start_q=0,
                       d=1,
                       max_p=8,
                       max_q=8,
                       max_d=3,
                       start_P=1,
                       start_Q=1,
                       test = 'adf',
                       D=1,
                       max_P=3, max_D=1, max_Q=3, max_order=5,
                       m=7,
                       disp=-1,
                       seasonal=True,
                       trace=True,
                       error_action='ignore', suppress_warnings=True,
                       stepwise=True)
```

Figura 45: Parâmetros para o algoritmo de procura auto_arima.

Os parâmetros sugeridos pelo algoritmo são os seguintes: SARIMA(0, 1, 2)(1, 1, 2)[7][3]. Os gráficos da Figura 46 permitem realizar uma análise qualitativa dos resíduos deste modelo.

3 No Anexo A.1 se encontra o *output* com a busca efetuada pelo algoritmo auto_arima

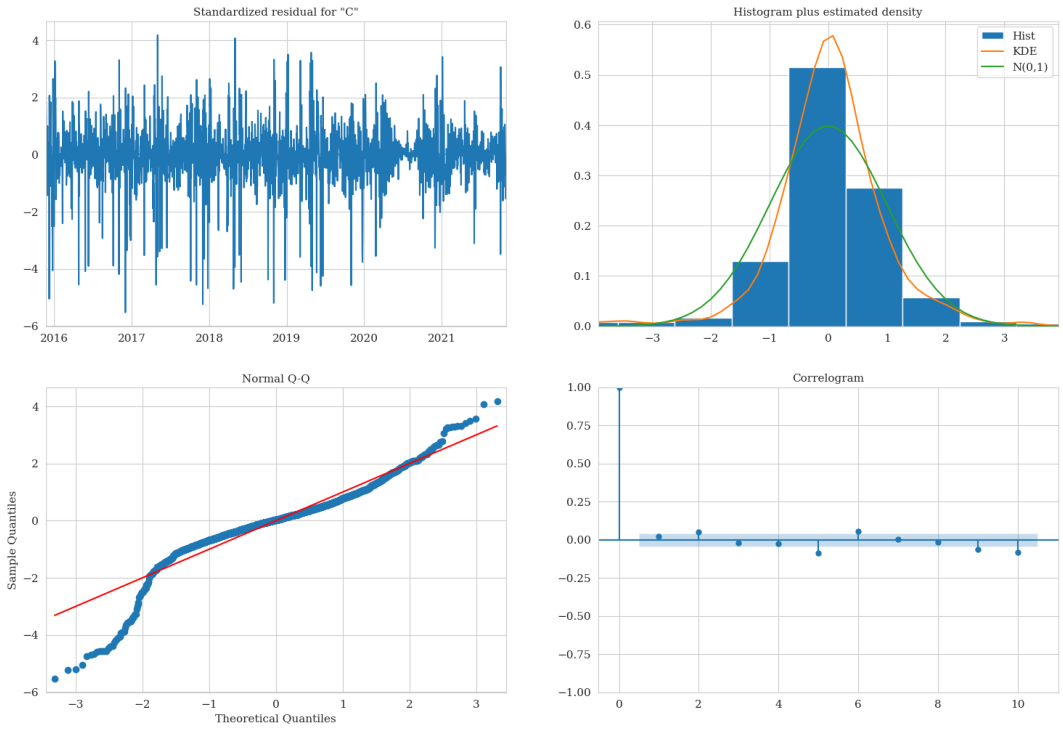


Figura 46: Análise qualitativa dos resíduos do modelo SARIMA(0, 1, 2)(1, 1, 2)[7].

Na Figura 46, o gráfico na parte superior esquerda mostra que os resíduos do modelo têm a média estável (não há tendência) o que é indicativo de estacionariedade. No entanto, o gráfico na parte superior direita mostra o histograma dos resíduos, onde se pode ver que os resíduos parecem não seguir uma distribuição normal.

Nota-se ainda no correlograma que os *lags* 2, 5, 6, 9 e 10 evidenciam a presença de autocorrelação. Sabe-se que o gráfico QQ compara a distribuição dos resíduos com uma distribuição normal teórica, e o QQ-plot na Figura 46, mostra que os pontos se afastam da linha reta sugerindo desvios da normalidade.

Portanto, na sequência, foi efetuada uma análise residual com um método quantitativo aplicando o teste Ljung-Box [14], sendo suas hipóteses as que constam na Tabela 16.

Tabela 16: Teste *Ljung-Box*.

Hipóteses para o Teste Ljung-Box	
Hipótese Nula (H_0):	Os Resíduos são distribuídos de forma independente.
Hipótese Alternativa (H_1):	Os Resíduos não são independentes e há autocorrelação em pelo menos um *lag*.
Alfa (nível de significância):	5%

Os *p_values* de 10 *lags*, obtidos com o teste *LJung_Box*, encontram-se listados na Tabela 17.

Tabela 17: *p_values* do teste *Ljung Box* para o modelo SARIMA(0, 1, 2)(1, 1, 2)[7].

Lag	*p_value*
1	1.441882×10^{-01}
2	8.227709×10^{-02}
3	7.777561×10^{-02}
4	1.113548×10^{-01}
5	5.950278×10^{-03}
6	9.523491×10^{-04}
7	1.903490×10^{-03}
8	3.254295×10^{-03}
9	1.644961×10^{-04}
10	7.773617×10^{-07}

O modelo SARIMA(0, 1, 2)(1, 1, 2)[7] apresenta um MAPE de 9.82% e um RMSE de 780.76 kWh. A análise dos *p-values* mostram que há evidência suficiente, ao nível de significância de 5%, para concluir que os resíduos deste modelo têm autocorrelação, e por consequência, poderá não ter bom ajuste em sua modelação.

O gráfico da Figura 47 permite comparar as curvas dos valores reais do conjunto de teste e as curvas com os valores previstos do modelo SARIMA(0, 1, 2)(1, 1, 2)[7][4] aonde sua visualisação confirma que as previsões do modelo não teve bom ajuste quanto aos pontos associados a máximos e minímos.

4 O dia zero no gráfico corresponde ao dia 01 de novembro de 2021, que se refere a primeira observação do conjunto de teste.

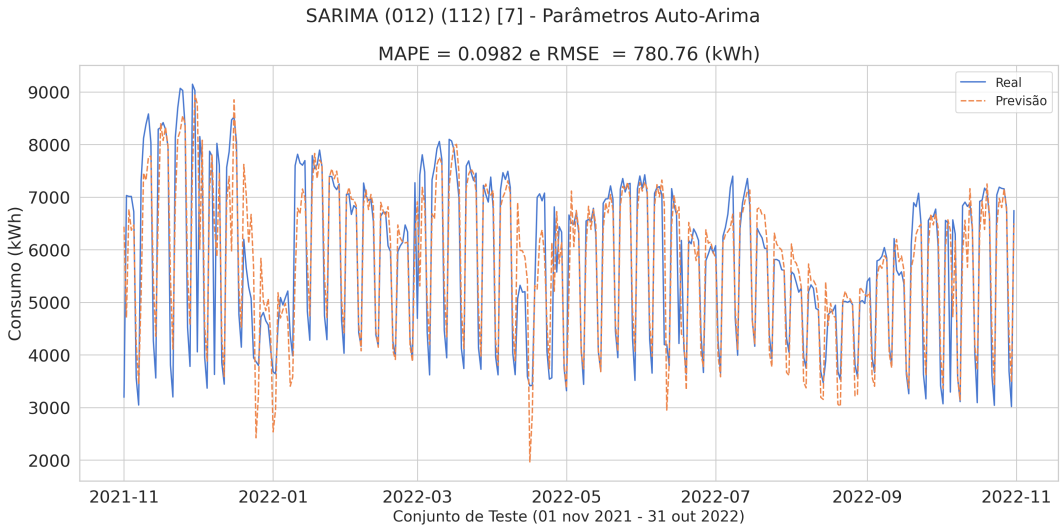


Figura 47: Valores reais *versus* previstos pelo modelo SARIMA(0, 1, 2)(1, 1, 2)[7].

Além do modelo SARIMA(0, 1, 2)(1, 1, 2)[7], utilizou-se a biblioteca *statsmodels* [75] para construir mais dois modelos SARIMA com pequenas alterações nos parâmetros (p, d, q) $(P, D, Q)_m$. São eles: o modelo SARIMA(0, 1, 1)(1, 1, 1)[7] e o modelo SARIMA(1, 1, 1)(1, 1, 2)[7], conforme detalhado no Anexo A.1. O valor dos respetivos MAPE e RMSE, destes três modelos se encontram na Tabela 18.

Destaca-se que a escolha pelo modelo SARIMA(1, 1, 1)(1, 1, 2)[7] foi feita propositadamente visando observar uma experiência utilizando parâmetros para modelação com AIC[5] infinito obtido pelo *auto_arima*, já que há poucas informações na literatura consultada a respeito de modelação com essa condição, conforme discutido no Anexo A.2.

5.2.2 *Função Sarimax para modelos SARIMA*

A análise gráfica e os testes estatísticos acima mencionados, também foram considerados aquando da definição do intervalo de valores possíveis para **p**, **q**, **P** e **Q**, ao se fazer uso da função *sarimax*(). O resto do procedimento permanece o mesmo, selecionando o modelo com o menor AIC e realização da análise residual. Encontram-se listados na Figura 48 os modelos sugeridos pela função *sarimax*().

5 Os critérios de informação (AIC) não são testes estatísticos, e o melhor modelo não pode ser considerado como a "verdade" [76].

```
Parâmetros
   (p, q, P, Q)     AIC
0  (3, 1, 3, 1) 35397.877932
1  (1, 2, 2, 3) 35411.248383
2  (2, 1, 3, 1) 35415.460382
3  (3, 1, 1, 2) 35416.600339
4  (1, 3, 1, 3) 35429.471697
...  ...  ...
250 (3, 0, 0, 0) 36444.148733
251 (0, 1, 0, 0) 36456.623825
252 (2, 0, 0, 0) 36470.020381
253 (1, 0, 0, 0) 36570.932916
254 (0, 0, 0, 0) 36770.714728

255 rows × 2 columns
```

Figura 48: Modelos SARIMA sugeridos pela função *sarimax*().

Logo a seguir à identificação dos três melhores parâmetros, a partir da função *sarimax*(), foram construídos modelos SARIMA e obtidos seus respetivos valores de MAPE e de RMSE. Na Tabela 18 encontram-se por ordem crescente do valor de MAPE dos modelos SARIMA, cujos parâmetros foram sugeridos pela função *sarimax*(), juntamente com o MAPE e o RMSE dos modelos obtidos a partir da função *auto_arima*().

Tabela 18: MAPE e RMSE dos modelos SARIMA.

Modelo Estatístico	**MAPE**	**RMSE** (kWh)	**Função**	**AIC**
SARIMA (1, 1, 1) (1, 1, 2)[7]	0.0936	773.72	Auto_Arima	inf
SARIMA (3, 1, 1) (3, 1, 1)[7]	0.0940	764.78	Sarimax	35397.878
SARIMA (2, 1, 1) (3, 1, 1)[7]	0.0941	770.34	Sarimax	35415.460
SARIMA (1, 1, 2) (2, 1, 3)[7]	0.0945	765.57	Sarimax	35411.248
SARIMA (0, 1, 2) (1, 1, 2)[7]	0.0982	780.76	Auto_Arima	35493.205
SARIMA (0, 1, 1) (1, 1, 1)[7]	0.1042	794.55	Auto_Arima	35615.978

A Tabela 18, mostra que o modelo SARIMA(1, 1, 1)(1, 1, 2)[7] tem o menor valor de MAPE (9.36%) entre todos os modelos SARIMA desenvolvidos, revelando assim que não se pode descartar de imediato um modelo com AIC infinito. No entanto, é o modelo SARIMA(3, 1, 1)(3, 1, 1)[7] que possui o menor valor de RMSE(764.78 kWh).

Convém mencionar que o RMSE e o MAPE são métricas complementares. O RMSE é útil para avaliar a magnitude dos erros, enquanto o MAPE é útil para avaliar o impacto dos erros em termos percentuais.

5.3 MODELOS SARIMAX

Após os modelos ARIMA Sazonal, foram desenvolvidos modelos SARIMAX, que incluem diferentes combinações das variáveis exógenas: Temperatura, Domingo e Feriado. Foi também utilizada uma frequência $\boldsymbol{m}$ de sazonalidade igual a 7.

5.3.1 *Visualização Gráfica Auxiliar*

Como primeiro passo verificou-se o coeficiente de Correlação Ponto-bisserial [77] para analisar o relacionamento entre as variáveis exógenas *domingo* e *feriado* (variáveis binárias) com a variável *consumo* - contínua - que se encontra na Figura 49, e a Correlação de *Spearman* referente a associação da variável *temperatura* com a variável *consumo* que se encontra na Figura 50.

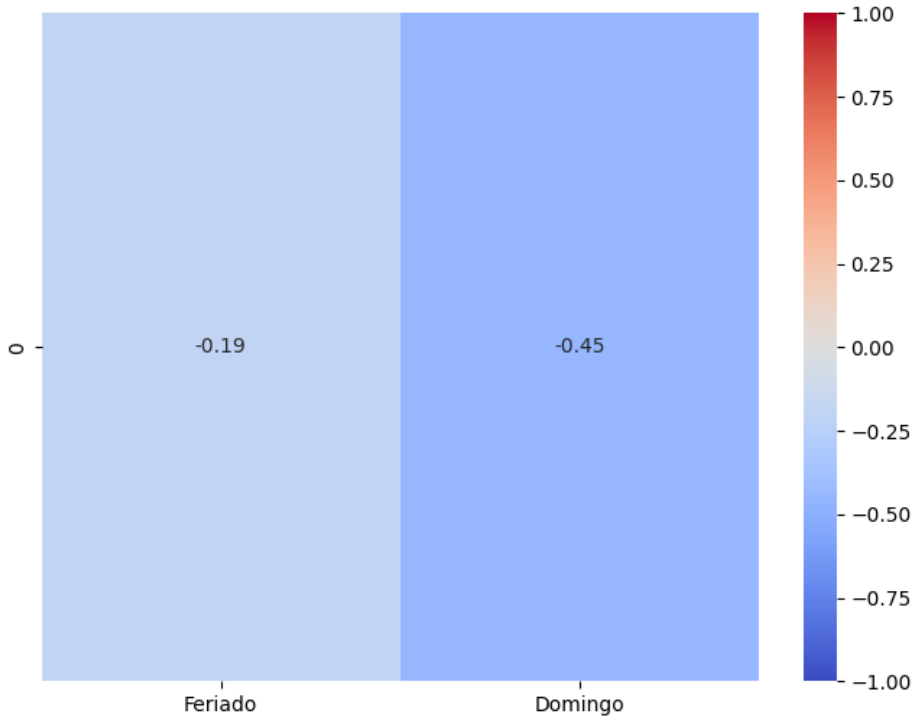


Figura 49: Matriz de correlação serial das variáveis *feriado* e *domingo* com a variável *consumo*.

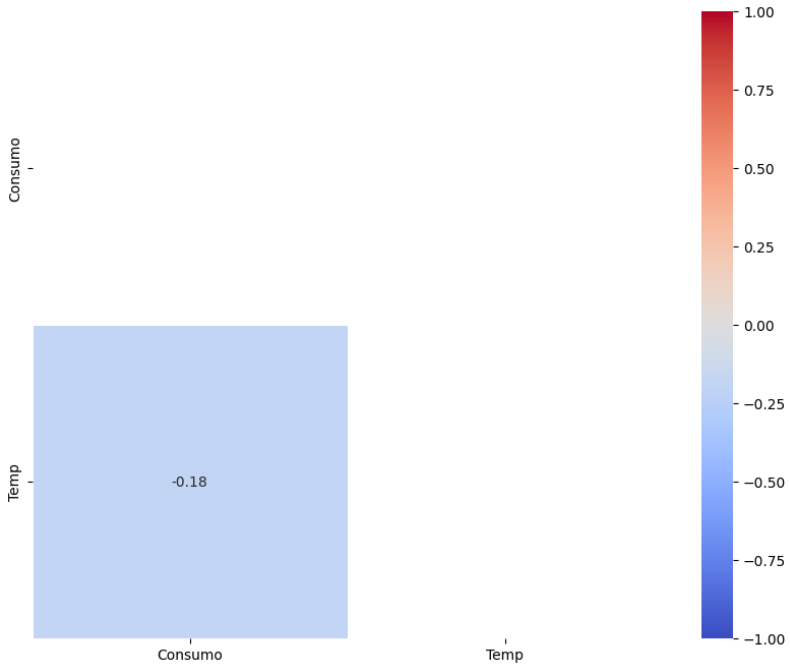


Figura 50: Matriz de correlação de *Spearman* da variável *temperatura* com a variável *consumo*.

Nota-se que o maior termo de correlação com a variável *consumo* é a variável *domingo* em sua correlação de -0.45, indicando uma associação moderada negativa, o que sugere que o consumo tende a ser menor aos domingos em comparação com outros dias da semana.

Por outro lado, as correlações mais baixas, em torno de -0.18 entre a variável *consumo* e a variável *temperatura* e ao redor de -0.19 entre a variável *consumo* e a variável *feriado* sugerem uma associação mais fraca. Isso pode indicar que a presença de feriado ou a temperatura pode ter menos impacto direto no consumo de energia em comparação com o domingo.

Relativamente à variável *feriado*, convém visualizar um *box-plot*, na Figura 51 , onde se mostra o efeito dos feriados no consumo diário de energia no *Campus* 2.

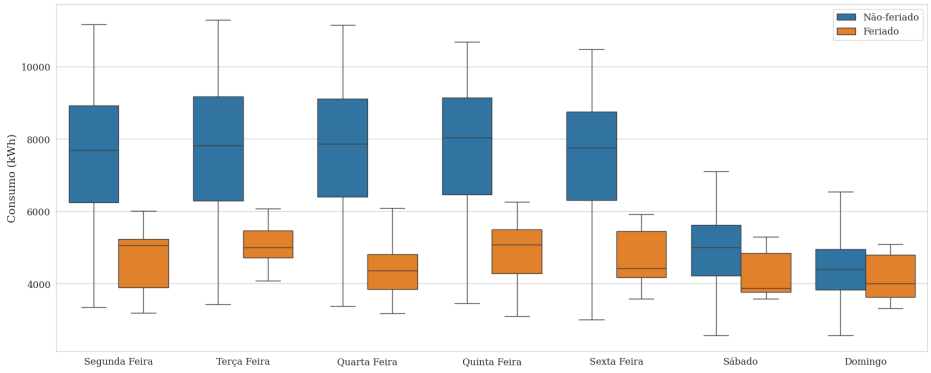


Figura 51: Efeito dos feriados na evolução diária do consumo.

O objetivo desse *plot*, na Figura 51, é auxiliar a análise da evolução do consumo ao longo da semana, diferenciando se é um feriado ou não. O *boxplot* fornece uma representação gráfica da distribuição dos dados, ajudando a identificar padrões e variações nos dados de consumo em diferentes dias da semana e nos feriados.

5.3.2 *Modelação SARIMAX*

Para efeitos de comparabilidade do valor de MAPE e de RMSE, inicialmente foram feitas experiências em modelos SARIMAX desenvolvidos com os mesmos parâmetros utilizados nos modelos SARIMA, listados anteriormente na Tabela 18.

- order=(1,1,1) e seasonal_order=(1,1,2)[7]
- order=(3,1,1) e seasonal_order=(3,1,1)[7]
- order=(2,1,1) e seasonal_order=(3,1,1)[7]
- order=(1,1,2) e seasonal_order=(2,1,3)[7]
- order=(0,1,2) e seasonal_order=(1,1,2)[7]
- order=(0,1,1) e seasonal_order=(1,1,1)[7]

Tendo em conta a combinação entre as variáveis exógenas com os parâmetros acima mencionados, foram construídos 30 modelos SARIMAX utlizando a biblioteca *statsmodels* [78]. Seguem abaixo: o *plot*, na Figura 52 com as curvas de previsão e dados reais do modelo SARIMAX(0, 1, 2)(1, 1, 2)[7] que obteve o menor valor de MAPE e a Tabela 19 com as métricas deste modelo. Na mesma Tabela 19 foi incluído para efeitos de comparação, as informações do modelo com as maiores métricas.[6]

6 O dia zero no gráfico corresponde ao dia 01 de novembro de 2021, que se refere a primeira observação do conjunto de teste.

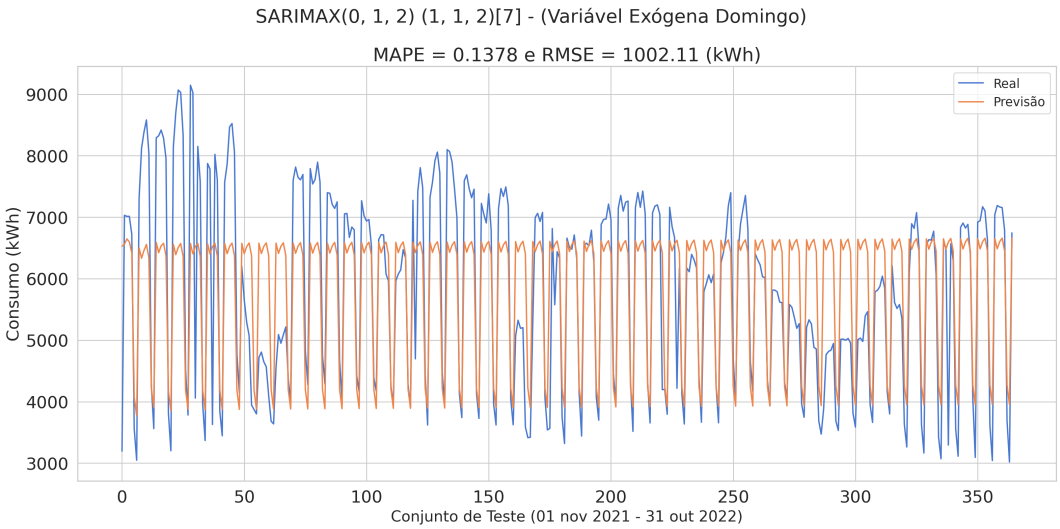


Figura 52: Valores reais *versus* previstos com o modelo SARIMAX(0, 1, 2)(1, 1, 2)[7].

Tabela 19: MAPE e RMSE dos modelos SARIMAX.

Modelo	**MAPE**	**RMSE** (kWh)	**Variável Exógena**
SARIMAX (0, 1, 2) (1, 1, 2)[7]	0.1378	1002.10	Domingo
SARIMAX (0, 1, 1) (1, 1, 1)[7]	0.6495	3832.92	Temperatura, Dom. e Fer.

Os resultados mostram que o modelo SARIMA(1, 1, 1)(1, 1, 2)[7] com o MAPE igual 9.36%, já destacado anteriormente na Tabela 18, tem melhor desempenho do que todos os modelos SARIMAX quando estes utilizam as variáveis exógenas com os mesmos parâmetros utilizados pelos modelos SARIMA.

5.3.3 *Função Sarimax para modelos SARIMAX*

Optou-se também pelo uso de ajuste de modelo automatizado construindo modelos com os parâmetros obtidos com a função *sarimax*(). Tal como aconteceu com os modelos SARIMA, foi realizada uma análise gráfica e também os testes estatísticos acima mencionados, aquando da definição do intervalo de valores possíveis para **p**, **q** e **P**, **Q**, ao se fazer uso da função *sarimax*(). O resto do procedimento permanece o mesmo, selecionando o modelo com o menor AIC e realização da análise residual. A Figura 53 mostra a parameterização dos modelos sugeridos pela função *sarimax*() por ordem crescente de AIC.

```
Parâmetros
     (p, q, P, Q)     AIC
0  (2, 1, 1, 2)  20.000000
1  (1, 3, 1, 3)  24.000000
2  (0, 3, 3, 3)  26.000000
3  (2, 3, 3, 3)  30.000000
4  (1, 3, 3, 3)  108.276082
...  ...  ...
234  (0, 1, 0, 0)  37744.690592
235  (3, 0, 0, 0)  37863.041767
236  (2, 0, 0, 0)  37916.759851
237  (1, 0, 0, 0)  38154.701886
238  (0, 0, 0, 0)  38191.768403
239 rows × 2 columns
```

Figura 53: Modelos SARIMAX sugeridos pela função *sarimax*().

Após a identificação dos parâmetros, e tendo em conta a combinação entre as variáveis exógenas com os parâmetros acima listados, foram construídos 15 modelos SARIMAX, com as parameterizações correspondentes aos **três menores** valores de AIC, constantes na Figura 53.

Visando uma informação comparativa, a Tabela 20 mostra os valores do MAPE e do RMSE dos modelos SARIMA e SARIMAX até aqui discutidos, com sua respetiva variável:

Tabela 20: Comparação das Métricas MAPE e RMSE dos modelos SARIMA/SARIMAX.

Modelo	**MAPE**	**RMSE** (kWh)	**Variável**
SARIMA (1, 1, 1) (1, 1, 2)[7]	0.0936	773.72	Consumo
SARIMA (3, 1, 1) (3, 1, 1)[7]	0.0940	764.78	Consumo
SARIMA (2, 1, 1) (3, 1, 1)[7]	0.0941	770.34	Consumo
SARIMA (1, 1, 2) (2, 1, 3)[7]	0.0945	765.57	Consumo
SARIMA (0, 1, 2) (1, 1, 2)[7]	0.0982	780.75	Consumo
SARIMA (0, 1, 1) (1, 1, 1)[7]	0.1042	794.55	Consumo
SARIMAX (1, 1, 3) (1, 1, 3)[7]	0.1229	896.90	Feriado
SARIMAX (1, 1, 3) (1, 1, 3)[7]	0.1287	942.32	Domingo, Temp. e Fer.
SARIMAX (0, 1, 2) (1, 1, 2)[7]	0.1378	1002.11	Domingo
SARIMAX (0, 1, 3) (3, 1, 3)[7]	0.2150	1307.72	Domingo
SARIMAX (0, 1, 1) (1, 1, 1)[7]	0.6495	3832.92	Domingo, Temp. e Fer.

Seguem abaixo os *plots* com as curvas de previsão e dados reais de dois destes modelos referidos na Tabela 20: o modelo SARIMA com menor valor de MAPE na Figura 54, e o modelo SARIMAX com menor valor de MAPE e RMSE na Figura 55.

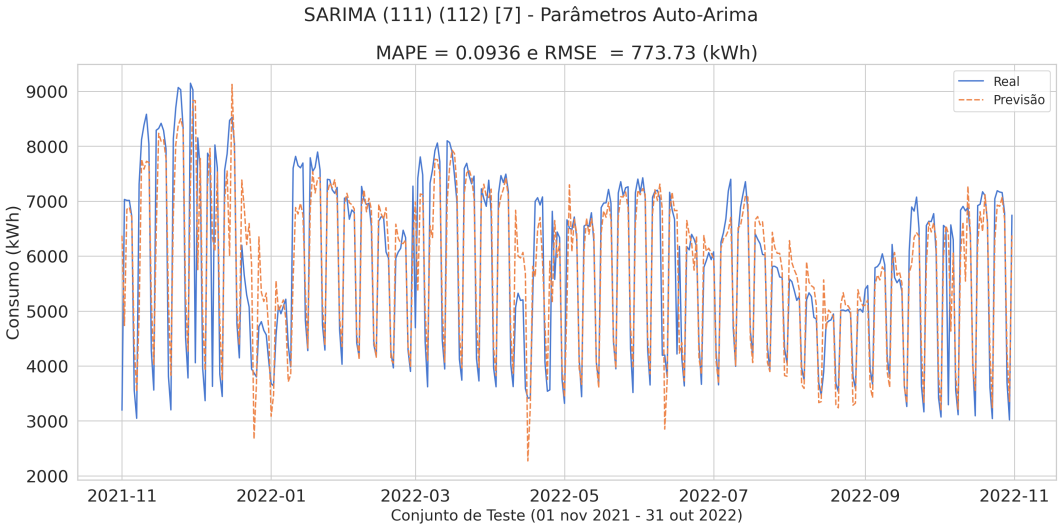


Figura 54: Valores reais *versus* previstos pelo modelo SARIMA(1, 1, 1)(1, 1, 2)[7].

No gráfico é visível o decréscimo de consumo na segunda quinzena do mês dezembro do ano de 2021 e na primeira quinzena do mês de janeiro de 2022. Nesse intervalo ocorre o período de recesso nas aulas e também há ocorrência de temperaturas mais baixas. Existem ainda, ocorrências semelhantes no mês de abril de 2022.

Esse tipo do ocorrência pode provocar problemas de identificação da continuidade do padrão de relacionamento entre as variáveis. Nas diversas experiências efetuadas ao longo deste trabalho, notou-se que esse facto é transversal a diversos modelos desenvolvidos.

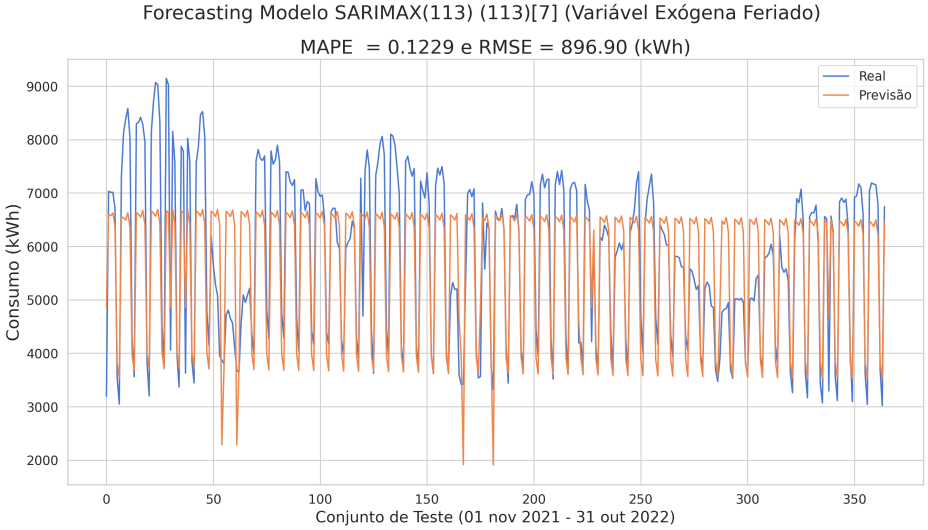


Figura 55: Valores reais *versus* previstos com o modelo SARIMAX(1, 1, 3)(1, 1, 3)[7].

Conforme se constata na Tabela 20, e nos gráficos com as curvas de previsão, o modelo SARIMA(1, 1, 1)(1, 1, 2)[7], com um valor de MAPE igual 9.36% e RMSE de 773.72 kWh tem mais robustez do que qualquer modelo SARIMAX.

Pode concluir-se esta secção mencionando que nas diversas experiências realizadas com os modelos SARIMA e SARIMAX, ficou evidenciado que os modelos SARIMA, somente com a variável *consumo*, obtiveram melhores valores de MAPE do que os modelos SARIMAX com utilização das variáveis exógenas.

5.4 VALIDAÇÃO CRUZADA

Nesta secção descrevem-se experiências com duas técnicas utilizadas neste projeto no que se refere a validação cruzada de séries temporais. O objetivo de utilizar tais técnicas de validação cruzada é o de verificar sua utilidade na melhoria dos modelos estatísticos deste trabalho, permitindo uma melhor comparação com o desempenho dos modelos de ML.

Numa modelação sem validação cruzada, a Figura 56 ilustra os conjuntos de treino e teste, onde as observações na faixa azul formam o conjunto de treino e as observações na faixa amarela formam o conjunto de teste.

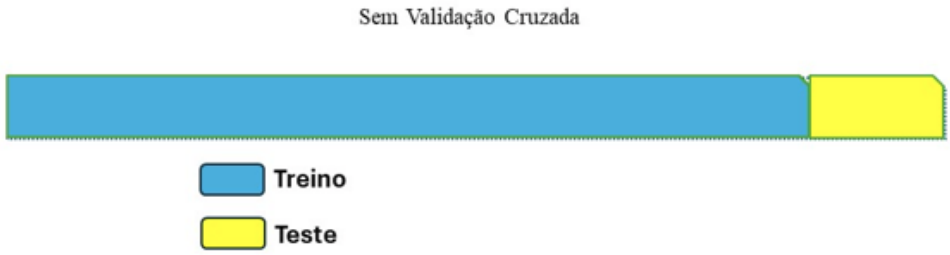


Figura 56: Modelação sem Validação Cruzada.

5.4.1 *Expanding Window*

A *Expanding Window* (EW) é uma abordagem de validação que consiste na utilização de uma janela como conjunto de treino que permite fazer previsões tendo em conta a temporalidade da série e diferentes tamanhos de dados no conjunto de teste. Isso possibilita explorar os recursos da série temporal, sendo o conjunto de treino expandido a cada

iteração do algoritmo, enquanto o conjunto de teste permanece um passo à frente. Ou seja, a otimização é realizada no conjunto de treino, que aumenta a cada iteração [35, 79].

O diagrama a seguir na Figura 57, ilustra os conjuntos de treino e teste, onde as observações na faixa verde formam os conjuntos de treino e as observações na faixa amarela formam os conjuntos de teste.

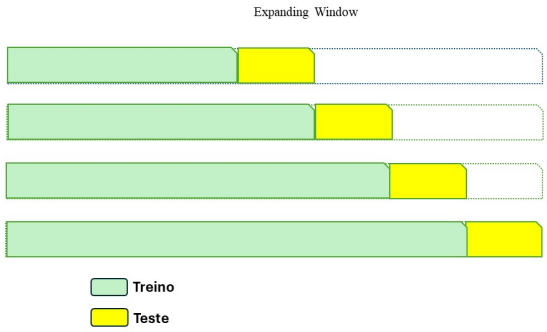


Figura 57: Técnica *Expanding Window*.

5.4.2 *Rolling Window*

O *Rolling Window* (RW) é uma técnica na qual o modelo é reajustado regularmente com uma janela deslizante de dados históricos à medida que novos dados se tornam disponíveis. A cada passo de tempo, a janela é atualizada, e o modelo é recalibrado com os dados mais recentes. Em seguida, o modelo é usado para fazer previsões para um determinado horizonte futuro com base nos dados disponíveis nessa janela. Essas previsões são comparadas com os valores reais para medir o desempenho do modelo [4].

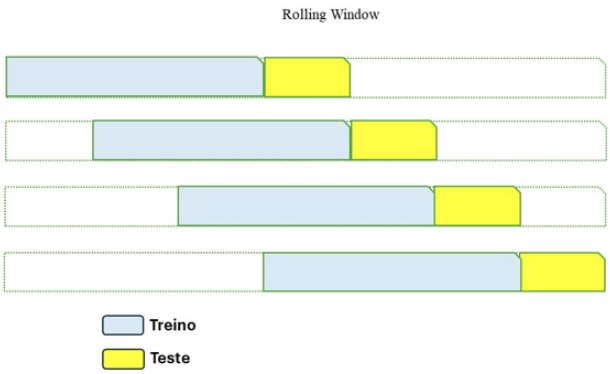


Figura 58: Técnica *Rolling Window*.

5.4.3 *Experiências*

Foram realizadas experiências em quatro dos muitos modelos construídos neste capítulo, que se enquadram nos seguintes requisitos:

- SARIMAX, utilizando as três variáveis exógenas: *domingo, temperatura e feriado*, com **menor** valor de MAPE, cujos parâmetros foram obtidos através da função *sarimax*();
- SARIMAX com o **maior** valor de MAPE com as três variáveis exógenas: *domingo, temperatura* e *feriado*;
- SARIMAX com o **menor** valor de MAPE com uma única variável exógena;
- SARIMA utilizando a variável *consumo*, com **menor** valor de MAPE.

Identificou-se, entre os modelos constantes na Tabela 20, os que atendem os critérios acima estabelecidos, e os mesmos são listados na Tabela 21.

Tabela 21: Seleção dos modelos SARIMA-SARIMAX para experiências com validação cruzada.

Modelo	**Parâmetros**	**Variável**
Mod 1	SARIMAX (1, 1, 3) (1, 1, 3)[7]	Domingo, Temp. e Feriado
Mod 2	SARIMAX (0, 1, 1) (1, 1, 1)[7]	Domingo, Temp. e Feriado
Mod 3	SARIMAX (1, 1, 3) (1, 1, 3)[7]	Feriado
Mod 4	SARIMA (1, 1, 1) (1, 1, 2)[7]	Consumo

Nas tabelas 22 e 23 encontram-se, respetivamente, os valores originais das métricas MAPE e RMSE dos modelos sem validação cruzada e os valores dessas métricas após as experiências em que foram aplicadas as técnicas de validação cruzada à esses modelos.

Tabela 22: MAPE dos modelos SARIMA-SARIMAX com Validação Cruzada.

Modelo	**MAPE** (Orig.)	**MAPE** (RW)	**MAPE** (EW)	**Variável**
Mod 1	0.1287	0.0838	0.0841	Domingo, Temp. e Feriado
Mod 2	0.6495	0.0897	0.0884	Domingo, Temp. e Feriado
Mod 3	0.1229	0.0949	0.0839	Feriado
Mod 4	0.0936	0.1007	0.1007	Consumo

O modelo SARIMAX(1, 1, 3)(1, 1, 3)[7], com a variável exógena *feriado*, obteve o valor de MAPE igual a 8.39% e o valor de RMSE igual a 647.23 kWh, após a aplicação da validação cruzada, e tem o melhor desempenho entre todos os modelos estatísticos desenvolvidos neste capítulo.

Tabela 23: RMSE (kWh) dos modelos SARIMA-SARIMAX com Validação Cruzada.

Modelo	**RMSE**(Orig.)	**RMSE** (RW)	**RMSE** (EW)	**Variável**
Mod 1	942.32	656.34	653.64	Domingo, Temp. e Feriado
Mod 2	3832.92	682.57	678.75	Domingo, Temp. e Feriado
Mod 3	896.90	765.61	647.23	Feriado
Mod 4	773.73	799.64	799.64	Consumo

Destaca-se ainda que no modelo SARIMAX(0, 1, 1)(1, 1, 1)[7], com as variáveis exógenas *domingo*, *temperatura* e *feriado*, ocorre uma grande redução da métrica MAPE de 64.95% para 8.84% e também no RMSE de 3832.92 kWh para 678.75 kWh, após o treino com a técnica *Expanding Window*, confirmando que o uso da validação cruzada melhorou significativamente o seu desempenho.

O gráfico da Figura 59 mostra as curvas de previsão obtidas no treino original do modelo SARIMAX(0, 1, 1)(1, 1, 1)[7], e a Figura 60 revela-nos uma outra imagem dos modelos obtidos com validação cruzada EW usando essa mesma parametrização.

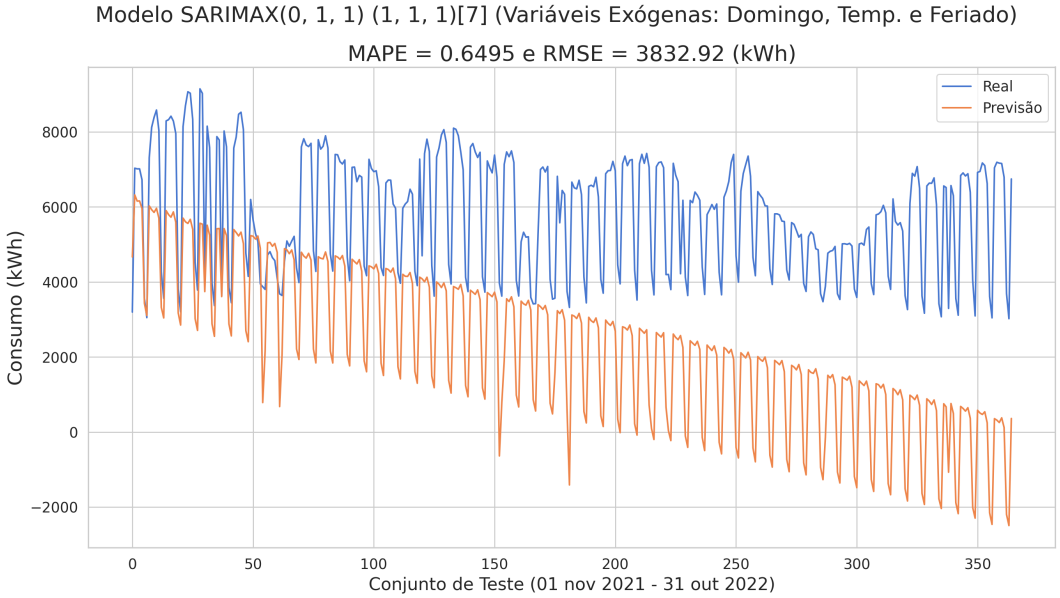


Figura 59: Valores reais *versus* previstos com o modelo SARIMAX(0, 1, 1)(1, 1, 1)[7] com as variáveis exógenas.

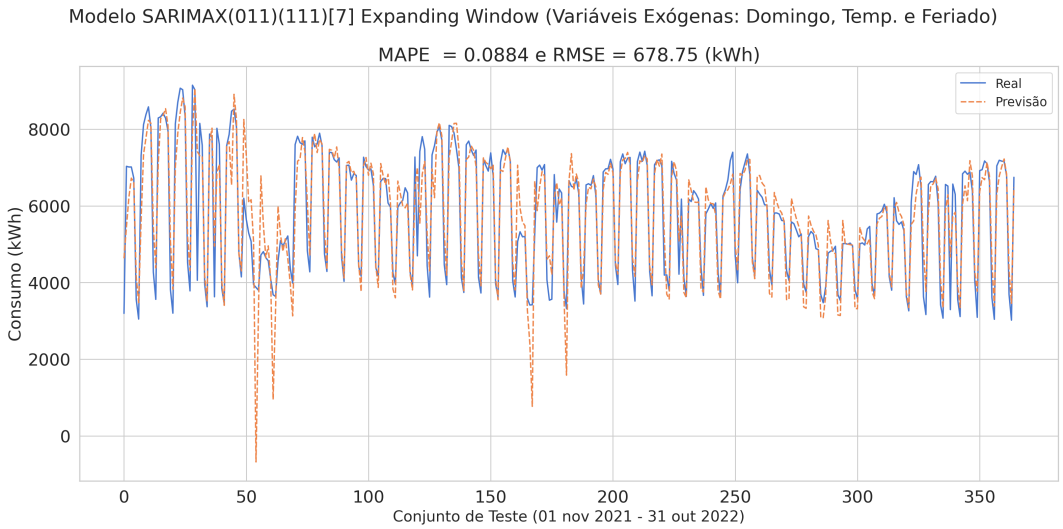


Figura 60: Valores reais *versus* previstos com o modelo SARIMAX(0, 1, 1)(1, 1, 1)[7] com *Expanding Window*.

O *plot* na Figura 60 revela que o modelo SARIMAX(0, 1, 1)(1, 1, 1)[7] com as três variáveis exógenas, pode ter conseguido incorporar informações quanto à presença das várias sazonalidades presentes na série temporal em estudo, após a aplicação da validação cruzada Expanding Window. Nota-se também no *plot* na Figura 60, que a mudança no padrão de comportamento dos dados afeta a predição para a segunda quinzena do mês dezembro do ano de 2021 e a primeira quinzena do mês de janeiro de 2022.

Ao concluir este capítulo, convém destacar que nas diversas experiências realizadas com os modelos SARIMA e SARIMAX, ficou evidenciado que os modelos SARIMAX obtiveram melhores valores de MAPE e de RMSE do que o modelo SARIMA após aplicação da validação cruzada. Estes resultados são obtidos quando se utilizam todas as variáveis exógenas (Domingo, Temperatura e Feriado).

6

MODELAÇÃO BASEADA EM APRENDIZAGEM COMPUTACIONAL

Neste capítulo apresentam-se os trabalhos realizados com os modelos de aprendizagem de máquina e o uso de redes neuronais para modelação da previsão do consumo de energia de curto prazo.

6.1 PROCEDIMENTO METODOLÓGICO

Foram utilizados para modelação os algoritmos KNN, XGBoost e as redes neuronais *Multilayer Perceptron* (MLP), *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU), enfatizando:

- A utilização das métricas MAPE e RMSE para avaliação do desempenho de previsão dos modelos, sendo utilizada uma janela (*days back*), especificada em 7 ou 14 dias; e,
- Quanto às redes neuronais, além de se obter os valores de MAPE, foram obtidos os respetivos valores do desvio padrão (σ) para análise da comparabilidade dos modelos.

Ressalta-se que as redes neuronais contêm uma componente estocástica no processo de treino devido à inicialização aleatória dos pesos. Isso leva a uma variabilidade nos resultados obtidos com os mesmos hiperparâmetros. Portanto, na avaliação do desempenho de um modelo, sobretudo num contexto de comparabilidade, a métrica a usar não se deve basear no resultado de uma única experiência [80]. Dessa forma, na modelação com as redes neuronais, decidiu-se verificar a variabilidade, conforme anteriormente mencionado, em 600 experiências para auxiliar na análise de busca de critério adequado para classificação de *melhor* modelo [80]. Portanto, foi acrescentada à análise a obtenção dos valores do desvio padrão dessa variabilidade para a comparabilidade dos modelos MLP, LSTM e GRU.

Para o desenvolvimento dos modelos, o conjunto de dados foi dividido em treino, validação e teste. Assim, no conjunto de teste foram considerados os últimos 365 dias, os primeiros 1792 dias como conjunto de treino e, para validação, utilizaram-se os 404 dias intermédios.

Em todos os modelos deste capítulo o conjunto de dados foi normalizado com o uso da seguinte fórmula:

```
min = df[conjunto de treino].min(axis=0)
max = df[conjunto de treino].max(axis=0)

df = (df - min) / (max - min),
```

onde `df` é o valor original dos dados, `min` é o valor mínimo do conjunto de dados, e `max` é o valor máximo.

Os valores das métricas de desempenho MAPE e RMSE dos modelos foram obtidas após desnormalização do valor previsto, para comparação com os valores do conjunto de teste.

A Figura 61, ilustra resumidamente os procedimentos adotados no desenvolvimento de modelos de ML e DL, de acordo com a metodologia estabelecida no Capítulo 3.

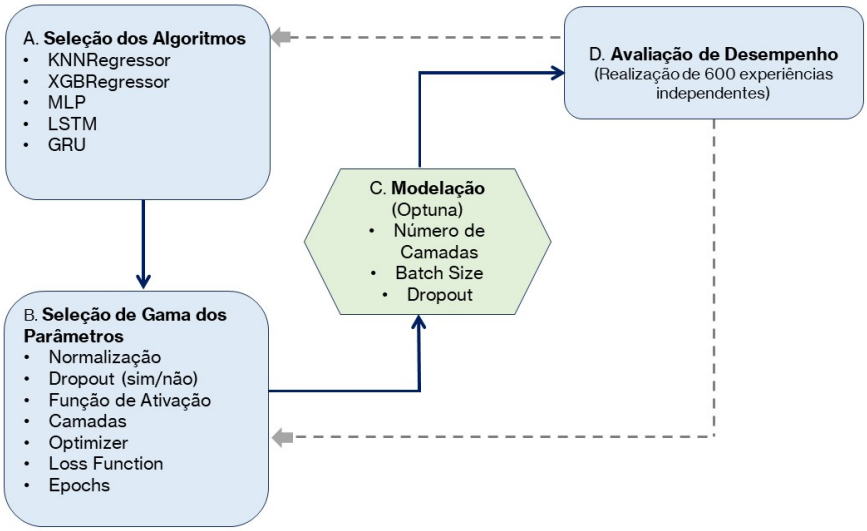


Figura 61: Metodologia adotada no desenvolvimento de modelos de ML e DL.

6.2 MODELAÇÃO COM KNN E XGBOOST

Nesta etapa do projeto, de entre diversas possibilidades, elencaram-se dois algoritmos que não redes neuronais, a saber: KNN e XGBoost. Estes algoritmos não estão nativamente preparados para previsão de séries temporais. Por isso, devem receber de uma vez só todos os valores necessários para produzir a saída. Assim, foi necessário transformar o conjunto

de dados de modo a que cada amostra inclua os valores dos ***n*** dias a serem utilizados para realizar essa previsão.

Neste trabalho, cada amostra deve consistir nos valores de consumo (e eventuais outras variáveis exógenas) dos 7 ou 14 últimos dias, de acordo com a janela temporal usada na modelação, para além do valor do consumo do dia que se pretende prever. Para isso, foi aplicada ao conjunto de dados uma transformação tabular[1] com a utilização da função *lagmat*()[2] da biblioteca *statsmodels* [81]. Porém, sabe-se que à medida que se acrescentam mais e mais variáveis, cada vez mais a sobreposição (ou seja, correlação) acontece entre as mesmas. Sendo as variáveis independentes altamente correlacionadas, o modelo poderá compensar os erros em uma variável com erros em outra, resultando em um valor geral de erro menor do que o real [82].

6.2.1 *KNNRegressor*

A opção pelo uso do algoritmo KNN se dá por ser um modelo não paramétrico, ou seja, não faz suposições sobre a distribuição dos dados. Baseia-se sim na ideia de que os valores futuros de uma série temporal são semelhantes aos valores passados que estão próximos, podendo ser usado para modelar uma variedade de séries temporais, incluindo séries temporais com tendência e sazonalidade [22].

Para o desenvolvimento dos modelos foi utilizada a classe KNNRegressor [83] da biblioteca *sklearn*. Foram efetuadas 72 experiências, com as seguintes características:

- Days back com tamanho: 7 e 14 dias;
- *K* vizinhos: 7, 14, 21 e 28;
- Normalização `min_max`;
- Variável *consumo*;
- Variáveis exógenas: *temperatura*, *domingo* e *feriado*.

KNNRegressor com a Variável Consumo

Encontram-se na Tabela 24, os resultados obtidos com os oito modelos considerando somente a variável *consumo*.

1 Uma transformação tabular na análise de séries temporais organiza os dados num formato estruturado e facilita a aplicação de técnicas de análise. O processo envolve transformar a série temporal bruta em uma tabela, onde cada linha representa uma amostra e cada coluna representa um atributo ou variável.

2 No contexto deste capítulo, "*lag*" se refere a janela de tempo, (ou *days back*) criada como nova coluna ao conjunto de dados.

Tabela 24: MAPE e RMSE dos modelos KNNRegressor com variável *consumo*.

Days_Back	***K*_Vizinhos**	**MAPE**	**RMSE** (kWh)
7	7	0.1152	848.54
7	14	0.1221	858.87
7	21	0.1238	845.58
7	28	0.1290	865.36
14	7	0.1345	898.34
14	14	0.1399	912.67
14	21	0.1430	931.73
14	28	0.1457	944.70

Considerando a métrica MAPE, o melhor desempenho entre os modelos KNN com a variável consumo, é o modelo com o MAPE igual a 11.52% , com *days back* e *k* vizinhos iguais a 7.

KNNRegressor com as Variáveis Exógenas

Também foram construídos 64 modelos considerando a combinação dos dois tamanhos de *days back*, dos quatro tipos de *k*_vizinhos, com as variáveis exógenas: *temperatura*, *domingo* e *feriado*. A Tabela 25, tem os valores do MAPE e RMSE dos modelos com melhor desempenho, por tamanho de janela igual a 7 e 14 dias, com a respetiva variável exógena.

Tabela 25: MAPE e RMSE dos modelos KNNRegressor com variáveis exógenas.

Days_Back	***K*_Vizinhos**	**MAPE**	**RMSE** (kWh)	**Variável Exógena**
7	7	0.1096	803.55	Domingo
7	14	0.1142	806.42	Domingo
7	21	0.1156	803.38	Domingo
7	28	0.1192	821.46	Domingo
14	7	0.1329	892.76	Domingo
14	14	0.1366	904.07	Domingo
14	21	0.1383	913.59	Domingo
14	28	0.1399	927.51	Domingo

A Tabela 25, revela que o melhor desempenho[3] entre os modelos KNN com as variáveis exógenas é o modelo KNN com *days back* e *k* vizinhos iguais a 7, com o MAPE igual a 10.96% e RMSE de 803.55 kWh. Ou seja, há um melhor desempenho ao acrescentar a variável *domingo* ao modelo, ao compararmos com o modelo KNN que usa somente a variável *consumo*, cujo MAPE é igual a 11.52% e RMSE igual 848.54 kWh.

Percebem-se ainda mais dois factos: *i*) quanto menor o tamanho de *days back* e o número de *k* vizinhos, menor o valor do MAPE, confirmando a estrutura decisória do KNN de escolher as observações mais recentes; *ii*) a variável *domingo* está presente em todos os melhores modelos com o KNNRegressor, independentemente do tamanho de janela ou *k* vizinhos.

6.2.2 *XGBRegressor*

Foram desenvolvidos modelos com o algoritmo XGBoost utilizando a classe XGBRegressor da biblioteca *sklearn*, considerando:

- *Days back* com tamanho igual a 7 e 14 dias;
- Variável *consumo*;
- Variáveis exógenas: *temperatura*, *domingo* e *feriado*;
- Uso da *framework* Otpuna para otimização dos hiperparâmetros;

3 Decidiu-se utilizar o rigor percentual na escolha pelo melhor desempenho dos modelos KNNRegressor.

- Com e sem normalização min_max;
- Uso de conjunto de validação.

O XGBoost funciona construindo árvores de decisão sequencialmente, com cada árvore buscando prever os resíduos da combinação de árvores anteriores e minimizando a *função perda* [12], possuindo assim, muitos hiperparâmetros que precisam de ser ajustados para obter um bom desempenho. Portanto, decidiu-se utilizar o Optuna para otimizar os hiperparâmetros em todos os modelos construídos com o XGBRegressor.

Conforme já destacado anteriormente, o XGBoost é um algoritmo baseado em árvores de decisão que não possui uma noção interna de sequência ou janela temporal. Dessa forma foi adicionado ao conjunto de dados janelas de tempo, ou características desfasadas (*lagged features*), para representar valores passados e ajudar o modelo a capturar dependências temporais.

Optou-se então, por desenvolver modelos com *days back* conforme acima descrito, utilizando a variável *consumo* e as variáveis exógenas, com dados normalizados para obter as métricas MAPE e RMSE, e posteriormente compará-las com modelos construídos com dados não normalizados.

XGBRegressor com a Variável Consumo

Para melhor visualização comparativa das métricas MAPE e RMSE dos quatro modelos XGBRegressor com a variável *consumo*, estas foram apresentadas em tabelas separadas, como listado nas tabelas 26 e 27.

Tabela 26: MAPE dos modelos XGBRegressor com e sem dados normalizados (variável *consumo*).

Days_Back	**MAPE**(min_max)	**MAPE**(não-norm.)	**Variável**
14	0.1237	0.1008	Consumo
7	0.1113	0.1072	Consumo

Observa-se nas tabelas 26 e 27, que o modelo XGBRegressor com um MAPE igual a 10.08% e RMSE igual 806.08 kWh tem o melhor desempenho, utilizando a variável *consumo*, com dados não normalizados e *days back* igual a 14 dias.

Convém destacar que, após a normalização min_max, o desempenho de ambos os modelos XGBRegressor utilizando a variável *consumo* piorou.

Tabela 27: RMSE dos modelos XGBRegressor com e sem dados normalizados (variável *consumo*).

Day_Back	**RMSE**(min_max)	**RMSE**(não-norm)	**Variável**
14	882.45	806.08	Consumo
7	828.75	846.88	Consumo

Percebe-se que, assim como na métrica MAPE, também houve aumento no valor do RMSE dos modelos XGBRegressor utilizando a variável *consumo*, após a normalização min_max.

XGBRegressor com as Variáveis Exógenas

Para melhor visualização comparativa das métricas MAPE e RMSE, dos dezasseis modelos XGBRegressor com as variáveis exógenas, estas foram apresentadas em tabelas separadas, como listado nas tabelas 28 e 29.

Tabela 28: MAPE dos modelos XGBRegressor com e sem dados normalizados (variáveis exógenas).

Days_Back	**MAPE**(min_max)	**MAPE**(não-norm.)	**Variável**
7	0.0948	0.1100	Temp., Domingo e Feriado
7	0.0997	0.1050	Domingo
14	0.0999	0.0972	Temp., Domingo e Feriado
14	0.1006	0.0994	Temp. e Domingo
14	0.1023	0.0999	Temp. e Feriado
7	0.1054	0.1031	Temp. e Domingo
14	0.1061	0.0966	Domingo
7	0.1095	0.1095	Temp. e Feriado

Visualiza-se na Tabela 28 que o modelo XGBRegressor com melhor desempenho tem um MAPE igual a 9.48%, utilizando todas as variáveis exógenas, dados normalizados e *days back* igual a 7 dias.

Os valores da métricas RMSE dos modelos com XGBRegressor, com e sem aplicação da normalização min_max, são apresentados na Tabela 29:

Tabela 29: RMSE dos modelos XGBRegressor com e sem dados normalizados (variáveis exógenas).

Days_Back	**RMSE**(min_max)	**RMSE**(não-norm)	**Variável**
14	770.23	792.44	Temp. e Domingo
7	774.89	860.17	Temp., Domingo e Feriado
14	784.57	781.31	Temp., Domingo e Feriado
7	791.29	834.96	Temp. e Domingo
14	791.54	823.55	Temp. e Feriado
7	791.97	831.44	Domingo
14	799.76	793.46	Domingo
7	833.67	834.31	Temp. e Feriado

Na compararação das tabelas 28 e 29, percebe-se que o modelo que possui o menor valor de MAPE não é o mesmo modelo que possui o menor valor de RSME. Identifica-se aqui, o mesmo ocorrido entre dois modelos SARIMA listados na Tabela 18 no Capítulo 5.

E como mencionado anteriormente, o RMSE e o MAPE focam em tipos distintos de erros: o RMSE mede os erros quadráticos, enquanto o MAPE mede os erros percentuais. Por isso, um modelo pode apresentar melhor desempenho em uma determinada métrica, enquanto outro modelo se destaca na outra.

Dessa forma, segue na Figura 62, o *plot* com as curvas de previsão e dados reais do modelo XGBRegressor da Tabela 29 com o valor de RMSE igual a 770.23 kWh considerando as variáveis exógenas *temperatura* e *domingo*, com um *days back* igual a 14 e dados de treino normalizados.[4]

4 A posição 2200 no gráfico corresponde ao dia 01 de novembro de 2021, que se refere a primeira observação do conjunto de teste.

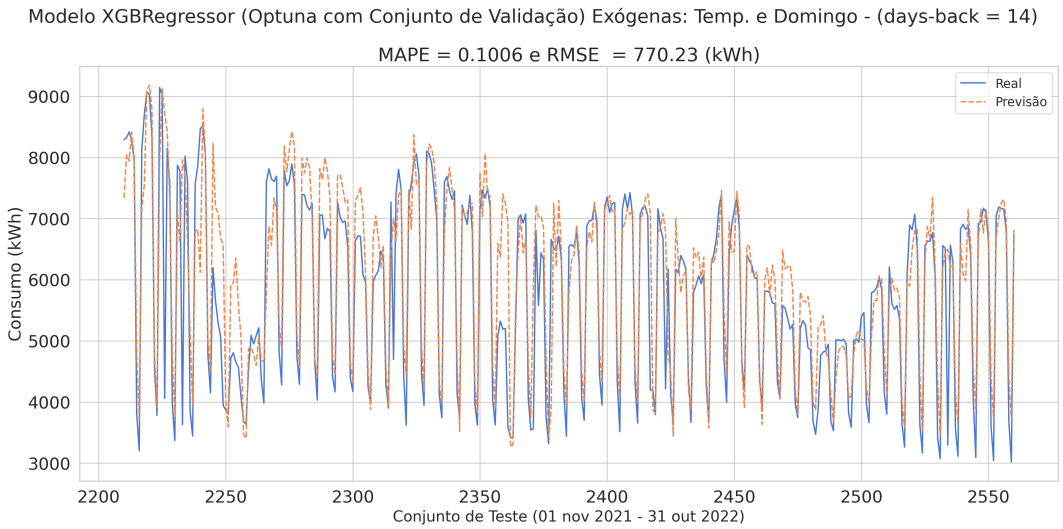


Figura 62: Valores reais *versus* previstos com o modelo XGBRegressor com menor RMSE.

Conforme já mencionado anteriormente, também aqui se percebe no *plot* da Figura 62, que uma alteração ocorrida no padrão de comportamento dos dados afetou a predição para a segunda quinzena do mês dezembro do ano de 2021 e a primeira quinzena do mês de janeiro de 2022. No entanto, no caso desse modelo o *plot* mostra também que as predições ficam mais desencontradas naqueles pontos referentes aos pontos máximos e mínimos do consumo de energia.

6.2.3 *Comparação XGBRegressor e KNNRegressor*

Optou-se por comparar esses dois modelos pois ambos utilizam algoritmos de aprendizagem de máquina, porém o KNN é um modelo não paramétrico (não faz suposições sobre a forma da relação entre as variáveis), e XGB um modelo paramétrico. Ou seja, este último permite ter suposições sobre a forma da relação entre as variáveis. Em comum nestas duas variantes, o *days back* é igual a 7.

Uma comparação dos modelos XGBRegressor e KNNRegressor de melhor desempenho se encontra na Tabela 30.[5]

5 Decidiu-se utilizar o rigor percentual na escolha pelo melhor desempenho dos modelos XGBRegressor.

Tabela 30: Comparação do MAPE dos modelos XGBRegressor com dados normalizados e KNNRegressor.

Modelo	**Tamanho de Janela**	**MAPE**	**RMSE**(kWh)	**Variável**
XGBRegressor	7	0.0948	774.89	Temp., Domingo e Feriado
KNNRegressor	7	0.1096	803.55	Domingo

A Figura 63, mostra o *plot* com as curvas de previsão e dados reais do modelo XGBRegressor com o menor valor de MAPE considerando as variáveis exógenas *temperatura*, *domingo* e *feriado*, com uma janela de tempo igual a 7 e dados de treino normalizados.

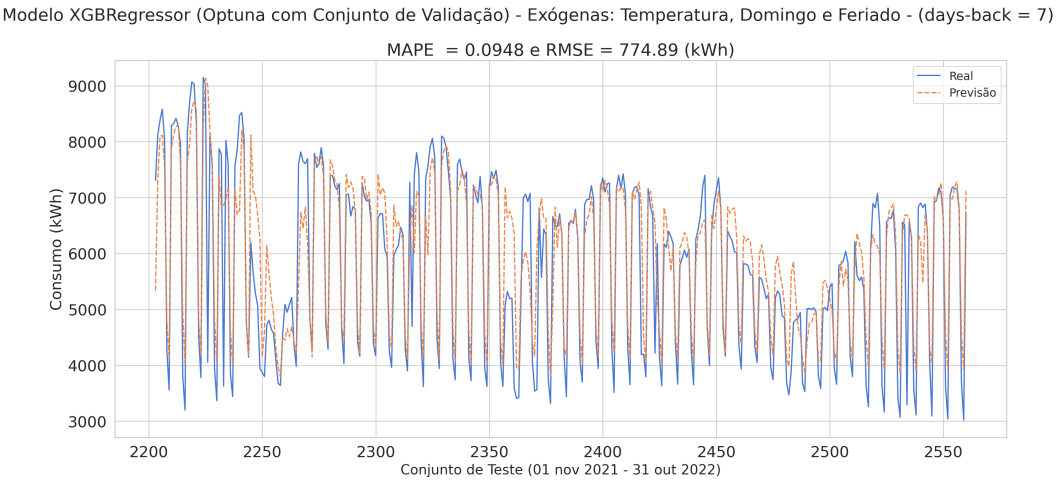


Figura 63: Valores reais *versus* previstos com o modelo XGBRegressor com menor MAPE.

Na Figura 64, o *plot* com as curvas de previsão e dados reais do modelo KNNRegressor com o menor valor de MAPE considerando a variável exógena *domingo* com uma janela de tempo igual a 7, dados de treino normalizados e k_vizinhos igual a 7.

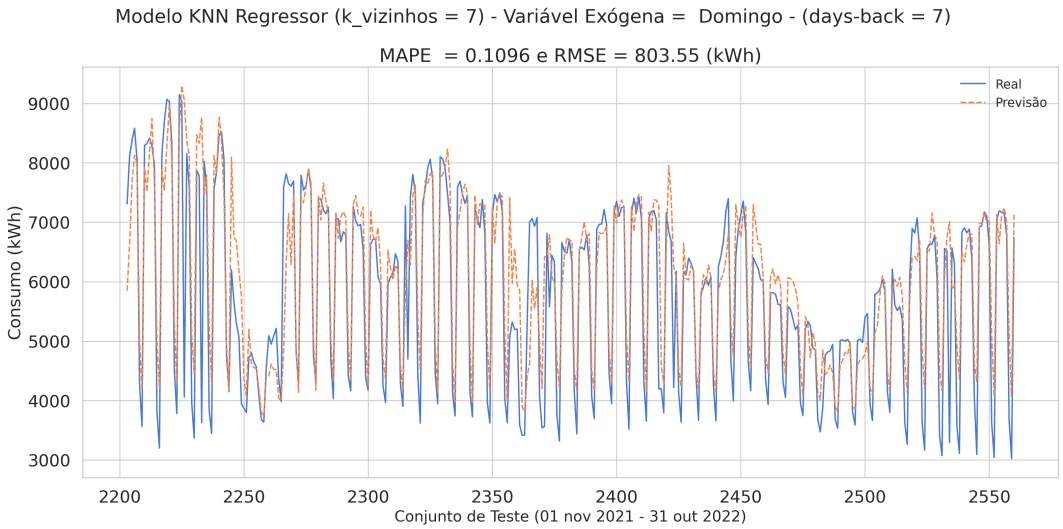


Figura 64: Valores reais *versus* previstos com o modelo KNNRegressor com menor valor de MAPE.

As experiências efetuadas com o KNNRegressor mostraram a necessidade de normalizar os dados para o KNN capturar adequadamente o comportamento das variáveis exógenas e que este algoritmo tem melhor desempenho quanto menor for o número de vizinhos.

De forma diferente aos demais modelos, percebe-se nos *plots* destes em particular, que os mesmos conseguem melhor aprendizagem quanto a alteração ocorrida no padrão de comportamento dos dados referente a segunda quinzena do mês dezembro do ano de 2021 e a primeira quinzena do mês de janeiro de 2022.

Contudo, os *plots* também revelam que as predições ficam mais desencontradas naqueles pontos referentes aos máximos e mínimos do consumo de energia, sendo o KNNRegressor mais afetado pelos pontos de mínimos do que o XGBRegressor.

Destaca-se ainda, que o valor do MAPE = 9.48% do modelo XGBRegressor é menor em aproximadamente 13.50% face ao valor do MAPE = 10.96% do modelo KNNRegressor. Por outro lado, o valor do RMSE = 774.89 kWh do modelo XGBRegressor é menor em aproximadamente 3.56% face ao valor do RMSE = 803.55 kWh do modelo KNNRegressor.

Ao concluir esta secção menciona-se que a presença da variável *domingo* tem sido transversal em todos os melhores modelos desenvolvidos até então neste projeto, e destaca-se que o XGBRegressor é um modelo de aprendizagem computacional mais complexo que o KNNRegressor, porém tendo em comum que seus melhores modelos tem o *days back* igual a 7 dias.

Entretanto, o XGBRegressor conseguiu capturar relações mais complexas entre as variáveis e, consequentemente, ter um melhor desempenho já que pôde encontrar a

melhor combinação de parâmetros para o modelo, nomeadamente, tamanho de janela e variáveis exógenas.

Quanto à variável *consumo*, o XGBRegressor teve dificuldade na aprendizagem após a aplicação da normalização `min_max` aos dados. Ao usar dados não normalizados, a escala original das variáveis é preservada. Isso pode ajudar o modelo a identificar corretamente o relacionamento nos dados para a previsão, e ao manter os dados em sua escala original, o modelo pôde capturar melhor essas relações.[6]

Uma vez que o XGBRegressor é sensível à escala das variáveis de entrada, quando os dados são normalizados, a escala é reduzida para um intervalo entre 0 e 1 e pode ter dificultado a aprendizagem do modelo, e talvez explique o aumento das métricas MAPE e RMSE.

6.3 MODELAÇÃO COM REDES NEURONAIS

Esta secção trata da utilização de redes neuronais para construção de modelos utilizando MLP, LSTM e GRU, tendo como objetivo a previsão do consumo de energia para o dia seguinte. Os modelos são construídos considerando as seguintes características:

- Utilização da plataforma *tensorflow_keras*[7] [84];
- Aos dados é aplicada a normalização `min_max`;
- No conjunto de teste foram considerados os últimos 365 dias. Os primeiros 1792 dias como conjunto de treino e, para validação, utilizaram-se os 404 dias intermédios;
- Número de dias de histórico a considerar (variável *days back*) utilizado para prever o consumo de energia para o dia seguinte: 7 e 14 dias;
- Número de camadas escondidas utilizadas nas redes MLP: 2;
- Número de células LSTM e GRU: 1;
- Nas redes MLP não foi utilizado *dropout*. Foram realizadas experiências com e sem *dropout* nas redes LSTM e GRU. Nestas redes, quando foi utilizado *dropout*, este foi aplicado aos valores de entrada e às células LSTM/GRU;

6 https://datascience.stackexchange.com/questions/60950/is-it-necessary-to-normalize-data-for-xgboost
7 https://colab.research.google.com/notebooks/gpu.ipynb

- Função de ativação[8] utilizada nas camadas escondidas das redes MLP e nas células LSTM e GRU: relu[9];
- Número de neurónios na camada de saída em todas as redes: 1;
- Função de ativação utilizada na camada de saída de todas as redes: função linear[10];
- Função de perda[11]: MSE;
- Otimizador[12]: rmsprop[13];
- Métricas: MAPE e RMSE;
- Épocas[14]: 200;
- *Patience*[15]: 20.
- Utilização da *framework* Optuna para otimização do número de neurónios por camadas, do *batch size*[16] e do *dropout*.[17]

Considerando a combinação entre o tipo de rede neuronal, MLP, LSTM ou GRU a ser utilizada no modelo, a variável *consumo*, as variáveis exógenas: *temperatura*, *domingo* e *feriado*, os dois valores de *days back* e o uso da técnica de regularização *dropout*, foram desenvolvidas diversas experiências totalizando 40 modelos. Após a otimização[18] do número de neurónios, *batch size* e *dropout* através do Optuna, foram geradas 600 experiências com cada combinação de hiperparâmetros, considerando a variável *consumo* e variáveis exógenas e *days back* igual a 7 e 14 para obter os valores das métricas MAPE e RMSE.

8 A função de ativação tem por objetivo limitar a saída de um neurónio em um intervalo de valores. É chamada de função de ativação porque governa o limite em qual o neurônio é ativado e a intensidade do sinal de saída [39].
9 A função de ativação linear retificada ou ReLU, para abreviar, é uma função linear por partes que produzirá a entrada diretamente se for positiva, caso contrário, produzirá zero. [85].
10 A função linear atua como uma transformação final sobre o valor calculado na etapa anterior (última camada escondida da rede). Essa operação gera um único valor que representa a saída da rede [86].
11 Função de perda se refere a como a rede será capaz de medir seu desempenho nos dados de treino e, portanto, como será capaz de se orientar na direção certa [86].
12 Otimizador é mecanismo através do qual a rede se atualizará, com base nos dados, para ajustar os pesos e os biases (ou viéses) das conexões entre os neurónios durante o treinamento da rede de modo a minimizar a função de perda.
13 O RMSProp é um otimizador estável para treinar redes neuronais [87].
14 Épocas se refere ao número de vezes que os dados de treino são apresentados à rede neuronal [86]
15 *Patience* se refere à capacidade da rede neuronal continuar aprendendo e melhorando seu desempenho mesmo quando o progresso parece ter estagnado. O treino é parado após a quantidade de épocas ter sido alcançada sem que haja uma melhoria no desempenho da rede [45].
16 Batch size define o número de amostras (o tamanho do lote) que serão processados pela rede neuronal ao mesmo tempo [88].
17 *Dropout* se refere a uma técnica de regularização utilizada durante o processo de treino, onde aleatoriamente se desliga um certo número de neurónios em uma camada da rede neuronal, juntamente com todas as suas conexões de entrada e saída [14].
18 No Anexo A.3 se encontra um exemplo de *script* em Python sobre a otimização com a *framework* Optuna.

6.3.1 *Modelos de Redes Neuronais com a Variável Consumo*

A Tabela 31 lista, **por ordem crescente** dos valores da métrica MAPE, os modelos com redes neuronais utilizando a variável *consumo*.

Quando aplicável, na coluna "*Dropout*" são discriminadas as probabilidades de *dropout* utilizadas em cada camada da seguinte forma: probabilidade na camada 1/probabilidade na camada 2. De forma semelhante, na coluna "Neurónios" é discriminada a quantidade de neurónios nas camadas escondidas. Os resultados da Tabela 31 mostram que os modelos com MLP, tanto com *days back* igual a 7 e 14, têm os melhores desempenhos entre os demais com redes neuronais, relativamente a variável *consumo*. Destaca-se ainda, que os modelos que não utilizam *dropout* têm menor valor de MAPE e de RMSE.

Tabela 31: MAPE e RMSE dos modelos com redes neuronais (variável *consumo*).

Modelo	**Dropout**	**Days_Back**	**MAPE**	**RMSE**(kWh)	**Batch Size**	**Neurónios**
MLP	não	7	0.0770	699.13	54	306 / 6
MLP	não	14	0.0774	711.65	49	300 / 16
GRU	não	14	0.0816	697.72	76	173
LSTM	não	14	0.0814	724.41	159	36
LSTM	não	7	0.0885	769.61	64	187
LSTM	0.32 / 0.57	14	0.0892	752.60	12	228
GRU	0.28 / 0.34	14	0.0904	724.19	24	321
LSTM	0.45 / 0.40	7	0.0942	779.88	20	272
GRU	não	7	0.1047	790.52	96	277
GRU	0.20 / 0.27	7	0.1052	775.12	29	240

Como complemento para análise comparativa, encontram-se na Figura 65 os *boxplots* com os valores de MAPE dos modelos[19] de redes neuronais com a variável *consumo*, referidos na Tabela 31, para auxiliar a análise visual.

19 Na Figura 65 e seguintes, quando aplicável, adotou-se a seguinte convenção: dp = *dropout*; 7d se refere a *days back* = 7; e 14d se refere a *days back* = 14.

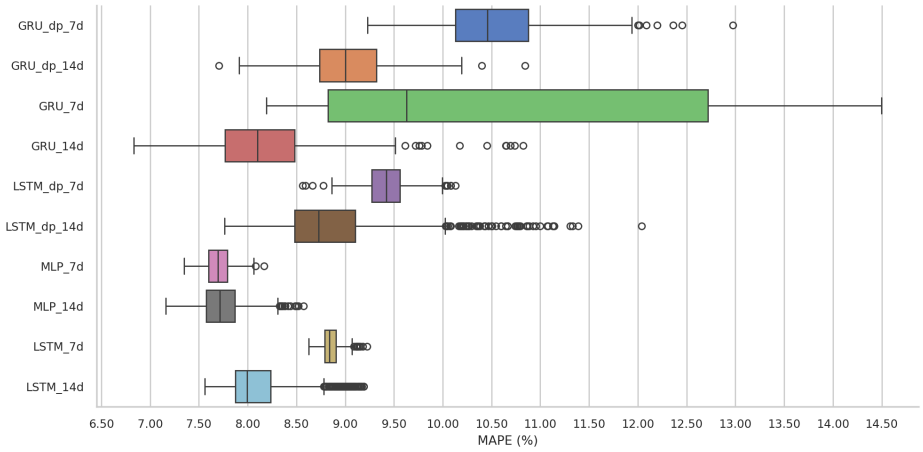


Figura 65: *Boxplot* do MAPE de 600 experiências com redes neuronais (variável *consumo*).

A visualização dos *boxplots* do MAPE dos modelos com a variável *consumo*, revela de imediato que o modelo MLP com *days back* igual a 7 é de facto aquele que tem o melhor desempenho, destacando-se ainda, ter somente poucas observações atípicas e pequena dispersão de erros.

Em função das características observadas nos *boxplots* acima, decidiu-se verificar ainda, a evolução de valores de σ nos modelos GRU, com e sem *dropout* e *days back* igual a 7, e o modelo LSTM com *dropout* e *days back* igual a 14. A Figura 66 mostra o gráfico com essa informação.

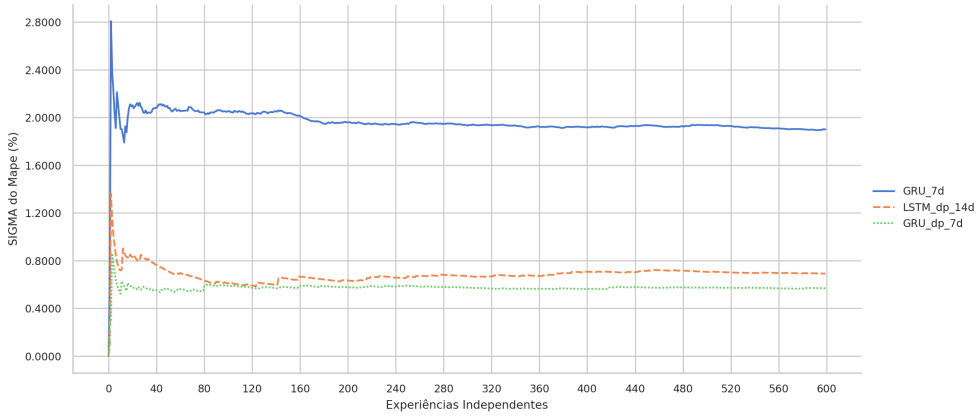


Figura 66: Evolução do σ do MAPE dos Modelos 'GRU_7d', 'LSTM_dp_14d' e 'GRU_dp_7d' (variável *consumo*).

Percebe-se no gráfico que os valores iniciais do σ de cada um destes modelos, têm oscilações nas 20 primeiras experiências e que pela ducentésima experiência já encontrou a estabilidade. Algo parecido a esse padrão de estabilidade também foi observado nos demais modelos de redes neuronais desenvolvidos neste trabalho.

6.3.2 *Modelos de Redes Neuronais com as Variáveis Exógenas*

Os resultados das experiências de redes neuronais utilizando as variáveis exógenas se encontram a seguir em tabelas separadas por rede neuronal.

Nas tabelas 32, 33 e 34, quando aplicável, na coluna "*Dropout*" são discriminadas as probabilidades de *dropout* utilizadas em cada camada da seguinte forma: probabilidade na camada 1/probabilidade na camada 2. De forma semelhante, na coluna "Neurónios" é discriminada a quantidade de neurónios nas camadas escondidas.

Modelos com MLP

A Tabela 32, mostra os valores das métricas MAPE e RMSE dos modelos MLP desenvolvidos com redes neuronais com variáveis exógenas.

Tabela 32: MAPE e RMSE dos modelos com MLP (variáveis exógenas)

Modelo	**Dropout**	**Days_Back**	**MAPE**	**RMSE**(kWh)	**Batch Size**	**Neurónios**	**Variável**
MLP	não	7	0.0686	643.91	13	154 / 1	Domingo
MLP	não	7	0.0725	654.72	58	258 / 3	Dom. e Fer.
MLP	não	14	0.0726	678.35	60	238 / 1	Domingo
MLP	não	14	0.0773	662.11	35	231 / 1	Dom. e Fer.
MLP	não	14	0.0797	704.34	42	349 / 1	Temperatura
MLP	não	7	0.0806	726.27	66	220 / 4	Temperatura

Observa-se na Tabela 32 que o modelo MLP, com o MAPE igual 6.86% e RMSE de 643.91 kWh, tem o melhor desempenho, quando:

- *Days back* = 7;
- *Batch size* = 13;
- se utilizam 154 neurónios na 1ª camada e 1 neurónio na 2ª camada;
- se utiliza a variável exógena *domingo*;
- não foi considerado *dropout* para MLP.

Assim, encontram-se na Figura 67 os *plots* com a variabilidade do MAPE e da evolução do σ do MAPE do modelo MLP, acima referido, com a variável *domingo* e *days back* = 7.

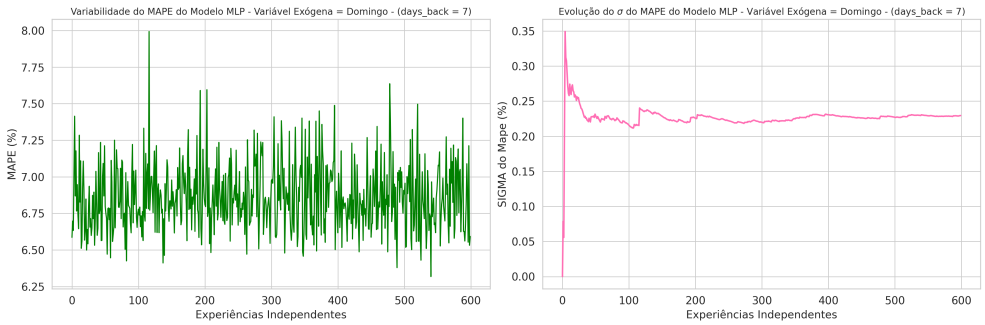


Figura 67: Variabilidade do MAPE e Evolução do σ do MAPE do modelo MLP.

As métricas acima relatadas e os *plots* aqui visualizados, confirmam que o modelo de previsão MLP está bem ajustado aos dados e é consistente.

Por último, na Figura 68 encontram-se os *boxplots* com os valores do MAPE dos modelos desenvolvidas com MLP usando variáveis exógenas.

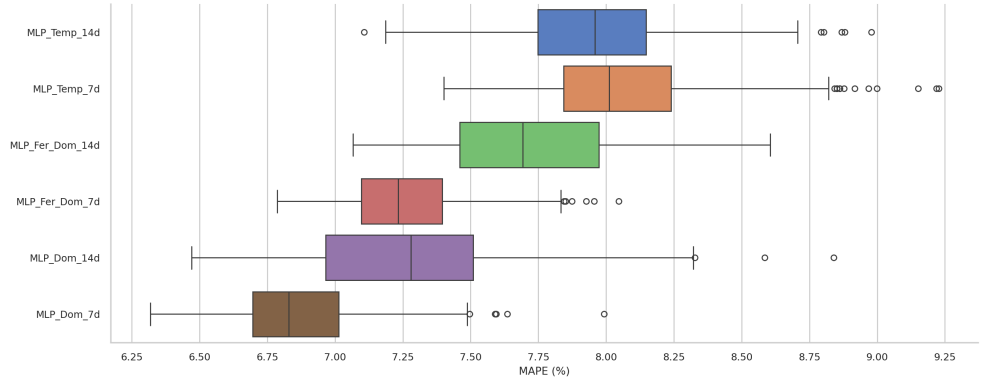


Figura 68: *Boxplot* do MAPE dos modelos com MLP (variáveis exógenas).

Pode-se visualmente confirmar nos *boxplots* acima que o modelo MLP usando a variável exógena *domingo* e com *days back* igual a 7 é, de longe o modelo com melhor desempenho, ainda que com uma quantidade menor de neurónios e menor valor de *batch size*.

Modelos com LSTM

A Tabela 33, mostra, por ordem crescente dos valores da métrica MAPE, os modelos LSTM desenvolvidos com redes neuronais com variáveis exógenas.

Tabela 33: MAPE e RMSE dos modelos com LSTM (variáveis exógenas)

.

Modelo	Dropout	Days_Back	MAPE	RMSE(kWh)	Batch Size	Neurónios	Variável
LSTM	não	14	0.0741	696.78	26	186	Domingo
LSTM	não	7	0.0751	669.46	64	221	Dom. e Fer.
LSTM	não	14	0.0755	662.44	19	144	Dom. e Fer.
LSTM	0.12 / 0.26	14	0.0760	667.72	28	210	Dom. e Fer.
LSTM	não	7	0.0770	697.47	37	168	Domingo
LSTM	0.14 / 0.39	7	0.0772	670.62	13	189	Dom. e Fer.
LSTM	0.27 / 0.38	7	0.0790	708.51	48	242	Domingo
LSTM	0.32 / 0.35	14	0.0798	705.13	49	181	Domingo
LSTM	não	14	0.0988	778.85	38	300	Temperatura
LSTM	0.25 / 0.20	14	0.1027	783.50	25	218	Temperatura
LSTM	0.10 / 0.51	7	0.1141	842.13	15	165	Temperatura
LSTM	não	7	0.1155	872.66	38	241	Temperatura

A Tabela 33 nos revela que o modelo LSTM com o MAPE igual a 7.41% e RMSE igual a 696.78 kWh, tem o melhor desempenho[20] nesse tipo de rede neuronal, possuindo as seguintes características:

- Sem *dropout*;
- *Days back* igual a 14;
- *Batch size* = 26;
- Uma camada com 186 neurónios;
- Variável exógena *domingo*.

Referente aos modelos LSTM, pode ver-se na Figura 69 os *boxplots* com os valores do MAPE dos modelos desenvolvidas com LSTM usando variáveis exógenas.

20 Decidiu-se utilizar o rigor percentual na escolha pelo melhor desempenho dos modelos LSTM.

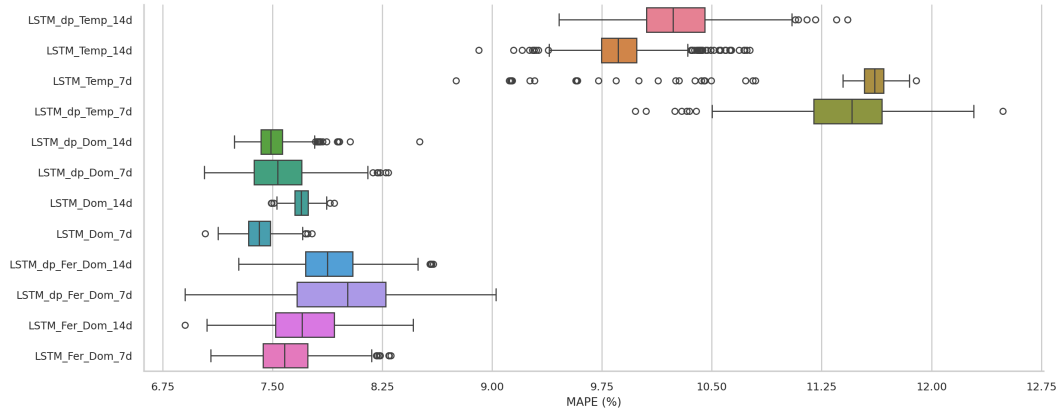


Figura 69: *Boxplot* do MAPE dos modelos com LSTM (variáveis exógenas).

Destaca-se nos *boxplots* acima a pequena dispersão de erros do modelo com melhor desempenho na Tabela 33 o modelo LSTM_Dom_14.

Modelos com GRU

A Tabela 34, mostra os valores das métricas MAPE e RMSE dos modelos GRU desenvolvidos com redes neuronais com variáveis exógenas.

Tabela 34: MAPE e RMSE dos modelos com GRU (variáveis exógenas)

.

Modelo	**Dropout**	**Days_Back**	**MAPE**	**RMSE**(kWh)	**Batch Size**	**Neurónios**	**Variável**
GRU	não	14	0.0772	671.49	70	294	Dom. e Fer
GRU	não	14	0.0773	646.79	46	299	Domingo
GRU	não	7	0.0815	678.28	58	246	Dom. e Fer
GRU	não	7	0.0834	693.26	54	265	Domingo
GRU	0.16 / 0.085	7	0.0835	707.26	37	295	Temperatura
GRU	0.02 / 0.12	14	0.0848	685.46	18	217	Domingo
GRU	não	7	0.0855	724.53	44	324	Temperatura
GRU	0.42 / 0.10	7	0.0893	704.17	20	268	Dom. e Fer
GRU	0.16 / 0.096	14	0.0899	691.55	23	169	Dom. e Fer
GRU	0.22 / 0.24	7	0.0902	727.48	20	209	Domingo
GRU	não	14	0.0907	728.83	42	258	Temperatura
GRU	0.28 / 0.065	14	0.0908	719.24	273	21	Temperatura

A Tabela 34, mostra o modelo GRU que possui um melhor conjunto de métricas, com o seu MAPE igual a 7.73% e RMSE de 646.79 kWh, tem as seguintes características:

- Sem *dropout*;
- *Days back* igual a 14,;
- *Batch_size* = 46;
- Uma camada com 299 neurónios;
- Variável exógena *domingo*.

Encontram-se na Figura 70 os *boxplots* com os valores do MAPE dos modelos desenvolvidos com GRU usando variáveis exógenas.

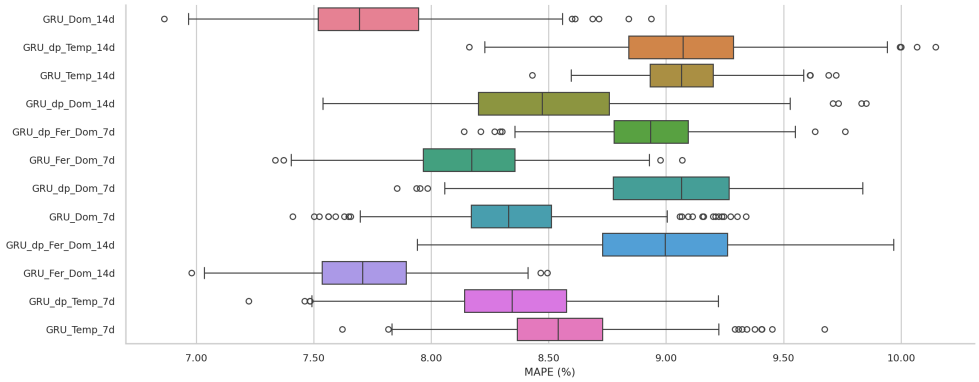


Figura 70: *Boxplot* do MAPE dos modelos com GRU (variáveis exógenas).

6.3.3 *Comparação LSTM e GRU*

Efetou-se uma comparação entre dois modelos com redes neuronais que foram originariamente concebidos para trabalhar com previsão de séries temporais. Assim, encontra-se na Figura 71, o *plot*[21] com as curvas de previsão e dados reais do modelo usando LSTM, cujo MAPE e RMSE são, respetivamente, igual a 7.41% e 696.78 kWh.

21 O dia zero no gráfico corresponde ao dia 01 de novembro de 2021, que se refere a primeira observação do conjunto de teste.

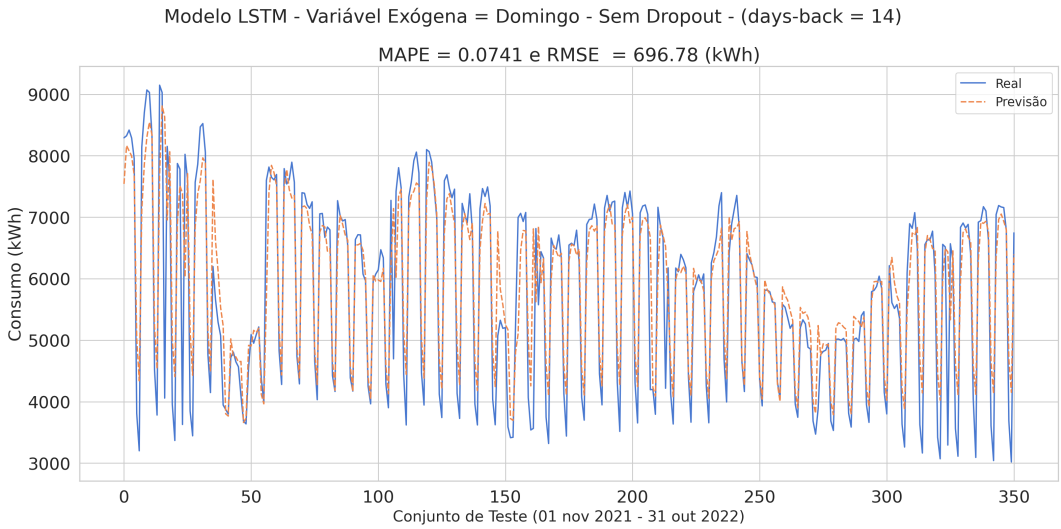


Figura 71: Valores reais *versus* previstos com o modelo LSTM com menor MAPE.

Na Figura 72, o *plot* com as curvas de previsão e dados reais do modelo GRU com o menor valor de MAPE considerando a variável exógena *domingo* e days back igual a 14.

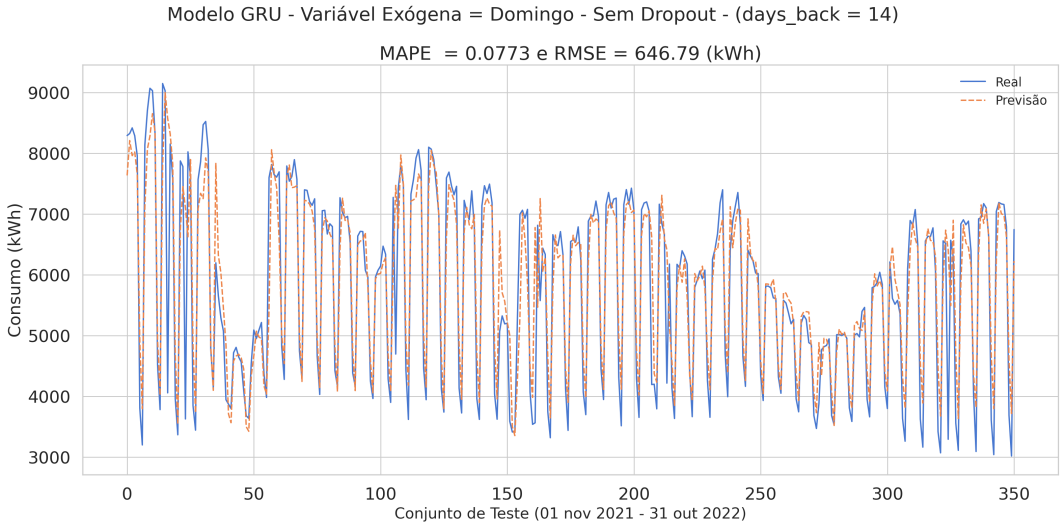


Figura 72: Valores reais *versus* previstos com o modelo GRU com menor MAPE.

Nos *plots* acima, percebe-se que o modelo LSTM consegue melhor aprendizagem quanto a alteração ocorrida no padrão de comportamento dos dados referente a segunda quinzena do mês dezembro do ano de 2021 e a primeira quinzena do mês de janeiro de 2022. O mesmo já não ocorre com o modelo GRU. Entretanto, os *plots* mostram também que há, em ambos os modelos, alguma dificuldade de aprendizagem quando há pontos associados a máximos e mínimos no consumo de energia.

Ao concluir este capítulo cabe realçar que o modelo MLP usando a variável exógena *domingo* com o MAPE igual 6.86% e RMSE de 643.91 kWh, tem o melhor desempenho entre os modelos deste capítulo, e os melhores modelos com as redes neuronais LSTM e GRU não utilizam *dropout*, e têm em comum a variável *domingo* e o *days back* igual a 14, e com 7.41% e 7.73% de MAPE, respetivamente.

7

DISCUSSÃO DE RESULTADOS

O propósito deste capítulo é apresentar uma comparação dos melhores modelos desenvolvidos neste trabalho, e particularmente o comportamento de modelo estatístico SARIMAX com o *Multilayer Perceptron*. Também se discute a influência das variáveis exógenas e da variável *days back*.

7.1 DISCUSSÃO E COMPARAÇÃO DE RESULTADOS

A Tabela 35 sintetiza os melhores resultados obtidos com cada tipo de modelo, sendo visível que as redes neuronais foram os únicos modelos que obtiveram valores da métrica MAPE abaixo dos 8.0% e que as redes neuronais MLP foram as únicas a obter um valor abaixo dos 7.0%.

Tabela 35: Seleção dos modelos com o melhor conjunto de métricas.

Modelo	**MAPE**	**RMSE**(kWh)	***Days_Back***	**Variável**
MLP	0.0686	643.91	7	Domingo
LSTM	0.0741	696.78	14	Domingo
GRU	0.0773	646.79	14	Domingo
SARIMAX	0.0838	656.35	7	Domingo, Temp. e Feriado
XGBRegressor	0.0948	774.89	7	Domingo, Temp. e Feriado
KNNRegressor	0.1096	803.55	7	Domingo

O modelo MLP, com *days back* igual a 7 e usando a variável exógena *domingo*, tem um MAPE = 6.86% e o RMSE = 643.91 kWh, sendo este o modelo que tem o melhor desempenho entre todos deste projeto, e o único com um valor de MAPE inferior a 7.0%, embora não originalmente projetado para trabalhar com séries temporais.

Também se percebe na Tabela 35, que a presença da variável exógena *domingo* é transversal aos modelos com melhor desempenho, assim como uma janela de tempo com tamanho igual a 7 está presente em grande parte deles.

7.2 ANÁLISE COMPARATIVA

Decidiu-se efetuar uma comparação do melhor desempenho entre os modelos de redes neuronais com os modelos estatísticos. Assim, encontra-se na Figura 73, o *plot*[1] com as curvas de previsão e dados reais do modelo usando MLP.

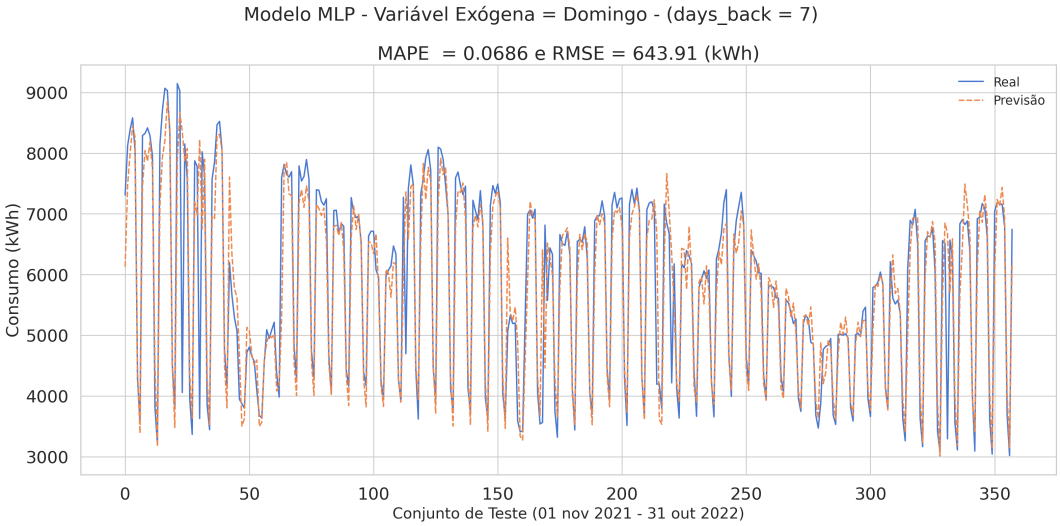


Figura 73: Valores reais *versus* previstos com o modelo MLP com menor MAPE.

A Figura 74 contém o *plot* com as curvas de previsão e dados reais do modelo SARIMAX(1, 1, 3)(1, 1, 3)[7], após uso da Validação Cruzada *Rolling Window* e frequência sazonal igual a 7, listado na Tabela 35.

1 O dia zero no gráfico se refere ao dia 01 de novembro de 2021, que corresponde a primeira observação do conjunto de teste.

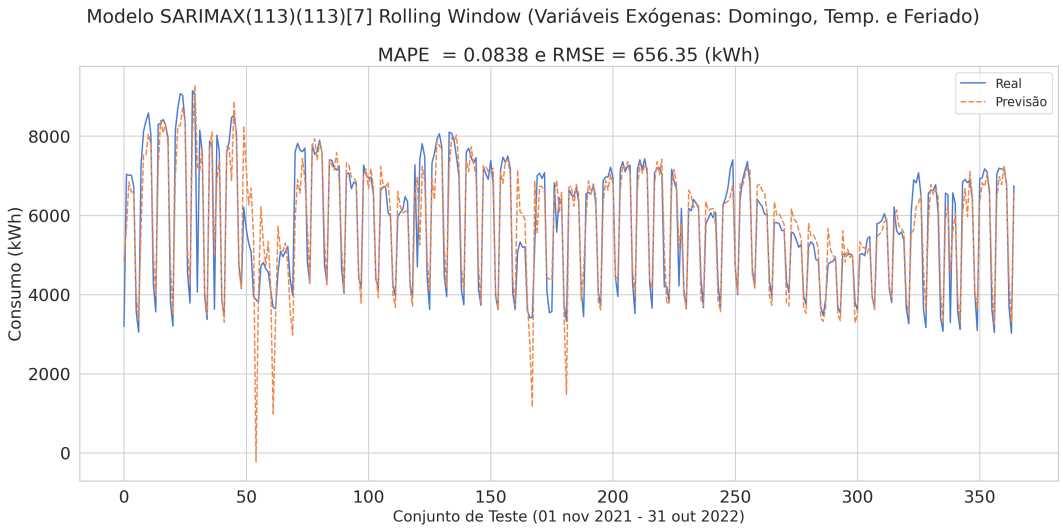


Figura 74: Valores reais *versus* previstos do modelo SARIMAX(1, 1, 3)(1, 1, 3)[7] com RW.

Numa visualização dos *plots* acima, percebe-se que o modelo MLP consegue melhor aprendizagem do que o SARIMAX quanto a alteração ocorrida no padrão de comportamento dos dados referente a segunda quinzena do mês dezembro do ano de 2021 e a primeira quinzena do mês de janeiro de 2022. Contudo, os *plots* mostram também que as previsões ficam mais desencontradas nos pontos associados a máximos e mínimos no consumo de energia, em ambos os modelos. Há ainda a destacar que o valor do MAPE do modelo MLP é menor, ao redor de 18.14% ao valor do MAPE do modelo SARIMAX. Por outro lado, o valor do RMSE é menor ao redor de 1.89%.

7.3 INFLUÊNCIA DAS VARIÁVEIS EXÓGENAS

Nesta secção se examina a influência que as variáveis exógenas: *domingo*, *temperatura* e *feriado*, produziu nos desempenhos dos modelos desenvolvidos neste trabalho.

Domingo

A opção pelo domingo deu-se porque os edifícios do *Campus 2* estão encerrados nesse dia da semana, não havendo qualquer atividade, e buscou-se identificar o comportamento do consumo nesse dia para avaliar a performance dos modelos.

A variável *domingo* possui uma correlação de -0.45 com a variável *consumo*, indicando uma associação moderada negativa. Percebe-se que o consumo, sendo menor aos domingos em comparação com outros dias da semana, provocou melhor aprendizagem nos modelos e como consequência melhor desempenho, mostrando assim que a sua inclusão se mostrou

correta, e pode-se afirmar que há transversalidade da mesma nos modelos com melhores desempenhos, sendo confirmado na modelação que a presença de *feriado* ou a *temperatura* tem menos impacto direto na previsão do consumo de energia em comparação com o *domingo*.

Temperatura

Sobre a inclusão da temperatura como variável exógena, sabe-se intuitivamente que existe uma relação entre procura de eletricidade e condições meteorológicas, em particular da temperatura, podendo este aspeto ser também considerado na previsão de consumo de energia no curto prazo, e buscou-se verificar este aspeto neste trabalho.

Contudo, é necessário adicionar que, no âmbito deste estudo, os modelos que utilizam *somente* a variável *temperatura* como variável exógena obtêm piores resultados do que aqueles que não a utilizam, embora, inicialmente se esperasse que a temperatura tivesse maior impacto na previsão do consumo de energia em um *campus* universitário.

Possível explicação para esse facto pode ser as diferentes sazonalidades da série temporal do consumo *versus* a sazonalidade da série temporal da temperatura. Entre as 00h00 e as 06h00, em função de não haver atividades no *campus*, não há consumo, embora a temperatura possa estar próxima do ponto mínimo e os modelos, trabalhando com granularidade diária, não conseguem capturar um padrão de relacionamento entre as variáveis naquele específico intervalo temporal. Adiciona-se a isso, o facto de que a climatização no *Campus* 2 do IPLeiria ser feita sobretudo com recurso a gás natural.

Feriado

Embora a correlação entre as variáveis *feriado* e *consumo* seja baixa, ao redor de -0.19, sua escolha como variável exógena se deu por dois motivos: *i*) verificar o desempenho dos modelos na presença dessa variável; e, *ii*) auxiliar a análise da evolução do consumo ao longo da semana, diferenciando se determinado dia é um feriado ou não, buscando identificar padrões e variações nos dados de consumo em diferentes dias da semana e durante feriados.

Entretanto, a pouca influência da variável *feriado* pode ter ocorrido por esta ter somente 89 observações no intervalo de dias analisado, uma vez que os modelos que utilizaram somente a variável *feriado* não obtiveram bons desempenhos.

Para concluir esta secção, destaca-se novamente na Tabela 35 que os modelos com melhor desempenho têm a presença de variáveis exógenas, e na modelação com aprendizagem computacional, ocorreu melhor desempenho ao usar as variáveis exógenas, destacando-se

a presença da variável *domingo* naqueles com melhor desempenho com redes neuronais. Na modelação estatística, com variáveis exógenas, em que se utilizou técnicas de validação cruzada, houve um aumento significativo de desempenho, com grande redução do MAPE em alguns modelos.

7.4 INFLUÊNCIA DA VARIÁVEL *DAYS BACK*

Esta secção inclui a análise da influência que o tamanho - 7 ou 14 dias - da janela de tempo ou *days back* ou *lags*, proporcionou no desempenho dos modelos desenvolvidos neste trabalho.

A escolha do tamanho igual a 7 para usar em *days back* se dá após análise da sazonalidade da série temporal do consumo, verificada no gráfico de autocorrelação na Figura 75, onde se visualizam autocorrelações em múltiplos de 7.

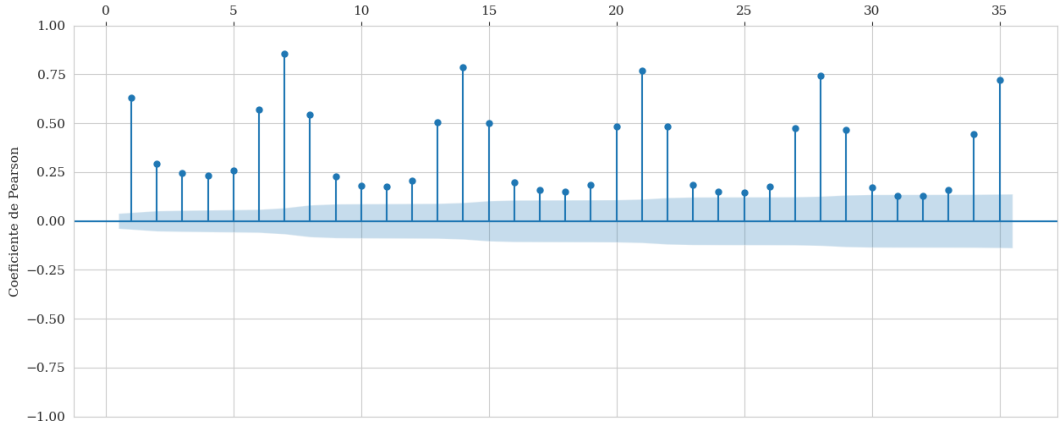


Figura 75: Função Autocorrelação da série temporal do consumo.

Além das autocorrelações identificadas no correlograma, e uma vez que os dados têm granularidade diária com o objetivo de se fazer previsão de consumo para o dia seguinte, optou-se por trabalhar com *days back* igual a 7 e 14.

O tamanho igual a 7 para *days back* está presente no modelo MLP, o de melhor desempenho neste trabalho. Para os modelos estatísticos foi estabelecido uma frequência ***m*** de sazonalidade igual a 7, e isto se mostrou adequado uma vez que o SARIMAX com melhor desempenho só foi superado por aqueles de redes neuronais, como se vê na Tabela 35 no início deste capítulo.

Este capítulo analisou e comparou os resultados da modelação com aprendizagem computacional e modelação estatística, ficando evidenciado que:

i) o modelo com melhor desempenho neste projeto foi desenvolvido com um *Multilayer Perceptron* utilizando *days back* igual a 7, obtendo um MAPE abaixo de 7.0%;

ii) os resultados obtidos com a MLP superam os resultados dos modelos com LSTM e GRU, apesar destes serem originalmente projetados para trabalhar com sequência de termos, enquanto aquela não;

iii) os modelos com as redes neuronais obtiveram melhor desempenho do que os modelos estatísticos;

iv) a variável *domingo* tem maior influência na previsão do consumo do que a variável *temperatura*.

8

CONCLUSÃO E TRABALHO FUTURO

A finalidade principal deste projeto, foi o desenvolvimento de modelos estatísticos e de aprendizagem computacional para a previsão para o dia seguinte de procura de energia elétrica num *campus* universitário, com a posterior comparação dos modelos desenvolvidos através do MAPE e do RMSE para avaliação do desempenho dos mesmos.

Se fez uso de dados brutos de consumo de energia elétrica do *Campus 2* do IPLeiria entre 27 de outubro de 2015 e 4 de novembro de 2022, coletados a cada 15 minutos no contador da E-Redes e também registos históricos da temperatura, obtidos junto do Instituto Português do Mar e da Atmosfera (IPMA).

Importante destacar, na fase de preprocessamento dos dados, a identificação de dados duplicados ou iguais a zero, provocados pela mudança de hora de verão, e o seu tratamento para que os dados estivessem corretos e prontos para a etapa de integração.

A metodologia adotada envolveu desenvolver modelos estatísticos e modelos baseados nos algoritmos KNN, XGBoost e de Redes Neuronais - MLP, LSTM e GRU. Para os modelos estatísticos fez-se uso da função *auto_arima*() [23] para modelos SARIMA e da função *sarimax*() [24] tanto para modelos SARIMA quanto modelos SARIMAX, para seleção automática dos parâmetros desses modelos. Para os modelos com XGBoost e Redes Neuronais - MLP, LSTM e GRU, utilizou-se a *framework* Optuna [52] para otimização de hiperparâmetros.

Na etapa de modelação foram construídos modelos de previsão de consumo para o dia seguinte com base em janelas temporais dos 7 e 14 dias anteriores.

Estes modelos poderão ser treinados e aplicados em outros *campi* universitários ou outros tipos de organizações com condições semelhantes, já que os resultados obtidos demonstraram a importância do uso de IA para previsão do consumo de energia.

8.1 LIMITAÇÕES

Esse projeto apresentou uma metodologia, através de um conjunto de algoritmos que foram escolhidos para desenvolver os modelos. Como foi demonstrado ao longo deste

relatório, o desempenho obtido, a partir dos mais variados métodos de análise aplicados, garantem a plausibilidade dos modelos desenvolvidos. Entretanto, sempre haverá incertezas em previsão de consumo de energia em edifícios, já que incluem os aspetos humanos, climáticos e de construção [1].

Há fatores relacionados com o consumo de energia que não podem ser previstos com precisão, já que a frequência de utilização de aparelhos de aquecimento de edifícios em *campi* universitários é decidida pela condição física, hábitos e ocupações dos utilizadores. Já os aspetos de construção têm o maior impacto sobre a eficiência energética geral do edifício. Eles incluem o tipo de construção, o uso de sistemas de aquecimento, ventilação e ar condicionado, assim como os materiais utilizados na construção, tipos de parede, tipos de janelas. Quanto aos aspetos climáticos pode-se dizer que devido à natureza mutável do clima, não pode ser previsto com precisão e portanto, causa incertezas em torno do uso de energia em edifícios [89].

8.2 TRABALHO FUTURO

No futuro existem aspetos que poderão ser alvo de desenvolvimento. Em particular, pretende-se aprofundar este estudo em modelos capazes de prever o consumo de energia elétrica, gás ou água de um determinado dia da semana com base em $\boldsymbol{n}$ dias da semana semelhantes anteriores (*similar days*). Por exemplo, prever o consumo da próxima 3ª feira com base no consumo das últimas $\boldsymbol{n}$ 3ªs feiras, e também a previsão de consumo em situações particulares, como feriados, já que o desenvolvimento de uma modelação de previsão para um intervalo específico pode ser útil como complementação de estratégias de auto-consumo.

Outra proposta poderá passar por considerar ainda uma janela de tempo (*days back*) de 8 dias, já que resultados de algumas experiências exploratórias realizadas ao longo do projeto, que envolveram otimização com o Optuna, demonstraram que esse tamanho de janela poderia também ser adequado às previsões.

Um trabalho futuro nesta área de investigação poderá ainda, incluir registos históricos de consumo de cada edifício do *campus* para treinar modelos de previsão de consumo especificamente para um determinado edifício, subdividindo a previsão em consumo desagregado por equipamentos, aquecimento, climatização e iluminação.

Monitorar e analisar o padrão de consumo de energia elétrica, água e gás em *campi* universitários ajudará a identificar oportunidades para reduzir custos e dessa forma contribuir para a sustentabilidade ambiental. Essa monitorização e análise trarão maior

compreensão do desempenho operacional dos prédios em relação a fatores como idade da edificação, horário de funcionamento e o tipo de equipamento instalado, já que essas variáveis exercem influencia significativa no consumo energético de um *campus*. A inclusão do uso de modelos de previsão utilizando IA, poderá também viabilizar uma melhor gestão de procura de energia, gerando assim, economia de recursos.

BIBLIOGRAFIA

[1] Sigrid Reiter. *Energy Consumption: Impacts of Human Activity, Current and Future Challenges, Environmental and Socioeconomic Effects*. Nova Science Publishers, New York, United States, 2013.

[2] Roman V Klyuev et al. «Methods of Forecasting Electric Energy Consumption: A Literature Review». Em: *Energies* 15.23 (2022), p. 8919.

[3] Navid Shirzadi et al. «Medium-term regional electricity load forecasting through machine learning and deep learning». Em: *Designs* 5.2 (2021), p. 27.

[4] *Forecasting: Principles and Practice (3rd ed)*. Acesso: 04 de agosto de 2023. URL: https://otexts.com/fpp3/.

[5] Khuram Pervez Amber et al. «Energy consumption forecasting for university sector buildings». Em: *Energies* 10.10 (2017), p. 1579.

[6] Luis G Baca Ruiz et al. «A time-series clustering methodology for knowledge extraction in energy consumption data». Em: *Expert Systems with Applications* 160 (2020), p. 113731.

[7] MA Islam et al. «Energy demand forecasting». Em: *Energy for sustainable development*. Elsevier, 2020, pp. 105–123.

[8] Paul Newbold, William L. Carlson e Betty M. Thorne. *Statistics for business and economics*. Pearson, 2023.

[9] ND Lewis. «Deep Time Series Forecasting with Python». Em: *Create Space Independent Publishing Platform* (2016).

[10] *Produção Industrial Indiana*. Acesso: 08 de maio de 2023. URL: https://search.r-project.org/CRAN/refmans/seasonal/html/iip.html.

[11] Hannah Ritchie, Max Roser e Pablo Rosado. «Energy». Em: *Our World in Data* (2022). https://ourworldindata.org/energy.

[12] Aileen Nielsen. *Practical time series analysis: Prediction with statistics and machine learning*. O'Reilly Media, 2019.

[13] Svend Hylleberg. *Modelling seasonality*. Oxford University Press, 1992.

[14] Marco Peixeiro. *Time series forecasting in python*. Simon e Schuster, 2022.

[15] Sean J Taylor e Benjamin Letham. «Forecasting at scale. 2017». Em: *facebookincubator.github.io/prophet* (2017).

[16] *Fernando Sebastiao, "Séries Temporais e Previsão course notes" MSc on Data Science, ESTG-Polytechnic of Leiria, 2022.*

[17] Francois Chollet. *Deep learning with Python*. Simon e Schuster, 2021.

[18] Warren L Young. «The Box-Jenkins approach to time series analysis and forecasting: principles and applications». Em: *RAIRO-Operations Research- Recherche Opérationnelle* 11.2 (1977), pp. 129–143.

[19] *Identificando os Parâmetros de Modelo ARIMA*. Acesso: 11 de agostode 2023. URL: https://perma.cc/P9BK-764B.

[20] Zaiyong Tang, Chrys De Almeida e Paul A Fishwick. «Time series forecasting using neural networks vs. Box-Jenkins methodology». Em: *Simulation* 57.5 (1991), pp. 303–310.

[21] Rob J Hyndman e George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[22] Joos Korstanje. *Advanced Forecasting with Python: With State-of-the-Art-Models Including LSTMs, Facebook's Prophet, and Amazon's DeepAR*. Springer, 2021.

[23] *Auto ARIMA*. Acesso: 05 de abril de 2023. URL: https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.arima.auto_arima.html.

[24] *Função Sarimax()*. Acesso: 05 de março de 2023. URL: https://www.statsmodels.org/dev/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html.

[25] *Auto Arima Rob Hyndman*. Acesso: 11 de agosto de 2023. URL: https://otexts.com/fpp3/arima-r.html.

[26] Alexandre Sartoris. *Estatística e introdução à econometria*. Saraiva Educação SA, 2017.

[27] Allen Downey. *Think stats: Exploratory data analysis*. "O'Reilly Media, Inc.", 2014.

[28] Soliman Abdel-hady Soliman e Ahmad Mohammad Al-Kandari. *Electrical load forecasting: modeling and model construction*. Elsevier, 2010.

[29] Robert H Shumway, David S Stoffer e David S Stoffer. *Time series analysis and its applications*. Vol. 3. Springer, 2000.

[30] Luiz Paulo Fávero e Patrícia Belfiore. *Manual de análise de dados: estatística e modelagem multivariada com Excel®, SPSS® e Stata®*. Elsevier Brasil, 2017.

[31] *Shapiro Wilk*. Acesso: 05 de novembro de 2023. URL: https://www.statskingdom.com/doc_shapiro_wilk.html.

[32] *Jarque Bera Test*. Acesso: 23 de junho de 2023. URL: http://www.ece.northwestern.edu/local-apps/matlabhelp/toolbox/stats/jbtest.html.

[33] *ADF Test*. Acesso: 23 de junho de 2023. URL: https://www.statsmodels.org/dev/generated/statsmodels.tsa.stattools.adfuller.html.

[34] Andreas C Müller e Sarah Guido. *Introduction to machine learning with Python: a guide for data scientists*. "O'Reilly Media, Inc.", 2016.

[35] Jason Brownlee. *Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery, 2018.

[36] Aurélien Géron. «Hands-on machine learning with scikit-learn and tensorflow: Concepts». Em: *Tools, and Techniques to build intelligent systems* (2017).

[37] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[38] Andriy Burkov. *The hundred-page machine learning book*. Vol. 1. Andriy Burkov Quebec City, QC, Canada, 2019.

[39] Jason Brownlee. *Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras*. Machine Learning Mastery, 2016.

[40] Andrew J Ashwood. *Portfolio selection using artificial intelligence*. 2014.

[41] Michael A Nielsen. *Neural networks and deep learning*. Vol. 25. Determination press San Francisco, CA, USA, 2015.

[42] *Redes Neurais Artificiais (Anderson Vinicius)*. Acesso: 05 de outubro de 2022. URL: https://medium.com/@avinicius.adorno/redes-neurais-artificiais-5b65a43614a0.

[43] John D Kelleher. *Deep learning*. MIT press, 2019.

[44] Jean Gallier e Jocelyn Quaintance. *Algebra, Topology, Differential Calculus, and Optimization Theory For Computer Science and Engineering*. 2019.

[45] *Arquitetura Redes Neuronais*. Acesso: 05 de novembro de 2022. URL: https://www.deeplearningbook.com.br/as-10-principais-arquiteturas-de-redes-neurais/.

[46] *Illustrated Guide to LSTM's and GRU's*. Acesso: 18 de agosto de 2023. URL: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21.

[47] *Árvores de Decisão*. Acesso: 09 de abril de 2023. URL: http://scikit-learn.org/stable/modules/tree.html.

[48] *Random Forest*. Acesso: 03 de março de 2023]. URL: https://serokell.io/blog/random-forest-classification.

[49] Leo Breiman. «Bagging predictors». Em: *Machine learning* 24 (1996), pp. 123–140.

[50] *XGBoost*. Acesso: 14 de julho de 2023. URL: https://xgboost.readthedocs.io/en/stable/.

[51] Rui Guo et al. «Degradation state recognition of piston pump based on ICEEMDAN and XGBoost». Em: *Applied Sciences* 10.18 (2020), p. 6593.

[52] Takuya Akiba et al. «Optuna: A Next-generation Hyperparameter Optimization Framework». Em: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2019.

[53] *Luís Távora, "Hyper-parameter tuning techniques" MSc on Data Science, ESTG-Polytechnic of Leiria, 2022.*

[54] Rodrigo Porteiro, Sergio Nesmachnow e Luis Hernández-Callejo. «Short term load forecasting of industrial electricity using machine learning». Em: Springer. 2020, pp. 146–161.

[55] Tuong Le et al. «Improving electric energy consumption prediction using CNN and Bi-LSTM». Em: *Applied Sciences* 9.20 (2019), p. 4237.

[56] Aghyad Al Skaif, Mohammad Ayache e Hussein Kanaan. «[1] Energy consumption clustering using machine learning: K-means approach». Em: *2021 22nd International Arab Conference on Information Technology (ACIT)*. IEEE. 2021, pp. 1–7.

[57] Niematallah Elamin e Mototsugu Fukushige. «Modeling and forecasting hourly electricity demand by SARIMAX with interactions». Em: *Energy* 165 (2018), pp. 257–268.

[58] Feng Sheng e Li Jia. «Short-term load forecasting based on SARIMAX-LSTM». Em: *2020 5th International Conference on Power and Renewable Energy (ICPRE)*. IEEE. 2020, pp. 90–94.

[59] Hakob Grigoryan. «Electricity consumption prediction using energy data, Socio-economic and weather indicators. A case study of Spain». Em: *2021 9th International Conference on Control, Mechatronics and Automation (ICCMA)*. IEEE. 2021, pp. 158–164.

[60] Ernesto Javier Aguilar Madrid. *Short-Term Electricity Demand Forecasting with Machine Learning*. 2021.

[61] Nevil Pooniwala e Rajendra Sutar. «Forecasting Short-Term Electric Load with a Hybrid of ARIMA Model and LSTM Network». Em: *2021 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE. 2021, pp. 1–6.

[62] Fj Vincent Atabay et al. «Multivariate Time Series Forecasting using ARIMAX, SARIMAX, and RNN-based Deep Learning Models on Electricity Consumption». Em: *2022 3rd International Informatics and Software Engineering Conference (IISEC)*. IEEE. 2022, pp. 1–6.

[63] Michael L. Waskom. «seaborn: statistical data visualization». Em: *Journal of Open Source Software* 6.60 (2021), p. 3021. DOI: 10.21105/joss.03021. URL: https://doi.org/10.21105/joss.03021.

[64] *Site IPMA*. Acesso: 02 de novembro de 2022. URL: https://www.ipma.pt/pt/.

[65] Jon Scott Armstrong. *Principles of forecasting: a handbook for researchers and practitioners*. Vol. 30. Springer, 2001.

[66] Geoffrey E Hinton et al. «Improving neural networks by preventing co-adaptation of feature detectors». Em: *arXiv preprint arXiv:1207.0580* (2012).

[67] *53 Semanas no Ano (Calendário Gregoriano)*. Acesso: 05 de fevereiro de 2024. URL: https://perma.cc/4ETJ-88QR.

[68] *Deteção de Anomalias*. Acesso: 14 de setembro de 2023. URL: https://www.analyticsvidhya.com/blog/2021/07/anomaly-detection-using-isolation-forest-a-complete-guide/.

[69] Ron Larson e Betsy Farber. *Elementary statistics*. Pearson Education Canada, 2019.

[70] *MSTL*. Acesso: 19 de agosto de 2023. URL: https://www.statsmodels.org/dev/examples/notebooks/generated/mstl_decomposition.html.

[71] Kasun Bandara, Rob J Hyndman e Christoph Bergmeir. *MSTL: A Seasonal-Trend Decomposition Algorithm for Time Series with Multiple Seasonal Patterns*. 2021. arXiv: 2107.13462 [stat.AP].

[72] *Jarque Bera*. Acesso: 14 de outubro de 2023. URL: http://www.ece.northwestern.edu/local-apps/matlabhelp/toolbox//stats/jbtest.html.

[73] Damodar N Gujarati e Dawn C Porter. *Econometria básica-5*. Amgh Editora, 2011.

[74] *Usando a função Auto Arima*. Acesso: 02 de fevereiro de 2024. URL: https://alkaline-ml.com/pmdarima/tips_and_tricks.html?highlight=aic.

[75] *Usando a Função Predict()*. Acesso: 13 de outubro de 2023. URL: https://www.statsmodels.org/stable/examples/notebooks/generated/predict.html.

[76] Stéphanie Portet. «A primer on model selection using the Akaike Information Criterion». Em: *Infectious Disease Modelling* 5 (2020), pp. 111–128.

[77] Diana Kornbrot. «Point biserial correlation». Em: *Wiley StatsRef: Statistics Reference Online* (2014).

[78] *Usando a Função Forecast()*. Acesso: 13 de outubro de 2023]. URL: https://www.statsmodels.org/stable/examples/notebooks/generated/statespace_forecasting.html.

[79] Ricardo Llugsi et al. «A novel Encoder-Decoder structure for Time Series analysis based on Bayesian Uncertainty reduction». Em: *2021 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*. IEEE. 2021, pp. 1–6.

[80] *Luís Távora, "Meeting Notes" MSc on Data Science, ESTG-Polytechnic of Leiria, 2024.*

[81] *Lags Adicionados*. Acesso: 08 de fevereiro de 2024. URL: https://www.statsmodels.org/dev/generated/statsmodels.tsa.tsatools.lagmat.

[82] Joseph F Hair. «Multivariate data analysis». Em: (2009).

[83] *KNNRegressor*. Acesso: 11 de julho de 2023. URL: https://scikit-learn.org/stable/modules/neighbors.html#regression.

[84] *Long Short-Term Memory Layer*. Acesso: 24 de setembro de 2023. URL: https://www.tensorflow.org/api_docs/python/tf/keras/layers/LSTM.

[85] *A Gentle Introduction to the Rectified Linear Unit (ReLU)*. Acesso: 26 de março de 2024. URL: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks.

[86] *Carlos Grilo, Rolando Miragaia "Deep Learning course notes" MSc on Data Science, ESTG-Polytechnic of Leiria, 2022-2023.*

[87] *TensorFlow (Optimizer RMSprop)*. Acesso: 23 de março de 2024. URL: https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/RMSprop.

[88] *Batch Size no Treinamento*. Acesso: 25 março de 2024. URL: https://www.deeplearningbook.com.br/o-efeito-do-batch-size-no-treinamento-de-redes-neurais-artificiais.

[89] Jiaqi Yu, Wen-Shao Chang e Yu Dong. «Building energy prediction models and related uncertainties: A review». Em: *Buildings* 12.8 (2022), p. 1284.

[90] *StackExchange (AIC Values)*. Acesso: 02 de fevereiro de 2024. URL: https://stats.stackexchange.com/questions/160612/auto-arima-doesnt-calculate-aic-values-for-the-majority-of-models.

[91] *AIC Infinito*. Acesso: 28 de janeiro de 2024. URL: https://www.statsmodels.org/stable/pitfalls.html#incomplete-convergence-in-maximum-likelihood-estimation.

[92] *Git Hub Arima*. Acesso: 29 de janeiro de 2024. URL: https://github.com/alkaline-ml/pmdarima.

[93] *Git Hub Rob Hyndman (Arima)*. Acesso: 02 de fevereiro de 2024]. URL: https://github.com/robjhyndman/forecast/blob/master/R/arima.R.

APÊNDICES

A

APÊNDICE A

A.1 *OUTPUT* FUNÇÃO AUTO_ARIMA

```
Performing stepwise search to minimize aic
 ARIMA(0,1,0)(1,1,1)[7]             : AIC=35884.101, Time=2.69 sec
 ARIMA(0,1,0)(0,1,0)[7]             : AIC=36770.715, Time=0.08 sec
 ARIMA(1,1,0)(1,1,0)[7]             : AIC=36218.324, Time=1.85 sec
 ARIMA(0,1,1)(0,1,1)[7]             : AIC=35661.647, Time=2.36 sec
 ARIMA(0,1,1)(0,1,0)[7]             : AIC=36456.624, Time=0.18 sec
 ARIMA(0,1,1)(1,1,1)[7]             : AIC=35615.978, Time=5.87 sec
 ARIMA(0,1,1)(1,1,0)[7]             : AIC=36100.474, Time=1.95 sec
 ARIMA(0,1,1)(2,1,1)[7]             : AIC=inf, Time=10.32 sec
 ARIMA(0,1,1)(1,1,2)[7]             : AIC=35613.689, Time=11.69 sec
 ARIMA(0,1,1)(0,1,2)[7]             : AIC=35615.134, Time=7.32 sec
 ARIMA(0,1,1)(2,1,2)[7]             : AIC=inf, Time=16.66 sec
 ARIMA(0,1,1)(1,1,3)[7]             : AIC=35613.647, Time=17.74 sec
 ARIMA(0,1,1)(0,1,3)[7]             : AIC=35616.610, Time=15.81 sec
 ARIMA(0,1,1)(2,1,3)[7]             : AIC=inf, Time=45.80 sec
 ARIMA(0,1,0)(1,1,3)[7]             : AIC=35884.407, Time=5.79 sec
 ARIMA(1,1,1)(1,1,3)[7]             : AIC=inf, Time=32.75 sec
 ARIMA(0,1,2)(1,1,3)[7]             : AIC=35493.439, Time=21.74 sec
 ARIMA(0,1,2)(0,1,3)[7]             : AIC=35497.316, Time=16.60 sec
 ARIMA(0,1,2)(1,1,2)[7]             : AIC=35493.205, Time=12.02 sec
 ARIMA(0,1,2)(0,1,2)[7]             : AIC=35495.932, Time=9.71 sec
 ARIMA(0,1,2)(1,1,1)[7]             : AIC=inf, Time=5.68 sec
 ARIMA(0,1,2)(2,1,2)[7]             : AIC=inf, Time=21.11 sec
 ARIMA(0,1,2)(0,1,1)[7]             : AIC=35541.079, Time=4.87 sec
 ARIMA(0,1,2)(2,1,1)[7]             : AIC=inf, Time=10.68 sec
 ARIMA(0,1,2)(2,1,3)[7]             : AIC=inf, Time=40.01 sec
 ARIMA(1,1,2)(1,1,2)[7]             : AIC=inf, Time=24.55 sec
 ARIMA(0,1,3)(1,1,2)[7]             : AIC=inf, Time=20.99 sec
 ARIMA(1,1,1)(1,1,2)[7]             : AIC=inf, Time=20.71 sec
 ARIMA(1,1,3)(1,1,2)[7]             : AIC=inf, Time=30.28 sec
 ARIMA(0,1,2)(1,1,2)[7] intercept   : AIC=35506.262, Time=23.54 sec

Best model:  ARIMA(0,1,2)(1,1,2)[7]
Total fit time: 441.394 seconds
```

Figura 76: *Output* de procura e otimização para o melhor modelo pelo auto_arima.

A.2 AIC INFINITO

Esta secção relata que encontrar informações específicas sobre o AIC infinito é desafiador, e não é o escopo deste trabalho Interpretação Bayesian, Máximo Verossímilhança e quaisquer outros aspetos matemáticos, estatísticos ou probabilísticos.

O fenômeno de AIC infinito pode ser menos comum e talvez não seja tão discutido na literatura em comparação com o uso padrão do AIC na seleção de modelos. Quanto a escolha de modelos encontra-se na literatura uma vasta discussão sobre alternativas de critério de informação, mas raras informações sobre o impacto de AIC Infinito em modelos da família ARIMA, embora os Livros-texto sobre séries temporais geralmente abordam em profundidade o uso do AIC na seleção de modelos.

No Fórun Cross Validated (Stack Exchange) encontra-se um comentário que "na documentação do algoritmo, o Inf é relatado quando a probabilidade do modelo é infinita ou quando a raiz mais baixa nos polinômios do modelo é inferior a 1,01"[90]

Mesmo a documentação do pacote auto_arima busca explicar como o AIC é calculado e que valores infinitos [91] podem surgir quando o polinômio autorregressivo do modelo está próximo de ser não estacionário ou quando o polinômio das médias móveis está próximo de ser não invertível, então o modelo é rejeitado pelo auto_arima, estabelecendo um valor infinito para o AIC relacionado a esse modelo [74, 92, 93], cabendo ao utilizador verificar a convergência.

A.3 OTIMIZAÇÃO OPTUNA

```
!pip install optuna
import optuna

def objective_LSTM2nd14_dp(trial):

# LSTM14 com Dropout

    n_units1 = trial.suggest_int('n_units1',16, 256)
    n_units2 = trial.suggest_int('n_units2',16, 256)
    dp1 = trial.suggest_float('dp1',0.1, 0.60)
    dp2 = trial.suggest_float('dp2',0.25, 0.65)
    dp3 = trial.suggest_float('dp3',0.25, 0.85)
    b_size_s= trial.suggest_int('b_size',20, 128)

\# codigo do utilizador para a rede neuronal

# numero de tentativas
trials=100

# Criando study
study_LSTM2nd14_dp = optuna.create_study()
study_LSTM2nd14_dp.optimize(objective_LSTM2nd14_dp, n_trials=trials)

print(study_LSTM2nd14_dp.best_trial.params) # melhor combinacao de valores.
print(study_LSTM2nd14_dp.best_trial.value) # valor_alvo obtido.

# grau de importancia dos hiperparametros
fig = optuna.visualization.plot_param_importances(study_LSTM2nd14_dp)
fig.show()


fig = optuna.visualization.plot_contour(study_LSTM2nd14_dp,
        params=["n_units1","n_units2", "dp1","dp2", 'dp3'])
fig.show()

fig = optuna.visualization.plot_contour(study_LSTM2nd14_dp,
        params=["b_size","n_units1",'n_units2'])
fig.show()
```

Figura 77: Exemplo de *Script* com Otimização Optuna.

DECLARAÇÃO

Declaro, sob compromisso de honra, que o trabalho apresentado neste Relatório, com o título: ***"Previsão de Curto Prazo para Consumo de Energia em Campi Universitários"***, é original e foi realizado por Paulo Roberto da Silva Oliveira (2213002) sob orientação dos Professores Carlos Fernando de Almeida Grilo, João Sousa, Luís Távora e Pedro Marques.

Leiria, Março de 2024

Paulo Roberto da Silva Oliveira