Faculdade de Engenharia da Universidade do Porto

Previsão de preços de mercado baseada em Deep Learning

Ana Rita Martins Cruz e Silva

VERSÃO FINAL

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Energia

Orientador: J. Nuno Fidalgo
Co-orientador: José Ricardo Andrade

28/06/2021

Resumo

As técnicas de *Deep Learning* têm sido alvo de muito interesse por parte da comunidade científica. Atualmente, esta tecnologia é aplicada em situações multidisciplinares que afetam diretamente a vida das pessoas seja na deteção de doenças cancerígenas realizada em meio hospitalar, como na sugestão de um filme pela Netflix no conforto das nossas casas. Este tipo de tecnologia permitiu que a Amazon lançasse a aplicação Alexa, que usa redes neuronais recorrentes para o reconhecimento de voz ou, por outro lado, que o reconhecimento de rostos e objetos se tornasse uma tarefa automática e bastante eficiente.

Esta dissertação tem como objetivo a utilização de técnicas de *Deep Learning* na previsão de preços do mercado de eletricidade. São testadas e comparadas três técnicas de *Deep Learning*: *Dense Neural Networks* (DNN), *Long Short Term Memory* (LSTM) e *Convolutional Neural Networks* (CNN).

Pretende-se, num primeiro estudo, verificar se estas técnicas são vantajosas neste problema relativamente a outras abordagens mais convencionais, em particular no que respeita à sua capacidade de proporcionar um menor erro de previsão.

Uma das tarefas que consome grande parte do tempo de desenvolvimento de um algoritmo de previsão é a fase de seleção das variáveis mais relevantes. O segundo estudo visa explorar a capacidade das CNN de ultrapassar esta fase de seleção de *inputs* através da consideração de todas as variáveis disponíveis em muitas instâncias temporais. Pretende-se testar se os efeitos convolutivos da CNN são suficientes para extrair a informação relevante, reduzindo ao mesmo tempo a possibilidade de *overfitting*, de forma a proporcionar previsões de grande qualidade.

Os resultados do primeiro estudo mostram que a aplicação de técnicas de *Deep Learning* se tornou vantajosa. Quando o conjunto de entradas se caracteriza apenas por variáveis endógenas, as LSTM e as CNN mostraram ter uma capacidade maior de aprendizagem, relativamente às demais. A inclusão de variáveis exógenas resultou numa diminuição significativa do erro para todas as arquiteturas de rede, em especial para as DNN.

Os resultados do segundo estudo mostram que o uso de redes convolucionais parece dispensar ou, pelo menos, aliviar a fase de seleção de variáveis.

Abstract

Deep Learning techniques have been the target of much interest from the scientific community. Currently, this technology is applied in multidisciplinary situations that directly affect people's lives, whether in the detection of cancerous diseases carried out in a hospital environment, or in the suggestion of a film by Netflix in the comfort of our homes. This type of technology required Amazon to launch Alexa's application, which uses recurrent neural networks for voice recognition or, on the other hand, that the recognition of faces and objects became an automatic and very efficient task.

This dissertation aims to use Deep Learning techniques to forecast electricity market prices. Three Deep Learning techniques are tested and compared: Dense Neural Networks (DNN), Long Term Memory (LSTM) and Convolutional Neural Networks (CNN).

In a first study, it is intended to verify if these techniques are advantageous in this problem compared to other common approaches, in particular with regard to their ability to provide a smaller forecast error.

One of the tasks that consumes a large part of the development time of a forecasting algorithm is the selection phase of the main variables. The second study, explores the ability of CNN to overcome this input selection phase by considering all available variables in many temporal moments. It is intended to test whether the convolutive effects of CNN are sufficient to extract the relevant information in order to provide high quality forecasts.

The results of the first study show that the application of Deep Learning techniques is advantageous. When the set of inputs is made up only of endogenous variables, the LSTM and CNN networks have greater learning capacity compared to the others. The inclusion of exogenous variables resulted in a reduction in the error value for all network architectures, especially for dense neural networks.

he results of the second study show that the use of convolutional networks seems to dispense (or at least alleviate) the variable selection phase.

Agradecimentos

Ao professor Nuno Fidalgo, por ter permitido a realização de uma dissertação sobre um tema tão contemporâneo e por todo o acompanhamento e auxílio prestados.

Ao Ricardo Andrade, coorientador deste trabalho, por todos os conhecimentos de *Deep Learning* que me transmitiu e por ter disponibilizado o seu tempo para a realização desta dissertação.

Ao Ivo, que sempre me deu a mão em qualquer projeto académico, que partilhou comigo muitos momentos difíceis nesta Faculdade e que é todos os dias uma fonte de inspiração intelectual.

Às minhas colegas de trabalho, Vera e Cristina, e ao Engenheiro Dionísio, que fazem parte do meu primeiro projeto profissional e que sempre compreenderam a minha indisponibilidade.

Aos meus amigos, por serem os melhores.

Ao meu pai, que há muitos anos é aquele que partilha comigo o dia-a-dia e que soube sempre aceitar os dias em que não me sentei com ele para jantar.

À minha irmã Matilde, que desde que nasceu me mostrou que, mesmo tendo todas as desculpas para sermos pouco, fomos o melhor que conseguimos.

À minha avó, que soube sempre engomar-me as desventuras da vida.

Às minhas tias, que sempre foram mães desde que a minha faleceu. Em especial, à Susana. Sem ela, nada disto teria sido possível!

À minha Mãe, que sempre me incutiu os ideais de mulher trabalhadora e independente. Que sempre me elogiou em segredo e que, mesmo não tendo tido a oportunidade de frequentar uma faculdade, me deu todos os recursos para que o meu percurso académico fosse vivido.

Por fim, ao Quim, pela eterna paciência e pelo amor incondicional. Por ser sempre a janela aberta quando todas as portas estão fechadas. Esta conquista não é só minha, é nossa! Obrigada!

Índice

Lista de figuras

Lista de tabelas

Abreviaturas e Símbolos

Lista de abreviaturas

CNN	*Convolutional Neural Network*
DNN	*Dense Neural Network*
IA	Inteligência Artificial
LR	Linear Regression
MIBEL	Mercado Ibérico da Eletricidade
LSTM	*Long short-term memory*
PD	Produção Distribuída
SEE	Sistema Elétrico de Energia

Capítulo 1

1. Introdução

1.1 - Enquadramento e objetivos

A sociedade atual é cada vez mais dependente da energia elétrica, quer a nível doméstico ou industrial. Esta dependência resulta num problema para o sistema elétrico de energia, que necessita de uma gestão das unidades de produção de forma a alimentar de forma eficiente todos os consumidores.

O preço da eletricidade está fortemente dependente do tipo de unidades produtoras que a geram. No caso das centrais convencionais, o custo de produção depende do custo dos combustíveis e também da sua eficiência. Por outro lado, sabe-se que o aumento da geração de energia elétrica a partir de fontes renováveis provoca geralmente a diminuição do preço de mercado. Ora, a produção de energia por este tipo de fontes está dependente das condições climatéricas, provocando variações significativas no preço.

A previsão de preço de mercado é cada vez mais necessária para que os produtores e os comercializadores possam apresentar as suas propostas de forma adequada, e de modo que não sejam penalizados no caso do erro de previsão ultrapassar os limites regulamentares permitidos. Atualmente, esta previsão é feita com recurso a diversas técnicas que vão desde a regressão linear até às redes neuronais convencionais.

Numa primeira fase desta dissertação, será feita uma comparação entre métodos convencionais e métodos de *Deep Learning*. É ainda realizada uma análise à variação dos resultados de acordo com a seleção dos hiperparâmetros das três técnicas de *Deep Learning* consideradas.

Na grande maioria dos métodos (convencionais ou não) utilizados para a previsão de preço de mercado, um dos focos principais é a seleção ótima das variáveis e não necessariamente a escolha do modelo estatístico. Nesta dissertação pretende-se desafiar o formato convencional da previsão, tentando-se eliminar em parte o trabalho de seleção de variáveis, olhando para as *features* como um todo e aplicando métodos de *Deep Learning*, dando ênfase às

Convolutional Neural Network (CNN), normalmente aplicadas com sucesso no processamento e análise de imagens e às *Long short-term memory* (LSTM), uma arquitetura de rede neural artificial recorrente que, contrariamente às redes neurais *feedforward* padrão, é capaz de estabelecer conexões de *feedback*. As LSTM são normalmente aplicadas em tarefas como o reconhecimento de manuscrito, o reconhecimento de voz e a deteção de anomalias no tráfego de rede (em sistemas de deteção de intrusão, por exemplo)[1].

1.2 – Estrutura da dissertação

Esta dissertação é composta por 5 capítulos.

No capítulo 1 é feito o enquadramento no tema das previsões do preço *spot* e são indicados os objetivos desta dissertação.

O capítulo 2 apresenta de forma sucinta os conceitos fundamentais para compreensão do problema em análise. Numa primeira fase, comenta-se a estrutura do Mercado Ibérico da Energia Elétrica (MIBEL) e a sua forma de atuação atual, bem como se dão a conhecer os novos limites para o preço *spot* assumidos pelo operador de mercado no futuro próximo. Para além disso, é feita uma abordagem geral ao tema da Inteligência Artificial (IA) e a sua aplicação na previsão do preço da eletricidade.

No capítulo 3 é analisada a metodologia de trabalho, apresentando-se os diferentes conjuntos de variáveis de *input* utilizadas nos modelos de previsão, dando-se a conhecer as estruturas das redes e fazendo um resumo das técnicas de *Deep Learning* usadas na dissertação. Ainda neste capítulo é feita uma análise geral aos dados de *input* e apresentada a sua divisão em treino, validação e teste.

No capítulo 4 são apresentados os resultados decorrentes da aplicação das várias técnicas de previsão aos conjuntos de dados disponíveis e são usados modelos de referência para comparação de resultados.

Finalmente, no capítulo 5 é apresentada uma conclusão do trabalho desenvolvido ao longo desta dissertação.

2. Conceitos

2.1 - MIBEL

Em 1995, o Mercado Elétrico Português deu o primeiro passo em prol da liberalização do setor energético, levando à existência de concorrência no fornecimento de energia. Em 2006, esta medida foi alargada a todos os consumidores de Portugal Continental, permitindo competitividade nos preços e na qualidade do serviço.

A liberalização do mercado possibilitou ao consumidor a livre escolha da empresa comercializadora pretendida para o fornecimento de energia, no entanto, não resolveu o problema de fiabilidade do Sistema Elétrico de Energia (SEE) motivado pela crescente necessidade de abastecimento de energia. Assim, em prol de uma maior eficiência do sistema energético, verificou-se um crescimento significativo do número de produtores, incluindo produtores de menor potência, resultando num novo conceito energético denominado Produção Distribuída (PD).

A PD provocou uma alteração significativa do SEE. Normalmente associada à produção de energia a partir de fontes de energia renovável, a PD é ainda caracterizada por ser implementada junto do local de consumo, diminuindo o trânsito de potência na rede e os custos associados a esse transporte de energia. Por outro lado, este novo conceito fortemente dependente das condições climatéricas é caracterizado por uma grande volatilidade e por transportar para o sistema problemas na regulação de tensão.

Todas estas alterações no SEE associadas ainda ao desenvolvimento tecnológico e às crescentes preocupações ambientais resultaram numa alteração da arquitetura do sistema elétrico de energia, que adquiriu uma estrutura desverticalizada, complementada com novos esquemas tarifários e novos agentes reguladores

No final de 2001, os governos de Portugal e Espanha assinaram de um memorando entre as Administrações para a criação do Mercado Ibérico da Eletricidade (MIBEL) que seria implementado em janeiro de 2003, com visa à constituição de um mercado regional de energia elétrica na Península Ibérica com interligação às redes transeuropeias e magrebinas de transporte e distribuição de eletricidade. Esta organização permite que os consumidores ibéricos sejam livres de escolher um qualquer produtor ou consumidor que atue em Portugal ou Espanha.[4]

O MIBEL considera as seguintes plataformas de negociação:[4]

- Contratos bilaterais entre comercializadores e consumidores;
- *Pool* comum (*day-ahead market*);
- Mercados de ajuste – intradiários;
- Mercado comum para contratos de futuro a curto e longo prazo.

De forma muito simplista, o Modelo em *Pool* funciona com base na apresentação de propostas de venda e compra de energia, por parte de empresas produtoras e de empresas comercializadoras ou consumidores elegíveis, respetivamente. O Operador de Mercado tem o compromisso de manter o equilíbrio entre produção e consumo, sendo responsável pela realização do despacho centralizado da eletricidade baseado nas propostas de compra e venda. Na prática, esta entidade ordena as propostas por ordem de preço e, numa segunda fase, determina o preço final de cada negociação. O valor que daqui resulta é o valor que as cargas e os geradores terão de pagar e receber por MWh, respetivamente. [5][6]

No MIBEL, o processo de negociação é designado por *day-ahead market*, isto porque é realizado para o dia seguinte (dia n). Isto significa que, no dia n-1, se determinam as propostas a serem aceites, os preços e as quantidades de energia transacionada para o dia n. Por este facto, no Mercado Ibérico, o horizonte de tempo interessa sobretudo a previsão de curto prazo, ou seja, mercados diários e intradiários. Este método de negociação faz com que as informações probabilísticas sobre os valores futuros do preço *spot* da energia elétrica sejam muito valiosas para o planeamento das empresas de geração de energia (renovável ou convencional), bem como para implementar reações de resposta à variação de consumo.

A necessidade crescente de conhecimento acerca do preço *spot* futuro, aliado ao desenvolvimento científico e tecnológico, tornou possível a aplicação da Inteligência Artificial para a previsão desta variável central.

Nesta dissertação consideram-se as regras atuais para o MIBEL, caracterizado pela negociação do mercado diário sujeita a um intervalo de 0 a 180 euros por megawatt hora (MWh). No entanto, a partir de 6 de julho de 2021, o operador de mercado aceitará ofertas entre os 500 euros negativos até aos 3000 por MWh. Esta medida visa o alinhamento do Mercado Ibérico com outros mercados grossistas de eletricidade do resto da Europa. Na Península Ibérica já foram registadas várias horas com preços *spot* muito próximos do zero, em momentos de abundante disponibilidade de recurso eólico e hídrico. Em contrapartida, o preço *spot* raramente atinge o limite máximo estipulado atualmente. [7]

2.2 – Inteligência Artificial, *Machine Learning* e *Deep Learning*

A Inteligência Artificial (IA) é um conceito desejado pela comunidade científica desde há muitos anos e "define-se pela criação de máquinas dotadas da capacidade de pensar" (*Alex Turing*).

Ainda que inicialmente a IA fosse aplicada na resolução de problemas que podem ser descritos através de regras matemáticas formais, o grande desafio centra-se nos problemas que um ser humano consegue resolver por intuição, como reconhecer sons ou imagens. A primeira solução para este tipo de problemas passa por permitir que os computadores aprendam com a experiência, evitando a necessidade dos humanos especificarem formalmente todo o conhecimento de que um computador precisa. A outra solução seria possibilitar às máquinas o

entendimento do mundo como uma hierarquia de conceitos, i.e., transformar conceitos complexos numa combinação de conceitos mais simples [2].

Ora, um ser humano é capaz de pensar e tomar decisões porque, ao longo da vida, vai aumentando o seu conhecimento. A maior dificuldade apresentada neste contexto da criação de máquinas capazes de pensar e tomar decisões passaria, então, na passagem formal de conhecimento para o computador. Assim, a IA teria de ser capaz de adquirir o seu próprio conhecimento, extraindo padrões através de um conjunto de dados. Esta capacidade é designada por *Machine Learning*[1]. O *Machine Learning* permitiu aos computadores a tomada de decisões que parecem subjetivas. O *Deep Learning* é uma classe de algoritmos de *Machine Learning* caracterizada pela aplicação de várias camadas de aprendizagem aos dados brutos, de forma a extrair conhecimento de uma forma cada vez mais "profunda" [3].

De uma forma geral, o desempenho dos algoritmos de *Machine Learning* tem uma forte dependência da representação dos dados de *input*. Na ciência dos computadores, operações como pesquisar uma seleção de dados podem decorrer de modo exponencialmente mais rápido se estes estiverem estruturados e indexados de forma inteligente. Esta ideia pode ser observada na **Figura 1**. Para além da representação ótima dos dados, também a escolha das variáveis se torna fundamental. Por exemplo, se o objetivo é determinar o preço da energia elétrica no MIBEL, devemos considerar variáveis como a quantidade de energia gerada a partir de fontes renováveis, cujo aumento resulta normalmente numa diminuição do preço. No entanto, neste e noutros casos, torna-se difícil entender quais as variáveis que devem ser incluídas para a previsão.

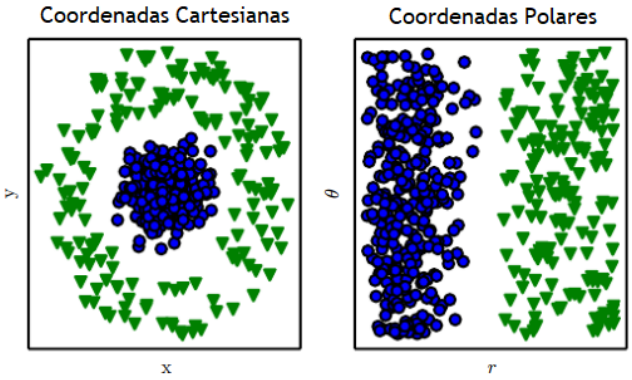

Figura 1 – Influência da representação na resolução de problemas: imaginemos que queremos separar duas categorias de dados separando-as por uma linha. Usando coordenadas cartesianas, esta missão seria impossível. Usando coordenadas polares, este problema fica de fácil resolução.[1]

[1] Em português, usa-se com frequência o termo Aprendizagem Automática. Nesta dissertação, optou-se por manter a nomenclatura inglesa, dado ainda não haver tradução "oficial" de diversas designações.

Na construção de um algoritmo, o objetivo deve passar por separar os fatores de variação que explicam os dados observados. Por exemplo, ao analisar a imagem de um carro, os fatores de variação incluem a posição do carro, a sua cor e o ângulo e intensidade da luz solar. Neste contexto, uma das maiores dificuldades assenta no facto dos fatores de variação influenciarem muitas vezes um grande conjunto de dados. Continuando com o exemplo da imagem de um carro, a cor observada está dependente da altura do dia em que a observamos, enquanto a forma do objeto está dependente do nosso ângulo de visão. Assim, torna-se importante ter a capacidade de separar os fatores de variação que são realmente importantes, daqueles que não aumentam o conhecimento da máquina.[1]

Os métodos de *Deep Learning* tentam resolver estes problemas de representação e de escolha de fatores de variação pelo facto de permitirem que o computador crie conceitos complexos a partir de conceitos mais simples. Em termos gerais, uma imagem passa por uma série de camadas ocultas que permitem a extração de conhecimento cada vez mais abstrato, i.e., a primeira camada é capaz de reconhecer bordas, comparando o brilho de um pixel com os pixels vizinhos. A segunda camada, que já adquiriu o conhecimento da camada anterior, é capaz de descrever cantos e contornos estendidos. Numa terceira camada, já pode ser possível a identificação de partes inteiras de objetos específicos. A passagem dos dados por estas camadas permite obter, não só uma representação mais abstrata dos dados de entrada, como também a sua representação num espaço de menor dimensão.

3. Metodologia

A previsão do preço *spot* será realizada com recurso a dois tipos gerais de técnicas:

- A primeira, apenas recorrendo a técnicas convencionais de previsão, tendo-se escolhido a regressão linear para o efeito;
- A segunda, recorrendo a métodos de *Deep Learning*, nomeadamente, DNN, LSTM e CNN.

O objetivo do primeiro estudo é perceber se as técnicas mais sofisticadas de *Deep Learning* têm alguma vantagem em relação à regressão linear por serem capazes de captar relações mais complexas entre variáveis.

Num segundo estudo, o objetivo é perceber até que ponto a utilização de redes convolucionais (CNN) permite aliviar a tarefa de seleção de variáveis de entrada (*input*).

3.1 – *Features*

As variáveis de *input* foram obtidas através da plataforma Sentinel (do INESC TEC) e dizem respeito ao intervalo entre 2015 e 2020. A partir daqui, foi realizada uma análise aos dados e considerados apenas os dias completos (24h) onde não existissem lacunas em qualquer uma das variáveis consideradas.

Para cada um dos métodos descritos serão feitos testes faseados cuja diferença se encontra nas variáveis usadas como *inputs* (*features*):

- A primeira fase caracteriza-se pela aplicação do conjunto A como variáveis de *input*, sendo elas: Preço *spot* e variáveis cronológicas (hora, dia da semana e mês). A inexistência de informação exógena torna o modelo mais simples e com menor custo;
- Numa segunda fase, que perfaz o conjunto B de *inputs*, as variáveis foram escolhidas de acordo com o artigo [8], cujas séries de dados são em tudo semelhantes às disponibilizadas para o desenvolvimento desta dissertação, ainda que os métodos aplicados não recorram a *Deep Learning*;
- Por fim, é feito o teste com recurso a todas as variáveis disponíveis, sendo este conjunto definido pela letra C.

A **Tabela 1** indica as variáveis usadas em cada um dos conjuntos de *input*. **A Figura 2** representa o diagrama de evolução da metodologia explicado nos parágrafos anteriores.

Tabela 1 - Variáveis do sistema usadas como *features* para cada um dos conjuntos A, B e C

Disponibilidade	Variáveis	ID	A	B	C
D	Preço *spot* MIBEL	P	X	X	X
D+1	Velocidade do vento PT	V_{PT}			X
	Velocidade do vento ES	V_{ES}			X
	Temperatura PT	T_{PT}			X
	Temperatura ES	T_{ES}			X
	Irradiância PT	I_{PT}			X
	Irradiância ES	I_{ES}			X
D-1	Geração de energia a partir do carvão ES	C_{ES}			X
D-6	Preço *spot* MIBEL	P	X	X	X
	Geração de energia a partir de centrais hidrelétricas reversíveis PT	HS_{PT}			X
	Geração de energia a partir de centrais hidrelétricas reversíveis ES	HS_{ES}			X
	Geração de energia a partir de reservatórios hídricos PT	HR_{PT}		X	X
	Geração de energia a partir de reservatórios hídricos ES	HR_{ES}		X	X
D+1	Previsão de carga PT	CF_{PT}			X
	Previsão de carga ES	CF_{ES}		X	X
	Previsão de geração eólica PT	EF_{PT}			X
	Previsão de geração eólica ES	EF_{ES}		X	X
	Previsão de geração solar PV PT	SF_{PT}			X
	Previsão de geração solar PV ES	SF_{ES}		X	X
	Previsão de geração solar térmica PT	STF_{ES}		X	X
Cronológicas	Hora	H	X	X	X
	Dia da semana	D	X	X	X
	Mês	M	X	X	X

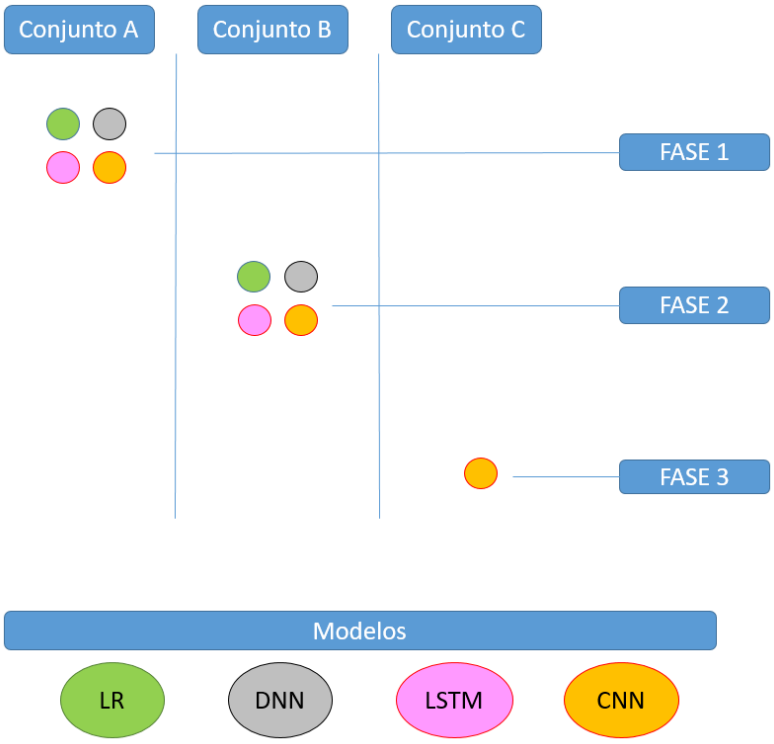

Figura 2 - Especificação dos cenários a analisar

3.2 – Estrutura das redes

A estrutura das redes é apresentada nesta dissertação da forma que se segue, sendo que depende do tipo de metodologia aplicada:

- DNN: nº de neurónios por camada – nº de camadas escondidas – função de ativação – *dropout*;
- LSTM: nº de unidades por camada – nº de camadas escondidas – função de ativação – *dropout*;
- CNN: nº de filtros – *kernel size* (altura, largura) – função de ativação – *dropout* (para cada uma das camadas da rede neuronal convolucional).

Onde o *dropout* é uma técnica de regularização que previne *overfitting* e representa a percentagem de ligações que, em cada ciclo do treino, são ignoradas. Esta eliminação de ligações permite que a rede adquira uma capacidade de generalização muito superior.

Para além destas características estruturais das redes, é de salientar que, na CNN, o resultado das sucessivas convoluções acaba por resultar numa camada denominada *flatten*, cujo objetivo é reduzir as dimensões espaciais dos dados a uma camada "plana". Após este processo, a camada de *output* da rede é uma DNN com 24 neurónios (24 horas do dia) com uma função de ativação usualmente não linear (*Leaky ReLU*) [9].

Em todas as arquiteturas de *Deep Learning*, nas camadas escondidas, foi vantajoso usar como função de ativação as funções *ReLU* ou *Leaky ReLU*, invés de *Tanh* ou *Sigmoid*, porque permitiram uma convergência mais rápida dos modelos.

O **Gráfico 1** e o **Gráfico 2** permitem a observação das duas funções de ativação usadas.

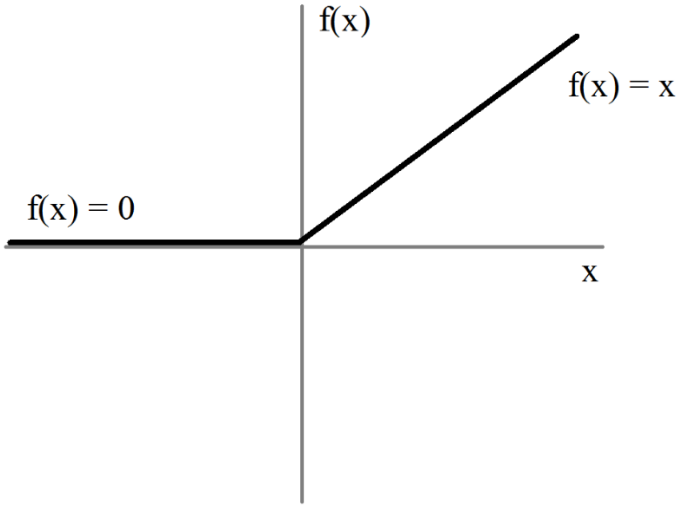

Gráfico 1 - Função de ativação *ReLU* [10]

A função *ReLU* funciona como um retificador e define-se por:

$$ReLU(x) = \max\{0, x\} \quad (1)$$

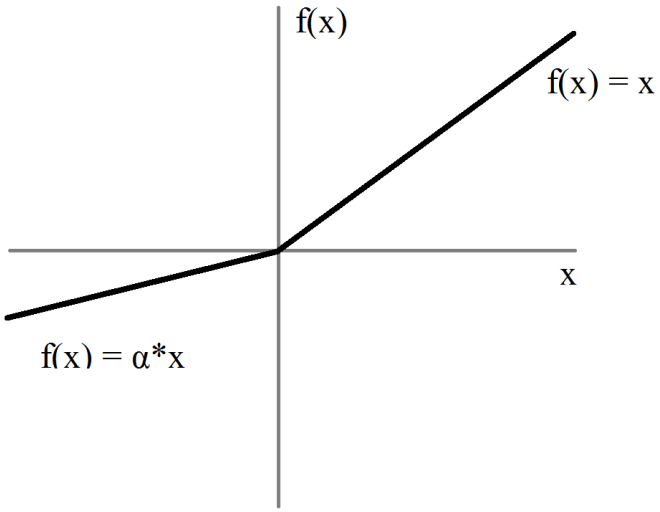

Gráfico 2 - Função de ativação Leaky ReLU [10]

A função *Leaky ReLU* é idêntica à função *ReLU*, no entanto, na parte negativa do eixo das abcissas, a sua derivada ainda é positiva. Matematicamente, caracteriza-se segundo a **Equação (2)**.

$$LeakyReLU(x, \alpha) = \max\{\alpha x, x\} \quad , \quad \alpha \epsilon]0,1[\quad (2)$$

Onde α é um hiperparâmetro denominado por vazamento, normalmente de valor muito baixo. No caso desta dissertação, este valor é de 0.1.

Para avaliar a qualidade das abordagens escolhidas para a previsão de preço *spot*, é utilizado um critério que avalia os resultados comparando-os com o valor dos preços de eletricidade que se verificam na realidade. Em todas as redes de *Deep Learning* usadas nesta dissertação, o objetivo passa por encontrar o melhor vetor $\hat{y}_i$ de forma a minimizar a função objetivo apresentada na **Equação (3)**.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{3}$$

Onde N é o número de amostras *(samples)*, y_i é o valor real do preço e $\hat{y}_i$, a previsão que resulta da aplicação de cada uma das redes.

3.3 – *Dense neural networks*

As camadas das *dense neural networks* são do tipo *fully conected,* i.e., cada neurónio numa dada camada recebe de entrada o output dos neurónios de sua camada anterior, tal como pode ser identificado na **Figura 3.**

Em termos matemáticos, o valor de entrada de cada neurónio é calculado pelo somatório dos produtos de cada neurónio anterior por um peso. Estes pesos são otimizados durante o treino.

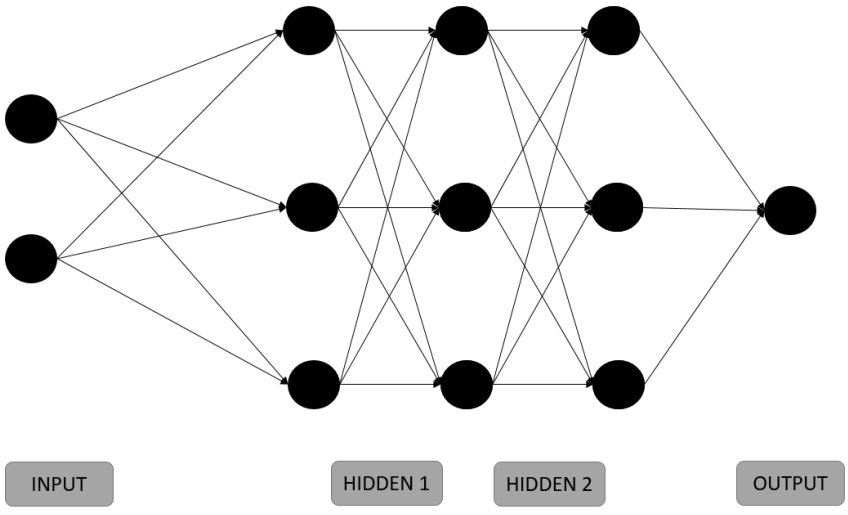

Figura 3 – Diagrama exemplificativo de uma dense neural network

Cada neurónio de uma DNN recebe um conjunto de valores de x (numerados de 1 a N) como entrada e é constituído por um conjunto de pesos, w, e um *bias*, b, que mudam durante o processo de aprendizagem. A cada iteração, o neurónio calcula uma média ponderada dos

valores do vetor x, com base num vetor de pesos atual, w, e adiciona o erro de enviesamento (erro de *bias*). Finalmente, o resultado desse cálculo é passado por uma função de ativação não linear (g) [11]. No caso das DNN, essa função de ativação é a *Leaky ReLU*. Os parâmetros w e b são atualizados de forma a minimizar a função objetivo apresentada na **Equação (3).** A **Figura 4** representa o que foi enunciado neste parágrafo.

$$z = w_1x_1 + w_2x_2 + w_3x_3 + \ldots + w_nx_n = \mathbf{w}^T \cdot \mathbf{x}$$

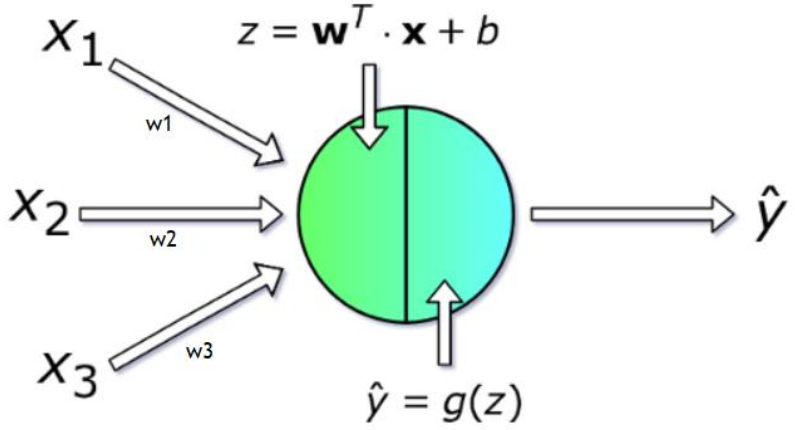

Figura 4 - Diagrama de funcionamento de um neurónio isolado [11]

Para além dos pesos, a rede apresenta hiperparâmetros que podem ser ajustados, como o número de neurónios em cada camada, o número de camadas, a função de ativação e a probabilidade de *dropout*. Por fim, existem diversos algoritmos de treino que podem ser usados na otimização do modelo. Nesta dissertação foi considerado o *Adam*, por ser reconhecido como dos mais eficientes. Este tipo de redes são frequentemente aplicadas a domínios que incluem visão computacional - campo científico interdisciplinar que trata de como os computadores podem obter compreensão de alto nível a partir de imagens ou vídeos digitais -, reconhecimento de fala, processamento de linguagem natural – cujo objetivo é permitir ao computador a compreensão de conteúdos de documentos incluindo as subtilezas contextuais da linguagem dentro deles -, tradução automática, bioinformática – por exemplo na análise de mutações em cancros -, projeto de medicamentos, análise de imagens, inspeção de materiais e programas de jogos de tabuleiro. Neste último, as DNN são capazes de produzir resultados comparáveis e, em alguns casos, superar o desempenho de humanos especialistas.[12]

3.4 – *Long Short-Term Memory*

As *Long Short-Term Memory* são redes neuronais recorrentes (RNN) que, contrariamente às redes neurais *feedforward* padrão, permitem conexões retroativas (*feedback*), como sugerido

pela **Figura 5**. As LSTM são um tipo especial de RNN, capaz de aprender dependências de longo prazo e, por este facto, são amplamente usadas em modelos de linguagem que tentam prever a próxima palavra. Vejamos o exemplo: "Eu vivo em Portugal. Eu falo fluentemente …". As últimas palavras (informações de curto prazo) sugerem que a próxima palavra será um idioma. No entanto, só com recurso à frase anterior (informações de longo prazo) é que conseguimos entender que esse idioma é o português. [15]

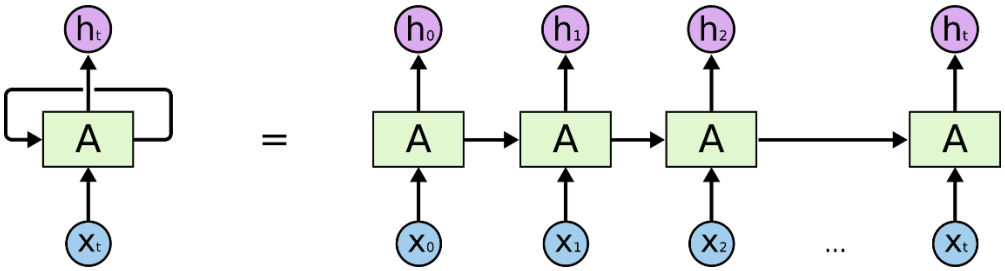

Figura 5 – Diagrama exemplificativo das *Long Short-Term Memory* [13]

A saída de informação da rede recorrente é determinada pela entrada X_t, mas também pelos estados obtidos nas saídas de redes anteriores h_{t-1}, h_{t-2}, etc.

Podemos analisar a LSTM ao nível de uma célula, tal como apresentado na **Figura 6**. Esta célula consiste em:

- *Forget Gate*: as informações que não são úteis no estado da célula são removidas com o *forget gate*. Esta porta filtra as informações de entrada atual (X_t) e de saída anterior (h_{t-1}) e decide qual deve ser lembrada ou descartada.
- *Input Gate:* a adição de informações úteis ao estado da célula é feita pelo *input* gate, que controla o fluxo das ativações de entrada na célula de memória.
- Output Gate: que controla o fluxo de saída de ativação da célula.

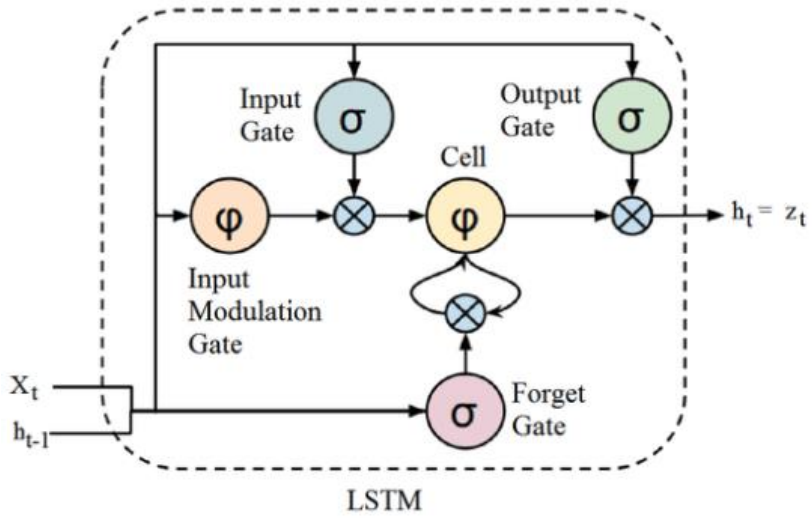

Figura 6 – Diagrama de funcionamento de uma célula isolada [13]

3.4 - *Convolutional Neural Network*

As redes neuronais convolucionais são estruturas de *Deep Learning* inspiradas na organização de córtex visual dos animais. Relativamente a outros métodos, requerem menos pré-processamento. A **Figura 7** representa um exemplo de aplicação de redes neuronais convolucionais e o esquema estrutural da rede. Neste exemplo, a função de ativação é a *ReLU*.

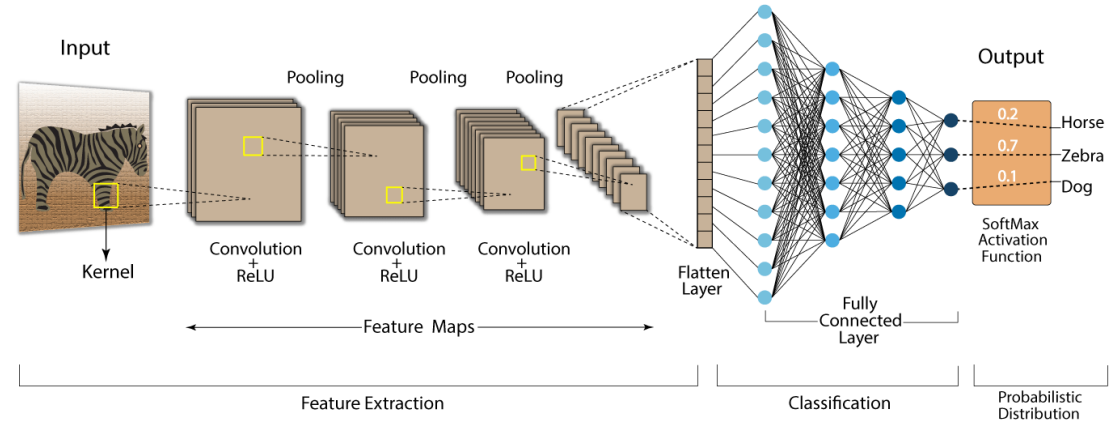

Figura 7 - Esquema da estrutura da rede de uma CNN para um caso de identificação de animais numa imagem [14]

Estas redes são caracterizadas, nas primeiras camadas, por uma operação matemática denominada convolução. Existem diversos tipos de convoluções, sendo que nesta dissertação é usado o mais popular. A convolução é um processo onde se considera uma pequena matriz de números (filtro), que é aplicada a um *input*, transformando-o com base nos valores desse mesmo filtro. Os valores do resultado das convoluções são calculados de acordo com a **Equação (4)** onde o *input* é representado por f e o filtro, por h. A matriz de resultados, G, tem a forma (mxn).

$$G[m,n] = (f * h)[m,n] = \sum_j \sum_k h[j,k] f[m-j, n-k] \tag{4}$$

A **Figura 8** demonstra o funcionamento prático de uma convolução. Nesta dissertação foram realizadas convoluções com filtros de tamanhos diferentes, no entanto, os filtros verticais (largura=1) mostraram-se os mais vantajosos no problema em estudo.

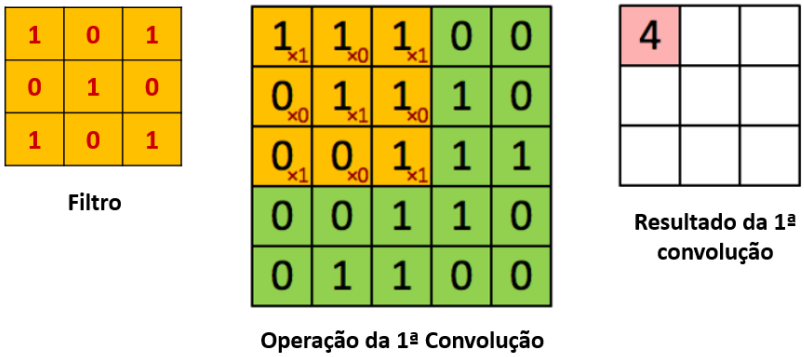

Figura 8 - Exemplificação de uma convolução num conjunto de dados; tamanho do filtro (3,3)
G[1,1] = 1x1+0x1+1x1+0x1+1x1+0x1+1x0+0x0+1x1=4

3.4 - Caracterização geral dos dados

O *dataset* original contempla dados no intervalo de 2015 a 2020. Em todos os testes realizados considerou-se a divisão dos dados da forma apresentada na **Figura 9**. Esta divisão permitiu que o *dataset* de validação e o *dataset* de teste tivessem uma sequência cronológica de aproximadamente um ano.

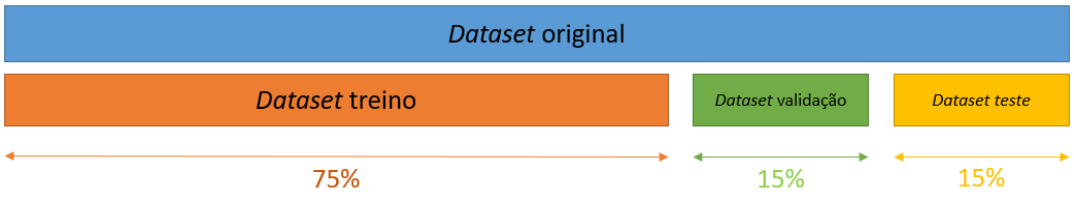

Figura 9 - Divisão em treino, validação e teste

A análise da média do preço *spot* nos três *datasets* secundários permitiu verificar que existem discrepâncias consideráveis, especialmente no que diz respeito ao *dataset* de teste, como verificado na **Tabela 2** e no **Gráfico 3.** Este último apresenta os dados relativos ao ano de 2020, um ano atípico, provavelmente devido à situação pandémica provocada pela COVID-19, caracterizado pela descida acentuada da média do preço da energia elétrica. Este motivo, agregado à tendência natural de aumento do erro de teste face ao erro de treino e validação, pode explicar algumas das diferenças apresentadas para os erros no decorrer desta dissertação.

Tabela 2 - Análise da média do preço *spot* para os diferentes *datasets*

***Dataset* treino (€/MWh)**	***Dataset* validação (€/MWh)**	***Dataset* teste (€/MWh)**
47.86	55.76	38.79

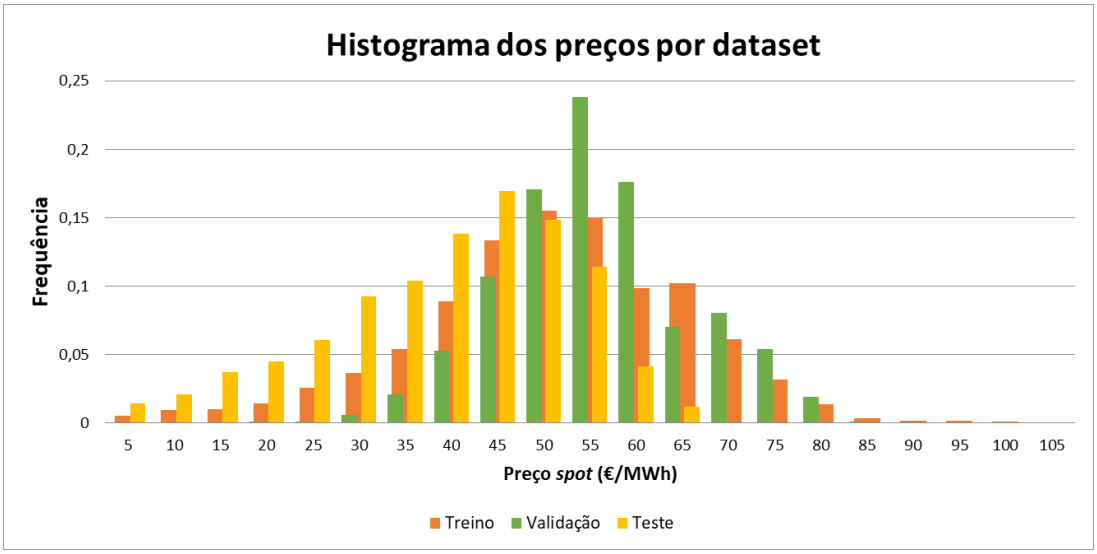

Gráfico 3 – Histograma dos preços *spot* da eletricidade por *dataset*

Os gráficos seguintes apresentam a separação dos dados de acordo com o apresentado na **Figura 9** e usando o mesmo registo de cores. É de notar que uma parte dos dados foi eliminada no processo da análise por não conter dias completos (medidas apresentadas para apenas algumas horas dos dias) ou ainda por ocorrerem erros nas medições (Nan).

A visualização dos dados permite verificar que durante o ano 2020 (referente ao período de teste), o preço *spot* da eletricidade diminuiu acentuadamente. Uma das maiores quebras ocorreu entre abril e junho de 2020, período no qual o país se encontrava em confinamento, no qual muitas indústrias se viram obrigaram a encerrar a atividade por questões pandémicas.

Para além disso, a visualização do **Gráfico 4,** que apresenta o *dataset* original, permite concluir que os meses de fevereiro a abril são caracterizados por descidas habituais do preço, motivado possivelmente pela elevada precipitação em simultâneo com albufeiras cheias e bom nível de produção eólica.

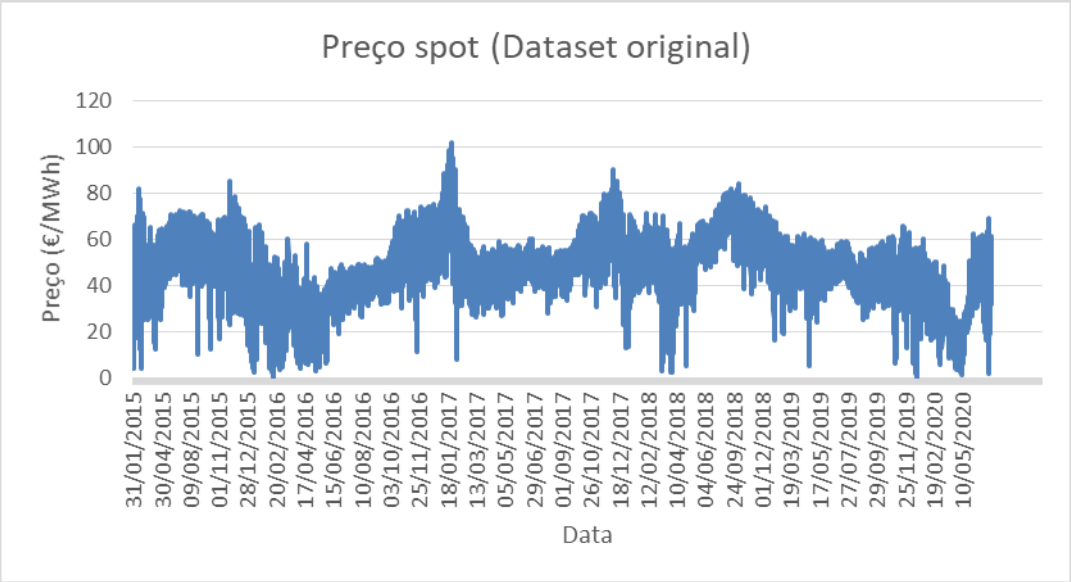

Gráfico 4 - Preço *spot* contemplado no *dataset* original

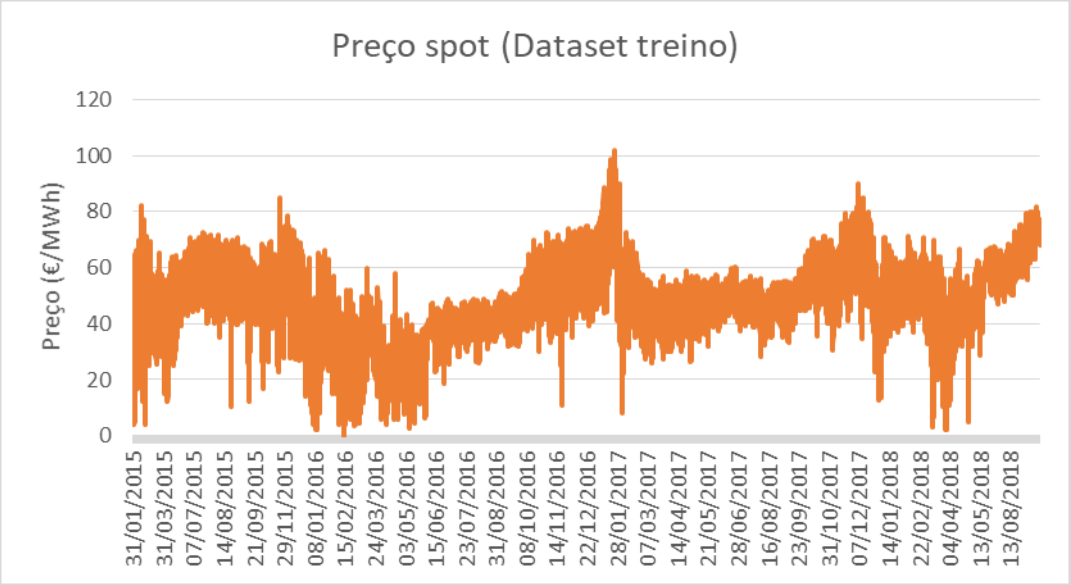

Gráfico 5 - Preço *spot* contemplado no *dataset* de treino

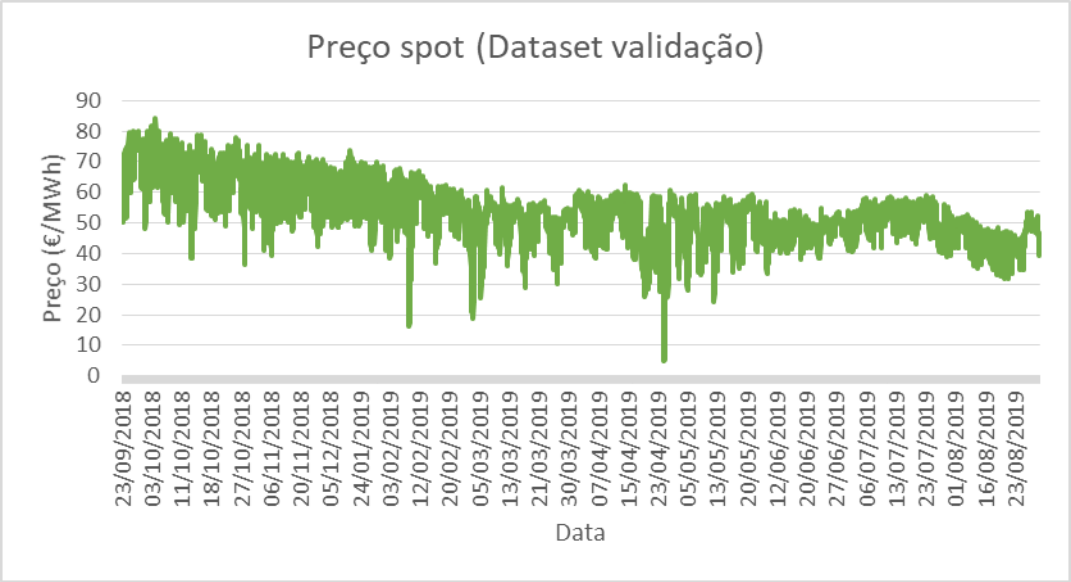

Gráfico 6 – Preço *spot* contemplado no *dataset* de validação

É possível verificar que o período de validação é caracterizado por uma tendência decrescente bastante estável.

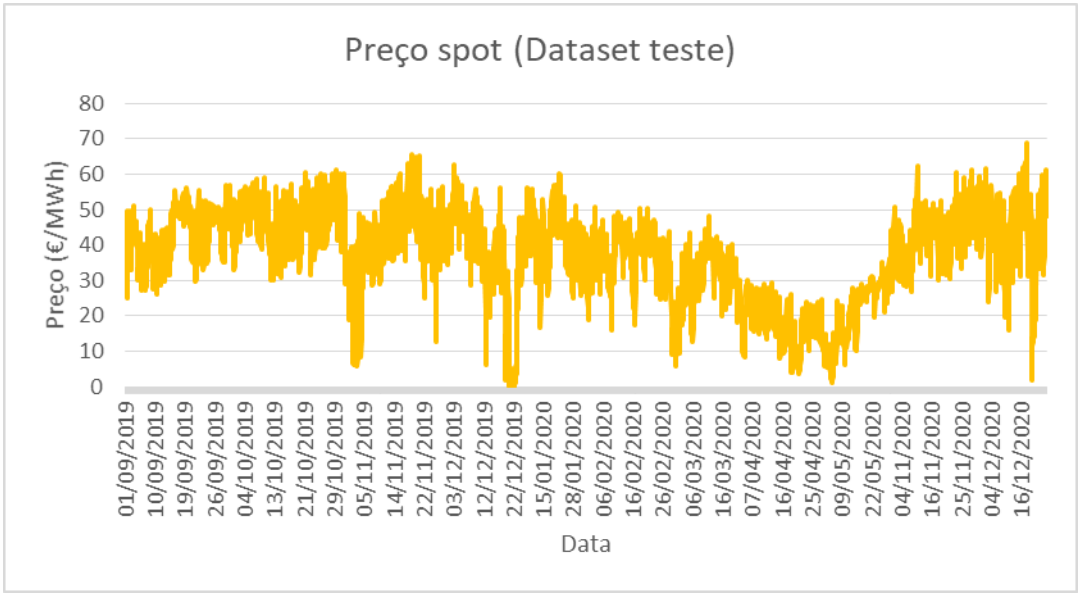

Gráfico 7 – Preço *spot* contemplado no *dataset* de teste

Os gráficos seguintes apresentam os resultados da análise do preço médio em cada mês do ano.

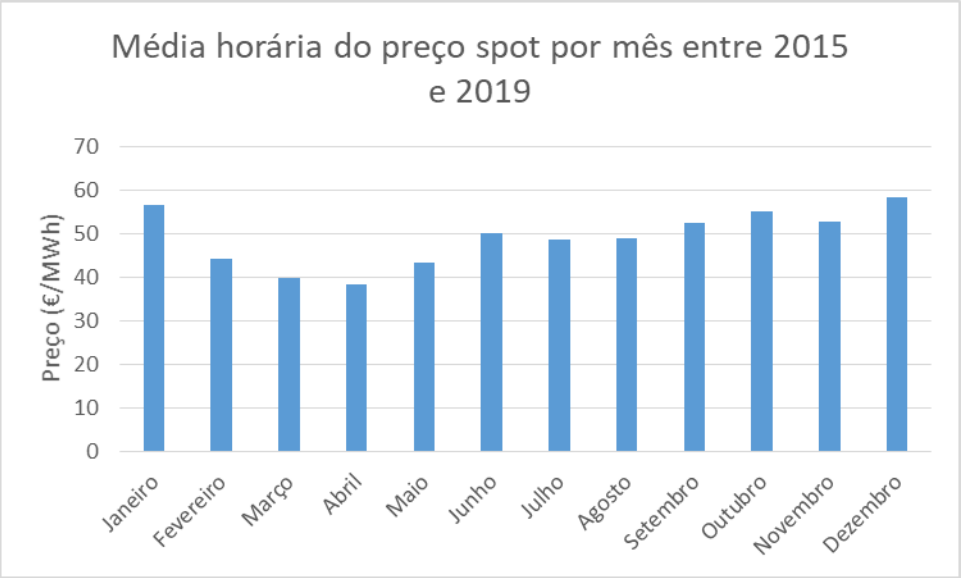

Gráfico 8 - Média horária discriminada por mês do preço *spot* entre os anos 2015 e 2019

É notória a queda do preço nos períodos decorrentes entre fevereiro e abril e junho e agosto. Outubro é habitualmente o mês em que a média do preço é mais elevada ao longo dos cinco anos. Como referido anteriormente, a queda de preço entre fevereiro e abril é justificada simultaneamente pela existência de precipitação, acumulação de água nas albufeiras, alto nível de produção eólica e também pelo aumento da temperatura face aos meses de dezembro e janeiro. O aumento da temperatura, nomeadamente a partir de março, leva à diminuição dos consumos e consequente diminuição do preço *spot*. O **Gráfico 9** representa a potência consumida a partir de reservas de água ao longo de todo o conjunto de dados. A informação deste gráfico é complementada como o seguinte.

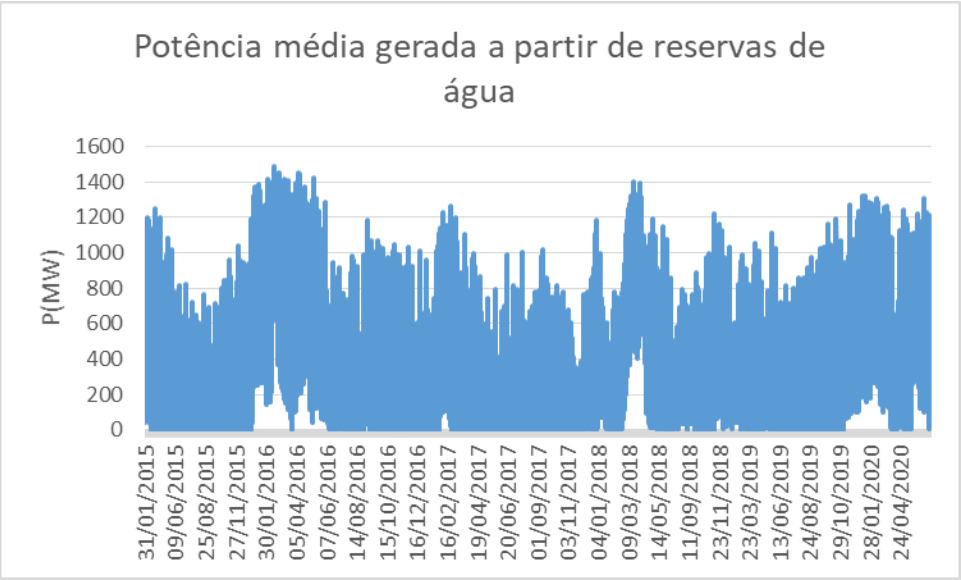

Gráfico 9 - Potência gerada (em MW) a partir das reservas de água

Podemos verificar que os meses de maior consumo a partir de reservas de água se encontram no período de fevereiro a abril, justificando-se, assim, a diminuição do preço no mesmo período. Esta análise permite concluir que o preço é fortemente dependente desta variável "Geração de energia a partir de reservatórios hídricos", considerada nos conjuntos B e C que serão alvo de análise do capítulo seguinte.

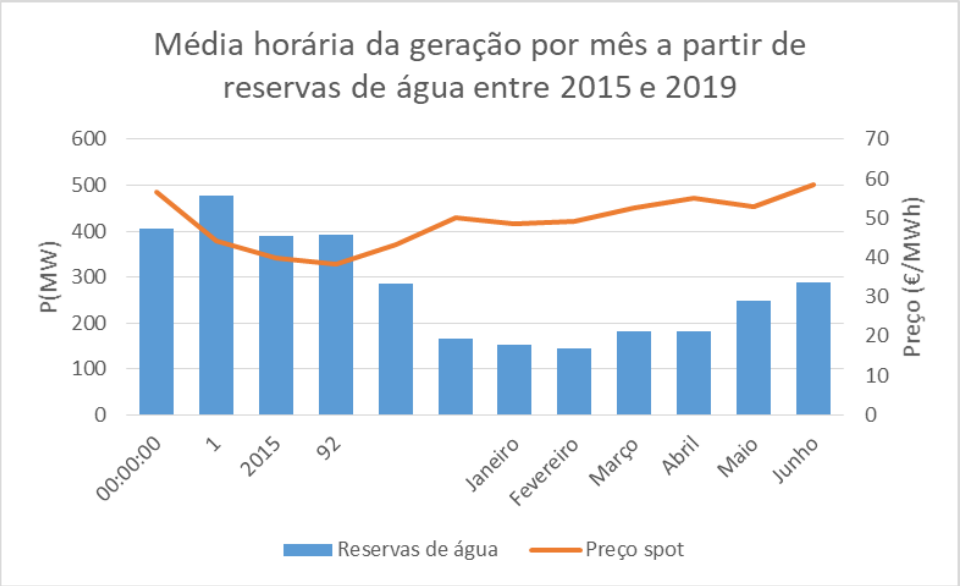

Gráfico 10 - Média horária discriminada por mês da potência gerada a partir de reservas de água no período de 2015 a 2020 combinada com a média horária por mês do preço *spot*

A geração eólica também pode ser analisada de acordo com os gráficos seguintes, visto ser uma variável de grande importância na previsão do preço da energia elétrica.

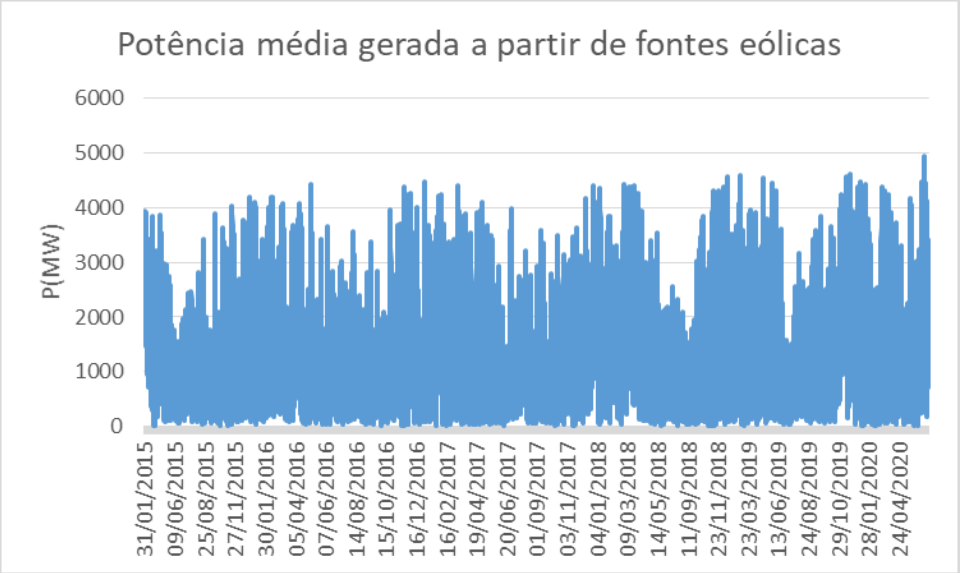

Gráfico 11 - Potência gerada (em MW) a partir de fontes eólicas

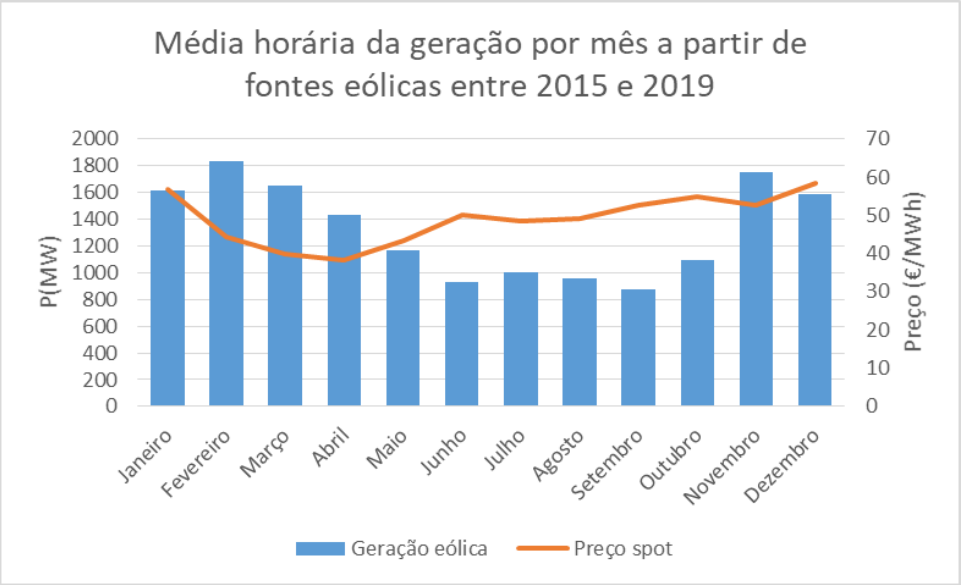

Gráfico 12 - Média horária discriminada por mês da potência gerada a partir de fontes eólicas no período de 2015 a 2020 combinada com a média horária por mês do preço *spot*

Neste caso, verifica-se uma depressão entre os meses de junho e setembro, caracterizados por temperaturas elevadas e ventos mais fracos. Os meses de inverno permitem um aumento da energia produzida a partir de fontes eólicas, contribuindo para a diminuição do preço.

Por fim, podemos analisar o consumo (em MW) durante o período em estudo. A análise do **Gráfico 13** permite concluir que de abril a junho de 2020 ocorreu uma diminuição acentuada do consumo. Esta variação é explicada pela situação pandémica vivida, que levou ao encerramento, mesmo que temporário, de várias indústrias. Ainda que o consumo doméstico possa ter aumentado significativamente, a paragem das indústrias tornou-se evidente durante o confinamento.

O **Gráfico 14** mostra a correlação evidente entre o consumo e o preço *spot*. O aumento do consumo resulta, geralmente, no aumento do preço da energia elétrica.

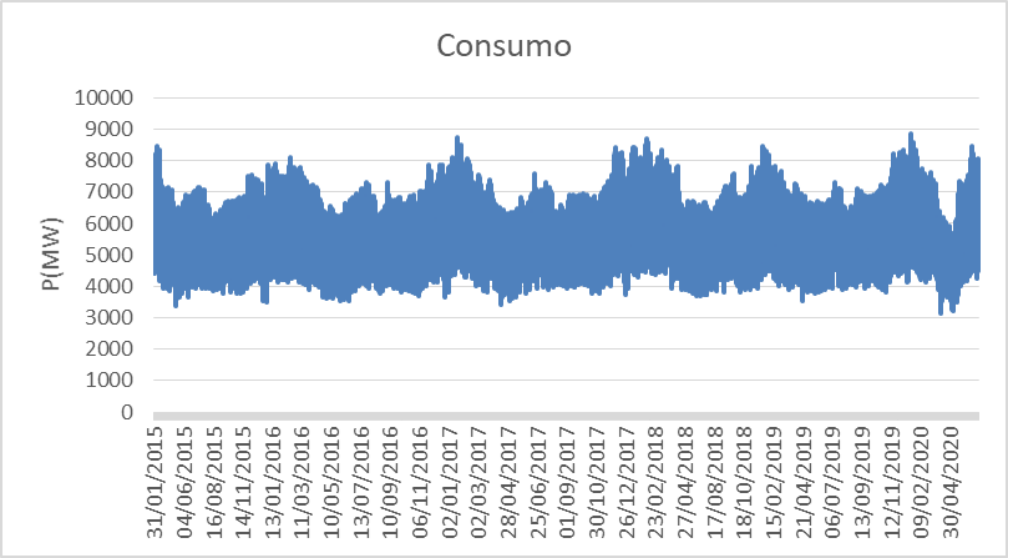

Gráfico 13 - Consumo (em MW)

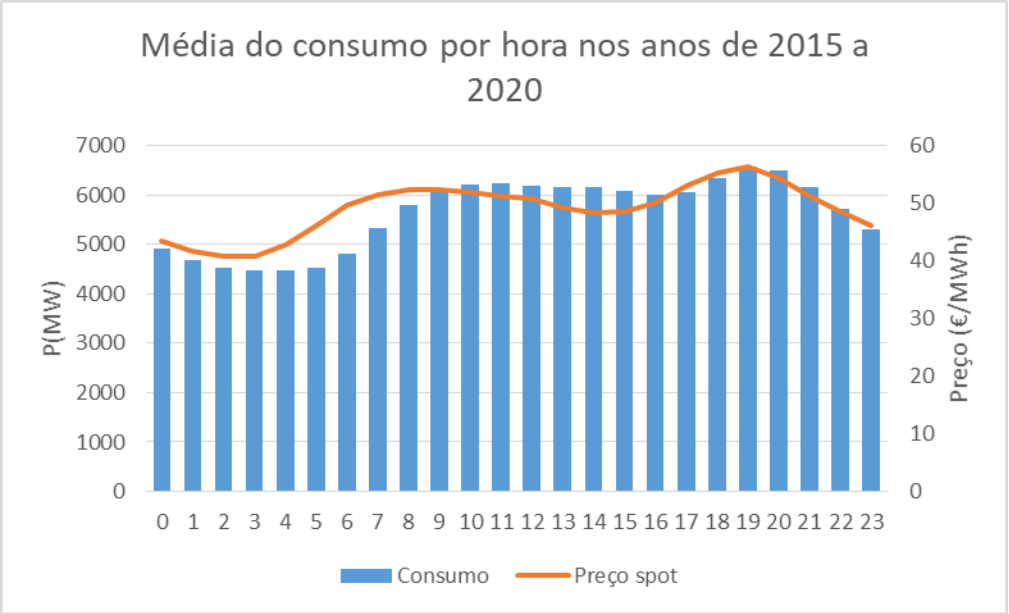

Gráfico 14 - Média do consumo por hora no período de 2015 a 2020

O **Gráfico 15**, retirado do Observatório da Energia, permite visualizar a variação do consumo por setor em 2020, comparativamente ao ano de 2019 para os meses de abril a junho. Esta análise é feita neste período devido ao confinamento decretado no dia 19 de março de 2020. É possível verificar a diminuição acentuada do consumo elétrico industrial e o simultâneo aumento do consumo doméstico. O confinamento resultou ainda na diminuição do consumo nos serviços e nos transportes, motivados pela obrigatoriedade de permanecer em casa.

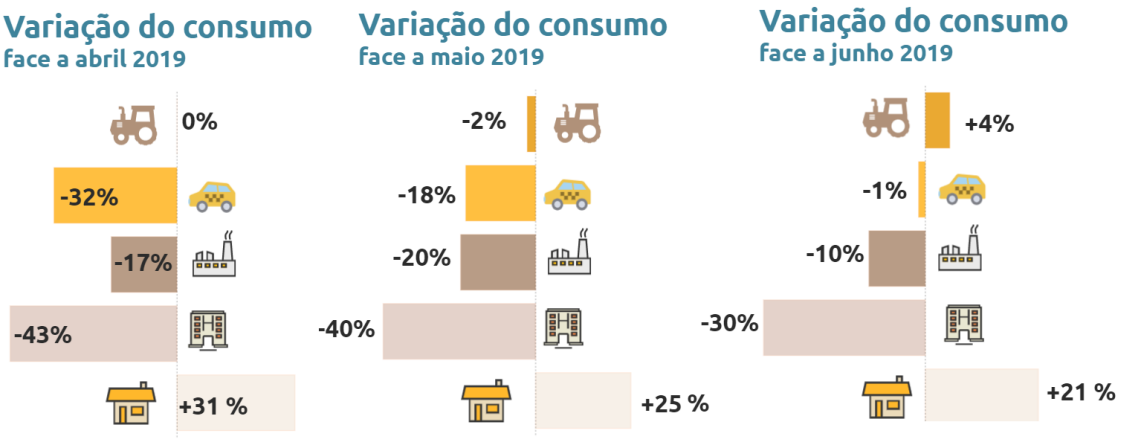

Gráfico 15 - Variação do consumo de eletricidade nos meses de abril a junho de 2020 face ao mesmo período do ano anterior [16][17][18]

4. Resultados

4.1 – Aplicação do conjunto A como variáveis de *input*

A primeira forma de prever o preço *spot* do MIBEL é aquela que se considera de menor custo e menor complexidade. A estes dados serão aplicados métodos cuja dimensão deve ser adequada à complexidade do problema. Para qualquer uma das tecnologias escolhidas foram realizados testes de ajuste de dimensão da rede, sendo apresentada a estrutura que produziu melhores resultados.

Para além de métodos de *Deep Learning*, é usado também o método de regressão linear. Este método não contempla hiperparâmetros e apresenta-se nesta dissertação apenas para estabelecer o patamar de performance, permitindo comparar a previsão de preço com e sem métodos de *Deep Learning*.

Os resultados obtidos com os métodos implementados apresentam-se na **Tabela 3**. Verifica-se que o método da Regressão Linear apresenta a variação mais elevada entre erro de treino e erro de teste (próxima dos 60%). Este método não inclui hiperparâmetros, pelo que não nos é possível ajustar a sua capacidade em prol do aumento da sua capacidade de generalização.

Tabela 3 – Resultados com a aplicação do conjunto A

Método	Estrutura da rede	Erro de treino	Erro de validação	Erro de teste	
		MSE (€/MWh)^2	MSE (€/MWh)^2	MSE (€/MWh)^2	MAE (€/MWh)
LR	-	21.90	29.31	55.68	5.54
DNN	50-1-lrelu-0.2	40.47	30.62	55.99	5.42
LSTM	30-0-lrelu-0.2	29.78	25.12	44.68	5.09
CNN	32-(5,1)-relu-0.2 32-(5,1)-relu-0.2 32-(5-1)-relu-0	35.18	26.92	39.05	4.58

Em termos gerais, as redes com recurso a *Deep Learning* apresentam um erro de validação menor do que o erro de treino (relação atípica), situação que nos leva a crer que os dados do *dataset* de validação serão mais fáceis de prever comparativamente aos outros *datasets*. Esta proximidade entre erro de validação e treino deveria levar a um erro de teste dentro da mesma faixa de valores, facto esse não verificado, possivelmente explicado pela análise da média do preço *spot* feita no ponto 3.3 desta dissertação, especialmente notada neste conjunto de dados de *input* onde não se incluem variáveis exógenas.

O modelo que utiliza as redes convolucionais foi o que apresentou melhores resultados, ainda que a LSTM apresente maior proximidade entre os erros de treino e validação, relação importante na aplicação deste conjunto de dados A. No modelo com CNN, foram feitas experiências com filtros *nxn*, no entanto, os filtros verticais (*nx1*) mostraram-se mais vantajosos na previsão do preço.

O **Gráfico 16** apresenta a comparação entre as evoluções dos erros de treino e validação para o caso da rede que obteve melhores resultados no teste.

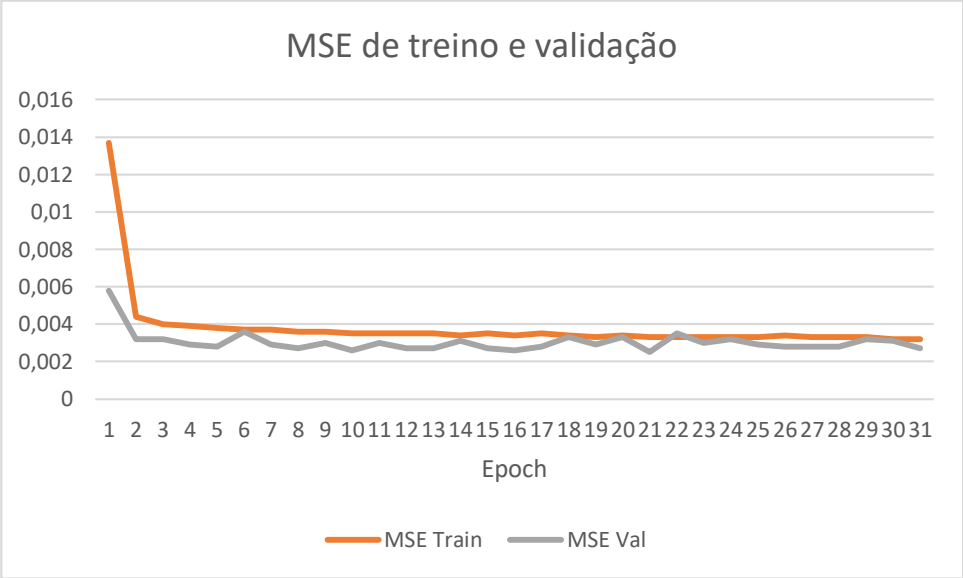

Gráfico 16 - Erro de treino e validação com aplicação da CNN no conjunto A

Verifica-se uma anormalidade, tantos nos valores, quanto na evolução do erro de validação. Em primeiro lugar, é anormal que este valor seja inferior ao erro de treino. Este comportamento pode estar relacionado com o facto do *dataset* de validação possuir dados com menor variância, como já foi descrito acima e se encontra demonstrado na **Tabela 4**. Esta característica do *dataset* pode ser mais evidente quando o *input* apresenta apenas valores de preço e variáveis cronológicas, face a outros conjuntos que apresentem variáveis exógenas. Para além disso, o erro de validação parece ser bastante variável, a partir da quinta *epoch*, justificado pela quantidade reduzida dos dados de validação.

Tabela 4 - Variância dos *datasets* de treino, validação e teste

***Dataset* treino (€/MWh)**	***Dataset* validação (€/MWh)**	***Dataset* teste (€/MWh)**
211,0898579	105,2688062	170,0731729

4.2 – Aplicação do conjunto B como variáveis de *input*

O conjunto B de variáveis foi determinado de acordo com o artigo [8], cujas séries de dados são em tudo semelhantes às disponibilizadas. Ainda que o artigo indique mais variáveis, foram consideradas, dentro desse conjunto, aquelas que foram disponibilizadas para esta dissertação.

Os resultados apresentam-se na **Tabela 5**. Verifica-se, face ao exemplo anterior, que a Regressão Linear apresenta um erro superior no conjunto de validação, ainda que o erro no teste tenha diminuído cerca de 60%.

Verifica-se uma semelhança no comportamento dos erros apresentados para a LR e DNN. Nos dois casos, o erro de validação é cerca do dobro do erro de treino, ainda que o erro de teste se aproxime deste último.

Comparativamente ao teste feito para o conjunto A, verifica-se que a LSTM tem uma diminuição de 20% do erro de validação. Esta arquitetura de rede resulta no menor erro de validação e no pior erro de teste. Ainda que pareça controverso, esta relação pode estar justificada pela razão apresentada no ponto 3.3 desta dissertação. Outra possível explicação é a natureza estocástica dos algoritmos de treino utilizados para *Deep Learning*, que podem levar a que ocorra uma melhor adaptação a um determinado conjunto de dados. Note-se que, se os conjuntos de validação e de teste tivessem uma distribuição semelhante, a LSTM seria provavelmente a técnica com melhor performance.

Tabela 5 - Resultados com a aplicação do conjunto B

Método	Estrutura da rede	Erro de treino	Erro de validação	Erro de teste	
		MSE (€/MWh)^2	MSE (€/MWh)^2	MSE (€/MWh)^2	MAE (€/MWh)
LR	-	14.47	33.04	28.93	4.14
DNN	150-1-lrelu-0.5	16.25	34.03	28.13	4.36
LSTM	50-2-lrelu-0.3	18.57	20.45	34.43	4.69
CNN	32-(6,1)-relu-0.3 32-(6,1)-relu-0.3 32-(3,1)-relu-0	20.49	27.15	26.72	4.05

É de salientar que entre junho e outubro de 2020 foi eliminada uma grande quantidade de dados devido à existência de erros nas medições ou dias incompletos.

A análise da **Tabela 5** permite identificar uma semelhança no comportamento dos erros de treino, validação e teste obtidos para a LR e a DNN. No caso da primeira, o erro de validação é 56% superior ao erro de treino e 12% superior ao de teste. No caso da DNN, estes valores são de 52 e 17%, respetivamente.

Podemos fazer uma análise comparativa no **Gráfico 17** da evolução do erro de treino e validação para os resultados obtidos na DNN, pelo facto de este modelo apresentar uma grande diferença entre estes dois erros. Conclui-se que o comportamento inicial do erro é o habitual, no entanto, esperava-se, nas últimas *epochs*, um aumento evidente do erro de validação, resultado do momento em que a rede já conhece muito bem os dados de treino, perdendo a capacidade de generalizar.

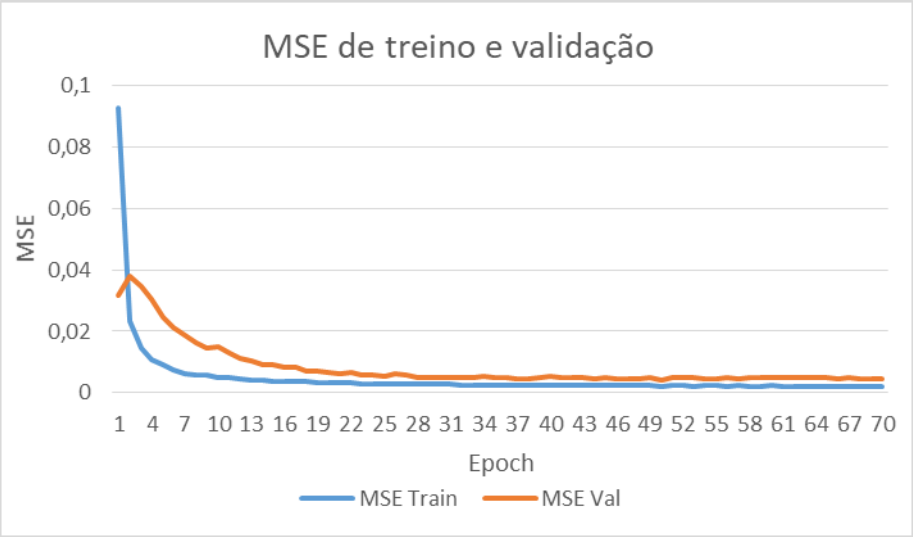

Gráfico 17 - Erro de treino e validação com aplicação da DNN no conjunto B

Para o caso da LSTM, o erro de validação é um valor próximo do erro de treino, o que indica que a rede não apresenta *overffiting*. No entanto, os resultados, quando aplicados ao teste, não mostram ser os melhores. Isto pode estar relacionado com o facto do *dataset* de teste ser composto na sua maioria por períodos afetados pela situação pandémica vivida.

Para o caso da CNN, o erro de validação é muito próximo do erro de teste, ainda que se encontrem ambos um pouco afastados do erro obtido no treino.

O **Gráfico 18** demonstra que as duas redes se comportaram de forma correta na maioria do *dataset*, no entanto, nos meses de confinamento, representados por um comportamento atípoco do preço, ambas prevêm valores mais elevados daqueles que realmente se verificaram.

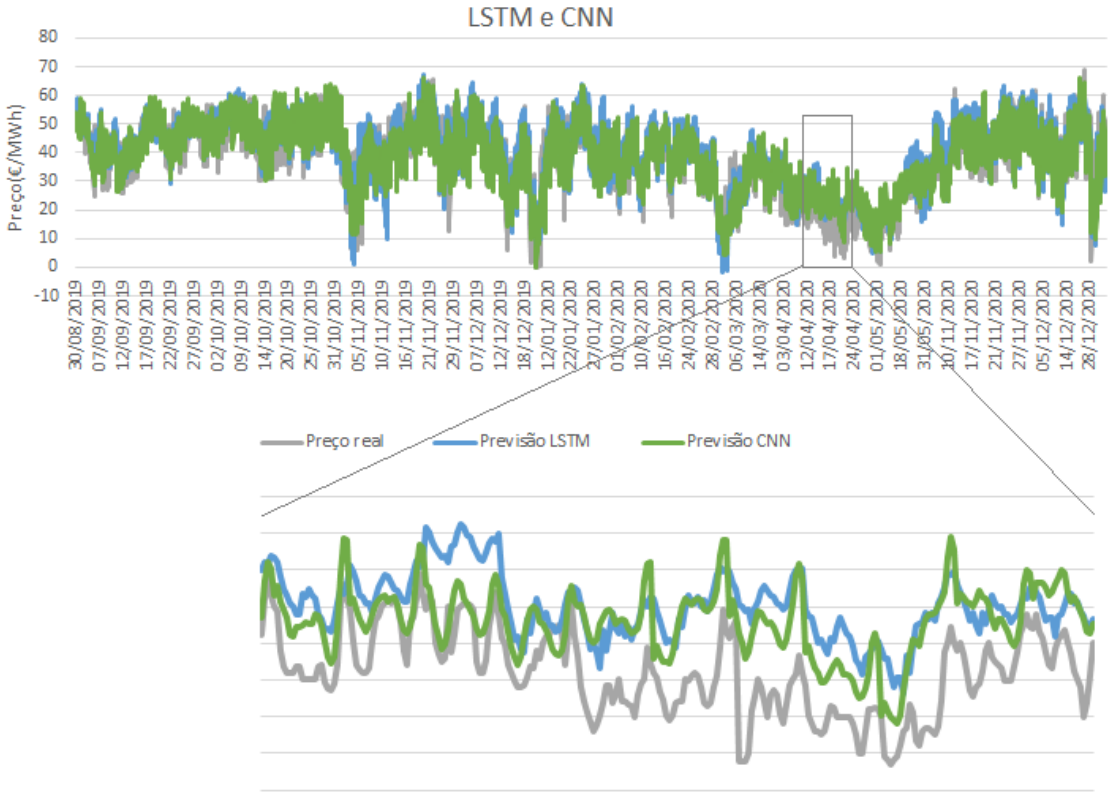

Gráfico 18 - Previsão obtida para a LSTM e CNN

4.3 – Aplicação do conjunto C como variáveis de *input*

O conjunto C é construído incluindo todas as variáveis disponíveis. Nesta fase do estudo, apenas são feitos testes com redes neuronais convolucionais. Numa primeira etapa, consideram-se as variáveis colocadas pela ordem apresentada na **Tabela 1**. São feitos estudos alterando as posições das variáveis, trocando aleatoriamente a sua posição na matriz. De seguida, é feito um teste repetindo a variável preço intercalada entre as restantes variáveis, tal como sugere o exemplo seguinte. A este conjunto de *inputs* passamos a denominar conjunto C1.

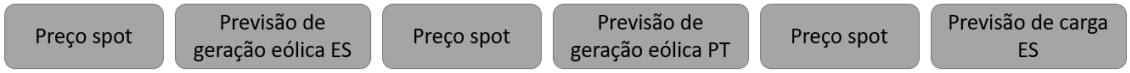

Figura 10 – Exemplo ilustrativo do preço intercalado entre variáveis

Os primeiros resultados obtidos com a aplicação do conjunto C usando para o efeito uma rede CNN com filtro verticais (do tipo *nx1*) apresentam-se na Tabela 6.

Tabela 6 - Resultados com a aplicação do conjunto C

Método	Estrutura da rede	Erro de treino	Erro de validação	Erro de teste	
		MSE (€/MWh)^2	MSE (€/MWh)^2	MSE (€/MWh)^2	MAE (€/MWh)
CNN	64-(5,1)-relu-0.3 32-(3,1)-relu-0.3 16-(3-1)-relu-0.3	18.70	21.25	24.12	3.83

A inclusão de todas as variáveis disponíveis permitiu uma diminuição do erro de teste em cerca de 10%, face ao resultado obtido com a seleção de variáveis feita no artigo [8]. No entanto, a maior vantagem encontra-se na proximidade verificada entre os erros dos três conjuntos de dados, o que indica que a rede conseguiu adquirir uma boa capacidade de treino e generalização. Pode concluir-se, portanto, que o uso de redes convolucionais parece dispensar ou, pelo menos, aliviar a fase de seleção de variáveis. A rede consegue interpretar internamente as relações entre as variáveis de entrada que melhor contribuem para uma boa previsão final. Note-se que os resultados obtidos com o conjunto B foram obtidos com as variáveis identificadas como as mais relevantes, de acordo com a análise aprofundada realizada no artigo [8]. Esta conclusão é extremamente favorável, uma vez que retira grande parte do trabalho de engenharia de dados aplicado nas previsões com recurso a métodos convencionais.

É possível comparar os resultados obtidos para a CNN quando aplicada ao conjunto B e C através da visualização do **Gráfico 19.**

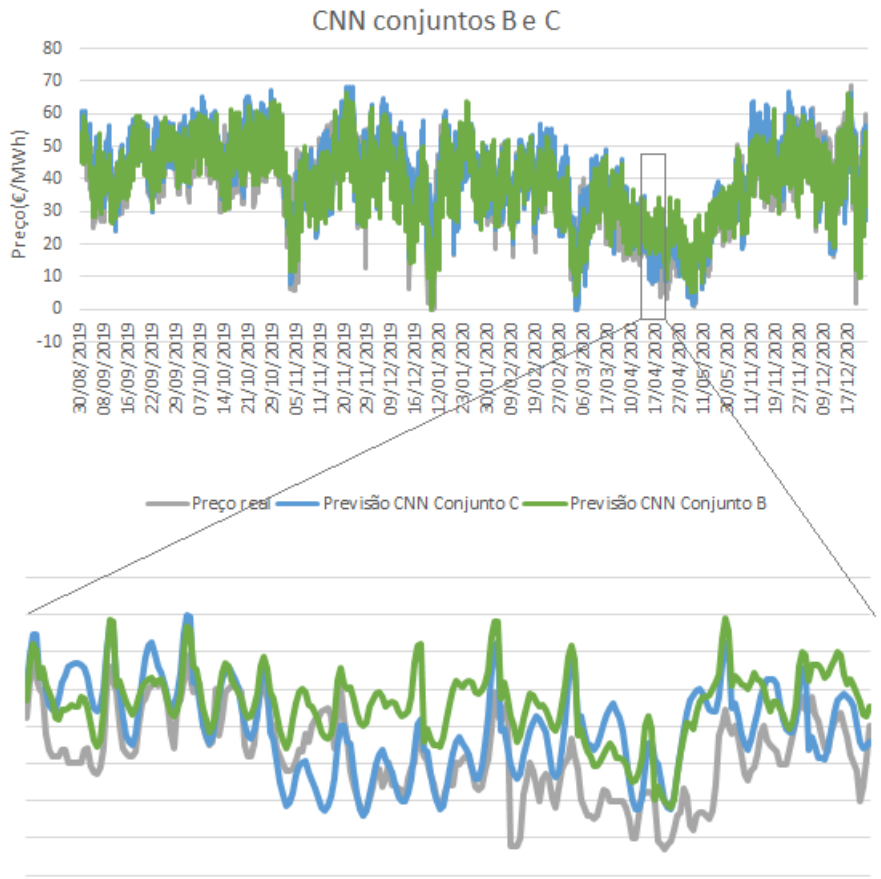

Gráfico 19 – Comparação da previsão obtida para a CNN variando o conjunto de dados de *input*

Podemos concluir que parece vantajoso assumir todas as variáveis disponíveis nas instâncias temporais. Visualmente, os efeitos convolutivos da CNN são suficientes para extrair a informação relevante, reduzindo ao mesmo tempo a possibilidade de *overfitting*, de forma a proporcionar previsões de grande qualidade.

4.3.1 - *Kernel Size* nxm

Mantendo o mesmo número de camadas e filtros da rede anterior, foram realizados testes com convoluções do tipo *nxm*, i.e., convolução cujo tamanho do canal de cada filtro (*kernel size*) é representado por uma matriz de altura n e largura m. Os resultados apresentam-se na Tabela 7. Estes filtros foram incluídos de forma a podermos verificar se a rede consegue entender relações entre as variáveis.

A utilização de filtros que estudassem a relação entre variáveis piorou o erro de validação e teste, contrariando aquilo que seria de esperar. O erro de teste diminuiu, o que indica o aumento do *overfitting*. Ainda que pudéssemos pensar que a aprendizagem da rede melhoraria com a aplicação de filtros com largura, é fácil entender que estamos a relacionar séries

temporais, que têm uma relação muito maior com o seu valor da hora h-1 do que com o valor de uma outra variável para a hora h.

Tabela 7 – Resultados com a aplicação do conjunto C; Filtros do tipo *nx2*

Método	Estrutura da rede	Erro de treino	Erro de validação	Erro de teste	
		MSE (€/MWh)^2	MSE (€/MWh)^2	MSE (€/MWh)^2	MAE (€/MWh)
CNN	32-(5,2)-relu-0.5 32-(3,2)-relu-0.5	18.01	24.21	27.55	4.19

A experiência seguinte usa o conjunto C1, exemplificado na **Figura 10**, usando também filtros do tipo *nx2*. Aqui o objetivo é verificar se a rede entende a relação entre a variável preço e cada uma das outras variáveis individualmente. Os resultados apresentam-se na **Tabela 8**.

Tabela 8 – Resultados com a aplicação do conjunto C1; Filtros do tipo *nx2*

Método	Estrutura da rede	Erro de treino	Erro de validação	Erro de teste	
		MSE (€/MWh)^2	MSE (€/MWh)^2	MSE (€/MWh)^2	MAE (€/MWh)
CNN	16-(5,2)-relu-0.6 16-(5,2)-relu-0.6 16-(3,2)-relu-0.6	19.26	22.20	26.86	4.16

Verificou-se a necessidade de aumentar uma camada, diminuir a quantidade de filtros e aumentar o *dropout* de forma a reduzir o *overfitting*. Os resultados são melhores, especialmente no que diz respeito à maior proximidade dos erros em cada uma das fases (treino, validação e teste), mostrando que a rede adquire uma maior capacidade de generalização. Ainda assim, o erro de teste diminuiu apenas 2.5% face ao exemplo anterior.

4.3.2 -Alteração da posição das variáveis na matriz de *input*

Um outro teste consistiu em alterar aleatoriamente a posição das variáveis, mantendo a rede inicial apresentada na **Tabela 6** (filtros apenas verticais), tendo-se concluído que a posição das variáveis pode influenciar, ainda que não significativamente, a aprendizagem da rede.

Os resultados obtidos para diferentes configurações aleatórias da matriz de *input* podem ser observados na **Tabela 9**.

Tabela 9 - Resultados com a aplicação do conjunto C alterando aleatoriamente a posição das variáveis na matriz de *input*

Método (configuração usada para as variáveis de *input*)	**Estrutura da rede**	**Erro de treino**	**Erro de validação**	**Erro de teste**	
		MSE (€/MWh)^2	**MSE (€/MWh)^2**	**MSE (€/MWh)^2**	**MAE (€/MWh)**
CNN (original)	64-(5,1)-relu-0.3 32-(3,1)-relu-0.3 16-(3-1)-relu-0.3	18.70	21.25	24.12	3.83
CNN (configuração 1)	64-(5,1)-relu-0.3 32-(3,1)-relu-0.3 16-(3-1)-relu-0.3	19.63	24.06	27.80	4.07
CNN (configuração 2)	64-(5,1)-relu-0.3 32-(3,1)-relu-0.3 16-(3-1)-relu-0.3	18.51	20.03	32.95	4.54
CNN (configuração 3)	64-(5,1)-relu-0.3 32-(3,1)-relu-0.3 16-(3-1)-relu-0.3	18.19	20.06	24.78	3.98

4.3.3 -Função de ativação linear na última camada da rede

Todos os testes até então foram realizados com redes CNN cuja *layer* final incluía uma função de ativação do tipo *Leaky ReLU*, tal como explicado no ponto 3.2. Podemos alterar esta função de ativação, substituindo-a por uma função linear, mantendo a estrutura da rede apresentada na **Tabela 6**. Os resultados desta experiência encontram-se na **Tabela 10**.

Tabela 10 - Resultados com a aplicação do conjunto C e função de ativação linear na última camada da rede

Método	**Estrutura da rede**	**Erro de treino**	**Erro de validação**	**Erro de teste**	
		MSE (€/MWh)^2	**MSE (€/MWh)^2**	**MSE (€/MWh)^2**	**MAE (€/MWh)**
CNN	64-(5,1)-relu-0.3 32-(3,1)-relu-0.3 16-(3-1)-relu-0.3	17.10	21.35	31.95	4.53

O último teste consiste na repetição da variável Preço intercalada entre as outras variáveis de entrada, tal como apresentado na **Figura 10.** Em ambos os casos foram considerados filtros verticais. Os resultados obtidos encontram-se na **Tabela 11**.

Tabela 11 - Resultados com a aplicação do conjunto C1 e função de ativação linear na última camada da rede

Método	**Estrutura da rede**	**Erro de treino**	**Erro de validação**	**Erro de teste**	
		MSE (€/MWh)^2	**MSE (€/MWh)^2**	**MSE (€/MWh)^2**	**MAE (€/MWh)**
CNN	32-(5,1)-relu-0.3 16-(5,1)-relu-0.3 16-(3-1)-relu-0	19.41	22.57	23.26	3.75

A repetição da variável Preço quando alteramos a função de ativação tem uma influência considerável no valor dos erros nas três fases (treino, validação e teste). Na verdade, este resultado é o melhor de todas as experiências consideradas nesta dissertação, tendo em conta o erro de teste. Comparando a **Tabela 10** e a **Tabela 11**, verifica-se que a repetição do preço resultou numa diminuição de 27% do erro de teste. Para além disso, verificou-se a maior proximidade entre os erros das três fases considerando todos os testes feitos ao longo da dissertação, resultado de uma maior capacidade de generalização por parte deste modelo.

Relativamente a este teste foi realizada uma análise ao erro de treino e validação, que se apresenta no **Gráfico 20**. É de salientar que estes valores de erro se encontram normalizados e são obtidos automaticamente em passos intermédios do próprio modelo. Verifica-se que, inicialmente, as curvas descrevem uma situação normal de treino e validação de redes deste tipo, ainda que não seja evidente a subida do erro de validação com o passar das *epochs*.

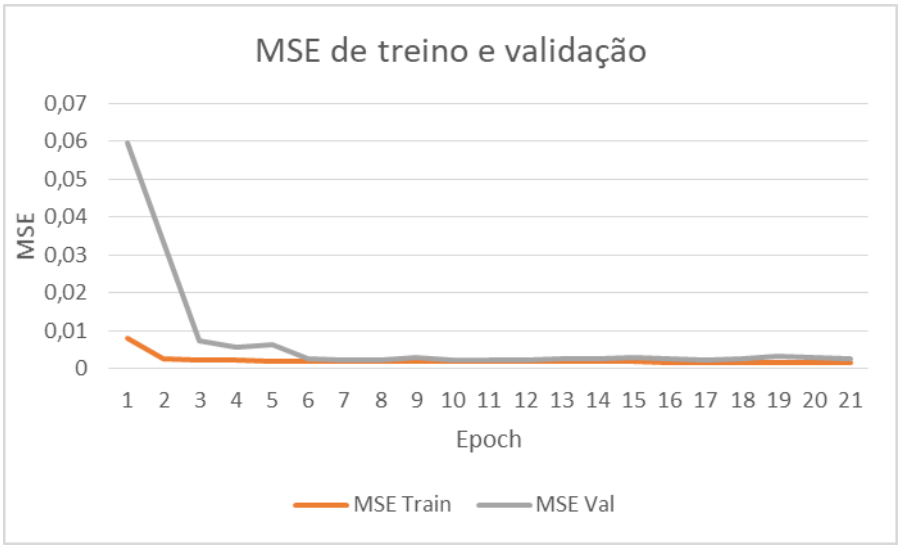

Gráfico 20 – Erro de treino e validação com aplicação da CNN no conjunto C1

A escolha da função de ativação não linear nos restantes estudos com aplicação da CNN tem como propósito o facto de a *flatten* ser composta, no caso do conjunto C1, por secções repetidas correspondentes à variável Preço *spot*. Em princípio, uma regressão linear não beneficiaria de ter a mesma variável repetida várias vezes. No caso de usarmos uma função de ativação não linear, haverá lugar a diferentes combinações não lineares das variáveis na camada *flatten*, podendo resultar numa previsão de maior qualidade.

A aplicação desta função de ativação não linear quando o *input* é definido pelo conjunto C mostrou ser favorável, analisando o erro de teste da previsão. Em trabalhos futuros, sugere-se que sejam estudadas outras possibilidades, tanto no tipo de convoluções, quanto no tipo de funções de ativação empregadas nas camadas escondidas e na camada que se segue à *flatten*.

4.4 – Comparação dos resultados obtidos

Neste ponto do capítulo 4 é feita uma comparação geral dos resultados em função do conjunto de *inputs* e do método usado para a previsão do preço. Relativamente ao ponto 4.3, apenas foi considerado para esta análise o resultado do erro apresentado na **Tabela 6.**

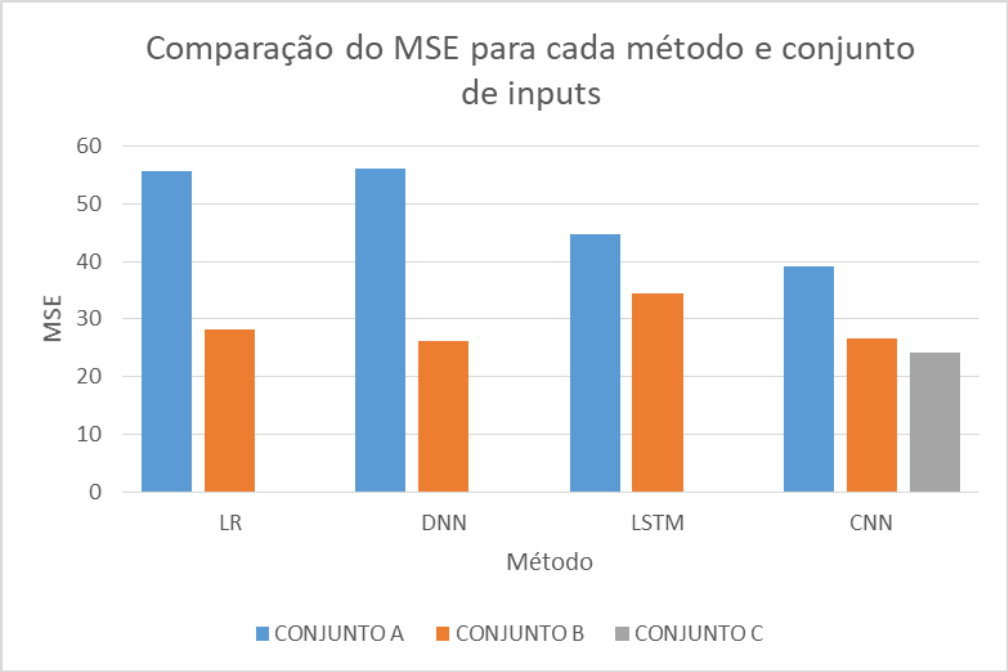

Gráfico 21 – Comparação do MSE para cada método e conjunto de *inputs*

Verifica-se que as redes neuronais convolucionais se comportam muito bem na previsão de séries temporais, ainda que sejam normalmente usadas na análise de imagens.

As DNN têm um resultado muito positivo aquando da aplicação do conjunto B como variáveis de *input*, no entanto, o erro de validação para esse método nesse caso mostrou-se elevado, como podemos observar na **Tabela** 5**.**

Contrariamente ao que se esperava, as LSTM mostraram boa capacidade de previsão nos *datasets* de treino e validação, no entanto, essa capacidade não foi refletido para o *dataset* de teste.

A regressão linear tem um comportamento semelhante à DNN, tendo erros de validação também muito elevados na aplicação do conjunto B como variáveis de *input*, mas resultando, nesse mesmo cenário, em erros de teste baixos.

Quando consideramos o conjunto A, caracterizado por apresentar como *features* apenas o preço *spot* e as variáveis cronológicas, as CNN e as LSTM conseguem retirar muito mais informação do treino comparativamente aos modelos LR e DNN.

Os resultados seguintes têm como horizonte de análise os resultados obtidos no ponto 4.3.3 desta dissertação. No **Gráfico 22** comparamos os vários erros de teste fazendo variar a função

de ativação. Aplicando no conjunto C a função de ativação linear, o erro de teste aumenta cerca de 25%. No entanto, aplicado ao conjunto C1, que intercala o preço *spot* entre as restantes variáveis, verifica-se que o MSE de teste diminui cerca de 4%. Este resultado, ainda que não se tenha encontrado uma explicação definitiva, é o melhor resultado obtido ao longo da dissertação.

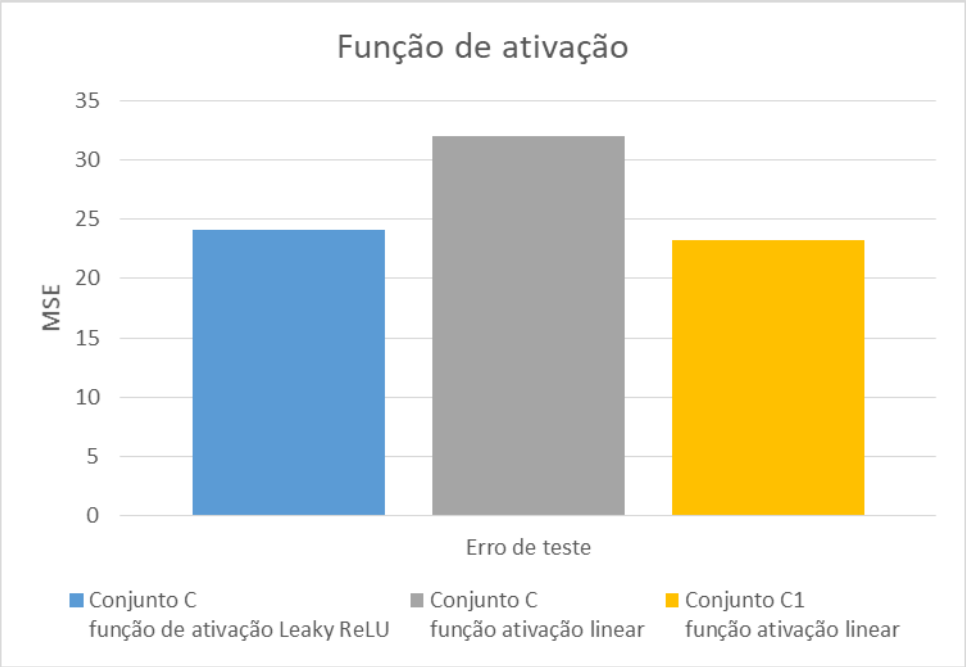

Gráfico 22 - Comparação do MSE para o conjunto C e C1 fazendo variar a função de ativação da última camada da rede

5. Conclusões

A análise dos *datasets* de treino, validação e teste permitiu concluir que existem diferenças evidentes no que diz respeito aos dados pertencentes a cada um deles. O *dataset* de validação apresenta uma variância bastante inferior à do *dataset* usado para o treino dos modelos. Para além disso, o *dataset* de teste apresenta uma média inferior aos restantes. Este último contempla essencialmente dados referentes ao ano 2020, caracterizado por uma diminuição acentuada do preço *spot* da energia elétrica, provavelmente motivada pelo período pandémico vivido, que levou ao encerramento temporário de indústrias e ao simultâneo aumento do consumo doméstico em tempos de confinamento.

Um dos objetivos desta dissertação passava por verificar se as técnicas de *Deep Learning* são vantajosas num problema de previsão de séries temporais relativamente a outras abordagens mais convencionais, em particular no que respeita à sua capacidade de proporcionar um menor erro de previsão.

Quando as entradas do modelo contemplam apenas variáveis endógenas (preço *spot* da eletricidade e variáveis cronológicas), as redes neuronais recorrentes (LSTM) e as redes neuronais convolucionais (CNN) resultam em erros menores. Em todas as redes, o erro de validação foi menor que o erro de treino, situação atípica, provavelmente motivada pela menor variância do *dataset* de validação. Ainda que o melhor resultado do teste tenha sido obtido na aplicação das redes neuronais convolucionais, a LSTM obteve o menor erro de validação.

Numa segunda fase, foram consideradas as variáveis identificadas como as mais relevantes, de acordo com a análise aprofundada realizada no artigo [8]. Quando se adicionam variáveis exógenas ao *input* dos modelos, o erro de validação supera o erro de teste em todas as arquiteturas de rede (situação típica). O modelo com melhor performance no teste continua a ser aquele que usa redes neuronais convolucionais, ainda que a LSTM tenha o mesmo comportamento na validação apresentado no parágrafo anterior, o que leva a crer que, se os conjuntos de validação e de teste tivessem uma distribuição semelhante, a LSTM seria provavelmente a técnica com melhor performance.

O segundo objetivo desta dissertação passava por perceber até que ponto a utilização de redes neuronais convolucionais permite aliviar a tarefa de seleção de variáveis de entrada. Quando são usadas todas as variáveis disponíveis com esta arquitetura de rede, o resultado do erro é menor em todos os *datasets*, comparativamente aos testes realizados anteriormente com as CNN, o que permite concluir que uso de redes neuronais convolucionais parece dispensar ou, pelo menos, aliviar a fase de seleção de variáveis.

O último teste consistiu em substituir a função de ativação da última camada da rede – inicialmente *Leaky ReLU* – por uma função de ativação linear. Os resultados pioraram face aos

anteriores para a mesma configuração dos dados de entrada. No entanto, quando repetida a variável preço entre as variáveis exógenas, a substituição da função de ativação resultou no melhor resultado no teste obtido ao longo de toda a dissertação.

Sugere-se que, em trabalhos futuros, sejam testadas outras possibilidades nas CNN, tanto no tipo de convoluções, quanto na função de ativação escolhida para as camadas escondidas e para a última camada do modelo.

Referências

[1] Ian Goodfellow, Yoshua Bengio e Aaron Courville em Deep Learning

[2] Charu C. Aggarwal em Neural Networks and Deep Learning

[3] Nikhil Buduma, Nicholas Locascio em Fundamentals Of Deep Learning

[4] João Tomé Saraiva, "Mercados de Electricidade – Uma Introdução" ", FEUP, Porto, Portugal Fevereiro de 2019

[5] João Tomé Saraiva, "Organização de Mercados de Electricidade", FEUP, Porto, Portugal Fevereiro de 2019

[6] "Previsão de preços de eletricidade para o mercado MIBEL". Disponível em https://repositorio-aberto.up.pt/bitstream/10216/75709/2/31953.pdf

[7] Miguel Prado, "Mercado ibérico de eletricidade passará a ter preços negativos em julho". Disponível em https://expresso.pt/economia/2021-05-20-Mercado-iberico-de-eletricidade-passara-a-ter-precos-negativos-em-julho-fb46b83f

[8] "Probabilistic Price Forecasting for Day-Ahead and Intraday Markets: Beyond the Statistical Model". Disponível em https://www.mdpi.com/2071-1050/9/11/1990/htm

[9] François Chollet em Deep Learning with Python

[10]"Intro to Optimization in Deep Learning: Vanishing Gradients and Choosing the Right Activation Function". Disponível em https://blog.paperspace.com/vanishing-gradients-activation-function/

[11]"Deep Dive into Math Behind Deep Networks". Disponível em https://towardsdatascience.com/https-medium-com-piotr-skalski92-deep-dive-into-deep-networks-math-17660bc376ba

[12]"Deep Learning". Disponível em https://en.wikipedia.org/wiki/Deep_learning

[13]"Arquitetura de Redes Neurais Long Short Term Memory (LSTM)". Disponível em https://www.deeplearningbook.com.br/arquitetura-de-redes-neurais-long-short-term-memory/

[14] "20 Questions to Test your Skills on CNN (Convolutional Neural Networks)". Disponível em https://www.analyticsvidhya.com/blog/2021/05/20-questions-to-test-your-skills-on-cnn-convolutional-neural-networks/

[15]"Understanding LSTM Networks". Disponível em https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[16]“CONSUMO DE ENERGIA ELÉTRICA: ABRIL 2020”. Disponível em https://www.observatoriodaenergia.pt/pt/comunicar-energia/post/8254/consumo-de-energia-eletrica-abril-2020/
[17]“CONSUMO DE ENERGIA ELÉTRICA: MAIO 2020”. Disponível em https://www.observatoriodaenergia.pt/pt/comunicar-energia/post/8356/consumo-de-energia-eletrica-maio-2020/
[18]“CONSUMO DE ENERGIA ELÉTRICA: JUNHO 2020”. Disponível em https://www.observatoriodaenergia.pt/pt/comunicar-energia/post/8414/consumo-de-energia-eletrica-junho-2020/