



# Redes Neurais Artificiais na Previsão de Consumos de Água

**JAVIER PRIETO CANO**

julho de 2017



# **REDES NEURONAIS ARTIFICIAIS NA PREVISÃO DE CONSUMOS DE ÁGUA**

JAVIER PRIETO CANO

Dissertação submetida para satisfação parcial dos requisitos do grau de

**MESTRE EM ENGENHARIA CIVIL – RAMO INFRAESTRUTURAS**

Orientador: Eduardo Bruno De Freitas Vivas

Co-Orientador: Pedro Miguel Leite Ferreira

**JUNHO DE 2017**



# ÍNDICE GERAL

Índice Geral .....	iii
Resumo.....	v
Abstract .....	vii
Índice de Texto .....	ix
Índice de Figuras.....	xiii
Índice de Tabelas.....	xvii
Abreviaturas .....	xix
CAPÍTULO 1    Introdução.....	1
CAPÍTULO 2    Revisão Bibliográfica .....	5
CAPÍTULO 3    Ferramentas utilizadas.....	47
CAPÍTULO 4    Caracterização e análise do caso de estudo .....	51
CAPÍTULO 5    Metodologia aplicada .....	65
CAPÍTULO 6    análise de resultados .....	99
CAPÍTULO 7    Considerações finais .....	107
Referências Bibliográficas .....	111
Anexo I – análise estatística dos dados .....	115
Anexo II – resultados dos testes efetuados para a escolha do modelo de RNA .....	124
Anexo III – Forma da FAC e da FAP para a identificação de modelos ARIMA .....	127



## RESUMO

Esta dissertação apresenta o desenvolvimento de uma ferramenta de previsão de consumos de água mediante a utilização de inteligência artificial, nomeadamente de modelos de redes neuronais artificiais.

Uma adequada previsão de consumos de água a curto, médio e longo prazo possibilita às empresas de abastecimento e distribuição de água uma informação imprescindível para estimar a capacidade de planeamento, atividades de manutenção, melhorias do sistema e otimização da operação de sistemas elevatórios e de tratamento.

São examinados, para além das redes neuronais artificiais, outros modelos estatístico-matemáticos aplicados à previsão dos consumos, como o modelo ARIMA, baseado em médias móveis integradas autorregressivas.

Em virtude do escasso número de trabalhos nesta área, considerou-se necessário aprofundar os conteúdos teóricos em matéria de redes neuronais e séries temporais, para depois aplicar uma metodologia própria que permitiu implementar o modelo na linguagem *Python*, sendo também utilizados outros *softwares* para explorar a capacidade das redes neuronais artificiais, como o *Neural Network Toolbox* para Matlab e o modelo ARIMA.

Como caso de estudo, a metodologia foi aplicada aos registos de consumos do concelho de Arouca, cujo abastecimento em alta corresponde à Águas do Douro e Paiva (AdDP), do grupo Águas de Portugal. Como dados de base foram considerados os valores de consumo a cada meia hora, bem como dados de precipitação e temperatura.

Os resultados da aplicação de redes neuronais foram satisfatórios, uma vez que o modelo aplicado sobre a série diária produziu previsões com uma precisão próxima de 96%, enquanto o modelo aplicado à série horária forneceu precisões próximas de 87 %. Este rigor nas previsões demonstra a efetividade da aplicação das redes neuronais na previsão dos consumos de água em vários horizontes de tempo.

**Palavras-chave:** abastecimento, consumos de água, inteligência artificial, redes previsão, otimização.





## ABSTRACT

This thesis presents the development of a tool for water consumption forecasting through the use of artificial intelligence, especially of artificial neuron models.

An adequate prediction of water consumption in the short, medium and long term provides essential information for water supply and distribution companies planning capacity, maintenance activities, system improvements and optimization of pumping treatment operation.

In addition to the artificial neural networks, other statistical and mathematical models are applied to the forecast of the time series of consumption, such as the ARIMA model, based on autoregressive integrated moving averages.

Due to the small number of available works in this area, it was necessary to deepen the theoretical contents regarding neural networks and time series, for the later development of a specific methodology and the respective implementation using Python language. Moreover, the capacity of artificial neural networks and of the ARIMA model were also explored using Neural Network Toolbox for Matlab.

As case study, the methodology was applied to the consumption records for the municipality of Arouca, whose water supply is delivered by the Águas do Douro and Paiva (AdDP) company, from Águas de Portugal group. The main data used was water consumption, precipitation and temperature records available for the area.

The results of the application of neural networks were satisfactory, since the model applied over a series of daily records produced predictions with an accuracy close to 96%, while the model applied to the hourly series provided accuracies close to 87%. This rigor in predictions demonstrates an effectiveness of the application of neuronal networks in the face of the problem for water consumption forecasting in various periods of time.

**Keywords:** supply, water consumption, artificial intelligence, ANN forecasting, optimization.



## ÍNDICE DE TEXTO

1.1	Enquadramento e objetivos .....	1
1.2	Estrutura do Relatório .....	4
2.1	Introdução.....	5
2.2	Generalidades .....	8
2.3	Razões para a utilização das Redes Neurais Artificiais .....	11
2.4	Paralelismo das Redes Neurais com o Sistema Nervoso .....	11
2.5	Topologia, arquitetura e tipos de Redes Neurais Artificiais .....	13
2.5.1	Função de entrada.....	15
2.5.2	Função de ativação .....	16
2.5.3	Função de saída .....	19
2.6	Modelos de Redes Neurais Artificiais .....	19
2.6.1	Perceptrão .....	20
2.6.2	Perceptrão multicamada PMC.....	21
2.6.3	Rede NARx .....	23
2.6.4	Rede LSTM .....	24
2.7	Treino das RNA.....	28
2.7.1	Introdução .....	28
2.7.2	A retropropagação.....	28
2.7.3	Generalização .....	31
2.7.4	Validação cruzada.....	33
2.8	Análise das séries temporais.....	35
2.8.1	Classificação das séries temporais.....	36

2.8.2	Análise univariante das séries temporais .....	38
2.8.3	Processos autorregressivos AR(p).....	39
2.8.4	Processos de média móvel MA(q) .....	40
2.8.5	Processos autorregressivos de média móvel ARMA (p,q) .....	41
2.8.6	Processos integrados I(d).....	41
2.8.7	Processos autorregressivos integrados de média móvel ARIMA (p,d,q).....	42
2.8.8	Processos ARIMA sazonais.....	42
2.8.9	Modelo ARIMA sazonal.....	44
2.8.10	Ferramentas para a criação do modelo ARIMA.....	45
3.1	<i>PyCharm</i> .....	47
3.2	<i>Neural Network Toolbox</i> .....	48
3.3	<i>Outros softwares</i> .....	49
3.4	<i>Hardware utilizado</i> .....	49
4.1	Caso de estudo .....	51
4.1.1	Dados disponibilizados .....	54
4.2	Seleção e tratamento de informação .....	55
4.2.1	Escolha inicial das variáveis .....	56
4.2.2	Análise multivariável das variáveis escolhidas .....	58
5.1	Introdução.....	65
5.1.1	Considerações prévias .....	65
5.1.2	Preparação dos dados.....	67
5.2	Procedimento de modelação da RNA.....	73
5.2.1	Escolha do modelo.....	73
5.2.2	Definição da arquitetura da rede .....	76
5.2.3	Validação do modelo .....	83
5.3	Procedimento de modelação do modelo ARIMA .....	88
5.4	Considerações finais.....	98

7.1	Conclusões .....	107
7.2	Desenvolvimentos Futuros .....	109



## ÍNDICE DE FIGURAS

Figura 2.1 – Aproximação de funções.....	10
Figura 2.2 – Classificação. ....	10
Figura 2.3 – Agrupamento ou <i>clustering</i> não supervisionado: encontrar grupos desconhecidos nos dados.....	10
Figura 2.4 – Previsão ou <i>forecasting</i> : prever os próximos resultados de uma série temporal. ....	10
Figura 2.5 – Diagrama de um neurónio, adaptado de Srinivasa e Brion (2005). ....	12
Figura 2.6 – Modelo biológico e a sua representação como Rede Neuronal Artificial, adaptado de Srinivasa e Brion (2005). ....	13
Figura 2.7 – Funções aplicadas num neurónio.....	15
Figura 2.8 – Função de ativação linear.....	17
Figura 2.9 – Função de ativação sigmoidal ....	18
Figura 2.10 – Função de ativação tangente hiperbólica ....	18
Figura 2.11 – Função de ativação arco-tangente.....	18
Figura 2.12 – Representação esquemática do perceptrão ....	20
Figura 2.13 – Representação esquemática de um PMC com uma camada oculta.....	22
Figura 2.14 – Representação esquemática de uma rede NARx. ....	24
Figura 2.15 – Representação esquemática de um bloco de memória com duas células, adaptado de Pérez (2002). ....	25
Figura 2.16 – Representação esquemática do funcionamento numa rede LSTM, adaptado de (Pérez, 2002). ....	27
Figura 2.17 – Utilização de diferentes funções de ajuste para uns pontos dispersos. ....	32
Figura 2.18 – Curvas de treino e teste nas que se produz <i>overtraining</i> . Adaptada de .Theodoris e Koutroumbas (2003). ....	33

Figura 2.19 – Variação da temperatura do ar numa determinada localização.....	36
Figura 2.20 – Variação dos consumos diários de água <i>versus</i> temperatura do ar.....	37
Figura 4.1 – Localização do concelho de Arouca e a sua divisão em freguesias.....	51
Figura 4.2 – Gráfico de barras representativo da superfície das diferentes freguesias que formam parte do concelho de Arouca.....	52
Figura 4.3 – Evolução da população no concelho de Arouca.....	52
Figura 4.4 – Evolução do tarifário de água em Arouca .....	53
Figura 4.5 – Mapa da rede de abastecimento da Águas do Douro e Paiva , adaptado de Águas de Douro e Paiva S.A., (2017).....	54
Figura 4.6 – Mapa localização da estação meteorológica de Castelo de Burgães, adaptado do <i>SNIRH : Sistema Nacional de Informação de Recursos Hídricos (2017)</i> . ....	55
Figura 4.7 – Fatores que podem afetar ao consumo de água.....	56
Figura 4.8 – Análise multivariável em relação aos consumos diários. ....	58
Figura 4.9 – Análise multivariável em relação aos consumos horários. ....	60
Figura 4.10 – Série temporal diária do consumo de água do período 2012-2014.....	61
Figura 4.11 – Série temporal horária do consumo de água do período 2015-2016.....	62
Figura 4.12 – Temperatura diária média do período 2007-2017.....	64
Figura 5.1 – Fluxograma da metodologia proposta. ....	67
Figura 5.2 – Identificação de outliers na série diária com média móvel de período 30 dias.....	69
Figura 5.3 – Série diária corrigida com média móvel de período 30 dias. ....	69
Figura 5.4 – Identificação de outliers na série horária com média móvel de período 24 horas.....	70
Figura 5.5 – Série horária corrigida com média móvel de período 24 horas. ....	70
Figura 5.6 – Série diária diferenciada após correção. ....	71
Figura 5.7 – Série horária diferenciada após correção.....	71
Figura 5.8 – Caso multivariante (2 variáveis) transformado a aprendizagem supervisionada. ....	72
Figura 5.9 – FAP dos primeiros 50 <i>lags</i> dos consumos diários.....	75
Figura 5.10 – FAP dos primeiros 50 <i>lags</i> dos consumos horários. ....	76



Figura 5.11 – EQM médio após validação cruzada para PMC não recorrente de uma e duas camadas ocultas, caso diário.....	79
Figura 5.12 – EQM médio após validação cruzada para rede NARx de uma camada, caso diário. ....	80
Figura 5.13 – EQM médio após validação cruzada para rede LSTM, com correções baseadas em média móvel de 7 e 30 períodos (1 e 2 camadas), caso diário. ....	80
Figura 5.14 – EQM médio após validação cruzada para PMC não recorrente de uma camada e duas camadas, caso horário. ....	82
Figura 5.15 – EQM médio após validação cruzada para rede NARx de uma camada oculta, caso horário. ....	82
Figura 5.16 – EQM médio após validação cruzada para rede LSTM de uma e duas camadas ocultas, caso horário.....	82
Figura 5.17 – EQM médio de validação e treino (padronizados) para a rede LSTM de uma camada oculta com 40 blocos de memória. Série diária.....	84
Figura 5.18 – Diagrama de correlação entre targets e saídas da rede LSTM com uma camada oculta e 40 blocos de memória, caso diário. ....	85
Figura 5.19 – Histograma e função de densidade dos resíduos, caso diário. ....	85
Figura 5.20 – EQM médio de validação e treino (padronizados) para a rede LSTM de 30 blocos de memória para um número de épocas igual a 10 (esquerda) e 20 (direita). Série horária. ....	87
Figura 5.21 – Diagrama de correlação entre targets e saídas da rede LSTM com uma camada oculta e 30 blocos de memória, caso horário.....	87
Figura 5.22 – Histograma e função de densidade dos resíduos, caso horário. ....	88
Figura 5.23 – Fluxograma do procedimento para a determinação de modelos ARIMA.....	89
Figura 5.24 – FAC e FAP da série diária após diferenciação regular. ....	90
Figura 5.25 – FAC e FAP da série diária após diferenciação sazonal.....	90
Figura 5.26 – FAC e FAP da série diária após diferenciação sazonal sobre diferenciação regular. ....	91
Figura 5.27 – FAC dos resíduos diários após aproximação mediante modelo ARIMA. ....	92
Figura 5.28 – FAC e FAP da série horária após diferenciação regular.....	95
Figura 5.29 – FAC e FAP da série horária após diferenciação sazonal. ....	96
Figura 5.30 – FAC e FAP da série horária após diferenciação sazonal sobre diferenciação regular.....	96

Figura 5.31 – FAC dos resíduos horários após aproximação mediante modelo ARIMA. ....	97
Figura 6.1 – Consumos diários médios anuais previstos face aos consumos de referência.....	100
Figura 6.2 – Média dos consumos diários médios mensais previstos face aos consumos de referência.	101
Figura 6.3 – Diagrama circular representativo do consumo diário médio semanal da série de referência. .....	102
Figura 6.4 – Diagrama circular representativo do consumo diário médio semanal previsto. ....	103
Figura 6.5 – Série temporal diária de referência e previsão aplicada ao conjunto de teste.....	104
Figura 6.6 – <i>Zoom</i> da série temporal dos consumos diários de referência e previsão aplicada ao conjunto de teste.....	104
Figura 6.7 – Série temporal horária de referência e previsão aplicada ao conjunto de teste. ....	105
Figura 6.8 – <i>Zoom</i> da série temporal horária de referência e previsão aplicada ao conjunto de teste. .	105
Figura 6.9 – Previsão da média horária dos consumos para cada dia da semana em comparação com a série de referência. ....	106

## ÍNDICE DE TABELAS

Tabela 1.1 – Cronograma de trabalhos. ....	4
Tabela 2.1 – Características das principais funções de ativação.....	19
Tabela 2.2 – Resumo das características principais das FAC e das FAP para os modelos AR(p), MA(q) e ARMA (p,q).....	46
Tabela 4.1 – Simulação de consumos para 120 e 180 m <sup>3</sup> para o município de Arouca.....	53
Tabela 4.2 – Valores do coeficiente $\rho$ entre as distintas variáveis diárias.....	59
Tabela 4.3 – Valores do coeficiente $\rho$ entre as distintas variáveis horárias. ....	60
Tabela 5.1 – Arquitetura das camadas de entrada para cada modelo de RNA. ....	78
Tabela 5.2 – Características úteis para a validação do modelo LSTM da série diária. ....	84
Tabela 5.3 – Resumo estatístico dos resíduos do modelo LSTM da série diária.....	86
Tabela 5.4 – Características úteis para a validação do modelo LSTM da série horária. ....	86
Tabela 5.5 – Resumo estatístico dos resíduos do modelo LSTM da série diária.....	88
Tabela 5.6 – Parâmetros ARIMA da série diária.....	92
Tabela 5.7 – Teste de RB de Box-Pierce da série diária.....	93
Tabela 5.8 – Primeiros valores da aproximação mediante método ARIMA, série diária.....	94
Tabela 5.9 – EQM dos modelos ARIMA da série diária. ....	95
Tabela 5.10 – Parâmetros ARIMA da série horária. ....	97
Tabela 5.11 – Teste de Box-Pierce da série horária. ....	98
Tabela 5.12 – EQM dos modelos ARIMA da série horária. ....	98
Tabela 6.1 – Consumos diários médios anuais previstos face aos consumos de referência. ....	99
Tabela 6.2 – Consumos mensais previstos face aos consumos de referência. ....	101
Tabela 6.3 – Consumo diário médio semanal previsto face ao consumo de referência. Caso diário.....	102



## ABREVIATURAS

AR	Autorregressivo.
EQM	Erro Quadrático Médio.
FAC	Função de Autocorrelação completa.
FAP	Função de Autocorrelação Parcial.
LSTM	Long Short-Term Memory.
NARx	Nonlinear AutoRegressive network with exogenous inputs.
PE	Ponto de Entrega.
PMC	Perceptrão Multi-Camada.
REQM	Raiz do Erro Quadrático Medio.
RNA	Rede Neuronal Artificial.
SNIRH	Sistema Nacional de Informação de Recursos Hídricos



# CAPÍTULO 1

## INTRODUÇÃO

### 1.1 ENQUADRAMENTO E OBJETIVOS

Num mundo onde a população e as áreas urbanas estão em constante expansão, um dos primeiros efeitos corresponde ao aumento das necessidades no que diz respeito ao consumo de água. Nas últimas décadas, as sociedades tornaram-se cada vez mais conscientes da importância da boa utilização da água. O uso eficiente da água tem benefícios, não só no que diz respeito à economia, mas também a nível ambiental e de sustentabilidade.

Tendo em conta a gestão e operação de um sistema de abastecimento de água e um uso eficiente da água disponível, é necessária a capacidade de prever as necessidades de consumos, bem como dos volumes disponíveis. A modelação e a previsão de consumos de água urbana são particularmente relevantes em cidades onde os recursos hídricos são limitados, mas constituem, igualmente, para os mais diversos sistemas, uma importante ferramenta para melhor gestão dos custos de operação e aumento da eficiência dos sistemas.

Um modelo preditivo com base na análise da série de consumos pode representar uma poderosa ferramenta para os gestores dos sistemas de abastecimento de água, resultando em benefícios, tais como uma operação mais eficiente que produz uma redução de fugas e uma poupança de energia, entre outros (Alejandro, Rodríguez e Marco, 2007).

Um método de otimização pode ser definido como um processo matemático que procura maximizar ou minimizar o valor de uma função e tem em conta as diferentes variáveis que influenciam o objetivo da função que se deseja otimizar.

Atualmente, estudam-se várias ferramentas para previsão dos consumos de água, mediante o uso de diferentes modelos matemáticos. Alguns destes métodos são (Ariza Ramírez, 2013):

- Modelos de tendência.
  - Modelos de regressão simples: são funções matemáticas que tentam modelar probabilisticamente uma variável de estudo em relação a um ou mais preditores de interesse.

No método de regressão simples, há apenas uma variável independente e dependente, onde a variável independente é útil para prever a variável dependente. Uma desvantagem é a não consideração de variáveis exógenas (por exemplo, a temperatura do ar). Alguns dos modelos de regressão simples mais conhecidos são:

- Método dos mínimos quadrados
  - Regressão Linear.
  - Regressão exponencial.
  - Regressão polinomial.
  - Regressão potencial.
  - Regressão logarítmica.
  - Regressão Logística.
- Regressão linear múltipla: para além da consideração dos valores observados, a regressão linear múltipla também considera os fatores que influenciam a variação da variável de estudo principal (consumos de água), isto é, a temperatura, o crescimento da população, etc.
  - Modelos baseados no estudo das séries temporais: este método requer pouca informação, pois a única variável independente é o tempo. Destacam os seguintes:
    - Processo auto-regressivo (AR).
    - Média móvel (MA).
    - Auto-regressivo e média móvel (ARMA).
    - Modelo ARIMA: a previsão de valores futuros das séries temporais estudo mediante a utilização do modelo ARIMA representa o processo estocástico com base apenas em valores passados da própria série, de modo que é possível fazer previsões de curto e médio prazo.
  - Redes neuronais artificiais (RNA): as redes neuronais são de grande utilidade para aproximar funções não-lineares, o que permite modelar fenómenos complexos onde as variáveis não estão correlacionadas.

Entre a variedade de metodologias que existem para a reprodução de padrões de consumo, destacam-se as Redes Neuronais Artificiais pela sua capacidade de adaptação e de considerar as diversas variáveis envolvidas. Considera-se que este é um tema de investigação promissor, ao proporcionar uma boa base para o desenvolvimento de um modelo económico-financeiro de forma que, aplicado conjuntamente com



os modelos de otimização, como o caso dos Algoritmos Genéticos, seja uma poderosa ferramenta para a otimização da operação de sistemas elevatórios integrados em sistemas de abastecimento.

Assim, das ferramentas matemáticas de inteligência artificial disponíveis, o trabalho baseia-se principalmente na utilização das RNA. No entanto, também será feita uma análise do modelo ARIMA, um dos mais utilizados no estudo dos consumos de água.

Uma das questões mais importantes quando se analisam os sistemas de abastecimento de água, corresponde aos custos de operação, em especial no que diz respeito aos volumes tratados na Estação de Tratamento de Águas e ao recurso à elevação para disponibilização dos volumes de água necessários à pressão adequada aos clientes finais.

Por tanto, o presente relatório tem como objetivo principal o desenvolvimento de um módulo de avaliação e reprodução de padrões de consumo com recurso às redes neuronais com vista a uma futura implementação numa ferramenta de otimização especialmente concebida para a gestão e operação de sistemas elevatórios integrados em sistemas de abastecimento.

Essa ferramenta, cujo desenvolvimento já foi iniciado no âmbito de outros trabalhos académicos (DIAS GONÇALVES, 2016), toma como base a utilização de Algoritmos Genéticos como método de otimização, o software EPANET como modelo de simulação hidráulica. A aplicação de Redes Neuronais para a avaliação e reprodução de padrões de consumos, poderá complementar as potencialidades da ferramenta em causa, tendo em vista a possível previsão de curto prazo das necessidades de elevação.

No processo de trabalho existem vários objetivos diferenciados, nomeadamente no que toca a:

- i. Avaliação das metodologias existentes para avaliação e reprodução de padrões de consumo, com especial destaque para o estudo das redes neuronais artificiais.
- ii. Desenvolvimento do módulo de criação do modelo de redes neuronais, com base na linguagem python, para futura aplicação da ferramenta de otimização em desenvolvimento.
- iii. Avaliação de um caso de estudo real para teste e validação do módulo criado.

Para a realização da presente dissertação, tornou-se necessário gerir o tempo de realização de cada uma das tarefas. Para tal foi seguido o seguinte cronograma de tarefas.

O plano de trabalho seguido é apresentado em forma de lista, em correspondência com a Tabela 1.1.

1. Revisão bibliográfica.
2. Avaliação de modelos de redes neuronais aplicáveis à reprodução de padrões de consumo de água em sistemas de abastecimento.

3. Avaliação de outros métodos de inteligência artificial aplicáveis e comparação com as redes neuronais artificiais.
4. Avaliação de dados de consumo de um caso de estudo real.
5. Desenvolvimento de um módulo de redes neuronais para incorporação na ferramenta de otimização.
6. Escrita do relatório final.

Tabela 1.1 – Cronograma de trabalhos.

Tarefas	Fevereiro	Marco	Abril	Maiο	Junho
1					
2					
3					
4					
5					

## 1.2 ESTRUTURA DO RELATÓRIO

O presente relatório é formado pelos seguintes capítulos:

- **Capítulo 1-Introdução.** Neste capítulo apresentam-se os objetivos do relatório. Também é definido o enquadramento da situação atual dos sistemas de otimização e especificamente das RNA, assim como o plano de trabalhos seguido durante a sua realização.
- **Capítulo 2-Revisão bibliográfica.** São apresentados os resultados da revisão bibliográfica realizada. O conteúdo de este capítulo é de carácter teórico e imprescindível para compreender a metodologia.
- **Capítulo 3- Ferramentas utilizadas.** Descrição breve das principais ferramentas computacionais utilizadas na elaboração do estudo.
- **Capítulo 4- Caracterização e análise do caso de estudo.**
- **Capítulo 5- Metodologia Aplicada.** Descreve-se a metodologia aplicada para a obtenção dos padrões de consumos no diferente *software* utilizado, a obtenção e tratamento dos dados, etc.
- **Capítulo 6-Análise dos resultados.** Análise dos resultados obtidos e a sua comparação com os resultados reais.
- **Capítulo 7- Conclusões.** São apresentadas as conclusões retiradas do estudo elaborado e algumas possíveis sugestões.

## CAPÍTULO 2

### REVISÃO BIBLIOGRÁFICA

#### 2.1 INTRODUÇÃO

Qualquer revisão superficial da literatura científica confirmará que um esforço considerável tem sido gasto na previsão de consumos de água desde há várias décadas. No caso do controlo operacional, o interesse é limitado a prazos curtos e médios (horários, diários e mensais). Por outro lado, o consumo anual apresenta um maior interesse para o planeamento dos recursos hídricos de longo prazo. (Alvisi, Franchini e Marinelli, 2007).

Um exemplo de previsão de consumos a curto prazo é o realizado pela Universidade de Ferrara no município de Castelfranco, situado em Emilia Romagna, Itália, entre Modena e Bolonha. (Alvisi *et al.*, 2007). A população apresenta grandes flutuações dos consumos de água. O modelo de previsão é baseado nos padrões implícitos na série temporal de consumo de água e, por esse motivo, é referido como o modelo de Previsão de consumo de Água Baseada em Padrões (Patt\_WDF). Considera dois módulos: o primeiro, cujo objetivo é estimar o consumo diário médio de água no dia (ou dias) coberto por um período de previsão de 24 h, os padrões sazonais e semanais são levados em consideração, enquanto no segundo módulo, esses montantes diários são combinados com os padrões de consumos diários e a persistência a curto prazo para fornecer previsões horárias num período de 24 h.

Basicamente, o modelo de previsão de consumos diários para o dia juliano  $m$  é definido como uma soma de consumos parciais:

$$Q_m^{d,for} = Q_m^{d,s} + \Delta_{i,j}^{d,w} + \delta_m^d \quad (2.1)$$

onde:

$Q_m^{d,s}$  — é a média diária a longo prazo representando a componente sazonal periódica, modelada mediante séries de Fourier;

$\delta_m^d$  — é o desvio, representando a componente de persistência a medio prazo, definida como o desvio padrão entre o valor médio estimado a partir das componentes  $Q_m^{d,s}$  e  $\Delta_{i,j}^{d,w}$ , modelada usando um processo autoregressivo  $AR(1)$ .

$\Delta_{i,j}^{d,w}$  — é uma correção representando a componente periódica semanal. Matematicamente, é definida como:

$$\Delta_{i,j}^{d,w} = Q_{i,j}^d - Q_j^w \quad (2.2)$$

Onde:

$Q_{i,j}^d$  — é o valor médio diário do consumo observado no dia  $i$  da semana ( $i = 1, 2, \dots, 7, 2^a - \text{feira} \dots \text{domingo}$ ) e estação do ano ( $j = 1, \dots, 4, \text{inverno}, \text{primavera}, \text{verão}, \text{outono}$ ).

$Q_j^w$  — é a média do consumo semanal de água na estação  $j$ .

O módulo de previsão do consumo horário é definido de maneira semelhante, mas com uma componente de persistência:

$$Q_{t+k}^{h,for} = Q_m^{d,for} + \Delta_{n,i,j}^h + \varepsilon_{t+k} \quad (2.3)$$

Como pode observar-se, a previsão horária é função da previsão diária. Portanto, quanto melhor for a previsão diária, mais acertada será a previsão horária, e vice-versa. As outras variáveis são:

$\Delta_{n,i,j}^h$  — é o desvio horário, definido como:

$$\Delta_{n,i,j}^h = Q_{n,i,j}^h - Q_{i,j}^d \quad (2.4)$$

Onde:

$Q_{n,i,j}^h$  — representa o valor da média horária do consumo de água observado na hora  $n$  ( $n = 1, 2, \dots, 24$ ) do dia  $i$ , na estação do ano  $j$ .

$\varepsilon_{t+k}$  — representa a persistência horária, modelada usando regressão dos erros  $\varepsilon_{t+k-1}$  e  $\varepsilon_{t+k-24}$ .

É definido mediante a expressão:

$$\varepsilon_{t+k} = \varphi_1 \varepsilon_{t+k-1} + \varphi_{24} \varepsilon_{t+k-24} \quad (2.5)$$

Os coeficientes  $\varphi_1$  e  $\varphi_{24}$  dependem da hora do dia  $t + k \equiv n = 1, 2, \dots, 24$ .

As conclusões que foram extraídas da aplicação do modelo foram determinantes: o modelo forneceu bons resultados para um determinado dia, mas surgiram dificuldades com a transição de um dia a seguir, especialmente entre dias úteis e fins de semana. Ao adotar uma abordagem baseada em padrões, que contém ciclos sazonais, semanais e diários, essas dificuldades poderiam ser evitadas. Contudo, a dependência dos padrões cíclicos também tem suas próprias falhas, pois entre outras coisas, nenhuma atenção é dada às variáveis exógenas relativas ao clima, que pode influenciar significativamente os consumos no dia-a-dia. Finalmente, os autores aconselham uma possível previsão com utilização de RNA, graças a sua alta capacidade de aprendizagem, que poderia resultar mais vantajosa do que os modelos matemáticos.

Outras investigações (Adamowski e Adamowski, 2015) consideraram a utilização das RNA para a previsão de consumos de água. Nomeadamente, desenvolveram um modelo rede neuronal dinâmica (DAN2) para a previsão a curto, médio e longo prazo. No seu trabalho, os investigadores fizeram experiências com variáveis climáticas, mas demonstraram que mediante a utilização de séries temporais da demanda de água, o modelo é capaz de produzir resultados excelentes sem necessidade dessas variáveis.

Os investigadores repararam que o consumo de água é uma série temporal não estacionária e estocástica, que pode incluir uma tendência na média, assim como uma variância não constante e descontinuidades. Outros investigadores também repararam em que existe uma grande semelhança entre os padrões de consumos elétricos e de água, portanto o procedimento de modelação é semelhante.

Em particular, para prever o consumo de água foram utilizados modelos lineares como a regressão linear multivariante ou ARIMA nas últimas décadas. Nos últimos anos têm sido utilizadas as RNA no âmbito da previsão de consumos elétricos, mas poucos autores utilizaram-nas para modelizar e prever consumos e demandas de água (Adamowski e Adamowski, 2015).

Insiste-se na não-inclusão de fatores climáticos ou económicos como variáveis exógenas na previsão de consumos com o modelo de RNA, pois a sua inclusão pode ser difícil ao requerer variáveis futuras que não são conhecidas e seria necessário prever. O resultado da sua inclusão é uma introdução de erro no modelo, que produz previsões inadequadas.

O caso de demanda modelizado com a DAN2 corresponde a uma empresa privada (SJW) em San Jose (Califórnia, EUA), e fornece também as populações de Campbell, Cupertino, Los Gatos, Monte Sereno e Saratoga.

A diferença dos modelos de RNA estáticos, os quais só são capazes de transformar uma entrada de dados (linear ou não linear) num dado de saída, as RNA dinâmicas estabelecem uma relação entre as saídas e as entradas prévias, o qual adiciona uma memória a estas redes. Matematicamente, esta memória traduz-se na aparição de equações diferenciais que formam parte do modelo.

As conclusões foram determinantes: o modelo de rede neuronal artificial dinâmica atingiu uma adequação aos dados excelente, muito melhor do que os modelos ARIMA que também foram desenvolvidos para realizar a comparação e outros modelos matemáticos e/ou estatísticos. Aliás, a inclusão de variáveis exógenas apenas melhora numa percentagem insignificante os resultados, que junto com a complexidade implícita na sua inclusão no modelo, não justificou a sua utilização no mesmo.

O facto de existirem periodicidades sazonais e semanais nos padrões de consumo diários, assim como periodicidades diárias nos padrões de consumos horários, para além do carácter sequencial dos dados, indica que é necessário tratar os dados de consumos como séries temporais e, portanto, considera-se necessária a inclusão do estudo e caracterização das séries temporais na revisão bibliográfica.

Em vista das conclusões dos trabalhos no ramo da previsão de consumos de água anteriormente expostos e da necessidade de levar a cabo uma previsão dos consumos deixando à margem métodos matemáticos que precisam de correções para aproximar a realidade, surge a justificação da utilização das RNA. Uma caracterização das RNA será, portanto, abordada neste capítulo.

## 2.2 GENERALIDADES

As Redes Neurais Artificiais, também simplesmente conhecidas como Redes Neurais (em inglês, *Neuronal Networks* o *Artificial Neuronal Networks*) são um tipo de ferramentas computacionais que trabalham de forma semelhante aos processos biológicos do cérebro. Uma RNA pode ser definida como uma rede com várias unidades de processamento simples (nós ou neurónios), os quais têm uma pequena quantidade de memória local, interligados por canais de comunicação (ligações) que transportam dados numéricos. Os neurónios operam apenas com os *inputs* e os dados locais que recebem através das ligações. As operações matemáticas das RNA são muito mais simples do que as suas homólogas, as redes biológicas (Srinivasa e Brion, 2005).

Embora a história das redes neuronais apenas se tenha iniciado em 1888, quando o científico espanhol Santiago Ramon y Cajal mostrou que o sistema nervoso é composto por uma rede de células individuais que estão interligados (Daza, 2008), foi em 1943 quando McCulloch e Pitts introduziram os modelos simples de neurónios biológicos e iniciaram o desenvolvimento das RNA. Mesmo assim, o seu modelo não possuía capacidade de aprendizagem. No ano 1949, Hebb descreveu a sua bem conhecida regra da aprendizagem "*Hebbian Learning Rule*". Taylor apresentou em 1956 uma rede de memória associativa a partir da regra Hebbiana. Em 1958, Rosenblatt criou o perceptrão através da utilização de um algoritmo iterativo de aprendizagem baseado no gradiente descendente fornecido ao neurónio artificial do McCulloch e do Pitts. Desde que Taylor demonstrou o teorema da convergência, foi criada uma grande excitação nos investigadores. Rosenblatt escreveu em 1959 um livro de neurocomputação titulado "*Princípios de Dinâmica neuronal*", traduzido do inglês "*Principles of Neurodynamics*" (Srinivasa e Brion, 2005).

Logo, em 1960, Widrow e o seu estudante Hoff desenvolveram outro tipo de elemento computacional chamado ADALINE (ADAPtative LINEar Element). Muitos investigadores relataram implementações bem-sucedidas e a aplicação da neuro-computação. A maior parte dos investigadores fizeram experiências utilizando redes neuronais em vez de com estudos analíticos (Matich, 2001).

Minsky e Papert (1969) demonstraram que um perceptrão de apenas uma camada não poderia efetuar algumas funções lógicas simples como a disjunção exclusiva (porta lógica XOR). Apesar de ser isto apenas uma limitação para o perceptrão de uma única camada (perceptrão simples), não se conhecia nenhum

algoritmo para perceptrões interligados em múltiplas camadas. Com os argumentos de Minsky e Papert, a maior parte do financiamento foi revertido para outros campos durante mais de uma década. Apenas alguns continuaram os seus esforços neste campo. As RNA só reapareceram depois de importantes provas realizadas no início da década de 1980; a *back-propagation* ou propagação para trás foi um dos mais importantes (Matich, 2001).

Alguns dos usos mais frequentes das RNA são a classificação de dados ou *grouping* e a aproximação de funções ou *mapping*. A maior parte das aplicações práticas das RNA utilizam aprendizagem supervisionada. Para aprendizagem supervisionada, os dados de treino têm de incluir ambos os inputs e os targets (objetivos) ou os resultados esperados. Depois do treino bem-sucedido, dados de entrada ou inputs podem ser introduzidos sem necessidade de introduzir *targets*, e a RNA computará os dados de saída ou *outputs* (Srinivasa e Brion, 2005).

Exemplos práticos de aplicação na área da engenharia incluem (Samarasinghe, 2006):

- Propriedades do comportamento dos sistemas naturais: como o tempo e a variação do espaço na contaminação dos aquíferos e sistemas atmosféricos.
- Compreender fatores de geração de resíduos e uma previsão de longo prazo da produção de resíduos.
- Previsão de caudais nos rios e lagos.
- Integração de muitos parâmetros de gestão para proporcionar uma solução eficaz para a gestão de um sistema.
- Gestão de Reservatórios.
- Previsão da demanda elétrica e de água.
- Previsões económicas.

Uma definição mais ampla de uma rede neuronal é a seguinte: é uma coleção de neurónios interligados que incrementalmente aprende com seu ambiente (dados) para capturar as tendências essenciais lineares e não lineares nos dados, de modo que fornece previsões confiáveis para novas situações.

As redes neuronais podem executar uma variedade de tarefas, incluindo a previsão ou função de aproximação (Figura 2.1), de classificação de padrões (Figura 2.2), agrupamento (Figura 2.3), e previsão, tal como mostrado na Figura 2.4. Imagens adaptadas de Samarasinghe (2006).

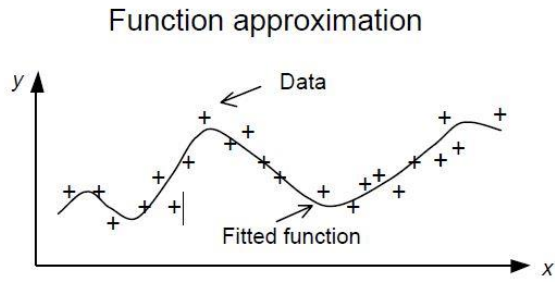


Figura 2.1 – Aproximação de funções.



a) Classificação de dados: atribuição dos dados numa classe.

b) Classificação do sinal: atribuição dum série temporal numa classe.

Figura 2.2 – Classificação.

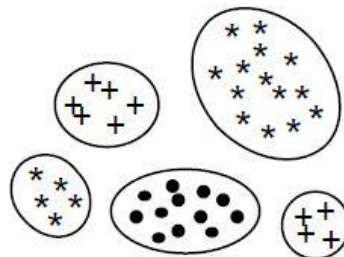


Figura 2.3 – Agrupamento ou *clustering* não supervisionado: encontrar grupos desconhecidos nos dados.

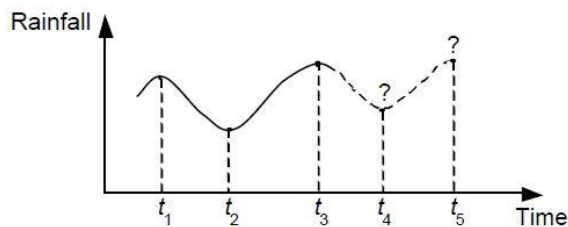


Figura 2.4 – Previsão ou *forecasting*: prever os próximos resultados de uma série temporal.

Para atingir os objetivos expostos no presente trabalho, o tipo de tarefa a ser executada será o *forecasting* ou previsão de consumos, devido à própria natureza do problema.



### 2.3 RAZÕES PARA A UTILIZAÇÃO DAS REDES NEURONAIS ARTIFICIAIS

Apesar de existirem diversos métodos para avaliação e previsão de consumos, nenhum deles é capaz de gerar esses padrões tendo em conta algumas variáveis cujas inter-relações não são conhecidas em profundidade. Os modelos tradicionais, baseados em métodos estatísticos, além de complexos, não atingem um grau de precisão e generalização desejável. Assim, justifica-se a utilização de novas ferramentas que permitam o estudo integrado e mais eficiente do problema apresentado.

As razões para o crescimento na utilização das RNA tanto em engenharia como em ciências estão relacionadas com características do cérebro humano, e benefícios são os seguintes (Matich, 2001):

- Aprendizagem adaptativa. Capacidade para aprender a realizar tarefas baseadas no treino ou uma experiência inicial.
- Auto-organização. Uma rede neuronal pode criar a sua própria organização ou representação da informação que recebe mediante uma etapa de aprendizagem.
- Robustez e tolerância a falhas. Se alguns neurónios falharem, a rede pode ter um bom desempenho. A destruição parcial de uma rede conduz a uma degradação da sua estrutura; no entanto, algumas capacidades da rede podem-se reter, mesmo que sofram algum dano.
- Generalização – Se for introduzido na rede um novo conjunto de dados diferente dos exemplos conhecidos, é produzido o melhor resultado possível a partir dos exemplos usados pela rede.

Estes e outros benefícios traduzem-se na prática no atingimento de boas soluções mesmo com dados que tenham falhas, imprecisão ou sujeitos a ruído. Mesmo assim, é importante ter presente que os outros métodos também apresentam vantagens, e cada um deles deve ser utilizado preferencialmente na sua área de aplicação.

### 2.4 PARALELISMO DAS REDES NEURONAIS COM O SISTEMA NERVOSO

As redes neuronais são outra forma de emular certas características próprias dos humanos, tais como a capacidade de memorizar e associar eventos. Se examinar cuidadosamente os problemas que não podem ser expressos através de um algoritmo, nota-se que todos eles têm uma coisa em comum: a experiência. O ser humano é capaz de resolver estas situações, através da experiência acumulada.

Então, parece claro que uma forma de abordar o problema consiste na construção de sistemas que são capazes de reproduzir essa característica humana. Em soma, as redes neuronais não são nada mais do que um modelo artificial e simplificado do cérebro humano, que é capaz de adquirir conhecimento através da experiência (Matich, 2001).

Assim, este modelo inspira-se na estrutura e funcionamento do sistema nervoso, cujo elemento fundamental é o neurónio. Cada célula neuronal consiste num corpo de célula, que contém o núcleo central e uma ou mais estruturas chamadas axónios e dendrites. Na Figura 2.5 apresenta-se o diagrama simplificado de um neurónio.

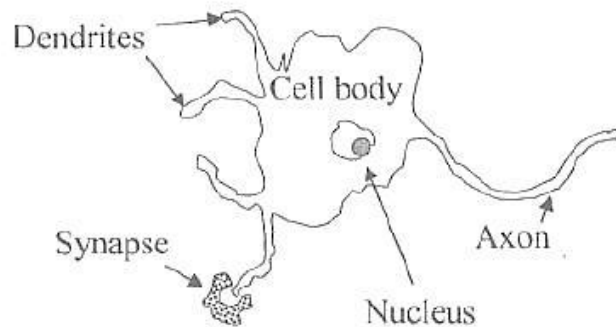


Figura 2.5 – Diagrama de um neurónio, adaptado de Srinivasa e Brion (2005).

O sinal nervoso é recebido nas dendrites dos neurónios e percorre ao longo do axónio. O axónio mantém o equilíbrio químico aumentando a concentração de iões de potássio e de sódio de dentro para fora. Quando um sinal é transmitido, a bainha de mielina em torno do axónio permite a passagem dos vários iões. Os iões de sódio e potássio mudam a sua posição. Assim, é gerado um sinal elétrico que se desloca ao longo da membrana. O espaço entre dois neurónios é chamado fenda sináptica e o processo de transmissão de informação sinapse, em que as vesículas descarregam neurotransmissores químicos que transportam os impulsos nervosos para o neurónio seguinte e assim por diante, a formação de uma vasta rede de comunicações. Esta célula processa internamente a informação recebida de outros neurónios, e envia o resultado para os seguintes neurónios aos quais está ligado.

Pode, então, ser estabelecida uma analogia entre a atividade sináptica e as RNA: os sinais que chegam ao neurónio através da sinapse são as entradas (inputs); estes são ponderados (atenuados ou amplificados) através de um parâmetro, chamado "peso associado", com a sinapse correspondente. Estes sinais de entrada podem excitar o neurónio (sinapse com o peso positivo) ou inibir (peso negativo). O efeito total é a soma das entradas ponderadas. Os neurónios têm um limiar de ativação, que é a quantidade de "entrada" necessária para ativar o neurónio, de modo que, se a soma é igual a ou maior do que o limiar do neurónio, este é ativado (transmitido), caso contrário, não é ativado. Esta é uma situação de tudo ou nada; cada neurónio é ativado ou não ativado, mas não há uma resposta intermédia. Esta capacidade de ajustar sinais é um mecanismo de aprendizagem (Nielsen, 2015).

Na Figura 2.6 apresenta-se um modelo biológico de 3 neurónios e a sua representação como rede neuronal artificial.

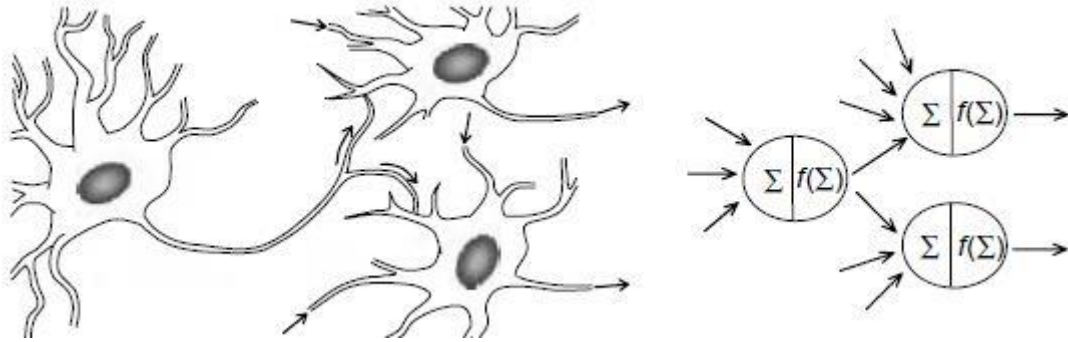


Figura 2.6 – Modelo biológico e a sua representação como Rede Neuronal Artificial, adaptado de Srinivasa e Brion (2005).

Os dendritos são as ligações entre os inputs ou entradas e os corpos das células, ponderados com pesos. No corpo da célula, os inputs ponderados ou sinais são recebidos, acumulados ou somados ( $\Sigma$ ) e processados depois [ $f(\Sigma)$ ] para produzir um output ou saída. O axónio transmite o sinal de saída ou output a outros neurónios ligados a ele.

## 2.5 TOPOLOGIA, ARQUITETURA E TIPOS DE REDES NEURONAIS ARTIFICIAIS

A topologia de uma RNA é a organização dos neurónios ligados na rede, formando camadas ou grupos de neurónios. A este respeito, os parâmetros principais são:

- O número de camadas.
- O número de neurónios por camada.
- O grau de conectividade.
- O tipo de ligações entre os neurónios.

Dependendo do **tipo de ligação**, as redes podem ser classificadas como de propagação para a frente ou para trás.

As redes não recorrentes ou *feedforward* são relativamente simples; os neurónios em cada nível só apresentam uma ligação com os neurónios de níveis posteriores e, por tanto, a informação é propagada para a frente; os sinais entram pela camada de entrada, atravessam as camadas ocultas e chegam até a camada de saída. Não há ligação com as camadas prévias. Exemplos de redes não recorrentes são: perceptrão simples, Adaline, Madaline, Linear Adaptive Memory ou Drive-Reinforcement.

As redes recorrentes ou *feedback* são aquelas nas quais a informação circula tanto para a frente como para atrás. Ao introduzir retroalimentação, obtêm-se um sistema dinâmico, mais complexo, no qual a resposta da rede será estabilizada após um determinado número de interações. Assim, os neurónios aceitam qualquer ligação; com níveis prévios, do seu próprio nível (ligações laterais, com os neurónios da

sua própria camada) e mesmo consigo próprias (ligação autorecorrente). Todas as ligações descrevem-se de seguida (Daza, 2008):

- Ligações intracamada ou laterais: são estabelecidas entre os neurónios de uma mesma camada.
- Ligações intercamada: são produzidas entre neurónios de diferentes camadas.
- Ligações realimentadas: aquelas que têm direção oposta à entrada-saída (direita à esquerda).
- Ligações autorrecorrentes: são as ligações de realimentação de um neurónio consigo próprio.

A rede autorecorrente é o tipo de rede que será utilizada ao longo desta dissertação. Neste tipo de rede, as saídas são utilizadas como neurónios de entrada.

De acordo com o **modo de treino ou aprendizagem**, as RNA podem classificar-se como supervisionadas ou não supervisionadas. Este termo refere-se à necessidade de registos de valores reais. Ou seja, numa rede de aprendizagem supervisionada, o treino tem de ser feito para cada entrada com os correspondentes alvos ou *targets*, e será capaz de fornecer alvos para qualquer nova entrada após informação suficiente.

Ao contrário, quando apenas se treina uma rede com um conjunto de entradas ou inputs o tipo de aprendizagem é não supervisionada, que será capaz de encontrar a estrutura ou as relações entre as diferentes entradas. A maior parte da aprendizagem não supervisionada é utilizada para agrupamento ou *clustering* (Samarasinghe, 2006).

No âmbito deste trabalho, serão utilizadas redes supervisionadas, pois são introduzidos na rede os diferentes inputs ou entradas e os seus correspondentes targets ou objetivos, associadas com cada input.

Segundo o **número de camadas**, pode distinguir-se entre redes de só uma camada (monocamada ou perceptrão simples) e as redes de múltiplas camadas (perceptrão multicamada).

As redes monocamada são as mais simples, pois apenas há uma camada de neurónios, os quais enviam as entradas a uma camada de saída onde são realizados diferentes cálculos (sendo esta a única que processa informação, pois na camada de entrada não é realizado qualquer cálculo).

Nas redes multicamada existe um conjunto de camadas intermédias (camadas ocultas) entre as camadas de entrada e saída, as quais estão interligadas. No âmbito deste trabalho será utilizado o perceptrão multicamada, uma vez que o perceptrão simples apenas é utilizado em tarefas de autoassociação, onde a rede aprende certa informação  $A_1, A_2 \dots A_n$ , de modo que, quando é apresentada uma informação de entrada, a rede realizará uma autocorrelação, respondendo com um dos dados armazenados; o mais parecido com os dados de entrada (Ariza Ramírez, 2013).

A arquitetura das RNA é formada por unidades de processamento denominadas neurónios ou nós, ligações entre estes, constantes (*bias*), funções de aptidão (*fitness functions*) e algoritmos de treino. Uma vez definida a topologia (número de camadas e distribuição de nós e ligações), uma matriz de dados ( $\mathbb{R}^n$ ) é utilizada para treinar a RNA. A única propriedade das ligações é o seu peso, o qual é multiplicado pelo nível de ativação do nó a montante para produzir a contribuição para o nó a jusante.

As operações que podem ser realizadas dentro dos nós das RNA são 3 (Ariza Ramírez, 2013):

- Função de propagação ou entrada (também chamada função de excitação), que geralmente consiste na soma de cada entrada multiplicada pelo peso de interligação (valor neto). Se o peso for positivo, a ligação é chamada excitatória; se for negativo, é chamada de inibidora.
- Função de ativação, que modifica o anterior. Modifica o estado de excitação dos nós.
- Função de transferência ou saída, que é aplicada ao valor de retorno da função. É usada para limitar a saída do neurónio e, geralmente, é dada pela interpretação que se deseja dar a essas saídas.

A representação esquemática das funções aplicadas num nó é ilustrada na Figura 2.7.

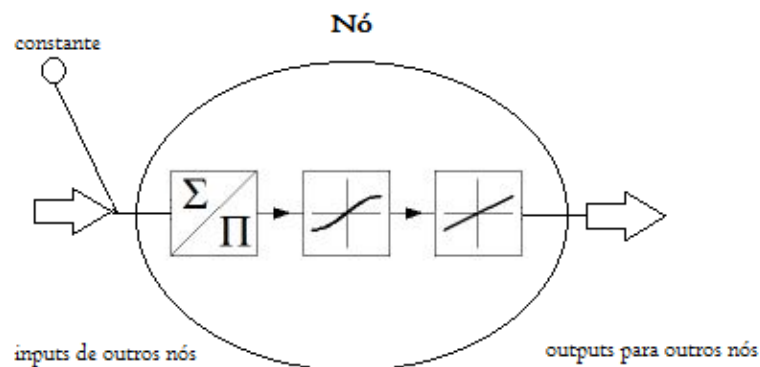


Figura 2.7 – Funções aplicadas num neurónio

### 2.5.1 Função de entrada

O nó trata os valores de entrada como se fossem apenas um valor; isto é conhecido como entrada global. Por tanto, existe o problema de como combinar essas entradas simples dentro da entrada global. Para resolver este problema, são utilizadas as funções de entrada:

$$input_i = (in_{i1} \cdot w_{i1}) * (in_{i2} \cdot w_{i2}) * \dots (in_{in} \cdot w_{in}) \quad (2.6)$$

Onde:

\* – representa o operador apropriado (somatório, máximo, etc.);

$n$  – representa o número de entradas no neurónio  $N_i$ ;

$w_i$  — é o peso.

As funções de entrada mais comuns são:

- 1) Somatório das entradas pesadas: é a soma de todos os valores de entrada ao neurónio, multiplicados pelos seus pesos correspondentes.

$$\sum_j (n_{ij}w_{ij}) \quad (2.7)$$

- 2) Produto das entradas pesadas é o produto de todos os valores de entrada ao neurónio, multiplicados pelos seus pesos correspondentes.

$$\prod_j (n_{ij}w_{ij}) \quad (2.8)$$

- 3) Máximo das entradas pesadas: só se tem em consideração o valor da entrada mais forte, previamente multiplicado pelo seu peso correspondente.

$$\max_j (n_{ij}w_{ij}) \quad (2.9)$$

Normalmente, é adotado o somatório das entradas como função de entrada nos nós. Esta consideração é feita ao longo desta dissertação.

### 2.5.2 Função de ativação

Um neurónio biológico pode estar ativo (excitado) ou inativo (não excitado), por tanto tem um *estado de ativação*. Os neurónios artificiais também têm diferentes estados de ativação (Matich, 2001).

As funções de ativação que são introduzidas nos nós de um PMC têm de ser contínuas e diferenciáveis para que o algoritmo de retropropagação possa ser aplicado. Aliás, de acordo com Minsky (1969), pelo menos algumas delas devem ser não-lineares, para que a rede seja capaz de aproximar funções não-lineares, (Matos, 2008).

Cada nó oculto ou de saída numa rede neuronal recebe valores através de ligações com os nós de uma camada anterior e são combinados e agregados num único valor. Geralmente, nas redes *feed-forward* são usadas funções que combinam vetores lineares com escalares. Logo da combinação, o valor escalar é passado através da função de transferência, também conhecida como função de ativação. Esta função é a verdadeira responsável pela introdução da não-linearidade da rede neuronal. A escolha de funções de transferência pode influenciar fortemente a complexidade e o desempenho da rede (Srinivasa e Brion, 2005).

Normalmente, a função sigmoideal ou logística é a mais adotada. Os outputs variam entre intervalos de 0 e 1. A função linear, também chamada função identidade, não pode aproximar muitas funções.

A função de ativação combina a entrada total no  $j$ -ésimo neurónio ( $Net_j$ ) com o estado inicial do neurónio [ $a_j(t - 1)$ ], para produzir um novo estado de ativação consoante com a nova informação recebida [ $a_j(t)$ ] (Ariza Ramírez, 2013).

$$a_j(t) = f[a_j(t - 1), Net_j(t)] \quad (2.10)$$

Muitos modelos de RNA consideram que o estado atual do neurónio não depende do estado prévio, de maneira que a expressão anterior pode expressar-se assim:

$$a_j(t) = f[Net_j(t)] = f \left[ \sum_{i=0}^N W_{ij}(t) \cdot X_i(t) \right] \quad (2.11)$$

De seguida, apresentam-se as funções de ativação mais utilizadas, apesar de existirem mais.

- a linear, ilustrada na Figura 2.8 e representada pela equação (2.12);
- a sigmoideal ou logística, ilustrada na Figura 2.9 e representada pela equação (2.13);
- a tangente hiperbólica, ilustrada na Figura 2.10 e representada pela equação (2.14);
- a arco-tangente, ilustrada na Figura 2.11 e representada pela equação (2.15).

$$f(x) = x \quad (2.12)$$

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.13)$$

$$f(x) = 2 \cdot \frac{1}{1 + e^{-x}} - 1 \quad (2.14)$$

$$f(x) = \frac{2}{\pi} \cdot \arctan\left(\frac{\pi}{2} \cdot x\right) \quad (2.15)$$

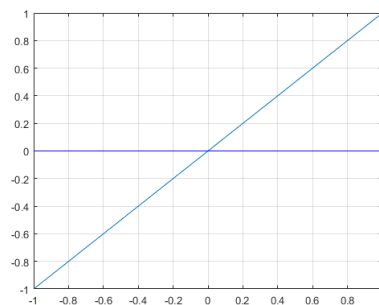


Figura 2.8 – Função de ativação linear

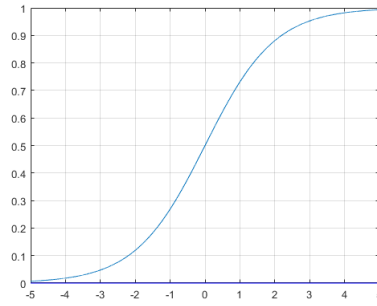


Figura 2.9 – Função de ativação sigmoideal

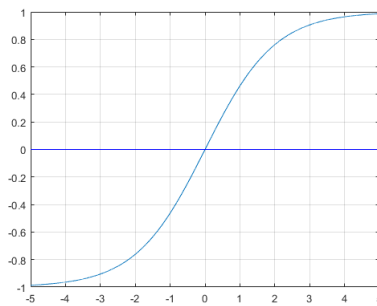


Figura 2.10 – Função de ativação tangente hiperbólica

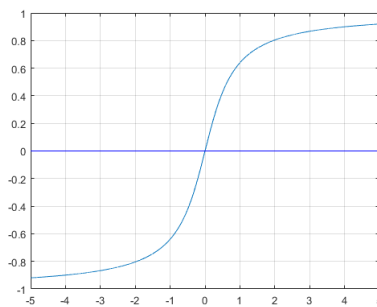


Figura 2.11 – Função de ativação arco-tangente

Não é necessário que as funções de todas as camadas sejam idênticas, mas é habitual que todos os nós da mesma camada tenham a mesma função de ativação. Normalmente, as camadas ocultas dos PMC estão compostas por funções de ativação não lineares, do tipo sigmoideal ou tangente hiperbólica, sendo comum adotar a função de ativação linear na camada de outputs, devido a não convergir assintoticamente, sem limitar os possíveis resultados.



De seguida, apresenta-se, na Tabela 2.1, um resumo das características das funções anteriormente descritas.

Tabela 2.1 – Características das principais funções de ativação.

Nome	Função	Derivada	Intervalos de saída
Função linear	$f(x) = x$	$\frac{dy}{dx} = 1$	Sem limite
Função logística ou sigmoideal	$f(x) = \frac{1}{1 + e^{-x}}$	$\frac{dy}{dx} = y(1 - y)$	0 a 1
Função Tangente hiperbólica	$f(x) = 2 \cdot \frac{1}{1 + e^{-x}} - 1$	$\frac{dy}{dx} = 1 - \operatorname{tgh}^2(y)$	-1 a 1
Função arcotangente	$f(x) = \frac{2}{\pi} \cdot \arctan\left(\frac{\pi}{2} \cdot x\right)$	$\frac{dy}{dx} = \frac{1}{1 + y^2}$	-1 a 1

Os valores introduzidos na RNA são padronizados, por isso há que voltar a escalar os outputs no final do treino. É por esta razão que, se a normalização dos resultados utilizados para treinar a rede foi de 0 para o menor dos valores e de 1 para o maior, uma vez que os resultados gerados são assintoticamente convergentes para 1, a rede neuronal nunca poderá prever valores superiores aos registados no grupo de treino (Srinivasa e Brion, 2005).

### 2.5.3 Função de saída

O último componente que um neurónio necessita é a *função de saída*. O valor resultante desta função é a saída do neurónio. Se a função de ativação está por abaixo de um limite determinado, nenhuma saída passa ao neurónio seguinte (Matich, 2001). Os dois tipos de funções de saída mais comuns são:

- Nenhuma: é o tipo de função mais simples, pois a saída é a mesma que a entrada. Também é chamada *função identidade*.
- Binária:  $\begin{cases} 1 & \text{ativ}_i \geq \alpha \\ 0 & \text{no caso contrário} \end{cases}$ , sendo  $\alpha$  o limite estabelecido.

## 2.6 MODELOS DE REDES NEURONAIS ARTIFICIAIS

De todos os tipos de redes neuronais, o mais importante é o conjunto de redes de aprendizagem supervisionadas, destacando pela sua generalidade e utilidade para as aplicações práticas. Os tipos de redes de aprendizagem supervisionada mais conhecidas são: o Perceptrão Simples, Adeline e Perceptrão Multicamada. O algoritmo de retropropagação apenas é aplicado neste último modelo, por ser o mais usado nas aplicações práticas das redes neuronais (Daza, 2008).

### 2.6.1 Perceptrão

O modelo do **perceptrão simples** (ou, simplesmente, perceptrão), cuja representação esquemática é ilustrada na Figura 2.12) é usado para classificar padrões linearmente separáveis. Foi proposto por Rosenblatt em 1985. É a forma mais simples; rede *feedforward* com uma única camada, formada por um só nó com pesos e constantes ajustáveis. (Srinivasa e Brion, 2005). Cada input  $X_i$  é multiplicado pelo peso da ligação ao nó de output,  $w_{i1}^1$  de maneira a obter-se o valor de output  $X_1^1$ :

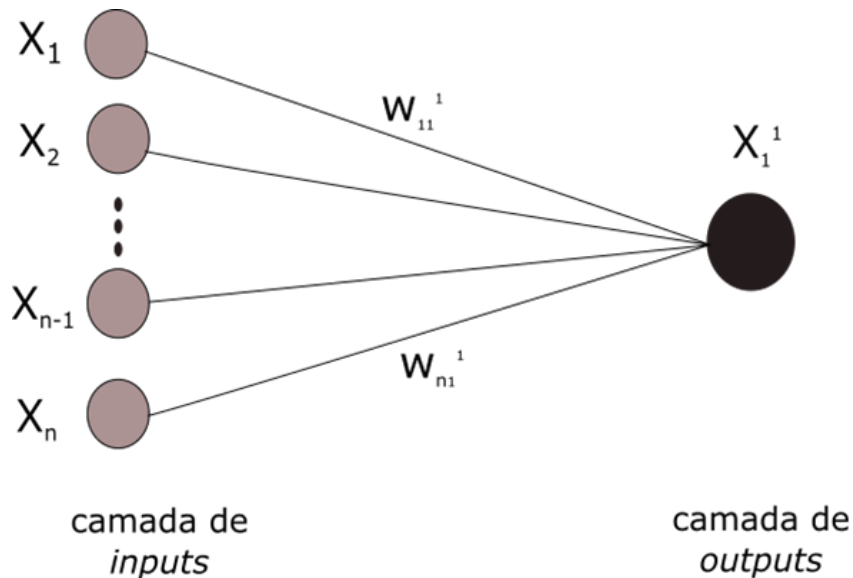


Figura 2.12 – Representação esquemática do perceptrão

$$v_1^1 = \sum_{i=1}^n w_{i1}^1 \cdot X_i \tag{2.16}$$

$$\begin{cases} X(i) = [1, X_1(i), X_2(i), \dots, X_n(i)]^T \\ W = [b_1^1, w_{11}^1, w_{21}^1, \dots, w_{n1}^1]^T \end{cases} \tag{2.17}$$

$$X = [X(1), X(2), \dots, X(j)] \tag{2.18}$$

$$v_1^1(i) = W^T X(i) \tag{2.19}$$

Sempre que as classes sejam linearmente separáveis, existirá um conjunto de pesos  $W$  com o qual  $v_1^1$  será positivo para todos os conjuntos  $\begin{cases} W(i+1) = W(i) - \eta(i)X(i) \\ W(i+1) = W(i) + \eta(i)X(i) \end{cases}$   $X(i)$  pertencentes a uma das classes e negativo no caso contrário.

Se  $X(i)$  for corretamente classificado  $W(i+1) = W(i)$ , não há alterações nos pesos sinápticos. Se o vetor não for classificado com sucesso, os pesos são atualizados de acordo com a regra:

$$\begin{cases} W(i+1) = W(i) - \eta(i)X(i) & \text{se } W^T(i)X(i) > 0 \text{ e } X(i) \in \text{Classe B} \\ W(i+1) = W(i) + \eta(i)X(i) & \text{se } W^T(i)X(i) \leq 0 \text{ e } X(i) \in \text{Classe A} \end{cases} \tag{2.20}$$

O algoritmo de Rosenblatt para separação de classes é o seguinte (Benítez Sánchez, 2001):

### Algoritmo de Rosenblatt

- 1: Iniciar  $\omega$  aleatoriamente
- 2: *While* haja exemplos mal classificados *do*
- 3:     Selecionar aleatoriamente um exemplo  $(x, d)$
- 4:     Calcular a saída  $y$  e comparar com  $d$
- 5:      $\Delta\omega = \eta(d - y)$
- 6: *End while*

Este algoritmo baseia-se num teorema essencial: O conjunto de dados  $\{(x, d)\}$  é finito e linearmente separável. Desta maneira, o algoritmo anterior (Benítez Sánchez, 2001) encontra uma solução em um tempo finito (converge).

Embora tenha constituído uma importante inovação, o perceptrão simples apresenta limitações devido ao seu comportamento linear (limitações formalizadas por Minsky e Papert, em 1969) e a sua incapacidade para aplicar o algoritmo de retropropagação, com as vantagens que têm associadas. Aliás, o objetivo do estudo não é realizar uma separação linear de classes, mas uma previsão de séries temporais, no entanto considera-se essencial a assimilação do conceito de perceptrão simples para ter noção do funcionamento do perceptrão multicamada.

### 2.6.2 Perceptrão multicamada PMC

O Perceptrão Multicamada (PMC) é baseado no perceptrão simples, o qual tem uma grande limitação: não pode fazer classificações em conjuntos de dados que não sejam linearmente separáveis e, portanto, as tarefas de classificação que podem ser executadas são limitadas. Para superar esta limitação, o perceptrão simples é transformado num perceptrão a vários níveis, através da introdução de uma camada oculta (pelo menos uma).

O PMC, cujo esquema é representado na Figura 2.13, é uma rede maior, mais complexa e de maior poder computacional do que o perceptrão simples e constitui a rede mais amplamente utilizada. O algoritmo utilizado para o treino é o conhecido por retropropagação do erro ou *backpropagation*, e baseia-se na regra delta (a detalhar adiante).

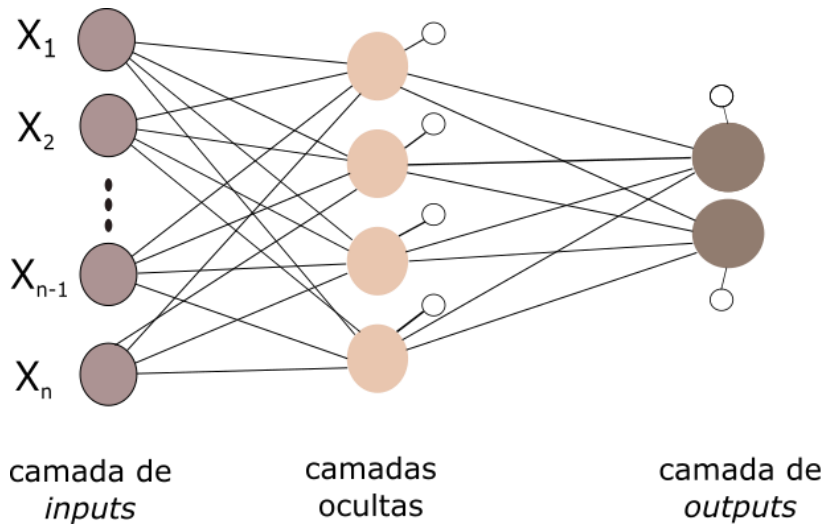


Figura 2.13 – Representação esquemática de um PMC com uma camada oculta

Cada neurónio de uma camada, exceto os neurónios da camada de entrada, recebe entradas dos neurónios da camada anterior e envia a sua saída para os neurónios na camada seguinte sem existirem ligações de realimentação para trás nem de tipo lateral na mesma camada. A disposição dos neurónios é realizada por camadas com uma forte restrição sobre a ligação é não são permitidas ligações que criem ciclos. A vantagem da adição de camadas ocultas é a expansão do espaço de hipóteses que a rede é capaz de representar (Daza, 2008).

Os PMC, por serem aproximadores universais, têm potencial para aproximar qualquer função contínua (com qualquer erro desejado), mas os resultados variam na prática, pois o número necessário de nós e dados de entrada aumenta com a complexidade da função a aproximar (Haykin, 1999).

Seja  $\varphi(\cdot)$  uma função não constante, estritamente crescente e contínua.

Seja  $I_{m_0}$  um hiper-cubo unitário como  $m_0$  dimensões.

O espaço das funções contínuas em  $I_{m_0}$  é designado  $c(I_{m_0})$ .

Então, dada qualquer função  $f \in C(I_{m_0})$  e  $\epsilon > 0$ , existe um inteiro  $M$  e conjuntos de constantes reais  $\alpha_i, b_i$  e  $w_{ij}$ , em que  $i = 1, \dots, m_1$  e  $j = 1, \dots, m_0$ , tais que é possível definir:

$$F(x_1, \dots, x_{m_0}) = \sum_{i=1}^{m_1} \alpha_i \varphi \left( \sum_{j=1}^{m_0} w_{ij} x_j + b_i \right) \quad (2.21)$$

Como uma aproximação da função  $f(\cdot)$ ; ou seja,

$$|F(x_1, \dots, x_{m_0}) - f(x_1, \dots, x_{m_0})| < \epsilon \quad (2.22)$$

Para todos os  $x_1, \dots, x_{m_0}$  que pertencem ao padrão de dados.

### 2.6.3 Rede NARx

O modelo NARx (*Nonlinear AutoRegressive network with eXogenous inputs*) é um modelo não linear autoregressivo com entradas externas.

As redes neuronais recorrentes são de interesse quando dados passados possuem correlação com os resultados futuros de uma amostra, como no caso das séries temporais. Utiliza-se o conceito de atraso (*delay* ou *lag*); o que na prática significa que se faz uma referência a  $N$  posições anteriores no conjunto de saída. A diferença entre um PMC recorrente e uma rede NARx, é que esta última é limitada a uma única retropropagação no seu neurónio de saída, em vez de realimentar os neurónios da sua camada oculta (Lopes, 2015).

Matematicamente, as redes NARx podem ser representadas pelas equações:

$$y(t) = f(u(t - n_u), \dots, u(t - 1), u(t), y(t - n_y) \dots, y(t - 1)) \quad (2.23)$$

$$y(t + 1) = f(u(t - n_u + 1), \dots, u(t), u(t + 1), y(t - n_y + 1) \dots, y(t)) \quad (2.24)$$

O objetivo é, portanto, prever a série  $y(t)$  dados  $d$  valores passados de  $y(t)$  e outra série de *inputs*  $x(t)$ . Na Figura 2.14 ilustra-se o esquema de uma rede NARx.

A arquitetura é parecida com o PMC multicamada. De facto, o PMC forma parte da rede: existe uma camada de entrada da função  $x(t)$  (sobre a qual é não atua a retropropagação) e uma camada de entrada dos *targets*,  $y(t)$ , a qual proporciona a retroalimentação; camadas ocultas com nós em que são aplicadas funções de somatório de pesos sinápticos e funções de ativação não lineares (constituindo o PMC), e, por fim, uma camada de saída única retroalimentada pela camada de entrada  $y(t)$  e com função de ativação linear. A retropropagação não se produz pelas camadas ocultas, mas da camada de saída para as camadas de entradas externas (Lopes, 2015).

O funcionamento do modelo NARx baseia-se na criação duma janela temporal, definida a partir da incorporação das  $p - 1$  entradas anteriores e das  $q$  saídas anteriores. Uma rede NARx com ordem de entrada  $p$  e ordem de saída  $q$ , é definida pelas equações:

$$y_i[t] = g_y \left( \sum_{j=1}^{n_z} W_{i,j}^{y,z} z_j[t] + W_i^y \right) \quad (2.25)$$

$$z_i[t] = g_z \left( \sum_{j=1}^{n_u} \sum_{k=1}^p W_{i,j(k)}^{z,u} u_j[t - k + 1] + \sum_{j=1}^{n_y} \sum_{k=1}^q W_{i,j(k)}^{z,y} y_j[t - k] + W_i^z \right) \quad (2.26)$$

Onde:

$n_z$  – é o número de nós na camada oculta;

$n_y$  – é o número de nós na camada de saída;

$W_i^y, W_i^z$  – são constantes (*bias*) aplicadas, respetivamente, sobre as camadas de saída e oculta, para contribuir à modificação do estado de ativação do nó;

$z_i[t], y_i[t]$  – são as saídas da camada oculta e os outputs, respetivamente;

$W_{i,j}^{z,u}, W_{i,j}^{y,z}$  – são os pesos sinápticos aplicados sobre os inputs provenientes da camada de entrada,  $u$ , nas ligações dela com a camada oculta, e os aplicados sobre os outputs da camada oculta na camada de saída  $y$ .

$g_y, g_z$  – são funções de ativação nas camadas de saída e na oculta, respetivamente.

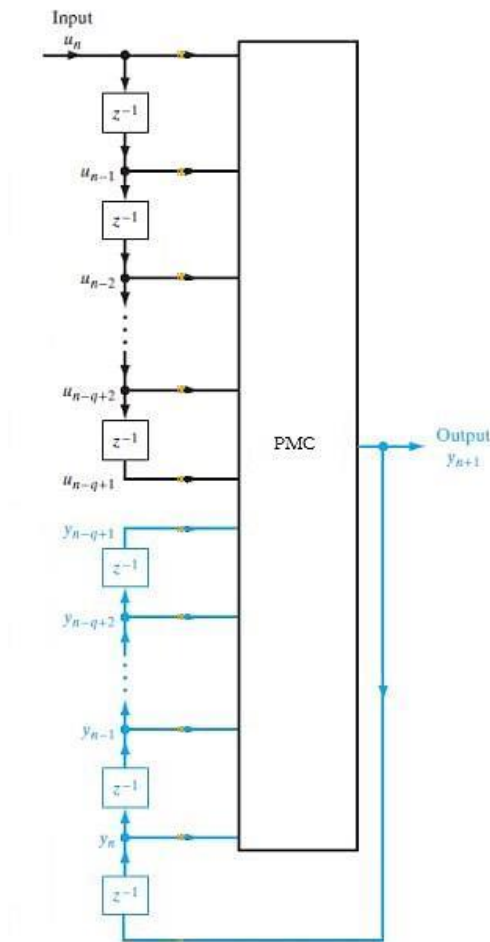


Figura 2.14 – Representação esquemática de uma rede NARx.

### 2.6.4 Rede LSTM

O modelo de memórias a curto e longo prazo (em inglês, *Long-Short Term Memory* ou, abreviadamente, LSTM), treinado pelo gradiente descendente, é capaz de resolver problemas muito complexos para os modelos tradicionais (PMC, por exemplo), e supõe uma melhora considerável dos resultados. Os LSTMs

são projetados explicitamente para evitar o problema de dependência a longo prazo. Reter a informação por longos períodos de tempo é praticamente o seu comportamento padrão. Portanto, os LSTM são um tipo de rede ótimo para previsão em séries temporais (Pérez, 2002).

Em vez de nós, as redes LSTM possuem blocos de memória ligados através de camadas. Um bloco possui componentes que o tornam mais inteligente do que um nó clássico e tem memória para sequências recentes.

A componente básica do modelo LSTM (criado por Hochreiter e Schmidhuber em 1997) é o bloco de memória, o qual contém uma ou mais células de memória, uma porta de entrada e uma porta de saída. As portas são unidades multiplicativas com ativação contínua (normalmente, dentro do intervalo  $[0,1]$ ) e são partilhadas por todas as células que pertencem a um mesmo bloco de memória. Cada célula contém uma unidade linear com uma ligação recorrente local chamada “carrossel de erro constante” (CEC). A ativação do CEC é conhecida como “estado da célula” (Pérez, 2002).

Na Figura 2.15 apresenta-se um bloco de memória com duas células partilhando as portas do bloco. A entrada de cada célula representa-se com  $Z$ , a ativação da porta de entrada com  $\phi$ , a ativação da porta de saída com  $\gamma$ , a ativação do CEC com  $x$  e a ativação global da célula com  $z$ .

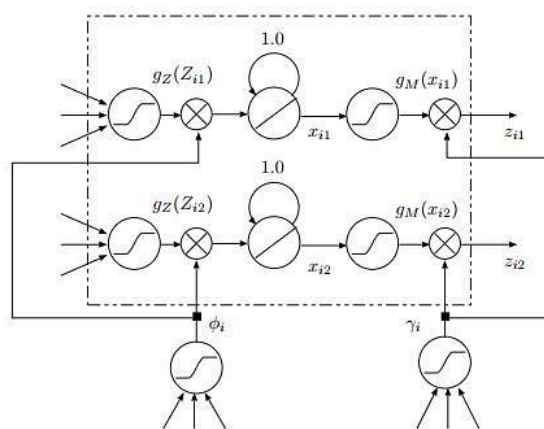


Figura 2.15 – Representação esquemática de um bloco de memória com duas células, adaptado de Pérez (2002).

Cada uma das células recebe como entrada uma coleção de valores (ponderados mediante os pesos correspondentes) da entrada da rede e das saídas de todas as células do modelo no instante prévio. A porta de entrada é responsável por permitir o impedir o acesso destes valores no CEC do interior da célula. A porta de saída realiza uma ação semelhante sobre a saída da célula, permitindo ou atenuando a difusão do estado do CEC no resto da rede (Pérez, 2002).

Sejam  $n_U, n_Y, n_M, n_C$  o número de nós de entrada, saída, blocos de memória e células por bloco, respetivamente. A entrada no tempo  $t$  denota-se como  $u[t]$  e a saída correspondente como  $y[t]$ . A saída da  $j$ -éssima célula do bloco  $i$ -éssimo é representada como  $z_{it} [t]$ .

Os superíndices indicam o cálculo em que está envolvido o peso: o “ $\phi, z$ ” em  $W^{\phi,z}$  indica que o peso é usado para calcular a ativação de uma porta de entrada ( $\phi$ ) a partir da ativação da célula ( $z$ ); o “ $\gamma$ ” em  $W^\gamma$  indica que a constante (*bias*) é usada para calcular a ativação de uma porta de saída. Os subíndices indicam as unidades particulares afetadas pelo peso.

### Cálculo da função de saída

A ativação da porta de entrada ( $\phi$ ) do bloco de memória  $i$  é calculada como:

$$\Phi_i[t] = \sum_{j=1}^{n_M} \sum_{k=1}^{n_C} W_{i,jk}^{\phi,z} z_{jk}[t-1] + \sum_{j=1}^{n_U} W_{i,j}^{\phi,u} u_j[t] + W_i^\phi \quad (2.27)$$

$$\phi_i[t] = g_c(\Phi_i[t]) \quad (2.28)$$

Onde

$g_c$  — é a função de ativação de todas as portas da rede (rede sigmoideal, normalmente).

A ativação da porta de saída ( $\gamma$ ) do bloco de memória  $i$  é calculada como:

$$\Gamma_i[t] = \sum_{j=1}^{n_M} \sum_{k=1}^{n_C} W_{i,jk}^{\gamma,z} z_{jk}[t-1] + \sum_{j=1}^{n_U} W_{i,j}^{\gamma,u} u_j[t] + W_i^\gamma \quad (2.29)$$

$$\gamma_i[t] = g_c(\Gamma_i[t]) \quad (2.30)$$

O estado interno da célula de memória é calculado mediante a soma da entrada modificada pela porta correspondente com o estado no instante anterior  $t - 1$ :

$$x_{ij}[t] = x_{ij}[t-1] + \phi_i[t] g_z(Z_{ij}[t]) \quad (2.31)$$

Onde

$g_z$  — é uma função de ativação (normalmente, sigmoideal) e:

$$Z_{ij}[t] = \sum_{k=1}^{n_M} \sum_{l=1}^{n_C} W_{ij,kl}^{z,z} z_{kl}[t-1] + \sum_{j=1}^{n_U} W_{ij,k}^{z,u} u_k[t] + W_{ij}^z \quad (2.32)$$

Com  $x_{ij}[0] = 0$  para todo  $ij$ . A saída da célula é calculada ajustando o estado do CEC através duma nova função de ativação  $g_M$  e multiplicando o resultado pela ativação da comporta de saída:

$$Z_{ij}[t] = \gamma_i[t] g_M(x_{ij}[t]) \quad (2.33)$$



Caso seja permitida uma ligação direta da entrada e os neurónios de saída, a saída global da rede é calculada como:

$$Y_i[t] = \sum_{j=1}^{n_M} \sum_{k=1}^{n_C} W_{i,jk}^{y,z} z_{jk}[t] + \sum_{j=1}^{n_U} W_{i,j}^{y,u} u_j[t] + W_i^y \quad (2.34)$$

$$y_{ij}[t] = g_Y(Y_i[t]) \quad (2.35)$$

Onde

$g_Y$  – é uma função de ativação adequada (por exemplo, sigmoial).

Os pesos que incidem nas portas de entrada e saída frequentemente são iniciados de forma que  $\phi_i [0]$  e  $\gamma_i [0]$  estejam pertos de 0; assim, os blocos de memória estão desativados no início, e o treino é focado nas ligações diretas entre a entrada e os neurónios de saída. De esta maneira, o protagonismo dos blocos de memória vai aumentando consoante ao algoritmo de aprendizagem realizar as suas iterações.

O número de pesos a ajustar numa rede LSTM é:  $(n_M n_C + n_U + 1)(n_Y + n_M n_C + 2n_M)$ .

Na Figura 2.16 ilustra-se o funcionamento de uma rede LSTM com dois blocos de memória de duas células cada um.

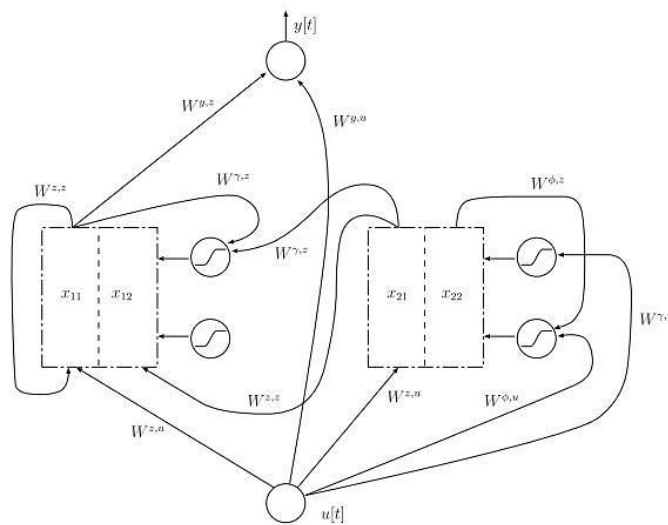


Figura 2.16 – Representação esquemática do funcionamento numa rede LSTM, adaptado de (Pérez, 2002).

## 2.7 TREINO DAS RNA

### 2.7.1 Introdução

Neste ponto realiza-se a análise do treino das RNA em pormenor; apresenta-se o algoritmo de retropropagação e a sua formulação, abordam-se os conceitos de generalização e da validação cruzada.

### 2.7.2 A retropropagação

O algoritmo de retropropagação é utilizado para atualizar pesos sinápticos no interior das RNA. É similar ao método dos mínimos quadrados, mas com a inovação de ter capacidade de generalizar em redes multi-camada. O seu nome deriva da propagação de erros para trás a partir da camada de saída em direção à camada de entrada, através de camadas ocultas intermédias e pesos sinápticos, de modo a reduzir o erro (Daza, 2008).

Este algoritmo calcula as derivadas do erro de treino com base nos pesos definidos através da regra da cadeia e a otimização do gradiente descendente para ajustar os pesos e assim minimizar o erro.

O objetivo de levar a cabo o treino da rede é ajustar os pesos sinápticos de maneira que a rede gere as saídas ou outputs desejados em resposta aos padrões de entrada. O algoritmo de retropropagação é um algoritmo de aprendizagem supervisionado, o qual implica que, para cada input ou entrada há um output ou saída correspondente, o qual atua como objetivo. Durante o treino, qualquer diferença entre o output e o objetivo é tratada como um erro, o qual tem de ser minimizado. Os aspetos que influem na retropropagação são os seguintes (Daza, 2008):

- Número de camadas e nós. Não existem métodos definitivos para escolher o número ótimo de camadas e/ou neurónios nas RNA. Normalmente, são determinados experimentalmente.
- Conjunto de treino. Novamente, a experiência é imprescindível para determinar esta característica. A única afirmação certa é que para uma boa aprendizagem do problema, ou uma boa generalização dos exemplos, o conjunto deve abarcar todo o espectro do domínio.
- Preparação dos dados. Geralmente, é realizado um processamento prévio para adaptar os dados reais de forma que o treino tenha um custo computacional menor e o erro quadrático médio possa ser calculado de forma mais exata.
- Velocidade de convergência. O algoritmo de propagação é relativamente lento em relação a outros algoritmos de treino.
- Mínimos locais. No método do gradiente descendente (variação do algoritmo de retropropagação), os cálculos podem ficar “presos” num mínimo local, do qual não pode sair e,

então, a aprendizagem não é processada de forma ótima. Para evitar este problema, é realizada uma aprendizagem várias vezes, partindo de pesos diferentes a cada vez e escolhendo a melhor solução ao problema.

- Sobretreino. Pode ocorrer quando a rede não é capaz de produzir boas respostas ao apresentar vetores de entrada diferentes ao conjunto de treino, não generalizando para o total de casos possíveis. Os pormenores do sobretreino serão expostos nas seguintes subsecções.
- Estabilidade e robustez. Refere-se a variações nos pesos e/ou nas entradas. Existem medidas de estabilidade estatística, mas no âmbito deste trabalho serão utilizados métodos integrados no próprio processo de treino da rede.

### A Regra delta

Quando um vetor de entrada é apresentado à rede, o erro de saída pode ser calculado como o erro quadrático médio ou EQM. O erro quadrático médio é calculado como a soma dos quadrados das diferenças entre os valores-alvo e valores de saída (Zarza Cano, 2012).

$$E = \frac{1}{2} \sum_{l=1}^L (y_l - z_l)^2 \quad (2.36)$$

Onde

$L$  – é o comprimento total do vetor da amostra de dados;

$l$  – é o índice de cada dado de  $L$ .

Desta expressão, normalizada para a totalidade do grupo de treino, define-se o erro quadrático médio, em que  $N$  é o número total de padrões de teste.

$$E_{méd} = \frac{1}{N} \sum_{i=1}^N E(i) \quad (2.37)$$

Se o modo de treino for estocástico, o erro a considerar será  $\bar{E}$ . Caso contrário, se o modo de treino for determinístico, o erro a considerar será  $E_{méd}$ .

O algoritmo de retropropagação é um algoritmo que funciona em duas fases.

1. O padrão é apresentado à rede e propaga-se para jusante, sendo o erro calculado no fim deste processo.
2. A rede é atualizada; o gradiente local dos nós de saída é calculado e são adaptados os pesos sinápticos das ligações. Logo, são calculados os gradientes locais e os pesos sinápticos para montante são atualizados até atingir a camada de input.

Uma vez que acaba o processo de adaptação, um novo padrão é apresentado à rede, até que o erro atinja o valor desejado ou se interrompa o treino. Em vez de se efetuar a retropropagação entre padrões, apenas se fazem alterações aos pesos sinápticos entre épocas (Zarza Cano, 2012).

Para reduzir o erro na rede, é preciso uma mudança dos pesos de forma que o erro tenda a zero. Isto é conhecido como minimização do erro mediante gradiente descendente, como foi mencionado anteriormente. Vem definido por:

$$\frac{\partial E}{\partial(\omega_{ij}^l, \omega_{ij}^o, \theta_j^l, \theta_j^o)} \equiv 0 \quad (2.38)$$

Onde

$\omega_{ij}^l, \omega_{ij}^o$  – são os pesos de entrada e saída;

$\theta_j^l, \theta_j^o$  – são as *bias* ou constantes de entrada e saída;

Para levar a cabo esta minimização, é preciso utilizar métodos matemáticos de convergência. O algoritmo original baseava-se no método do gradiente. Não obstante, é possível modificar o método para criar outros métodos mais sofisticados. Alguns deles são:

- Levenberg-Marquardt.
- Adam.
- Adadelta.
- RMSprop.

O método de Levenberg-Marquard produz resultados geralmente bons em redes de pequeno e médio tamanho tanto para o EQM como para os tempos de treino, utilizando a segunda derivada. O método de Adam também produz resultados bons em tempos de treino aceitáveis. Ambos serão utilizados para o treino das redes.

Adam é um algoritmo de otimização de primeira ordem baseado, como já foi indicado, no gradiente de funções objetivo estocásticas, com base em estimativas de adaptação dos momentos de ordem inferior. O método é simples de implementar, é computacionalmente eficiente, tem requisitos de memória pequenos, é invariante para reescalonamento dos gradientes diagonais, e é bem adequado para os problemas com elevado número de dados. O método é também adequado para problemas não-estacionários (séries temporais) com muito ruído. Os resultados empíricos demonstram que Adam funciona bem na prática e se compara favoravelmente com outros métodos de otimização estocástica.

### 2.7.3 Generalização

Haykin, (1999) afirmava que os PMC generalizam bem quando o mapeamento de inputs e outputs realizado é correto ou praticamente correto para dados de teste, nunca usados na criação ou treino da rede.

Por tanto, a capacidade de generalização de uma RNA se refere ao seu desempenho ao prever padrões de teste que não foram utilizados durante o treino. A incapacidade de generalização de uma RNA pode ser outorgada aos seguintes fatores:

- o sobre-ajuste (*overfitting*)
- o sobre-treino (*overtraining*).

Os PMC podem armazenar informação nos nós, sempre que haja um número elevado. No entanto, se o número de nós não se adaptar ao número de padrões e à complexidade da função, existe o risco de ocorrência de sobre-ajuste ou *overfitting*. Há dois problemas que surgem se a RNA gerar espaços de decisão mais complexos do que o requerido pelo problema em questão (Matos, 2008):

1. Se a topologia da rede for mais complexa que o necessário, haverá desperdício de esforço computacional. Isto é conhecido como custo computacional.
2. Menor poder de generalização. A complexidade do espaço de decisão de uma RNA é proporcional ao número de graus de liberdade que esta tem (isto é, o número de parâmetros utilizados). A RNA tende a se adaptar aos valores específicos dos dados durante o treino, o qual pode produzir uma diminuição na fiabilidade da rede para fazer previsões. Para evitar o *overfitting*, é preferível que a arquitetura seja o mais simples possível, de modo que seja atribuída mais relevância às maiores regularidades nos dados e as irregularidades serão ignoradas, pois podem ser fruto de ruídos. Também se pode reduzir este problema por meio de um aumento no número de dados ou padrões de treino, mas isto não reduzirá o custo computacional. Segundo Haykin, (1999), os fatores dos que depende a capacidade de generalização são:
  - a. A complexidade do problema (função a aproximar).
  - b. O tamanho e representatividade do grupo de treino.
  - c. A arquitetura da rede.

Uma vez que o primeiro e segundo pontos não variam para este problema (consideram-se fixos para o período de estudo), só cabe atuar sobre a arquitetura da rede (número de nós e a sua distribuição).

Na Figura 2.17 ilustra-se diferentes funções de ajuste para uns dados dispersos. No primeiro caso, à esquerda, a função utilizada (linear) não se ajusta bem aos pontos (existe “sub-ajuste” ou *underfitting*). De seguida, a função exponencial parece denotar um bom ajuste para todos os pontos de dispersão representados e para outros possíveis pontos que poderiam aparecer. Por último, no gráfico à direita, exhibe-se um ajuste polinómico de grau 6, o qual integra praticamente todos os pontos de dispersão, mas não seria válido para outros pontos da série, apesar de ter um alto coeficiente de correlação, pois ao interpolar não se atingiriam valores verdadeiros. Neste caso, a função utilizada não está a generalizar para outros valores, sendo muito específico para os valores do gráfico, mas incapaz de prever novos valores (*overfitting*).

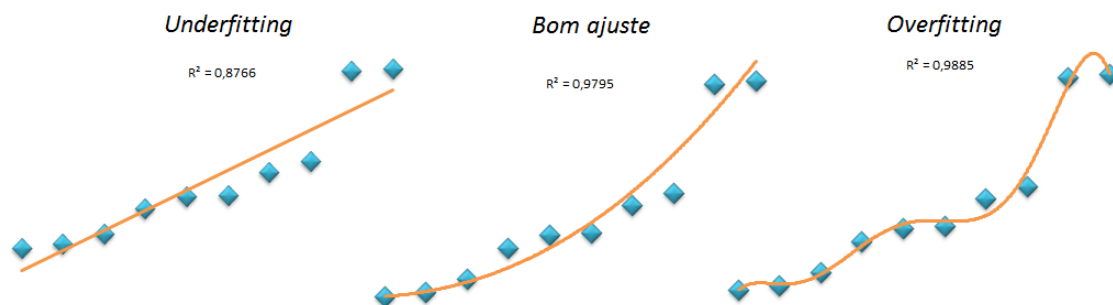


Figura 2.17 – Utilização de diferentes funções de ajuste para uns pontos dispersos.

O Sobre-treino ou *overtraining* apresenta uma característica em comum com o *overfitting*: ambos são originados pela adaptação da RNA às singularidades do conjunto de treino (dados de treino), mas com a diferença principal de que, no *overtraining*, o problema ocorre quando a rede neuronal é treinada com um conjunto muito grande de exemplos de padrões com muitas iterações (número de épocas de treino). O resultado é uma capacidade de generalização reduzida. Uma maneira de evitar *overtraining* é determinar qual é o número adequado de épocas de treino. Para determiná-lo, pode ser criado um gráfico como o da Figura 2.18 e identificar o ponto mínimo da curva de EQM do conjunto de testes, pois quando a curva cresce, está a produzir-se *overtraining* (Campos, 2000).

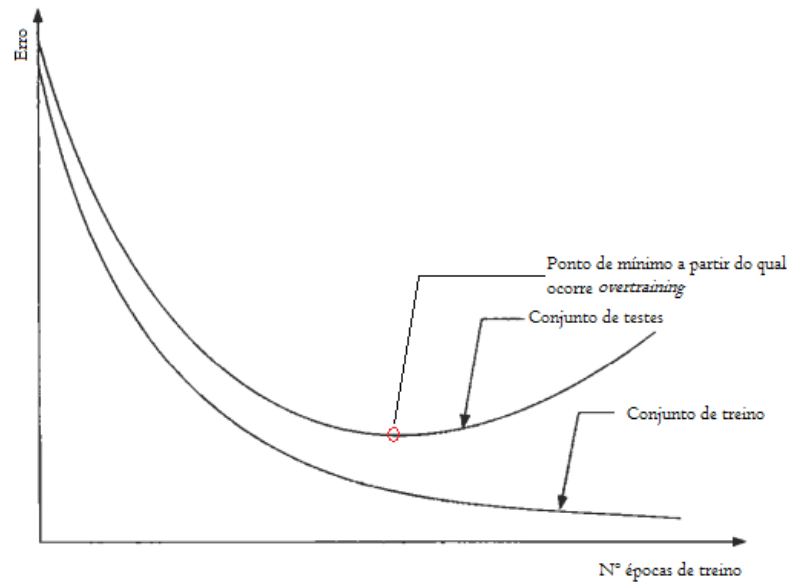


Figura 2.18 – Curvas de treino e teste nas que se produz *overtraining*. Adaptada de Theodoris e Koutroumbas (2003).

Em relação à aproximação de funções, o PMC pode aproximar qualquer tipo de função, sempre que seja contínua (teorema do aproximador universal). Mesmo assim, o número de nós e de padrões disponíveis para treino necessários não é fácil de definir. Então, a arquitetura da rede tem de confrontar dois aspetos opostos:

- Por uma parte, há de existir um número mínimo de nós para que a função possa ser devidamente aproximada.
- Por outra parte, um grande número de nós pode degradar a capacidade de generalização da rede.

Na prática, é frequente realizar experiências com várias redes, levando a cabo uma análise da capacidade de generalização, com a finalidade de seleccionar a melhor arquitetura mediante um método conhecido como validação cruzada. Outra opção complementária é iniciar o treino com uma arquitetura complexa, eliminando a posteriori as ligações que armazenam mais ruído, isto é, aquelas que contam com os pesos sinápticos mais “fracos” (Campos, 2000).

#### 2.7.4 Validação cruzada

Validar um modelo é obter um grau de confiança nele para o propósito para o qual está previsto. Não há uma “demonstração” da validade do modelo, mas uns resultados positivos num conjunto de teste que fornece essa confiança. A validade de um modelo pode arruinar-se por causa de apenas um teste negativo (Matos, 2008).

A validação cruzada ou *cross-validation* é uma técnica utilizada para treinar e avaliar a capacidade de generalização de uma rede neuronal: o conjunto de dados é dividido em três subconjuntos, nomeadamente:

- Conjunto de treino.
- Conjunto de validação.
- Conjunto de teste.

Assim, o conjunto de treino é usado para determinar os pesos sinápticos, parâmetros livres da rede, enquanto o grupo de validação é utilizado para conhecer a capacidade de generalização da rede, sem intervir de forma direta no treino, sendo possível escolher a melhor arquitetura da rede. Em relação ao grupo de teste, a sua missão é garantir que a rede adotada não ficou com sobre-treino para os padrões do grupo de validação.

Existe uma variante da validação cruzada conhecida como paragem antecipada (em inglês, *early stopping*), na qual é adotado um processo pelo qual o treino é interrompido quando a capacidade de generalização começa a degradar-se. Na paragem antecipada, ao contrário do que sucede na validação cruzada, o erro também é estimado para o subconjunto de validação, de forma que o processo de treino finaliza quando o erro começar a crescer. O subconjunto de teste serve, novamente, para garantir os resultados atingidos (Ariza Ramírez, 2013).

Vários autores, como Brownlee,(2016) mencionam a percentagem do conjunto de dados a adotar como conjunto de validação, mas não é uma tarefa fácil. Habitualmente, aconselha-se adotar um conjunto de validação em relação à complexidade da função que se deseja aproximar e o número de padrões disponível. Há distintos métodos para calcular essa percentagem para o caso da paragem antecipada. Os mais conhecidos são:

- Validação cruzada *k-fold*: é o método estreia. Fornece uma estimativa robusta do desempenho de um modelo, dividindo o conjunto de dados de treino em  $k$  subconjuntos e alternando em modelos de treino em todos os subconjuntos, exceto um que é mantido para fora, e avalia o desempenho do modelo no conjunto de dados de validação. O processo é repetido até que todos os subconjuntos tenham uma oportunidade de ser o conjunto de validação retido. A média de desempenho é então média em todos os modelos que são criados.
- Validação cruzada *leave one out*: método muito parecido com o anterior; nele, apenas um padrão é deixado à parte em cada treino, isto é,  $K$  é igual ao número de padrões de dados.



No entanto, o método *k fold* muitas vezes não é utilizado para avaliar modelos de RNA por causa do maior custo computacional. Por exemplo, para  $k=5$ , devem ser construídos e avaliados 5 modelos, aumentando excessivamente o tempo de avaliação. No entanto, quando o problema é pequeno ou se houver recursos computacionais suficientes, a validação *k-fold* pode fornecer uma estimativa menos tendenciosa do desempenho de um modelo.

## 2.8 ANÁLISE DAS SÉRIES TEMPORAIS

Uma série temporal é o resultado de uma sequência de dados, observações ou valores compreendidos em determinados intervalos de tempo e estão dispostas por ordem cronológica. Normalmente, estes intervalos de observação são geralmente regulares, isto é, hora a hora, diariamente, mensalmente, etc. (Samarasinghe, 2006).

Seja qual for o intervalo de observação, uma característica fundamental das séries temporais é a relação de dependência entre observações sucessivas. Isto implica que a análise de séries temporais deve ser realizada em conformidade com a ordem temporal em que foram feitas as observações.

Segundo o tipo de observações realizadas, as séries temporais podem ser discretas ou contínuas. Uma série temporal é determinística quando é possível prever com precisão os valores, enquanto que se o futuro não pode ser totalmente previsto por observações passadas, a série temporal é estocástica. A evolução do consumo de água é uma série temporal do último tipo, porque, embora seja possível obter valores muito próximos aos valores reais previstos, os vários fatores que o influenciam tornam praticamente impossível saber os valores futuros com precisão (Ariza Ramírez, 2013).

Um processo estocástico é um conjunto de variáveis aleatórias, ocupando uma posição dentro de um conjunto de observações. Assim, uma série temporal na qual não podem ser conhecidos com precisão os valores futuros é uma realização parcial de um processo estocástico. Diz-se parcial porque um processo totalmente estocástico segue um caminho com infinitos elementos, enquanto que o número de elementos de uma série temporal é limitado.

Para caracterizar completamente um processo estocástico, é necessário definir tanto a distribuição de probabilidade conjunta como a marginal de todas as variáveis aleatórias de um conjunto de observações. De um modo geral, para se obter a distribuição conjunta do processo requer a observação de um grande número de dados. Em muitos casos (como o caso de estudo) isso não é possível. Neste tipo de situações, para caracterizar o processo estocástico são utilizadas (Fernández Jiménez, 2016):

- A função de médias:

$$E[y_t] = \mu(t) \tag{2.39}$$

- A função de variâncias:

$$\text{Var}(y_t) = \sigma^2(t) \quad (2.40)$$

- A função de covariâncias:

$$\text{Cov}(y_t, y_{t+j}) = E[(y_t - \mu(t))(y_{t+j} - \mu(t+j))] \quad (2.41)$$

Muitos eventos que acontecem na vida quotidiana podem ser explicados total ou parcialmente mediante séries temporais, pois podem justificar o comportamento passado de uma variável ao longo do tempo e permitir a identificação de certa inércia no comportamento do mesmo, a fim de antecipar desenvolvimentos futuros. Atualmente é um dos principais focos de estudo da comunidade científica.

### 2.8.1 Classificação das séries temporais

As séries temporais podem ser classificadas segundo diferentes aspetos (Ariza Ramírez, 2013).

- Em função do número de respostas ou variáveis que evoluem no período de tempo de observação.
  - Séries univariantes. Apenas existe uma variável de estudo e em cada período de observação só é recolhido o valor que adota essa variável. Por exemplo, o estudo da temperatura ambiente diária ao longo de um ano, representado na Figura 2.19.

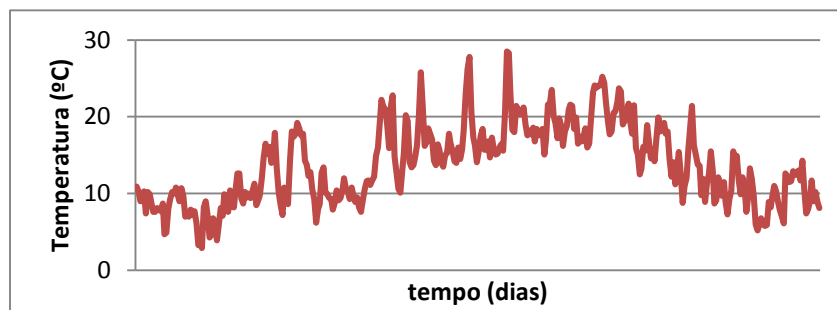


Figura 2.19 – Variação da temperatura do ar numa determinada localização.

- Séries multivariantes: apresentam mais do que uma variável de estudo e em cada período de observação é recolhido o valor de cada uma delas. Um exemplo é a evolução o consumo de combustível e o número de vendas de veículos ou a evolução do consumo diário médio de água em relação à temperatura média diária do ar, como mostrado na Figura 2.20.

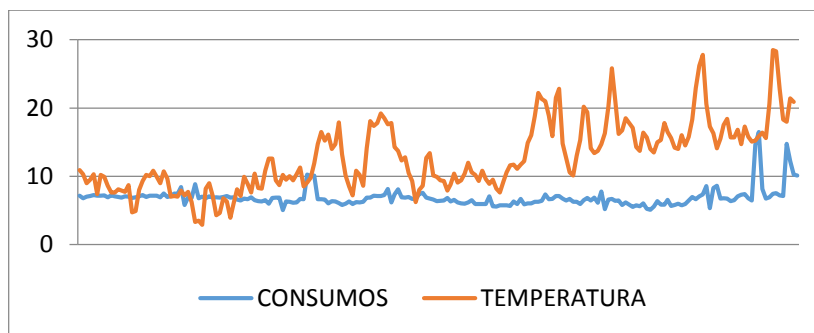


Figura 2.20 – Variação dos consumos diários de água *versus* temperatura do ar

- Em função da sua estacionariedade:
  - Séries estacionárias na média: os valores oscilam em torno de uma média constante com uma variabilidade que também se mantém constante com respeito de dita média ao longo do tempo. São séries estáveis, sem aumentos ou diminuições dos seus valores. Normalmente são utilizadas para a modelagem de processos físicos.
  - Séries não estacionárias na média: são séries com médias que mudam ao longo do tempo. As mudanças na média mostram uma tendência para crescer ou diminuir a longo prazo e, por tanto, a série não oscila em torno de um valor constante. Normalmente são utilizadas em estudos socioeconómicos.
  - Séries estacionárias na variância: a maior parte dos valores situam-se à mesma distância respeito da média, isto é, apresentam um grau de dispersão constante no tempo.
  - Séries não estacionárias na variância: os valores não se encontram à mesma distância em relação à média ao longo do tempo. Ao representar graficamente os valores, podem-se observar zonas nas quais se produz um estreitamento ou um alargamento desses valores em torno da média.
- Em função da sua sazonalidade:
  - Séries sazonais: uma série tem um comportamento sazonal se apresentar uma tendência que é repetida cada certo período de tempo (horário, diário, mensal...) aproximadamente com a mesma intensidade. Geralmente, este tipo de comportamento aparece em variáveis climáticas, económicas ou sociais.
  - Séries não sazonais: são séries que não apresentam um padrão reconhecível e repetitivo ao longo do tempo. O seu comportamento não é influenciado por nenhuma componente estacional.

### 2.8.2 Análise univariante das séries temporais

As séries temporais univariantes podem ser decompostas em três componentes básicas que explicam a sua variação (Fernández Jiménez, 2016):

$$Y_t = T_t + S_t + I_t \quad (2.42)$$

Estas componentes são:

- Tendência ( $T_t$ ): refere-se às mudanças produzidas a longo prazo em relação à média da série ou, também, à mudança da média nesse período temporal. Induz um movimento suave da série a longo prazo.
- Sazonalidade ( $S_t$ ): induz um padrão na série o qual é repetido a cada período de tempo. O efeito da sazonalidade pode ser medido explicitamente e pode mesmo ser eliminado da série (série “desazonalizada”). Um exemplo de como a sazonalidade influi numa série é o aumento do consumo de água nos meses de verão devido às altas temperaturas.
- Componente aleatória ( $I_t$ ): após a remoção da tendência e a sazonalidade da série, são obtidos valores aleatórios.

Uma vez que as componentes básicas de uma série univariante são conhecidas, é possível agrupar as mesmas:

$$Y_t = f(Y_{t-1}, Y_{t-2}, \dots) + a_t \quad (2.43)$$

$$Y_t = Y_t^* + a_t \quad (2.44)$$

onde:

$Y_t$  – corresponde à série observada.

$Y_t^*$  – é a componente previsível da série.

$a_t$  – é a parte aleatória ou não previsível da série. Esta componente é independente do seu passado e deve ser um processo de ruído branco. Um processo de ruído branco é caracterizado por:

$$E[a_t] = 0, \quad t = 1, 2, \dots \quad (2.45)$$

$$Var[a_t] = \sigma^2(t), \quad t = 1, 2, \dots \quad (2.46)$$

$$Cov[a_t, a_{t-k}] = 0, \quad k = \pm 1, \pm 2 \quad (2.47)$$

As séries de tempo univariantes resultam de grande interesse para a análise e previsão dos consumos de água independentemente do horizonte de tempo utilizado. Em qualquer caso, os modelos que serão

implementados nesta dissertação exigem que, para além da série ser univariante, ser estacionária. Por tanto, é conveniente aprofundar o conceito de estacionariedade.

O conceito de estacionariedade estabelece que a distribuição conjunta de um grupo qualquer de variáveis não é modificado se forem deslocadas no tempo. Desta forma, a estacionariedade pode ser expressa como:

$$F_{t_1, t_2, \dots, t_k}(y_1, y_2, \dots, y_k) = F_{t_1+h, t_2+h, \dots, t_k+h}(y_1, y_2, \dots, y_k) \quad (2.48)$$

A estacionariedade constitui uma condição muito restritiva, pois exige a disponibilidade das funções de distribuição conjunta para qualquer grupo de variáveis do processo estocástico, o qual não sempre é possível. Para estas situações surge o conceito de estacionariedade fraca, a qual é caracterizada pelos aspetos seguintes (Fernández Jiménez, 2016):

- Estabilidade na média. A esperança da variável aleatória  $Y_t$  tem de ser independente do tempo:

$$E[Y_t] = E[Y_{t+d}] = cte \quad \forall d \quad (2.49)$$

- Estabilidade na variância. A variância da variável  $Y_t$  tem de ser constante e finita ao longo do tempo:

$$Var[Y_t] = Var[Y_{t+d}] \neq \infty \quad \forall d \quad (2.50)$$

- Estabilidade na estrutura das covariâncias. A covariância entre dois variáveis só depende da separação entre as mesmas.

$$Cov[Y_t, Y_s] = Cov[Y_{t+d}, Y_{s+d}] \quad \forall d \quad (2.51)$$

Por conseguinte, daqui por diante, as séries estacionárias verificarão, pelo menos, as características acima descritas. As séries temporais univariantes não-estacionárias, a priori, não podem ser usadas no modelo, mas podem ser realizadas determinadas transformações, a fim de obter a sua estacionariedade. Para os casos em que a série tem um valor médio que muda a longo prazo pode ser aplicado diferenciações entre os vários termos da série enquanto que se a variância da série não é estável ao longo do tempo, normalmente são aplicadas transformações logarítmicas ou exponenciais.

### 2.8.3 Processos autorregressivos AR(p)

A dependência dos valores passados numa série temporal produz a aparição de uma determinada inércia na mesma ao longo do tempo. Os modelos mais simples para representar essa inércia são os autorregressivos, os quais aplicam a ideia de regressão simples com a finalidade de representar a dependência linear em um instante de tempo  $t$  entre uma variável aleatória  $Y_t$  e a mesma variável em um momento anterior  $Y_{t-d}$ . De forma geral, para referir-se a um processo autorregressivo, utiliza-se a notação AR(p), onde p refere-se à ordem do mesmo (Fernández Jiménez, 2016).

Caso a série temporal  $Y_t$  seja explicada por um processo autorregressivo de primeira ordem ou AR (1), a sua expressão matemática poderia ser escrita como:

$$Y_t = \phi Y_{t-1} + a_t \quad (2.52)$$

onde:

$\phi$  – é uma variável que há de ser determinada, sendo  $-1 < \phi < 1$

$a_t$  – é um processo de ruído branco com variância sigma ao quadrado. É chamada “inovação” e representa a nova informação adicionada ao processo em cada instante.

Então, para o caso do modelo geral AR(p):

$$Y_t = \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + a_t \quad (2.53)$$

Caso a ordem do modelo AR(p) seja muito elevada, escrever a expressão matemática anterior pode ser entediante. É por isso que existe uma forma mais compacta de expressar a equação de um AR(p), utilizando o operador “atraso” B, definido como um operador linear que, aplicado sobre uma função temporal, proporciona essa mesma função, mas retardada um determinado período de tempo. Outro operador é o operador diferença,  $\nabla$ : relaciona-se com o operador atraso mediante a seguinte expressão:

$$\nabla = (1 - B) \quad (2.54)$$

A série temporal pode expressar-se, então, da seguinte maneira para cada instante  $t - n$ :

$$B z_t = z_{t-1} \quad (2.55)$$

$$B^2 z_t = z_{t-2} \quad (2.56)$$

$$B^p z_t = z_{t-p} \quad (2.57)$$

$$\nabla z_t = (1 - B) z_t = z_t - z_{t-1} \quad (2.58)$$

Uma vez que a série temporal é definida de forma mais compacta, a expressão geral será:

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) z_t = a_t \quad (2.59)$$

#### 2.8.4 Processos de média móvel MA(q)

Os processos autorregressivos da seção anterior são de memória de longo prazo (*long-term memory*), isto é, o valor de uma variável no instante  $t$  é correlacionado com o valor dessa variável num conjunto alargado de instantes anteriores. O problema dos processos AR é a incapacidade para representar séries de muito curto prazo (*short-term memory*), isto é, séries onde o valor atual tem correlação com um número reduzido de valores prévios. Para este caso, são utilizados processos de média móvel ou MA (Fernández Jiménez, 2016).

Num processo de média móvel o valor da variável  $Y_t$  para qualquer instante de tempo  $t$  é uma função da última inovação conhecida e um processo de ruído branco. Geralmente, para se referir a um processo de média móvel, é utilizada a notação MA (q), em que q se refere à ordem do mesmo.

Uma série temporal  $Y_t$  explicada por um processo de média móvel de ordem um, pode expressar-se como (Dannecker, 2006):

$$y_t = a_t - \theta a_{t-1} \quad (2.60)$$

A expressão geral para um processo de média móvel de ordem q, pode ser expressa como:

$$y_t = a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \quad (2.61)$$

Ou, alternativamente, utilizando o operador atraso:

$$y_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) a_t \quad (2.62)$$

### 2.8.5 Processos autorregressivos de média móvel ARMA (p,q)

Um processo ARMA (p,q) é uma combinação de um processo AR (p) e um processo MA(q), de modo a obter as vantagens de cada um deles. Permitem a representação, com poucos parâmetros, de processos cujos primeiros q coeficientes, são arbitrários, enquanto que os p seguintes diminuem com a distância temporal segundo leis simples (Fernández Jiménez, 2016).

Em um processo autorregressivo de média móvel o valor da variável  $y_t$  para um instante de tempo  $t$  depende principalmente de três fatores: o valor da variável  $y_t$  para determinados instantes prévios, o valor das inovações acontecidas nesses instantes prévios e um processo de ruído branco (UNED, 2015).

Assim, se uma variável  $y_t$  estiver representada por um processo ARMA de ordem 1, ARMA (1,1), poderia escrever-se a expressão matemática seguinte:

$$Y_t = \phi_1 Y_{t-1} + a_t - \theta_1 a_{t-1} \quad (2.63)$$

Então, a expressão geral pode ser escrita como:

$$Y_t = \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \quad (2.64)$$

Ou, de forma mais compacta utilizando o operador atraso:

$$(1 - \phi_1 B - \dots - \phi_p B^p) Y_t = (1 - \theta_1 B - \dots - \theta_q B^q) a_t \quad (2.65)$$

### 2.8.6 Processos integrados I(d)

Os processos ARMA só podem ser utilizados para modelar séries temporais estacionárias, o qual limita o seu campo de aplicação, pois a maioria das séries reais não são estacionárias e a sua média muda ao longo do tempo, seguindo uma tendência crescente ou decrescente. No entanto, por vezes, as diferenças

relativas são estacionárias. Estes processos não estacionários com a propriedade de que ao serem diferenciados se obtêm processos estacionários são chamados processos integrados (Fernández Jiménez, 2016).

Estes processos são descritos por  $I(d)$ , sendo  $d$  o número de diferenças necessárias para obter um processo estacionário a partir de um processo não estacionário. Uma propriedade importante é a forma em que desaparece a dependência com o tempo (isto é refletido nas funções de autocorrelação e autocorrelação parcial, as quais serão definidas proximamente). Ao contrário do que acontece nos processos ARMA, onde as autocorrelações diminuem geometricamente até chegar a zero após um número reduzido de atrasos. Neste caso, as autocorrelações diminuem linearmente com o tempo e podem encontrar-se coeficientes de autocorrelação diferentes de zero após um número elevado de atrasos (UNED, 2015).

Se uma série necessitar apenas uma diferença, então é um processo estacionário de ordem um,  $I(1)$ , e a sua expressão matemática será:

$$w_t = \nabla(z_t) = z_t - z_{t-1} \quad (2.66)$$

Da mesma forma que nos outros modelos, é possível deduzir a expressão geral a partir do exemplo anterior, sendo definido como:

$$w_t = \nabla^d(z_t) \quad (2.67)$$

### 2.8.7 Processos autorregressivos integrados de média móvel ARIMA (p,d,q)

Os modelos *ARIMA* ( $p, d, q$ ) são uma combinação de todos os processos estudados previamente. A letra  $p$  corresponde à ordem da parte autorregressiva estacionária, enquanto que a letra  $d$  representa à ordem de integração do processo e a letra  $q$  à ordem da parte média móvel. Por tanto, os processos ARIMA podem ser caracterizados mediante a seguinte expressão:

$$(1 - \phi_1 B - \dots - \phi_p B^p)(1 - B)^d y_t = c + (1 - \theta_1 B - \dots - \theta_q B^q) \quad (2.68)$$

Ou, de maneira mais compacta, utilizando o operador diferença:

$$(1 - \phi_1 B - \dots - \phi_p B^p) \nabla^d y_t = c + (1 - \theta_1 B - \dots - \theta_q B^q) a_t \quad (2.69)$$

### 2.8.8 Processos ARIMA sazonais

No estudo de processos não estacionários há um caso de falta de estacionariedade na média, que é repetido frequentemente; o comportamento sazonal. Como consequência, a média dos valores observados não é constante, mas a evolução da série é previsível e segue um padrão cíclico (Dannecker, 2006).



Uma série tem uma sazonalidade de período  $s$  quando o seu valor esperado varia ao longo do tempo com um padrão que é repetido cada vez que é completado um intervalo de tempo  $s$ . Isto é:

$$E(y_t) = E(y_{t+s}) \quad (2.70)$$

O período de tempo  $s$  determina o número de observações que compoem o ciclo sazonal. Por exemplo,  $s=6$  será uma série semestral (no caso de séries de variação mensal,  $s$  varia em função dos dias de cada mês, com valores de 30,31 ou mesmo 28).

Uma das formas mais simples de modelar a sazonalidade é considerá-la aditiva, isto é, como um efeito constante somado aos valores da série, em concordância com a expressão seguinte:

$$y_t = S_t^{(s)} + n_t \quad (2.71)$$

Da equação anterior pode deduzir-se que não é estacionária pois, com a componente sazonal a assumir diferentes valores, a média não será constante no tempo.

Usualmente, são admitidas três hipóteses para explicar o comportamento da componente sazonal:

- i.  $S_t$  é um processo determinista e, portanto, é uma função de valor constante para um mesmo mês ao longo de diferentes anos, para um mesmo dia ao longo de diferentes meses, etc. Este comportamento é descrito por:

$$S_t^{(s)} = S_{t+ks}^{(s)}, k = \pm 1, \pm 2, \dots \quad (2.72)$$

- ii.  $S_t^{(s)}$  depende de uma constante que varia em função do dia, mes, ano...e à qual se adiciona um processo estacionário de média zero para ter em conta o efeito da variação na sazonalidade (mesmo que os fatores sazonais não sejam constantes, estes evoluem segundo um processo estacionário). Matematicamente, se expressa como:

$$S_t^{(s)} = \mu^{(s)} + V_t \quad (2.73)$$

- iii. Considerar que a componente sazonal varia ao longo do tempo sem oscilar em torno de um valor médio fixo (não segue um processo estacionário). Supõe-se que  $S_t^{(s)}$  evolui segundo um caminho aleatório, em concordância com a expressão a seguir:

$$S_t^{(s)} = S_{t-s}^{(s)} + V_t \quad (2.74)$$

Onde:

$V_t$  – é um processo estacionário de média zero.

Seja qual for a hipótese utilizada, é possível transformar uma série estacional em estacionária através da aplicação de uma diferença sazonal. Por exemplo, partindo da última hipótese:

$$S_t^{(s)} = S_{t-s}^{(s)} + V_t \quad (2.75)$$

$$V_t = \nabla_s y_t^s \quad (2.76)$$

Sendo  $\nabla_s$  o operador diferença sazonal de período  $s$ , relacionado com o operador atraso:

$$\nabla_s = 1 - B^s \quad (2.77)$$

Então, aplicando os operadores obtém-se:

$$\nabla_s y_t = \nabla_s S_t^{(s)} + \nabla_s n_t \quad (2.78)$$

Como conclusão, o operador  $\nabla_s$  transforma um processo sazonal em estacionário.

### 2.8.9 Modelo ARIMA sazonal

É possível converter uma série  $y_t$  não estacionária na sua média e com comportamento sazonal noutra série  $z_t$  estacionária, mediante a transformação (Dannecker, 2006):

$$z_t = \nabla_s^D \nabla^d y_t \quad (2.79)$$

Quando existe sazonalidade,  $D=1$ . Pelo contrário, quando não há sazonalidade,  $D=0$ .

As séries sazonais podem ser representadas por meio de duas aproximações:

- Por um lado, uma solução é somar a dependência sazonal à regular, adicionando aos operadores AR ou MA no operador  $B$  termos  $B^s$ . O problema principal é o excessivo comprimento dos polinómios.
- Por outro lado, é mais simples considerar separadamente a dependência sazonal e a regular de forma multiplicativa. Então, o modelo é chamado ARIMA estacional multiplicativo, e pode definir-se segundo a equação matemática subsequente:

$$\phi_p(B^s)\phi_p(B)\nabla_s^D\nabla^d z_t = \theta_q(B)\Theta_Q(B^s)a_t \quad (2.80)$$

Onde:

$\phi_p(B^s) = (1 - \phi_1 B^s - \dots - \phi_p B^{sP})$  – é o operador AR sazonal de ordem  $P$ ;

$\phi_p(B) = (1 - \phi_1 B - \dots - \phi_p B^P)$  – é o operador AR regular de ordem  $P$ ;

$\nabla_s^D = (1 - B^s)^D$  – indica as diferenças sazonais;

$\nabla^d = (1 - B)^d$  – indica as diferenças regulares;

$\theta_q(B) = (1 - \theta_1 B - \dots - \theta_q B^q)$  – é o operador média móvel regular de ordem  $q$ ;

$\Theta_Q(B^s) = (1 - \Theta_1 B^s - \dots - \Theta_Q B^{sQ})$  – é o operador média móvel sazonal de ordem  $Q$ ;

$a_t \sim N(0, \sigma_a^2)$  – representa um processo de ruído branco.

O modelo ARIMA sazonal pode expressar-se, de forma mais compacta, como  $ARIMA(P, D, Q)_s \times (p, d, q)$ .

### 2.8.10 Ferramentas para a criação do modelo ARIMA

Há várias ferramentas muito úteis para a elaboração do modelo ARIMA. As mais importantes (e que serão utilizadas para a determinação dos parâmetros do modelo) são (UNED, 2015):

- **Gráfico da série temporal.** A observação da evolução da série ao longo do tempo permite identificar algumas características. Por exemplo, tendências que produzem mudanças no valor médio da série, máximos e mínimos, etc. É ideal para uma análise preliminar da série para obter as características básicas para depois aplicar uma análise em profundidade daqueles aspetos de interesse.
- **Função de autocorrelação completa (FAC).** Mede a relação de uma variável com as suas observações prévias. Esta relação pode produzir uma inércia ao longo do tempo. Uma forma de estabelecer o tipo de dependência linear entre variáveis de uma série temporal estacionária separadas  $k$  intervalos de tempo é a utilização da função de covariâncias, mas tem o inconveniente de não ser adimensional. Para evitar este inconveniente, são utilizadas autocorrelações em vez de autocovariâncias. A informação é a mesma, mas com a vantagem de ser adimensional. Matematicamente, a correlação entre observações separadas  $k$  períodos é:

$$\rho_k = \frac{cov(z_t, z_{t-k})}{\sqrt{var(z_t)var(z_{t-k})}} \quad (2.81)$$

A FAC é a representação gráfica da função de autocorrelação, e permite observar o valor dos atrasos em relação ao coeficiente de autocorrelação. A complexidade da sua representação (subidas e descidas) tem a desvantagem de dificultar a determinação da ordem do modelo.

- **Função de autocorrelação parcial (FAP).** É a representação gráfica do valor que adota cada atraso em função do coeficiente de autocorrelação parcial. Permite identificar a ordem do modelo a aplicar. Mede a correlação entre observações separadas  $k$  intervalos uma vez eliminada da relação de dependência entre ambas variáveis a parte devida aos valores das observações intermédias da relação de dependência entre ambas variáveis.

Na Tabela 2.2 resumem-se as características dos processos AR(p), Ma(q) e ARMA (p,q) em relação às ferramentas FAC e FAP.

Tabela 2.2 – Resumo das características principais das FAC e das FAP para os modelos AR(p), MA(q) e ARMA (p,q).

	<b>FAC</b>	<b>FAP</b>
<b>AR(p)</b>	Elevado número de coeficientes não nulos	Todos os coeficientes são nulos exceto os primeiros
<b>MA(q)</b>	Apenas são não nulos os primeiros q coeficientes	Alto número de coeficientes diferentes de zero
<b>ARMA (p,q)</b>	Elevado número de coeficientes não nulos	Elevado número de coeficientes não nulos

## CAPÍTULO 3

### FERRAMENTAS UTILIZADAS

O desenvolvimento do trabalho baseia-se na utilização de várias linguagens de programação para a criação, treino e validação de redes neuronais artificiais, sendo a principal a linguagem Python, mediante o editor de código *PyCharm*, desenvolvido pela JetBrains.

#### 3.1 *PYCHARM*

A linguagem Python é uma linguagem de código aberto, sem fins lucrativos e, por tanto, pode ser usada por qualquer tipo de utilizador.

O *PyCharm* não é apenas um editor de código, mas também um compilador e interpretador do mesmo. Proporciona ao utilizador a utilização dos *packages* ou módulos do Python, sendo relativamente fácil a instalação e desinstalação dos mesmos. Para a criação, treino e validação das RNA é utilizado o módulo *Keras*.

O *Keras* é uma API de redes neuronais de alto nível, escrita em Python, e capaz de funcionar em *TensorFlow* ou *Theano*. A vantagem principal do Python é o facto de ser um software livre. Além disso, permite criar e treinar qualquer tipo de rede e combinações entre elas, sendo, por tanto, muito flexível e personalizável, permitindo variar o adicionar qualquer tipo de parâmetro e, por conseguinte, obter resultados diversos. A sua principal desvantagem é a dificuldade de escrever o código, ao não ter qualquer interface gráfica como outros *softwares*.

Outros módulos são necessários para o desenvolvimento da ferramenta, cada um deles com diversos fins. Os seguintes são imprescindíveis para o funcionamento do *Keras*:

- *SciPy* é um ambiente que inclui diversos packages para realizar operações matemáticas e outras funções científicas. O package inclui:
  - *NumPy*, para implementar cálculo matricial no python. Para além do cálculo matricial, inclui funcionalidades de álgebra linear, transformada de Fourier e geração de números aleatórios.

- *Matplotlib*, para criação de gráficos e outras representações visuais em 2D (histogramas, gráficos de correlação, etc.).
  - *Sympy*, para utilização de simbologia matemática.
  - *SciPy library*. Proporciona rotinas numéricas eficientes, como as rotinas de integração numérica e otimização.
  - *IPython*. É um kernel para o notebook Jupyter.
  - *Pandas*, para trabalhar com estruturas de dados e a sua análise. Soluciona o problema de análise de dados no Python. É limitado a funcionalidades lineares.
- *Scikit-learn*, imprescindível para realizar a análise não linear de dados. Realiza tarefas de classificação, regressão, *clustering* ou agrupamento, redução de dimensionalidade, seleção de modelos e pré-processamento, imprescindíveis para implementar uma RNA.
  - *Statsmodels*, um módulo para Python que proporciona funções e classes para a estimativa de vários modelos estatísticos, para além de testes estatísticos.
  - *TensorFlow* é uma biblioteca de código aberto para aprendizagem automática, desenvolvido pelo Google para satisfazer as suas necessidades de sistemas capazes de construir e treinar RNA para detetar padrões e correlações, semelhantes ao raciocínio e aprendizagem humanos. É, portanto, imprescindível para que o *Keras* possa funcionar, pois define a aprendizagem, enquanto este último implementa a estrutura da RNA e as funções são baseadas no *Tensorflow*. Também é fundamental para definir, otimizar e avaliar expressões matemáticas que incluem vetores e matrizes de dados multidimensionais e tensores. Outra alternativa a *Tensorflow* é o *Theano*, desenvolvido pelo departamento de *machine learning* da Universidade de Montreal.

### 3.2 NEURAL NETWORK TOOLBOX

Por outro lado, também é utilizada a Neuronal Network Toolbox para Matlab™, a qual fornece algoritmos, modelos pré-treinados e aplicativos para criar, treinar, visualizar e simular redes neuronais. Podem-se executar as tarefas de classificação, regressão, agrupamento, redução de dimensionalidade, **previsão de séries temporais** e modelagem e controle de sistemas dinâmicos. A sua vantagem principal é que não é necessário programar ao ter uma interface gráfica, de maneira a ser possível obter resultados em pouco tempo. A sua maior desvantagem é o facto de ter uma menor flexibilidade à hora de definir as RNA ao estarem a maior parte dos parâmetros definidos. Aliás, este software não é de uso livre, ao ser uma toolbox de Matlab.

### 3.3 OUTROS SOFTWARES

Outro dos *softwares* utilizados é o *Microsoft Excel*. É utilizado para diversas tarefas, entre as quais destacam a gestão de dados e a análise dos mesmos. Estas tarefas tornaram-se mais simples através da programação na linguagem *Visual Basic (VB)*.

Para o cálculo dos parâmetros dos diferentes modelos ARIMA implementados, foi utilizado o programa *Gretl*, assim como para a criação das funções de autocorrelação e autocorrelação parcial. No entanto, as FAC e FAP também foram criadas na linguagem Python. O *Gretl* é orientado a análise econométrica, mas é uma ferramenta muito potente para qualquer estudo estatístico. É um *software* de livre distribuição.

Como apoio estatístico, também foi utilizado o programa STATGRAPHICS, uma potente ferramenta de análise de dados que combina uma ampla gama de procedimentos analíticos com gráficos interativos e funções analíticas avançadas que permite um rápido cálculo de parâmetros estatísticos, utilizados para verificar que os mesmos parâmetros obtidos no Python são corretos.

### 3.4 HARDWARE UTILIZADO

Considera-se importante definir as características principais do hardware no qual é realizada a presente dissertação, pois algumas limitações, como a falta de memória RAM ou os tempos computacionais, dependem delas.

- Fabricante: ASUSTeK Computer Inc.
- Processador: Intel (R) Core (TM) i5-2450M CPU @2.5 GHz.
- Memória RAM: 8.00 GB (6.66 GB utilizáveis).
- Tipo de sistema: Sistema operativo de 64 bits.
- Sistema Operativo: Windows 7 Home Premium.





## CAPÍTULO 4

### CARACTERIZAÇÃO E ANÁLISE DO CASO DE ESTUDO

#### 4.1 CASO DE ESTUDO

O estudo foca-se na estimativa do consumo de água no concelho de Arouca, pertencente à Área Metropolitana do Porto e da Região Norte de Portugal. Está integrado no distrito de Aveiro. O concelho está subdividido em 16 freguesias e a sua sede é a vila de Arouca, situada na União das Freguesias de Arouca e Burgo.

De seguida, apresenta-se um esquema das freguesias que compõem o concelho, na Figura 4.1, e a percentagem territorial ocupada por cada freguesia, na Figura 4.2.

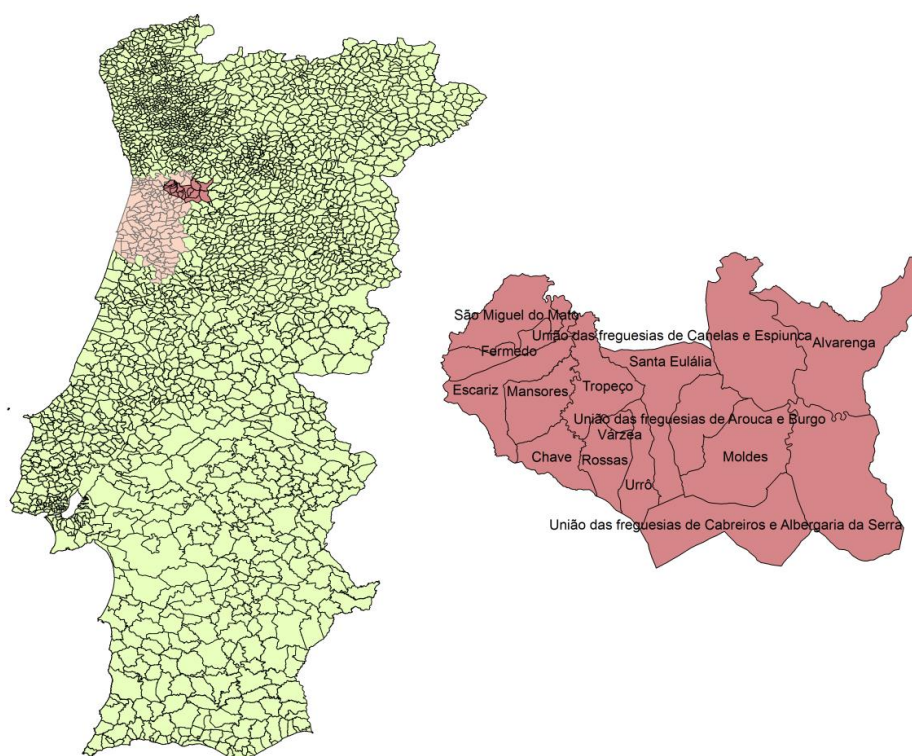
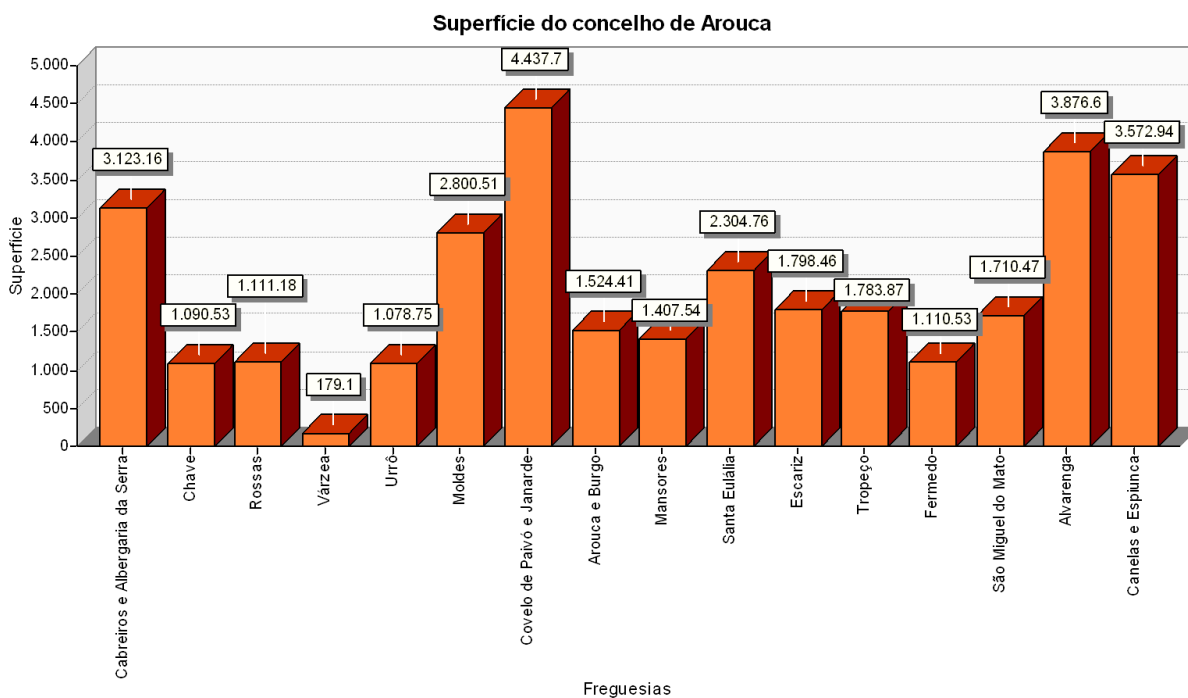


Figura 4.1 – Localização do concelho de Arouca e a sua divisão em freguesias

Na elaboração dos mapas cartográficos, foram utilizados dados do CAOP (Carta Administrativa Oficial de Portugal), disponibilizados de *DGTerritório* (2017) para o seu tratamento nos Sistemas de Informação Geográfica (SIG).



Dados de superfície em Ha

Figura 4.2 – Gráfico de barras representativo da superfície das diferentes freguesias que formam parte do concelho de Arouca

Nos últimos anos, a população de Arouca tem sofrido uma diminuição da população até ficar, na atualidade, com um número próximo de 22.000 habitantes. Isto é um dos indicadores da redução do consumo de água, para além das políticas de consumo responsável, baseadas na Diretiva Quadro da água, e a melhora dos sistemas da rede de abastecimento. Esta evolução negativa da população evidencia-se na Figura 4.3, para dados disponíveis até 2011.

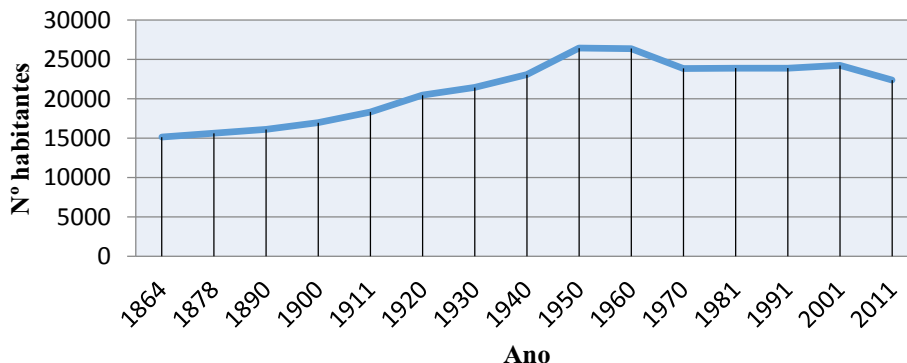


Figura 4.3 – Evolução da população no concelho de Arouca.

O concelho de Arouca apresenta uns preços pela utilização de água elevados: 246,30 € para 120 m<sup>3</sup> anuais. Daí a necessidade de uma otimização energética na operação dos sistemas elevatórios. A Tabela 4.1 apresenta o montante de preços anuais para um consumo de 120 m<sup>3</sup> e 180 m<sup>3</sup> em Arouca, tendo em conta os preços anuais de saneamento e RSU (resíduos sólidos urbanos), abastecimento e saneamento. (Deco Proteste, 2016)

Tabela 4.1 – Simulação de consumos para 120 e 180 m<sup>3</sup> para o município de Arouca

Volume anual (m <sup>3</sup> )	Abastecimento de água	Saneamento	Resíduos sólidos urbanos	Total
	Preço (€)			
120	129,81	74,49	42,00	246,30
180	185,82	106,73	52,50	345,04

O tarifário no município de Arouca tem sido acrescentado desde o ano 2014. O único preço que se mantém é o correspondente aos RSU. Na Figura 4.4 apresenta-se um gráfico com a evolução dos preços da água em Arouca (ERSAR, 2016).

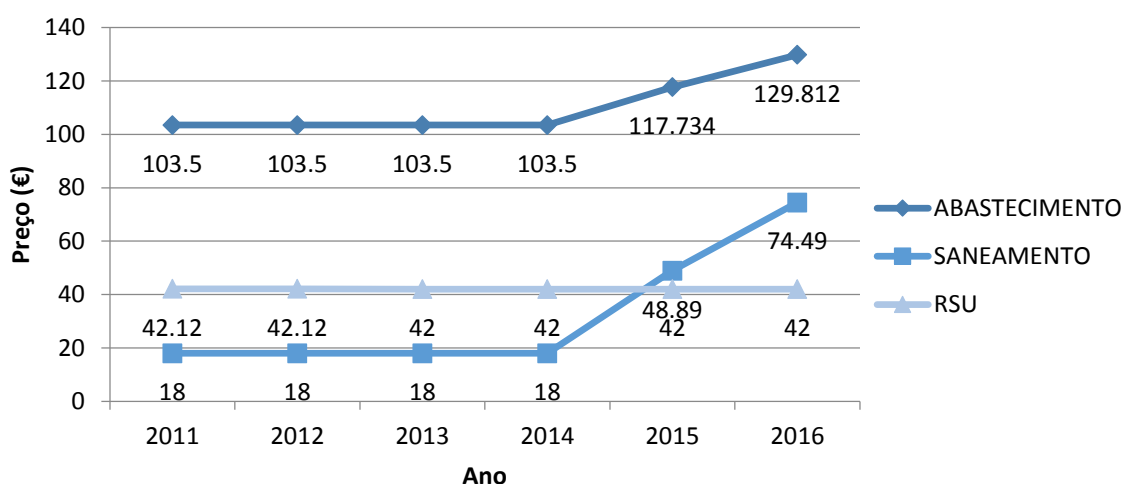


Figura 4.4 – Evolução do tarifário de água em Arouca

As empresas que gerem essas diferentes operações da água em Arouca, são:

- **Entidade de Abastecimento de água (AA):** Águas do Douro e Paiva, S.A.
- **Entidade de Saneamento (SAN):** SIMDOURO, S.A.
- **Entidade de Resíduos sólidos urbanos (RSU):** Câmara Municipal (CM) de Arouca.

No concelho de Arouca, Águas do Douro e Paiva é a encarregada do abastecimento, e na Figura 4.5 apresenta-se o mapa da sua rede de abastecimento.

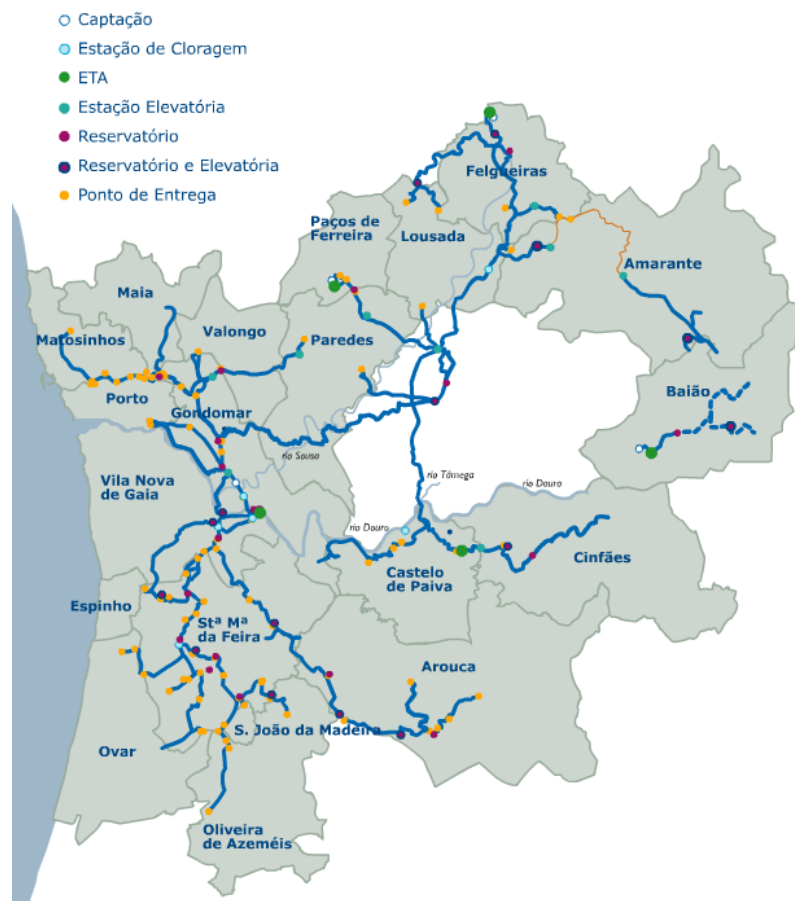


Figura 4.5 – Mapa da rede de abastecimento da Águas do Douro e Paiva , adaptado de Águas de Douro e Paiva S.A., (2017).

Como pode observar-se, no concelho de Arouca há:

- 6 pontos de entrega (PE)
- 2 reservatórios
- 2 reservatórios + estação elevatória

#### 4.1.1 Dados disponibilizados

Os dados de consumos disponibilizados são os correspondentes aos pontos de entrega. Estes pontos de entrega são: Albelheira, Provizende, Souto Redondo-Arouca, Ameixeira, Forcada e Moldes.

Por outra parte, os consumos que serão analisados no concelho de Arouca correspondem aos seguintes períodos temporais:

- Consumos diários do período 2007-2017.
- Consumos num intervalo de meia hora do período 2015-2016.

Estes dados de consumos foram obtidos da base de dados da AdDP disponível online.

Por outro lado, os dados relativos à temperatura e precipitação, foram obtidos do(SNIRH :: *Sistema Nacional de Informação de Recursos Hídricos*, 2017). Devido à falta de grande parte dos dados no concelho de Arouca, serão utilizados os dados correspondentes à estação meteorológica mais próxima que conta com toda a informação necessária (Castelo de Burgães), tal como apresentada na Figura 4.6.

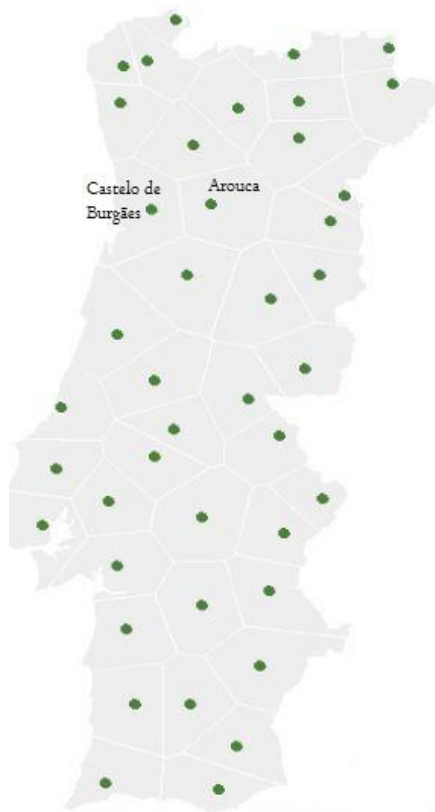


Figura 4.6 – Mapa localização da estação meteorológica de Castelo de Burgães, adaptado do SNIRH : *Sistema Nacional de Informação de Recursos Hídricos (2017)*.

Os dados de temperatura e precipitação diárias podem ser introduzidos junto aos consumos diários de forma direta. Pelo contrário, os dados horários devem ser transformados em dados num intervalo de meia hora para serem coerentes com os dados de consumos.

## 4.2 SELEÇÃO E TRATAMENTO DE INFORMAÇÃO

Apesar da capacidade dos PMC para aproximar qualquer função contínua, o processo de preparação dos dados é crucial para o sucesso da sua aplicação. Por esta razão, a primeira fase da metodologia dedica-se a uma análise pormenorizada dos dados disponíveis.

Assim, deverá ser analisada a adequação das RNA ao problema, em função dos dados disponíveis. Caso as RNA tenham vantagens sobre outros métodos, há de escolher-se o PMC com a arquitetura que melhores resultados produza. Não é aconselhável uma abordagem baseada na experimentação de diversas redes sem uma análise prévia.

Por outro lado, a escolha das variáveis mais importantes para descrever o problema deve tomar em consideração que, se o número de variáveis crescer e as suas relações se tornarem mais complexas, a RNA precisará de uma maior quantidade de dados para ser eficientemente treinada.

#### 4.2.1 Escolha inicial das variáveis

A primeira tarefa consiste na escolha das variáveis que serão consideradas no treino. Isto constitui uma fase muito importante, pois quanto maior seja a relação entre as variáveis, melhor serão os resultados, e pelo contrário, umas variáveis pouco correlacionadas produzirão resultados péssimos. Esta etapa é particular para cada tipo de problema, pois cada variável de estudo é afetada por diferentes fatores em função da sua natureza. Os principais fatores que influenciam o consumo de água (Martínez, 2014) são ilustrados na Figura 4.7.

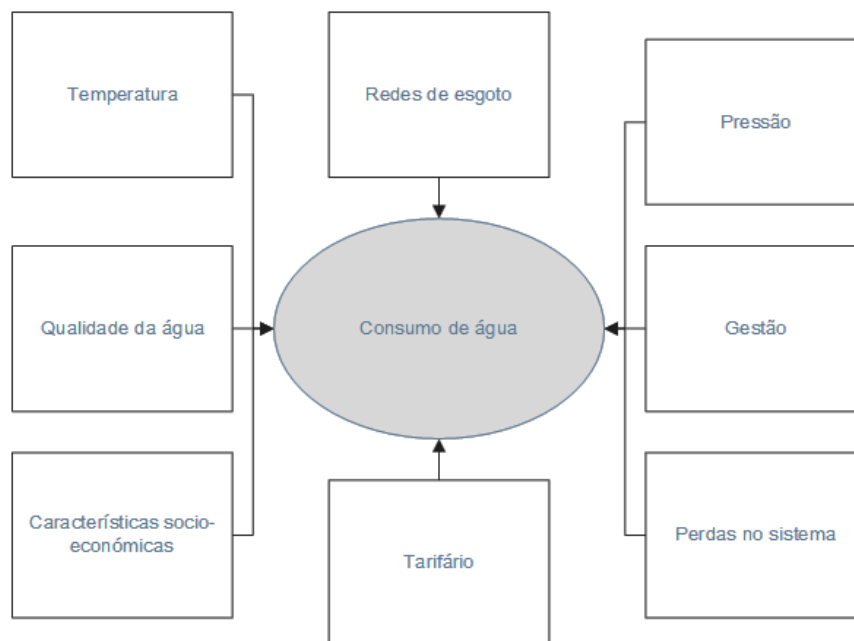


Figura 4.7 – Fatores que podem afetar ao consumo de água.

Alguns de estes fatores são controláveis, por exemplo a gestão e administração dos sistemas de água ou o controlo de perdas no sistema, enquanto outros, como a temperatura, não o são. Estes fatores são descritos de seguida:

- **Temperatura.** Quanto maior for a temperatura, maior o consumo de água. Por exemplo, nos meses de verão a população bebe mais água, toma banho de maneira mais regular e utiliza grandes quantidades de água para enchimento de piscinas, para além da rega de jardins e hortas e combate a incêndios.
- **Qualidade da água.** Por razões lógicas, o consumo de água será maior na medida em que as pessoas têm a segurança de uma boa qualidade na água fornecida pelo sistema.

- **Características sociais e económicas.** O consumo de água também depende muito do nível de educação e nível de rendimento da população. Por esta razão, em cidades maiores, como nas capitais, o consumo de água é maior do que em pequenas cidades ou vilas.
- **Redes de esgoto.** A disponibilidade de uma rede de esgotos aumenta grandemente o consumo de água. Nos casos extremos, o volume de água consumida pode variar de 300 L/hab.dia para grandes cidades até 40 l/hab.dia para cidades sem serviços de esgoto. Em Portugal a inexistência de redes de esgoto é mais frequente em habitações isoladas e/ou rurais nas que o abastecimento de água é feito mediante poços de água alimentados por aquíferos, nas quais as águas residuais são evacuadas através de fossas sépticas.
- **Pressão na rede de distribuição de água.** Se existir uma alta pressão na rede, haverá maior desperdício no consumo doméstico quando abrir as torneiras e outros elementos. Do mesmo modo, pode haver um maior número de ruturas nas tubagens nos domicílios ou na própria rede de distribuição, aumentando assim o volume de água.
- **Gestão.** Uma gestão eficaz controlará o consumo de água e de resíduos, mediante a redução de perdas e o controlo de ligações ilegais. Para realizar o trabalho são necessários equipamentos especializados, tais como amplificadores de som eletrônicos ou traçadores radioativos fracos e de curta duração, os quais têm um preço elevado e não podem ser adquiridos por todos os municípios.
- **Tarifário.** Os preços no consumo de água causam um impacto psicológico sobre os utilizadores, de modo que o consumo diminui se os preços aumentarem. É preciso implementar um sistema tarifário equilibrado para racionalizar o consumo de água.
- **Perdas no sistema.** Evidentemente, quanto maior seja a quantidade de perdas numa rede de abastecimento, maior será o volume de água que é preciso fornecer, mesmo que não seja consumido pelos utilizadores da rede ao perder-se pelo caminho.

Devido à grande quantidade de variáveis que influenciam o consumo de água e à falta de disponibilidade destes dados ao serem restritos à própria empresa (AdDP), apenas são tidos em conta os seguintes fatores:

- Temperatura média (diária e horária, dependendo do caso).
- Precipitação (diária e horária, mesmo que não forme parte da listagem acima, poderá ter alguma relação com o consumo).

Foram considerados os valores médios de temperatura devido à falta de informação sobre temperaturas máximas diárias nas diferentes estações climáticas do SNIRH.

A seguir, deve ser analisado se, para o caso específico do concelho de Arouca, as variáveis escolhidas estão mesmo relacionadas com o consumo nessa população. Para este efeito, é realizada uma análise da correlação FAC e FAP entre essas variáveis, além de uma análise estatística.

#### 4.2.2 Análise multivariável das variáveis escolhidas

Uma vez pré-selecionadas as variáveis (ou *features*, em termos de rede neuronal), é preciso analisá-las. Para tal fim, a análise vai ser feita em dois níveis: diário (período 2007-2017 e horário 2015-2016). A primeira fase para selecionar as variáveis e determinar a sua relação com o consumo de água. Para tal, é utilizado o coeficiente de correlação de Pearson (US, 2013), o qual mede o grau da correlação (e a direção dessa correlação - se positiva ou negativa) entre duas variáveis. No seu cálculo intervêm a covariância entre ambas as variáveis e a variância de cada uma de elas. Assume apenas valores entre -1 e 1 .

$$\rho = \frac{cov(X,Y)}{\sqrt{var(X)var(Y)}} \tag{4.1}$$

- $\rho = 1$  → Significa uma correlação perfeita positiva entre as variáveis.
- $\rho = -1$  → Implica uma correlação negativa perfeita entre as duas variáveis – (Se uma aumenta, a outra sempre diminui).
- $\rho = 0$  → Significa que as duas variáveis não dependem linearmente uma da outra. No entanto, pode existir uma dependência não linear.

Na Figura 4.8 mostra-se o gráfico de correlação entre as diferentes variáveis (análise multivariável) para análise diária. Na Tabela 4.2 recolhem-se os valores de  $\rho$  para cada caso.

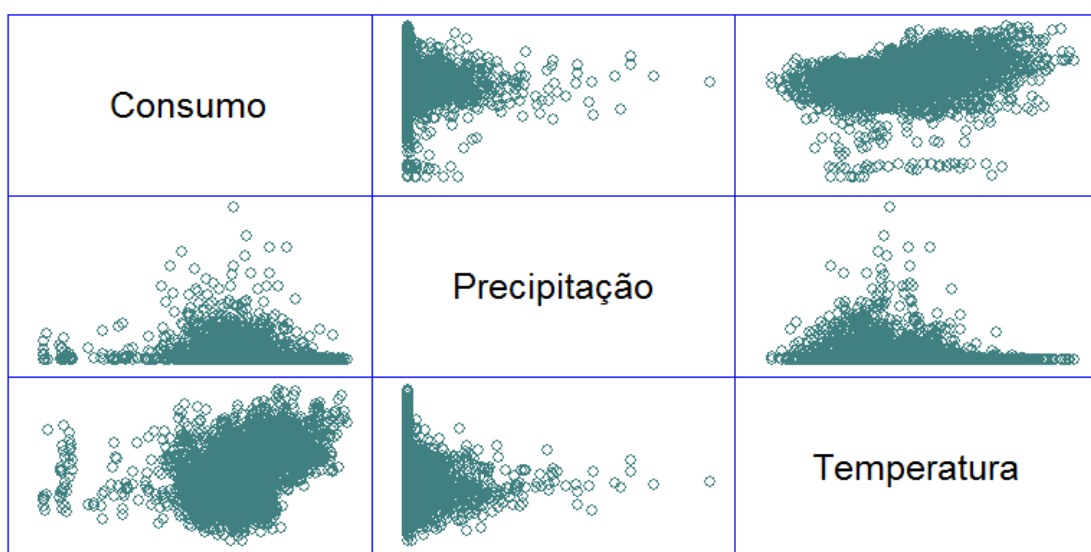


Figura 4.8 – Análise multivariável em relação aos consumos diários.



Tabela 4.2 – Valores do coeficiente  $\rho$  entre as distintas variáveis diárias.

	Consumo	Precipitação	Temperatura
Consumo	1	-0,01443	0,4005
Precipitação	-0,01443	1	-0,1764
Temperatura	0,4005	-0,1764	1

A partir do anterior, podem ser feitas as interpretações a seguir:

- **Relação consumo-precipitação.** A relação linear que existe entre o consumo de água e a precipitação é muito fraca (existe, mas é praticamente insignificante, ao ter um valor de  $\rho = 0.01443$ , muito próximo de zero). O sinal negativo implica que a correlação é negativa por tanto, quer dizer que, caso chova, teoricamente o consumo deve diminuir fracamente. O gráfico praticamente apresenta uma forma vertical (ou horizontal, se mudar os eixos), em concordância com o anteriormente dito.
- **Relação precipitação-temperatura.** A relação entre a precipitação e a temperatura é relativamente fraca em quanto é próxima de zero, mas muito mais significativa do que a relação consumo-precipitação ( $\rho = -0.1764$ ). Seja como for, esta correlação não é muito interessante, pois não intervém o consumo.
- **Relação consumo-temperatura.** É a maior das correlações existentes, com um valor de  $\rho = 0.4005$ , ainda que não suponha uma correlação muito forte. O gráfico mostra uma distribuição dos pontos com certa tendência diagonal (muito diferente do resto de distribuições, nas que predomina a verticalidade ou a horizontalidade).

A partir desta análise, chega-se à conclusão de que pode resultar mais proveitoso a retirada da precipitação da previsão dos consumos, enquanto a sua consideração pode tornar-se negativa (importa lembrar-se que, se o número de variáveis crescer e as suas relações se tornarem mais complexas, a RNA precisará de mais dados para ser devidamente treinada, ao não mostrar uma correlação forte com os consumos). Por outra parte, a temperatura põe de manifesto uma correlação importante e, por tanto, deve ser, sob todos os efeitos, considerada.

Para o caso dos consumos horários, esta avaliação é feita da mesma forma. Na Figura 4.9 apresentam-se os gráficos de correlação de esta análise, sintetizados na Tabela 4.3.

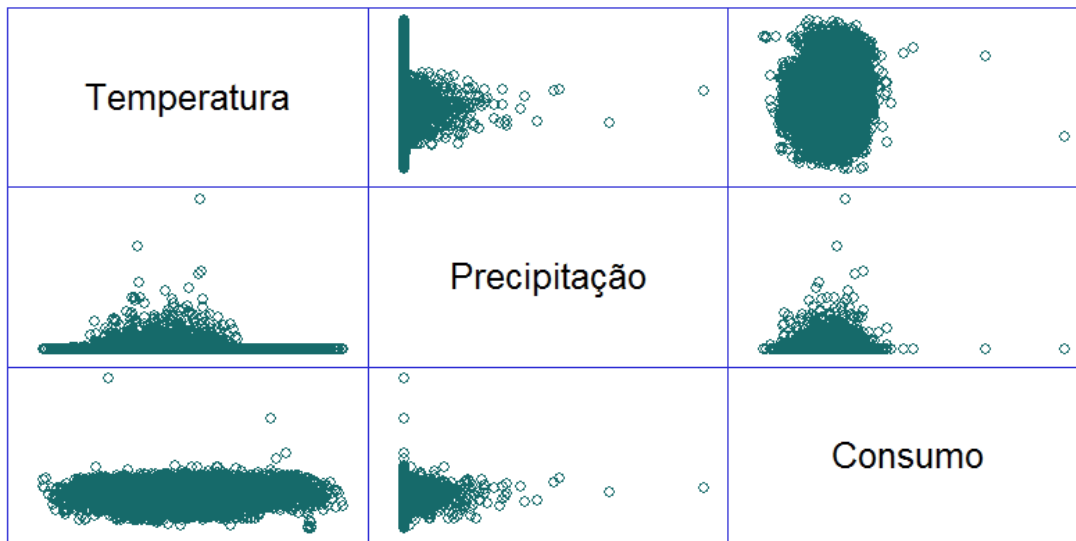


Figura 4.9 – Análise multivariável em relação aos consumos horários.

Tabela 4.3 – Valores do coeficiente  $\rho$  entre as distintas variáveis horárias.

	Consumo	Precipitação	Temperatura
Consumo	1	-0,0099	0,0801
Precipitação	-0,0099	1	-0,0627
Temperatura	0,0801	-0,0627	1

Procedendo de forma semelhante ao caso dos consumos diários, pode concluir-se a partir desta análise, que não convém a inclusão da temperatura e a precipitação na previsão dos consumos com RNA, enquanto que as suas correlações são mesmo fracas, facto refletido nos gráficos (nos quais a distribuição da correlação é praticamente vertical ou horizontal para todos os casos).

Uma vez feita a escolha definitiva das variáveis, é preciso analisá-las para caracterizá-las tanto para o seu conhecimento e a posterior comparação com os resultados como para o tratamento dos dados.

No Anexo I recolhem-se todos os parâmetros estatísticos obtidos em forma de tabelas e gráficos, enquanto que nesta secção apenas são mostrados os mais importantes e o desenvolvimento da análise a partir desses dados.

### Análise diária dos consumos

- Consumo de água

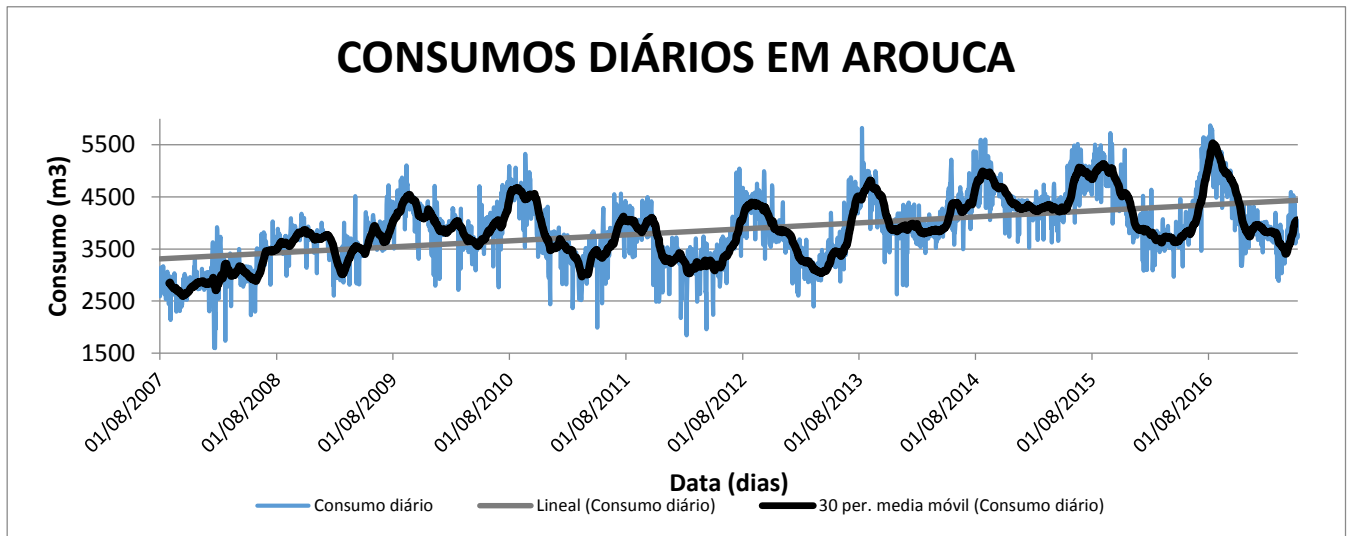


Figura 4.10 – Série temporal diária do consumo de água do período 2012-2014

A partir do gráfico da série de consumos diários, apresentado na Figura 4.10, podem estabelecer-se as seguintes considerações:

- Existe uma tendência linear crescente. Isto é, há um aumento geral no consumo de água médio com o passo dos anos. Esse incremento linear vem dado em função do dia ao longo da série:

$$V = 0,3147 \cdot \text{dia} + 3312 \quad (4.2)$$

- Existem grandes aumentos no consumo nos meses de verão, o qual é o primeiro indicador de uma possível relação entre o consumo de água e a temperatura. Aliás, os valores são mais dispersos nesses meses, nos quais acontecem picos de consumo pontuais que introduzem os maiores desvios padrões (estão mais longe da média).
- Nos meses de inverno os consumos são muito mais regulares, pois não apresentam tanta variação respeito da média.
- Existe uma clara tendência cíclica ao longo do ano. Isto é, há presença de sazonalidade.
- A série possui muito ruído, por tanto, isto vai influir negativamente nos resultados, através da introdução de muita dispersão, o que fará com que o EQM vá crescer muito tendo essas grandes diferenças entre os valores reais e as previsões.

A partir da análise estatística, são obtidas algumas conclusões:

- A partir do ano 2013 produz-se um aumento considerável na média dos consumos por qualquer razão (por exemplo, perdas no sistema ou a criação de nova indústria).

Adicionalmente, são estudadas as variações do consumo em cada mês, para cada um dos anos. Desta informação conclui-se:

- Como já foi refletido no gráfico da série temporal, nos meses de verão os consumos crescem, enquanto que no inverno são menores. Agosto é o mês no qual sempre é verificado o maior consumo (com uma média de  $4566,68 \text{ m}^3$ ), seguido pelo mês de setembro ( $4472,49 \text{ m}^3$ ) e julho ( $4347,32 \text{ m}^3$ ).
- Ao final da primavera e ao começo do outono, experimenta-se um aumento (tanto maior quando mais próximo do verão) e uma descida (com os valores mais elevados nos meses próximos ao verão, como outubro, e vice-versa). A média dos consumos nestes casos oscila entre os  $3500\text{-}4200 \text{ m}^3$ .
- Nos meses nos quais são verificados os menores consumos são janeiro, fevereiro, março e abril. Neste período, dificilmente são ultrapassados os  $3500 \text{ m}^3$ .

De novo são analisados os consumos de água, mas desta vez focando a análise no período semanal. Da sua análise concluem-se os seguintes aspetos:

- O dia de maior consumo da semana é atingido no fim de semana, nomeadamente no sábado.
- O menor consumo é atingido na segunda-feira.
- De segunda-feira a sexta-feira, os consumos aumentam levemente cada dia.

#### Análise horária dos consumos

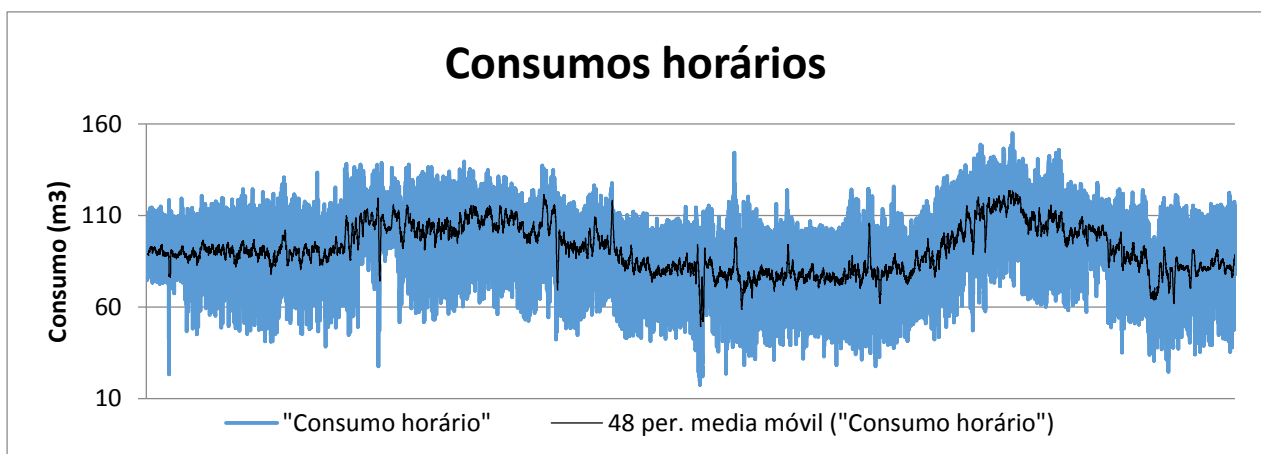


Figura 4.11 – Série temporal horária do consumo de água do período 2015-2016

A partir do gráfico dos consumos horários, apresentado na Figura 4.11, podem ser feitas as seguintes apreciações:

- A tendência permanece pouco variável ao longo do inverno.

- A partir da primavera, em especial do mês de abril, o consumo volta a crescer (nomeadamente no mês de abril).
- De maneira geral, os consumos médios são da ordem de  $83 \text{ m}^3$  e separados dessa média  $19 \text{ m}^3$  por acima e por abaixo.

Novamente são analisados os consumos de água, mas desta vez no período semanal. Da sua análise concluem-se os seguintes aspetos:

- Os maiores consumos são claramente produzidos no fim de semana. O dia de maior consumo semanal é o sábado, com uma média de  $86,74 \text{ m}^3$ .
- O dia no qual é verificado o menor consumo médio é segunda-feira ( $81,43 \text{ m}^3$ ).
- Desde segunda-feira a sexta-feira, se produz um aumento (fraco) continuado dos consumos médios.
- O padrão de consumos horário tem carácter cíclico (sazonal). Isto é, ao longo dum dia, existem variações que são verificadas (diariamente, em este caso).
- O padrão de consumos não é igual nos dias da semana e no fim de semana, devido aos hábitos de vida diferentes em ambos períodos.
  - **Dias da semana.** Ao início do dia (00h00min) o consumo é próximo da média ( $83 \text{ m}^3$ ). Ao longo da noite desce até atingir o mínimo absoluto do dia (aprox.  $65 \text{ m}^3$ ), por volta das 4.30 da manhã. Com o início das atividades humanas, o consumo cresce até atingir o máximo absoluto (aprox.  $98 \text{ m}^3$ ) entre o meio-dia e a tarde, em concordância com a hora da refeição (almoço) em Portugal. A partir de aqui o consumo de água diminui até às 18.00-18.30 da tarde, quando é atingido o mínimo local (aproximadamente, corresponde com o consumo médio horário). A maior parte da população acaba a sua jornada de trabalho nesse intervalo e dirigem-se aos seus domicílios. O máximo local (aprox.  $95 \text{ m}^3$ ) é atingido à hora da refeição (jantar) em Portugal, por volta das 20.00 horas. A partir de aqui, o consumo diminui.
  - **Fim de semana.** Não há grandes variações ao longo do dia; este começa de maneira semelhante ao caso do dia da semana, mas neste caso se produz um aumento acentuado a partir de 5.30 da manhã (aproximadamente  $65 \text{ m}^3$ ), o qual se estabiliza aproximadamente às 11.30 da manhã ( $98 \text{ m}^3$ ), e apresenta flutuações leves ao longo da tarde, permanecendo quase invariável, mas geralmente com um pequeno aumento atingindo o máximo absoluto (aproximadamente  $100 \text{ m}^3$ ) por volta das 20 horas (hora da refeição em Portugal). Logo, os consumos diminuem até ficar perto da média.

Análise diária da temperatura

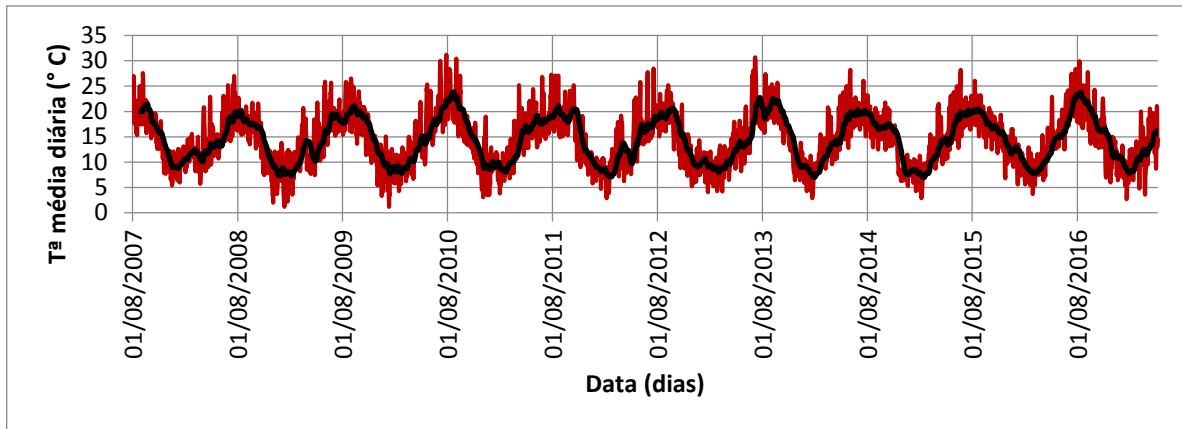


Figura 4.12 – Temperatura diária média do período 2007-2017.

Da série completa de temperaturas apresentada na Figura 4.12, podem deduzir-se as seguintes particularidades:

- A temperatura média diária é um processo completamente sazonal, pois varia em função das estações do ano.
- A temperatura média diária aumenta ano após ano. Linearmente, esta relação pode ser expressa como:

$$T^a(^{\circ}C) = 0,0004 \cdot dia + 13,801 \tag{4.3}$$

- A temperatura média nos meses de verão é, aproximadamente, de 20 °C, enquanto que nos meses mais frios (inverno) as temperaturas médias são próximas de 9 °C.

No que toca aos consumos mensais, as conclusões da análise estatística são as seguintes:

- A temperatura máxima é verificada nos meses de verão (julho ou agosto, segundo o ano), em consonância com os consumos, e correspondem às formas convexas do gráfico. Os valores máximos são próximos de 20 °C.
- A temperatura mínima habitualmente é verificada nos meses de janeiro e fevereiro. Oscilam perto dos 9 °C. Corresponde às formas côncavas do gráfico.
- A temperatura aumenta nos meses de primavera e diminui nos meses de outono.

# CAPÍTULO 5

## METODOLOGIA APLICADA

### 5.1 INTRODUÇÃO

A previsão de consumos de água não é uma tarefa fácil. A maior parte das dificuldades foram encontradas no âmbito do conjunto dos dados de base. A primeira complicação é a disponibilidade dos dados, pois estes são muito limitados e não se dispõe de uma continuidade total no intervalo de tempo estudado. O segundo obstáculo tem sido a abundante quantidade de falhas e erros nesses dados e a necessidade da sua correção e adequação à realidade.

A seguinte metodologia proposta tem intuito de ultrapassar as dificuldades apontadas mediante a aplicação das RNA e o modelo ARIMA para o caso dos consumos de água, mas também pode ser aplicada para problemas análogos que tenham por base uma série temporal e cujo objetivo seja a obtenção de um padrão de consumo, isto é, a sua evolução no tempo, como por exemplo a previsão do consumo elétrico, de modelos econométricos, etc. Para conhecer mais sobre a previsão de consumos elétricos, recomenda-se a leitura de (Ariza Ramírez, 2013) e (Fernández Jiménez, 2016).

#### 5.1.1 Considerações prévias

Escolher a rede mais adequada para o problema em questão pode tornar-se um desafio. Face a esta adversidade, alguns utilizadores consideram um amplo grupo de redes e algoritmos de treino e realizam uma comparação dos mesmos, escolhendo o melhor. Outra forma frequente de lidar com este problema é basear-se em trabalhos precedentes.

Embora a segunda abordagem seja mais utilizada, os resultados obtidos por esse método não são específicos para o problema em questão. Por vezes, diferentes autores propõem a utilização de medidas distintas e eventualmente contraditórias, por causa do anterior.

Devido a este motivo, considera-se que é frutífero levar a cabo uma comparação de soluções fundamentada, mesmo que demore certo tempo.

Os modelos que serão criados para prever os consumos são aqueles que foram definidos no capítulo 2 (Revisão Bibliográfica). Recomenda-se a análise das subsecções correspondentes a cada modelo.

- Percepção multicamada PMC.
- Rede NARx.
- Rede LSTM.

A definição dos três modelos é diferente, pois cada um deles tem as suas particularidades. Os modelos de PMC e rede NARx foram definidos na toolbox de Matlab, a qual permite criar as redes mediante o ajuste de vários parâmetros.

1. PMC não recorrente:

- Número de camadas ocultas.
- Número de nós na camada oculta.
- Dimensão dos subconjuntos de dados (treino, validação e teste).
- Algoritmo de treino.

2. Rede NARX:

- Número de nós na camada oculta (única).
- Número de atrasos ou *delays*.
- Dimensão dos subconjuntos de dados (treino, validação e teste).
- Algoritmo de treino.

Por outro lado, a rede LSTM foi definida na linguagem Python, como tentativa de ver o comportamento e a resposta deste modelo em relação ao resto, e com a intenção de criar uma ferramenta completa capaz de integrar algoritmos de correção, diferenciação, treino, previsão e validação e análise da série temporal dos consumos, com a capacidade de apresentar gráficos interativos.

3. As possibilidades que oferece o *Keras* enquanto a definição da rede LSTM são:

- Número de *delays* ou atrasos.
- Número de neurónios por camada.
- Número de camadas ocultas.
- Tipo de camada oculta (podem ser misturadas camadas LSTM com camadas tipo PMC ou outro tipo de camadas, mas não serão utilizadas no âmbito de aproximação de funções).
- Número de épocas de treino.
- Possibilidade de adicionar métodos de regulação (para evitar a presença de *overfitting*).



- Algoritmo de treino.

Portanto, o procedimento para a definição da rede LSTM é muito mais complexo do que nos outros modelos, estes últimos criados basicamente mediante a variação dos parâmetros anteriormente definidos. Por esta razão, é apresentado um fluxograma do procedimento mais complexo, relativo à rede LSTM na Figura 5.1.

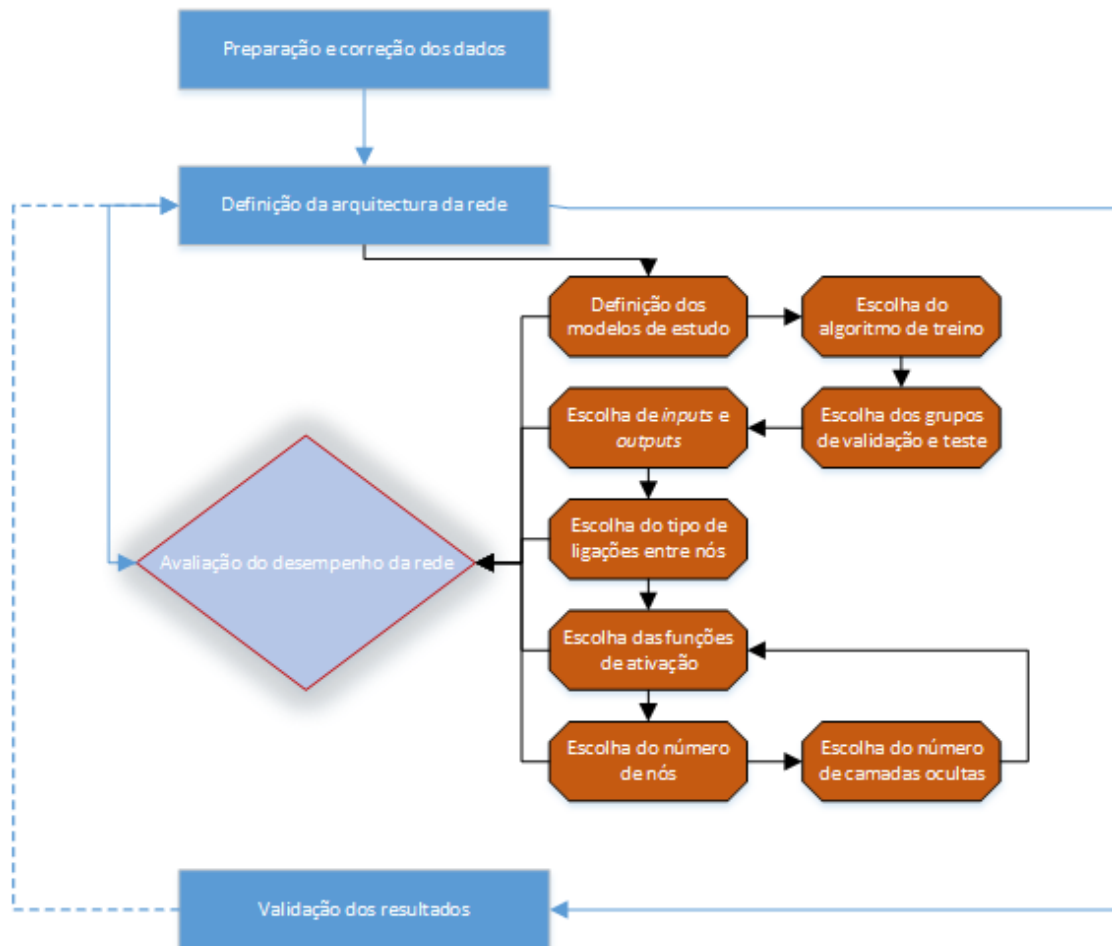


Figura 5.1 – Fluxograma da metodologia proposta.

### 5.1.2 Preparação dos dados

Os dados que serão utilizados numa RNA devem ser preparados antes de serem introduzidos nela. Em seguida, são descritos os passos de atuação seguidos no processo de preparação de dados.

1. Correção de falhas e dados presumivelmente errados.
2. Diferenciação da série.
3. Conversão do problema atual em problema supervisionado (só no caso do modelo LSTM).
4. Padronização dos dados para o seu futuro treino

A distribuição dos dados é armazenada num arquivo com formato \*.csv (valores separados por vírgulas ou *comma separated values*).

Os dados de consumos diários disponibilizados são armazenados diariamente em folhas de cálculo separadas. Por tanto, é preciso uni-las todas manualmente e, uma vez arrumados, unir os consumos de todos os PE para ter o total de consumos horários no concelho de Arouca. Este passo é realizado através de programação na linguagem *Visual Basic (VB)*, aplicada às folhas de cálculo. A partir destes dados, calculam-se os consumos diários.

Uma vez selecionados e estruturados, é preciso adaptar os dados de entrada aos consumos. Isto é, caso os consumos sejam diários, não é necessário adaptá-los, pois os dados de temperatura disponíveis também são diários. Pelo contrário, os consumos horários disponíveis estão separados por um intervalo temporal de meia hora, enquanto os dados de temperatura obtidos do *Sistema Nacional de Informação de Recursos Hídricos*, são representados cada hora. Por tanto, para que o intervalo de dados corresponda ao intervalo de observações de consumo de água, é feita a média de duas horas consecutivas, de maneira a obter 24 valores adicionais correspondentes às meias horas, além dos 24 disponíveis na série, de forma a ter 48 dados diários.

Seguidamente, é necessário identificar falhas na leitura de dados e corrigi-los. Esta tarefa torna-se árdua, como no caso de dias nos que apenas se têm dados até uma determinada hora do dia e o resto é composto de valores nulos ou também, em casos extremos, dias em que não se tem qualquer informação. Nestes casos, a solução levada a cabo foi o preenchimento de dados em falta mediante um programa desenvolvido na linguagem VB, e preencher esses vazios com dados similares aos dias prévios e posteriores.

Uma vez preparados os arquivos de dados diários e horários, estes são, ainda, corrigidos com um algoritmo desenvolvido em linguagem Python que segue estas fases:

- 1) Identificar *outliers* ou valores anómalos na série temporal, considerando como tal aqueles que estejam a uma distância da média móvel maior de duas vezes o desvio padrão da série. Desta forma, ao não ser a série estacionária da sua média, a remoção dos valores anómalos também não é, devido à utilização da média móvel. O código mostra ao utilizador uma imagem com a série original, com a representação da média móvel e a identificação dos potenciais *outliers*. A média móvel ideal é aquela que é representativa da tendência e também apresenta pouco ruído. Após vários testes, considera-se apropriada a utilização de uma média móvel de 30 períodos (equivalente a um mês) para o caso da série diária e de 48 (equivalente a um dia) para o caso da série horária. A identificação de valores anómalos para o caso dos consumos diários e horários é representada na Figura 5.2 e na Figura 5.3, respetivamente.

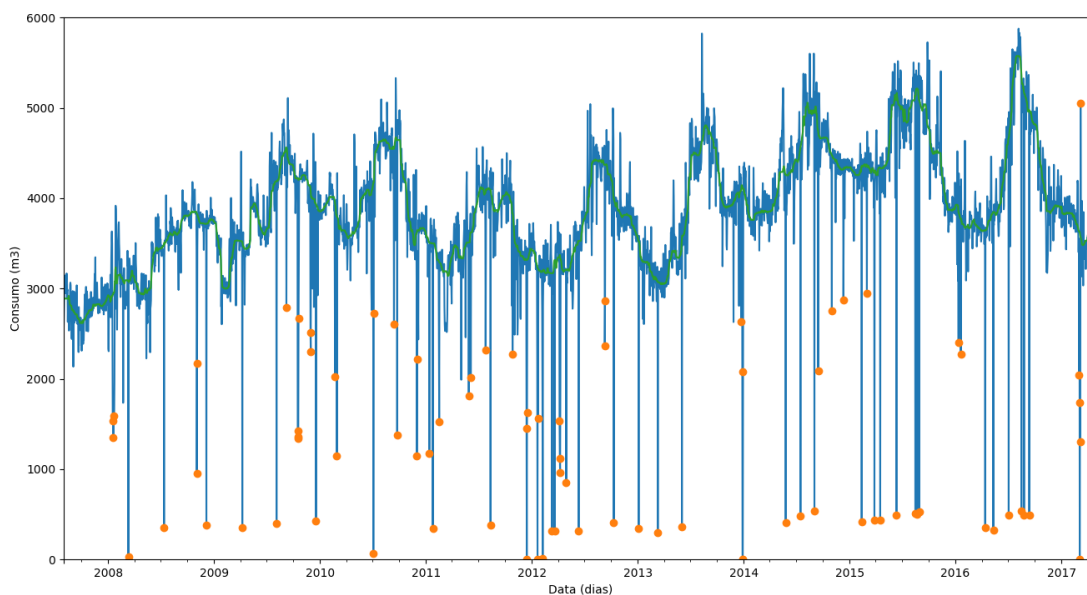


Figura 5.2 – Identificação de outliers na série diária com média móvel de período 30 dias.

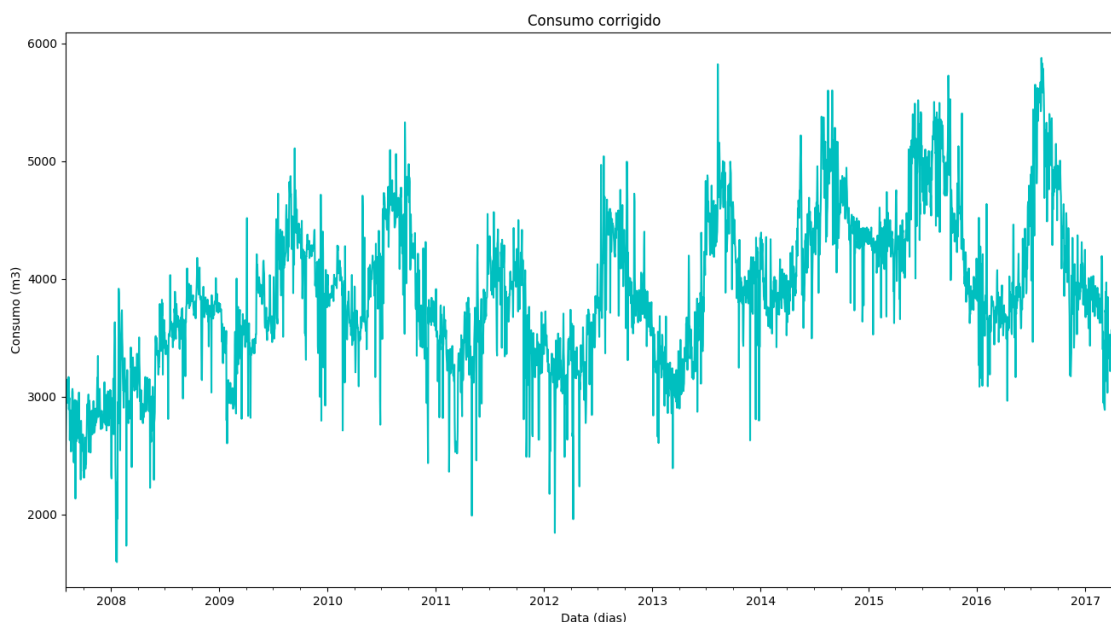


Figura 5.3 – Série diária corrigida com média móvel de período 30 dias.

- 2) No caso de ser *outlier*, o valor é corrigido como a diferença entre a média móvel nesse ponto e duas vezes o desvio padrão, no caso dos dados serem diários. Caso sejam horários, o valor anômalo substitui-se pelo valor imediatamente anterior. A correção da série temporal diária é apresentada na Figura 5.4 enquanto a correção da série temporal horária é apresentada na Figura 5.5.

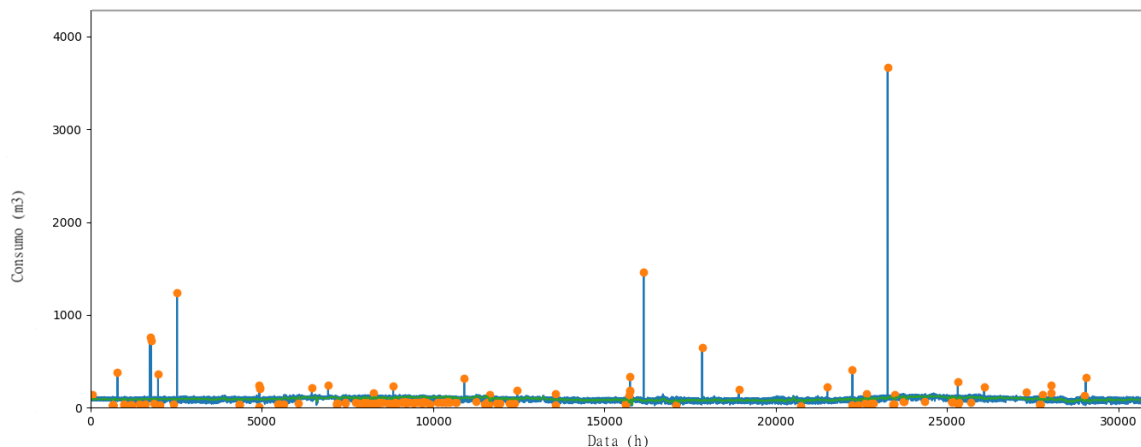


Figura 5.4 – Identificação de outliers na série horária com média móvel de período 24 horas.

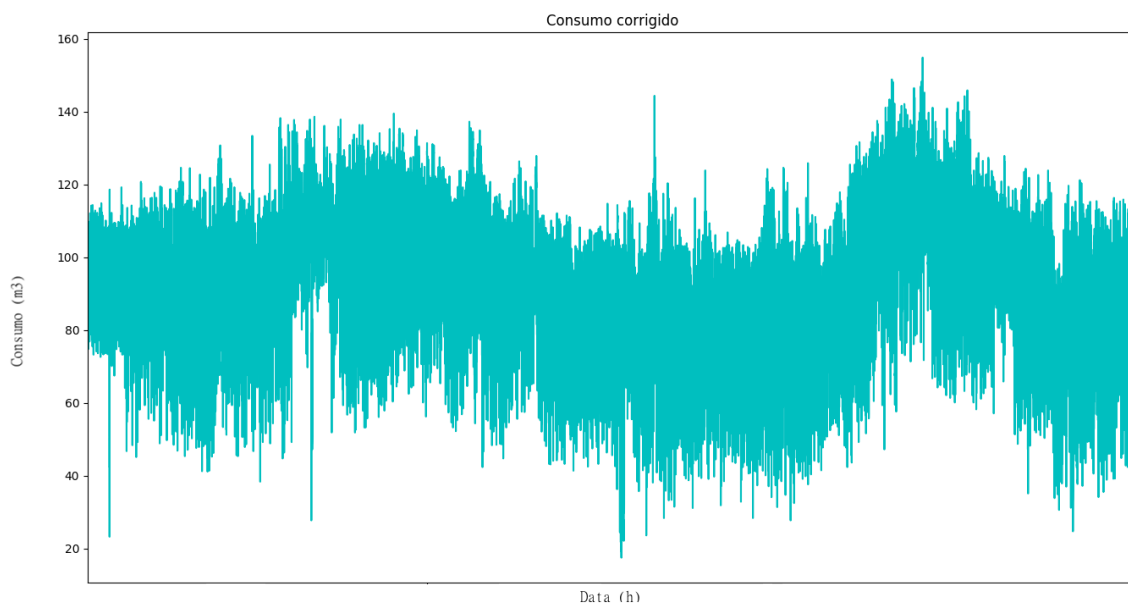


Figura 5.5 – Série horária corrigida com média móvel de período 24 horas.

Para conseguir uma maior qualidade nos resultados, no caso da rede LSTM, considera-se útil realizar uma diferenciação da série temporal, de maneira que seja estacionária, através da diferença  $x_t - x_{t-1}$  ao longo da mesma. O resultado da série diária diferenciada é ilustrado na Figura 5.6, enquanto a série horária diferenciada é apresentada na Figura 5.7.

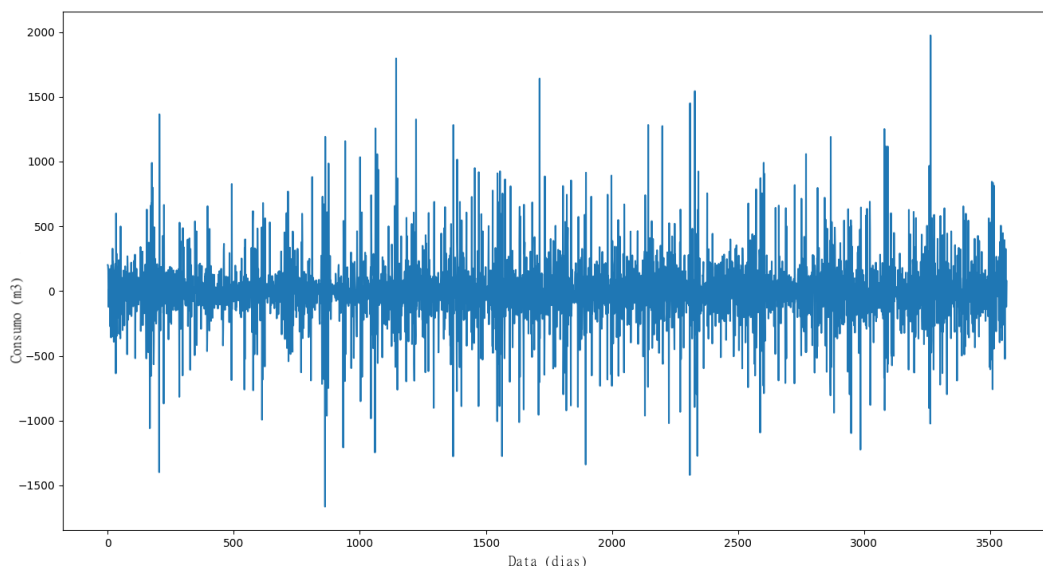


Figura 5.6 – Série diária diferenciada após correção.

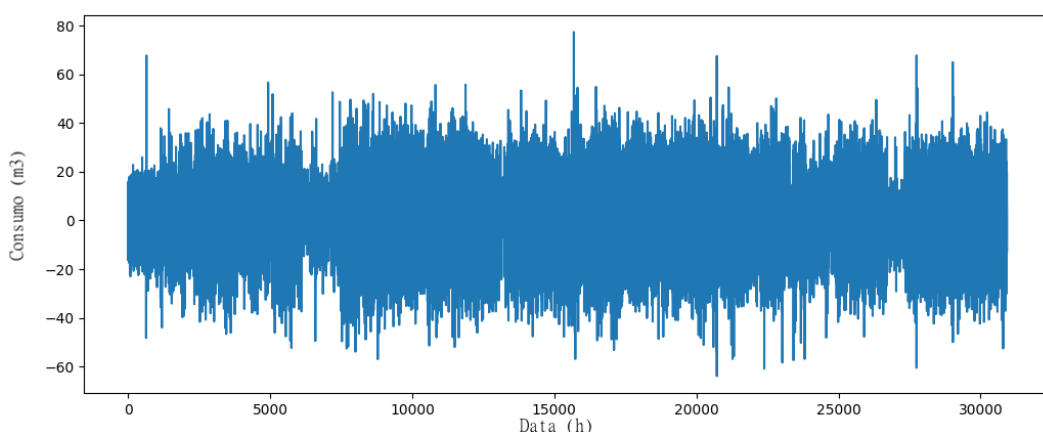


Figura 5.7 – Série horária diferenciada após correção.

Só no caso da rede LSTM, é necessário transformar a série temporal num problema supervisionado (no resto de modelos já têm implementado um algoritmo para converter a série temporal num problema supervisionado).

Uma série temporal é uma sequência de números ordenados por um índice de tempo. Um problema de aprendizagem supervisionada é composto por padrões de entrada ( $x$ ) e padrões de saída ( $Y$ ), de maneira que um algoritmo possa aprender como prever os das saídas a partir das entradas.

Tecnicamente, na previsão de séries temporais são utilizados o tempo atual ( $t$ ) e tempos futuros ( $t + 1, t + n$ ) como tempos de previsão, e as observações passadas ( $t - 1, t - n$ ) são utilizadas para realizar as previsões. Isto não só permite a previsão clássica  $X \rightarrow Y$ , mas também  $X \rightarrow Y$  onde ambos são sequências.

Dito de outra forma, é uma maneira de indicar aos algoritmos do *Tensorflow* qual é o valor (ou valores) prévio ( $t - 1, t - 2, \dots, t - n$ ), qual é o atual ( $t$ ) e quais as sequências ou valores a utilizar como targets ( $t + 1, t + 2, \dots, t + n$ ).

Para transformar uma série temporal numa série com estrutura supervisionada é necessário estruturar os dados em tantas colunas como:

$$ncol = nvar + nlag (t - 1, t - 2 \dots) \times nvar + nseq (t + 1, t + 2 \dots) \times nvar \quad (5.1)$$

Onde

$ncol$  – é o número de colunas final;

$nvar$  – é o número de variáveis, incluída a variável a prever;

$nseq$  – é o número de sequências ou valores futuros;

$nlag$  – é o número de sequências ou valores prévios.

Portanto:

$$ncol = nvar \times [1 + nlag (t - 1, t - 2 \dots) + nseq (t + 1, t + 2 \dots)] \quad (5.2)$$

Um exemplo de caso multivariável (2 variáveis) transformado em aprendizagem supervisionada é mostrado na Figura 5.8.

var1(t-1)	var2(t-1)	var1(t)	var2(t)	var1(t+1)	var2(t+1)
0.0	90.0	1	91	8.0	98.0
1.0	91.0	8	98	6.0	96.0
8.0	98.0	6	96	8.0	98.0
6.0	96.0	8	98	9.0	99.0
8.0	98.0	9	99	6.0	96.0
9.0	99.0	6	96	7.0	97.0
6.0	96.0	7	97	8.0	98.0
7.0	97.0	8	98	9.0	99.0

Figura 5.8 – Caso multivariante (2 variáveis) transformado a aprendizagem supervisionada.

Para finalizar, estes são importados nas diferentes ferramentas de computação, e são normalizados (portanto, os dados armazenados no arquivo .csv não sofrem qualquer modificação fora dos softwares). A normalização, habitualmente é feita para valores no intervalo [0,1] ou [-1,1], para que o computador possa trabalhar de uma maneira mais eficiente e muito mais rápida, com menor consumo de memória. Há muitas formas de normalizar os valores, por exemplo, seguindo uma distribuição normal. Neste caso é feito da seguinte forma:

$$X_i = \frac{X_i}{X_{máx}} \quad (5.3)$$

Onde:

$X_i$  – é cada valor de um input ou target;

$X_{máx}$  – é o valor máximo desse input ou target.

Cada input é normalizado com o máximo valor cada coluna (para cada *feature*), não para o maior valor de todos os inputs. Noutras palavras; por exemplo, para a *feature* “temperatura”, cada valor de temperatura é normalizado com o máximo valor de temperatura, não com o máximo valor de todos os *inputs*.

## 5.2 PROCEDIMENTO DE MODELAÇÃO DA RNA

### 5.2.1 Escolha do modelo

O procedimento para escolher o modelo ótimo de RNA é o seguinte: diferentes tipos de redes neurais suscetíveis de serem aplicáveis a uma série temporal serão definidos usando o módulo de Matlab, Neuronal Network Toolbox, devido a sua facilidade e rapidez ao definir e treinar uma RNA. Além da vantagem de possuir uma interface gráfica, dispõe do algoritmo de Levenberg–Marquardt, o qual converge muito rapidamente e sem a necessidade de definir o número de épocas (a Neural Network Toolbox utiliza paragem antecipada). Depois que sejam obtidos os resultados para cada modelo de RNA, será escolhido aquele que seja melhor, isto é, aquele que produza o EQM menor. Além do EQM, será tido em conta o custo computacional, pois tempos computacionais muito elevados não são desejáveis, assim como a correlação entre os valores reais (targets) e as saídas (outputs).

Uma vez que o melhor modelo seja encontrado no Neuronal Network Toolbox, será definido o modelo LSTM na linguagem Python e serão comparados os resultados obtidos entre os diferentes modelos, sendo o modelo ótimo aquele que apresente um EQM menor com um tempo computacional aceitável.

#### Consumos horários

1. Rede não recorrente.

Em problemas de ajuste, deseja-se que uma rede neuronal mapeie um conjunto de dados de entradas numéricas –inputs- e um conjunto de objetivos –targets-. Exemplos deste tipo de problema incluem estimativa dos níveis de emissão dos motores com base nas medições de consumo de combustível e velocidade; ou prever o nível de gordura corporal de um paciente com base em medidas corporais.

Uma rede *feed-forward* com uma camada oculta, e funções de ativação sigmoidais e uma camada de saída com uma função de ativação linear, pode ajustar problemas de mapeamento multidimensional arbitrariamente bem, tendo dados consistentes e nós suficientes na sua camada oculta.

Neste caso, o único fator que se pode mudar e faz variar o resultado é o número de nós na camada oculta. Os inputs e os targets são treinados como um conjunto; na camada oculta são aplicados uns pesos sinápticos e é aplicada uma função de transferência (soma dos pesos), e uma função de ativação (sigmoidal). Sobre o resultado anterior voltam a ser aplicados novos pesos sinápticos na camada de saída e novamente uma função de transferência (soma dos pesos) e uma função de ativação linear com vista à redução do EQM.

2. Rede autorecorrente

A previsão é um tipo de filtragem dinâmica, na qual valores passados de uma ou mais séries temporais são usados para prever valores futuros. As redes neuronais dinâmicas, que incluem linhas de atraso encaminhadas, são usadas para filtragem e predição não-lineares.

Os pesos sinápticos são diferentes para os outputs e os inputs, independentes, mas ambos são somados na camada oculta. A função de ativação aplicada na camada oculta é sigmoideal, e uma vez obtida a soma dos pesos e transformada sigmoidealmente, na camada de saída são aplicados novos pesos sinápticos, mas neste caso já não chegam a ela dois pesos independentes por causa do anterior. Novamente, a função de transferência aplicada é a soma dos pesos sinápticos, enquanto que a função de ativação de a camada de saída é linear. Neste caso, os parâmetros que serão modificados são:

O procedimento a seguir é o seguinte:

1. Estabelecer vários números de nós para a camada oculta.
2. Fixar o *delay* ou atraso. Para determinar qual é o *delay* a utilizar, são utilizadas a FAC e a FAP.

No que toca à escolha de *delays*, estes devem estar em concordância com aqueles valores da FAC que sejam representativos. Mesmo assim, devem ser tidos em consideração alguns critérios que limitam a sua escolha, nomeadamente:

- i. Conforme mais longe esteja um valor de outro, a correlação será menor, podendo estar perto dos limites aceitáveis do 95% e não aportando uma melhora à rede.
- ii. Quanto maior seja o número de *lags* utilizados, mais lento será o treino e mais memória será consumida no processo, sem aportar uma grande melhora aos resultados.

Os intervalos de confiança são desenhados como um cone. Por defeito, isso é definido como um intervalo de confiança de 95%, sugerindo que os valores de correlação fora destes limites estão correlacionados. Caso haja uma diminuição progressiva das correlações e, subitamente, um aumento para um determinado *lag*, que é repetido cada  $x$  atrasos, então existe uma clara correlação de tipo sazonal cada  $x$  *lag*.

Na Figura 5.9 ilustra-se a FAP para o caso dos consumos diários, enquanto na Figura 5.10 é apresentada a FAP para o caso dos consumos horários, as quais são representativas da sazonalidade da série: a função de autocorrelação mostra flutuações na correlação ao longo do tempo. Daí podem ser extraídas várias conclusões:

### Série diária



1. Efetivamente, a partir de um número elevado de *lags* (valores passados da série), a autocorrelação diminui, como era suposto. Os valores cada vez ficam mais próximos de zero.
2. A sazonalidade é refletida: aproximadamente, cada 100 dias a correlação deixa de ser relevante. 100 dias corresponde aproximadamente a um período de três meses e meio, e essa é a duração aproximada de uma estação do ano. Portanto, cada trimestre se produz uma mudança sazonal.
3. Cada 7 dias existe um aumento da autocorrelação respeito dos 6 dias anteriores. Consequentemente, é um bom indicador de que existe uma tendência semanal e poderia, a priori, ser utilizado como *delay* no caso dos consumos diários.

A autocorrelação parcial no *lag*  $k$  é a correlação que resulta após a remoção do efeito de quaisquer correlações devido aos desfasamentos em *lags* mais próximos.

A resposta da FAP determina a existência de uma boa correlação do período  $k$  com os  $k - 7$  *lags* ininterruptamente. A partir desse valor, existem valores cuja autocorrelação com o período  $k$  é nulo ou quase nulo (por exemplo,  $k = 8$ ). Existem alguns valores que apresentam correlação, mas nesses casos é fraca e existem valores não representativos entre eles. Desse modo se estaria incorrendo na introdução nas predições de valores que removem parte da autocorrelação entre *inputs* e *targets*. Aliás, alguns desses valores apresentam uma autocorrelação negativa (poderiam considerar-se até 19 valores, mas a sua distância em relação ao período  $k$  é muito grande).

Tendo em conta os resultados da FAC e FAP e todas as considerações prévias, parece uma boa alternativa a escolha de um número de *delays* de 7 (neste caso, 7 dias ou uma semana).

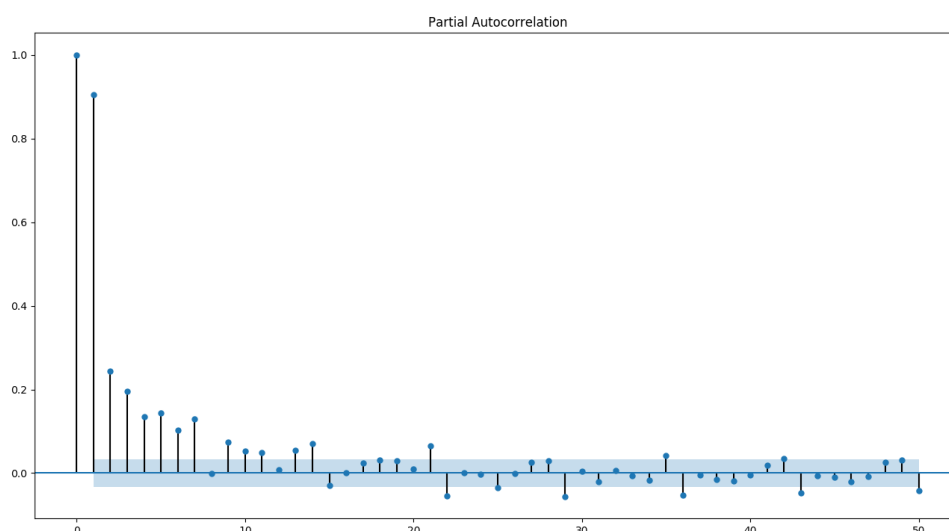


Figura 5.9 – FAP dos primeiros 50 *lags* dos consumos diários.

### Série horária

A FAC apresenta, novamente, a mudança sazonal. O mais relevante é a variação que existe ao longo de um dia. A priori, poderia ser estabelecido um número de *lags* de vários dias, mas devem ser tidas em conta as limitações de tempo computacional e consumo de memória. Neste caso, pode resultar mais útil recorrer diretamente à FAP.

A FAP manifesta uma autocorrelação com os 48 valores anteriores. A partir daí, as correlações não são significativas (por ser a série expressada em intervalos de meia hora, 48 valores equivalem a 24 horas, isto é, um dia). Observando o gráfico pode-se detetar uma correlação relativamente fraca (os valores de autocorrelação flutuam entre 0,27 e 0,013, mas os limites de confiança são próximos de 0,01, portanto não são tão fracos como cabia pensar). Tendo em consideração este resultado, o qual parece lógico, considera-se uma boa opção a escolha de um *delay* de 48 intervalos de tempo.

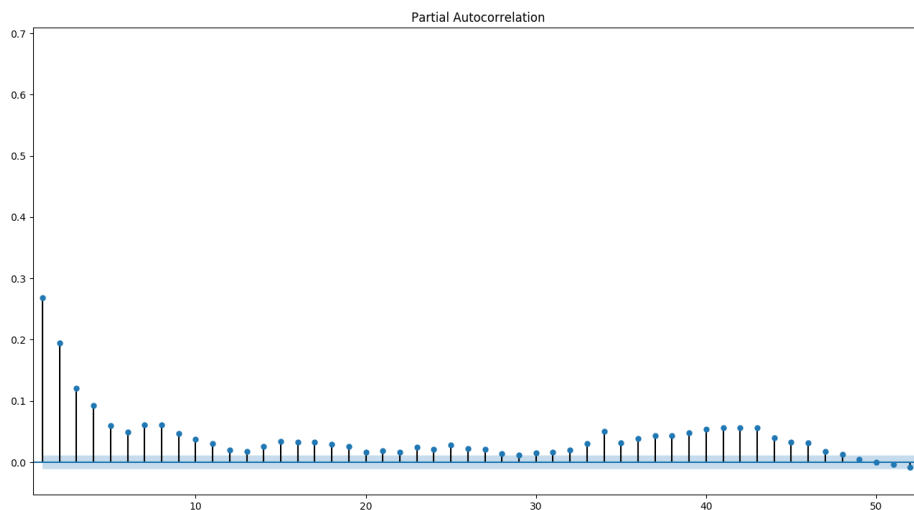


Figura 5.10 – FAP dos primeiros 50 *lags* dos consumos horários.

## 5.2.2 Definição da arquitetura da rede

### Escolha das funções de ativação

Para que o PMC seja capaz de aproximar funções não lineares (como é o caso de estudo) é necessário que as funções de ativação nas camadas ocultas também sejam não lineares. Sendo que é possível utilizar qualquer função diferenciável e monotonamente crescente, podem ser utilizadas as funções descritas no capítulo de revisão bibliográfica (função tangente hiperbólica, arcotangente e sigmoideal).

- Devido ao conforto que supõe trabalhar com saídas positivas, optou-se por utilizar a função sigmoideal (por ser estritamente positiva, limitada ao intervalo  $[0,1]$ ) nas camadas ocultas.
- Na camada de output é frequente a utilização de funções lineares, e por esta razão é o tipo de função utilizada na camada de saída.

### Escolha da dimensão dos grupos de validação e teste

Como já referido no capítulo 2, a dimensão a adotar para os subconjuntos de validação e teste varia com relação à complexidade da função que se deseja aproximar e o número total de padrões.

O tamanho dos grupos de validação e teste não tem grande influência no resultado final após dado número de padrões de treino, sendo que 20% é uma percentagem que resulta num amplo conjunto de casos (Brownlee, 2016).

Desta forma, se houver bastantes dados disponíveis (e devem havê-los para que a RNA tenha bom desempenho) – o mais adequado seria a utilização de subconjuntos de validação e teste com aproximadamente 30% do total de dados disponíveis.

Serão utilizadas as percentagens seguintes relativas à percentagem do conjunto total dos dados, consoante às recomendações recompiladas da literatura (Brownlee, 2016).

- *Conjunto de treino: 55%.*
- *Conjunto de teste: 30%.*
- *Conjunto de validação: 15%.*

#### **Definição da dimensão da camada de input**

No PMC existe uma única camada de input, a qual possui um número de nós igual a:

$$N^{\circ} \text{ nós} = N^{\circ} \text{ features} + 1 \quad (5.4)$$

Sendo que o número de *features* é coincidente com o número de colunas do conjunto de dados, aos quais é adicionado mais um neurónio associado à constante o *bias*. Muitos autores, como Heaton (2005), não consideram necessário adicionar mais um nó

A rede NARx possui duas camadas de entrada; a primeira, para uma função de inputs  $x(t)$ , e a segunda para os *targets*,  $y(t)$ . Cada uma delas possui um número de nós definido de igual forma que no caso do PMC.

O número de camadas de entrada na rede LSTM é único, porém a dimensão da camada de entrada é mais complexa; a estrutura de entrada deve ser definida da forma [*samples, time steps, features*].

Na Tabela 5.1 são apresentadas as arquiteturas das camadas de entrada para cada modelo de rede utilizado.

Tabela 5.1 – Arquitetura das camadas de entrada para cada modelo de RNA.

Modelo de RNA	Nº camadas de entrada	Dimensão camada/s de entrada caso diário	Dimensão camada/s de entrada caso horário
PMC	1	3 nós	2 nós
NARx	2	2 nós na camada $x(t) \rightarrow$ temperatura	2 nós*
		2 nós na camada $y(t) \rightarrow$ consumo	
LSTM	1	[1962,1,14] **	[19298,1,48] **

\* Neste caso, ao não existirem inputs exógenos, a rede passa a ser NAR, com um funcionamento similar à NARx (a retropropagação é transmitida desde a camada de output diretamente à entrada  $y(t)$ , sem passar pelas camadas ocultas, a diferença do PMC recorrente).

\*\*Devido ao facto da rede ser supervisionada, o número de *timesteps* é fixado em 1, de maneira que o número de *features* passa a ser  $n^\circ$  de *features*  $\times$   $n^\circ$  de *delays* ou *lags*.

No caso da série diária: 2 *features* (*temperatura e o próprio consumo de água*)  $\times$  7 *delays* = 14.

No caso da série horária: 1 *feature* (*consumo de água*)  $\times$  48 *delays* = 48.

#### Definição da dimensão da camada de output

A previsão de consumos ou qualquer variável numérica é um problema de regressão. Neste tipo de problemas, ao contrário dos problemas de classificação, apenas é utilizado um nó na camada de saída (também única).

Todos os tipos de rede definidos apresentam esta característica, incluído o modelo LSTM.

#### Definição da dimensão e número de camadas ocultas

A definição da dimensão da camada(s) oculta(s) é, possivelmente, a parte mais importante e complexa no que diz respeito à definição da arquitetura dos diferentes modelos de RNA. Poucos nós podem originar resultados deficientes devido à simplicidade da aproximação, enquanto a eleição de muitos nós pode dar numa aproximação ótima no treino, mas aproximações completamente inapropriadas na validação.

Existe um consenso para estabelecer o número de camadas ocultas: aquelas situações em que a *performance* ou desempenho da rede neuronal melhora com uma segunda (ou terceira, quarta...) camada oculta são muito escassas. Uma camada escondida é suficiente para a maior parte dos problemas. Por vezes, a inclusão de mais do que uma camada oculta produz resultados similares ao caso de uma única camada oculta, mas com tempos computacionais superiores (Heaton, 2005).

Relativamente ao número de nós, existem algumas regras empíricas para a sua escolha. A mais utilizada é “a dimensão ótima da camada oculta encontra-se entre a dimensão das camadas de output e input”.

Em conclusão, na maior parte dos problemas, uma camada oculta podia produzir resultados decentes, estabelecendo a configuração da camada oculta usando duas regras (Heaton, 2005):

- i. O número de camadas ocultas é igual a um.
- ii. O número de nós nessa camada é a média dos nós de input e output.

Apesar destas indicações, decidiu-se experimentar com um amplo intervalo de número de nós por camada, e com uma e duas camadas ocultas.

A escolha do melhor resultado baseia-se na configuração com a qual seja obtido o menor EQM num tempo razoável. A informação mais relevante dos diferentes treinos pode ser consultada no Anexo II.

Em seguida, são apresentados gráficos com a relação do EQM o de teste de cada tipo de modelo de rede neuronal e o número de nós na camada oculta, tanto para a série diária de consumos como para a horária.

#### Série diária

- PMC não autorregressivo: Figura 5.11.
- Rede NARx: Figura 5.12.
- Rede LSTM: Figura 5.13.

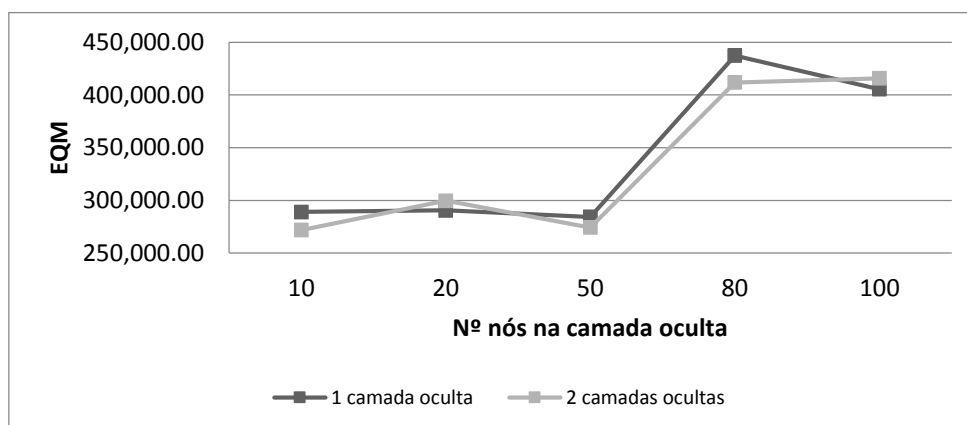


Figura 5.11 – EQM médio após validação cruzada para PMC não recorrente de uma e duas camadas ocultas, caso diário.

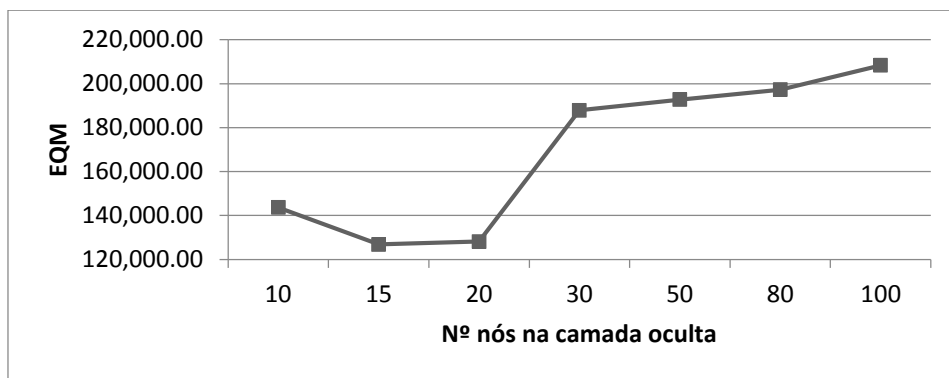


Figura 5.12 – EQM médio após validação cruzada para rede NARx de uma camada, caso diário.

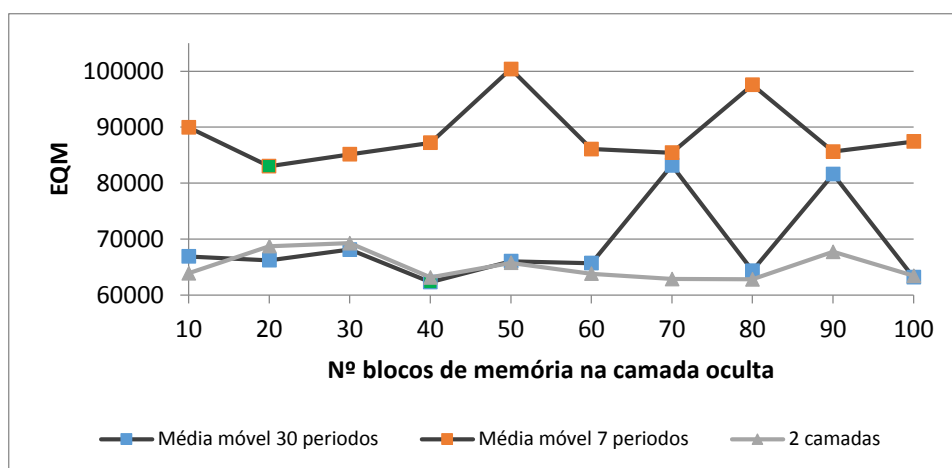


Figura 5.13 – EQM médio após validação cruzada para rede LSTM, com correções baseadas em média móvel de 7 e 30 períodos (1 e 2 camadas), caso diário.

A partir deste procedimento, conclui-se:

- O PMC não recorrente produz um EQM de teste muito elevado. Aqui se reflete a particularidade das séries temporais: é necessário ter presentes os tempos prévios para a obtenção duma boa aproximação. Aliás, a inclusão de duas camadas ocultas não produz uma melhora significativa respeito à utilização de uma única camada oculta. Outro fator em contra é a presença de *overfitting* que se produz.
- A utilização do modelo NARx produz resultados muito melhores do que no caso do PMC não recorrente, com tempos computacionais muito baixos (os mais baixos de todos os modelos, devido à utilização do algoritmo de treino de Levenberg-Marquardt, combinado com *early-stopping*, portanto, o número de épocas não é escolhido pelo utilizador, mas o algoritmo para quando o EQM de validação começa a crescer). Uma característica contra este modelo é o alto consumo de memória computacional consumida no processo de treino, o que, por vezes, faz com que este fosse muito lento e o programa acabe por ter um conflito e terminar. Em princípio, a rede foi testada num intervalo de [10-100], com um incremento de 10 nós cada vez, mas ao ver um claro aumento a partir da utilização de 20 nós, experimentou-se com um valor intermédio de

15 nós na camada oculta, resultando ser o valor ótimo, que produz um menor EQM. Nem sempre utilizar um número de nós elevado garante um melhor resultado, pois é necessário um maior tempo e memória computacional e mesmo assim produzir piores resultados do que um baixo número de nós.

- A rede LSTM resulta ser o melhor modelo para o treino da série temporal diária, com bastante diferença com as anteriores. Contrariamente a isso, a rede LSTM precisa de tempos computacionais mais elevados. Por efeito desta limitação, o número de épocas a escolher depende do tempo computacional a considerar como aceitável. Experimentando com várias épocas, conclui-se que:
  - 100 épocas produzem tempos computacionais de entre 19-25 minutos. Os resultados de EQM são muito semelhantes aos produzidos com 50 épocas.
  - 50 épocas implicam tempos computacionais entre 9-14 minutos.
  - 70 épocas (valor intermédio) produzem tempos computacionais num intervalo de 16-21 minutos, com resultados similares (ou até piores) à utilização de 50 épocas.
  - Menos de 50 épocas produzem resultados de EQM que, por vezes, podem resultar pouco satisfatórios, porém os tempos computacionais são menores.

Daí que o número de épocas selecionado seja 50.

De outro lado, decidiu-se ver como afeta a utilização das diferentes correções já referidas anteriormente, sendo que a correção baseada na média móvel de 30 períodos (um mês) produz uns resultados mais satisfatórios, como já foi dito. Isto é feito para observar a diferença de resultados que existe quando adotar uma ou outra consideração. O melhor resultado é obtido para um número de blocos por camada de 40, e o EQM 25% menor (EQM de 63000 face 83000) que na dimensão ótima da rede com correção com um período de 7 dias (20 nós). A utilização de duas camadas ocultas (com uma correção de média móvel de 30 períodos) produz resultados bons, semelhantes aos melhores valores obtidos com uma única camada oculta, mas com um tempo computacional duas vezes superior ao requerido com uma única camada oculta.

Por conseguinte, a rede LSTM resulta ser a mais vantajosa para o caso dos consumos diários.

#### Série horária

- PMC não autorregressivo: Figura 5.14.
- Rede NARx: Figura 5.15.
- Rede LSTM: Figura 5.16.

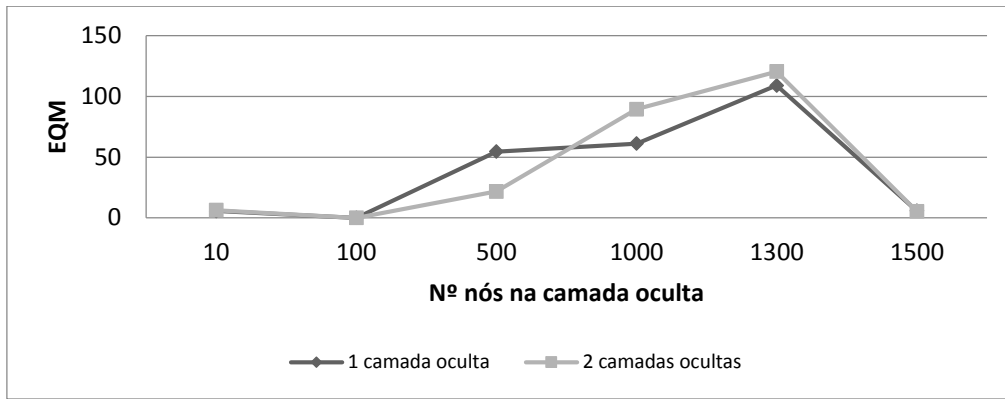


Figura 5.14 – EQM médio após validação cruzada para PMC não recorrente de uma camada e duas camadas, caso horário.

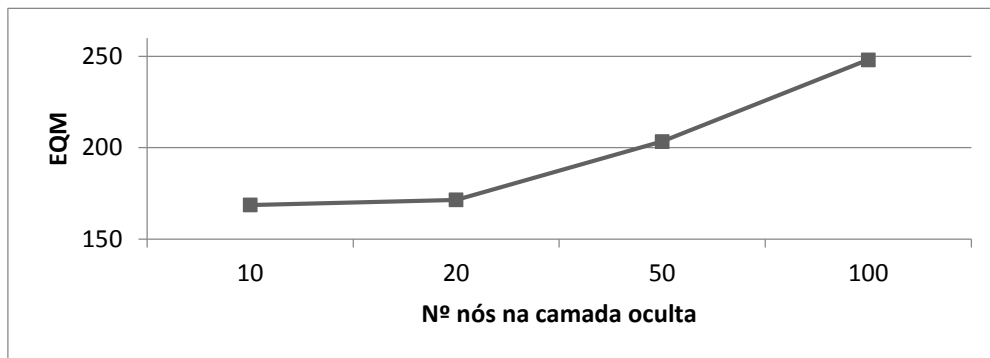


Figura 5.15 – EQM médio após validação cruzada para rede NARx de uma camada oculta, caso horário.

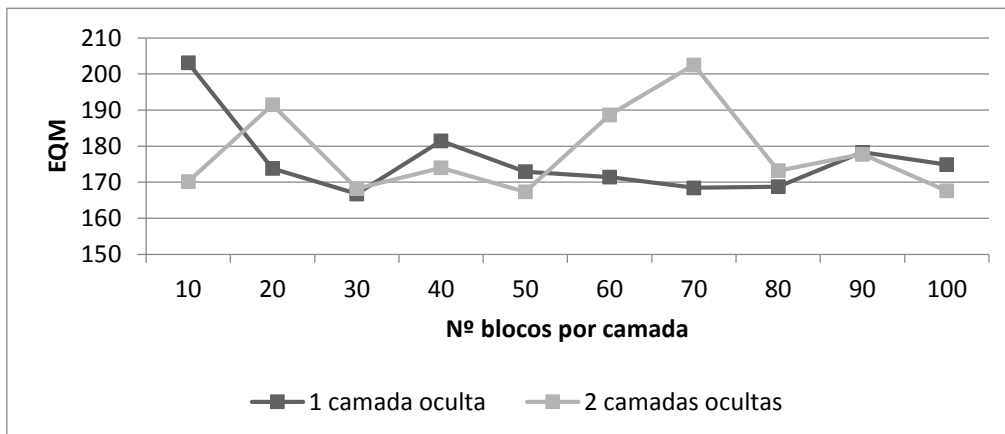


Figura 5.16 – EQM médio após validação cruzada para rede LSTM de uma e duas camadas ocultas, caso horário.

Destes gráficos pode tirar-se esta informação:

- O PMC não recorrente parece apresentar uns resultados excelentes. Porém, a tabela do Anexo II verifica a presença de overfitting para todas as configurações, o qual invalida os resultados e demonstra, mais uma vez, a ineficácia do modelo de PMC não recorrente para a aproximação de séries temporais. Isto é verificado tanto para a utilização de uma camada oculta como para duas.



- O modelo NARx apresenta resultados adequados quanto menor for o número de nós utilizado. Seria conveniente a utilização de uma camada oculta com 10 nós, produzindo assim um EQM de teste de aproximadamente 169. O pioramento dos resultados parece ter certa parecença com uma lei exponencial, conforme o número de nós aumentar. O modelo NARx apresenta uma grande vantagem sobre o modelo LSTM: os tempos computacionais são muito menores. Contudo, a correlação entre outputs e targets resultante é baixa.
- Por último, o modelo LSTM apresenta mais estabilidade com a utilização de uma única camada oculta. O número de nós ideal poderia ser 30 ou 50 (produzindo um EQM de teste próximo de 167). A utilização de duas camadas ocultas não é conveniente, em virtude à relação tempo computacional – resultados, pois é necessário aproximadamente o dobro de tempo para produzir resultados similares aos obtidos com uma única camada oculta.

Em conclusão, poderia dizer-se que tanto o modelo NARx como o LSTM poderiam ser utilizados por produzirem os melhores resultados em termos de EQM. Não obstante, o modelo LSTM apresenta algumas vantagens, como são:

- O EQM de teste é ligeiramente menor (167 face 169).
- Existe uma (muita) maior correlação entre os targets e os outputs ( $R = 0,7443$  no caso da rede NARx face  $R = 0.9255$  no caso da rede LSTM).

Portanto, para o caso da série horária, conclui-se que o modelo mais adequado é, de novo, o LSTM.

Relativamente ao número de épocas utilizado, a escolha foi feita de maneira semelhante ao caso dos consumos diários. Apenas foi usado um número de épocas igual a 10, devido a produzir tempos computacionais de 10 a 13 minutos, enquanto um número de épocas igual a 50 produz tempos computacionais de horas (01h10min:25 como valor de referência), não existindo uma melhora considerável, e até eventualmente piores (EQM=179.22 para uma rede de 50 blocos de memória e 50 épocas). Não se considera apropriada a inclusão de outra camada oculta, pois os resultados manifestam tempos computacionais de aproximadamente 30 minutos, para produzir os resultados do gráfico.

### 5.2.3 Validação do modelo

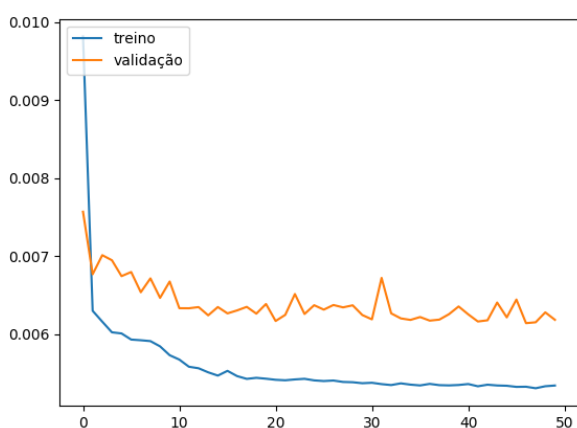
#### Série diária

Para validar o modelo, decide-se começar pelos modelos com menor EQM da escolha realizada previamente. Se o primeiro deles não verificar a validação, o próximo que produza um EQM imediatamente superior, é submetido a validação, e assim por diante. Para tal, é preciso utilizar os resultados dos dados do conjunto validação, mas sobretudo, a análise gráfica.

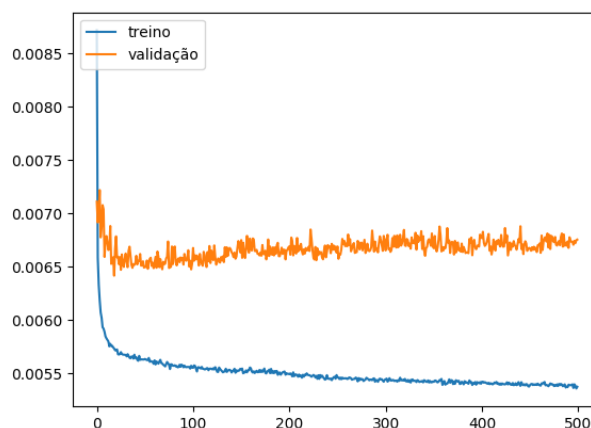
Os resultados de validação das configurações da rede LSTM duma camada oculta que apresentaram o EQM mais baixo são apresentados na Tabela 5.2.

Tabela 5.2 – Características úteis para a validação do modelo LSTM da série diária.

ARQUITECTURA	REQM ( $m^3$ )	OVERFITTING	Correlação outputs-targets
1 CO/40 nós	249,56	Não	0,9834
1 CO/100 nós	251,32	Não	0,9821
1 CO/80 nós	253,68	Não	0,9888



a) EQM 50 épocas.



b) EQM 500 épocas.

Figura 5.17 – EQM médio de validação e treino (padronizados) para a rede LSTM de uma camada oculta com 40 blocos de memória. Série diária.

Na Figura 5.17 é apresentado, à esquerda, um gráfico com o EQM de validação e teste face ao número de épocas para a melhor configuração (uma camada oculta com 40 blocos de memória). Observa-se que existe uma tendência assintótica no EQM de validação a partir das 20 épocas. De facto, para um maior número de épocas (próximo às 60 épocas) existe um alto risco de aparição de *overfitting*, tal como mostrado no gráfico situado à direita.

A correlação que existe entre os valores de saída (previsão de teste) e a função objetivo é muito elevada ( $R=0,9834$ ), o qual indica uma boa relação linear entre ambos. Evidentemente, a relação é positiva, como ilustra a Figura 5.18. A elevada correlação não só é verificada para a configuração de 40 blocos de memória, mas também para o resto de configurações.

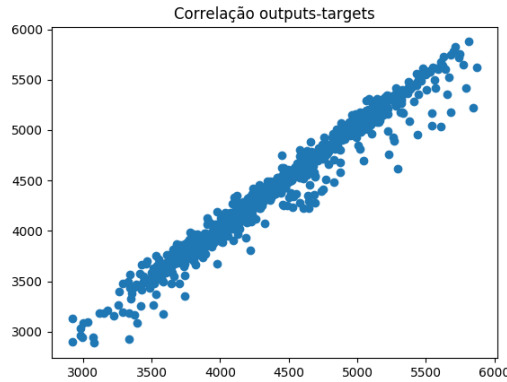


Figura 5.18 – Diagrama de correlação entre targets e saídas da rede LSTM com uma camada oculta e 40 blocos de memória, caso diário.

No que diz respeito aos resíduos, estes foram definidos como a diferença entre os valores reais ou de referência e os valores obtidos mediante a aproximação com a RNA. A ordem definida é importante, pois a partir dela, pode ser determinado se as previsões são sobrestimadas ou subestimadas, sempre referido ao total dos resíduos.

Na Figura 5.19 apresenta-se o histograma dos resíduos e a sua aproximação com uma função de distribuição. Na Tabela 5.3 é apresentado um resumo estatístico dos resíduos, obtidos correndo 5 vezes o modelo e calculando a média.

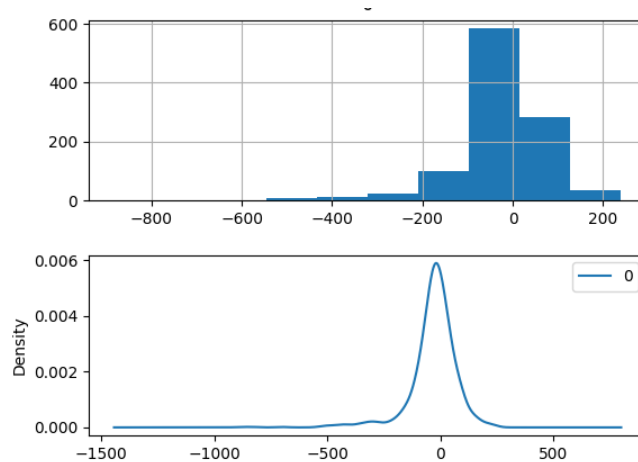


Figura 5.19 – Histograma e função de densidade dos resíduos, caso diário.

Tabela 5.3 – Resumo estatístico dos resíduos do modelo LSTM da série diária.

<b>Média</b>	-28,72
<b>Desvio padrão</b>	105,959
<b>min</b>	-882,87
<b>25</b>	-60,72
<b>50</b>	-18,62
<b>75%</b>	24,25
<b>máx</b>	240,23
<i>Unidades em m<sup>3</sup></i>	

Com base nestas estatísticas, pode afirmar-se que o modelo tende a sobrestimar os resultados (média negativa). No que diz respeito ao valor mínimo de -882.87, este é apenas um valor pontual devido à dificuldade da rede para estimar os primeiros valores, tal e como pode observar-se no gráfico de autocorrelação.

Um REQM de 249,56 m<sup>3</sup> supõe um erro sobre a média dos consumos (3873,97 m<sup>3</sup>) do 6,44%, ou, visto desde outra perspectiva, um acerto do 93,56%. Como consequência de todo o anterior, o modelo é válido.

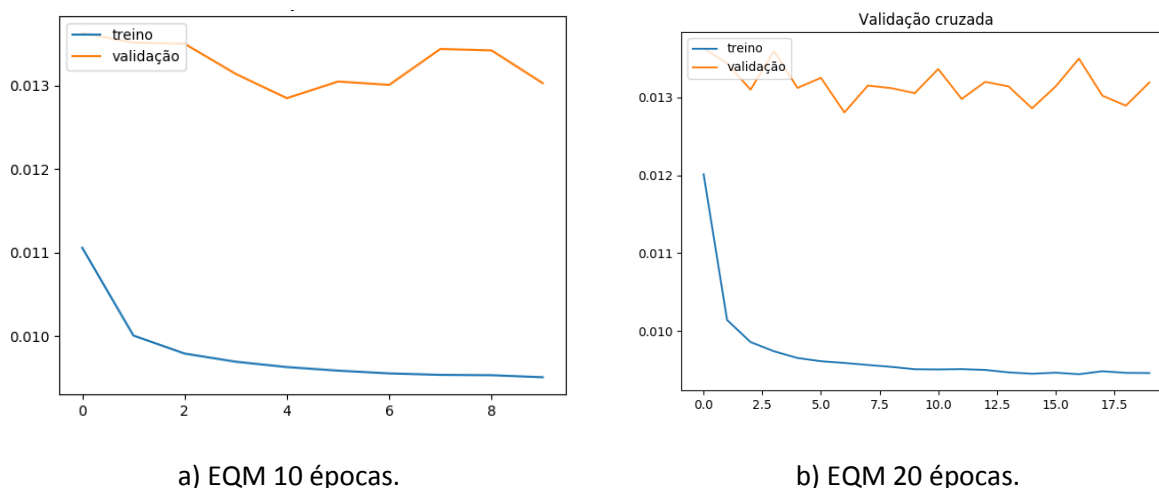
#### Série horária

Para avaliar o modelo de consumos horários, é seguido o mesmo procedimento utilizado no caso da série diária. Se o primeiro deles não verificar a validação, o próximo que produza um EQM imediatamente superior, é submetido a validação, e assim por diante.

Os resultados de validação das configurações da rede LSTM numa camada oculta que apresentaram o EQM mais baixo são apresentados na Tabela 5.4.

Tabela 5.4 – Características úteis para a validação do modelo LSTM da série horária.

<b>ARQUITECTURA</b>	<b>REQM (m<sup>3</sup>)</b>	<b>OVERFITTING</b>	<b>Correlação outputs-targets</b>
1 CO/30 nós	12,91	Não	0,9255
1 CO/70 nós	12,97	Não	0,9371
1 CO/80 nós	12,99	Não	0,9335



a) EQM 10 épocas. b) EQM 20 épocas.  
 Figura 5.20 – EQM médio de validação e treino (padronizados) para a rede LSTM de 30 blocos de memória para um número de épocas igual a 10 (esquerda) e 20 (direita). Série horária.

Na Figura 5.20 é apresentado um gráfico com o EQM de validação e teste face ao número de épocas para a melhor configuração (uma camada oculta com 30 blocos de memória). À esquerda, observa-se que existe um aumento local no EQM de validação. Isto não indica presença de *overfitting*, pois para um maior número de épocas, o EQM continua estável (direita).

A correlação que existe entre os valores de saída (previsão de teste) e a função objetivo é aceitável ( $R=0,9255$ ), o qual indica uma boa relação linear entre ambos, mesmo que seja uma das menores correlações obtidas. Evidentemente, a relação é positiva, como ilustra a Figura 5.21.

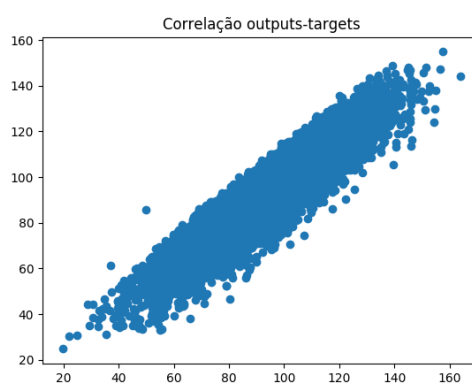


Figura 5.21 – Diagrama de correlação entre targets e saídas da rede LSTM com uma camada oculta e 30 blocos de memória, caso horário.

No que toca aos resíduos, estes foram definidos de maneira semelhante ao caso diário, tal como mostrado na Figura 5.22 e na Tabela 5.5.

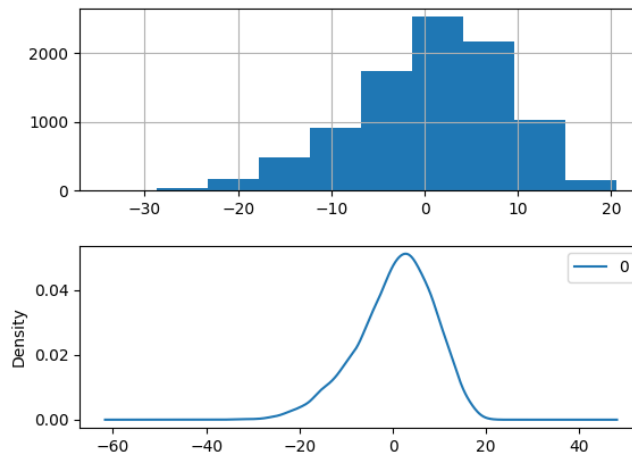


Figura 5.22 – Histograma e função de densidade dos resíduos, caso horário.

Tabela 5.5 – Resumo estatístico dos resíduos do modelo LSTM da série diária.

<b>Média</b>	0,63
<b>Desvio padrão</b>	8,22
<b>min</b>	-34,24
<b>25</b>	-4,31
<b>50</b>	1,50
<b>75%</b>	6,509727
<b>máx</b>	20,615
<i>Unidades em m<sup>3</sup></i>	

Com base nestes estatísticos, pode afirmar-se que o modelo é muito próximo da distribuição normal, pois a média é aproximadamente zero. Os valores máximo e mínimo não são muito elevados, pelo que as previsões estão bastante delimitadas, sem previsões anómalas. O facto da média ser positiva indica que as previsões estão a subestimar vagamente os valores de referência.

### 5.3 PROCEDIMENTO DE MODELAÇÃO DO MODELO ARIMA

O procedimento para identificar o modelo ARIMA idóneo é baseado na análise da série temporal, já realizada em secções prévias, e na observação das funções de autocorrelação (FAC) e autocorrelação parcial (FAP), as quais também foram estudadas previamente.

Na Figura 5.23 é apresentado um fluxograma com o procedimento completo a seguir para a determinação do modelo ARIMA.

Com base na análise horária, o primeiro passo –conhecer se a série é ou não estacionária- já foi respondido anteriormente: nenhuma das séries, diária e horária, é estacionária, pois existe sazonalidade, o que faz

com que a média seja diferente ao longo do tempo. Outra forma de sabê-lo é mediante a FAC: se os valores diminuem muito rapidamente para cada atraso, então a série é estacionária. Porém, a série diminui muito devagar (consultar Anexo I); conseqüentemente, a série é claramente não estacionária.

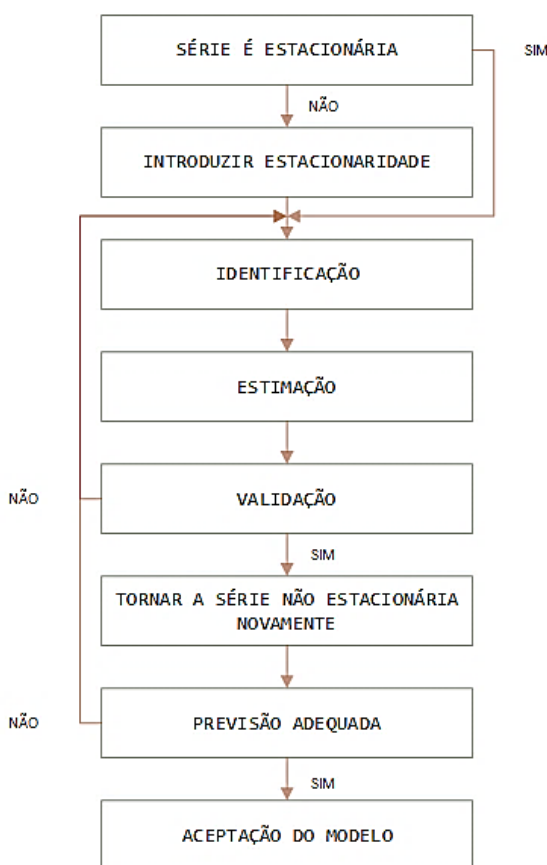


Figura 5.23 – Fluxograma do procedimento para a determinação de modelos ARIMA.

A metodologia ARIMA assume que a forma de conseguir séries estacionárias consiste em diferenciar regular e/ou sazonalmente (UNED, 2015). Uma série é integrada de ordem zero caso seja estacionária  $I(0)$  e integrada de ordem um  $I(1)$  se for necessária uma primeira diferença regular para conseguir, e assim continuamente. Caso sejam consideradas ambas a parte regular e sazonal conjuntamente, então uma série, por exemplo  $I(1,1)$  é aquela que se faz estacionária, ou integrada de ordem zero  $I(0)$ , realizando uma primeira diferença regular  $[d(Z_t) = Z_t - Z_{t-1}]$  e outra sazonal  $[ds(Z_t) = Z_t - Z_{t-s}]$ .

### Série diária

A diferenciação já foi feita para os modelos de RNA. Portanto, procede-se ao estudo da série diferenciada. A FAC da série diária com a componente regular diferenciada (Figura 5.24) não apresenta não-estacionariedade regular, mas continua a ter não-estacionariedade sazonal, pois cada 7 dias existe uma componente estacional forte (para os atrasos 7,14,21,28,35...), isto é, a série não é integrada de ordem  $I(1)$ . Portanto, é necessário diferenciar sazonalmente para saber se a série é integrado de ordem um sazonal  $I(0,1)$ .

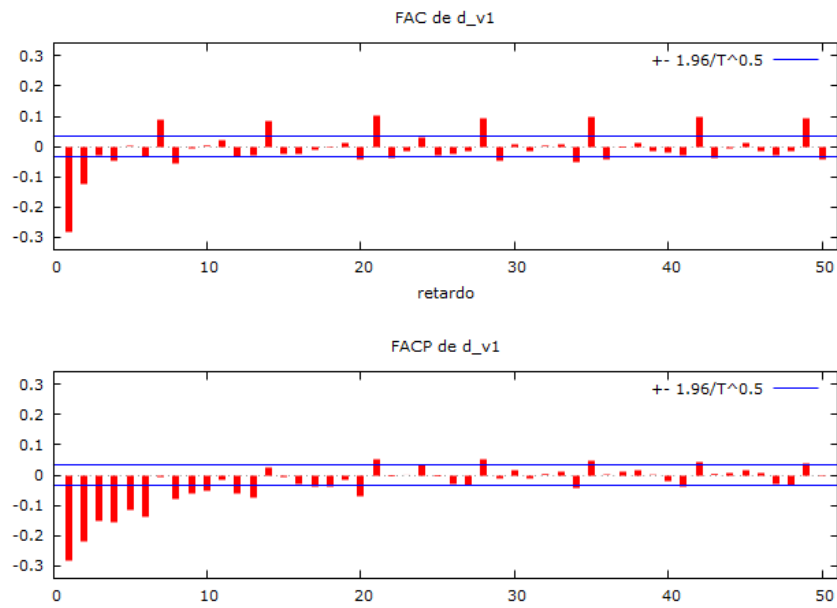


Figura 5.24 – FAC e FAP da série diária após diferenciação regular.

É necessário conhecer a série diferenciada sazonalmente. Neste caso, a série apresenta a FAC ilustrada na Figura 5.25.

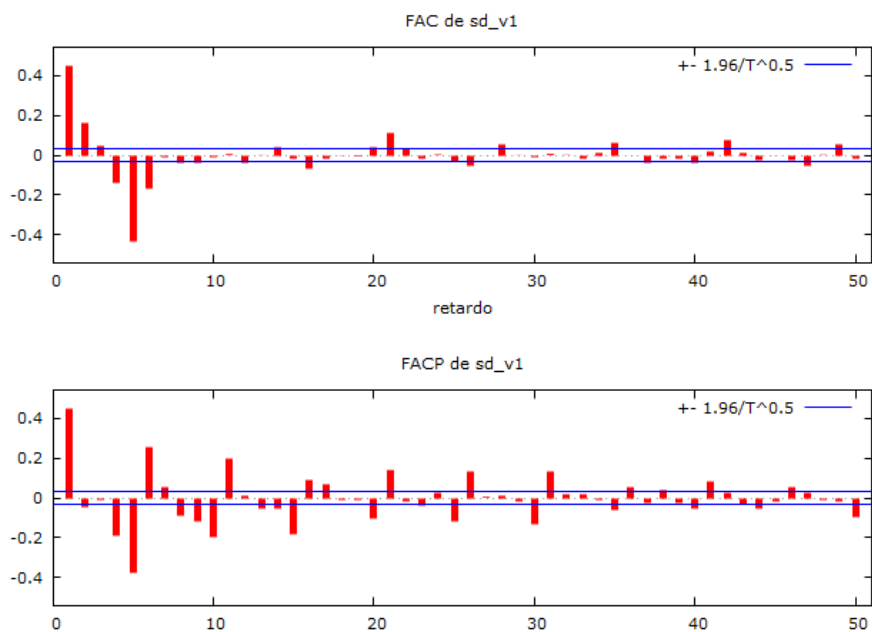


Figura 5.25 – FAC e FAP da série diária após diferenciação sazonal.

A função de autocorrelação total (FAC) diminui rapidamente. Consequentemente, o consumo de água é (aproximadamente) estacionário sazonalmente. Portanto, a série é integrada de ordem um sazonal  $[I(0,1)]$ .

Mesmo assim, pode ser corroborado que a série não é integrada de ordem um regular e sazonal  $[I(1,1)]$  calculando uma diferença sazonal a partir da série diferenciada regularmente. A respectiva FAC é apresentada na Figura 5.26.



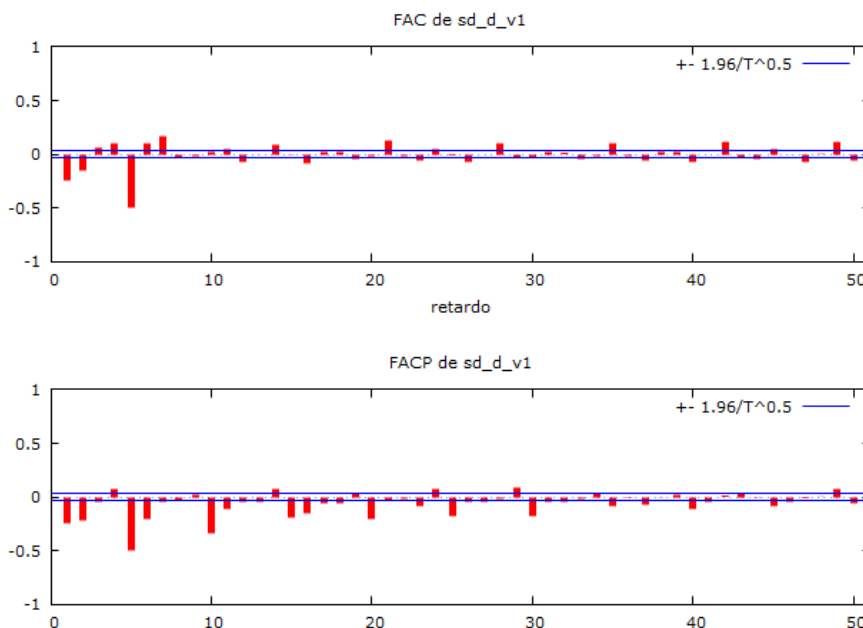


Figura 5.26 – FAC e FAP da série diária após diferenciação sazonal sobre diferenciação regular.

Efetivamente, a FAC não diminui lentamente. Por conseguinte, verifica-se que a série não é  $[I(1,1)]$ , mas  $[I(0,1)]$ .

Para conhecer o resto de parâmetros da série ARIMA, é necessário comparar a forma das FAC e FAP com os seus homónimos do Anexo III.

Em vista da comparativa de correlogramas (ou FAC e FAP), estima-se que a parte regular corresponde:

- Ao modelo  $AR(2)$ .
- Ao modelo  $MA(1)$ .

De outra parte, conjectura-se que a parte sazonal corresponde:

- Ao modelo  $AR(0)$ .
- Ao modelo  $MA(1)$ .

Por conseguinte, o modelo ARIMA estimado é da forma:  $ARIMA(2,0,1)(0,1,1)_7$ . Portanto, conclui a parte de identificação do modelo. Porém, os parâmetros  $AR$  e  $MA$  foram estimados em relação a uma aproximação gráfica, e por causa disso podem existir erros. Por esta razão, experimenta-se com outras alternativas possíveis

De seguida, procede-se à obtenção dos parâmetros ARIMA com o programa *Gretl*, os quais são apresentados na Tabela 5.6.

Tabela 5.6 – Parâmetros ARIMA da série diária.

Parâmetro	Estimação
AR(1)	1,5231
AR(2)	-0,5261
MA(1)	0,8934
Constante	0,007936

Onde:

$AR(1)$  – é o termo  $a_1$ ;

$AR(2)$  – é o termo  $a_2$ ;

$MA(1)$  – é o termo  $b_1$ ;

Uma vez identificado o modelo pré-definido, é realizada a validação dos resultados. Para tal, é útil recorrer às FAC dos resíduos (Ruído Branco, RB). A sua representação é mostrada na Figura 5.27.

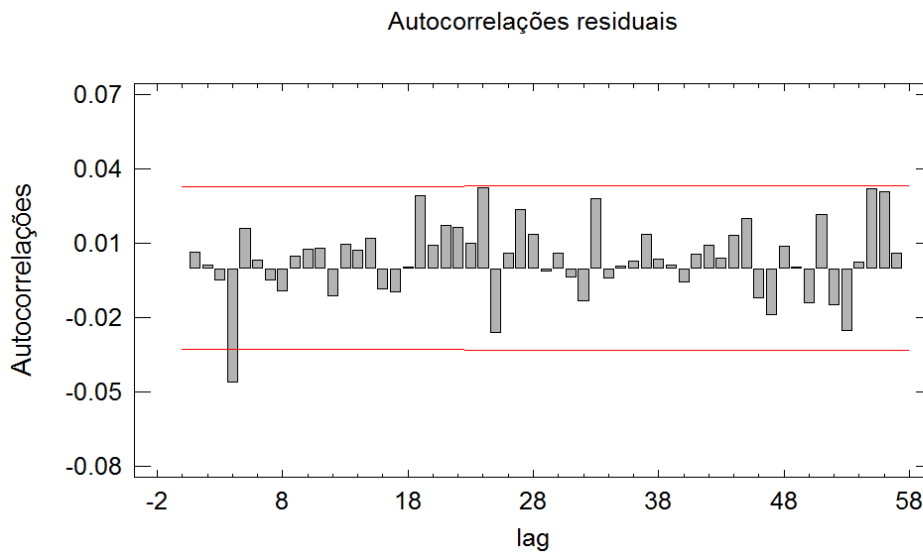


Figura 5.27 – FAC dos resíduos diários após aproximação mediante modelo ARIMA.

A validação do(s) modelo(s) é efetuada recorrendo à análise das correlações dos resíduos (FAC e FAP residual), isto é, determinar se os resíduos são ou não ruído branco (RB). Um ruído branco é uma série de dados que se caracteriza por ter distribuição normal, média e covariância nulas e variância constante (UNED, 2015). A hipótese de um modelo ser válido vai ser definida por um ou vários estatísticos. No âmbito deste trabalho, utiliza-se o estatístico de Box-Pierce ( $Q_k$ ).

Estabelece-se, portanto, um procedimento para validar o modelo:

- Representar as FAC e FAP para os resíduos. Se os resíduos estimados fossem ruído branco, não deve aparecer (quase) nenhum valor significativo, isto é, que não ultrapassem os limites de confiança (bandas assintóticas do 95% de confiança), definidos por:

$$\left(\frac{-1,96}{\sqrt{T}}, \frac{1,96}{\sqrt{T}}\right) \tag{5.5}$$

Onde  $T$  é o número de valores da série  $X_t$ .

- Teste de Box-Pierce. É um estatístico que indica se os resíduos são RB. Segue uma distribuição  $\chi^2$ , sendo  $n$  o número de observações. Caso cada valor  $r_k$  seja próximo de zero,  $Q_k$  será muito pequeno. De outro modo, se alguns  $r_k$  são grandes (negativa ou positivamente)  $Q_k$  será muito grande (Quesada P., 2011).

$$Q_k = n \sum_{k=1}^h r_k^2 \tag{5.6}$$

Uma vez calculado o estatístico de Box-Pierce, compara-se com as tabelas de  $\chi^2$ , onde  $h$  corresponde a  $v$ . Se o p-valor de  $Q_k$  é maior do que o p-valor 0,05, então os resíduos possuem um comportamento de RB e o modelo pode ser validado.

Se o comportamento é de RB, podem ser considerados aleatórios.

A FAC só apresenta um valor significativo, por acima dos limites, no atraso 4 (-0,04622).

As autocorrelações serão não significativas (ruído branco) se o P-valor é menor do que 0,05 (isto é,  $a_t$  é aleatório).

O valor de  $Q_k$  tendo em consideração 50 valores. O resultado do teste é apresentado na Tabela 5.7.

Tabela 5.7 – Teste de RB de Box-Pierce da série diária.

Teste de Box-Pierce para 50 autocorrelações	
Estatística de teste da amostra	36,5022
P-valor	0,84065 > 0,05

Portanto, os resíduos podem ser considerados RB, e o modelo é assim validado.

Após a validação do modelo, a série é novamente transformada a não-estacionária.

Uma vez definido e validado o modelo ARIMA, ele pode ser escrito analiticamente como:

$$(1 - B)(1 - B^7)W_t = Z_t = a_1Z_{t-1} - 1 + a_2Z_{t-2} - 2 + bV_{t-7} + V_t \tag{5.7}$$

Cuja forma compacta é:

$$(1 - a_1B - a_2B^2)Z_t = (1 + bB^7)V_t \quad (5.8)$$

Portanto, a equação analítica do modelo  $ARIMA(2,0,1)x(0,1,1)_7$  é:

$$Z_t = 0,007936 + 1,5231Z_{t-1} - 0,5261Z_{t-2} + 0,8943V_{t-7} \quad (5.9)$$

Então:

$$(1 - B)(1 - B^7)W_t = 0,00794 + 1,523Z_{t-1} - 0,526Z_{t-2} + 0,894V_{t-7} + V_t \quad (5.10)$$

Operando na parte esquerda da equação, obtém-se:

$$(1 - B^7 - B + B^8)W_t = 0,00794 + 1,523Z_{t-1} - 0,526Z_{t-2} + 0,894V_{t-7} + V_t \quad (5.11)$$

$$W_t - W_{t-7} - W_{t-1} + W_{t-8} = 0,00794 + 1,523Z_{t-1} - 0,526Z_{t-2} + 0,894V_{t-7} + V_t \quad (5.12)$$

O consumo a partir de  $Z_t$  é definido pela Equação (120). Um exemplo das primeiras previsões é mostrado na Tabela 5.8.

$$W_t = Z_t + W_{t-7} - W_{t-1} + W_{t-8} \quad (5.13)$$

onde:

$W_t$  – é o valor de consumo para o tempo atual;

$W_{t-i}$  – é o valor de consumo correspondente a  $i$  lags.

Na Tabela 5.8 são calculadas as primeiras aproximações a partir da expressão anterior.

Tabela 5.8 – Primeiros valores da aproximação mediante método ARIMA, série diária.

Período	$W_t$ ( $m^3$ )	Previsão ( $m^3$ )	Valor Residual ( $m^3$ )
01/08/07	2791,42		
02/08/07	2987,32		
03/08/07	3145,54		
04/08/07	3023,4		
05/08/07	2955,79		
06/08/07	2970,95		
07/08/07	2946,35		
08/08/07	3114,08	2939,33	174,745
09/08/07	2996,63	3041,7	-45,0652
10/08/07	3166,94	3080,65	86,2936

Uma vez realizado esse passo, é calculado o EQM para a estimação inicial e as outras possíveis alternativas tal como na Tabela 5.9. Isto é possível graças à introdução do modelo(s) definido(s) no programa *Gretl*.

Tabela 5.9 – EQM dos modelos ARIMA da série diária.

Modelo ARIMA	REQM ( $m^3$ )	EQM ( $m^6$ )
$ARIMA(2,0,1)x(0,1,1)_7$	265,445	70461,0480
$ARIMA(2,0,1)x(0,1,2)_7$	265,514	70497,6842
$ARIMA(2,0,1)x(2,1,1)_7$	265,508	70494,4981
$ARIMA(2,0,2)x(1,1,1)_7$	265,605	70546,0160
$ARIMA(2,0,2)x(1,1,2)_7$	265,504	70492,3740

Efetivamente, tinha-se feita uma boa escolha do modelo, por ser o que menor EQM produz. No entanto, os outros modelos apenas variam umas décimas respeito dele.

Consequentemente, o modelo  $ARIMA(2,0,1)x(0,1,1)_7$  pode ser aceite.

Série horária

O procedimento a seguir é exatamente o mesmo. Eis um resumo com a informação necessária para entender a escolha do modelo ARIMA horário:

- FAC e FAP após diferenciação regular. Figura 5.28.  $\rightarrow AR(2), I[(1)]$  (não sazonalidade),  $MA(1)$ .
- FAC e FAP após diferenciação sazonal. Figura 5.29.  $\rightarrow AR(0), I[(0)]$  (sazonalidade),  $MA(2)$ .
- FAC e FAP de diferenciação sazonal sobre diferenciação regular. Figura 5.30.  $\rightarrow$  confirmação de ordem  $I[(1,0)]$ .

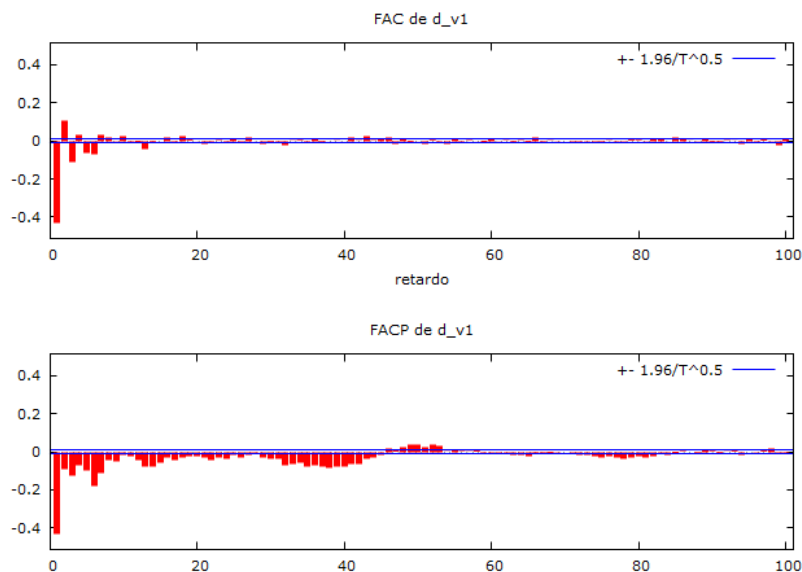


Figura 5.28 – FAC e FAP da série horária após diferenciação regular.

Novamente, a função de autocorrelação são comparadas com os gráficos do Anexo III.

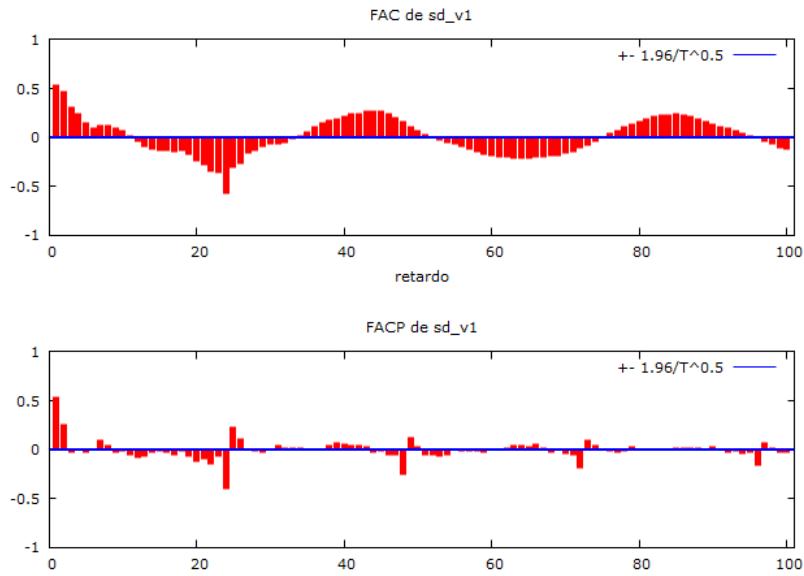


Figura 5.29 – FAC e FAP da série horária após diferenciação sazonal.

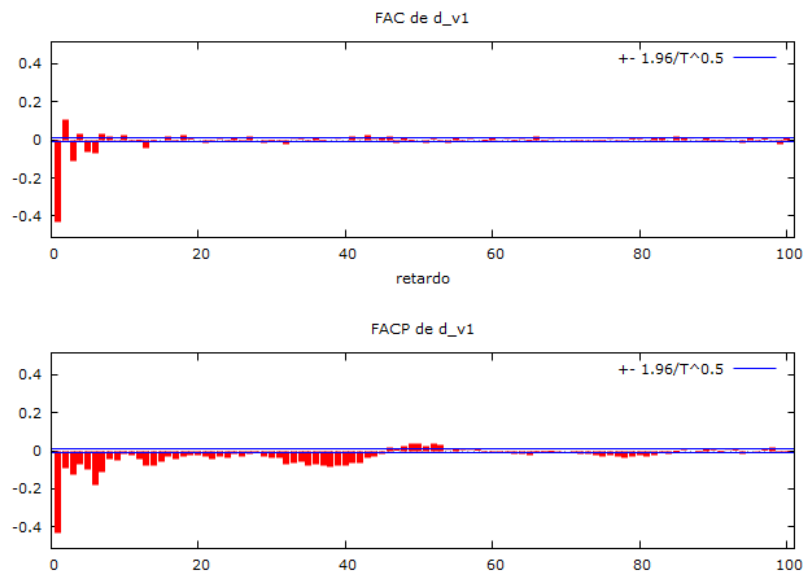


Figura 5.30 – FAC e FAP da série horária após diferenciação sazonal sobre diferenciação regular.

Assim, o modelo pré-definido é o  $ARIMA(2,1,1) \times (0,0,2)_{48}$ .

Os parâmetros do modelo ARIMA estimado são calculados e apresentados na Tabela 5.10:

Tabela 5.10 – Parâmetros ARIMA da série horária.

Parâmetro	Estimação
AR(1)	0,38916
AR(2)	0,245961
MA(1)	0,992145
SMA(1)*	-0.0241438
SMA(2)*	0.0109784

\*A letra “S” de SMA faz referência ao termo sazonal. Um modelo ARIMA (p,q,d)(P,Q,D) é, realmente, um SARIMA.

Onde:

$AR(1)$  – é o termo  $a_1$ ;

$AR(2)$  – é o termo  $a_2$ ;

$MA(1)$  – é o termo  $b_1$ ;

$SMA(1)$  – é o termo  $b_2$ ;

$SMA(2)$  – é o termo  $b_3$ .

Em seguida, é realizada a validação dos resultados. Para tal fim, recorre-se à FAC dos resíduos (Ruído Branco, RB). A sua representação é mostrada na Figura 5.31.

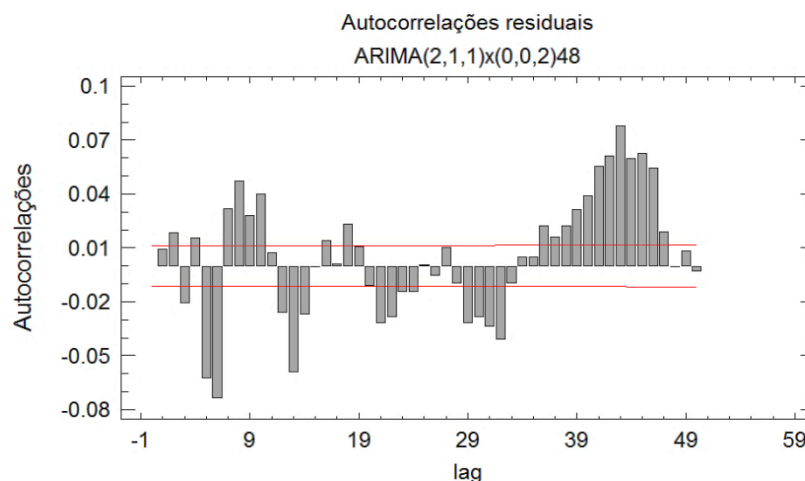


Figura 5.31 – FAC dos resíduos horários após aproximação mediante modelo ARIMA.

Observando a FAC dos resíduos, parece óbvio que existem muitos *lags* significativos ao ultrapassarem os limites de confiança. Daí que se possa pensar na rejeição do presente modelo. Contudo, é indispensável realizar o teste de Box-Pierce, cujo resultado é apresentado na Tabela 5.11:

Tabela 5.11 – Teste de Box-Pierce da série horária.

Teste de Box-Pierce para 120 autocorrelações	
Estatística de teste da amostra	2800,68
P-valor	0,0 < 0,05

Sendo o p-valor menor do que 0,05 (intervalo de confiança), efetivamente ratifica-se que os resíduos são muito importantes, e daí que o modelo não possa ser considerado válido.

Em concordância com o fluxograma inicial, haveria que voltar ao início e experimentar com outros modelos ARIMA que verificassem que os resíduos são ruído branco. A seguir, expõe-se, na Tabela 5.12, um resumo com os diferentes modelos testados e o seu EQM, para além da verificação mediante o método de Box-Pierce.

Tabela 5.12 – EQM dos modelos ARIMA da série horária.

Modelo ARIMA	REQM ( $m^3$ )	EQM ( $m^6$ )	Supera teste de Box-Pierce
$ARIMA(2,0,1)x(0,1,1)_{48}$	12,5998	158,7542	Não
$ARIMA(2,1,1)x(2,0,0)_{48}$	12,5600	157,7536	Não
$ARIMA(2,1,1)x(0,0,2)_{48}$	12,5997	158,7575	Não
$ARIMA(2,1,1)x(2,0,1)_{48}$	12,6004	158,7701	Não

Nenhum dos modelos supera o teste de Box-Pierce. Daí que o modelo não possa ser validado para o caso dos consumos horários.

Em conclusão: os consumos horários não podem ser adequadamente estimados através da metodologia ARIMA, ainda que o EQM seja mais baixo do que na aplicação das RNA. Doutro lado, os consumos diários respondem corretamente ante um modelo ARIMA.

## 5.4 CONSIDERAÇÕES FINAIS

Posto que o modelo de RNA demonstrou ser mais competente para prever os consumos do caso de estudo, apenas serão analisados os modelos LSTM diário e horário entendendo-os como os modelos mais adequados ao produzirem um EQM mais baixo.



## CAPÍTULO 6

### ANÁLISE DE RESULTADOS

Neste capítulo serão apresentados os resultados relativos aos modelos de RNA relativos ao modelo LSTM de uma camada oculta com 40 blocos de memória (caso diário) e ao modelo LSTM de uma camada oculta com 30 blocos de memória por produzirem o EQM mais baixo, através da comparação estatística da previsão correspondente aos dados de teste com os valores da série real, apoiada por informação gráfica.

#### Série diária

Após a validação do modelo, obtiveram-se os seguintes resultados em relação aos:

- Consumos anuais. Tabela 6.1 e Figura 6.1
- Consumos mensais. Tabela 6.2 e Figura 6.2.
- Consumos semanais. Tabela 6.3 Figura 6.3 e Figura 6.4.

Tabela 6.1 – Consumos diários médios anuais previstos face aos consumos de referência.

	PERÍODO							
	REFERÊNCIA	PREVISÃO	REFERÊNCIA	PREVISÃO	REFERÊNCIA	PREVISÃO	REFERÊNCIA	PREVISÃO
	2014	2014	2015	2015	2016	2016	2017	2017
<b>Média</b>	4562,59	4572,84	4549,09	4565,16	4219,42	4234,33	3760,11	3775,50
<b>Desvio padrão</b>	356,05	379,254	457,64	469,00	654,11	658,20	320,84	3122,48
<b>Mínimo</b>	3495,12	3508,80	3527,25	3552,71	2964,88	2988,74	2888,04	2949,55
<b>25%</b>	4332,19	4333,20	4243,50	4246,18	3737,10	3736,66	3582,65	3628,58
<b>50%</b>	4488,06	45,07	4438,25	4470,25	3972,50	3979,66	3793,35	3803,62
<b>75%</b>	4741,00	47,66	4948,13	4938,43	4732,25	4682,23	3909,97	3910,21
<b>Máximo</b>	5601,37	5735,09	5726,63	5853,07	5876,88	5998,97	4593,78	4709,75
<i>Unidades em m<sup>3</sup></i>								

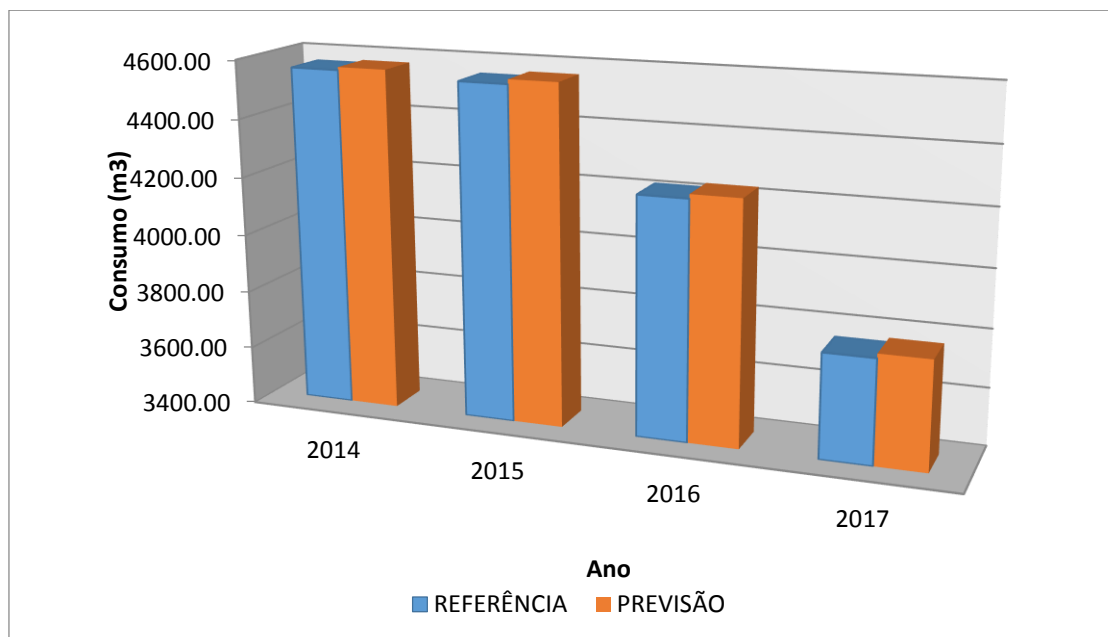


Figura 6.1 – Consumos diários médios anuais previstos face aos consumos de referência.

A análise estatística da Tabela 6.1 põe de manifesto uma boa semelhança dos valores previstos face aos valores de referência para cada ano. Tanto a média anual como o desvio padrão apresentam valores muito próximos, sendo que a média de um ano para outro apenas varia entre 10-20 m<sup>3</sup>, o qual supõe um erro aproximado do 0,35% no cálculo de consumos médios anuais (suposto um erro medio de 15 m<sup>3</sup> e consumos para um ano completo de aproximadamente 4300 m<sup>3</sup>, valor situado entre as médias de 2015 e 2016). Portanto, os resultados de previsão são muito favoráveis à escala. Mesmo assim, caberia dizer que os resultados de teste são sempre um pouco superiores aos dados de referência, tal como foi constatado na validação dos resultados. Na Figura 6.1 pode observar-se o paralelismo entre os valores correspondentes às médias de consumo diário anual, praticamente afins. O ano 2017 apresenta uma média menor porque a previsão apenas foi aplicada sobre os 4 primeiros meses do ano, correspondentes ao inverno e à primavera, razão pela qual se produz uma descida repentina (é nos meses de verão quando se produz um aumento da média. Pelo contrário, no ano 2014 apenas foram previstos 6 meses, desde julho a dezembro, portanto os meses mais quentes (julho-outubro) elevam a média em relação aos 2 meses mais frios desse período (novembro-dezembro).

Tabela 6.2 – Consumos mensais previstos face aos consumos de referência.

	PREVISÃO				REFERÊNCIA			
	2014	2015	2016	2017	2014	2015	2016	2017
JAN		4276,49	3829,84	3830,17		4262,50	3781,36	3823,63
FEV		4305,97	3692,00	3807,81		4306,98	3683,71	3800,97
MAR		4284,23	3723,12	3456,07		4255,49	3718,56	3411,54
ABR		4286,03	3674,03	3986,42		4293,32	3667,81	4016,53
MAI		4613,27	3840,46			4629,97	3832,30	
JUN		5069,79	4238,24			5035,36	4269,03	
JUL	4612,40	4859,60	5083,73		4647,75	4846,79	5086,51	
AGO	4986,10	5102,71	5372,63		4965,16	5097,55	5333,40	
SET	4808,80	5050,24	4950,64		4744,57	5044,47	4936,00	
OUT	4632,06	4642,90	4548,06		4626,62	4577,34	4501,66	
NOV	4383,90	4390,29	3841,69		4366,11	4347,75	3802,79	
DEC	4295,85	3892,65	3949,78		4288,94	3885,08	3953,75	

*Unidades em m<sup>3</sup>*

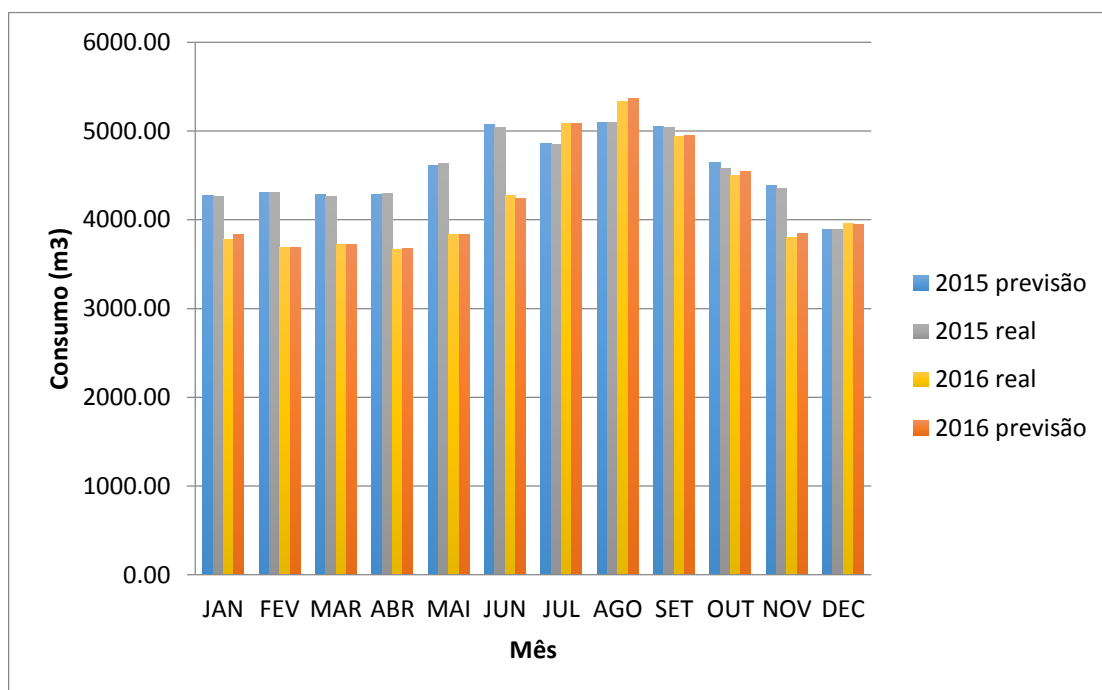


Figura 6.2 – Média dos consumos diários médios mensais previstos face aos consumos de referência.

No caso dos consumos cada mês, sucede algo da mesma índole; na Tabela 6.2 pode ser verificado como os valores de previsão são um bocado maiores que os dados de referência. Mesmo assim, da observação do gráfico de barras da Figura 6.2 pode ser verificado que as barras para um mesmo ano alcançam

praticamente valores análogos, portanto a previsão é adequada. Aliás, as tendências (subidas e descidas) ao longo do ano são aprendidas pela RNA de maneira apropriada, como pode ser constatado no gráfico.

Tabela 6.3 – Consumo diário médio semanal previsto face ao consumo de referência. Caso diário.

	REFERÊNCIA				PREVISÃO			
	2014	2015	2016	2017	2014	2015	2016	2017
<b>SEGUNDA-FEIRA</b>	4545,03	4477,45	4199,33	3796,11	4517,65	4504,48	4230,53	3835,00
<b>TERÇA-FEIRA</b>	4592,96	4540,92	4217,78	3708,53	4491,04	4500,09	4172,66	3631,35
<b>QUARTA-FEIRA</b>	4574,91	4535,21	4211,30	3687,80	4543,74	4553,52	4245,06	3756,20
<b>QUINTA-FEIRA</b>	4601,29	4528,08	4158,86	3708,91	4585,71	4585,00	4171,70	3706,16
<b>SEXTA-FEIRA</b>	4601,86	4541,22	4192,66	3748,36	4582,71	4549,31	4200,78	3776,71
<b>SÁBADO</b>	4724,62	4688,30	4360,95	3960,18	4715,75	4707,49	4385,27	3977,71
<b>DOMINGO</b>	4610,69	4532,86	4177,91	3713,63	4576,97	4555,89	4216,99	3746,95

*Unidades em m<sup>3</sup>*

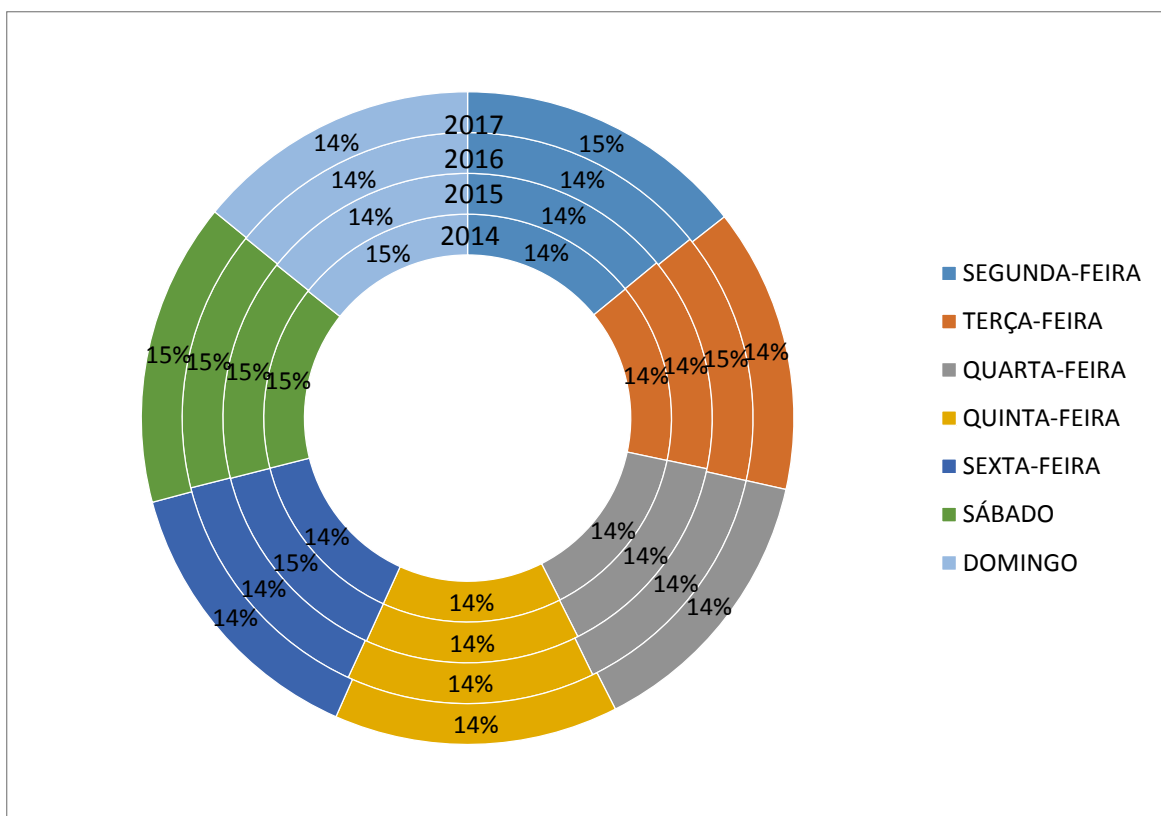


Figura 6.3 – Diagrama circular representativo do consumo diário médio semanal da série de referência.

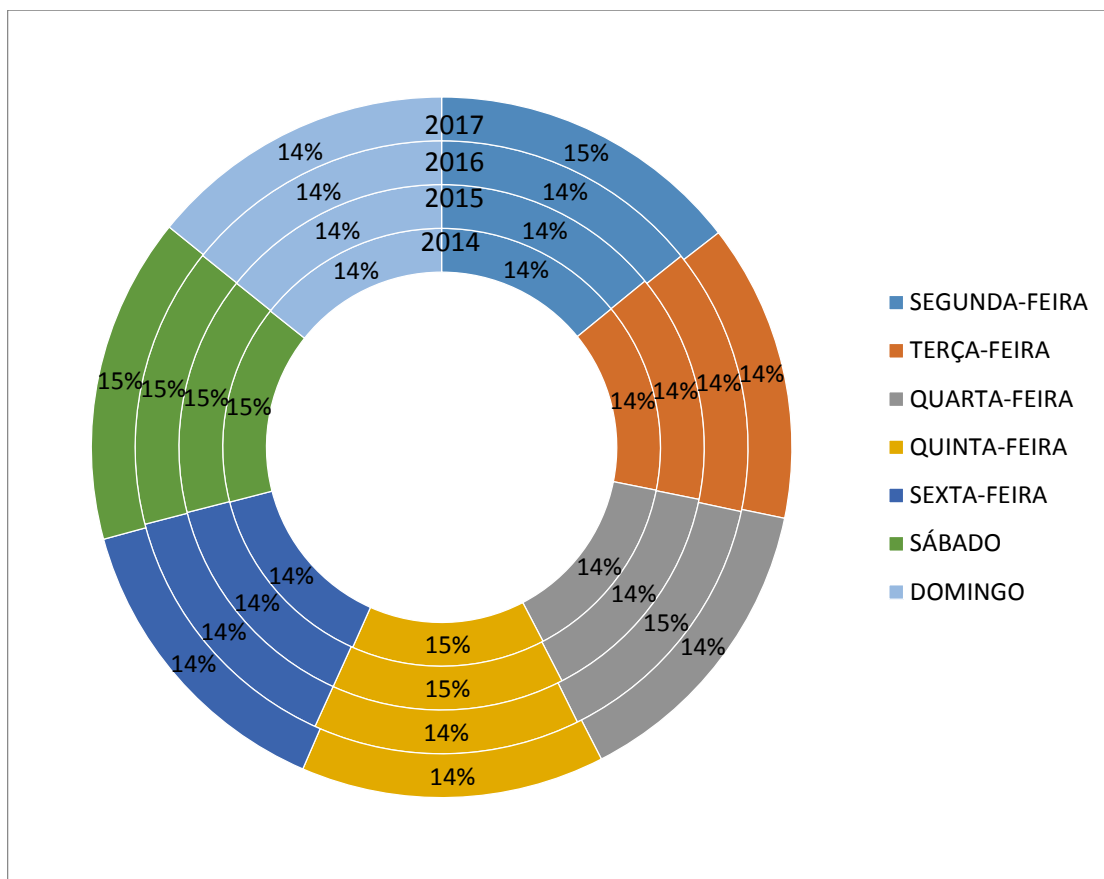


Figura 6.4 – Diagrama circular representativo do consumo diário médio semanal previsto.

Os valores de consumo diário médio semanal também são verificados com umas diferenças que oscilam entre os  $30-40 m^3$ , o qual não é considerado um erro significativo para os volumes que estão a ser estimados ( $3500-4500 m^3$ ). O valor de consumo médio que é verificado cada dia da semana é verificado mediante a comparação dos gráficos circulares da Figura 6.3 e da Figura 6.4 os consumos para qualquer dia da semana representam aproximadamente 14 % do consumo semanal, mas no sábado são verificados sempre os consumos mais elevados da semana, ainda que seja com um valor pouco superior: 15% do total semanal.

Por fim, é apresentada, na Figura 6.5, a série real completa e a previsão correspondente ao conjunto de teste. Uma vista mais detalhada é apresentada na Figura 6.6, onde pode notar-se a previsão correspondente a um período de tempo de vários dias.

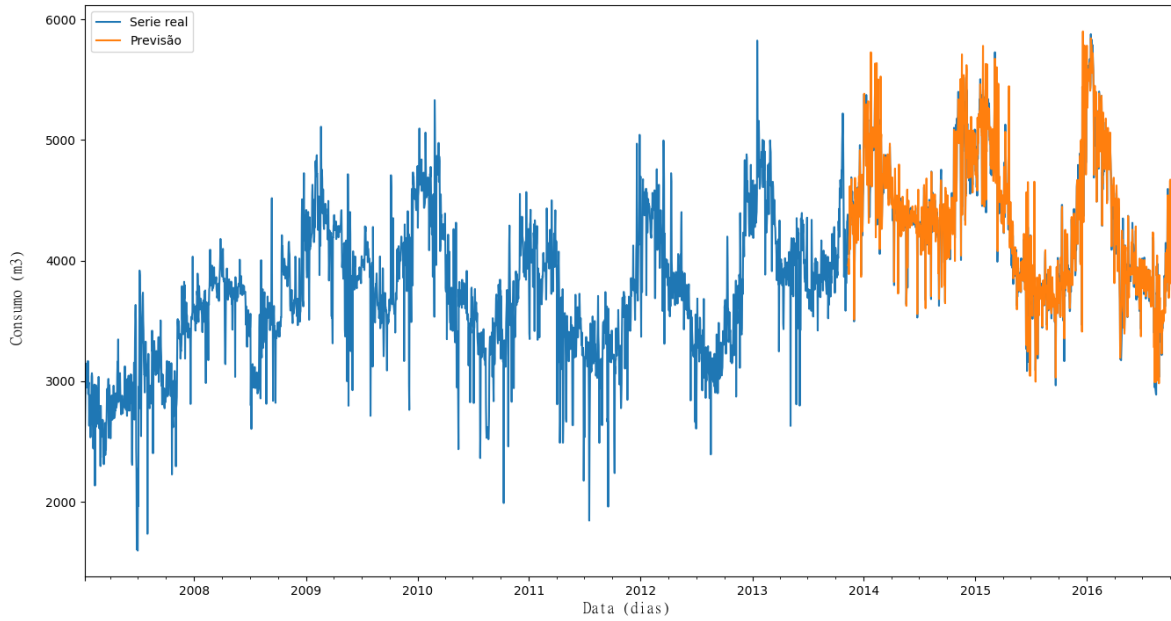


Figura 6.5 – Série temporal diária de referência e previsão aplicada ao conjunto de teste.

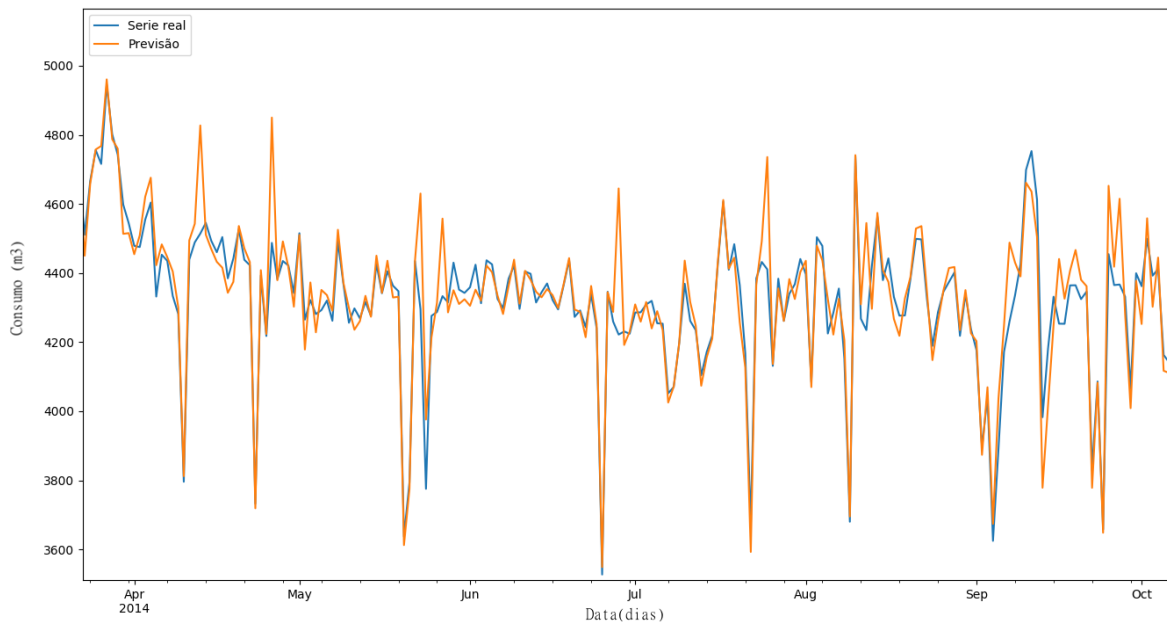


Figura 6.6 – Zoom da série temporal dos consumos diários de referência e previsão aplicada ao conjunto de teste.

Série horária

A série horária apresenta uma maior dificuldade em relação à capacidade de previsão de consumos, o qual já foi inferido na validação, ao apresentar um coeficiente de correlação menor do que no caso da série de consumos diários (o qual não significa que seja uma fraca correlação). A série completa e as previsões são apresentadas na Figura 6.7, e um aumento da mesma é mostrado na Figura 6.8.

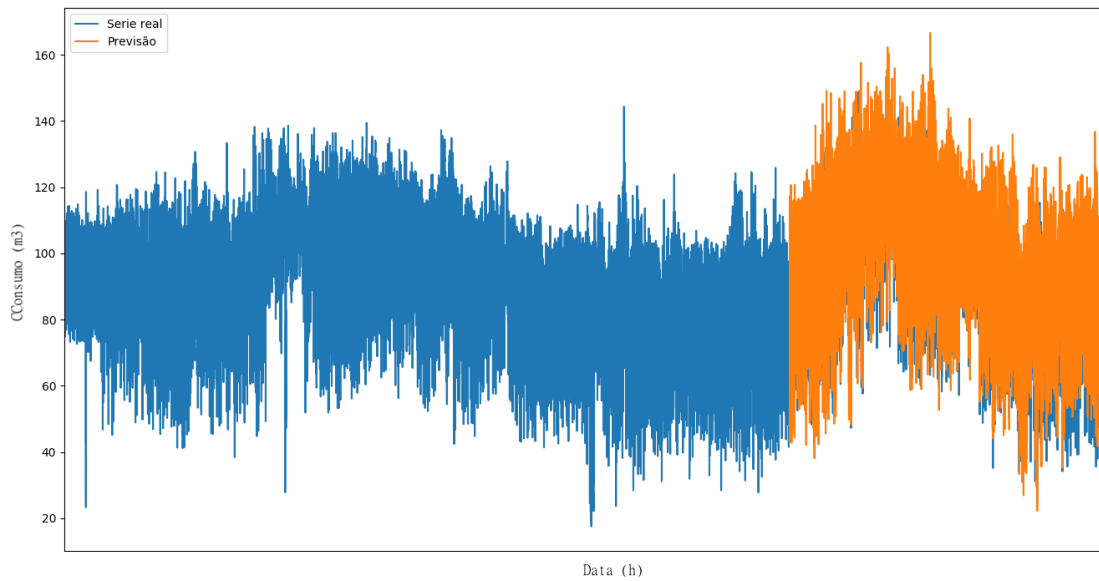


Figura 6.7 – Série temporal horária de referência e previsão aplicada ao conjunto de teste.

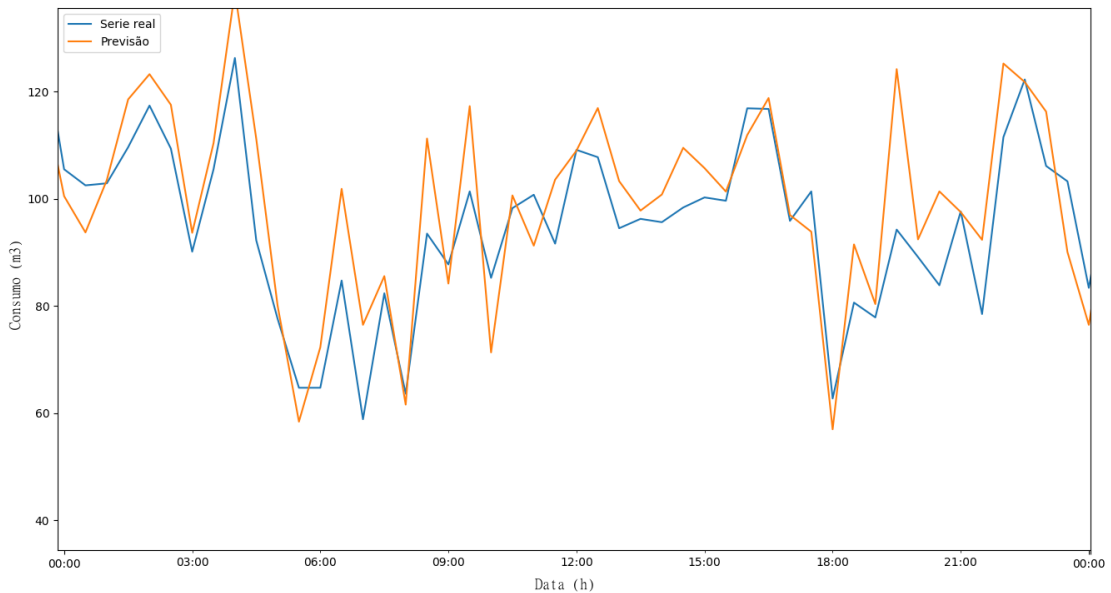


Figura 6.8 – Zoom da série temporal horária de referência e previsão aplicada ao conjunto de teste.

Por causa da grande quantidade de informação (48 dados por cada dia), considera-se melhor comentar os gráficos de consumo médio horário de referência e previsão para cada dia da semana, nomeadamente no período de estudo (último 30% da série, isto é, os últimos 2019 dias) em lugar de apresentar as extensas tabelas. Tal é representado na Figura 6.9.

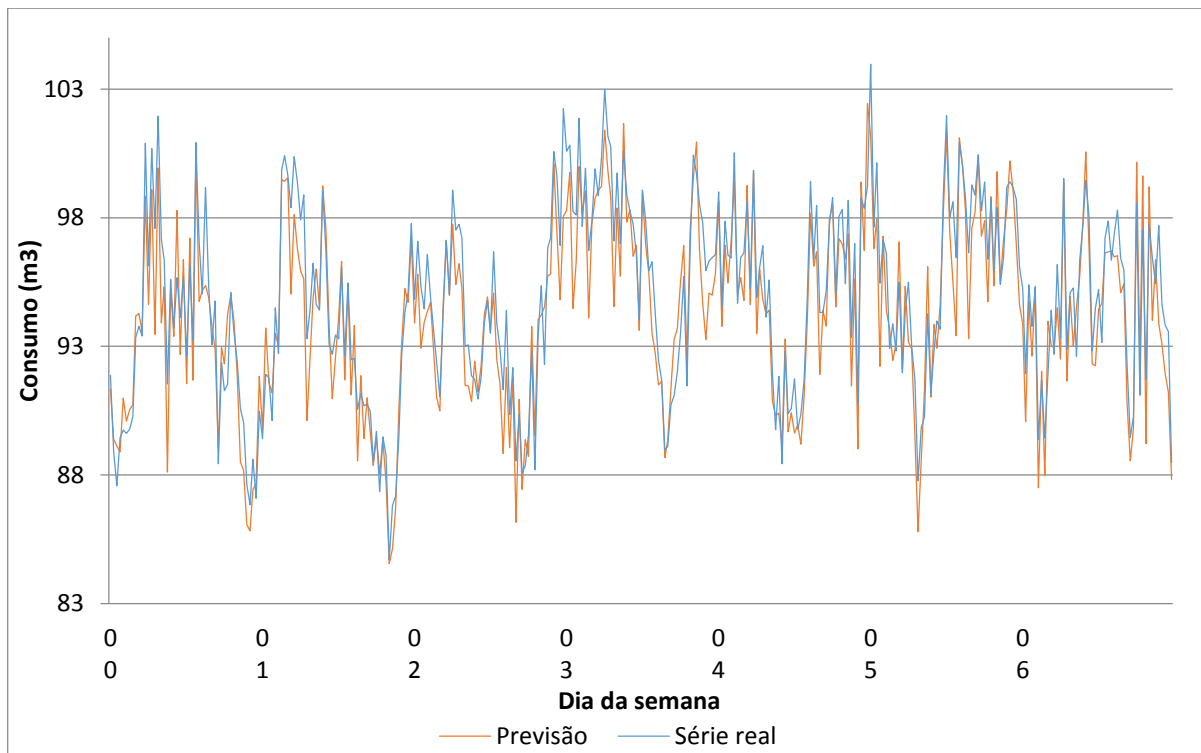


Figura 6.9 – Previsão da média horária dos consumos para cada dia da semana em comparação com a série de referência.

Neste caso, acontece o contrário do caso diário; a série está sensivelmente subestimada. Para o período de estudo não se verificam os padrões horários do anexo I, mas as tendências são respeitadas de uma forma bastante apropriada. Pelo contrário, a média dos consumos previstos é próxima dos  $92 \text{ m}^3$ , enquanto nos consumos horários registado, a média é aproximadamente de  $95 \text{ m}^3$ . Este pequeno desvio na média não é considerado como uma grande falha do sistema de previsão, o qual é bastante acertado.



# CAPÍTULO 7

## CONSIDERAÇÕES FINAIS

Neste último capítulo da presente dissertação serão expostas as conclusões retiradas da análise realizada, assim como serão referidos possíveis desenvolvimentos futuros propostos para dar continuidade ao trabalho atual.

### 7.1 CONCLUSÕES

No presente trabalho, foi desenvolvida uma metodologia para a criação de modelos que aproximem os consumos reais numa determinada população, mas tal metodologia tem carácter universal, na medida em que pode ser aplicada sobre quaisquer populações, desde a correção do modelo até a análise de resultados, sempre que o utilizador ajuste os parâmetros ao caso particular, isto é, ajustar os parâmetros da arquitetura da RNA e seja feita uma boa escolha dos parâmetros dos diferentes modelos ARIMA, tal e como esclarecido na própria metodologia.

Em trabalhos anteriores (Adamowski e Adamowski, 2015) foram reportados resultados satisfatórios mediante a utilização de RNA face à utilização de métodos matemáticos mais tradicionais (Alvisi *et al.*, 2007). Um dos objetivos do estudo atual é verificar se os métodos mais tradicionais são, efetivamente, menos adequados frente ao problema de previsão de séries temporais, face aos modelos de RNA. Assim, este trabalho procurou explorar algumas potencialidades singulares das RNA e, contribuir para o desenvolvimento desta matéria em Portugal no âmbito da engenharia civil.

Entende-se, deste modo, que o utilizador possui uma poderosa ferramenta para correção, criação, ajuste, previsão e avaliação do modelo, assim como análise estatística da série original de consumos e a consequente previsão.

Em vista dos resultados obtidos tanto com os modelos ARIMA como com os diferentes modelos de RNA, podem ser descritas algumas conclusões e comparações entre ambos tipos de modelo, destacando-se os seguintes aspetos:

- O modelo ARIMA produziu uns resultados bons para o caso diário, até melhor que em modelos mais clássicos (não autorregressivos, como o PMC não recorrente), o qual indica que tem um

funcionamento adequado para a previsão diária de consumos. Em contrapartida, os resultados obtidos para o caso dos consumos horários não foram aceitáveis, pois estes ficavam muito próximos da média, não tendo sido bem reproduzidos os máximos e mínimos.

- O modelo LSTM produziu uns resultados esplêndidos para o caso diário e bastante bons para o caso horário, existindo neste último caso uma diferença considerável em relação ao modelo ARIMA. Por tanto, um modelo de RNA bem definido resulta mais poderoso do que um modelo ARIMA, pois o modelo ARIMA aplica uma equação matemática aproximada, enquanto que a rede neuronal artificial realmente segue um processo de aprendizagem, com o qual aprende tendências sazonais, regulares, cíclicas, etc., sem necessidade de estudar em profundidade as relações de regressão da série, a exceção da definição de *delays* (mesmo assim, a diferenciação da série melhorou os resultados, pois é uma “ajuda extra” para a rede).
- Uma desvantagem dos modelos de RNA face aos modelos ARIMA é a grande quantidade de dados necessários e os tempos computacionais requeridos. Isto foi um fator muito limitador, pois para fazer qualquer tipo de modificação na arquitetura da RNA, foram necessárias esperas de entre 10 e 20 minutos de média, e até de horas para realizar algumas experiências. Isto fez com que o número de épocas, por exemplo no caso horário, fosse limitado a 10, de forma que os resultados poderiam ter sido ainda melhores se o tempo computacional não fosse tão elevado. Isto também limitou o número de nós por camada e número de camadas, entendendo que poderiam existir arquiteturas que melhorassem ainda mais os resultados, mas que requereriam muito mais tempo para ser ensaiadas e testadas. É por isso que o treino de uma RNA necessita de alguma experiência por parte de quem o executa. Por outro lado, computadores com uma maior memória podem realizar as mesmas tarefas em tempos mais reduzidos.
- Os modelos baseados em RNA são mais orientados a obter resultados do que a contribuir para a compreensão dos fenómenos, ao contrário do que acontece com os modelos estatísticos ou matemáticos. Aliás, a capacidade de generalização é limitada e resulta necessário um importante trabalho de análise dos resultados.
- No que concerne às diferenças entre os *softwares*, é conveniente assinalar as vantagens e desvantagens entre eles; a Neural Network Toolbox de Matlab possui um funcionamento simples ao ter uma interface gráfica e intuitiva. É relativamente rápido treinar as redes neuronais ao contar com algoritmos de treino que incorporam paragem antecipada, a diferença da rede definida na linguagem Python. Mas este facto limita, por acaso, os resultados, pois existem máximos locais no EQM de validação, o qual faz com que o treino pare. Não obstante, o EQM de validação poderia continuar a descer após esse mínimo local, com o qual se estaria a perder possibilidade de melhora do EQM. Isto não acontece no código Python, onde o utilizador define

o número de épocas e observa onde se produz o mínimo absoluto. Outro aspeto negativo da Neural Network Toolbox corresponde às limitadas possibilidades de modificar a arquitetura da rede: apenas podem ser modificados os *delays* e o número de nós por camada, enquanto no PyCharm podem ser definidos muitos mais fatores, como a possibilidade de incluir regularização para evitar *overfitting*, uma ampla escolha de algoritmos de treino (na ferramenta do Matlab esta escolha é limitada a 3 algoritmos), a escolha do medidor de error no sistema (EQM, REQM, erro médio absoluto, erro quadrático médio logarítmico, etc.), as funções de ativação, e outros parâmetros que definem a RNA.

- A regressão realizada com o modelo definido é aplicada para o caso dos consumos de água, mas também poderia ser aplicada a qualquer tipo de predição de séries temporais sejam ou não semelhantes.
- A qualidade dos dados disponibilizados não é muito boa, pois estes se encontravam cheios de falhas e valores anómalos devidos a erros no sistema de medição. Daí surgiu a necessidade de corrigi-los, mas a própria correção introduziu valores que não são reais e modificaram a aprendizagem da rede, reduzindo assim a possibilidade de realizar uma ótima estimativa. Poderia esperar-se, em consequência, que a aplicação da ferramenta de previsão sobre dados mais congruentes fosse capaz de produzir resultados ainda mais certos.
- Um dos maiores problemas que surgiram foi a necessidade de uma grande quantidade de dados para poder treinar a rede de maneira eficiente e que esta possa generalizar. A maior quantidade de dados, mais apropriados serão os resultados. Isto é uma das desvantagens das RNA face aos métodos estatísticos, que não precisam de um extenso número de dados.

## 7.2 DESENVOLVIMENTOS FUTUROS

Entende-se que há vários aspetos futuros que poderiam ser desenvolvidos em relação ao presente trabalho:

- Ligar a presente ferramenta de previsão de consumos com o software EPANET mediante o *package* para python EPANETTOOLS, de maneira a introduzir as previsões realizadas num sistema hidráulico definido previamente no EPANET. Para além disso, a ferramenta poderia ser implementada junto com outra ferramenta existente, precedente a este trabalho (DIAS GONÇALVES, 2016), na qual são definidos algoritmos genéticos para a otimização da paragem e funcionamento de bombas de água. Ambas ferramentas unidas podem constituir uma poderosa ferramenta de otimização dos consumos de energia.

- Existe uma interessante possibilidade de realizar previsões de consumo em tempo real, a qual poderia ser implementada mediante a obtenção, obviamente também em tempo real, das variáveis climáticas e dos próprios consumos. Esta tarefa tornar-se-ia mais fácil no caso de ter acesso direto aos dados duma empresa de abastecimento de águas ou mediante sistemas de medição próprios.
- Exploração de outros modelos de RNA dinâmicos, para além dos estudados nesta dissertação, e realizar experiências com outras configurações para além das exploradas neste trabalho.
- Implementação da validação *k-fold* para selecionar uma possível melhor definição dos conjuntos de treino, teste e validação.
- Considerar uma constante o *bias* para a correção das previsões. Normalmente, essa correção é realizada em base à média dos resíduos, a qual é somada às previsões para corrigi-las, de maneira que quando voltar a calcular a média residual, esta seja próxima de zero e, por tanto, a função de densidade seja aproximadamente uma função normal (centrada no zero), o qual indica uma distribuição aleatória.

## REFERÊNCIAS BIBLIOGRÁFICAS

ADAMOWSKI, J.; ADAMOWSKI, J. F. - Peak Daily Water Demand Forecast Modeling Using Artificial Neural Networks Using Artificial Neural Networks. , 9496(October), 337–346. doi: 10.1061/(ASCE)0733-9496(2008)134.

ÁGUAS DE DOURO E PAIVA S.A. - **Águas de Douro e Paiva, S.A.** [Em linha] [Consult. 24 abr. 2017]. Disponível em WWW:URL:<http://adnorte.addp.pt/pt/home.php>.

ALEJANDRO, C.; RODRÍGUEZ, E.; MARCO, B. - Análisis estadístico de la demanda de agua potable de la ciudad de Valencia . Aplicación a la predicción con análisis de series temporales Resumen. **Revista ingeniería del agua** . , (1).

ALVISI, S.; FRANCHINI, M.; MARINELLI, A. - A short-term, pattern-based model for water-demand forecasting. **Journal of Hydroinformatics**. ISSN 14647141 , 9(1), 39. doi: 10.2166/hydro.2006.016.

ARIZA RAMÍREZ, A. M. - Métodos Utilizados Para El Pronóstico De Demanda De Energía Eléctrica En Sistemas De Distribución. Universidad Tecnológica De Pereira. Programa De Ingeniería Eléctrica [Em linha] Disponível em WWW:URL:<http://recursosbiblioteca.utp.edu.co/dspace/bitstream/11059/3862/1/62131A719.pdf>.

BENÍTEZ SÁNCHEZ, J. M. - Aprendizaje de Sistemas Redes Neuronales Artificiales. Em . Granada : Universidad de Granada, 2001

BROWNLEE, J. - Deep Learning With Python.

CAMPOS, T. E. - **Vision - Computer Vision Research Group - IME/USP - [escience.ime.usp.br/vision](http://escience.ime.usp.br/vision)** [Em linha] [Consult. 22 abr. 2017]. Disponível em WWW:URL:<http://escience.ime.usp.br/vision/>.

DANNECKER, L. - Energy time series forecasting : efficient and accurate forecasting of evolving time series from the energy domain. [Em linha] [Consult. 24 jun. 2017]. Disponível em WWW:URL:<https://books.google.pt/books?id=ufhUCgAAQBAJ&pg=PA229&dq=Taylor,+J.W.,+de+Menezes,+L.M.+%26+McSharry,+P.E.+%282006%29+A+comparison+of+univariate+methods+for+forecasting+electricity+demand+up+to+a+day+ahead,+International+Journal+of+Forecasting.+%26+hl=es&sa=X&ISBN9783658110390>.

DAZA, F. - Demanda de agua en zonas urbanas en Andalucía. . [S.l.] : Universidad de Córdoba, 2008.

## REFERÊNCIAS BIBLIOGRÁFICAS

ISBN 9788478019281.

DECO PROTESTE - **DECO PROTESTE** [Em linha] [Consult. 24 abr. 2017]. Disponível em WWW:URL:<https://www.deco.proteste.pt/>.

**DGTerritório - Início** - [Em linha] [Consult. 16 jun. 2017]. Disponível em WWW:URL:<http://www.dgterritorio.pt/>.

DIAS GONÇALVES, A. M. - Otimização Energética da Operação de Sistemas de Abastecimento de Água em Alta.

ERSAR - **ERSAR-Entidade Reguladora dos Serviços de Águas e Resíduos** [Em linha] [Consult. 24 abr. 2017]. Disponível em WWW:URL:<http://www.ersar.pt/pt>.

FERNÁNDEZ JIMÉNEZ, D. - DEMANDA ELÉCTRICA HORARIA A MUY CORTO PLAZO : APLICACIÓN AL. . [S.I.] : Universidad Politécnica de Madrid (UPM)., 2016.

HAYKIN, S. - Neural networks: a comprehensive foundation. The Knowledge Engineering Review (Vol. 13) ISBN 0132733501.

HEATON, J. - Introduction to Neural Networks for JAVA. . [S.I.] : Heaton Research, 2005. (Second ed.) ISBN 1604390085.

LOPES, F. - Projeto de diplomação sistema de planeamento de irrigação baseado em redes narx. . [S.I.] : Universidade Federal Do Rio Grande do Sul, 2015.

MARTÍNEZ, L. - **Slideshare** [Em linha] [Consult. 16 jun. 2017]. Disponível em WWW:URL:<https://es.slideshare.net/lauratejedormartinez/consumo-de-agua-31055468>.

MATICH, D. J. - Redes Neuronales: Conceptos Básicos y Aplicaciones. Historia [Em linha]. [S.I.] : Universidad Tecnológica Nacional – Facultad Regional Rosario Departamento de Ingeniería Química, 2001. Disponível em WWW:URL:<ftp://decsai.ugr.es/pub/usuarios/castro/Material-Redes-Neuronales/Libros/matich-redesneuronales.pdf>.

MATOS, J. P. G. D. S. C. - Utilização de Redes Neurais Artificiais para a Modelação da Degradação de Sistemas de Distribuição de Água. . [S.I.] : Instituto Superior Técnico da Universidade de Lisboa, 2008.

NIELSEN, M. A. - **Neural Networks and Deep Learning** [Em linha]. [S.I.] : Determination Press, 2015 [Consult. 29 dez. 2016]. Disponível em WWW:URL:<http://neuralnetworksanddeeplearning.com/>.

PÉREZ, J. A. - MODELOS PREDICTIVOS BASADOS DE TIEMPO DISCRETO. . [S.I.] : Universidad de Alicante, 2002.

QUESADA P., M. - Análisis de Series. Modelos Heterocedásticos. , 87.

SAMARASINGHE, S. - Neural Networks for Applied Sciences and Engineering. . [S.I.] : Taylor & Francis

Group, 2006. ISBN 0-8493-3375-X.

**SNIRH :: Sistema Nacional de Informação de Recursos Hídricos** - [Em linha] [Consult. 16 jun. 2017].

Disponível em WWW:URL:<http://snirh.apambiente.pt/>.

SRINIVASA, L.; BRION, G. M. - Artificial Neural Networks in Water Supply Engineering. . [S.l.] : American Society of Civil Engineers, 2005. (10. ed.) ISBN 0-70844-0765-7.

THEODORIS, S.; KOUTROUMBAS, K. - Pattern Recognition. . Atenas : ELSEVIER ACADEMIC PRESS, 2003. (Segunda ed.)

UNED, E. Docente De E. I. - **Estimación de modelos ARIMA : Paro y empleo registrado**

US - **COEFICIENTE DE CORRELACIÓN LINEAL DE PEARSON** [Em linha] Disponível em

WWW:URL:<https://personal.us.es/vararey/adatos2/correlacion.pdf>.

ZARZA CANO, E. A. - Desarrollo de redes neuronales artificiales para el cálculo del transporte neoclásico en reactores de fusión.





## ANEXO I – ANÁLISE ESTATÍSTICA DOS DADOS

### CONSUMOS DIÁRIOS 2007-2017

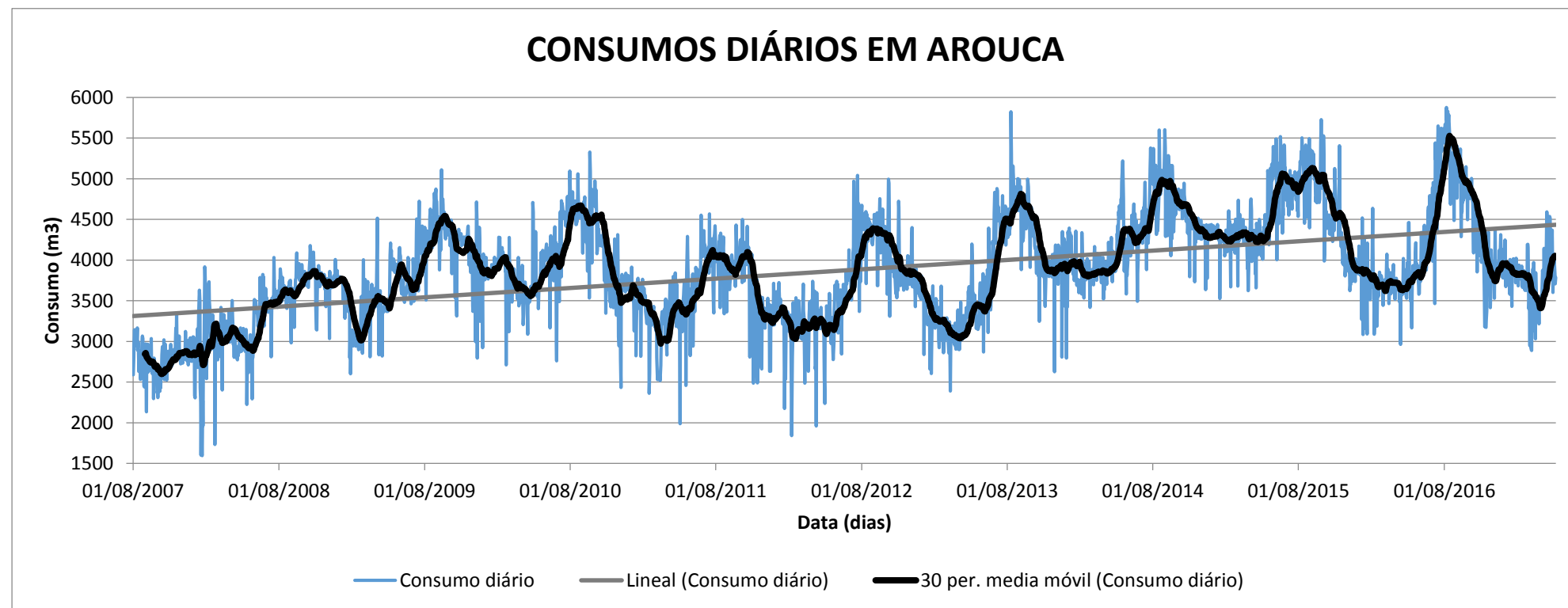


Figura A.1 – Série temporal do consumo de água diário do período 2007-2017

Tabela A.1 – Resumo dos principais parâmetros estatísticos da série temporal de consumos diários.

Unidades em m <sup>3</sup>	PERÍODO											Série completa
	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	
<b>Média</b>	2788,08	3381,03	3832,05	3998,66	3583,06	3671,19	3805,19	4325,73	4549,09	4219,42	3764,52	3873,97
<b>Desvio padrão</b>	190,83	440,06	479,57	461,43	442,67	528,78	620,27	441,83	457,64	654,11	313,58	643,46
<b>Mínimo</b>	2134,22	1594,86	2603,72	2435,69	1989,38	1843,56	2392,10	3419,81	3527,25	2964,88	2888,04	1594,86
<b>25%</b>	2679,83	3066,88	3512,72	3670,50	3319,31	3296,63	3264,93	3912,18	4243,50	3737,10	3611,30	3459,50
<b>50%</b>	2791,42	3479,00	3849,00	3951,56	3548,81	3650,50	3792,56	4324,87	4438,25	3972,50	3787,72	3838,35
<b>75%</b>	2915,23	3723,84	4210,78	4299,50	3931,50	4001,87	4311,00	4592,38	4948,13	4732,25	3910,02	4302,91
<b>Máximo</b>	3345,97	4179,09	5109,22	5329,81	4567,50	5042,07	5823,62	5601,37	5726,63	5876,88	4593,78	5876,88

Unidades em m<sup>3</sup>

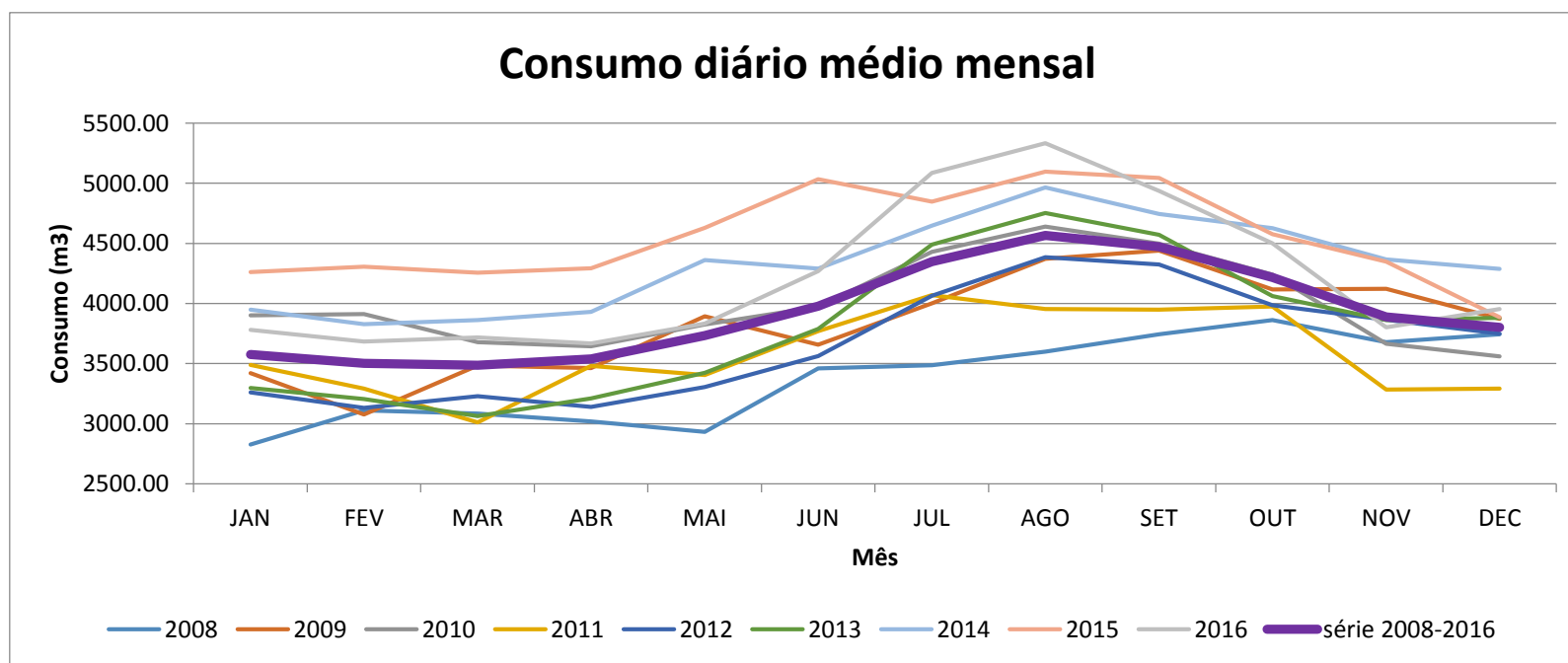


Figura A.2 – Consumo diário médio mensal.

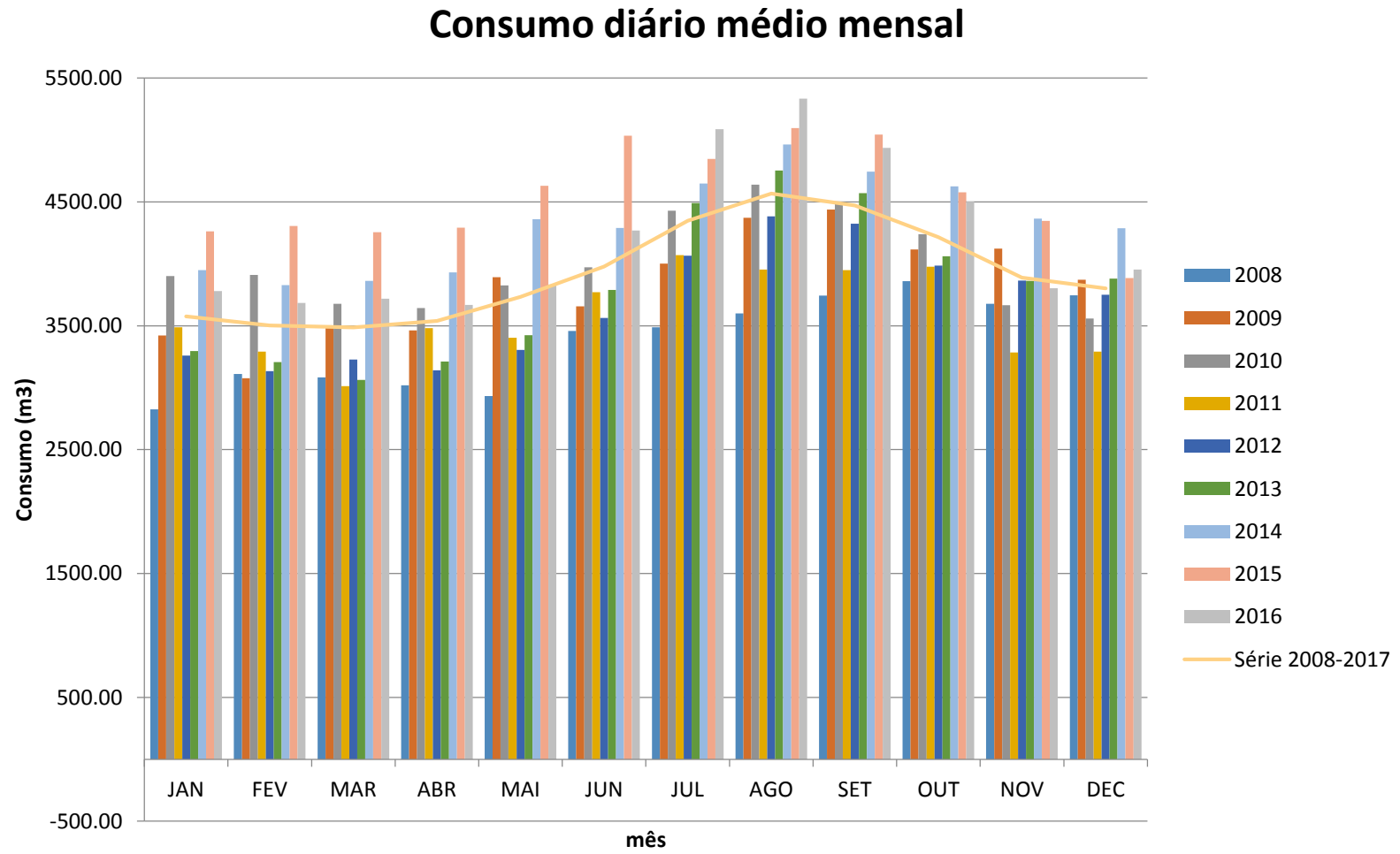


Figura A.3 – Consumo diário médio mensal da série diária. Gráfico de barras.

Tabela A.2 – Consumos diários médios mensais para cada ano.

<b>CONSUMO DIÁRIO MÉDIO MENSAL DE CADA ANO DA SÉRIE (m<sup>3</sup>)</b>												
	<b>2007</b>	<b>2008</b>	<b>2009</b>	<b>2010</b>	<b>2011</b>	<b>2012</b>	<b>2013</b>	<b>2014</b>	<b>2015</b>	<b>2016</b>	<b>2017</b>	<b>Série 2008-2017</b>
<b>JAN</b>	-	2826,56	3421,22	3901,95	3488,13	3260,63	3297,03	3950,07	4262,50	3781,36	3823,63	3576,60
<b>FEV</b>	-	3110,34	3077,62	3911,38	3292,03	3133,31	3206,09	3829,29	4306,98	3683,71	3800,97	3503,33
<b>MAR</b>	-	3083,86	3485,15	3678,30	3011,99	3227,97	3064,10	3861,82	4255,49	3718,56	3411,54	3487,47
<b>ABR</b>	-	3018,59	3463,53	3643,59	3481,04	3141,22	3210,58	3930,26	4293,32	3667,81	4016,53	3538,88
<b>MAI</b>	-	2932,01	3892,95	3825,65	3404,41	3304,25	3424,30	4361,32	4629,97	3832,30	3840,05	3734,13
<b>JUN</b>	-	3459,37	3658,07	3973,14	3770,81	3563,13	3788,66	4289,69	5035,36	4269,03	-	3978,58
<b>JUL</b>	-	3487,74	4001,14	4429,68	4069,90	4065,19	4491,21	4647,75	4846,79	5086,51	-	4347,32
<b>AGO</b>	2842,942581	3600,17	4372,08	4640,48	3953,32	4384,70	4753,24	4965,16	5097,55	5333,40	-	4566,68
<b>SET</b>	2690,633	3744,56	4439,75	4498,83	3948,63	4324,14	4571,51	4744,57	5044,47	4936,00		4472,49
<b>OUT</b>	2686,976452	3861,64	4116,24	4240,27	3976,35	3986,31	4061,01	4626,62	4577,34	4501,66	-	4216,38
<b>NOV</b>	2857,142	3677,47	4123,27	3665,40	3284,79	3865,20	3862,17	4366,11	4347,75	3802,79	-	3888,33
<b>DEC</b>	2861,788065	3745,19	3872,02	3559,88	3292,08	3750,05	3881,30	4288,94	3885,08	3953,75	-	3803,14

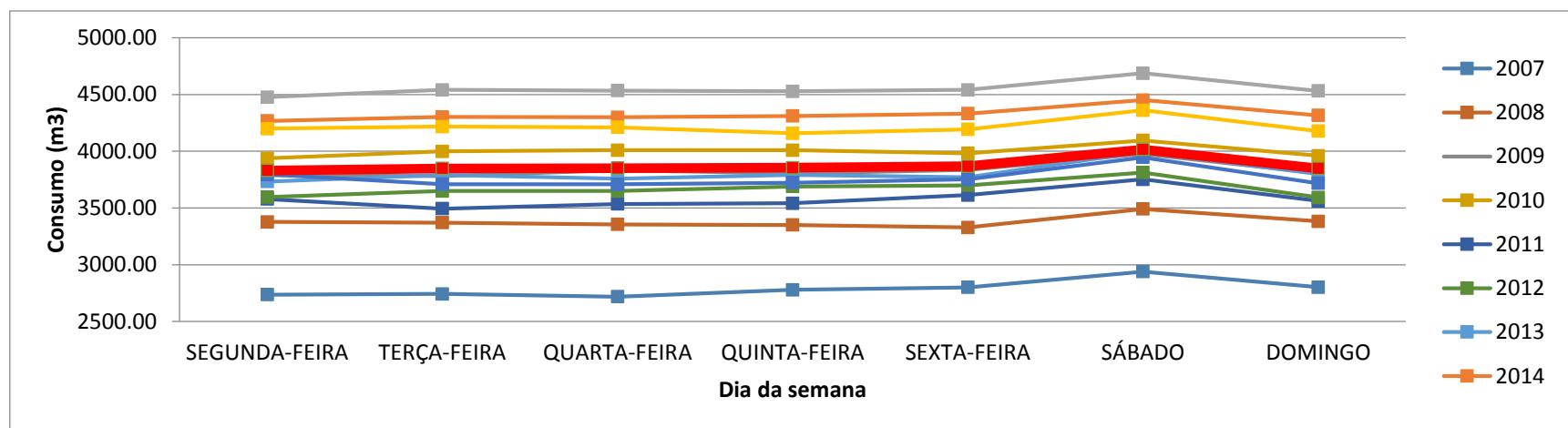


Figura A.4 – Consumo diário médio para cada dia da semana no período 2007-2017

Tabela A.3 – Consumo diário médio para cada dia da semana.

	PERÍODO										
	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
<b>SEGUNDA-FEIRA</b>	2735,87	3376,74	3788,45	3936,82	3579,24	3595,14	3733,49	4267,36	4477,45	4199,33	3795,64
<b>TERÇA-FEIRA</b>	2742,73	3370,40	3781,12	3998,54	3494,12	3649,03	3790,55	4302,90	4540,92	4217,78	3710,60
<b>QUARTA-FEIRA</b>	2717,67	3355,70	3832,46	4008,90	3535,29	3650,11	3757,19	4299,44	4535,21	4211,30	3709,12
<b>QUINTA-FEIRA</b>	2777,86	3350,80	3803,69	4009,12	3541,60	3688,79	3791,50	4310,90	4528,08	4158,86	3722,33
<b>SEXTA-FEIRA</b>	2799,79	3327,93	3834,03	3981,94	3613,50	3700,23	3770,47	4331,39	4541,22	4192,66	3752,82
<b>SÁBADO</b>	2939,64	3491,85	3977,17	4094,96	3752,73	3812,45	3994,83	4451,88	4688,30	4360,95	3946,26
<b>DOMINGO</b>	2800,94	3382,85	3807,94	3960,64	3561,68	3593,47	3798,61	4316,73	4532,86	4177,91	3717,47
<i>Unidades em m<sup>3</sup></i>											

Tabela A.4 – Resumo estatístico dos consumos diários para cada dia da semana.

DIA DA SEMANA	Média	Desvio padrão	Mínimo	25%	50%	75%	Máximo
Segunda-Feira	3830,482734	643,1562865	1594,86	3460,38	3801,75	4251,16	5755,75
Terça-Feira	3848,93376	646,073688	1843,56	3448,69	3826,25	4278,22	5830,88
Quarta-Feira	3850,703978	641,4552625	1963,88	3424,6075	3841,45	4286,453	5672,25
Quinta-Feira	3854,863779	623,7865526	2254,637166	3440,2225	3822,375	4271,28	5746,25
Sexta-Feira	3867,797468	639,7802528	1603,84	3469,55	3827,775	4297,875	5785,88
Sábado	4016,323607	636,3293147	2263,862166	3605,78	3962,235	4424,068	5876,88
Domingo	3848,520602	657,9286811	1606,89	3412,82	3800,42	4298,03	5726,63
<i>Unidades em m<sup>3</sup></i>							

## TEMPERATURA DIÁRIA 2012-2014

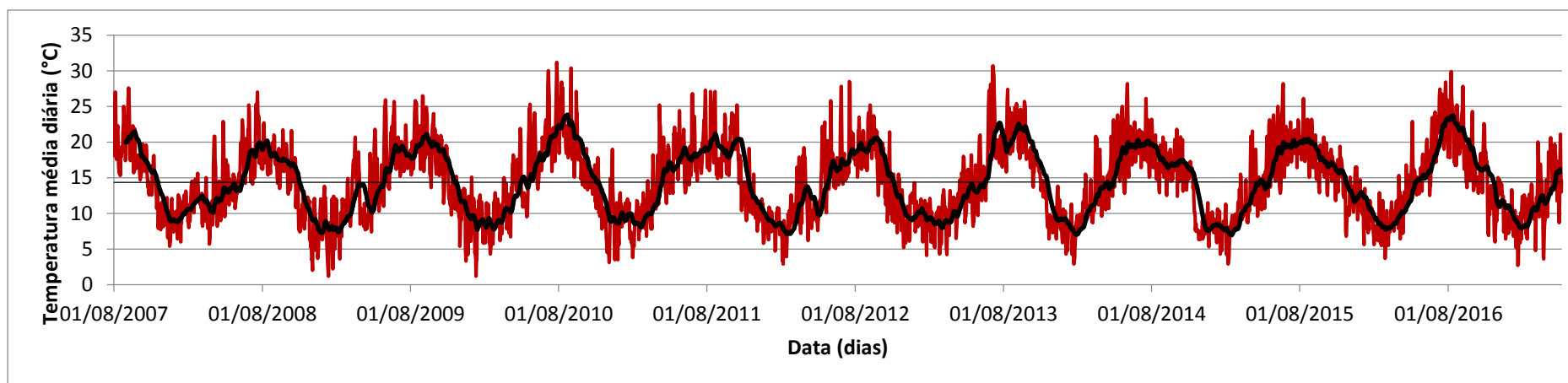


Figura A.5 – Temperatura diária média do período 2012-2014.

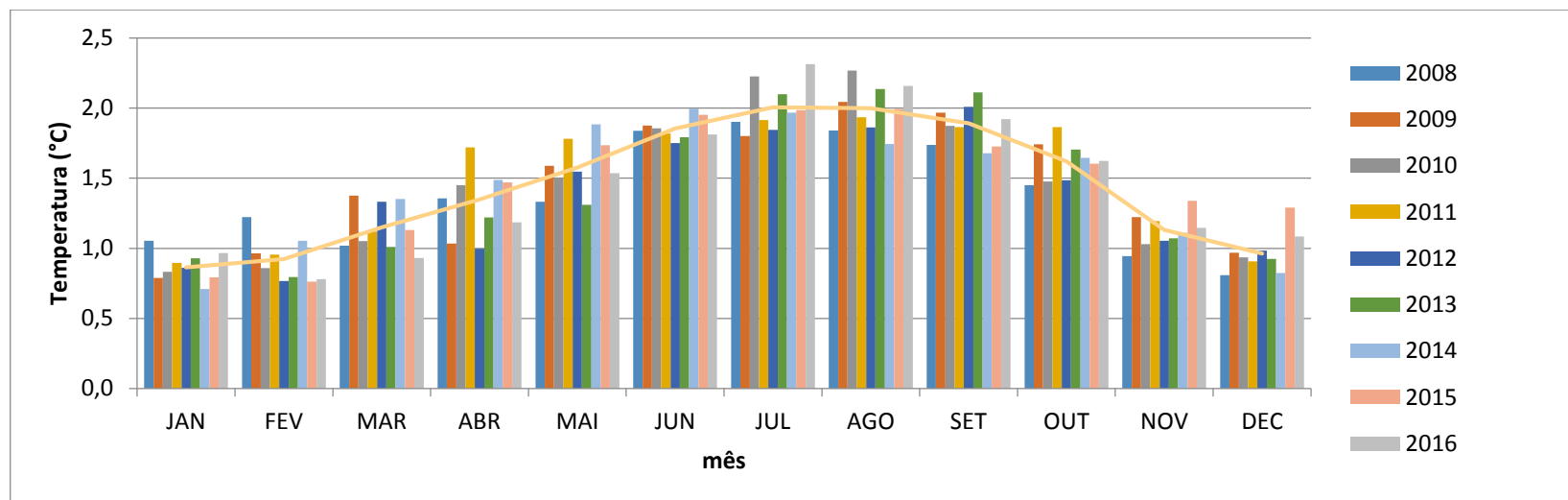


Figura A.6 – Temperatura diária mensal

Tabela A.5 – Temperatura diária média mensal.

Ano	Nº dados	Média	Desvio padrão	Mínimo	25%	50%	75%	Máximo
2007	153	15,5	5,2	5,4	10,7	15,9	18,8	27,6
2008	366	13,8	4,5	2,0	10,5	13,1	17,2	27,0
2009	365	14,5	5,2	1,2	10,5	14,8	18,6	26,5
2010	365	14,5	6,0	1,2	10,1	14,1	18,7	31,2
2011	365	15,0	5,2	3,8	10,7	15,3	18,8	27,3
2012	366	13,8	5,1	2,9	9,6	13,5	17,7	28,5
2013	365	14,3	5,7	4,1	9,6	13,4	18,2	30,7
2014	365	14,6	5,1	2,9	10,0	15,3	18,4	28,2
2015	365	14,9	4,9	2,9	10,8	15,1	18,5	28,2
2016	366	14,6	5,6	3,7	10,0	13,9	18,3	29,9
2017	127	11,8	4,1	2,7	9,3	11,4	14,2	21,1
<b>Série completa</b>	3568	14,4	5,3	1,2	10,2	14,2	18,1	31,2

*Unidades em °C*

Quadro A.6 – Temperatura diária média mensal.

Mês/ano	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	Série completa
JAN		10,5	7,9	8,3	9,0	8,6	9,3	7,1	7,9	9,7	8,2	8,6
FEV		12,2	9,6	8,6	9,6	7,7	8,0	10,5	7,6	7,8	10,7	9,2
MAR		10,2	13,8	10,5	11,2	13,3	10,1	13,5	11,3	9,3	11,7	11,5
ABR		13,6	10,3	14,5	17,2	10,0	12,2	14,9	14,7	11,9	15,8	13,5
MAI		13,3	15,9	15,1	17,8	15,5	13,1	18,9	17,4	15,4	14,7	15,8
JUN		18,4	18,8	18,6	18,2	17,5	17,9	20,0	19,5	18,1		18,5
JUL		19,0	18,0	22,3	19,1	18,4	21,0	19,7	19,9	23,2		20,1
AGO	20,0	18,4	20,4	22,7	19,4	18,6	21,4	17,4	19,9	21,6		20,0
SET	20,2	17,4	19,7	18,7	18,6	20,1	21,1	16,8	17,3	19,2		18,9
OUT	16,0	14,5	17,4	14,8	18,6	14,9	17,0	16,5	16,0	16,2		16,2
NOV	12,1	9,4	12,2	10,3	11,9	10,5	10,7	11,1	13,4	11,5		11,3
DEC	9,0	8,1	9,7	9,4	9,1	9,8	9,2	8,2	12,9	10,8		9,6

Unidades em °C

## CONSUMOS HORÁRIOS 2015-2016

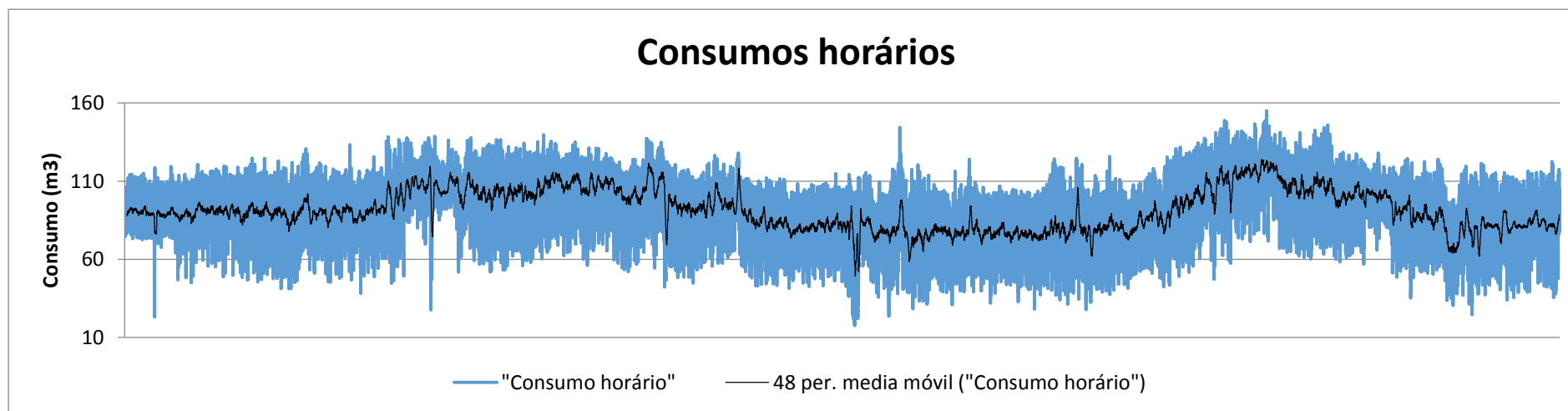


Figura A.7 – Temperatura média diária mensal.



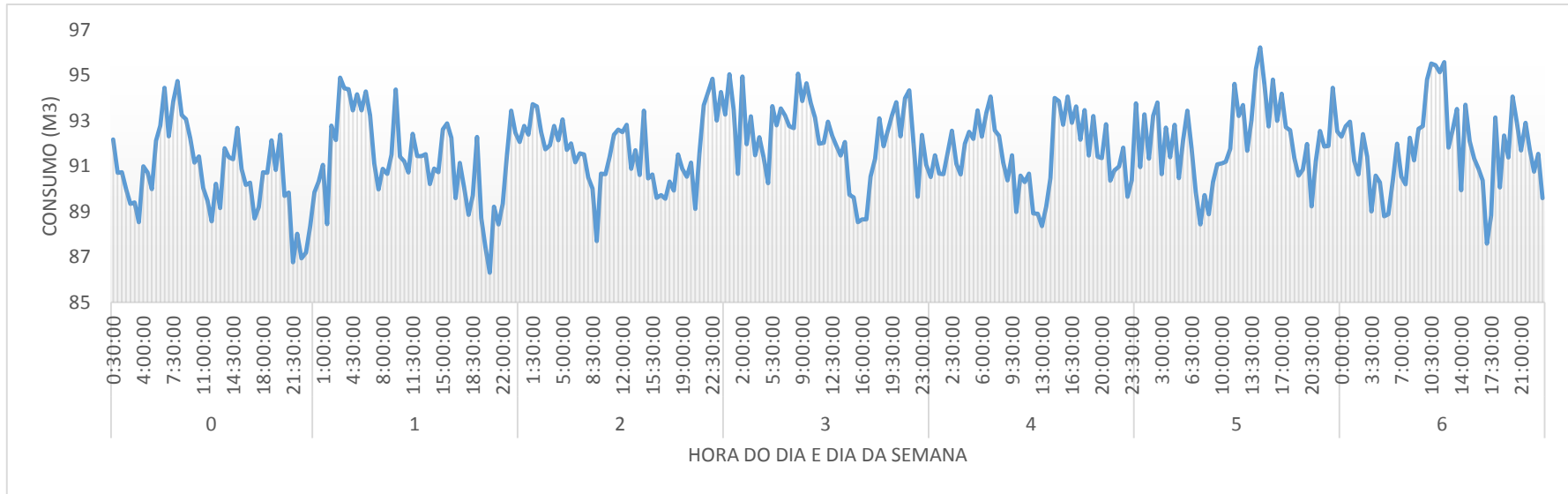


Figura A.8 – Consumo horário médio semanal.

▪ **EVOLUÇÃO CONJUNTA TEMPERATURA-CONSUMOS DE ÁGUA DIÁRIOS**

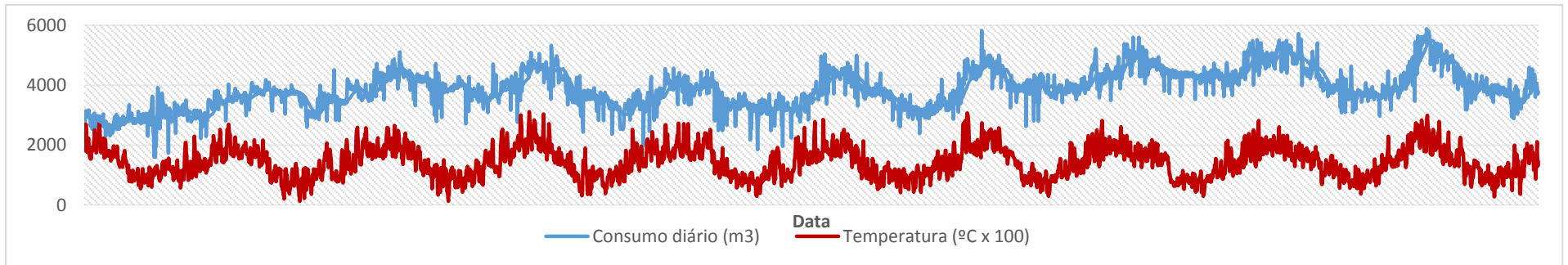


Figura A.9 Evolução conjunta da temperatura e o consumo de água diário.

## ANEXO II – RESULTADOS DOS TESTES EFETUADOS PARA A ESCOLHA DO MODELO DE RNA

Tabela A.7 – Resultados do treino do PMC não recorrente da série diária.

Número de camadas ocultas	Número de nós na camada oculta	Nº DE ÉPOCAS	EQM MÉDIO DE TESTE	R	TEMPO COMPUTACIONAL	PRESENÇA DE OVERFITTING
1	10	13	289.034,06	0,4439	0:01	NÃO
	20	14	290.610,87	0,4585	0:01	NÃO
	50	10	284.350,85	0,4922	0:02	NÃO
	80	12	437.357,54	0,3166	0:02	NÃO
	100	12	405.500,19	0,3916	0:02	NÃO
2	10	19	271.943,21	0,5032	0:05	NÃO
	20	17	299.712,34	0,4604	0:05	NÃO
	50	16	274.321,68	0,4976	0:06	SIM
	80	14	411.789,43	0,3537	0:06	SIM
	100	14	415.778,32	0,3508	0:09	SIM

Tabela A.8 – Resultados do treino da rede NARx da série diária.

Número de delays	Número de nós na camada oculta	Nº DE ÉPOCAS	EQM MÉDIO DE TESTE	R	TEMPO COMPUTACIONAL	PRESENÇA DE OVERFITTING
7	10	15	143.649,63	0,8422	0:01	NÃO
	20	12	128.126,01	0,8509	0:01	NÃO
	30	11	187.847,42	0,7851	0:02	NÃO
	50	17	192.751,81	0,7761	0:12	NÃO
	80	20	197.273,25	0,7503	0:19	NÃO
	100	9	208.326,03	0,6751	0:28	SIM
	15	13	126.844,12	0,821	0:01	NÃO

Tabela A.9 – Resultados do treino da rede LSTM da série diária.

Número de delays	Número de períodos para o cálculo da média móvel	Nº de camadas ocultas	Número de blocos de memória (cada um contém 2 células)	Nº DE ÉPOCAS	EQM MÉDIO DE TESTE	R	TEMPO COMPUTACIONAL	PRESENÇA DE OVERFITTING		
7	30	1	10	50	66.904,27	0,9879	10:58	NÃO		
			20	50	66.217,17	0,9821	10:22	NÃO		
			30	50	68.120,87	0,9895	11:04	NÃO		
			40	50	62.280,65	0,9834	10:55	NÃO		
			50	50	66.038,99	0,9893	10:46	NÃO		
			60	50	65.695,06	0,9883	12:06	NÃO		
			70	50	83.165,75	0,9998	12:27	NÃO		
			80	50	64.357,58	0,9888	11:51	NÃO		
			90	50	81.597,42	0,9997	11:31	NÃO		
			100	50	63.166,49	0,9821	10:28	NÃO		
		7	2	Camada 1	Camada 2					
				10	5	50	63886,01	0,9811	9:21	NÃO
				20	10	50	68732,9	0,9797	9:23	NÃO
				30	15	50	69279,28	0,9858	10:42	NÃO
	40			20	50	63149,82	0,9813	12:26	NÃO	
	50			25	50	65744,5	0,9897	17:19	NÃO	
	60			30	50	63791,54	0,9821	22:25	NÃO	
	70			35	50	62855,91	0,9837	14:35	NÃO	
	80			40	50	62795,84	0,9845	10:45	NÃO	
	90			45	50	67724,87	0,9901	12:40	NÃO	
	100			50	50	63436,24	0,9839	12:26	NÃO	
	7			1	10	50	89.948,16	0,9939	9:35	NÃO
					20	50	83.009,13	0,9896	9:53	NÃO
					30	50	85.147,91	0,9852	9:13	NÃO
		40	50		87.192,31	0,9866	9:21	NÃO		
		50	50		100.397,52	0,9985	10:21	NÃO		
		60	50		86.085,60	0,9871	9:43	NÃO		
		70	50		85.443,48	0,9896	9:49	NÃO		
80		50	97.577,24		0,9925	10:03	NÃO			
90		50	85.620,37		0,9901	10:11	NÃO			
100		50	87.446,12		0,9925	10:21	NÃO			
Teste nº de épocas	30	1	50	100	91.341,19	0,9884	19:13*	NÃO		

Tabela A.10 – Resultados do treino do PMC não recorrente da série horária.

Número de camadas ocultas	Número de nós na camada oculta	Nº DE ÉPOCAS	EQM MÉDIO DE TESTE	R	TEMPO COMPUTACIONAL	PRESENÇA DE OVERFITTING
1	10	1000	5,73	0,9926	0:10	SIM
	100	19	2,47E-03	0,9999	0:02	SIM
	500	10	54,56	0,9319	0:24	SIM
	1000	24	61,14	0,9204	3:45	SIM
	1300	12	109,06	0,8641	3:01	SIM
	1500	11	5,78	0,9918	3:52	SIM
2	10	978	6,32	0,9904	0:25	SIM
	100	28	0,0543	0,9997	0:09	SIM
	500	13	21,76	0,9632	0:35	SIM
	1000	19	89,54	0,8832	5:32	SIM
	1300	16	120,54	0,8317	5:35	SIM
	1500	15	5,43	0,9943	5:56	SIM

Tabela A.11 – Resultados do treino da rede NARx da série horária.

Número de delays	Número de nós na camada oculta	Nº DE ÉPOCAS	EQM MÉDIO DE TESTE	R	TEMPO COMPUTACIONAL	PRESENÇA DE OVERFITTING
7	10	22	155,68	0,7443	0:31	NÃO
	20	17	171,44	0,7218	3:00	NÃO
	50	16	203,39	0,6776	17:38	NÃO
	100	12	248,03	0,6347	52:40:00	SIM

Tabela A.12 – Resultados do treino da rede LSTM da série horária.

Número de <i>delays</i>	Número de camadas ocultas	Número de blocos de memória (cada um contém 2 células)	Nº DE ÉPOCAS	EQM MÉDIO DE TESTE	R	TEMPO COMPUTACIONAL	PRESENÇA DE OVERFITTING
48	1	10	10	203,11	0,9004	13:31	NÃO
		20	10	173,82	0,9509	13:33	NÃO
		30	10	166,76	0,9255	13:17	NÃO
		40	10	181,47	0,9537	13:00	NÃO
		50	10	172,93	0,9197	12:18	NÃO
		60	10	171,45	0,9019	12:34	NÃO
		70	10	168,45	0,9371	12:58	NÃO
		80	10	168,75	0,9335	14:14	NÃO
		90	10	178,33	0,9167	15:42	NÃO
	100	10	174,91	0,9259	15:47	NÃO	
	2	10.-5	10	170,19	0,9273	19:02	NÃO
		20.-10	10	191,47	0,9041	16:25	NÃO
		30-15	10	168,24	0,9422	19:14	NÃO
		40-20	10	174	0,9355	16:58	NÃO
		50-25	10	167,33	0,9319	19:57	NÃO
		60-30	10	188,71	0,9546	18:27	NÃO
		70-35	10	202,53	0,9072	48:00:00	NÃO
		80-40	10	173,19	0,9354	19:59	NÃO
		90-45	10	177,78	0,9355	21:09	NÃO
100-50		10	167,63	0,9242	23:33	NÃO	
Teste nº de épocas	1	10	100	179,22	0,9282	01:10:25*	NÃO

Unidades:

- EQM:  $m^6$
- REQM:  $m^3$
- Tempo computacional: h:min:s

## ANEXO III – FORMA DA FAC E DA FAP PARA A IDENTIFICAÇÃO DE MODELOS ARIMA

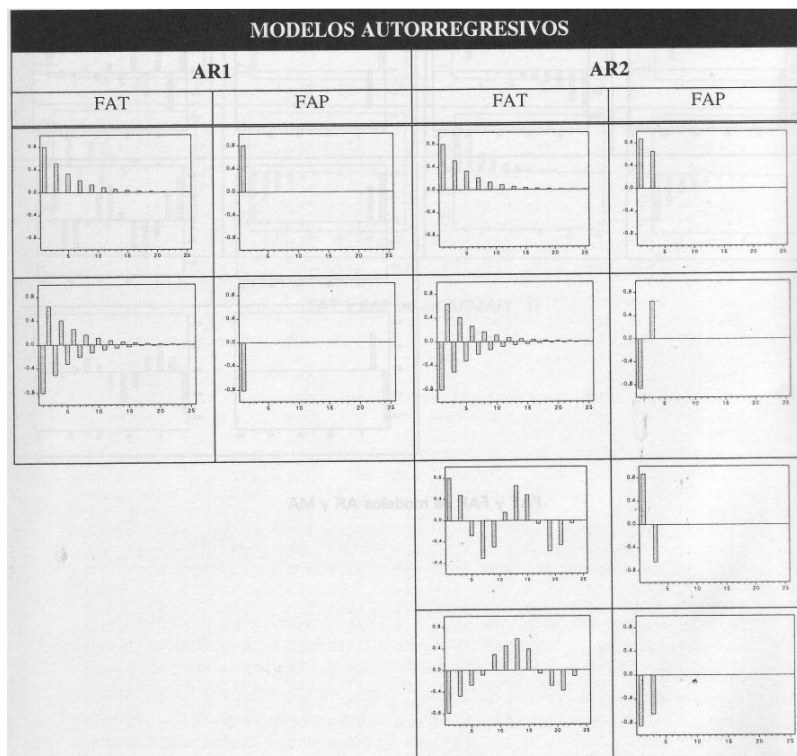


Figura A.10 – Gráficos representativos dos processos AR.

O modelo AR(1) ou  $ARMA(p,0)$  apresenta uma FAC que diminui rapidamente a zero, de forma regular, sinusoidal ou alternando valores positivos e negativos, junto com uma função de autocorrelação parcial com tantos valores distintos de zero como ordem do autorregressivo.

Um modelo  $ARMA(0,q)$  apresenta uma FAP que diminui rapidamente até zero, de forma regular, sinusoidal ou alternando valores positivos e negativos, junto com uma função de autocorrelação com tantos valores distintos de zero como ordem da média móvel.

Imágenes adaptadas de (UNED, 2015).

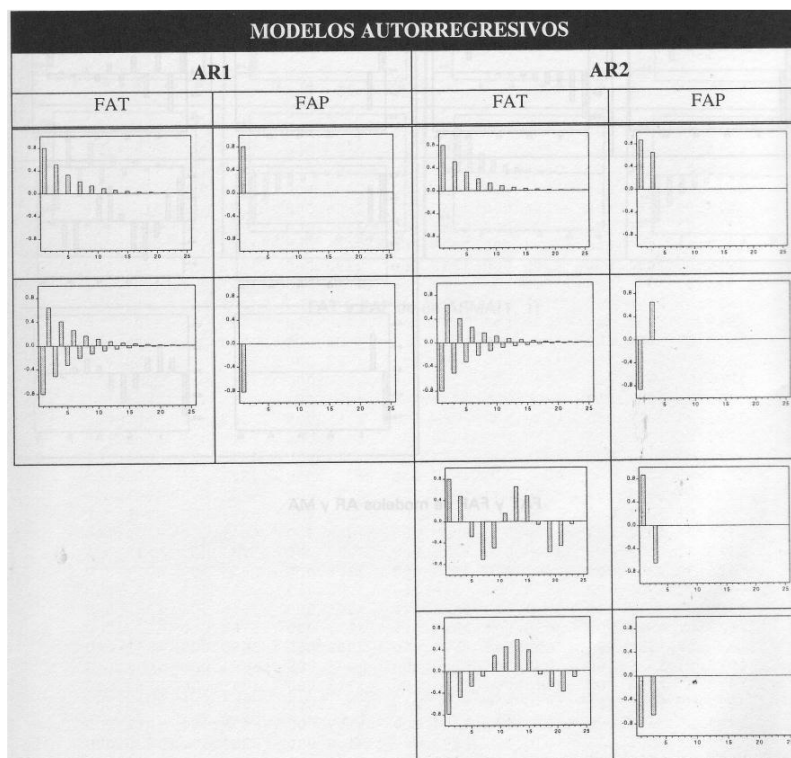


Figura A.11 – Gráficos representativos dos processos MA ou média móvel.

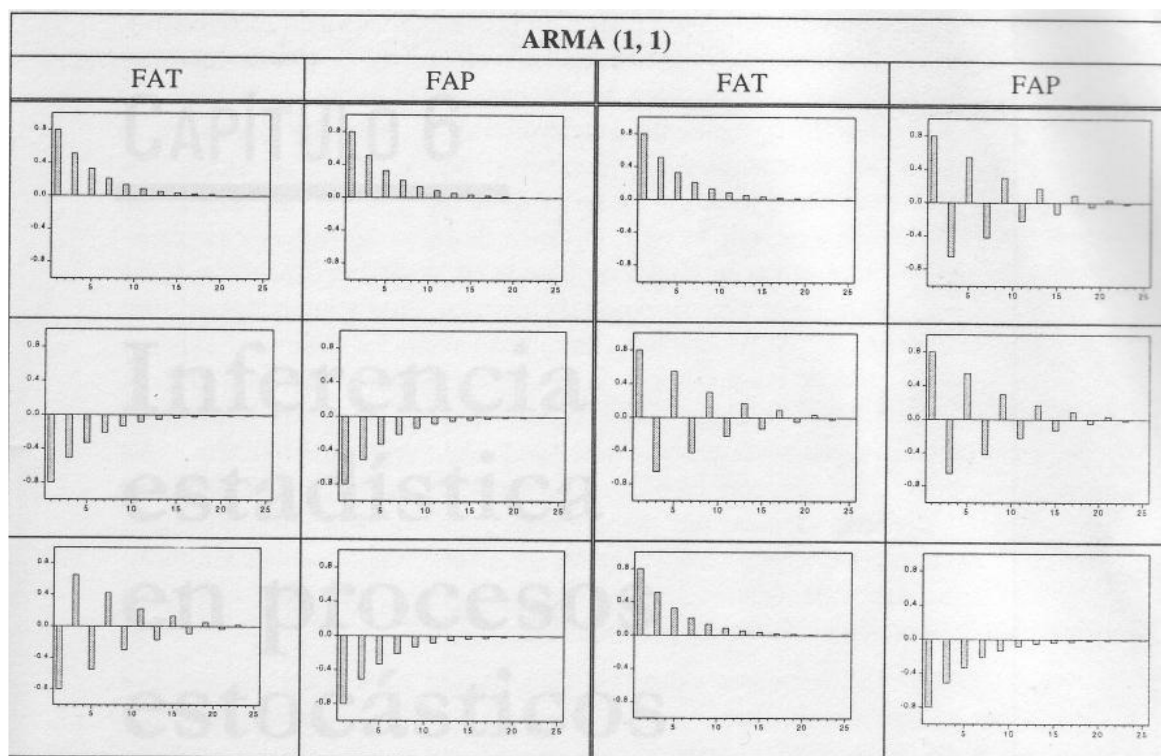


Figura A.12 – Gráficos representativos dos processos ARMA.