

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 crescente penetração das fontes de energia renovável nos mercados de produção energética, aliada à mudança de paradigma nos tamanhos e centralização dos produtores, trás instabilidades e variabilidade não coompreendidas nos modelos actuais de mercado.

Por sua vez, devido à sua natureza menos constante, estes produtores têm maiores dificuldades em conseguir lucros na situação corrente.

As tecnologias de previsão estão a migrar cada vez mais para modelos de *machine learning*, com resultados mais precisos.

Este trabalho propõe a aplicação de tecnologias de redes neuronais para previsão das bandas de alocação de reserva secundária, usando os dados do mercado ibérico.

**Palavras chave:** sistemas de reserva, mercados de energia, redes neurais, previsões

# Abstract

The growing penetration of renewable energy sources, in the energy production markets, together with the paradigm shift in size and centralization of producers, brings instabilities and variability not in the scope of current market models.

In turn, due to their less constant nature, these renewable producers have greater difficulties in making a profit in current models.

Forecast technologies are increasingly migrating to *machine learning* models, with more accurate results.

This work proposes the application of neural network technologies to forecast secondary reserve allocation bands, using data from the Iberian market.

**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 de acordo com 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
$\eta$	Eficiência
p	Pressão
T	Temperatura

# Índice

<b>Resumo</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>Agradecimentos</b>	<b>iii</b>
<b>Nomenclatura</b>	<b>iv</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>ix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Enquadramento . . . . .	1
1.2 Objetivos e perguntas de investigação . . . . .	2
1.3 Organização do documento . . . . .	2
<b>2 Revisão bibliográfica</b>	<b>3</b>
<b>3 Contextos</b>	<b>4</b>
3.1 Mercado de Serviços de Sistema . . . . .	4
3.2 MIBEL . . . . .	4
<b>4 Dados</b>	<b>5</b>
4.1 Dados Utilizados . . . . .	5
4.1.1 Aquisição dos Dados . . . . .	5
4.2 Estudo dos dados . . . . .	5
4.2.1 Correlações . . . . .	7
4.2.1.1 Correlações entre atributos . . . . .	7
4.2.1.2 Correlações Temporais . . . . .	9
4.2.2 Agrupamento . . . . .	10
4.3 Tratamento dos dados . . . . .	12
4.4 Considerações adicionais . . . . .	14
<b>5 Architecturas de Modelos</b>	<b>15</b>
5.1 Blocos . . . . .	15
5.1.1 Bloco Dense . . . . .	15
5.1.2 Blcoo CNN . . . . .	16
5.1.3 Blcoo LSTM . . . . .	16

5.2	Arquiteturas . . . . .	16
5.2.1	Vanilla . . . . .	16
5.2.2	Stacked . . . . .	16
5.2.3	MultiHead . . . . .	16
5.2.4	MultiTail . . . . .	17
5.2.5	UNET . . . . .	17
5.3	Considerações adicionais . . . . .	17
<b>6</b>	<b>Métodos</b>	<b>18</b>
6.1	Benchmark . . . . .	18
6.2	Modelos estatísticos . . . . .	19
6.2.1	Univariate Analysis . . . . .	19
6.2.1.1	MA . . . . .	19
6.2.2	ARMA . . . . .	20
6.2.3	ARIMA . . . . .	21
6.2.4	SARIMA . . . . .	21
6.3	Forecat . . . . .	22
6.3.1	Construtor de modelos . . . . .	22
6.3.2	Gerador de dados . . . . .	23
6.4	Treino e Resultados . . . . .	23
6.4.1	Arquiteturas e numeros de epocas . . . . .	23
6.4.2	Funções de Perda (Loss) . . . . .	24
6.4.3	Hiperparametrização . . . . .	24
6.4.3.1	Activação . . . . .	24
6.4.3.2	Optimizadores . . . . .	24
6.4.4	Janelas Temporais . . . . .	24
6.4.5	Classificação . . . . .	25
6.4.6	Pesos . . . . .	25
6.4.6.1	Modelos lineares . . . . .	25
6.4.6.2	Modelos Lineares e de Classificação . . . . .	25
6.5	Considerações adicionais . . . . .	25
<b>7</b>	<b>Resultados e discussão</b>	<b>26</b>
7.1	Métricas . . . . .	26
7.2	Experiências . . . . .	26
7.2.1	Arquiteturas e numeros de epocas . . . . .	27
7.2.1.1	Arquiteturas . . . . .	27
7.2.1.2	Epocas . . . . .	27
7.2.2	Funções de Perda (Loss) . . . . .	27
7.2.3	Hiperparametrização . . . . .	28
7.2.3.1	Activação . . . . .	28
7.2.3.2	Optimização . . . . .	28
7.2.4	Janelas Temporais . . . . .	28
7.2.5	Classificação . . . . .	29
7.2.6	Pesos . . . . .	29



<b>8</b>	<b>Conclusões e sugestões futuras</b>	<b>30</b>
<b>9</b>	<b>Referências</b>	<b>31</b>
<b>A</b>	<b>Anexos</b>	<b>33</b>

# List of Figures

<b>Figura 1.1</b> Objetivos desenvolvimento sustentável ONU . . . . .	1
<b>Figura 4.1</b> Serie Temporal dos dados alvo . . . . .	6
<b>Figura 4.2</b> Janelas Temporais dos dados alvo . . . . .	6
<b>Figura 4.3</b> Frequência dos dados alvos . . . . .	7
<b>Figura 4.4</b> Correlação entre atributos . . . . .	8
<b>Figura 4.5</b> Valores de correlação entre atributos . . . . .	9
<b>Figura 4.6</b> Serie Temporal dos dados alvo . . . . .	9
<b>Figura 4.7</b> Serie Temporal dos dados alvo . . . . .	10
<b>Figura 4.8</b> Serie Temporal dos dados alvo . . . . .	11
<b>Figura 4.9</b> Histograma das classes . . . . .	12
<b>Figura 4.10</b> Outliers . . . . .	13
<b>Figura 4.11</b> Dados em falta . . . . .	14
<b>Figura 5.1</b> Bloco Densa . . . . .	15
<b>Figura 6.1</b> Serie Temporal do benchmark . . . . .	18
<b>Figura 6.2</b> Serie Temporal do benchmark 2021 . . . . .	19
<b>Figura 6.3</b> Previsões 2021 com modelo MA . . . . .	20
<b>Figura 6.4</b> Previsões 2021 com modelo ARMA . . . . .	20
<b>Figura 6.5</b> Previsões 2021 com modelo ARIMA . . . . .	21
<b>Figura 6.6</b> Previsões 2021 com modelo SARIMA . . . . .	22
<b>Figura 7.1</b> Exemplo de como considerar um gráfico . . . . .	29
<b>Figura A.1</b> Exemplo de como considerar um gráfico nos anexos. . . . .	33

# List of Tables

<b>Tabela 6.1</b> Dados Benchmark . . . . .	18
<b>Tabela 6.2</b> Dados Benchmark de validação . . . . .	19
<b>Tabela 7.1</b> Resultados Arquitecturas . . . . .	27
<b>Tabela 7.2</b> Resultados Funções de perda . . . . .	27
<b>Tabela 7.3</b> Resultados Ativação . . . . .	28
<b>Tabela 7.4</b> Resultados Janelas Temporais . . . . .	28
<b>Tabela A.1</b> Isto é um exemplo de uma tabela. Se fôr igual(copiada) a outro autor deve ser pedido autorização para reproduzir. . . . .	33



# Introdução

## 1.1 Enquadramento

Esta dissertação encontra-se no âmbito do projecto [TradeRES](#), o qual estuda um sistema de mercado eléctricos que consiga dar resposta às necessidades da sociedade num sistema quase todo 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 nº7, “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 nº8 “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 nº13, “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: Objetivos desenvolvimento sustentável ONU

### 1.2 Objetivos e perguntas de investigação

Foram aprovadas a nível europeu (2020), medidas de alteração aos serviços de sistema, que serão seguidas pelos Estados-Membros. Nesta dissertação far-se-á a aplicação dessas medidas, identificando as melhorias face ao desenho actual e, avaliando se as novas medidas serão suficientes para assegurar a operação de um sistema eléctrico 100% renovável, eventualmente identificando acções adicionais que garantam a robustez e segurança do sistema eléctrico sem recurso a combustíveis fósseis.

- 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?

Como forma de responder a estas questões vamos tentar utilizar dados de previsão de geradores renováveis para prever a energia necessária para alocação secundária.

Os valores de previsão deste mercado estão actualmente longe do que realmente é utilizado, fazendo com que a alocação no dia anterior não seja em conformidade com o necessário, logo não existindo o optimo aproveitamento de recursos.

O objectivo neste trabalho é perceber se utilizando técnicas de *Machine Learning* (Aprendizagem automática) conseguimos obter previsões mais próximas da energia utilizando, possibilitando assim uma melhor gestão das alocações, logo um menor gasto de recursos, energéticos mas especialmente financeiros.

### 1.3 Organização do documento

Este documento está dividido em capítulos. Sendo que os primeiros apresentam uma introdução às ideias e temas 1, o estado de arte do temas na literatura publicada 2, e por fim uma contextualização dos temas abordados 3.2.

Segue um capítulo de explicação dos dados usado 4.4, 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 arquitecturas 5.3 utilizadas, incluindo uma explicação dos componentes das mesmas.

Os capítulos 6 e 7 são bastante paralelos, sendo que o sexto 6.5 apresenta a metodologia do trabalho, e explica todas as experiências, e o sétimo 7.2.6 apresenta apenas os resultados e conclusões da experiência.

Termina com um capítulo conclusivo 8 onde são avaliadas as experiências como um todo, e o seu impacto no âmbito dos mercados de reserva.

# Revisão bibliográfica

reserve allocation forecasting

A alocação de reservas

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 e muitos casos resultados melhores do que usados na indústria corrente. [1] [2]

Como muitos trabalhos apresentados em [3], o uso destas técnicas está a crescer e a produzir frutos. Como concluído nesse trabalho as várias arquitecturas 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

ESTUDOS EM MERCADOS DE RESERVAS [4]

# Contextos

## 3.1 Mercado de Serviços de Sistema

[5] [6] [7]

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

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 [4] [8].

Neste caso de estudo estamos interessados nos serviços de controlo de frequência. A nível europeu estes serviços são impostos pela ENTO-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 prevêem 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.

## 3.2 MIBEL

[9] [8] [10] [11]



# Dados

## 4.1 Dados Utilizados

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

indicators	names	units
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

### 4.1.1 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](#):

## 4.2 Estudo dos dados

Os dados que propunho a prever são: "UpwardUsedSecondaryReserveEnergy",  
"DownwardUsedSecondaryReserveEnergy"

#### 4. DADOS

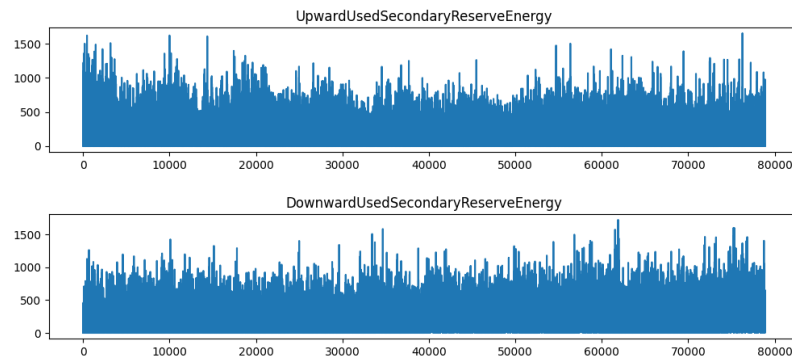


Figure 4.1: Serie Temporal dos dados alvo

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

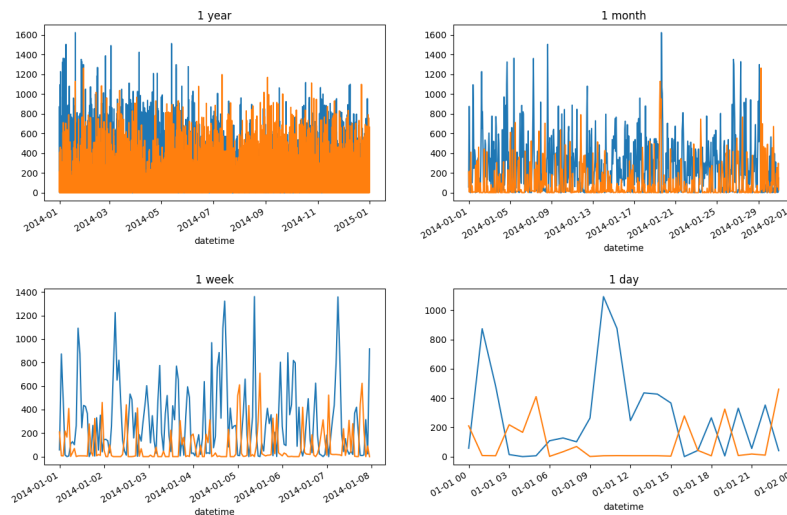


Figure 4.2: Janelas Temporais dos dados alvo

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 parametros no modelação

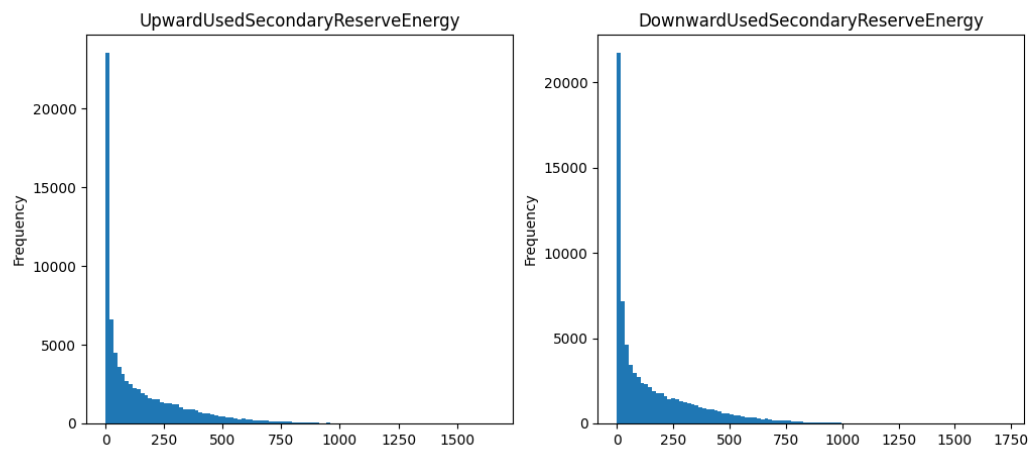


Figure 4.3: Frequência dos dados alvos

## 4.2.1 Correlações

### 4.2.1.1 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.

#### 4. DADOS

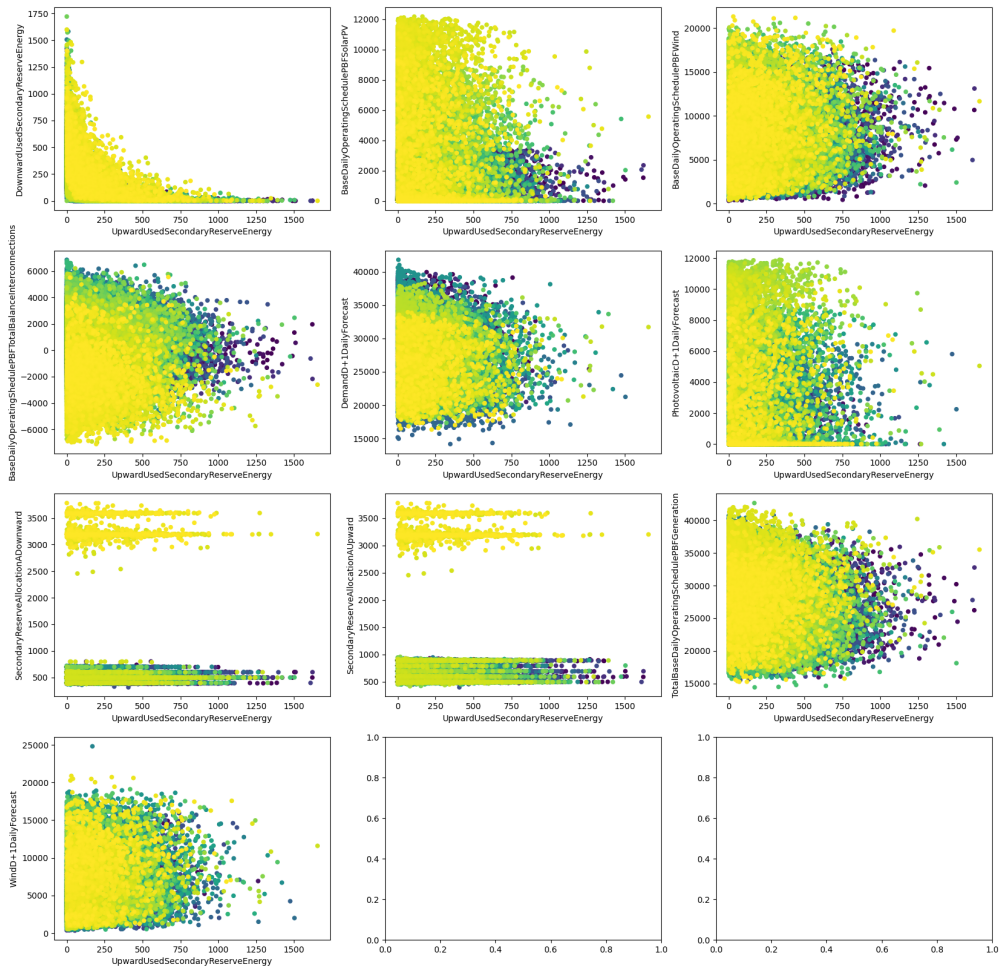


Figure 4.4: Correlação entre atributos

As correlações entre variáveis parecem muito escassas o que apresenta já que a previsão deste 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.

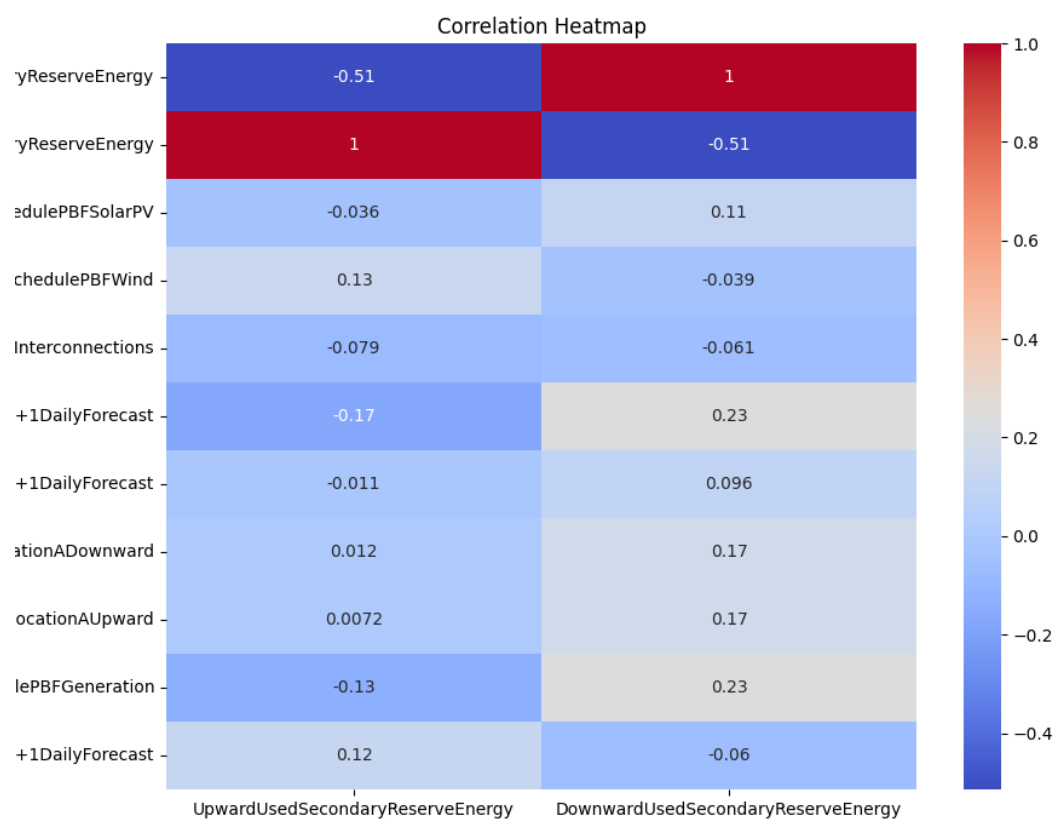


Figure 4.5: Valores de correlação entre atributos

4.2.1.2 Correlações Temporais

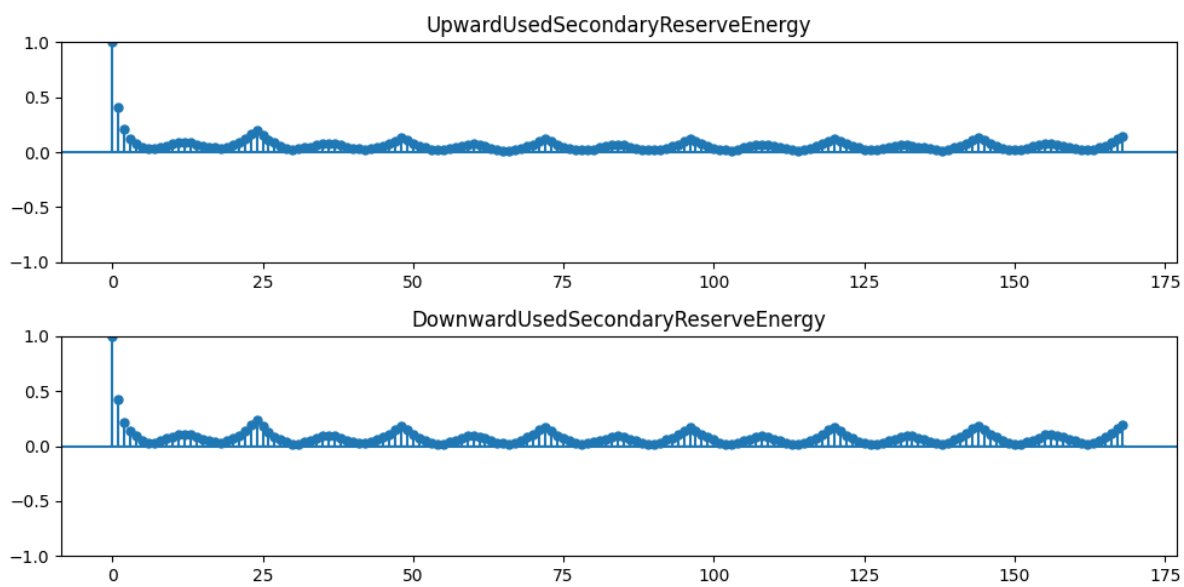


Figure 4.6: Serie Temporal dos dados alvo

#### 4. DADOS

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 uma 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.

#### 4.2.2 Agrupamento

Uma das possibilidades na modelação será a utilização de grupo de valores, classes, em conjunto com a linearidade.

Devido ao comportamento não exclusivamente linear é também estudado as possíveis agregações (clustering) em que podemos dividir os valores em classe.

Tendo por base que uma das classes é o valor zero, devido ao comportamento não linear desta série, vamos apenas testar quantos, e quais, as melhores classes em que podemos dividir os dados.

É realizado o teste de silhueta (ref) e o teste do cotovelo (ref), que se baseiam nos resultados de silhueta do modelo GMM (ref) e nos valores de inércia do modelo K-Means (ref).

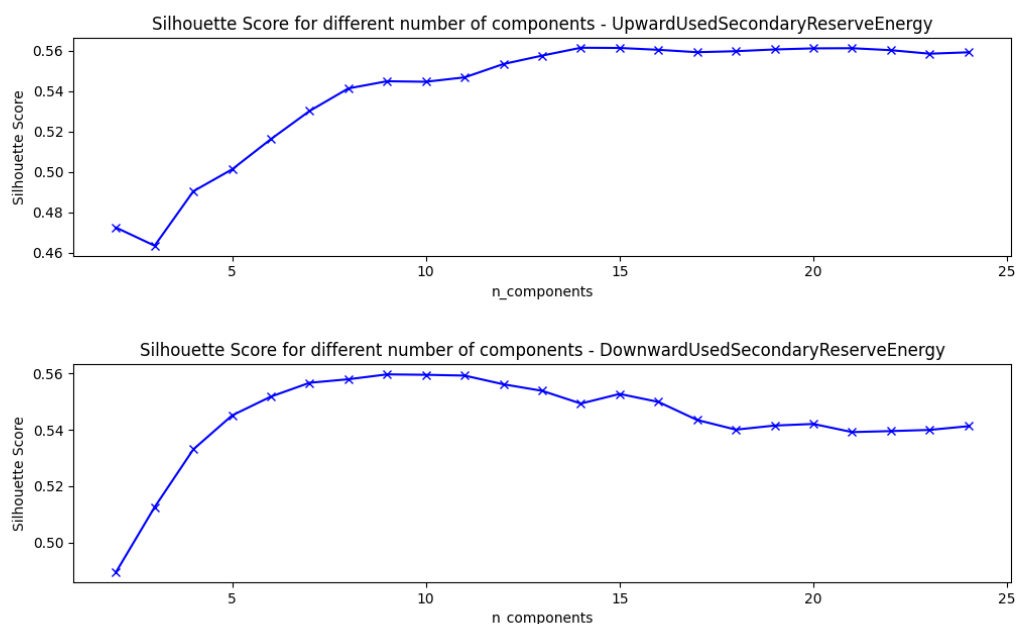


Figure 4.7: Serie Temporal dos dados alvo

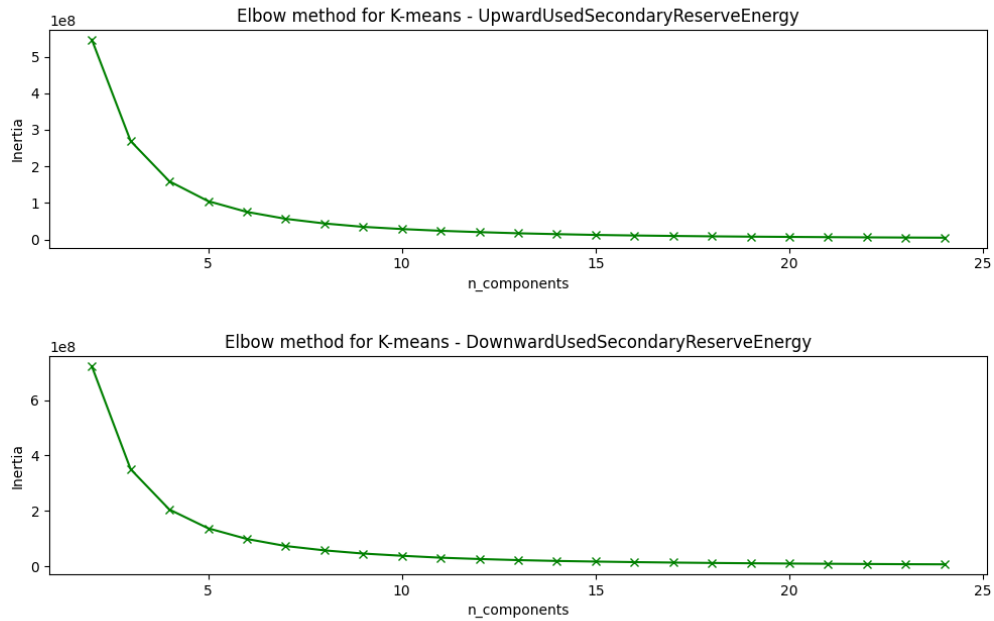


Figure 4.8: Serie Temporal dos dados alvo

Ambos os casos apontam um assintota na relação interna dos clusters, a partir de cerca de 5 clusters, sendo que o melhor valor dos verificados seria com 14 clusters em "UpwardUsedSecondaryReserveEnergy" e 9 em "DownwardUsedSecondaryReserveEnergy".

Para a nossa questão, queremos algumas classes, mas quanto menos classes mais facil será para os modelos correctamente identificar a que classe pertence. Logo para os valores apresentados, escolhemos 5 clusters, sendo que este já pode ser um numero elevado de classes, logo usemos 3 clusters se os modelos tiverem muita dificuldade com 5.

O método do cotovelo apresenta como numero ideal de clusters 4, mas sendo que K-Means é um metodo mais apropriado para distribuições normais, em que a distancia nos limites dos clusters não varia, deixamos apenas informativo.

O histograma das divisões pode ser visto em baixo. O valor é a sua própria classe para além das apresentadas.

## 4. DADOS

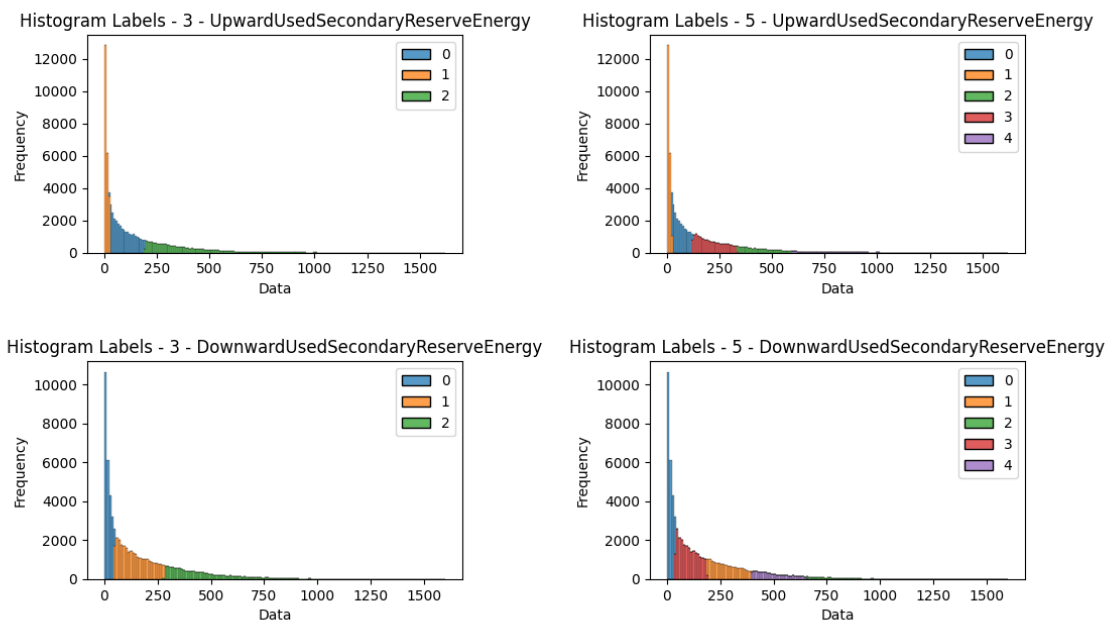


Figure 4.9: Histograma das classes

Os limites retirados são:

Atributo	Nº Clusters	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5
UpwardUsedSecondaryReserveEnergy	3	26.2	"26.3-194.9"	195.0		
UpwardUsedSecondaryReserveEnergy	5	20.2	"20.3-119.2"	"119.3-331.9"	"332.0-592.6"	592.7
DownwardUsedSecondaryReserveEnergy	3	47.0	"47.1-281.9"	282.0		
DownwardUsedSecondaryReserveEnergy	5	38.9	"39.0-186.9"	"187.0-393.1"	"393.2-658.0"	658.2

## 4.3 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 graficos seguintes que a existem alguns outliers, sendo estes definidos como 3 desvios padrão de distância à média.

Estes graficos 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 mesmo, sendo os dados passados a cru para os modelos.



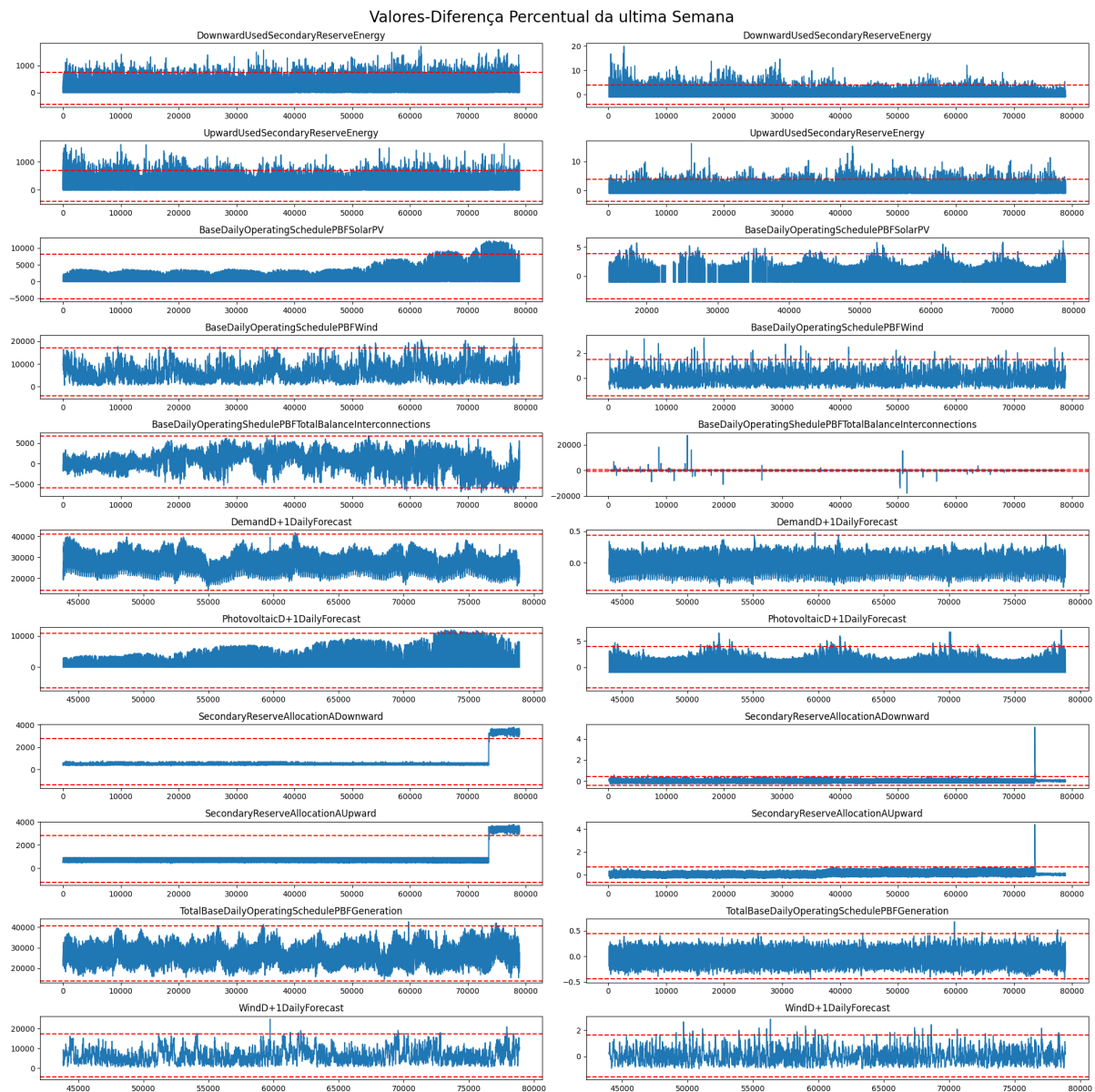


Figure 4.10: Outliers

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.

Visto estarmos a usar dados de 2014 a 2022, argumento até que podemos nem fazer divisão no treino, e usar como teste e validação dados de 2023.

#### 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.

## 4. DADOS

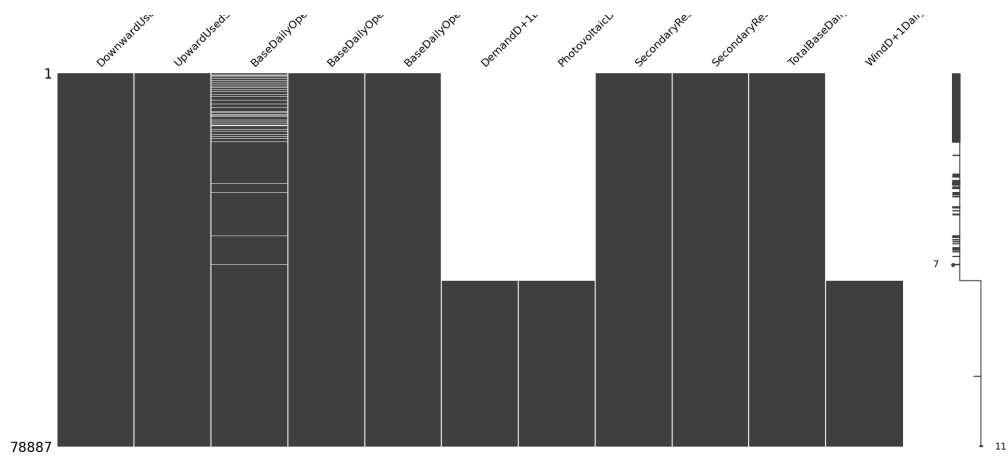


Figure 4.11: Dados em falta

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

Este método é baseado nos trabalhos de[12] e de[13]

Por último foi adicionado aos 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.

### 4.4 Considerações adicionais

Talvez aqui uma secção para finalizar e mostrar algumas coisas

# 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.

## 5.1 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.

### 5.1.1 Bloco Dense

O bloco dense sendo ele o mais simples é formado por duas camadas Dense [[empty citation](#)], em que a primeira apresenta um numero 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 [[14](#)].

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

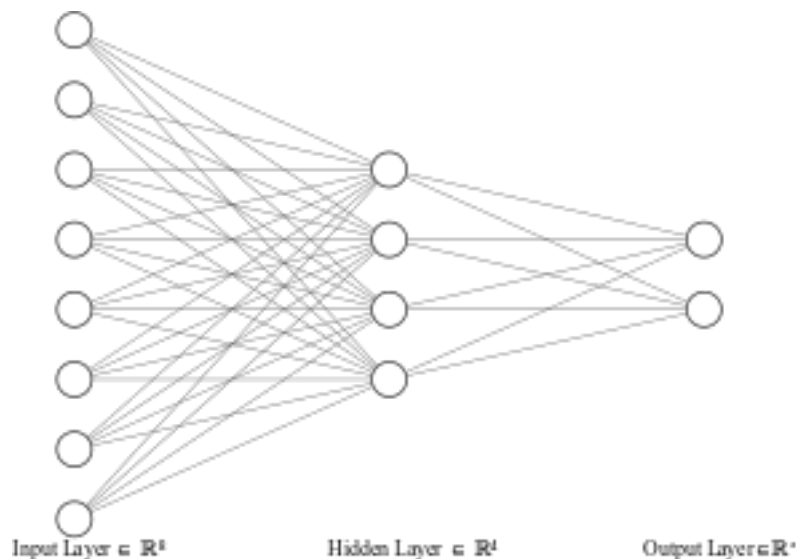


Figure 5.1: Bloco Dense

### 5.1.2 Bloco CNN

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

Normalmente usada em processamentos de imagens, o uso de convoluções temporais é também por si mesmo uma ideia forte.

explicar om que e CNN imagem

Usada também as ideias de attention, residual e o que eu chamei broad

### 5.1.3 Bloco LSTM

O uso de LSTM para previsões é uma área comum, mas aqui é seguido através das ideias partilhadas em [15], e reforçado pelo uso em previsões energéticas demonstrados em [16]

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

## 5.2 Arquiteturas

### 5.2.1 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"

imagem da mesma

### 5.2.2 Stacked

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

Como exemplo a arquitetura de "StackedCNN"

imagem da mesma

### 5.2.3 MultiHead

Multihead é o termo para quando os blocos de entrada e principais são repetidos paralelamente, um caminho para cada atributo, ou uma outra paralelização à escolha. Sendo depois concatenadas essas camadas e passadas juntas para a camada interpretativa.

Aqui foi usado sempre a paralelização por atributos, e ao invés de fazer Multihead no sentido de múltiplas entradas, para simplicidade de programação, foi feita uma paralelização interna no modelo, após a camada de entrada, onde a mesma é repetida para cada atributo.

Foi testado a diferença, e para os dados usados não havia diferenças de qualidade, mas sim em tempo de treino, logo a mais rápida foi a escolhida.

Como exemplo a arquitetura de "MultiheadCNN"

imagem da mesma

### 5.2.4 MultiTail

Esta arquitectura tem o mesmo conceito que a anterior a nível de paralelização, mas neste caso esta é feita apenas na camada interpretativa. Sendo que o resultado do bloco principal é repetido para criar a paralelização.

Neste caso foi paralelizado com o numero de tempos a prever, 24 horas, 24 objectos de saída destas modelos.

A grande diferença desta arquitectura para a "Vanilla" que preve 24 horas, é que aqui cada hora tem o seu proprio valor de função de perda, logo o modelo como que está a treinar 24 modelos diferentes, e no caso "Vanilla" a função de perda é única e é a media do erro das horas todas.

Como exemplo a arquitetura de "MultiTailCNN"

imagem da mesma

### 5.2.5 UNET

Normalmente usando em modelção de imagens, a arquitectura UNET passa por criar uma rede de expansão dos filtros, usando convoluções, e de seguida uma rede de contracção dos mesmo, 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 tambem adicionar aos blocos principais portões de atenção, portões residuais. Estas duas tecnicas são tambem 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

## 5.3 Considerações adicionais

Aqui e dizer que os modelos utilizados para teste sao as combinacoes deste blocos nestas aquiteturas. Imagens de layers criadas com dense <http://alexlenail.me/NN-SVG/index.html>

# Métodos

Neste capítulo percorremos as experiências realizadas. Estas foram feitas através dos usos dos programas criados para o efeito, disponíveis no repositório GitHub do [projecto](#).

## 6.1 Benchmark

Como modelo benchmark iremos usar a alocação feita. Pois são estes valores que procuramos melhorar no caso prático.

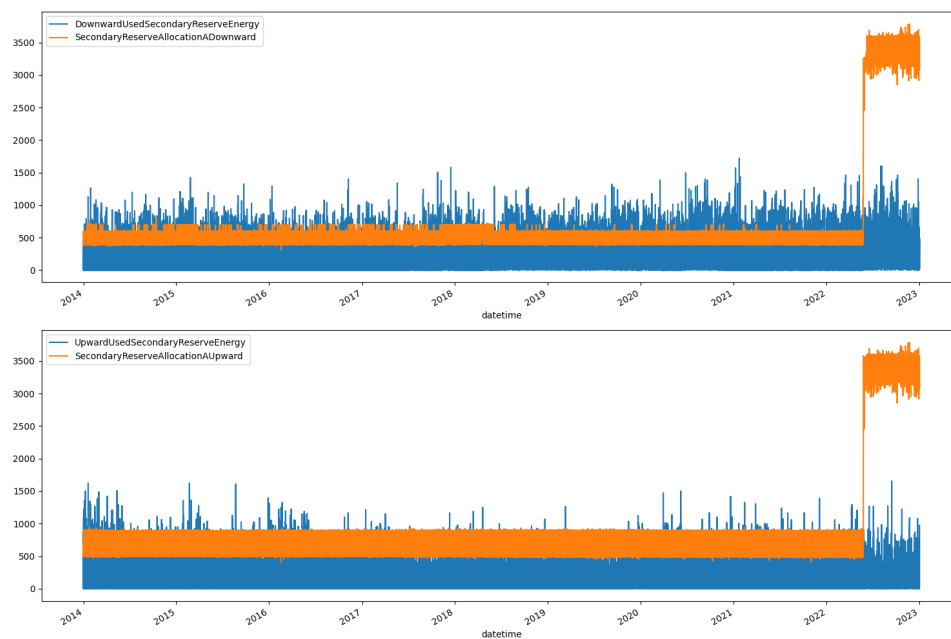


Figure 6.1: Serie Temporal do benchmark

Table 6.1: Dados Benchmark

rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus
963.1909591866699	53254091.2	False	-26.147637046658637	90.44161200506936	430902.00000000006	52823189.2

Para validação dos mesmo, vamos usar o ano 2021, devido aquele salto nos valores de alocação em 2022.

Para esses temos os seguintes dados:

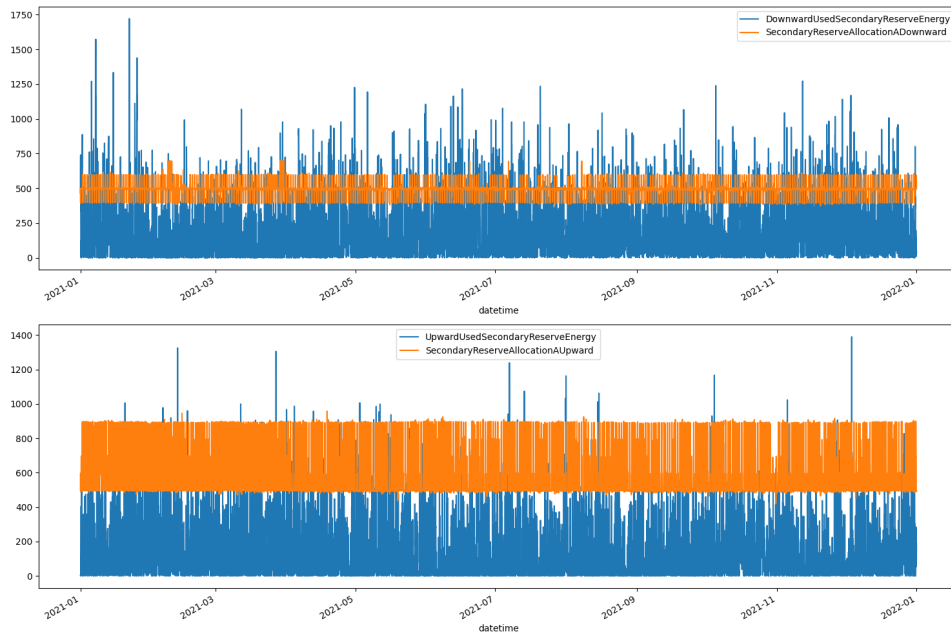


Figure 6.2: Serie Temporal do benchmark 2021

Os metodos em estudo vão ser comparados a esta medida. Sendo que o principal é baixar tanto a alocação perdida, como a alocação a mais. Que se traduzem no erro absoluto.

Table 6.2: Dados Benchmark de validação

rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus
488.55368458838365	3898946.9000000004	False	-6.008942605247387	85.98375795472323	57046.4	3841900.5

## 6.2 Modelos estatísticos

Antes de entrar para o desenvolvimento de modelos vamos usar métodos e modelos abertos para usar comparativamente.

Para as equações apresentadas temos que  $Y_t$  é a variável alvo, no tempo  $t$

Os modelos estatísticos recorrentes em previsões são AR, MA, ARMA, ARIMA, SARIMA para previsões so com um atributo, e para múltiplos atributos VAR.

O modelo AR não teve resultados aplicáveis, logo foi desconsiderado.

### 6.2.1 Univariate Analysis

Estas análises apenas aplicam uma fórmula à variável em questão.

#### 6.2.1.1 MA

MA - Moving Average

A MA

$$Y_t = c + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (6.1)$$

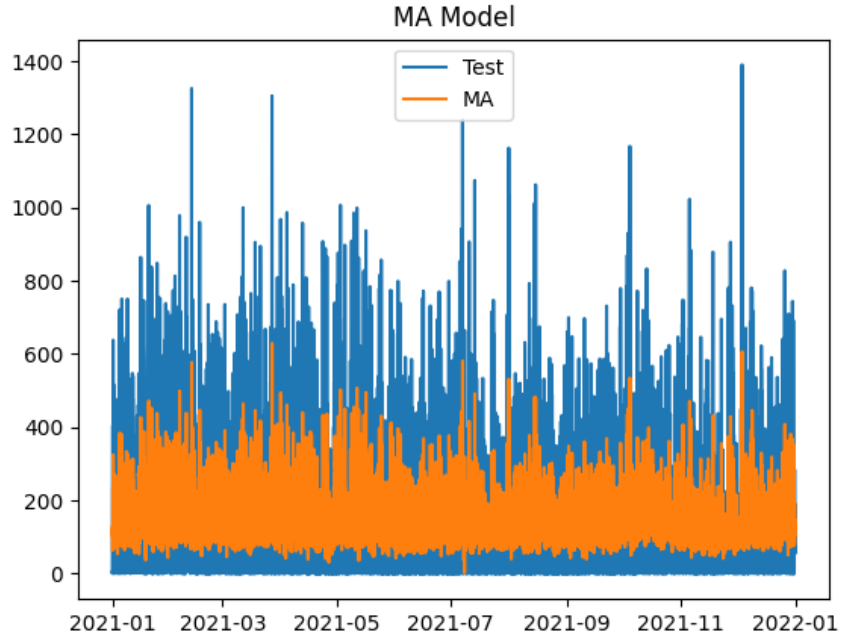


Figure 6.3: Previsões 2021 com modelo MA

### 6.2.2 ARMA

AR é blabla

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (6.2)$$

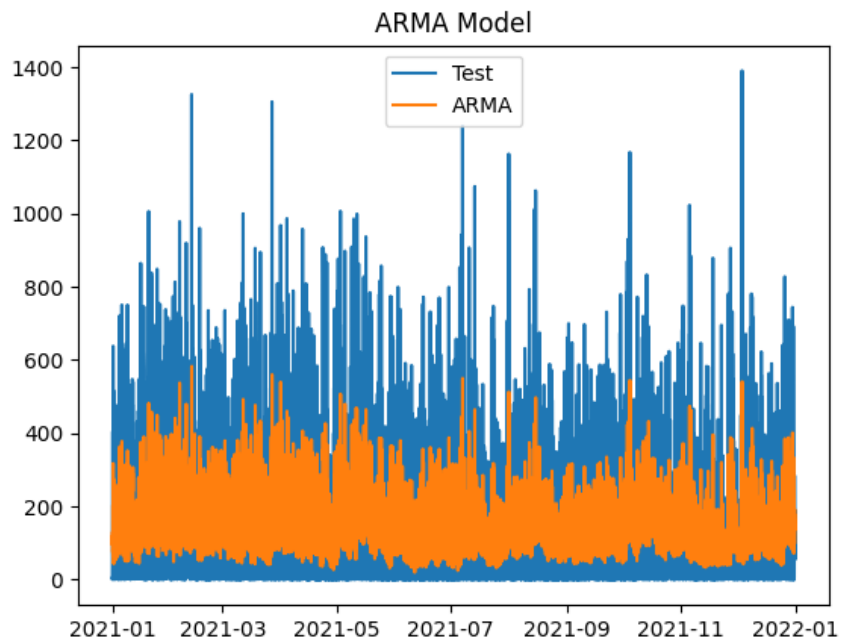


Figure 6.4: Previsões 2021 com modelo ARMA



### 6.2.3 ARIMA

AR eé blabla

$$Y'_t = c + \phi_1 Y'_t - 1 + \phi_2 Y'_t - 2 + \cdots + \phi_p Y'_t - p + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (6.3)$$

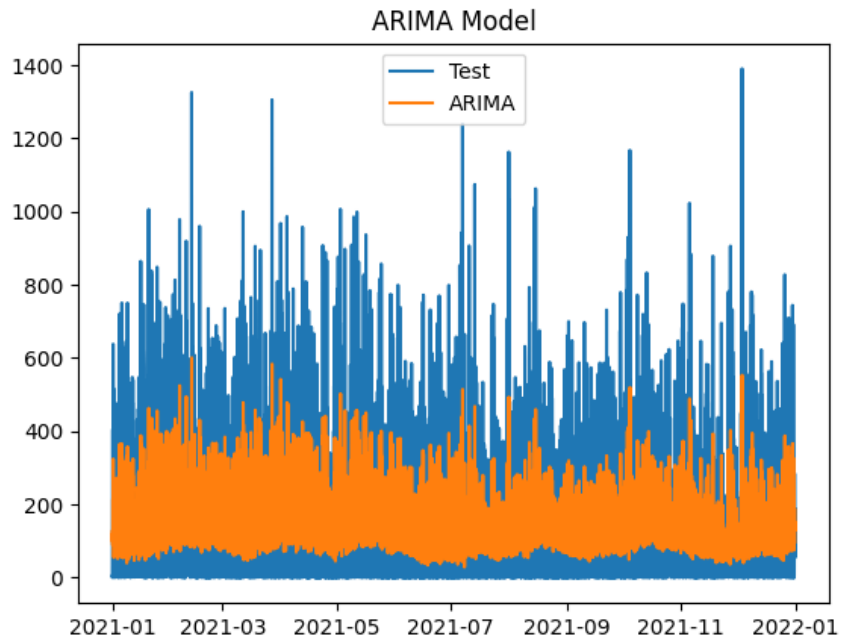


Figure 6.5: Previsões 2021 com modelo ARIMA

### 6.2.4 SARIMA

AR eé blabla

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (6.4)$$

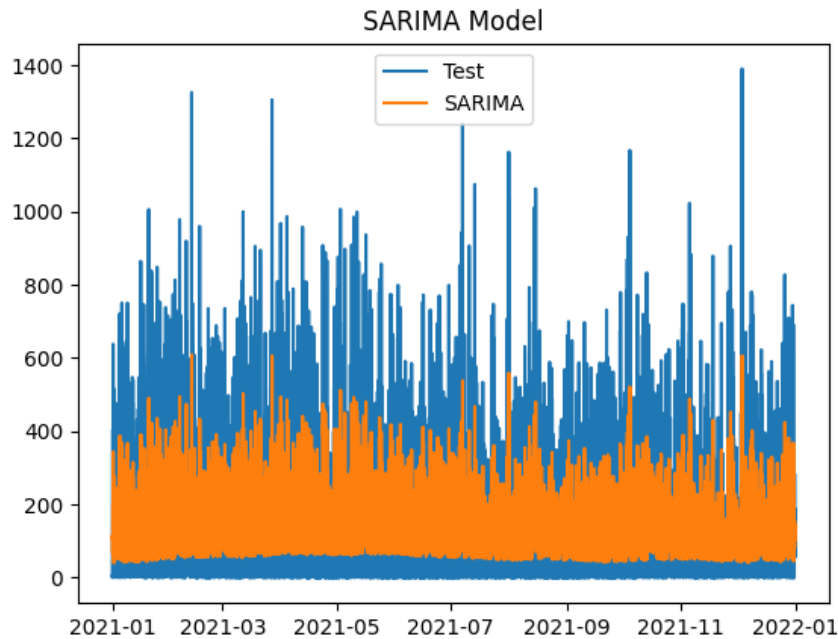


Figure 6.6: Previsões 2021 com modelo SARIMA

rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus	optimal percentage	better allocation	beter percentage
NaN	0.00	True	-0.65	NaN	NaN	NaN	0.00	0.00	0.00
171.14	1123214.13	True	0.14	17.04	560507.02	562707.11	63.79	63.79	86.84
169.99	1108555.94	True	0.15	16.36	554443.34	554112.60	64.04	64.04	87.17
170.13	1111215.12	True	0.15	16.52	556281.62	554933.50	64.03	64.03	87.01
171.88	1115725.36	True	0.13	16.43	568538.03	547187.33	63.45	63.45	86.92
184.60	1253196.02	True	-0.00	21.63	619929.79	633266.23	64.82	64.82	86.07

Apenas pelos métodos estatísticos verificamos que no ano de 2021 teria havido uma melhoria de cerca de 80% das vezes, usando qualquer um dos métodos apresentados.

Embora a alocação em falta seja de uma ordem de grandeza superior.

## 6.3 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 arquiteturas em estudo.

TODO: url

### 6.3.1 Construtor de modelos

Seguindo as arquiteturas descritas anteriormente esta ferramenta constroi os modelos automaticamente, sendo que precisamos apenas de fornecer os parametros variáveis.

O construtor assenta na idea de três camadas abstrastas de redes neuronais. A camada de entrada, a camada de bloco, e a camada intrepertativa.

A camada de entrada recebe os dados e normaliza, podendo tambem 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 multiplas 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 hiperparametrizações das mesmas, ou de cada camada, ou então da repetição do circuito, em paralelo ou em série. ou uma combinação destes.

### 6.3.2 Gerador de dados

O gerador construído 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 dos 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 seguintes. 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 manteria-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.

## 6.4 Treino e Resultados

Realizaram-se várias experiências, onde em cada uma se ia eliminando alguns dos objectos em estudo. Em casa experiências toda a parametrização era igual, à excepção do objecto de estudo.

### 6.4.1 Arquiteturas e números de épocas

Nesta experiência foi testado o resultado das várias arquiteturas em estudo, como também o impacto do número de épocas na qualidade dos modelos.

As arquiteturas estudadas foram:

- VanillaDense
- VanillaCNN
- VanillaLSTM
- StackedCNN
- StackedLSTM
- EncoderDecoder
- UNET

O modelo foi treinado em 200 épocas, sendo que foram salvos a cada 10 épocas, de forma conseguirmos perceber os contextos nos saltos de épocas.

As parametrizações usadas:

## 6. MÉTODOS

- loss : mean squared error
- Metodo activação no meio : relu
- Metodo activação no fim : relu
- optimizador : Adam
- Janela temporal em X : 168 horas (1 semana)
- Janela temporal em Y : 24 horas (1 semana)
- Fracção de treino : 95

### 6.4.2 Funções de Perda (Loss)

TODO: o que é a loss function?

Esta experiência consiste em rever que função de perda é melhor aplicavel ao problema. Sendo um problema de regressao linear, de valores bastante oscilatórios e com uma distribuição exponencial, temos algumas loss functions que já são reconhecidas para o problema.

- mean absolute error
- mean squared error
- loss : mean absolute error

### 6.4.3 Hiperparametrização

#### 6.4.3.1 Activação

#### 6.4.3.2 Optimizadores

### 6.4.4 Janelas Temporais

Um dos pontos deste trabalho é perceber a fesiabilidade de usar dados de previsão do dia anterior (DA) para estes atributos energéticos.

Algo que pode ser também aplicado no futuro a outros dados que não DA, mas sim a 3 horas, ou a 8 horas.

Para perceber esta flexibilidade, mas especialmente para escolher as melhores janelas temporais a usar neste modelos, vamos testar várias combinações.

Mantendo em mente que o objectivo é prever 24 horas, para os casos onde o alvo não dá um previsão de 24 horas, é necessario criar um numero de modelos para fazer as 24 horas.

Para validação apenas é usado o espaço temporal previsto, e não multiplos modelos.

Dado as análises de autocorrelação iremos usar como janelas para treino o conjunto [24, 48, 96, 168] para prever o conjunto [1, 4, 8, 12, 24]

Para alem destes foram também testadas combinações com janelas de treino 8 e 12 horas. Estas mostraram rapidamente que janelas de treino menores que as de previsão funcionam muito mal.

### 6.4.5 Classificação

Como descrito em (ref)... existe também o uso de tanto classes como valores lineares para resolução de problemas de regressão, também chamado *cluster-wise regression*.

Para este teste mudamos um pouco o modelo em uso. Ao invés de apenas uma camada interpretativa, fazemos duas, em paralelo, sendo que uma resolve a regressão e a outra a classificação.

Outro caso, proposto aqui, é usar uma nova camada interpretativa, que combina as duas saídas anteriores (linear e classificação), e resolve novamente para os valores lineares.

Estes modelos não teram apenas uma saída, mas varias, como as arquiteturas MultiTail, mas neste caso cada uma resolve para um problema diferente, com funções de perda, e ativações diferentes.

TODO: desenho destas duas camadas interpretativas

### 6.4.6 Pesos

Por ultimo foi testado o impacto do uso de pesos nos modelos. Estes pesos são o peso que aquele alvo TODO: explicar pesos

#### 6.4.6.1 Modelos lineares

Para os modelos lineares o peso que é adiciona ao modelo é a distância à média.

Este peso serve para dar mais importância a valores facilmente considerados outliers.

#### 6.4.6.2 Modelos Lineares e de Classificação

Aqui o peso é dado por saída. Para as saídas lineares o peso dados é o mesmo que apresentado anteriormente, para a saída de classificação, o peso é o inverso da frequência da classe.

Distribuindo assim a importância de treino pela frequência das classes. Sendo um prática comum especialmente quando as distribuições são muito desiguais, como o caso em estudo.

É aqui estudada a aplicação destes pesos individualmente, e em conjunto.

Os pesos aqui são também normalizados de modo a que o maior peso em cada um deles seja 1, e logo a multiplicação dos dois esteja dentro das mesmas dimensões de relevância.

$$P_m = |y - \text{mean}| \quad (6.5)$$

## 6.5 Considerações adicionais

Foram realizados testes adicionais que não obtiveram resultados passivos de boa interpretação, e foram imediatamente descartados, como:

- Janela temporal em X : 96, 48, 24
- otimizador : todos os otimizadores disponiveis na biblioteca keras
- loss : todas as outra loss functions de regressão disponiveis.
- epocas : influência do numero de epocas nos modelos, foram treinados até 20000 epocas alguns modelos mas à medida que a perda ia estagnando na assintomta, o modelo ia apenas piorando.

Todos os metodos foram realizados utilizando código em python, que está aberto em <https://github.com/JotaFan/renewable-generation-into-reserve-markets>

# Resultados e discussão

Os resultados das experiências são apresentados por experiência, sendo que cada um vai testando e eliminando parametros na modulação.

Após análise inicial na modelação foi concluído que usar tentar modelar apenas um dos atributos de cada vez leva a melhor resultados que tentar modelar os dois no mesmo modelo.

Foi também visto que os dois atributos em causa são analogos, logo os sistemas que melhor represem um dos atributos tambem são semelhantemente eficazes no outro.

Assim todas as experiências foram realizadas usando apenas um dos atributos alvo, sendo este o "UpwardUsedSecondaryReserveEnergy", e em que o atributo de alocação comparativo é "SecondaryReserveAllocationAUpward".

## 7.1 Métricas

A nível de comparação efectiva de modelos iremos usar as seguintes metricas: RMSE, erro absoluto, r2 score, percentagem optima, alocação em falta, alocação a mais.

Sendo que as métricas de decisão final são as que representam melhor o objectivo: baixar os custo de alocação das reservas secundárias. logo serão as metricas de alocação em falta e em demasia, sendo que a soma das mesmas é o erro absoluto.

$$RMSE(y, \hat{y}) = \sqrt{\frac{\sum_{i=0}^{N-1} (y_i - \hat{y}_i)^2}{N}} \quad (7.1)$$

$$\text{Absolute Error} = \sum_{i=0}^{N-1} |y_i - \hat{y}_i| \quad (7.2)$$

$$R^2 \text{ score} = 1 - \frac{\sum_{i=0}^{N-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{N-1} (y_i - \bar{y})^2} \quad (7.3)$$

$$\text{alocação em falta} = \begin{cases} 0 & , \text{if } \hat{y} \geq y \\ y - \hat{y} & , \text{otherwise} \end{cases} \quad (7.4)$$

$$\text{optimal percentage} = \frac{1}{N} \sum_{i=1}^n 1[\hat{y}_i \geq y_i \& \hat{y}_i \leq alloc](\text{not: [17]}) \quad (7.5)$$

Onde  $\hat{y}$  são as previsões dos modelos,  $y$  são os valores reais utilizados, "UpwardUsedSecondaryReserveEnergy", e  $alloc$  são os valores alocados "SecondaryReserveAllocationAUpward"

## 7.2 Experiências

Para comparação o erro absoluto para o ano de 2021 na alocação feita é de 3889367.4

## 7.2.1 Arquiteturas e numeros de epocas

### 7.2.1.1 Arquiteturas

Table 7.1: Resultados Arquitecturas

name	rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus	optimal percentage	better allocation	beter percentage
StackedLSTMA	NaN	0.00	True	-0.65	NaN	NaN	NaN	0.00	0.00	0.00
VanillaDense	1200.20	10073188.28	False	-41.27	192.50	487.24	10072701.04	1.13	0.77	1.17
StackedCNN	453.12	3717806.99	True	-5.03	82.67	52038.16	3665768.83	35.04	34.48	38.34
UNET	176.73	1138605.66	True	0.08	16.85	619845.54	518760.12	62.20	62.20	86.14
VanillaCNN	174.96	1124205.72	True	0.10	16.45	614355.90	509849.82	61.79	61.79	86.18

### 7.2.1.2 Epocas

## 7.2.2 Funções de Perda (Loss)

As funções de perda foram estudas com duas arquitecturas diferentes, de modo a conseguir ter uma melhor noção do impacto das mesmas.

Table 7.2: Resultados Funções de perda

name	rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus	optimal percentage	better allocation	beter percentage
UNETmse	180.84	1269085.40	True	0.04	23.32	521000.39	748085.00	68.78	68.78	87.31
StackedCNNmae	198.29	1083872.29	True	-0.15	5.93	937786.28	146086.01	43.65	43.65	83.32
StackedCNNmape	232.60	1257578.80	True	-0.59	1.01	1252220.35	5358.46	19.15	19.15	80.60
VanillaCNNmse	162.44	1038022.80	True	0.23	14.30	568479.54	469543.26	61.02	61.02	87.08
VanillaCNNmsde	796.03	6765714.27	False	-17.60	136.09	3529.99	6762184.27	3.46	2.63	3.83
StackedCNNmse	176.71	1103152.68	True	0.08	14.81	673158.53	429994.14	58.75	58.75	85.68
StackedCNNwl	175.87	1186767.36	True	0.09	20.02	572945.10	613822.26	64.92	64.92	86.81
VanillaCNNwl	1071.18	8912349.22	False	-32.67	171.33	63940.63	8848408.59	1.01	0.62	3.89
UNETmape	3545800.24	18933122401.84	False	-368973766.69	276314.32	0.00	18933122401.84	0.00	0.00	0.00
UNETmae	193.39	1082997.34	True	-0.10	8.09	862394.46	220602.88	47.89	47.89	83.85
VanillaCNNmae	181.82	994893.68	True	0.03	6.10	812734.44	182159.24	42.43	42.43	84.75
StackedCNNmsle	290.70	2299543.31	True	-1.48	54.80	193283.75	2106259.56	86.34	86.34	92.14
StackedCNNmsde	612.31	4978280.05	False	-10.00	81.86	3325165.78	1653114.27	18.27	18.06	30.14
UNETmsde	199.82	1111926.57	True	-0.17	7.56	912655.91	199270.66	45.02	45.02	83.68
UNETmsle	201.13	1103254.30	True	-0.19	6.65	936874.71	166379.59	45.34	45.34	83.23
UNETwl	30011.42	141637978.61	False	-26431.60	2558.08	10334.28	141627644.33	9.35	8.62	9.91
VanillaCNNmsle	1250.10	10759371.90	False	-44.86	201.97	0.00	10759371.90	0.22	0.00	0.22

A nível de percentagem de modelo melhor, elas estão todas bastante renhidas, mas ha uma claro vantagem quando vemos a percentagem de melhor alocação ou de alocação optima.

Sendo que a perda que avança é a de MSLE (Mean Square Log Error).

Esta perda é tendicionalmente usada para distribuições exponencias, e onde temos bastantes outliers, que devem ser considerados. O que é o caso no nosso problema.

Esta experiência valida a observação feito na análise estatística sobre o mesmo.

## 7. RESULTADOS E DISCUSSÃO

### 7.2.3 Hiperparametrização

#### 7.2.3.1 Activação

Table 7.3: Resultados Ativação

name	rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus	optimal percentage	better allocation	beter percentage
StackedCNN relu relu	207.66	1113406.16	True	-0.27	3.90	1028475.81	84930.35	38.64	38.64	82.31
StackedCNN tanh tanh	236.24	1289434.20	True	-0.64	0.85	1288660.90	773.30	11.47	11.47	80.40
StackedCNN relu tanh	236.24	1289434.20	True	-0.64	0.85	1288660.90	773.30	11.47	11.47	80.40
StackedCNN relu softsign	236.31	1290138.00	True	-0.64	0.86	1289376.80	761.20	11.29	11.29	80.40
StackedCNN linear tanh	236.24	1289434.20	True	-0.64	0.85	1288660.90	773.30	11.47	11.47	80.40
StackedCNN softsign linear	1112.70	8936608.01	False	-35.34	178.94	8185.60	8928422.41	7.99	7.50	8.60
StackedCNN softsign softsign	236.24	1289436.03	True	-0.64	0.85	1288662.99	773.04	11.00	11.00	80.39
StackedCNN softplus elu	294.00	2287913.46	True	-1.54	52.62	273134.72	2014778.74	79.27	79.25	88.22
StackedCNN softplus softplus	393.20	3191326.78	True	-3.54	72.87	84140.16	3107186.62	69.22	69.15	71.84
StackedCNN relu selu	262.76	1986551.87	True	-1.03	43.04	261245.83	1725306.04	77.31	77.17	86.49
StackedCNN linear elu	297.39	2368520.21	True	-1.60	55.86	177934.14	2190586.07	87.47	87.47	92.53
StackedCNN softsign tanh	236.24	1289434.20	True	-0.64	0.85	1288660.90	773.30	11.47	11.47	80.40
StackedCNN softplus exponential	461.92	3782992.35	True	-5.26	82.96	50599.52	3732392.83	32.55	31.88	35.55
StackedCNN softplus selu	287.86	2249601.96	True	-1.43	53.57	218301.68	2031300.27	84.44	84.44	91.23
StackedCNN softplus softsign	236.36	1291380.19	True	-0.64	0.90	1290725.89	654.30	9.39	9.39	80.40
StackedCNN softplus linear	208.37	1113888.40	True	-0.27	3.74	1032928.56	80959.84	38.23	38.23	82.30
StackedCNN softsign relu	203.70	1100693.28	True	-0.22	4.92	986235.38	114457.90	41.64	41.64	82.69
StackedCNN softsign exponential	508.90	4187396.73	False	-6.60	91.15	32147.81	4155248.92	25.81	24.91	28.09
StackedCNN linear softsign	236.24	1289434.20	True	-0.64	0.85	1288660.90	773.30	11.47	11.47	80.40
StackedCNN relu exponential	542.88	4482255.28	False	-7.65	96.23	31472.22	4450783.06	22.04	21.01	24.00
StackedCNN softsign selu	2053.61	17501165.80	False	-122.77	323.70	73467.70	17427698.09	0.01	0.00	2.98
StackedCNN tanh softplus	203.53	1097751.79	True	-0.22	4.71	986626.52	111125.27	41.31	41.31	82.73
StackedCNN tanh linear	543.57	4512608.50	False	-7.67	97.01	27030.50	4485578.00	20.81	19.78	22.73
StackedCNN relu elu	481.58	3739379.02	True	-5.81	83.92	84681.14	3654697.88	50.13	49.65	53.43
StackedCNN linear exponential	457.52	3752241.97	True	-5.14	83.28	52642.15	3699599.81	29.83	29.21	33.15
StackedCNN relu softplus	187.41	1193440.68	True	-0.03	17.36	716295.64	477145.04	59.08	59.08	85.03
StackedCNN linear selu	559.84	4606574.05	False	-8.20	98.93	35641.95	4570932.10	22.53	21.82	24.82
StackedCNN softplus relu	380.03	3059764.47	True	-3.24	69.45	99012.20	2960752.27	70.70	70.58	74.00
StackedCNN linear relu	614.15	4962488.83	False	-10.07	100.79	66478.05	4896010.78	19.20	18.28	23.60
StackedCNN linear softplus	321.88	2577000.96	True	-2.04	59.88	149354.46	2427646.50	88.59	88.59	92.51
StackedCNN tanh relu	548.29	4556413.79	False	-7.82	97.85	26362.43	4530051.37	20.66	19.65	22.52
StackedCNN softsign softplus	900.71	6351850.55	False	-22.81	117.73	22655.51	6329195.04	17.30	16.55	18.83
StackedCNN linear linear	238.22	1866499.25	True	-0.67	43.41	292863.60	1573635.64	81.02	81.02	90.41
StackedCNN tanh softsign	236.24	1289436.20	True	-0.64	0.85	1288663.18	773.02	11.00	11.00	80.39
StackedCNN softsign elu	205.05	1105893.27	True	-0.23	4.88	995748.76	110144.51	41.17	41.17	82.57
StackedCNN relu linear	431.38	3464311.65	True	-4.46	76.14	133075.54	3331236.11	45.08	44.54	51.57
StackedCNN softplus tanh	236.24	1289434.20	True	-0.64	0.85	1288660.90	773.30	11.47	11.47	80.40

#### 7.2.3.2 Optimização

### 7.2.4 Janelas Temporais

Table 7.4: Resultados Janelas Temporais

name	rmse	abs erro	erro comp	r2 score	mape score	alloc missing	alloc surplus	optimal percentage	better allocation	beter percentage
StackedCNN 48X 24Y	479.11	3989558.86	False	-5.75	86.62	43807.71	3945751.15	29.86	29.19	32.72
StackedCNN 98X 1Y	445.28	3671816.03	True	-4.83	81.15	54078.28	3617737.75	37.10	36.50	40.30
StackedCNN 98X 24Y	1167.21	9770108.13	False	-39.09	191.54	1578.73	9768529.40	2.25	1.83	2.41
StackedCNN 24X 1Y	209.47	1149047.56	True	-0.29	4.43	1055993.70	93053.86	39.55	39.55	82.21
StackedCNN 98X 4Y	495.81	4120357.48	False	-6.23	88.95	36888.60	4083468.88	27.15	26.22	29.58
StackedCNN 24X 4Y	453.22	3779420.50	True	-5.04	82.35	52131.12	3727289.38	33.55	32.94	36.88
StackedCNN 24X 12Y	450.27	3753076.87	True	-4.96	82.29	53778.38	3699298.49	34.86	34.35	38.25
StackedCNN 24X 24Y	468.78	3907629.32	True	-5.46	84.62	48042.45	3859586.87	26.35	25.65	29.52
StackedCNN 168X 4Y	247.11	1941513.71	True	-0.79	45.26	272577.02	1668936.69	82.07	82.07	90.72
StackedCNN 48X 1Y	510.34	4249569.03	False	-6.65	91.65	35151.96	4214417.07	23.58	22.63	26.05
StackedCNN 24X 8Y	209.71	1150163.22	True	-0.29	4.41	1058328.49	91834.73	39.32	39.32	82.24
StackedCNN 48X 4Y	443.47	3672502.43	True	-4.78	80.47	56035.63	3616466.80	33.50	33.02	36.99
StackedCNN 48X 12Y	451.66	3755277.12	True	-4.99	82.42	51133.01	3704144.12	35.83	35.34	39.13
StackedCNN 48X 8Y	425.76	3528446.38	True	-4.33	78.29	63798.09	3464648.29	35.06	34.73	38.87
StackedCNN 98X 8Y	401.18	3287580.95	True	-3.73	74.38	76725.24	3210855.71	51.78	51.65	55.36
StackedCNN 98X 12Y	622.31	5179835.43	False	-10.41	108.86	19985.94	5159849.49	17.78	16.78	19.33
StackedCNN 168X 24Y	352.28	2757367.10	True	-2.64	65.61	155940.39	2601426.71	70.78	70.65	75.40
StackedCNN 168X 8Y	277.45	2189089.58	True	-1.26	51.96	226599.69	1962489.89	84.47	84.47	91.59
StackedCNN 168X 12Y	404.20	3219562.53	True	-3.79	70.85	90454.58	3129107.95	67.74	67.30	71.01
StackedCNN 168X 1Y	568.14	4694571.86	False	-8.47	100.18	23253.65	4671318.21	20.47	19.49	22.15



### 7.2.5 Classificação

### 7.2.6 Pesos

Aqui devem constar gráficos e sua análise crítica e ligação com a secção 2 da revisão bibliográfica no sentido de comparar valores e discutir diferenças, por exemplo. A legenda dos gráficos deve seguir a das figuras, isto é, porque também são figuras.

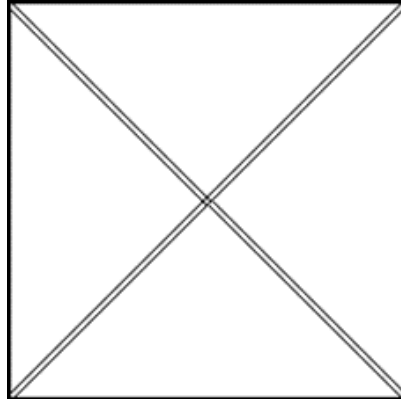


Figure 7.1: Exemplo de como considerar um gráfico

## **Conclusões e sugestões futuras**

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 7.2.6. Não esquecer de indicar sugestões futuras para que um colega possa dar continuidade ao trabalho desenvolvido.

# Referências

- [1] 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).
- [2] 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).
- [3] 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).
- [4] 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>.
- [5] Fernando Lopes. “From wholesale energy markets to local flexibility markets: structure, models and operation”. In: *Local Electricity Markets* (Jan. 2021), pp. 37–61. DOI: [10.1016/B978-0-12-820074-2.00009-5](https://doi.org/10.1016/B978-0-12-820074-2.00009-5).
- [6] William F. Watson, Paul L. Joskow, and Richard Schmalensee. “Markets for Power: An Analysis of Electrical Utility Deregulation”. In: *Southern Economic Journal* 51 (2 Oct. 1984), p. 640. ISSN: 00384038. DOI: [10.2307/1057863](https://doi.org/10.2307/1057863).
- [7] Fred C. Schweppe et al. “Spot Pricing of Electricity”. In: (1988), p. 384. URL: [https://books.google.com/books/about/Spot\\_Pricing\\_of\\_Electricity.html?hl=pt-PT&id=QZ8ACAAAQBAJ](https://books.google.com/books/about/Spot_Pricing_of_Electricity.html?hl=pt-PT&id=QZ8ACAAAQBAJ).
- [8] 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>.
- [9] Ricardo J. Bessa et al. “Reserve Setting and Steady-State Security Assessment Using Wind Power Uncertainty Forecast: A Case Study”. In: *IEEE Transactions on Sustainable Energy* 3 (4 Oct. 2012), pp. 827–836. ISSN: 1949-3029. DOI: [10.1109/TSTE.2012.2199340](https://doi.org/10.1109/TSTE.2012.2199340).
- [10] Camila Fernandes, Pablo Frías, and Javier Reneses. “Participation of intermittent renewable generators in balancing mechanisms: A closer look into the Spanish market design”. In: *Renewable Energy* 89 (Apr. 2016), pp. 305–316. ISSN: 0960-1481. DOI: [10.1016/J.RENENE.2015.12.037](https://doi.org/10.1016/J.RENENE.2015.12.037). URL: <https://linkinghub.elsevier.com/retrieve/pii/S0960148115305437>.
- [11] Marco Agostini et al. “The participation of small-scale variable distributed renewable energy sources to the balancing services market”. In: *Energy Economics* 97 (May 2021), p. 105208. ISSN: 0140-9883. DOI: [10.1016/J.ENER.2021.105208](https://doi.org/10.1016/J.ENER.2021.105208).

## REFERÊNCIAS

- [12] 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>.
- [13] 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>.
- [14] 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).
- [15] 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).
- [16] 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).
- [17] Kenneth E. Iverson. *A programming language*. 1962, p. 11. ISBN: 0471430145. URL: <https://www.jsoftware.com/papers/APL.htm>.

# Anexos

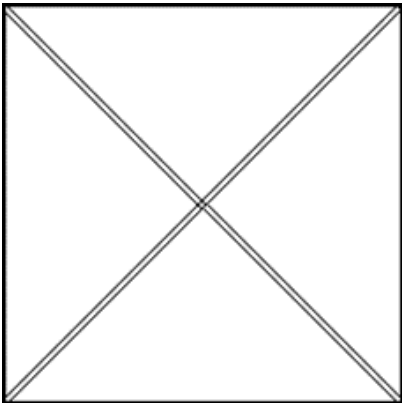


Figure A.1: Exemplo de como considerar um gráfico nos anexos.

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