

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE ENGENHARIA GEOGRÁFICA, GEOFÍSICA E ENERGIA



Participação da geração renovável no mercado de reservas de um sistema eléctrico

João Pedro Passagem dos Santos

Mestrado em Engenharia da Energia e Ambiente

Dissertação orientada por:
Professora Doutora Ana Estanqueiro
Doutor Hugo Algarvio

2023

Resumo

A penetração na rede e o investimento crescente em fontes de energia renováveis variáveis estão a mudar a forma como os mercados da eletricidade se comportam. Os intervenientes baseiam-se em previsões para participar nos mercados, que fecham entre 1 e 37 horas antes das transacções de energia em tempo real.

Normalmente, os operadores de redes de transmissão (TSO) utilizam uma aquisição simétrica para a atribuição de potência secundária ascendente e descendente com base na procura prevista para os consumos horários do dia seguinte. Este trabalho utiliza técnicas de *machine learning* que calculam dinamicamente as reservas de energia secundária a montante e a jusante, utilizando os despachos programados para o dia seguinte e os despachos esperados de energias renováveis variáveis, a procura e outras características.

O modelo introduz dados abertos operacionais do TSO espanhol de 2014 a 2023 para treino. Os dados de referência e de teste datam de 2019 a 2022. A metodologia proposta melhora a utilização da potência reservada secundária descendente em cerca de 36% e da potência reservada secundária ascendente em quase 43%.

Palavras chave: sistemas de reserva, mercados de energia, redes neuronais, previsões

Abstract

The penetration in the grid and growing investment in variable renewable energy sources are changing the way electricity markets behave. Players rely on forecasts to participate in markets, closing between 1 and 37 hours ahead of real-time energy transactions.

Usually, transmission system operators (TSOs) use a symmetrical procurement for up and down secondary power allocation based on the expected demand for next day hourly consumptions. This work uses machine learning techniques that dynamically compute up and down secondary power reserves, using the day-ahead programmed and expected dispatches of variable Renewables, demand, and other features.

The model inputs operational open data from the Spanish TSO from 2014 to 2023 for training. Benchmark and test data date from 2019 to 2022. The proposed methodology improves the usage of down secondary reserved power by around 36% and up secondary reserved power by nearly 43%.

Palavras chave: reserve systems, energy markets, neural networks, forecast

Agradecimentos

Opcional, embora no caso de dissertações que decorram no âmbito de projetos financiados, por exemplo, pela FCT ou programas europeus devem ser mencionados aqui a referência e nome do projeto, e mais alguma informação conforme as regras de publicitação do projeto em questão.

Nome do Autor

Nomenclatura

Lista de siglas, acrónimos, abreviaturas e simbologia apresentadas por ordem alfabética.

Abreviaturas

(A/F)	Relação mássica ar/combustível
pme	Pressão média efectiva
vol	Volume

Siglas e acrónimos

ODS	Objetivos de Desenvolvimento Sustentável
vRES	variable Renewable Energy Systems
MIBEL	Mercado Ibérico de Eletricidade
APA	Agência Portuguesa do Ambiente
EDP	Eletricidade De Portugal
RNN	Recurrent neural network
CNN	Convolutional neural network
LSTM	Long short-term memory
AR	dec
MA	dec
ARMA	dec
ARIMA	dec
SARIMA	dec

Simbologia

A	Área
η	Eficiência
p	Pressão
T	Temperatura

Índice

Resumo	i
Abstract	ii
Agradecimentos	iii
Nomenclatura	iv
List of Figures	vii
List of Tables	viii
1 Introdução	1
1.1 Enquadramento	1
1.2 Objetivos e Perguntas de Pesquisa	2
1.3 Organização do Documento	2
2 Revisão bibliográfica	3
3 Contextos	4
3.1 Mercado de Serviços de Sistema	4
4 Estudo 1: Estimativa do parâmetro ρ da fórmula da REN	5
5 Estudo 2: Dimensionamento dinâmico da potência alocada na reserva secundária	13
5.1 Dados Utilizados	13
5.1.1 Aquisição dos Dados	13
5.2 Estudo dos dados	13
5.2.1 Correlações	15
5.2.1.1 Correlações entre atributos	15
5.2.1.2 Correlações Temporais	17
5.3 Tratamento dos dados	18
5.3.1 Dados de treino	20
6 Arquitecturas de Modelos	21
6.1 Camadas	21
6.1.1 Dense	21
6.1.2 Convolution	22
6.1.3 MaxPooling	23

6.1.4	Dropout	24
6.2	Blocos	25
6.2.1	Bloco Dense	25
6.2.2	Bloco CNN	25
6.2.3	Bloco LSTM	26
6.3	Arquiteturas	26
6.3.1	Vanilla	26
6.3.2	Stacked	26
6.3.3	UNET	26
6.4	Considerações adicionais	26
7	Ferramentas	27
7.1	Forecat	27
7.1.1	Construtor de modelos	27
7.1.2	Gerador de dados	27
7.2	MuadDib	28
8	Métricas	29
8.1	Métricas de modelo	29
8.2	Métricas de comparação modelo/benchmark	30
9	Métodos	32
9.1	Estatísticos	32
9.2	Redes Neurais	34
9.2.1	Arquiteturas	34
9.2.2	Função de Perda	34
9.2.2.1	Funções de Perda	35
9.2.2.2	Função de Perda Avançada	35
9.2.3	Função de Activação	36
9.2.4	Pesos	37
9.2.4.1	Temporais	37
9.2.4.2	Distância à média	37
9.3	Dados de Validação	37
9.4	Benchmark	37
10	Conclusões e sugestões futuras	41
10	Referências	42
A	Anexos	45

List of Figures

Figura 1.1 Objectivos de Desenvolvimento Sustentável da ONU	1
Figura 4.1 Histograma ρ	6
Figura 4.2 Valor do parâmetro ρ (hora)	7
Figura 4.3 Histograma ρ	8
Figura 4.4 Histograma ρ	9
Figura 4.5 Média historica de Banda de Reserva	10
Figura 4.6 Erro médio por hora Banda de Reserva	10
Figura 4.7 Erro médio por hora Banda de Reserva (Arredondamento)	12
Figura 5.1 Serie Temporal dos dados alvo	14
Figura 5.2 Janelas Temporais dos dados alvo	14
Figura 5.3 Frequência dos dados alvos	15
Figura 5.4 Correlação entre atributos	16
Figura 5.5 Valores de correlação entre atributos	17
Figura 5.6 Serie Temporal dos dados alvo	17
Figura 5.7 Outliers	19
Figura 5.8 Dados em falta	20
Figura 6.1 Ilustração de um neurónio. Adaptado de Haykin1999[Haykin1999]	21
Figura 6.2 Ilustração de uma camada dense.	22
Figura 6.3 Ilustração da operação de Convolução	23
Figura 6.4 Ilustração da camada de Convolução	23
Figura 6.5 Ilustrção do efeito da camada de Pooling	24
Figura 6.6 Ilustrção do efeito da camada de dropout	24
Figura 6.7 Ilustrção do bloco de Dense	25
Figura 6.8 Ilustração do bloco de CNN	25
Figura 9.1 Resultados de alocações totais em diferentes arquiteturas	34
Figura 9.2 Resultados de alocações totais em diferentes racios	36
Figura 9.3 Série Temporal dos dados de Benchmark c/ consumo real	38
Figura 9.4 Janelas temporais de benchmark energia a subir	39
Figura 9.5 Janelas temporais de benchmark energia a descer	40
Figura A.1 Exemplo de como considerar um gráfico nos anexos.	45

List of Tables

Tabela 4.1 Dados REN	5
Tabela 4.2 Valores de ρ apresentado em [16]	6
Tabela 4.3 Erros de Banda de Reserva por método de normalização ρ	9
Tabela 4.4 Valores de ρ propostos	11
Tabela 5.1 Indicadores retirados do site da ESIOS	13
Tabela 5.2 Dados de Treino	20
Tabela 9.1 Variáveis de estudo dos modelos AR/MA	33
Tabela A.1 Isto é um exemplo de uma tabela. Se fôr igual(copiada) a outro autor deve ser pedido autorização para reproduzir.	45

Introdução

1.1 Enquadramento

Esta dissertação encontra-se no âmbito do projeto [TradeRES](#), que estuda um sistema de mercado elétrico capaz de atender às necessidades da sociedade num sistema quase totalmente renovável. Tendo as características para se integrar nos [ODS 1.1](#).

O estudo da acessibilidade das energias renováveis ao mercado vigente integra-se nos ODS n.º7, Energia Renováveis e Acessíveis, indo directamente de encontro a um dos pontos deste objectivo: 7.2.1 Peso das energias renováveis no consumo total final de energia. Por meio deste objectivo, a participação das renováveis no mercado faz também cumprir, embora indirectamente, o objectivo n.º8 Trabalho Digno e Crescimento Económico, através do ponto 8.4, onde é dada primazia à eficiência dos recursos globais no consumo e na produção. Indirectamente, pois, ao haver um melhor uso das renováveis, o uso de energias não limpas vai diminuir, melhorando a gestão de recursos, e baixando o consumo de recursos naturais não renováveis.

Por último podemos incluir o objectivo n.º13, Acção Climática, no qual, referimos de novo a diminuição de consumo de recursos finitos, mas mais importante, a melhor gestão de recursos renováveis. Promovendo o planeamento e estratégias de combate a emissões de gases de efeito estufa.



Figure 1.1: Objectivos de Desenvolvimento Sustentável da ONU

1.2 Objetivos e Perguntas de Pesquisa

Foram aprovadas a nível europeu (2020) medidas de alteração aos serviços de sistema, que serão seguidas pelos Estados-Membros. Nesta dissertação, será realizada a aplicação dessas medidas, identificando as melhorias em relação ao design atual e avaliando se as novas medidas serão suficientes para garantir a operação de um sistema elétrico 100 renovável, potencialmente identificando ações adicionais para garantir a robustez e segurança do sistema elétrico sem o uso de combustíveis fósseis.

As seguintes perguntas guiarão esta pesquisa.

- a) É positivo para as vRES participar no mercado de reserva?
- b) Como configurar essa participação para otimizar o lucro do ponto de vista das vRES?
- c) Essa participação é positiva para o sistema eléctrico num todo?

Para responder a essas perguntas, utilizaremos dados de previsão de geração de energia renovável para estimar a energia necessária para alocação secundária. Atualmente, os valores de previsão desse mercado estão distantes do consumo real, o que resulta em alocações no dia anterior que não estão em conformidade com as necessidades reais. O objetivo deste trabalho é investigar se podemos obter previsões mais precisas da energia necessária utilizando técnicas de *Machine Learning* (Aprendizado de Máquina). Isso possibilitará uma melhor gestão das alocações, resultando em um menor gasto de recursos energéticos e financeiros.

1.3 Organização do Documento

Este documento está dividido em capítulos. Sendo que os primeiros apresentam uma introdução às ideias e temas 1.3, o estado de arte do temas na literatura publicada 2, e por fim uma contextualização dos temas abordados 3.1.

Segue um capítulo de explicação dos dados usado ??, onde se apresentam os mesmos juntamente com alguns estudos preliminares para compreender a natureza e qualidades dos mesmos.

O capítulo seguinte entra no âmbito experimental da dissertação, onde se apresenta as diferentes arquiteturas 6.4 utilizadas, incluindo uma explicação dos componentes das mesmas.

Os capítulos 6 e 7 são bastante paralelos, sendo que o sexto 9.4 apresenta a metodologia do trabalho, e explica todas as experiências, e o sétimo 9.4 apresenta apenas os resultados e conclusões experiência a experiência.

Termina com um capítulo conclusivo 10 onde são avaliadas as experiências como um todo, e o seu impacto no âmbito dos mercados de reserva.

Revisão bibliográfica

A análise de séries temporais é um tema recorrente em pesquisa. Especialmente para previsões.

Desde as previsões para mercados de acções[1], fenómenos meteorológicos[2], e especialmente mercados energéticos, onde se quer ter em consideração o impacto das gerações mais voláteis.

As energias renováveis, devido à sua natureza, são as produções mais voláteis, logo alvo de estudo ideal para estas tecnologias[3], energia eólica[4], energia solar[5], aplicabilidade dos vários sistemas[6], procura[7]. Sendo que cada problema já apresenta arquiteturas e soluções diferentes, como a geração de energia fotovoltaica em casas pode ser melhor prevista com LSTM[8] mas também com uso de SVM[9] As várias faces destas tecnologias estão optimamente apresentadas em[10]

Para o estudo de previsões de séries temporais chega a ser o caso se pesquisar primeiramente com *deep learning*, antes de procurar outras soluções.

Em [11] é visto o impacto dessa decisão, e se realmente compensa emergir em machine learning. O trabalho conclui que modelos simples, com alguma engenharia de atributos inteligente, consegue competir, ou mesmo passar as qualidades de redes neuronais profundas.

Esta conclusão mostra também que por vezes a procura por modelos mais complexos não compensa, e que cada problema/dataset deve ter a sua própria investigação e conclusão, consoante a quantidade/qualidade de recursos disponíveis.

Os sistemas de reservas de frequências dos mercado espanhol já foram alvo de estudo predictivo, através de redes neuronais profundas[12].

Este trabalho propõe prever o custo da alocação e faz usando um elevado número de atributos disponíveis pela ESIOS, 32 variáveis. Estes modelos tiveram de métrica MASE cerca de 64%, o que não foi considerado um bom valor.

Este

reserve allocation forecasting

A alocação de reservas ESTUDOS EM MERCADOS DE RESERVAS [13]

energy forecasting with ML

Previsões são um dos problemas mais comuns a ser tratados com aprendizagem automática.

[<empty citation>]

No âmbito de energias vários trabalhos vieram mostrar que o uso de *machine learning* para previsões energéticas tem aplicabilidade [14] e muitos casos resultados melhores do que usados na indústria corrente. [6] [7]

Como muitos trabalhos apresentados em [10], o uso destas técnicas está a crescer e a produzir frutos. Como concluído nesse trabalho as várias arquiteturas e modelos comuns de ML já foram aplicados em energia, especial nas áreas de consumo e produção.

deep neural network on forecasting

O uso de redes neuronais profundas para previsões é também por si próprio alvo de estudo [12]

Contextos

3.1 Mercado de Serviços de Sistema

O mercado de serviço de sistema é parte integrante dos mercados de energia e mantém responsabilidade sobre a segurança do mesmo.[15]

Serve para garantir o equilíbrio entre a energia produzido e a consumida. Esta qualidade e segurança é controlada através da frequência e da potência activa, controlo de tensão e potência reactiva, arranque automático e outras técnicas de sistemas [13] [16].

Neste caso de estudo estamos interessados nos serviços de controlo de frequência. A nível europeu estes serviços são impostos pela ENTSO-E (*European Network of Transmission System Operators for Electricity*), e a operação dos mesmos é da responsabilidade dos TSO (*Transmission System Operator* ou *Operador da Rede de Transporte*) nacionais.

Para manter o controlo de frequência o gestor de sistema deverá manter reservas para responder às diferenças entre a energia consumida e produzida na rede, que deve ser mantida em equilíbrio. Quando o serviço de sistema precisa de actuar para manter a frequência no seu valor nominal, 50Hz na Europa, isto é feito alterando a potência activa dos geradores.

Quando é necessário um aumento na potência chama-se a isto Banda de Reserva/Regulação a Subir, e quando é necessária uma diminuição chama-se à mesma a Descer.

Para isto, no mercado ibérico, a tarefa é dividida em três reservas, primária, secundária e terciária. Esta divisão assenta no tempo de resposta que os sistemas precisam de ter, e na capacidade de actuação (MWh/Hz).

A reserva secundária, como sistema de segurança à reserva primária, regula-se pelo mercado de banda das reservas secundárias, que decorre no dia anterior ao que será necessário utilização da mesma.

Este valor alocado tem um custo para as operadoras, como tal a previsão do mesmo é importante para a gestão destes sistemas de segurança. Estas previsões são feitas através de estatísticas dos sistemas, e tendo em conta as áreas de balanço que o mesmo têm.

Uma melhor previsão deste valor poderia levar a uma poupança, tanto financeira, como de recursos.

Estas previsões são feitas ao uso de formulas. Que por si só não preveem a variabilidade dos sistemas de produção de energia renovável. Esta variabilidade sendo dificilmente previsível, tem sido alvo de estudo com modelos de *machine learning* [**<empty citation>**]

Com bons resultados apresentados em estudo de energias renováveis, a aplicação dos mesmos métodos para as reservas de sistema parece um passo natural.

Estudo 1: Estimativa do parâmetro ρ da fórmula da REN

Para responder a primeira questão estudou-se o comportamento do parâmetro ρ na equação publicada pela REN para a Banda de Regulação Secundária a Subir:

$$BR = \rho \times \sqrt{a \times L_{max} + b^2} - b \quad (4.1)$$

onde:

- BR : Banda de Reserva de regulação secundária necessária (MW).
- ρ : Parâmetro horário.
- a e b : Coeficientes empíricos, $a=10\text{MW}$ e $b=150\text{MW}$.
- L_{max} : Pico máximo de consumo (MW).

Aqui queremos descobrir qual o valor do parâmetro ρ (por hora do dia) que melhor nos descreve os dados reais. Para isso estudamos os valores reais usados para a Banda de Reserva, os valores resultantes da proposta de ρ em [16] e os valores resultados do estudo aqui proposto. Aproximar o parâmetro ρ utilizando os dados históricos.

Todos os dados necessários são disponibilizados pelo operador do sistema no [site da REN](#), com exceção do consumo máximo expectável. Este parâmetro é então substituído pelo consumo real, como uma aproximação à formulação indicada previamente.

Os dados estudados contêm entradas horárias desde 2010 até ao fim de 2018. Com as seguintes variáveis:

Table 4.1: Dados REN

Variável	Unidades
BANDA SUBIR	MW
BANDA DESCER	MW
Consumo real	MW
Consumo Máximo ENTSO-E	MW

Na equação 4.1 BR equivale à soma de BANDA SUBIR e BANDA DESCER, onde aqui é sempre considerado que Banda a subir são $\frac{2}{3}$ da Banda de Reserva total e a Banda a descer é o restante $\frac{1}{3}$.

Aqui iremos aplicar o mapa de parâmetro ρ apresentado em [16] na formula 4.1 para o cálculo da Banda de Reserva Carneiro2016 como benchmark.

4. ESTUDO 1: ESTIMATIVA DO PARÂMETRO ρ DA FÓRMULA DA REN

Table 4.2: Valores de ρ apresentado em [16]

Hora	ρ
1/2/8/9/24	1,6
3/7/10/11/19/20	1,4
4	1,3
5/6/12/13/14/15/16/17/18/21/22/23	1,2

Por outro lado, usando os dados de consumo real, calculamos o ρ ideal para cada entrada, de onde estudamos a melhor normalização dos mesmos para cada hora.

O cálculo deste $\rho_{proposto}$ é apenas a utilização da fórmula 4.1 mas em função de ρ :

$$\rho = \frac{(BR + b)}{\sqrt{a \times L_{max} + b^2}} \quad (4.2)$$

Arredondando o $\rho_{proposto}$ a uma casa decimal, podemos verificar que o histograma das diferentes propostas difere bastante. Sendo que esta apresenta uma curva de distribuição normal.

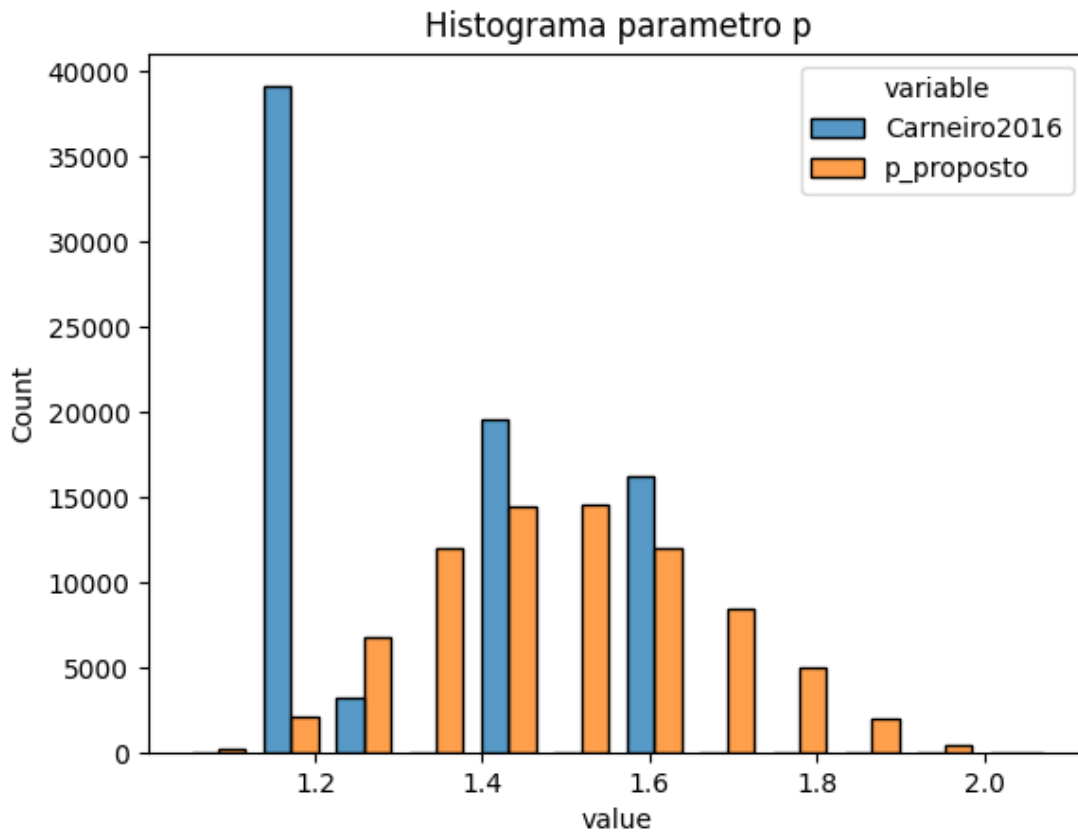


Figure 4.1: Histograma ρ

Olhando as distribuições por hora:

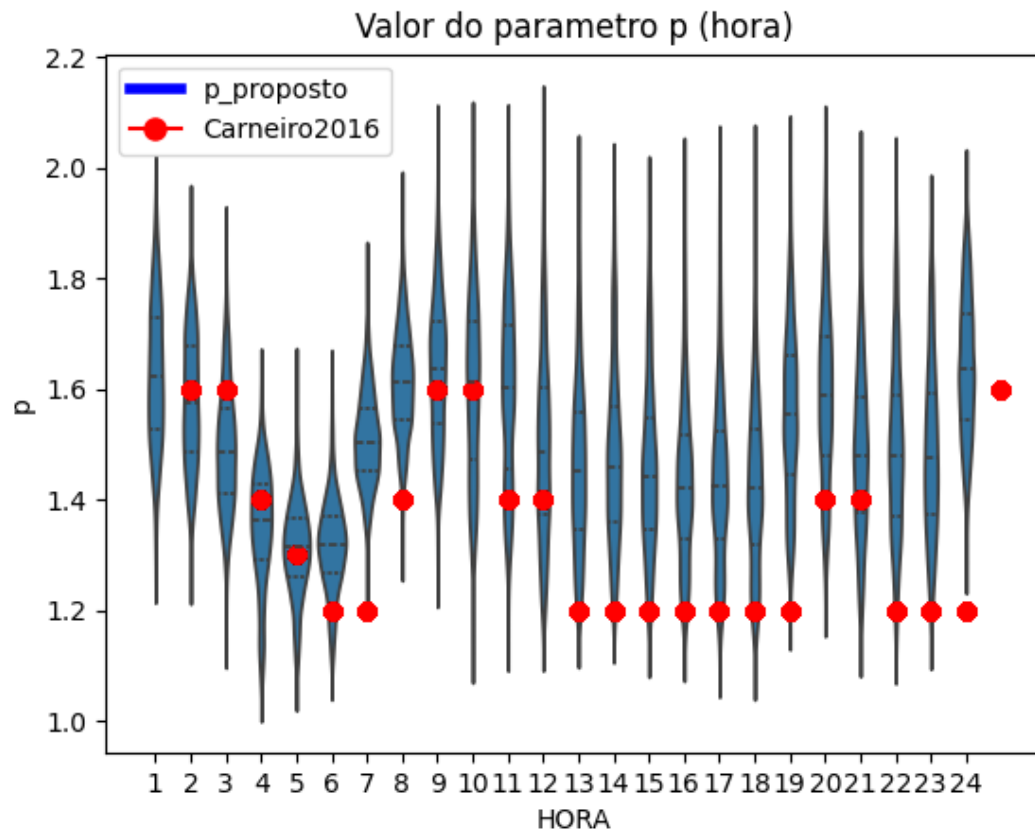


Figure 4.2: Valor do parâmetro ρ (hora)

O $\rho_{proposto}$ apresenta um grande variabilidade em todas as horas, embora de notar que em todas tem um maior peso perto da mediana. O ρ de comparação embora sempre dentro da distribuição note-se que cai quase sempre em zonas com pouco peso nestes dados históricos.

Calculamos ρ possíveis para proposta final usando as seguintes aproximações: média, mediana, e média ponderada ao consumo, e à banda.

As distribuições por hora são as seguintes:

4. ESTUDO 1: ESTIMATIVA DO PARÂMETRO ρ DA FÓRMULA DA REN

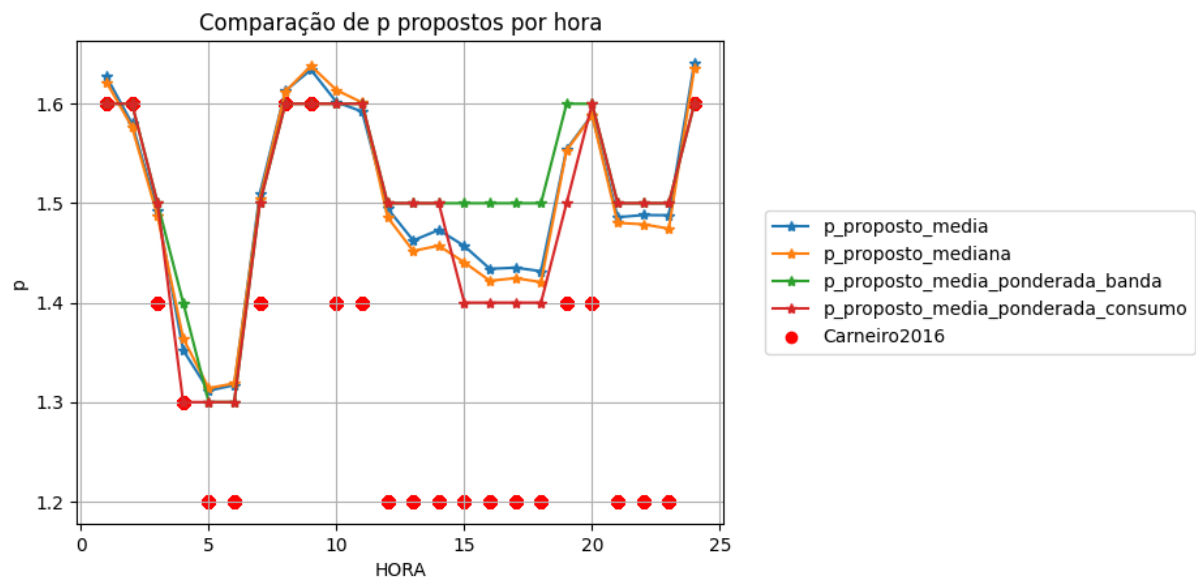


Figure 4.3: Histograma ρ

Todas seguem um percurso semelhante ao longo do dia, o qual também pode ser extrapolado para Carneiro2016. A média e mediana destacam-se seguindo muito parecidas, enquanto que as ponderadas também parecidas entre elas são bastante mais discretas.

Para a escolha da normalização deste parâmetro à Hora, estudou-se o erro entre a Banda Reserva calculada através das normalizações e a Banda Reserva disponível nos dados.

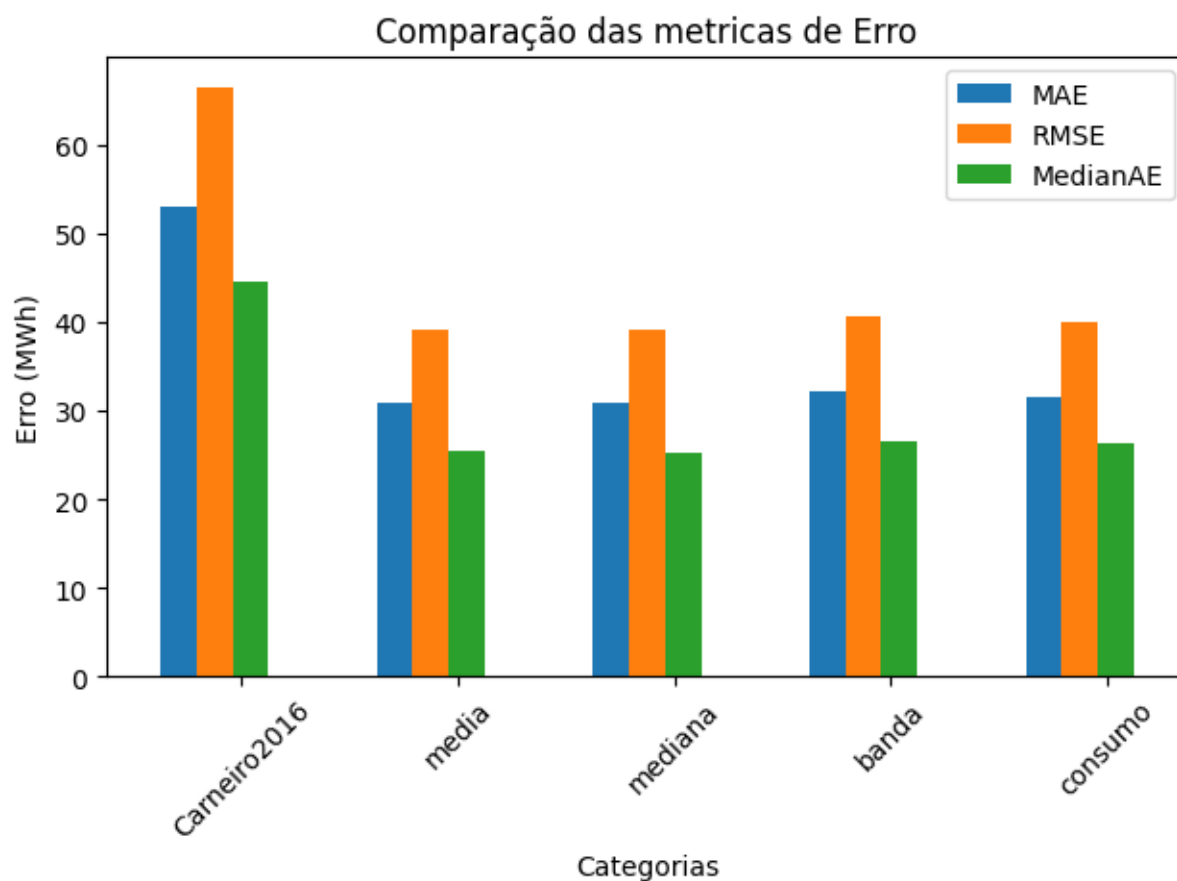


Figure 4.4: Histograma p

Table 4.3: Erros de Banda de Reserva por método de normalização ρ

Normalização	MAE (MW)	RMSE (MW)	MedianAE (MW)	MAPE (%)
Carneiro2016	53.07	66.54	44.53	18.70
média	30.94	39.19	25.38	11.58
mediana	30.85	39.20	25.17	11.51
média ponderada banda	32.15	40.61	26.45	12.19
média ponderada consumo	31.54	39.91	26.20	11.73

A normalização com erros mais baixos é a mediana. Com um erro médio (de todo o histórico) para o consumo real de 11.51% o que comparando com o benchmark de 18.70% é uma melhoria bastante considerável.

Comparando as bandas calculadas a uma média em cada hora:

4. ESTUDO 1: ESTIMATIVA DO PARÂMETRO ρ DA FÓRMULA DA REN

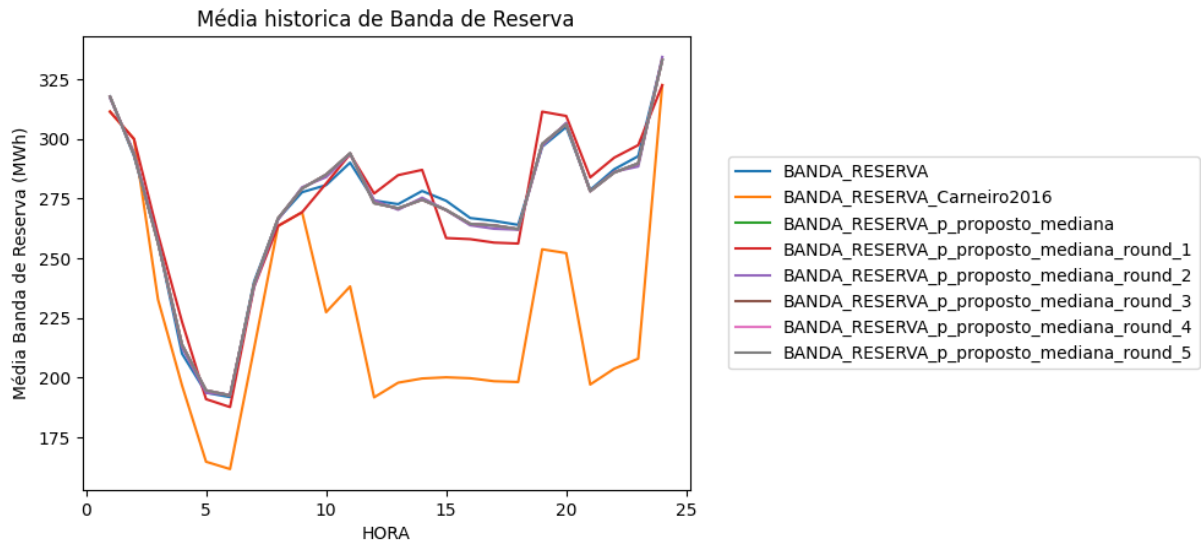


Figure 4.5: Média histórica de Banda de Reserva

Podemos ver que em termos de média horária, a Banda de Reserva calculada através do $\rho_{proposto}$ apresenta quase uma sobreposição por inteiro ao valor médio real.

Retiramos as médias dos erros percentuais e podemos observar:

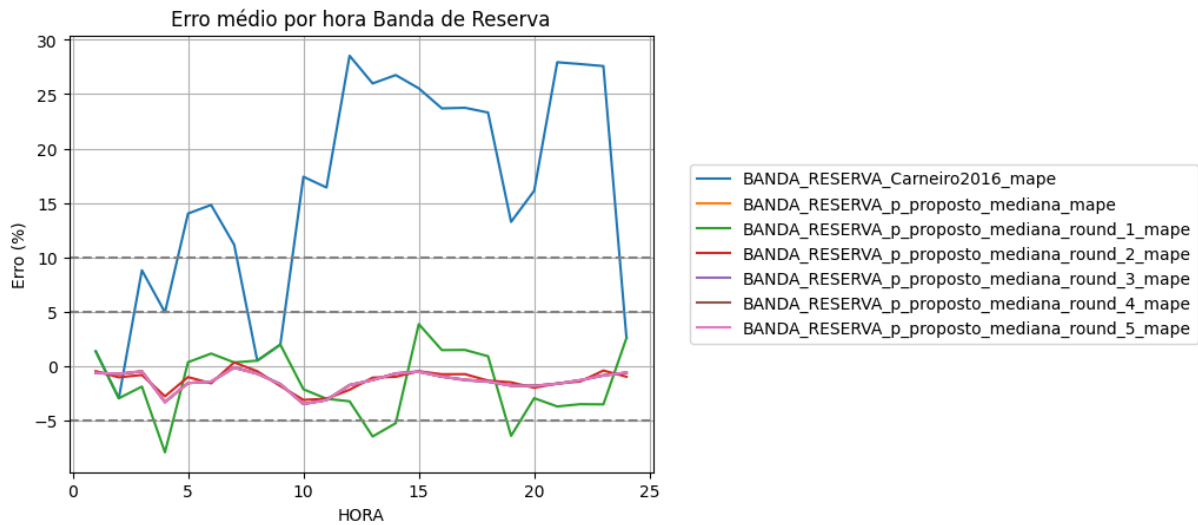


Figure 4.6: Erro médio por hora Banda de Reserva

Em termos de média diária o erro pelo método proposto está bem abaixo da margem de erro do 5% na banda, em todas as horas. E na outra tese apenas 10% cai dentro dessa margem de erro.

Como tal o $\rho_{proposto}$ a partir do estudo dos dados históricos é:

Table 4.4: Valores de ρ propostos

Hora	ρ
1	1.621694
2	1.576623
3	1.486929
4	1.364176
5	1.313958
6	1.318832
7	1.504499
8	1.612361
9	1.638188
10	1.613728
11	1.601277
12	1.485861
13	1.451995
14	1.457233
15	1.440454
16	1.421988
17	1.424636
18	1.420682
19	1.553086
20	1.588201
21	1.480219
22	1.478815
23	1.474412
24	1.635658

Em relação a perdas por arredondamento, apresento o resultado dos erros por arredondamento em cada um das casas possíveis, concluindo que até à primeira casa decimal, pode ser feito arredondamento do parâmetro ρ , sem influenciar muito o erro:

4. ESTUDO 1: ESTIMATIVA DO PARÂMETRO ρ DA FÓRMULA DA REN

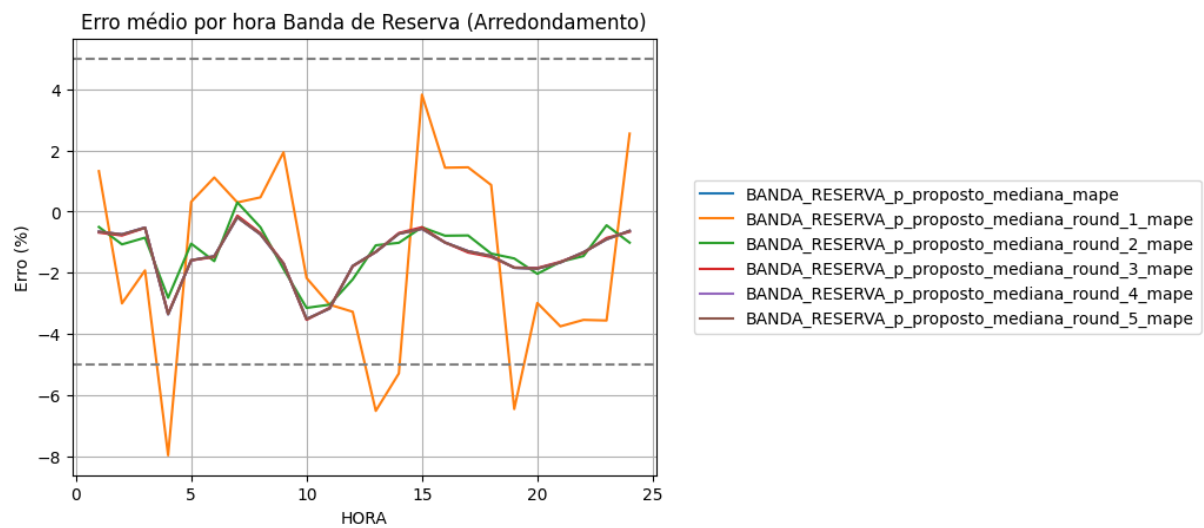


Figure 4.7: Erro médio por hora Banda de Reserva (Arredondamento)

Estudo 2: Dimensionamento dinâmico da potência alocada na reserva secundária

5.1 Dados Utilizados

Os dados em estudo são do mercado energético espanhol, retirados do site da [ESIOS](#).

Table 5.1: Indicadores retirados do site da ESIOS

Indicador ESIOS	Nome	Unidades
632	SecondaryReserveAllocationAUpward	MW
633	SecondaryReserveAllocationADownward	MW
680	UpwardUsedSecondaryReserveEnergy	MWh
681	DownwardUsedSecondaryReserveEnergy	MWh
1777	WindD+1DailyForecast	MWh
1779	PhotovoltaicD+1DailyForecast	MWh
1775	DemandD+1DailyForecast	MWh
10258	TotalBaseDailyOperatingSchedulePBFGeneration	MWh
14	BaseDailyOperatingSchedulePBFSolarPV	MWh
10073	BaseDailyOperatingSchedulePBFWind	MWh
10186	BaseDailyOperatingShedulePBFTotalBalanceInterconnections	MWh

5.1.1 Aquisição dos Dados

No âmbito da automatização destes dados foi modificado o repositório [ESIOS](#) para ser usado como uma biblioteca de python, aberta, em pypi.

Sendo uma ferramenta mais facilmente acessível para a extrair dados do mercado espanhol, [pyesios](#).

No âmbito de automatizar o processo, foram feitas contribuições a esta ferramenta para tornar mais acessível, e uma ferramenta aberta de python

5.2 Estudo dos dados

Os dados que proponho a prever são os de Energia Usada na Banda de Reserva Secundária, tanto a subir como a descer:

"UpwardUsedSecondaryReserveEnergy", "DownwardUsedSecondaryReserveEnergy".

5. ESTUDO 2: DIMENSIONAMENTO DINÂMICO DA POTÊNCIA ALOCADA NA RESERVA SECUNDÁRIA

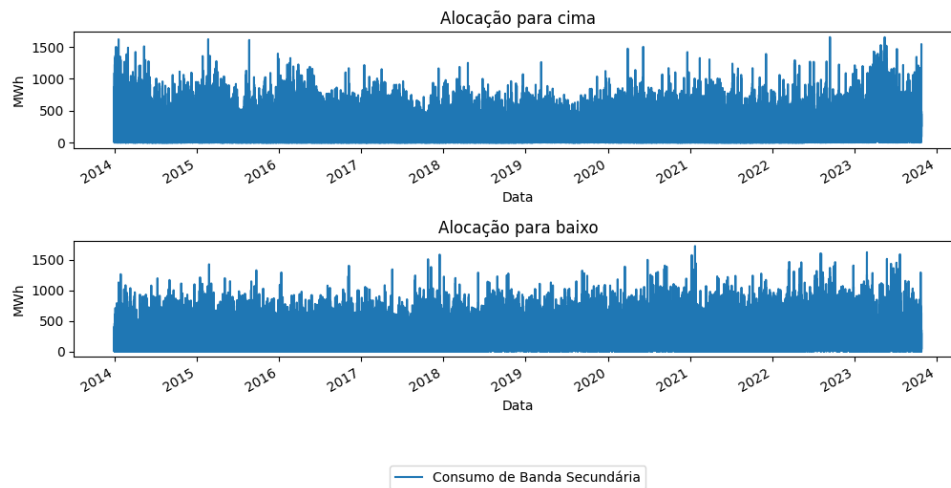


Figure 5.1: Serie Temporal dos dados alvo

Para termos uma melhor percepção dos mesmos segue algumas janelas temporais mais pequenas.

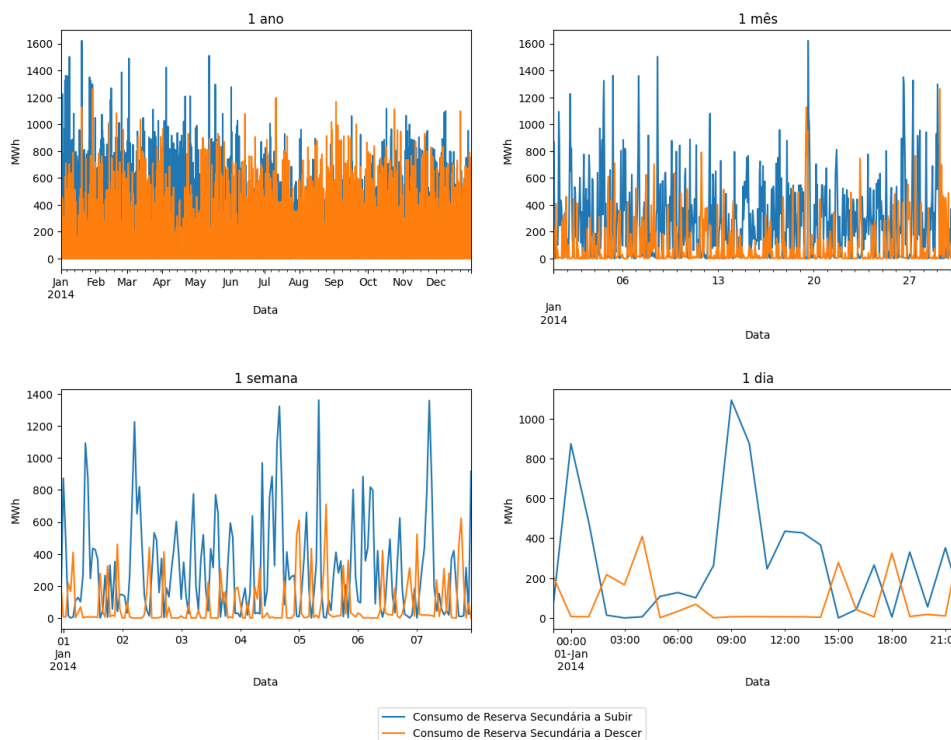


Figure 5.2: Janelas Temporais dos dados alvo

Estas mostram claramente que ambos os atributos mantêm um comportamento tanto discreto, como linear, isto é, que ou existe algum valor, ou é zero, e se existe valor este tem comportamento linear.

A distribuição destes dados é claramente exponencial. O que é importante para a escolha de alguns parâmetros na modelação.

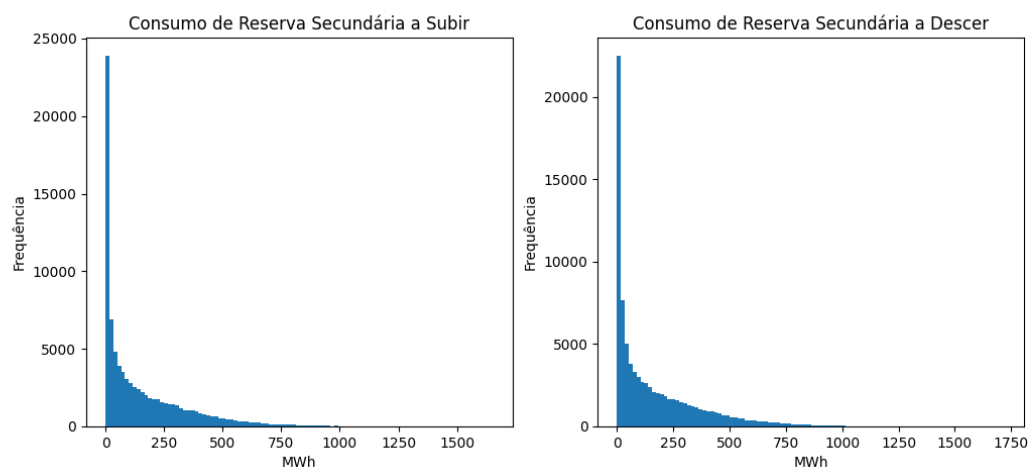


Figure 5.3: Frequência dos dados alvos

5.2.1 Correlações

5.2.1.1 Correlações entre atributos

Os modelos vão depender bastante de correlação entre variáveis.

Nesta secção queremos tentar identificar se há visíveis relações entre as variáveis, e se há relações temporais visíveis nas colunas alvo.

5. ESTUDO 2: DIMENSIONAMENTO DINÂMICO DA POTÊNCIA ALOCADA NA RESERVA SECUNDÁRIA

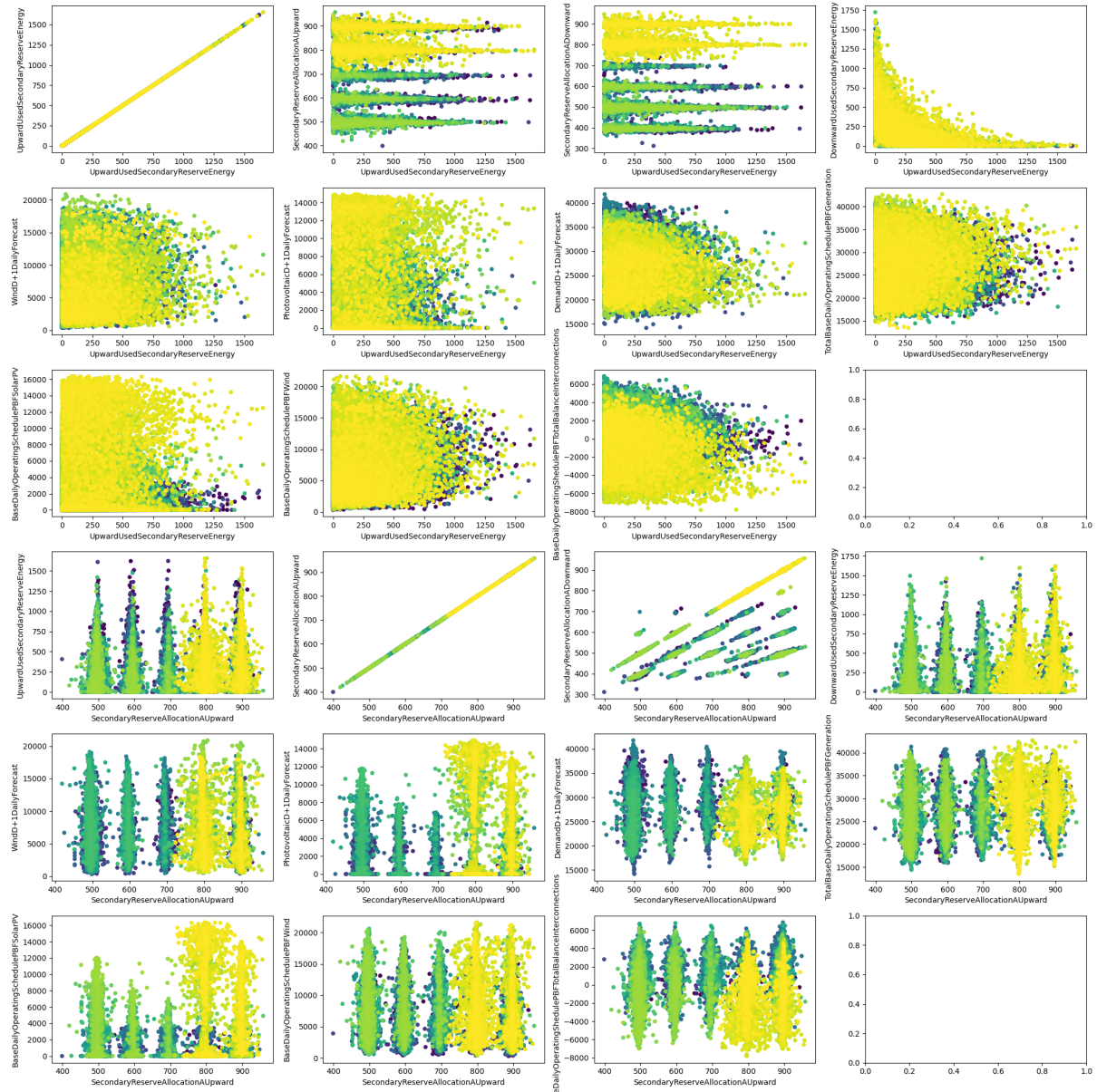


Figure 5.4: Correlação entre atributos

As correlações entre variáveis parecem muito escassas, o que apresenta já que a previsão destes dados usando estas variáveis vai ser um problema difícil.

Por norma é feito uma seleção de atributos baseado nestas correlações, eliminando assim os atributos que ajudam menos, ou até prejudicam os modelos.

Segue os valores de correlação onde podemos ver numericamente que existe muito pouca correlação entre os atributos.

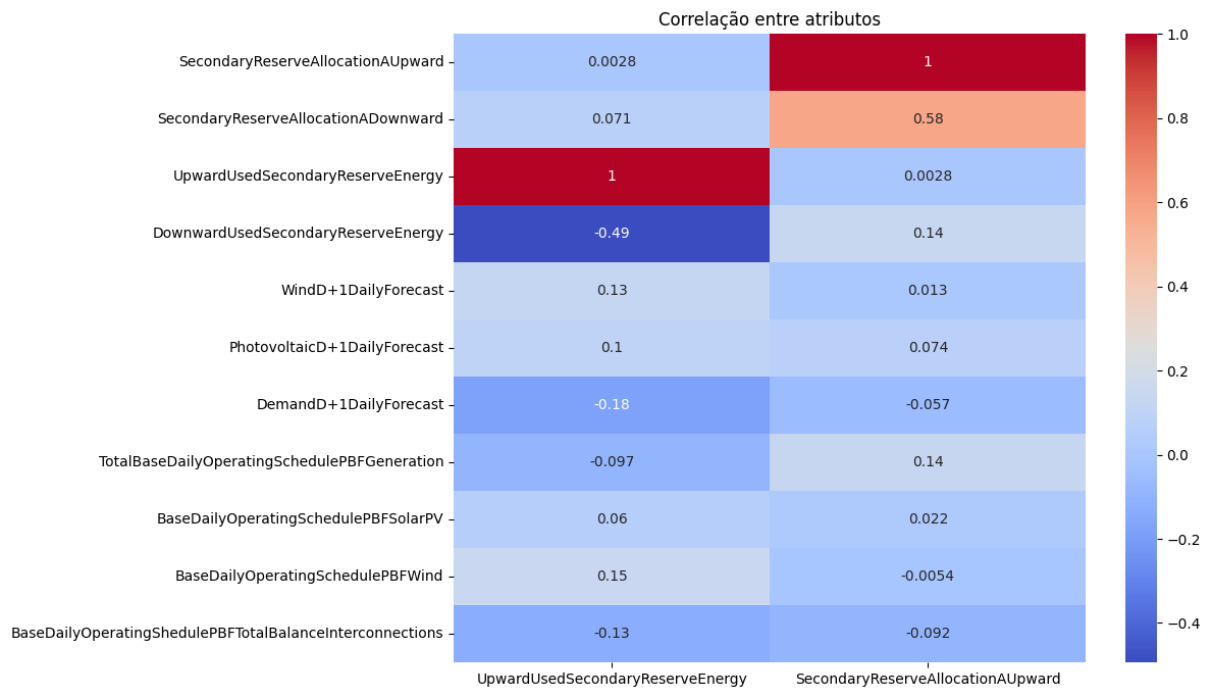


Figure 5.5: Valores de correlação entre atributos

5.2.1.2 Correlações Temporais

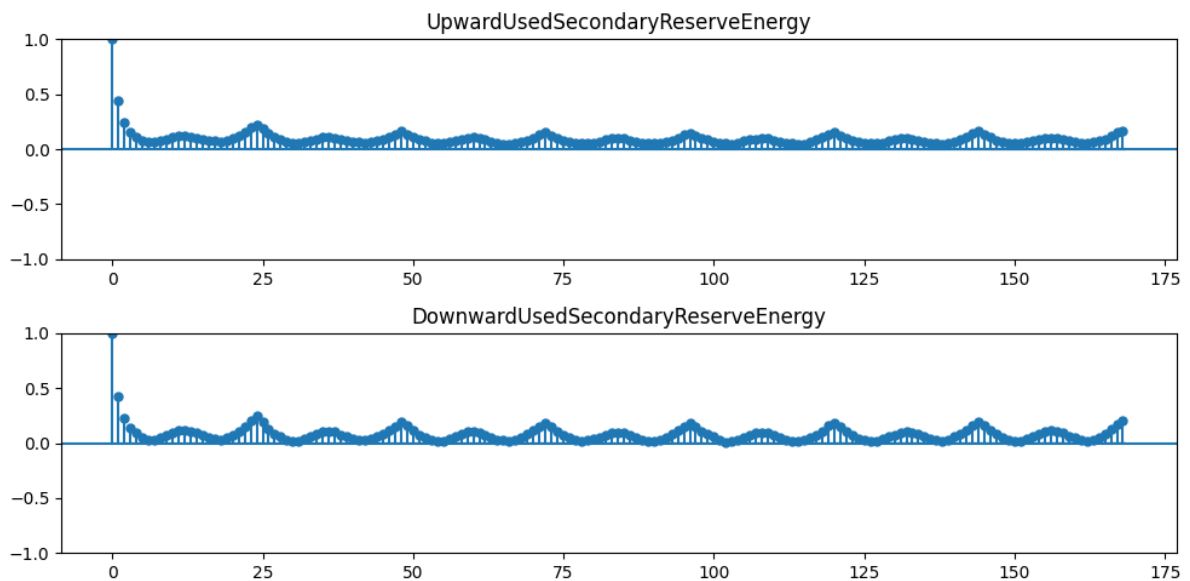


Figure 5.6: Serie Temporal dos dados alvo

A autocorrelação, em ambos os alvos, é mais forte nas 3 horas mais próximas, e nos pontos com diferença de 12 e 24 horas.

É de notar que estes valores são baixos, prometendo já também uma baixa regressividade temporal.

Outro ponto a denotar é que os objectos não têm um comportamento completamente linear, i.e., parece existir um comportamento discreto na questão ser alocado ou não estas reservas secundárias, e caso seja alocado, aí existir alguma linearidade.

Logo qualquer tipo de modelação terá de resolver primeiramente este problema.

Estas relações mostram que em termos de atributos usados vai ser um desafio complicado para qualquer tipo de modelo.

No âmbito desta dissertação queremos verificar a qualidade das previsões usando estes mesmo atributos, logo, não será feita seleção dos mesmos.

A nível da relação temporal, a maior parte dos modelos que iremos testar aplica um janela na dimensão temporal, usando todos os valores nessa janela, e aplicando os pesos nessas distâncias que mais se enquadram. Logo também não é relevante escolher apenas as distâncias temporais com maior correlação, pois os modelos vão fazer essa pesagem.

5.3 Tratamento dos dados

Normalização

A normalização foi deixada por ser aprendida nos modelos, sendo que todos têm como segunda camada, uma de normalização.

Limpeza

Podemos ver pelos gráficos seguintes que existem alguns outliers, sendo estes definidos como 3 desvios padrão de distância à média.

Estes gráficos mostram também que existe uma variação do que são os valores normais de cada atributo a nível temporal. Logo um método de limpeza não se poderia basear apenas numa definição geral de outliers, mas teria de ser feito em janelas temporais.

Pelo mesmo argumento e visto que os outliers fazem parte do que queremos também descobrir, não é aplicada nenhum método de remoção dos mesmos, sendo os dados passados a cru para os modelos.

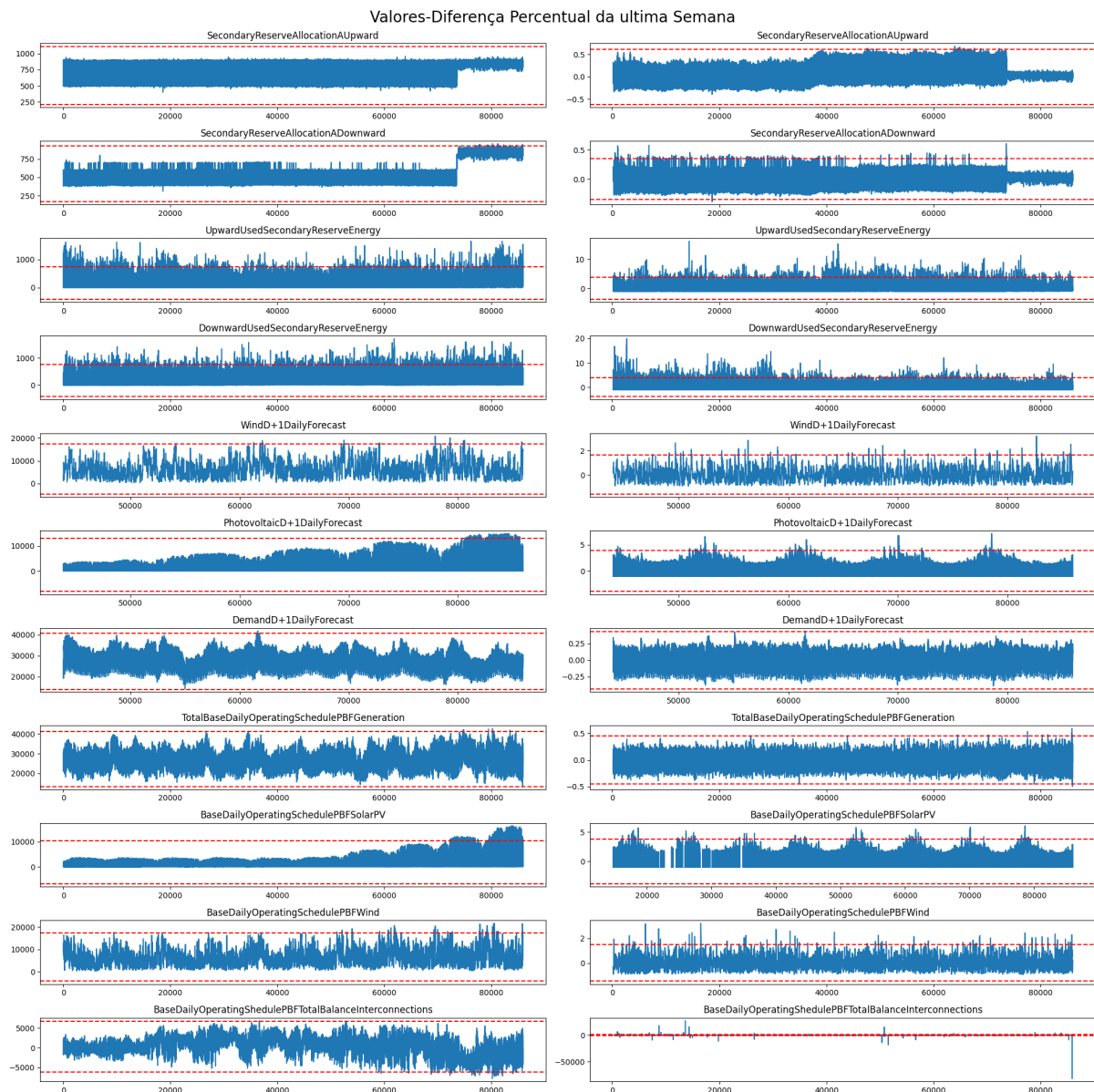


Figure 5.7: Outliers

Outra análise desta variação dos atributos a nível temporal leva-nos a que qualquer divisão dos dados para treino e teste deva levar as variações em consideração. Isto sendo que o treino deve ter representatividade de todas, ou maior parte, das condições diferentes.

Dados em falta (Missing Data)

Estudemos também o caso de dados em falta. Alguns destes atributos têm certas entradas vazias, e como podemos ver alguns não têm alguns anos inteiros.

Como queremos usar o máximo de dados possíveis iremos usar técnicas de imputing nesses dados.

Podemos ver que temos dados em falta de vários anos, em três atributos, e um tem algumas horas esporádicas em falta nos primeiros anos.

5. ESTUDO 2: DIMENSIONAMENTO DINÂMICO DA POTÊNCIA ALOCADA NA RESERVA SECUNDÁRIA

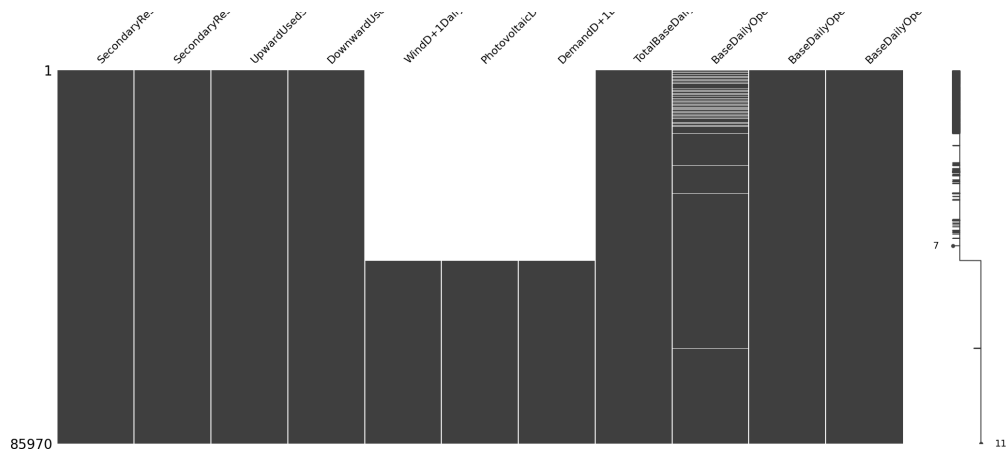


Figure 5.8: Dados em falta

Vamos aplicar o método experimental [IterativeImputer](#) da biblioteca de python [sklearn](#).

Este metodo é baseado nos trabalhos de [17] e de [18]

Por ultimo foi adicionado ao dados mais atributos, sendo eles todos de cariz temporal. É adicionado atributos correspondentes à hora, ao dia do ano, ao dia da semana, ao dia do mês, mês, ano.

5.3.1 Dados de treino

Após o tratamento apresentado as estatísticas gerais dos dados usados para treinar o modelo são:

Table 5.2: Dados de Treino

	média	desvio padrão	min	max
DownwardUsedSecondaryReserveEnergy	168.20	199.67	0.00	1721.40
SecondaryReserveAllocationAUpward	662.94	150.62	399.00	958.00
SecondaryReserveAllocationADownward	549.27	126.67	312.00	956.00
UpwardUsedSecondaryReserveEnergy	158.10	191.62	0.00	1654.80
WindD+1DailyForecast	5824.12	3413.15	71.33	20879.30
PhotovoltaicD+1DailyForecast	1666.31	2719.60	0.00	14925.30
DemandD+1DailyForecast	27944.24	4479.39	14170.00	41773.00
TotalBaseDailyOperatingSchedulePBFGeneration	27249.43	4603.58	13470.50	42707.60
BaseDailyOperatingSchedulePBFSolarPV	1714.09	2815.35	0.00	16358.90
BaseDailyOperatingSchedulePBFWind	6525.51	3582.36	308.60	21619.60
BaseDailyOperatingShedulePBFTotalBalanceInterconnections	290.58	2157.11	-7817.00	6858.50

Arquitecturas de Modelos

Grande parte da literatura sobre previsões em modelos de aprendizagem apresenta as mesmas arquiteturas, sendo que são depois aprimoradas consoante os dados e o problema.

Apresento aqui as arquiteturas mais usadas em previsões, como também algumas usadas noutros ramos tentando prever a compatibilidade neste problema.

As arquiteturas irão seguir um esquema lógico comum, um bloco de camadas de entrada, um bloco principal e um por fim um bloco interpretativo.

As dimensionalidades destas camadas é o que irá formar as diferentes arquiteturas em estudo.

6.1 Camadas

Para uma construção de modelos usando a ferramenta [keras](#) a unidade básica são as camadas. Estas representam uma operação, com uma entrada, e uma saída, e com possíveis parametrizações específicas.

Estas camadas ligadas entre si, perfazem um *profundo* de camadas neurais, chamado *profundo* pois tem mais que uma camada.

Apresento aqui as camadas utilizadas nos modelos aplicados.

6.1.1 Dense

inputs

weights

activation function

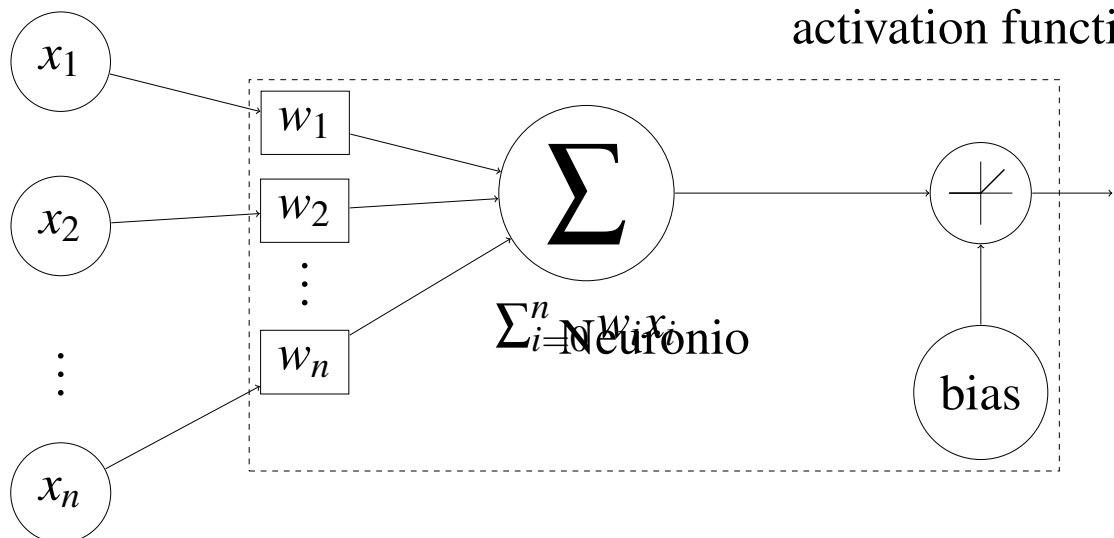


Figure 6.1: Ilustração de um neurônio. Adaptado de Haykin1999[Haykin1999]

A camada dense pega num input, cria um número de neurónios, N , também chamado número de filtros, onde cada neurónio (filtro), recebe informação de cada uma das entradas, e todos os neuronios ligam a todas as dimensões de saída.

Cada neurónio gera uma operação, inicialmente aleatória, para tentar reproduzir uma função que traduza a entrada na saída ideal.

Esta camada é altamente influenciada pelo *Perceptão* inicialmente proposto por Franck Rosenblatt[19]. Este apresentava um *Perceptão* que fazia uma decisão binária baseado na somas pesadas de todas as entradas.

A ideia é a base utilizada actualmente, mas apresentava algumas limitações, e muita computação, o proposto por Minsky and Papert[Minsky1969], eleva a ideia com a introdução da função de activação e o bias.

A utilização mais recorrente actual é a proposta por Haykin[Haykin1999], que baseada nas anteriores tem a seguinte apresentação:

O conjunto destes neuroneis perfaz a camada de dense:

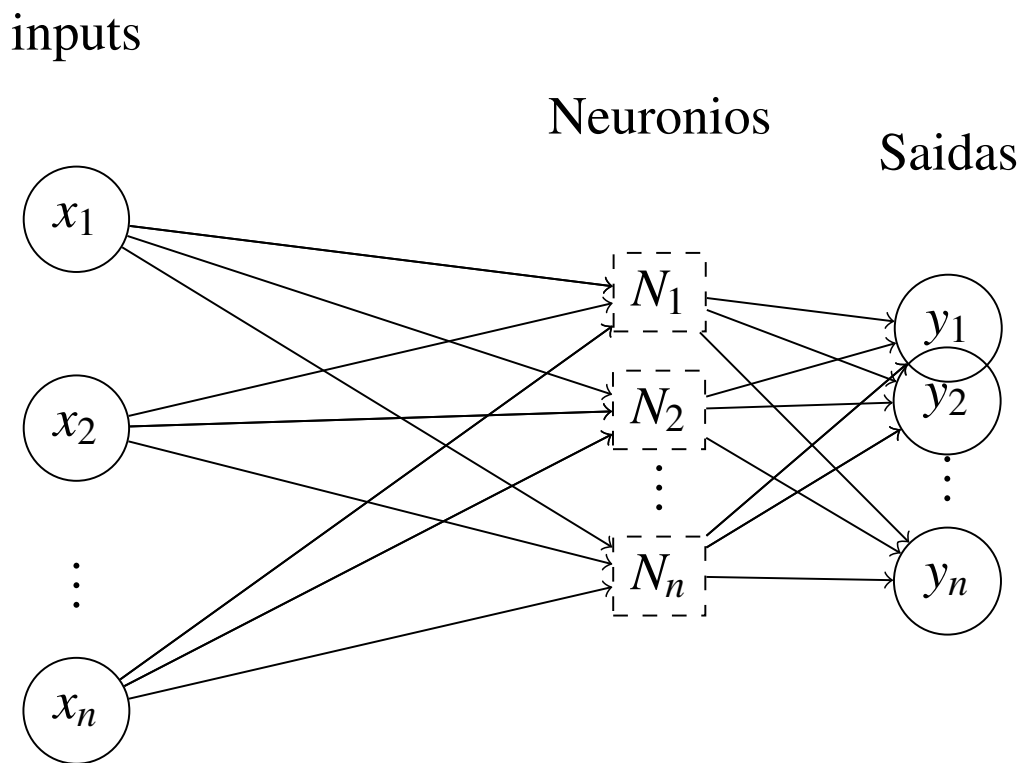


Figure 6.2: Ilustração de uma camada dense.

6.1.2 Convolution

A camada de convoluções difere da dense no sentido em que os filtros (neurónios) não são criados aleatoriamente, mas sim cada filtro trata de uma parte da camada de entrada. Nas convoluções é criada uma janela móvel que percorre a camada, criando uma saída desse conjunto de pontos. Esta janela move-se sempre subsequentemente.

Esta operação é normalmente feita na dimensão (ou dimensões) em que queremos perceber padrões. Nos nossos dados a convolução será na dimensão temporal.

Se tivermos uma matriz com nove passos temporais ($N, 9, 1$), se o tamanho da janela de convolução for 3, teremos uma saída de tamanho 6 ($N, 6, 1$).

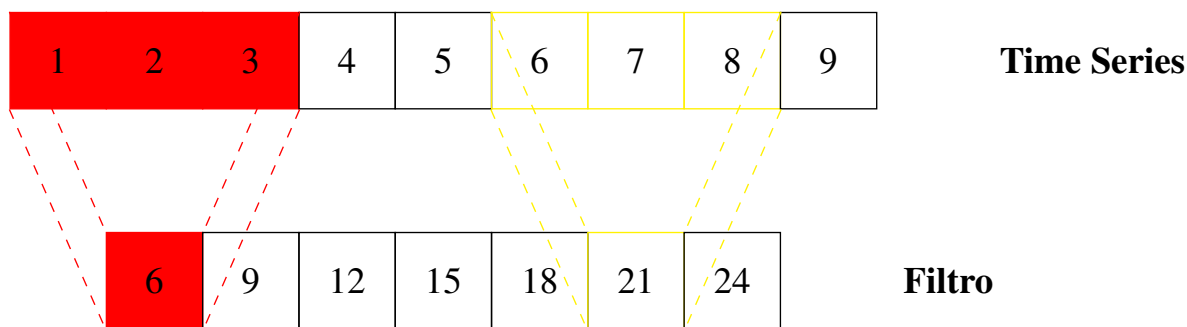


Figure 6.3: Ilustração da operação de Convolução

Anteriormente ignoramos o número de filtros. Mas as convoluções criam o número pedido de filtros para cada janela temporal. Aqui cada filtro vai funcionar como na camada dense, onde cada um começa com uma operação pseudo aleatoria. Esta operação está normalmente é feita na dimensão dos atributos. Ou seja, a quantidade de filtros que esta camada irá produzir por convolução.

Se tivermos a mesma entrada que anteriormente mas com 4 atributos (N, 9, 4), e se definir o número de filtros para 2 teremos uma saída (N, 6, 2).

Ou seja dois filtros por cada janela temporal.

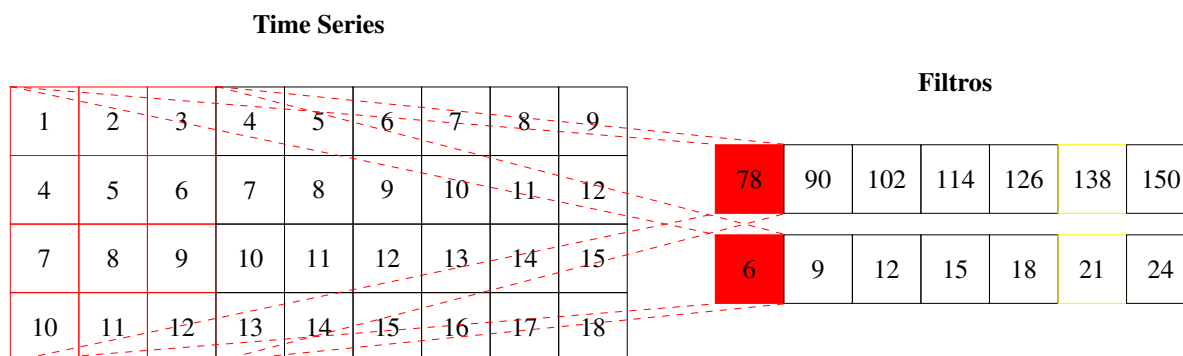


Figure 6.4: Ilustração da camada de Convolução

As convoluções podem realizar as operações em mais dimensões, é comum usar 2D para imagens, e 3D para videos. Neste trabalho apenas trabalharemos com convoluções 1D.

6.1.3 MaxPooling

As camadas de pooling fazem operações para redimensionar os filtros anteriores.

Usando também uma janela movel, estas camadas escolhem um dos valores da janelas para o resultado na saída. Operações comuns de pooling são, o maximo, ou a media desses valores. Sendo que a operação é feita na dimensão, ou nas dimensões, não dos filtros.

Ou seja, se tivermos um tensor de formato (N, 4, 4), e tivermos uma janela de tamanho 3, iremos produzir uma reposta com o formato (N, 2, 4).

Estas camadas são usadas principalmente para permitir uma escolha dos filtros mais relevantes e assim combater tanto overfitting como acelerar o processo de treino. [20] A camada usada nestas arquitecturas é MaxPooling, que escolhe o maior valor dentro da janela de strides, e aplica na saída.

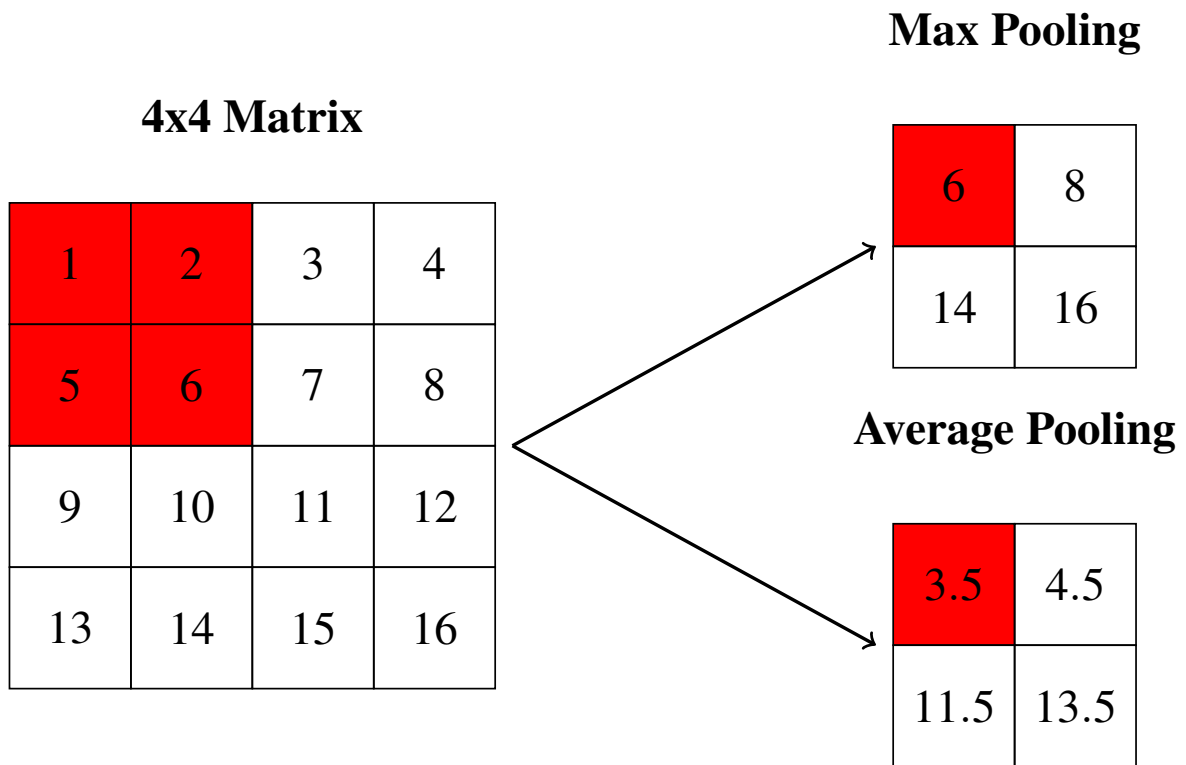


Figure 6.5: Ilustração do efeito da camada de Pooling

6.1.4 Dropout

Dropout é uma camada que elimina/ignora alguns dos neurônios da camada anterior. Este procedimento impede o overfitting, ajudando na generalização.

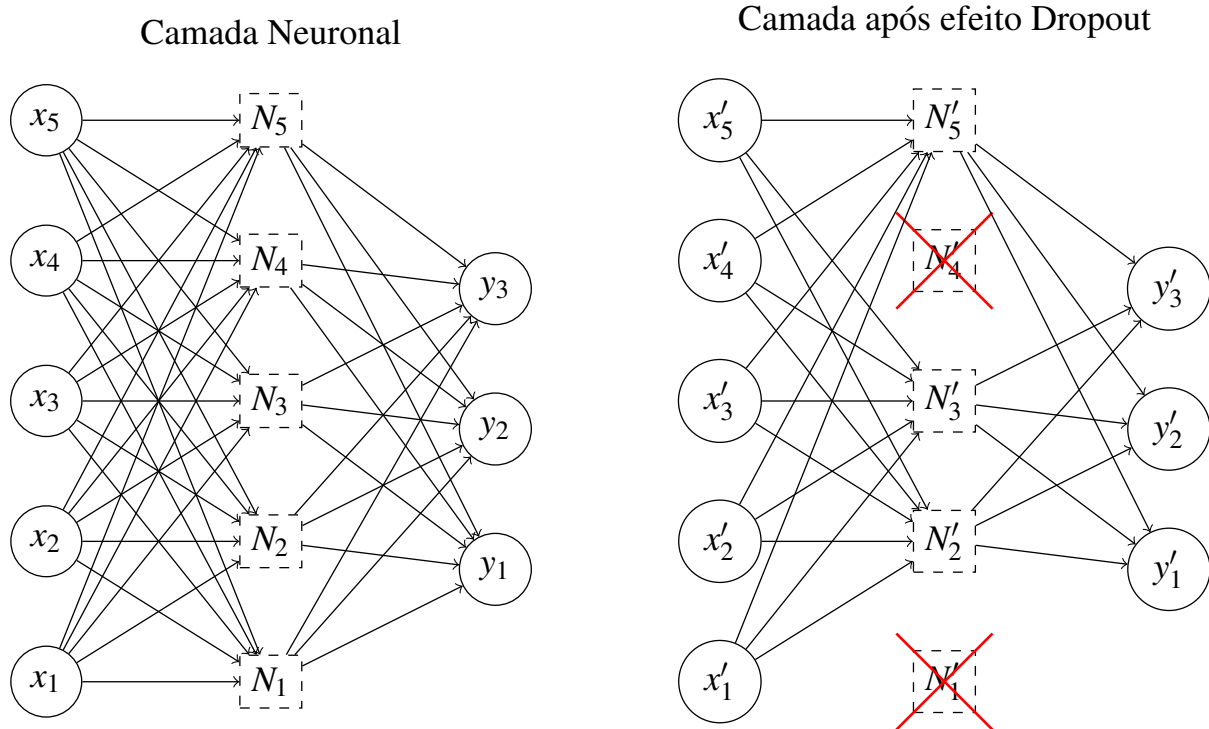


Figure 6.6: Ilustração do efeito da camada de dropout

6.2 Blocos

Todas as arquiteturas em análise irão ter por base um bloco de camadas neuronais. A formação dessas arquiteturas passa pelas diferentes maneiras que se pode utilizar o bloco principal. Repetições em serie ou em paralelo são um exemplo.

6.2.1 Bloco Dense

O bloco dense sendo ele o mais simples é formado por duas camadas Dense 6.1.1 [**<empty citation>**], em que a primeira apresenta um número maior de filtros que a segunda.

Estas camadas não são mais do que uma criação de filtros aleatórios combinando as entradas, para criar todos os filtros de saída. São a base das camadas interpretativas. A acumulação em série (stacked) de camadas de dense está ligada a melhorias nas capacidades predictivas dos modelos [21].

Exemplo ilustrativo do nosso bloco basico onde entrariam 16 filtros na primeira camada e para finalizar o bloco com 2 filtros

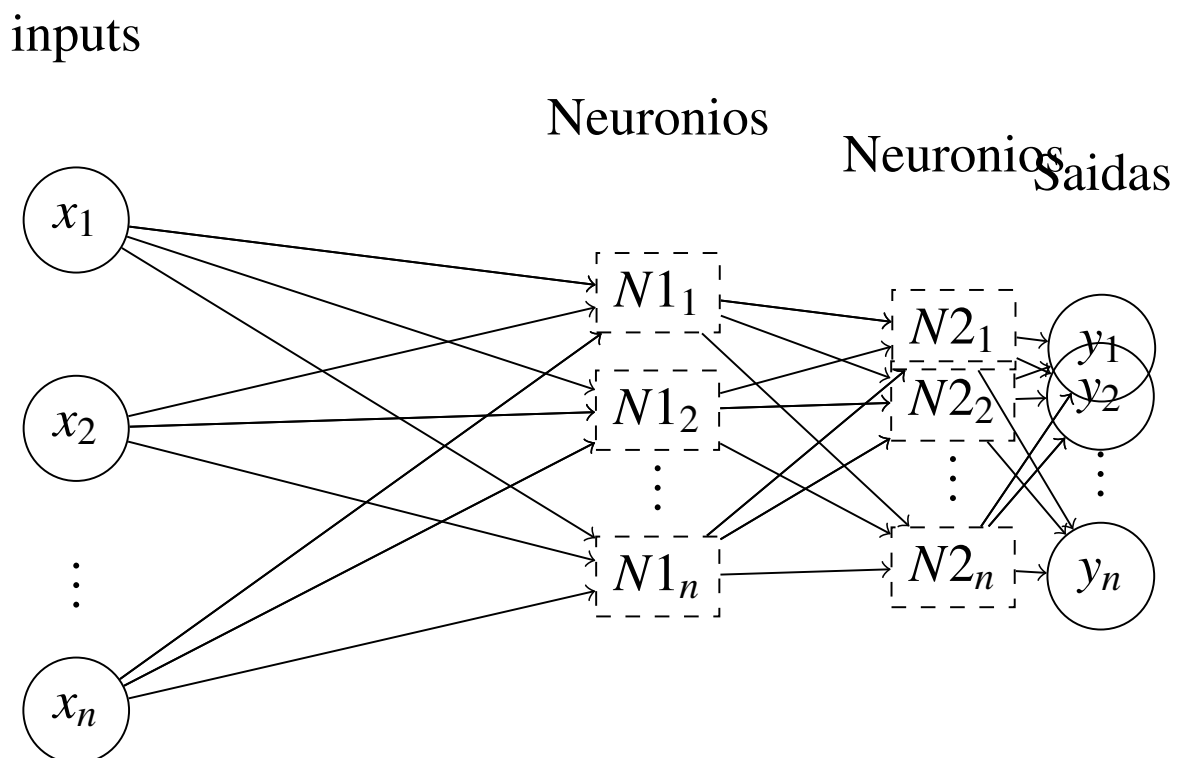


Figure 6.7: Ilustração do bloco de Dense

6.2.2 Bloco CNN

Bloco de CNN é aqui definido como uma convolução na dimensão temporal seguido de camadas para combater o overfitting, MaxPooling e Dropout.

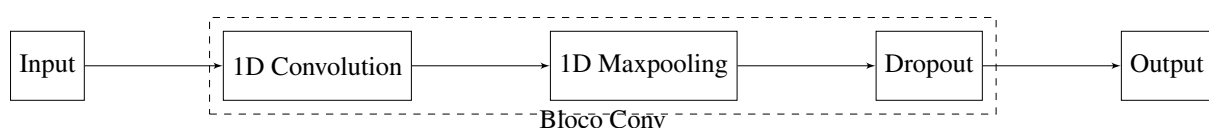


Figure 6.8: Ilustração do bloco de CNN

6.2.3 Bloco LSTM

O uso de LSTM para previsões é uma area comum, mas aqui é seguido através das ideias partilhadas em [22], e reforçado pelo uso em previsões energéticas demonstrados em [8]

O bloco LSTM é a aplicação das RNN, aqui sendo apenas definido como uma camada de LSTM.

Estes blocos mantêm dentro de si ligações a diferentes camadas temporais, e cada filtro criado, mantêm uma "memória" dos filtros passados.

Bastante utilizado em modelação de linguagem.

imagem

6.3 Arquiteturas

6.3.1 Vanilla

O termo "Vanilla" aqui é aplicado para arquiteturas que apenas usam um bloco de cada, um de entrada, um principal, e um interpretativo.

Como exemplo a arquitetura de "VanillaCNN" utiliza o bloco de convulsão apenas para terminar com uma camada interpretativa.

6.3.2 Stacked

Stacked refere-se a "amontoado" onde se utiliza o bloco principal várias vezes em série. E apenas um bloco de entrada e um interpretativo.

Como exemplo a arquitetura de "StackedCNN" utiliza dois blocos (ou mais) de convulsão depois da camada de entrada, e finaliza com uma camada interpretativa.

imagem da mesma

6.3.3 UNET

Normalmente usando em modelação de imagens, a arquitetura UNET passa por criar uma rede de expansão dos filtros, usando convulsões, e de seguida uma rede de contracção dos mesmos, até aos tamanhos pretendidos.

O bloco principal contextualmente o mesmo que o CNN.

Nas suas ligações UNET junta informação de filtros passados (não de nível temporal mas de rede neuronal) para realçar informação já trabalhada, e assim identificar padrões de vários contextos diferentes.

É habitual também adicionar aos blocos principais portões de atenção, portões residuais. Estas duas técnicas são também estudadas aqui.

É chamada assim pois é uma rede (NET) que forma um U na sua expansão e contracção.

Como exemplo a arquitetura de "UNET"

imagem da mesma

6.4 Considerações adicionais

Os modelos testados são combinações destes blocos e arquiteturas.

Ferramentas

7.1 Forecat

Com o propósito de desenvolver este estudo, e deixar ferramentas para a replicação do mesmo, foi criado uma biblioteca em python para desenhar as arquitecturas em estudo.

7.1.1 Construtor de modelos

Seguindo as arquitecturas descritas anteriormente esta ferramenta constrói os modelos automaticamente, sendo que precisamos apenas de fornecer os parâmetros variáveis.

O construtor assenta na ideia de três camadas abstratas de redes neuronais. A camada de entrada, a camada de bloco, e a camada interpretativa.

A camada de entrada recebe os dados e normaliza, podendo também fazer outras operações de preparação para a camada de bloco.

A camada de bloco é a camada descritiva da arquitetura, é a que tem as operações fundamentais.

A camada interpretativa é a que recebe o sinal de múltiplas redes neuronais internas, e traduz para o objectivo, usando Dense layers.

Esta abstracção segue sempre esta ordem. As variações dentro de cada arquitetura dependem das governamentalizações das mesmas, ou de cada camada, ou então da repetição do circuito, em paralelo ou em série. ou uma combinação destes.

7.1.2 Gerador de dados

O gerador construido trata da formatação dos dados para entrada nos modelos. Formatação esse que se baseia nos valores de janelas temporais a usar, e na divisão treino/teste.

Esta ferramenta agrega os dados em tensores de formato (N, t, a) , onde N é o número de casos, t é a janela temporal, e a é o número de atributos.

A ferramenta permite também definir o tempo de salto entre cada entrada.

Usando como exemplo uma janela temporal de 168 (horas, uma semana) para treino, e 24 (horas) para o alvo. Com um salto temporal de 1 a primeira entrada teria como treino as primeiras 168 horas dos dados, e como alvo as 24 horas consequentes. A segunda entrada seria a partir da segunda hora dos dados, e assim consecutivamente. Para um caso em que o tempo de salto seria 24, a primeira entrada mantinha-se, mas a segunda começaria 24 horas depois, e não apenas uma.

Como estamos também a lidar com dados desfasados, o gerador permite um TODO: shift em atributos a especificar. No caso em estudo temos que os atributos são de DA (day-ahead), logo estão desfasados 24 horas. O que implica termos de aplicar este shift nos dados que não são DA, nomeadamente os dados alvo. Esta propriedade permite também o fácil uso da ferramenta noutros dados desfasados, como as previsões a 3 ou 8 horas.

7.2 MuadDib

Esta ferramenta criada para desenvolver as experiências desta dissertação, permite ao utilizador apenas com os dados que quer utilizar e a especificação das metricas pretendidas, facilmente ter um modelo optimizado para os seus dados e problema.

Criada especificamente no ambito deste trabalho, está focada em criação de previsões, podendo no entanto ser utilizada para outros fins.

A configuração é bastante facil de usar, e tem em conta um publico não especializado. O que significa que qualquer utilizador com um conjunto de dados pode fazer previsões utilizando machine learning.

Métricas

As métricas utilizadas serviram maioritariamente dois propósitos, com valorizações distintas na escolha de melhores modelos.

O primeiro intuito é de estudo de cada modelo, utilizando as métricas comuns de regressão linear, comparando os valores reais com os valores das previsões.

Outro objectivo das métricas aplicadas e este mais relevante, era o estudo comparativo do desempenho de cada modelo com o modelo de benchmark.

t : Valor real. p : Previsão n : número de amostras

8.1 Métricas de modelo

RMSE - Root Mean Squared Error

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (t_i - p_i)^2} \quad (8.1)$$

Métrica comum em problemas de regressão, dando mais peso a erros maiores, mas retorna um valor que pode ser diretamente comparado ao valor em estudo. Neste caso podemos considerar que o RMSE representa o erro quadrático em MWh.

SAE - Sum Abs Error

$$SAE = \sum_{i=1}^n |t_i - p_i| \quad (8.2)$$

Este simboliza a soma absoluta de todos os erros, dentro da janela temporal em questão. Que representa a quantidade total da energia alocada/não alocada em erro, este é também a soma das duas próximas métricas.

AllocF - Alocação em Falta

$$AllocF = \begin{cases} 0 & , \text{se } p \geq t \\ t - p & , \text{se } p < t \end{cases} \quad (8.3)$$

Representa a soma total de toda a energia que faltou ser alocada.

AllocD - Alocação em Demasia

$$AllocD = \begin{cases} 0 & , \text{se } p \leq t \\ p - t & , \text{se } p > t \end{cases} \quad (8.4)$$

Representa a soma total de toda a energia que for alocada em demasia.

8.2 Métricas de comparação modelo/benchmark

GPD - Ganho Percentual de Desempenho

$$GPD = \frac{SAE_{benchmark} - SAE_{modelo}}{SAE_{benchmark}} \times 100 \quad (8.5)$$

O Ganho Percentual de Desempenho é a nossa métrica basilar. Representa, dentro da janela temporal de validação, a percentagem de melhoria do modelo em relação ao benchmark. Isto é representa a percentagem de energia que foi melhor alocada que o modelo. Onde 100% representa uma melhoria perfeita, onde o modelo não tem erro. E 0% representa nenhuma melhoria, ou seja, igual ao benchmark. Pode também ter valores negativos, que representa a percentagem em que o modelo é pior que o benchmark, podendo ser infinitamente pior.

Esta métrica é representativa da totalidade de energia, tanto alocado como em falta.

As próximas métricas são variações desta que ajudam a escolher o melhor modelo em cada experiência, conseguindo distinguir entre alocação em falta e em demasia.

GPDF - Ganho Percentual de Desempenho (alocação em) Falta

$$GPDF = \frac{AllocF_{benchmark} - AllocF_{modelo}}{AllocF_{benchmark}} \times 100 \quad (8.6)$$

O mesmo que o GPD mas apenas para as somas totais de alocação em falta.

GPDD - Ganho Percentual de Desempenho (alocação em) Demasia

$$GPD - D = \frac{AllocD_{benchmark} - AllocD_{modelo}}{AllocD_{benchmark}} \times 100 \quad (8.7)$$

O mesmo que o GPD mas apenas para as somas totais de alocação em falta.

GPD Norm - Ganho Percentual de Desempenho Normalizado

$$GPDNorm = \frac{GPDF + GPDD}{2} \quad (8.8)$$

Aqui o GPD é calculado a partir dos já calculados GPDF e GPDD, sendo a média destes. Desta maneira conseguimos ter uma percentagem de melhoria em relação ao benchmark, onde a melhoria da alocação em demasia e a melhoria da alocação em falta têm o mesmo peso.

GPD Norm² - Ganho Percentual de Desempenho Normalizado (negativos) Quadrado

$GPD\ Norm^2 = GPD$ norm mas os GPD são ao quadrado se forem negativos

$$GPDNorm^2 = \begin{cases} GPDNorm & , \text{se } GPDF \ \& \ GPDD \geq 0 \\ \frac{GPDF^2 + GPDD}{2} & , \text{se } GPDF < 0 \\ \frac{GPDF + GPDD^2}{2} & , \text{se } GPDD < 0 \end{cases} \quad (8.9)$$

O mesmo que GPD norm mas os GPDF ou GPDD que sejam negativos o seu valor é ao quadrado e mantendo-se negativo. Serve para dar mais peso aos valores negativos, assim não tendo GPD altos mesmo se um dos GPD for negativo (pior que o benchmark).

Esta métrica é a principal na escolha do melhor modelo em cada experiência visto manter ambos os GPD mas penalizando se algum deles é negativo.

GPD Positivo - Ganho Percentual de Desempenho Positivo

$$GPDPositivo = \begin{cases} GPD & , \text{se } GPDF \ \& \ GPDD \geq 0 \\ 0 & , \text{se } GPDF \ || \ GPDD < 0 \end{cases} \quad (8.10)$$

Esta métrica é igual a GPD mas apenas nos casos em que ambos são positivos, logo o modelo é melhor que o benchmark, senão é zero. Serve para medir o GPD real, mas apenas nos casos em que o modelo já supera o benchmark.

Métodos

A todas as experiências foram aplicadas as métricas apresentadas, excepto no benchmark onde teríamos métricas de comparação a ele próprio.

Para a escolha de melhor modelo dentro das experiências foi escolhido o modelo que melhor resultados apresentou para a métrica em questão. Sendo esta o GPD $Norm^2$ enquanto não encontramos uma melhoria tanto na alocação em falta como na em demasia, ao que se seguida foi usado o GPD Positivo.

No caso das Redes Neurais a hiper parametrização foi feita também deste modo, sendo que primeiro se encontrou uma hiper parametrização adequada aos dados, usando uma arquitetura simples e só depois com essa configuração foi feita a experiência nas várias arquiteturas.

O objectivo é conseguir um modelo que tenha a uma Alocação Total em Falta e Alocação Total em Demasia menor que o Benchmark, dentro dos anos 2019 a 2022. Logo em termos de métricas um GPD Positivo mais elevado possível.

9.1 Estatísticos

Em estatística conseguimos encontrar vários métodos de estudo de séries temporais. Estes métodos são normalmente usados como primeira abordagem para fazer previsões.

Estes modelos podem ser auto-regressivos (AR), onde vão fazer previsões baseados num número (p) de dados anteriores. Estes modelos são construídos com a noção de que um valor é linearmente dependente de p valores anteriores numa série temporal.

X_t : Valor no t a prever.	p : O número observações anteriores.
φ_i : Coeficiente na observação i .	q : O número observações anteriores.
ε_i : Erro na observação i .	θ_i : Coeficiente na observação i .
μ : Média dos valores X .	

AR - Auto-Regressivos

$$X_t = \sum_{i=1}^p \varphi_i X_{t-i} \quad (9.1)$$

Outra família destes modelos são os de média móvel (MA), onde a média de um número de observações (q) em conjunto com os erros (ε) e os coeficientes (θ) é usada para prever os valores seguintes.

MA - Média Móvel

$$X_t = \mu + \sum_{i=1}^q (\theta_i \varepsilon_{t-i}) + \varepsilon_t \quad (9.2)$$

Estes dois tipos de modelos podem ser juntos criando os modelos auto-regressivos de média móvel (ARMA), que junta as capacidades dos modelos anteriores.

MA - Média Móvel

$$X_t = \sum_{i=1}^p \varphi_i X_{t-i} + \mu + \sum_{i=1}^q (\theta_i \varepsilon_{t-i}) + \varepsilon_t \quad (9.3)$$

Existem mais modelos de previsão estatística baseados nestes com algumas variações, mas para este trabalho, e apenas como ponto de comparação às redes neurais, ficamos apenas por estes.

As variáveis em estudo por tipo de modelo foram retiradas das autocorrelações temporais usando os métodos de sugestão da ferramenta MuadDib:

	p	q
AR	1 / 2 / 23 / 24 / 25 / 48 / 144 / 168 / 192 / 336	NA
MA	NA	1 / 24
ARMA	1	1

Table 9.1: Variáveis de estudo dos modelos AR/MA

Todos estes modelos foram testados usando o software disponível na package de python [statsmodel](#), com a classe [ARIMA](#).

9.2 Redes Neurais

As redes neurais podem ser descritas como uma função desconhecida $f(x)=y$ onde durante o treino a função f é criada através da manipulação dos pesos da sua arquitetura usando os dados de treino, x , de forma a diminuir ao máximo uma função de perda. Sendo $f'(x)=y'$ um modelo já treinado onde y' é a previsão, a função de perda $fp(y, y')$ idealmente igual a 0, com $y'=y$.

Neste trabalho o x são todos os dados apresentados no capítulo Estudo 2, em grupos de 128 (horas), e o y é a energia usada, "UpwardUsedSecondaryReserveEnergy" no modelo de previsão de energia a subir e "DownwardUsedSecondaryReserveEnergy" no modelo de previsão de energia a descer, nas 24 horas subsequentes.

Assim utilizamos os 168 horas (1 semana) para prever as 24 horas seguintes. A fp é um dos factores de estudo, assim como outros parâmetros dentro das arquiteturas de modelos, f .

As condições em estudo são feitas através da ferramenta MuadDib, seguindo vários percursos entre as combinações possíveis, de modo a conseguir a combinação óptima, maior GPD Positivo.

9.2.1 Arquitecturas

FCNN, CNN, RNN são as arquiteturas mais simples que vamos estudar. Estas vão apenas pegar nos blocos e vamos criar as mesmas "Vanila" e "Stacked" com 2 blocos (ex: StackedCNN) ou 6 blocos (ex: Stacked6CNN).

UNET, LSTM são arquiteturas mais complexas e pesadas. Como descrito anteriormente uma mais utilizada em análise de imagens, e outra em análise de texto respectivamente.

Transformers são as mais pesadas qualidade comum da família de "generative AI".

9.2.2 Função de Perda

Nos primeiros testes mais simples foi imediato a discrepância entre os erros da energia alocada em demasia e em falta. Sendo que estes erros estão em dimensões completamente diferentes.

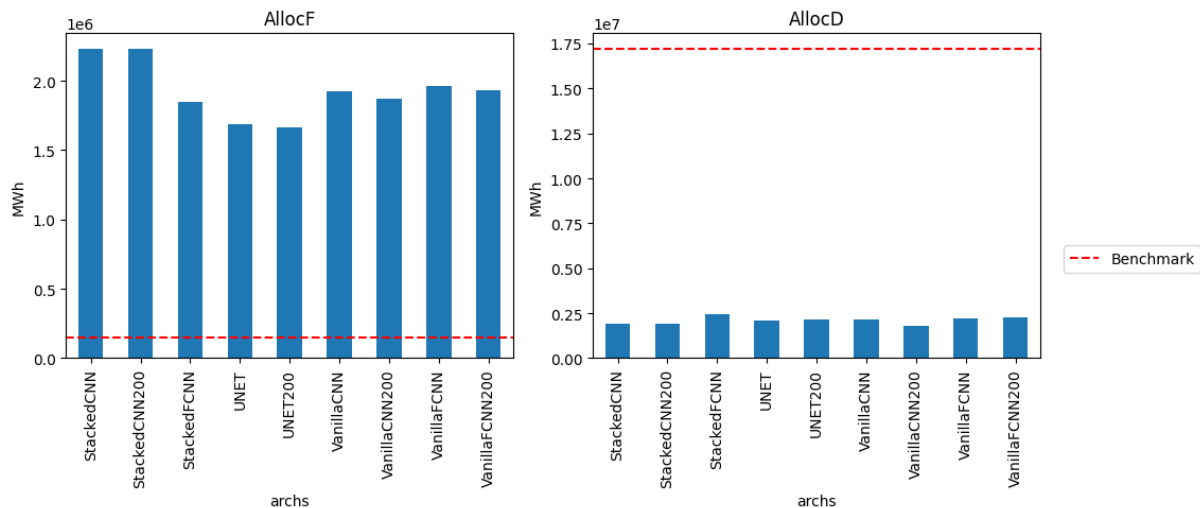


Figure 9.1: Resultados de alocações totais em diferentes arquiteturas

Na energia em falta, estamos a lidar com valores na dimensão de 10^6 nos resultados, sendo que o benchmark está nos 10^5 . Logo estão bastante acima do que queremos. Por outro lado na Energia em

Demasia temos resultados na ordem dos 10^6 e o benchmark está na ordem dos 10^7 . Isto dá-nos espaço para aumentar os resultados da Energia em Demasia mantendo-os ainda abaixo do benchmark para diminuir os resultados da Energia em Falta com objectivo de a ter também abaixo do benchmark.

Para combater esta desigualdade foram criadas várias funções de perda para atribuir melhor peso a ambas de modo a atingir melhor o objectivo geral.

De maneira que partimos esta experiência em duas partes. A primeira parte, Função de Perda Avançada, vai estudar diferentes maneiras de distribuir pesos entre a energia alocada em demasia e a em falta. A segunda vai escolher qual a melhor função de perda a aplicar nessa distribuição de pesos, ou vice-versa.

9.2.2.1 Funções de Perda

Depois de escolhidos os pesos nos diferentes grupos são testadas as funções a aplicar. Aqui vamos manter simples e testar apenas as mais comuns em problemas de regressão linear: Mean Absolute Error (MAE), Mean Squared Error (MSE), Mean Logarithmic Error (MLE).

MAE é usada no geral em problemas em que os dados têm um histograma linear, e um erro normalmente distribuído.

MSE é usado para atribuir maior peso aos erros maiores, do que no MAE. Fazendo com que o modelo se concentre mais em aprender a diminuir erros maiores.

MSLE é sugerido em dados que têm uma histograma exponencial.

9.2.2.2 Função de Perda Avançada

Para escolher a melhor maneira de distribuir pesos foi criada uma função de perda com diferentes regras, que distribuem o peso da amostra.

Mirror Weights (Pesos Espelhados)

Que vai distribuir os pesos da amostra consoante um rácio predefinido e o próprio erro da amostra.

Os pesos nas amostras vão ser divididos entre os erros negativos (alocação em demasia) e os positivos (alocação em falta). Consoante uma variável lógica, uns terão peso 1 e os outros serão o próprio erro em absoluto. Dando assim um peso equivalente ao erro, quanto maior o erro maior o peso da amostra na função de perda, do lado da amostra escolhido (em demasia ou em falta).

Pode ser multiplicado um rácio tanto a um dos pesos como a outro, sendo estes rácios que irão equilibrar as diferenças entre a alocação em falta e a em demasia. E o sinal do rácio influencia qual o lado a ser multiplicado.

Este pesos são passados directamente à função de perda em uso.

9. MÉTODOS

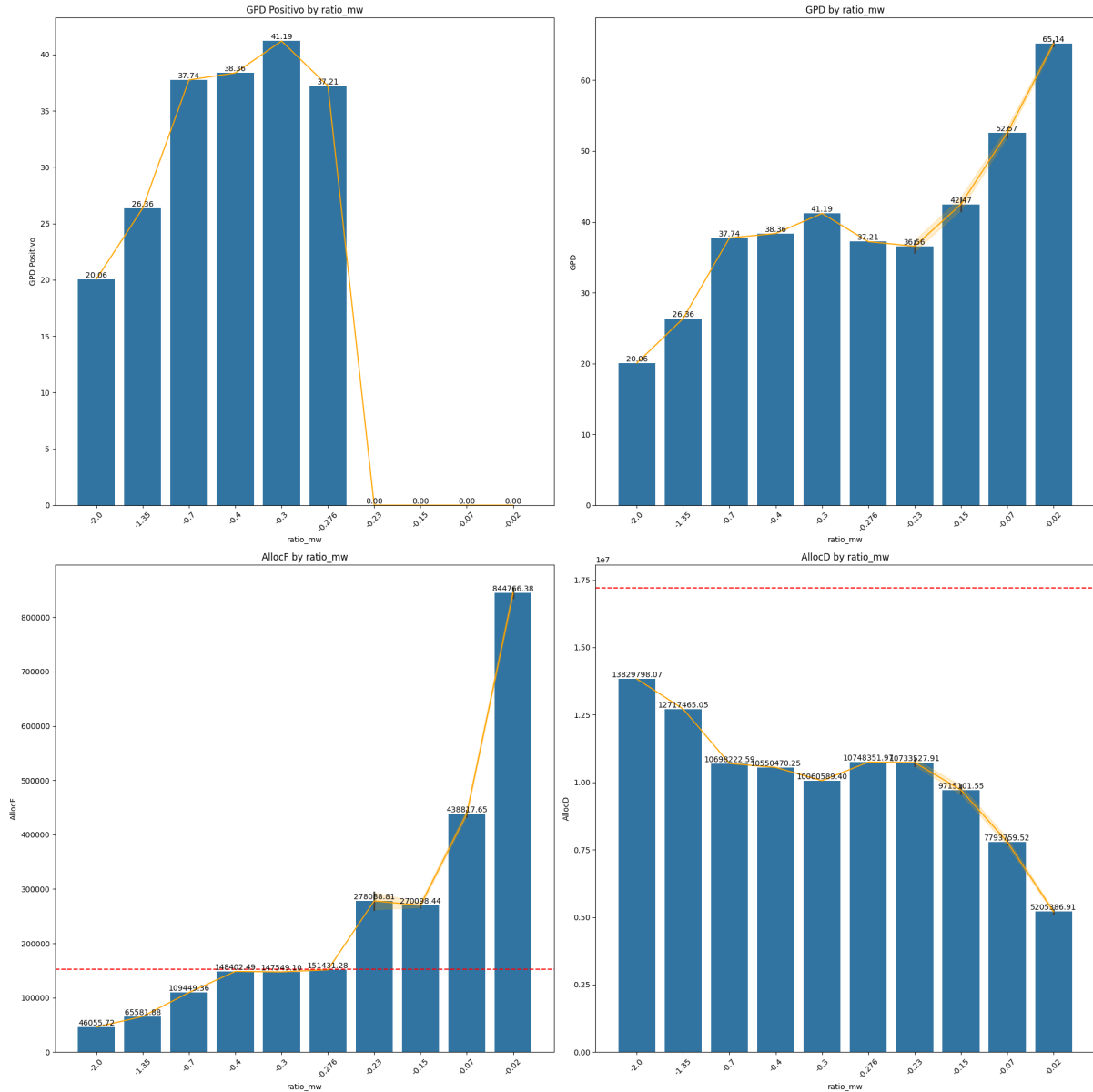


Figure 9.2: Resultados de alocações totais em diferentes raios

Estas variações no rácio produzem diferentes dimensões nas alocações, modificando assim a sua posição em relação ao benchmark. Aqui para cada arquitetura o rácio ideal para o melhor GPD Positivo diferencia ligeiramente, tendo sido procurado com tentativa/erro baseado em suposições perante a aparente distribuição rácio/alocações.

9.2.3 Função de Activação

Como mostrado em [23], e [24], o uso de uma activação mais apropriada aos dados pode ser crucial para um salto na qualidade do modelo.

Vamos dividir as função de activação usadas nas camadas intermédias e a usada na camada final. Isto porque as camadas intermédias tendem a funcionar melhor com a mesma activação e a final é que mais define o valor que sai do modelo.

Esta experiência vai testar a combinações das seguintes activações nas duas variáveis descritas anteriormente: linear, relu, gelu.

9.2.4 Pesos

Esta experiência serve para testar diferentes pesos por amostra, não por grupo como na experiência anterior. Aqui os pesos são aplicados no momento da função de perda final.

Normalmente é usado para dar mais pesos a amostras com menor amostragem. Mais facilmente aplicável em modelos de classificação. Com este é um problema de regressão linear com séries temporais vamos testar aplicar os seguintes pesos, ou nenhum peso.

Este peso é multiplicado peso peso em peso espelhados.

9.2.4.1 Temporais

Aqui a primeira amostra tem o menor valor de peso (1) e todas as amostras seguintes incrementam 1. Dando mais peso consecutivamente a amostras mais recentes. É testado em vários casos de séries temporais onde o objectivo é prever o futuro. Podendo assim dar mais peso a tendências e valores mais recentes.

9.2.4.2 Distância à média

Neste peso cada amostra tem como valor a sua distância à média total dos dados. Vai servir para o modelo conseguir criar pesos relevantes a valores mais distantes à média.

Logo as amostras que tenham picos de valores têm um peso maior, forçando o modelo a aprender melhor estas ocasiões.

9.3 Dados de Validação

Os dados de validação são os mesmos que os dados de treino, embora apenas durante os anos de 2019 a 2022, inclusive.

Usamos como benchmark as capacidades alocadas, "SecondaryReserveAllocationAUpward" e "SecondaryReserveAllocationADownward", e como validação e objectivo, y, a própria energia usada, "UpwardUsedSecondaryReserveEnergy" e "DownwardUsedSecondaryReserveEnergy".

9.4 Benchmark

Como método de comparação a todas as experiências foi criado uma base que servirá de benchmark.

Este base não é nada mais do que a própria previsão feita pela entidade reguladora ESIOS. Dentro do nossos dados são os valores nos campos "SecondaryReserveAllocationAUpward" e "SecondaryReserveAllocationADownward".

Para os dados utilizados, podemos ver a totalidade e comparação do benchmark (Energia Alocada) com a energia utilizada.

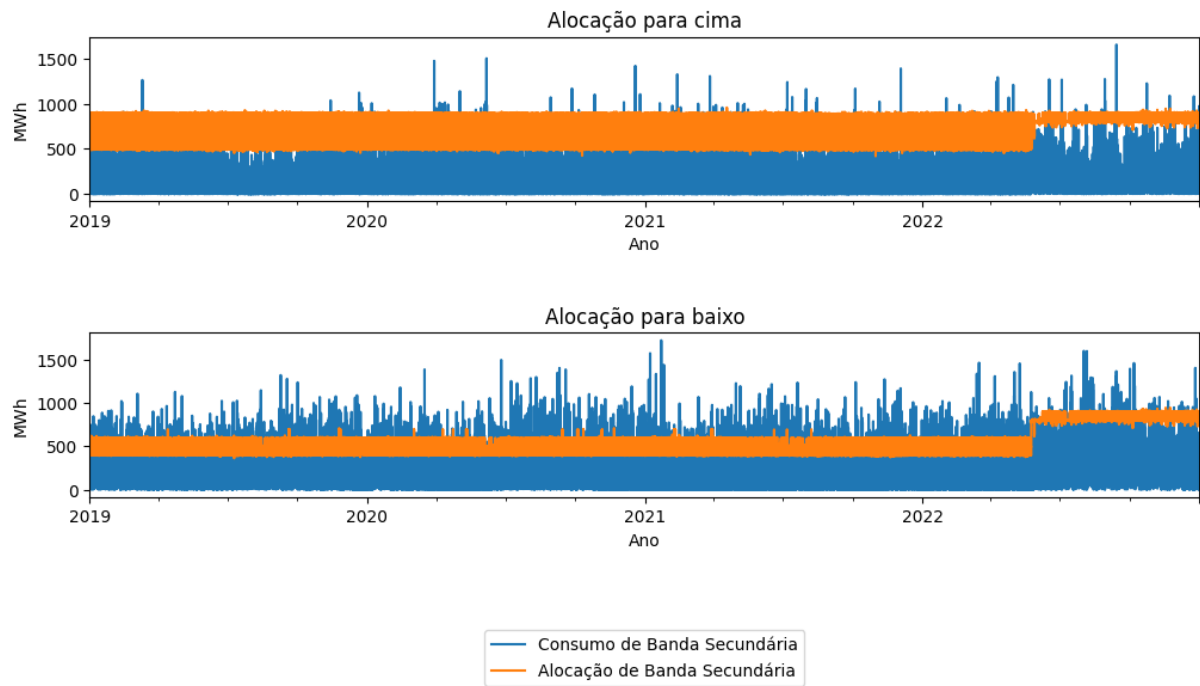


Figure 9.3: Série Temporal dos dados de Benchmark c/ consumo real

Imediatamente podemos verificar que o método para prever a energia necessária actualmente está dentro de um espectro limitado de valores, sendo que esses valores estão perto dos valores de ponta na alocação para cima, e perto dos valores médios na alocação para baixo.

Isto deve-se ao facto de ser uma função fixa, baseado no dia em questão. Notamos também que a meio de 2022 houve uma mudança dessa função que limitou os alcances tornando os valores mais elevados. Devido à guerra na Ucrânia e à forte incerteza que esta trouxe aos mercados de eletricidade por causa da crise de gás na Europa, que aumentou significativamente o preço deste recurso e levou à adaptação dos consumidores e países, a Red Eléctrica de España (REE) aumentou as necessidades de reserva secundária para responder a esta incerteza.

Do ponto de vista de dados faz sentido para diminuir a quantidade de vezes em que não é alocada energia suficiente.

Mas o mais importante a notar é a forma estática destes métodos, dado a natureza flutuante dos da energia necessária este método apresenta frequentemente um erro grande.

Podemos ver em pormenor analisando algumas janelas temporais dentro do período de validação. Vendo o melhor e pior resultado, em termos de erro absoluto, em janelas temporais de ano, mês, semana e dia.

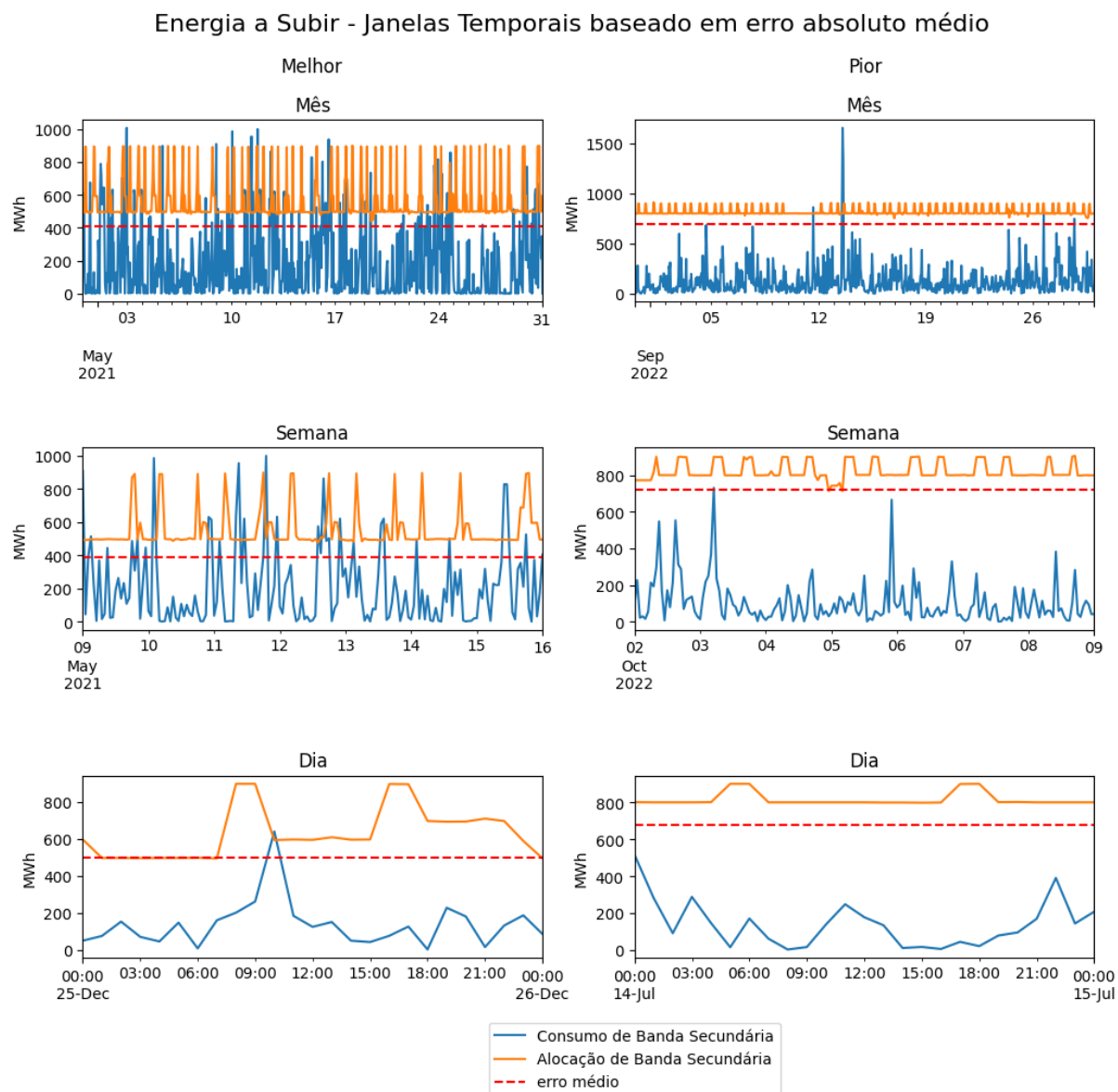


Figure 9.4: Janelas temporais de benchmark energia a subir

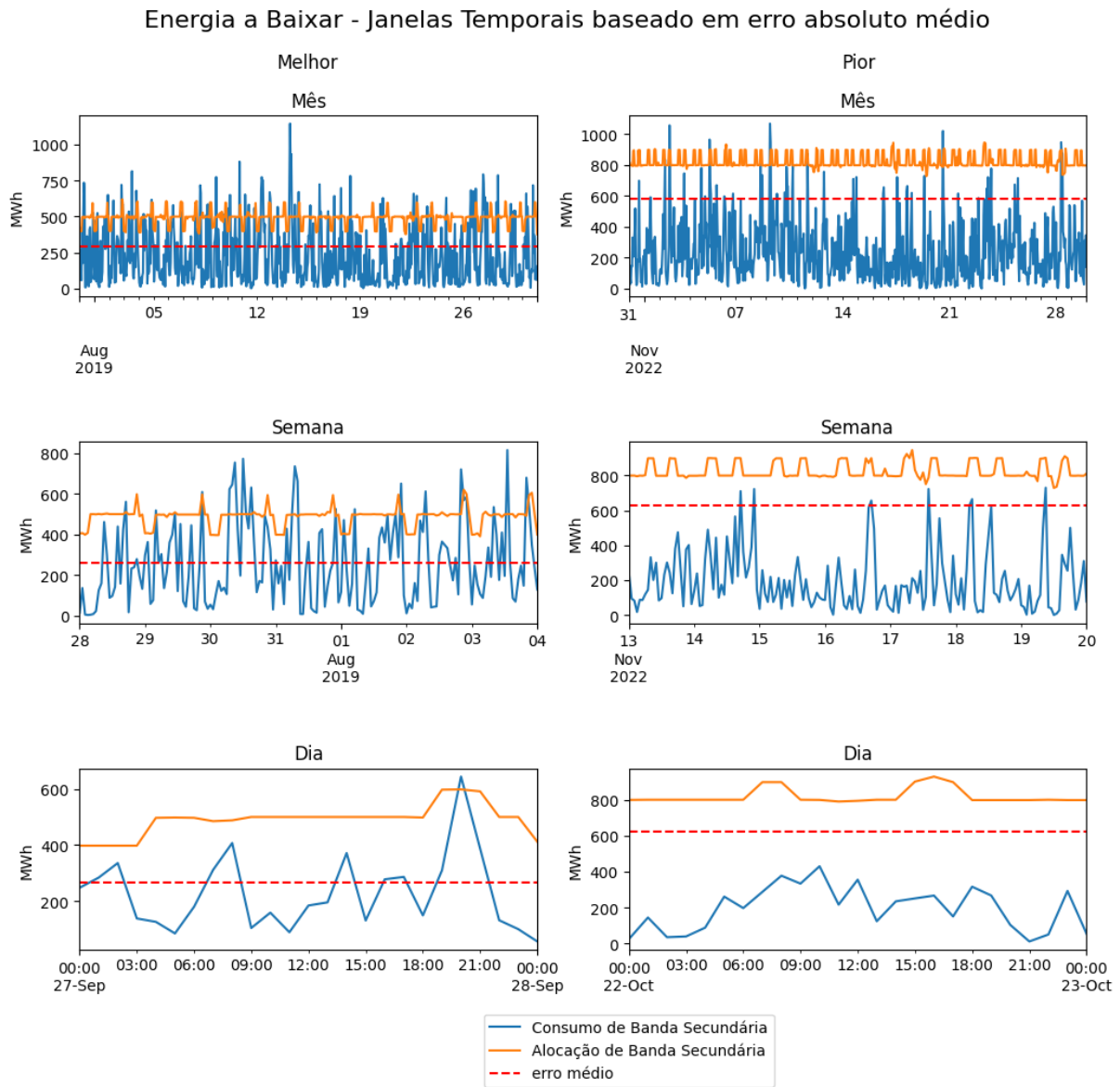


Figure 9.5: Janelas temporais de benchmark energia a descer

Dentro destas janelas temporais conseguimos ter melhor a percepção da natureza estática deste modelo actual, e qual longe está dos valores reais necessários.

Os resultados a queres melhorar são:

Table 9.2: Resultados métricas benchmark

	RMSE	SAE	AllocF	AllocD
Alocação a Subir (MW)	536.55	17357826.75	152679.00	17205147.75
Alocação a Descer (MW)	408.99	12981575.55	479191.60	12502383.95

Conclusões e sugestões futuras

Primeiramente podemos ver pela análise estatística?? e aplicando a ideia[11], simples modelos estatísticos conseguiriam resolver o problema em mãos, melhor do que o que é utilizado actualmente??, e melhor do que muitos dos modelos profundos que testamos.

E se considerarmos ainda que os modelos estatísticos apresentados que apresentam estes resultados, utilizam apenas a variável em questão, e não todos os outros atributos, a nível de aplicabilidade já são um melhoria.

Com modelos relativamente simples conseguimos melhorias muito grandes na alocação de energia, em relação à alocada actualmente. Este métodos podem criar ganhos financeiros grandes.

Aqui são dadas as respostas às perguntas de investigação formuladas na secção 1.2. Não fazer aqui a discussão dos resultados. Essa discussão deve ser feita no capítulo 9.4. Não esquecer de indicar sugestões futuras para que um colega possa dar continuidade ao trabalho desenvolvido.

sintaxe executiva:

Os modelos de redes neuronais para previsões são um campo já bastante explorado. As várias gerações de arquiteturas sempre tentaram resolver este problema.

Como podemos verificar neste estudo esses métodos são mais eficazes que a técnica actualmente aplicada para a previsão de alocação secundária.

Uma primeira análise apenas linear já consegue resultados mais lucrativos que o modelo aplicado, então o poder das redes neuronais consegue ampliar essa capacidade. Contudo a escolha e aplicação das mesmas deve estar fortemente ligado ao tipo de estrutura que se pretende construir para a mesma. Em casos de grande capacidade computacional e previsões recorrentes os modelos de arquiteturas mais recente, e os modelos mais complexos, como UNET e a família de Transformers, são os que mostram mais fiabilidade. No entanto se estes cálculos só podem ser efectuados em máquinas menos poderosas, pode-se ter de ficar pelos arquiteturas mais simples, e mais antigas (dar exemplos).

Independentemente do caso específico, é bastante possível melhorar o modelo actual de previsão usando redes neuronais, sendo que estas vão manter ainda a capacidade de ir aprendendo com dados novos, dando ainda mais robustez e dinamismo às previsões.

Embora os resultados sejam positivos do ponto de vista do problema a procurar ser resolvido, penso que poderia ser atacado de maneira mais eficaz. Os primeiros passos para um modelo mais robusto seria a escolha de dados de treino. Neste estudo usamos dados de outras previsões, dados esse que são eles próprios artificiais na sua maioria. Isto não só adiciona mais uma camada de abstracção, como representa uma perda na qualidade de informação. Ao invés destes dados de previsões um primeiro passo seria criar um modelo apenas com dados reais de consumo e produção. Em adição a isto os principais dados a utilizar numa procura de melhor modelo seriam os de outras reservas, primária e terciária. Visto estas estarem intrinsecamente ligadas com a reserva secundária, deduzo que a informação contida nestes seria bem mais eficaz na produção de modelos de previsão de reserva de energia secundária.

Outra recomendação a ter em mente para futuros modelos requer já um conhecimento destes sistemas superior, e a criação de camadas neuronais com restrições na modelação das variáveis. Este tipo de camadas vai permitir modelos muito mais robustos, visto não aprenderem fora do contexto necessário, e também modelos muito mais leves, o que irá permitir uma adopção dos mesmos mais eficaz.

Referências

- [1] Hum Nath Bhandari et al. “Predicting stock market index using LSTM”. In: *Machine Learning with Applications* 9 (Sept. 2022), p. 100320. ISSN: 2666-8270. DOI: [10.1016/J.MLWA.2022.100320](https://doi.org/10.1016/J.MLWA.2022.100320).
- [2] Bin Wang et al. “Deep uncertainty quantification: A machine learning approach for weather forecasting”. In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (July 2019), pp. 2087–2095. DOI: [10.1145/3292500.3330704](https://doi.org/10.1145/3292500.3330704). URL: <https://dl.acm.org/doi/10.1145/3292500.3330704>.
- [3] Siyuan Lu et al. “Machine learning based multi-physical-model blending for enhancing renewable energy forecast - Improvement via situation dependent error correction”. In: *2015 European Control Conference, ECC 2015* (Nov. 2015), pp. 283–290. DOI: [10.1109/ECC.2015.7330558](https://doi.org/10.1109/ECC.2015.7330558).
- [4] Haibo Sun, Li Li, and Haonan Wang. “Short-term Wind Power Forecasting Model Based on Stacking Fusion Learning”. In: *IEEE Joint International Information Technology and Artificial Intelligence Conference (ITAIC) 2022-June* (2022), pp. 53–57. ISSN: 26932865. DOI: [10.1109/ITAIC54216.2022.9836510](https://doi.org/10.1109/ITAIC54216.2022.9836510).
- [5] T. Rajasundrapandiyanleebanon et al. “Solar Energy Forecasting Using Machine Learning and Deep Learning Techniques”. In: *Archives of Computational Methods in Engineering* 2023 30:5 30 (5 Feb. 2023), pp. 3059–3079. ISSN: 1886-1784. DOI: [10.1007/S11831-023-09893-1](https://doi.org/10.1007/S11831-023-09893-1). URL: <https://link.springer.com/article/10.1007/s11831-023-09893-1>.
- [6] Tanveer Ahmad and Huanxin Chen. “A review on machine learning forecasting growth trends and their real-time applications in different energy systems”. In: *Sustainable Cities and Society* 54 (Mar. 2020), p. 102010. ISSN: 2210-6707. DOI: [10.1016/J.SCS.2019.102010](https://doi.org/10.1016/J.SCS.2019.102010).
- [7] Ioannis Antonopoulos et al. “Artificial intelligence and machine learning approaches to energy demand-side response: A systematic review”. In: *Renewable and Sustainable Energy Reviews* 130 (Sept. 2020), p. 109899. ISSN: 1364-0321. DOI: [10.1016/J.RSER.2020.109899](https://doi.org/10.1016/J.RSER.2020.109899).
- [8] Rogério Luís de C. Costa. “Convolutional-LSTM networks and generalization in forecasting of household photovoltaic generation”. In: *Engineering Applications of Artificial Intelligence* 116 (Nov. 2022), p. 105458. ISSN: 0952-1976. DOI: [10.1016/J.ENGAPPAI.2022.105458](https://doi.org/10.1016/J.ENGAPPAI.2022.105458).
- [9] R. Meenal and A. Immanuel Selvakumar. “Assessment of SVM, empirical and ANN based solar radiation prediction models with most influencing input parameters”. In: *Renewable Energy* 121 (June 2018), pp. 324–343. ISSN: 0960-1481. DOI: [10.1016/J.RENENE.2017.12.005](https://doi.org/10.1016/J.RENENE.2017.12.005).
- [10] Natei Ermias Benti, Mesfin Diro Chaka, and Addisu Gezahegn Semie. “Forecasting Renewable Energy Generation with Machine Learning and Deep Learning: Current Advances and Future Prospects”. In: *Sustainability (Switzerland)* 15 (9 May 2023). ISSN: 20711050. DOI: [10.3390/SU15097087](https://doi.org/10.3390/SU15097087).
- [11] Shereen Elsayed et al. “Do We Really Need Deep Learning Models for Time Series Forecasting?” In: ().

- [12] Javier Cardo-Miota, Emilio Pérez, and Hector Beltran. “Deep learning-based forecasting of the automatic Frequency Reserve Restoration band price in the Iberian electricity market”. In: *Sustainable Energy, Grids and Networks* 35 (Sept. 2023), p. 101110. ISSN: 2352-4677. DOI: [10.1016/J.SEGAN.2023.101110](https://doi.org/10.1016/J.SEGAN.2023.101110). URL: <https://linkinghub.elsevier.com/retrieve/pii/S2352467723001182>.
- [13] Hussein Ashraf Rassid, Orientador : Fernando, and Jorge Ferreira Lopes. *Mercado de Reservas: Simulador Multiagente SIMREEL e Caso de Estudo Envolvendo o Mercado Ibérico*. 2017. URL: <https://run.unl.pt/handle/10362/30815>.
- [14] Auteur : Stassen, Théo Promoteur, and Damien Ernst. *Master’s Thesis : Comparison of probabilistic forecasting deep learning models in the context of renewable energy production*. URL: <https://lib.uliege.be><https://matheo.uliege.be>.
- [15] *Mercado de Serviço de Sistemas*. URL: <https://www.dgeg.gov.pt/pt/areas-transversais/mercados-e-mecanismos-de-capacidade/outros-mercados/mercado-de-servico-de-sistemas/>.
- [16] Célia Carneiro. “Mecanismos de funcionamento do Mercado de Serviços de Sistema - aplicação ao MIBEL”. In: (Nov. 2016). URL: <https://core.ac.uk/download/pdf/302870251.pdf>.
- [17] Stef van Buuren and Karin Groothuis-Oudshoorn. “mice: Multivariate Imputation by Chained Equations in R”. In: *Journal of Statistical Software* 45 (3 Dec. 2011), pp. 1–67. ISSN: 1548-7660. DOI: [10.18637/JSS.V045.I03](https://doi.org/10.18637/JSS.V045.I03). URL: <https://www.jstatsoft.org/index.php/jss/article/view/v045i03>.
- [18] S F Buck. “A Method of Estimation of Missing Values in Multivariate Data Suitable for use with an Electronic Computer”. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 22 (2 1960), pp. 302–306. ISSN: 00359246. URL: <http://www.jstor.org/stable/2984099>.
- [19] F. Rosenblatt. “The perceptron: A probabilistic model for information storage and organization in the brain”. In: *Psychological Review* 65 (6 Nov. 1958), pp. 386–408. ISSN: 0033295X. DOI: [10.1037/H0042519](https://doi.org/10.1037/H0042519).
- [20] Kyle Matoba et al. “The Theoretical Expressiveness of Maxpooling”. In: (2022). URL: <https://pytorch.org/vision/stable/models.html..>
- [21] V L Helen Josephine, A.P. Nirmala, and Vijaya Lakshmi Alluri. “Impact of Hidden Dense Layers in Convolutional Neural Network to enhance Performance of Classification Model”. In: *IOP Conference Series: Materials Science and Engineering* 1131 (1 Apr. 2021), p. 012007. ISSN: 1757-8981. DOI: [10.1088/1757-899X/1131/1/012007](https://doi.org/10.1088/1757-899X/1131/1/012007).
- [22] Hansika Hewamalage, Christoph Bergmeir, and Kasun Bandara. “Recurrent Neural Networks for Time Series Forecasting: Current status and future directions”. In: *International Journal of Forecasting* 37 (1 Jan. 2021), pp. 388–427. ISSN: 01692070. DOI: [10.1016/j.ijforecast.2020.06.008](https://doi.org/10.1016/j.ijforecast.2020.06.008).
- [23] Ashish Vaswani et al. “Attention Is All You Need”. In: *Advances in Neural Information Processing Systems* 2017-December (June 2017), pp. 5999–6009. ISSN: 10495258. URL: <https://arxiv.org/abs/1706.03762v7>.

REFERÊNCIAS

- [24] Zhuang Liu et al. “A ConvNet for the 2020s”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2022-June* (Jan. 2022), pp. 11966–11976. ISSN: 10636919. DOI: [10.1109/CVPR52688.2022.01167](https://doi.org/10.1109/CVPR52688.2022.01167). URL: <https://arxiv.org/abs/2201.03545v2>.

Anexos

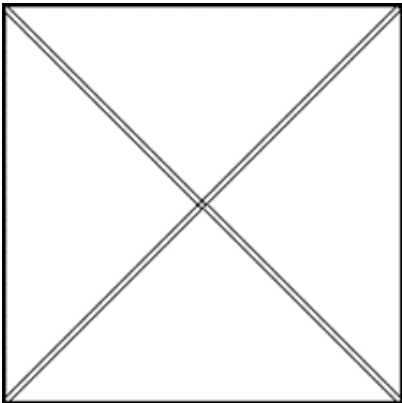


Figure A.1: Exemplo de como considerar um gráfico nos anexos.

Table A.1: Isto é um exemplo de uma tabela. Se fôr igual(copiada) a outro autor deve ser pedido autorização para reproduzir.

Title 1	Title 2	Title 3	Title 4
entry 1	data	data	data
	data	data	data
	data	data	data
entry 2	data	data	data
	data	data	data
entry 3	data	data	data
	data	data	data
	data	data	data
	data	data	data
entry 4	data	data	data