FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Classificação do desempenho energético de edifícios residenciais com base em algoritmos imunológicos

José Pedro Oliveira Martins da Silva Alves

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Orientador: Professor Doutor José Nuno Moura Marques Fidalgo

7 de Fevereiro de 2019

Resumo

A cada ano que passa, as preocupações ambientais ganham relevância na agenda política a nível mundial. A redução das emissões de gases de efeito de estufa e poupanças económicas são os principais temas visados nestas políticas.

A eficiência energética e o uso de energias renováveis são pilares fundamentais de políticas sustentáveis a longo prazo e, para atingir resultados eficazes, ambas as abordagens devem ser implementadas lado a lado. É essencial não só o investimento em novas fontes de energias renováveis, mas também requalificar estruturas já existentes e melhorar processos de implementação de estruturas ainda a serem desenvolvidas. Para ajudar a implementar estas políticas novas ferramentas capazes de lidar com dados já existentes são necessárias para apoiar e simplificar o processo de compreender os modelos reais e, com isto, implementar medidas eficazes.

Em edifícios residenciais, os equipamentos de conforto térmico e qualidade do ar representam na Europa mais de 50% do consumo energético total [1]. No processo de desenhar edifícios residenciais mais ecológicos e que e que consumam menos recursos são necessárias melhores ferramentas de simulação que de forma a estimar com precisão a sua eficiência energética.

Esta dissertação analisa a viabilidade de aplicar sistemas imunológicos artificiais (SIA) para identificar edifícios com características estruturais e de conjuntura semelhantes e se possível com alguma relação com o seu consumo energético.

Numa segunda fase irão são utilizadas redes neuronais artificiais (RNA) para estimar o consumo energético com aquecimento e arrefecimento. No final aplica-se o conceito de análise das sensibilidades para identificar quais das variáveis de entrada tem um maior impacto em ambos os consumos.

Palavras-chave: Algoritmos imunológicos artificiais, *Clustering*, Eficiência energética, Edifícios residenciais, Redes neuronais.

Abstract

With every passing year environmental concerns are increasing in the agenda of world politics. Reduction on the GHG emissions and economical savings are the main interest behind these policies.

Energy efficiency and renewable energy are the key pillars of sustainable energy policies and to achieve effective results, both approaches must be pursued side by side. Not only investment in new sources of clean energy production is essential but also the upgrading of existing structures and improvement of future facilities. In order to support these policies, new data handling tools are required to support and simplify the complex process of understanding real models and, in consequence, implement effective measures.

In residential buildings, the thermal comfort and air quality systems represent over 50% of the total building consumption [1]. In the process of designing more ecological residential buildings that consume less resources, complex simulation tools are required in order to estimate their thermal efficiency.

This paper describes the application of Artificial Immune Systems (AIS) to separate data that includes structural and conjuncture characteristics of residential buildings into groups with similar characteristics and hopefully with similar thermal efficiency.

In a second phase, Artificial Neural Networks (ANN) are used to estimate the buildings heating and cooling loads. A final sensitivity test is performed to identify the inputs with a larger impact on heating and cooling load.

Agradecimentos

Termina aqui, com esta dissertação, uma das etapas mais importantes do meu percurso académico. Como nenhum projeto desta dimensão se realiza sozinho, serve este espaço como um homenagem a todos os que nele participaram e ajudaram na sua conclusão.

Em primeiro lugar quero agradecer ao meus pais, pela motivação, pelo apoio e pela paciência, que se mostraram essenciais no meu sucesso académico. Também a toda a minha família por sempre estarem a meu lado e pela disponibilidade incondicional que demonstraram.

Agradeço a todos os meus amigos que sempre acreditaram em mim e tiveram sempre disponível palavras de incentivo mesmo nos momentos difíceis.

Finalmente quero agradecer ao meu orientador, Professor Doutor José Nuno Fidalgo por me ter aceite como orientando e por ter demonstrado sempre disponibilidade na partilha de conhecimento que se demonstrou preponderante no culminar desta dissertação.

A todos, um muito obrigado!

José Pedro Alves

"There is nothing noble in being superior to your fellow man.
True nobility is being superior to your former self"

Ernest Hemingway

Conteúdo

Lista de Figuras

Lista de Tabelas

Abreviaturas e Símbolos

AIS	Artificial immune systems
ANN	Artificial neural networks
GHG	Greenhouse Gas
IPCC	Intergovernmental Panel on Climate Change
MAPE	Mean Absolute Percentage Error
MSE	Mean Square Error
kWh	Quilo-Watt Hora
UCI	University of California, Irvine

Capítulo 1

Introdução

Este trabalho foi realizado no âmbito da dissertação relativa ao Mestrado Integrado em Engenharia Eletrotécnica e de Computadores, da Faculdade de Engenharia da Universidade do Porto (FEUP).

Ao longo desta secção introdutória irão ser discutidos o enquadramento e a motivação que estiveram na génese do tema desta dissertação, quais os objetivos que se esperam atingir e a estrutura do documento.

1.1 Motivação

Em 2013 um órgão sob a tutela das nações unidas apresentou um relatório chamado *"Fifth Assessment Report"*. Nele o *IPCC* conclui com 95% de certeza que a atividade humana, devido ao aumento da concentração de gases de efeito de estufa, foi a causa dominante do aquecimento verificado no clima desde meados do século XX [2].

Dada a interdependência entre os países no mundo atual, os impactos das alterações climáticas terão certamente influência nos recursos globais e será muito provável que os estes impactos se irão manifestar a nível global com consequências nos preços, cadeias de fornecimento, investimento, e relações políticas.

A eficiência energética e as energias renováveis são os pilares essenciais de políticas energéticas sustentáveis e para se obterem resultados eficazes, ambas as abordagens devem ser aplicadas lado a lado. Não só o investimento em novas abordagens relativamente à produção baseada em energias limpas é essencial, mas também a requalificação de estruturas existentes e a melhoria de futuras instalações.

No que toca a instalações residenciais, o consumo energético com aquecimento e arrefecimento pode representar uma fatia significativa do consumo total como se pode constatar na Figura 1.1.

De forma a apoiar estas políticas, devem existir ferramentas que apoiem e simplifiquem o complexo processo de caracterização de modelos reais e implementação de medidas eficazes.

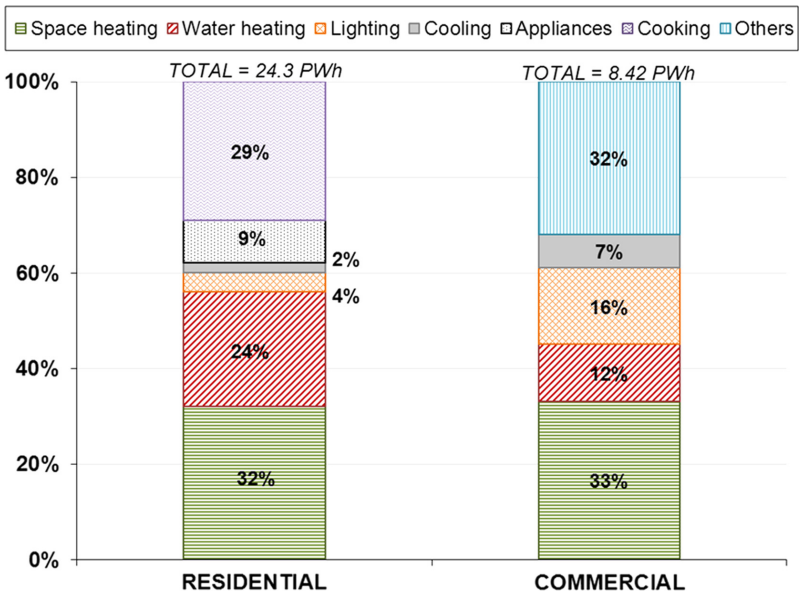


Figura 1.1: Consumo energético de edifícios residenciais [1]

1.2 Objetivos

No âmbito desta dissertação, o autor pretende, a partir de um conjunto complexo de dados onde se incluem características estruturais e de conjuntura de edifícios residenciais, simplificar a sua classificação energética recorrendo à implementação um algoritmo de *clustering* baseado em sistemas imunológicos. Este algoritmo deve ser capaz de procurar características semelhantes nos edifícios, que permitam a sua classificação em grupos com consumos energéticos semelhantes, e simplificar o processo de classificação energética de um edifício residencial.

Para além do algoritmo de *clustering*, foram treinadas diversas redes neuronais capazes de prever, a partir das características dos edifícios, o valor dos consumos energéticos. Baseado nestas previsões, foi implementado um outro algoritmo capaz de, a partir destas redes neuronais, calcular as sensibilidades das saídas relativamente às entradas. Este procedimento permite oferecer uma indicação ao utilizador de quais as variáveis mais importantes no que toca aos consumos energéticos e, por atuação naquelas que possível, melhorar a eficiência energética das habitações.

A abordagem utilizando algoritmos imunológicos pretende simplificar o processo de construção de um edifício ecológico e eficiente, utilizando ferramentas computacionais simples e obtendo resultados com uma precisão aceitável em pouco tempo.

1.3 Estrutura da Dissertação

Para além da introdução, esta dissertação contém mais 4 capítulos. No capítulo 2, é descrito o estado da arte e são apresentados trabalhos relacionados. No capítulo 3, é descrita a abordagem

geral do trabalho, o software utilizado, a descrição e organização dos dados e a metodologia implementada. No capítulo 4 faz-se uma descrição dos resultados obtidos. O capítulo 5 é composto pelas conclusões e sugestões para trabalhos futuros.

Capítulo 2

Revisão Bibliográfica

Neste capítulo é feita uma exposição acerca de implementações semelhantes de trabalhos com temas próximos da área de estudo, e também de alguns métodos e ferramentas utilizados ao longo desta dissertação. Em primeiro lugar será feita uma abordagem básica acerca dos objetivos do processo de *clustering* e também acerca de sistemas imunológicos e a sua base na implementação de algoritmos inspirados neste sistema. Finalmente serão abordadas diversas técnicas que auxiliaram o processo de construção do algoritmo.

2.1 Trabalhos prévios

O comportamento energético de edifícios foi alvo de diversos estudos ao longo do tempo, com análises que envolveram variados objetivos e técnicas. Com o trabalho de Pablo Bermejo et al., foi criado um sistema de controlo térmico de um edifício capaz de se adaptar às preferências dos ocupantes utilizando um sistema de aprendizagem baseada em lógica *fuzzy* [3]. Jin Woo Moon et al. criaram um método de controlo térmico em edifícios residenciais baseado em ANN [4].

Também a aplicação de AIS já viu desenvolvimentos extensos com aplicações em outros campos como a eletrotecnia com a deteção de distúrbios de tensão [5], no cálculo de *powerflows* em redes elétricas [6], problemas de pré-despacho (ou *unit commitment*) [7] e mecanismos automáticos de correção do fator de potência em redes elétricas [8].

No campo da informática e inteligência computacional trabalhos como [9], [10], [11] e [12] demonstram a versatilidade dos AIS em problemas de classificação e de otimização.

No campo da aplicação de AIS a edifícios temos o trabalho de F. Parra et al., com a aplicação de algoritmos de seleção negativa à análise da integridade estrutural de edifícios [13]. Com o trabalho de Jiawei Zhu et al., foi realizada uma análise do conforto térmico de edifícios residenciais utilizando sistemas imunológicos artificiais [14].

A pesquisa bibliográfica não permitiu encontrar trabalhos semelhantes ao proposto nesta dissertação: usar AIS para classificação da eficiência térmica de edifícios.

2.2 Ferramentas e métodos

2.2.1 Clustering

No campo da análise de dados a utilização de um tipo de algoritmos que se adapte ao tipo de dados é essencial. De entre os tipos de algoritmos existentes destacam-se dois grandes grupos, os supervisionados e os não supervisionados. Algoritmos supervisionados são implementados com a função de, a partir de um conjunto de entradas e respetivos valores de resposta (saídas), encontrar uma função que faça a correspondência entre os dois. Por oposição, uma aprendizagem não supervisionada deve procurar relações entre os dados disponíveis e inferir padrões que ajudem a classificar a informação em grupos [15].

Clustering é uma técnica de aprendizagem não supervisionada que procura encontrar uma estrutura num conjunto de dados não categorizados, e organizar os elementos em grupos que partilhem características semelhantes. Uma boa implementação de um algoritmo de *clustering* deve atender a uma série de requisitos como [16]:

1. Ser escalável para grandes conjuntos de dados;
2. Ter capacidade em lidar com diferentes tipos de atributos;
3. Ter capacidade em lidar com ruído e *outliers*;
4. Produzir resultados semelhantes independentemente da ordem de entrada dos dados;
5. Ser capaz de lidar com dados com grande número de características (dimensões elevadas);
6. Produzir conclusões simples e resultados fáceis de interpretar.

Ao longo deste trabalho estes requisitos vão servir como base para melhorar e otimizar o funcionamento do algoritmo implementado.

2.2.2 Sistemas imunológicos

Os sistemas imunológicos são mecanismos característicos de seres vivos de deteção e eliminação, contra agentes externos infeciosos. Este comportamento permite a diferenciação entre as células próprias, que permitem o funcionamento natural do organismo, e não próprias que podem ser potencialmente prejudiciais [17]. Existem duas estratégias principais que funcionam lado a lado e que permitem lidar com estes patogénicos:

- Sistema imune inato;
- Sistema imune adquirido.

O sistema inato é o mais geral dos dois e permite ao organismo responder de maneira genérica a infeções através da ativação de células especializadas no combate a patogénicos. Este sistema tem também a capacidade de ativar o sistema imune adquirido, que após a resposta inicial do

sistema inato, vai criar uma memória da infeção que vai permitir uma resposta mais rápida e eficaz num próximo encontro com o mesmo agente infecioso.

Partindo do comportamento dos mecanismos do sistema imunológico, uma classe de de algoritmos imuno-inspirados foram desenvolvidos, tendo aparecido como uma área de estudo no campo da inteligência computacional [18], com as primeiras aplicações a surgirem por volta dos anos 90 [19].

Os sistemas imunológicos artificiais modernos, ou *artificial immune systems* na literatura inglesa, são tipicamente baseados em um dos três processos principais do sistema imunológico [20].

- Redes imunológicas;
- Seleção clonal;
- Seleção negativa.

O algoritmo implementado obteve as suas bases numa teoria que tenta explicar o funcionamento deste sistema adquirido, o mecanismo de tolerância central. Este mecanismo, também conhecido como seleção negativa, permite ao organismo reconhecer e eliminar células do sistema imunitário que reajam com células próprias do organismo, garantindo assim que estas células imunitárias ataquem apenas agentes externos e nunca o próprio corpo. Este processo ocorre durante a fase de maturação destas células, designadas por linfócitos [5].

O procedimento geral que descreve a implementação de um algoritmo de seleção negativa pode ser dividido em duas partes, e começa pela definição de um conjunto de vetores que representam o funcionamento normal do sistema, e que devem ser protegidos [21]. Nesta fase são então testados os vetores disponíveis e a sua afinidade com os vetores próprios são avaliados. Caso a afinidade seja superior a um limiar o vetor é rejeitado, caso contrário é armazenado num conjunto de detetores. Por outras palavras, guarda o novo detetor, se este for suficientemente distinto dos já existentes.

A segunda parte corresponde à monitorização onde é avaliada a afinidade entre cada uma das cadeias próprias e os detetores. Se afinidade for superior a um limite definido, então é identificado um elemento não próprio[22].

2.2.3 Redes neuronais

As redes neuronais constituem um tema já muito comum na literatura científica, pelo que não se aprofunda aqui a descrição desta técnica, sendo apenas relembrados alguns conceitos gerais.

Tal como no caso dos algoritmos imunológicos, as redes neuronais artificiais ou *artificial neural networks* (ANN) na literatura inglesa, inspiram-se no comportamento de processos e sistemas da biologia, neste caso o funcionamento do sistema nervoso central humano. Esta estrutura é composta por unidades, conectadas entre si, cujas ligações podem ou não ser ativadas, resultando numa grande capacidade de processamento paralelo.

AAs redes neuronais artificiais são então capazes de realizar aprendizagem baseada em exemplos (conjunto de treino), num processo supervisionado. Depois de treinada a rede, é possível aplicar os modelos encontrados a novos casos [23].

Estas redes são compostas por neurónios e as respetivas ligações (sinapses) entre eles. Cada conexão transmite o resultado das saídas de um neurónio para outro ou outros, permitindo um processamento de informação paralelo.

Optou-se por este método para realizar as previsões por já ser uma técnica extensivamente estudada e com provas dadas em termos de capacidade de aprendizagem e boa performance.

Capítulo 3

Metodologia

3.1 Abordagem geral

Pretende-se numa primeira fase caracterizar as circunstâncias que condicionam o desempenho energético de edifícios. Para isso são utilizados algoritmos de clustering, de modo a identificar conjuntos de edifícios homogéneos e, a partir daí, relacionar estes conjuntos com os consumos energéticos. Numa segunda fase, pretende-se estabelecer um procedimento para estimar o consumo energético de edifícios a partir das suas características estruturais.

As metodologias e algoritmos desenvolvidos neste trabalho são testados num conjunto de dados retirados do repositório *UCI Machine Learning Repository*, pertencente à *University of California, Irvine*, que incluem os dados de características estruturais e de conjuntura de edifícios de habitação.

Como ponto central do trabalho o autor desenvolveu um algoritmo inspirado em sistemas imunológicos artificiais de seleção negativa [5]. Este algoritmo analisa os dados e procura características comuns que permitam o agrupamento de edifícios em classes semelhantes. Espera-se que estas classes sejam bons indicadores do tipo de consumo energético que o edifício apresenta.

O consumo energético foi dividido em dois tipos. Por um lado o consumo energético com o aquecimento (Heating load na nomenclatura inglesa) e por outro o consumo energético com o arrefecimento (Cooling load na nomenclatura inglesa). Pretende-se relacionar aquelas classes (ou clusters), com o consumo, determinar que combinações de variáveis caracteriza cada classe e quais as variáveis que irão ter mais impacto em cada um dos dois tipos de consumo.

No final do trabalho para além do algoritmo imunológico, foi implementada uma regressão baseada em redes neuronais cujo o objetivo foi estimar numericamente o consumo a partir dos dados de entrada. Além disto, foi implementado um outro algoritmo que, através da análise das sensibilidades das redes neuronais criadas, procura estimar quais as variáveis com maior influência nos consumos energéticos.

Finalmente os resultados foram condensados sob a forma de uma árvore de decisão. A construção da árvore foi baseada nos resultados obtidos ao longo do trabalho, com ajuda de um algoritmo implementado numa fase final.

As secções seguintes apresentam uma descrição detalhada dos procedimentos e métodos organizacionais relativos ao tratamento dos dados bem como à conceção do algoritmo de *clustering*. As secções seguintes estão organizadas do seguinte modo:

- Software utilizado e organização;
- Descrição dos dados;
- Pré processamento de dados;
- Algoritmo imunológico desenvolvido;
- Regressão com Rede Neuronal;
- Árvore de decisão.

3.2 Software utilizado e organização

3.2.1 Software utilizado

Neste trabalho foram usados três tipos de plataformas computacionais: Um ambiente de programação (*Matlab*) onde foi implementado um algoritmo, um ambiente de programação (*RStudio*) onde foi implementado um algoritmo de apoio à construção de uma árvore de decisão e folhas de cálculo onde são guardados todos os dados de entrada e resultados.

O software utilizado para manipular estas folhas de cálculo foram o *Microsoft® Excel* (2016) e o *LibreOffice Calc* (Versão 6).

Todo o código relativo aos algoritmos foram implementados utilizando o *software Mathworks MATLAB R2018b* e o *software RStudio® 1.1.463* e foram realizados na integra pelo autor.

3.2.2 Organização

Para lidar com os dados foi implementado um modelo de entrada e de saída que funciona da seguinte forma:

- As variáveis de entrada são lidas a partir da folha de cálculo na página 1 e esta não será modificada pelo algoritmo;
- Todos os resultados do algoritmo serão escritos na mesma folha de cálculo, e estando a primeira página reservada, serão colocados em páginas subsequentes;
- Na segunda página o programa irá produzir uma nova lista de entradas com as variáveis recodificadas, de acordo com o pré processamento de dados descrito na Secção 3.4;
- A terceira página destina-se a guardar os resultados do *clustering*, onde se incluem:
  - Uma lista de todos os centróides encontrados;

  - O número de vetores atribuídos a cada centróide.

- Na quarta página estarão os resultados da classificação das entradas em função dos centróides identificados no ponto anterior. Esta lista está ordenada por *cluster*, onde se inclui:
  - Entradas recodificadas;
  - Saídas (Y1 e Y2);
  - Atribuições de cada entrada a cada *cluster*.
- Na quinta página, serão escritos os resultados da análise dos mínimos médias e máximos dos *clusters*, por variável de entrada;
- Finalmente na última página será apresentada a tabela da análise das correlações.

A Figura 3.5 descreve os passos principais do modelo implementado.

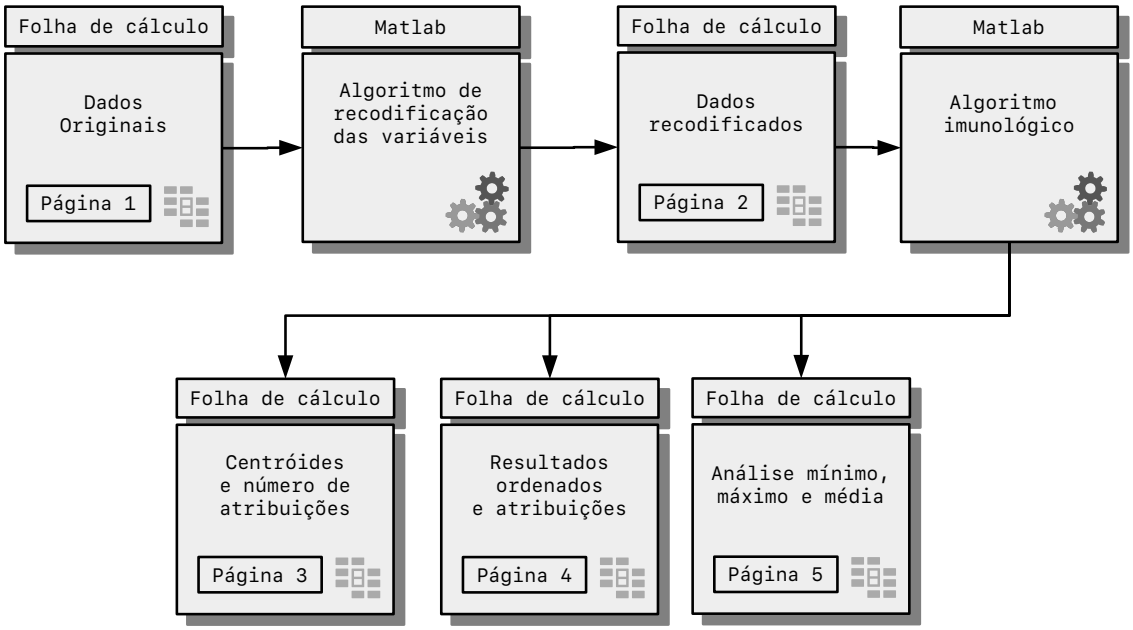


Figura 3.1: Esquema do comportamento do modelo implementado

3.3 Descrição dos dados

Os dados foram simulados e gerados no software *Autodesk Ecotect Analysis* [24]. Segundo os autores, estes dados foram gerados considerando que todos os edifícios apresentam o mesmo volume ($771,75\,m^3$), mas com áreas de superfície e dimensões diferentes. A simulação assume que os edifícios estão localizados em Atenas, Grécia.

Juntamente com os dados, os autores incluem um documento [24], em que se faz uma descrição detalhada do tipo de dados bem como uma análise estatística dos mesmos.

Os dados de entrada, provenientes da folha de cálculo, estão contidos na primeira página do documento, que se manterá inalterada. Esta página incluirá então os dados originais com as seguintes designações:

- **X1** – Compacticidade;
- **X2** – Área total da superfície interior (chão + teto + paredes);
- **X3** – Área das paredes;
- **X4** – Área do telhado;
- **X5** – Altura;
- **X6** – Orientação;
- **X7** – Área Envidraçada;
- **X8** – Distribuição da área envidraçada;
- **Y1** – Consumo energético para aquecimento;
- **Y2** – Consumo energético para arrefecimento;

As que as variáveis iniciadas por 'X' representam as variáveis de entrada, e por 'Y' as variáveis de saída. A Tabela 3.1 apresenta o número de instâncias e a gama de variação de cada uma das grandezas.

Tabela 3.1: Descrição numérica dos valores das variáveis de entrada

Designação	Nº de valores únicos	Alcance de valores
X1	12	0,62 - 0,98
X2	12	514 - 808
X3	7	245 - 416
X4	4	110 - 220
X5	2	3,50 7
X6	4	2 - 5
X7	4	0 - 0,4
X8	6	0 - 5
Y1	586	6.01 - 43,1
Y2	636	10,9 - 48,03

Proveniente do documento acima citado, estão as seguintes descrições de cada uma das variáveis de entrada:

As áreas envidraçadas (X7) estão expressas como percentagens da área do chão cujos valores são 10%, 25%, 40%.

A distribuição das áreas envidraçadas (X8) inclui seis cenários diferentes:

- O cenário 0 representa a situação em que o edifício não apresenta área envidraçada.
- No cenário 1 o edifício apresenta uma distribuição uniforme de 25% da área envidraçada em cada fachada.
- No cenário 2 o edifício apresenta 55% da área envidraçada na fachada Norte e 15% nas restantes.
- No cenário 3 o edifício apresenta 55% da área envidraçada na fachada Este e 15% nas restantes.
- No cenário 4 o edifício apresenta 55% da área envidraçada na fachada Sul e 15% nas restantes.
- No cenário 5 o edifício apresenta 55% da área envidraçada na fachada Oeste e 15% nas restantes.

No que toca às restantes variáveis, o documento não faz menção acerca das unidades em que se exprimem. Foi realizada uma pesquisa no sentido de se perceber as unidades que tipicamente se utilizam no *software* onde foram realizadas as simulações. As conclusões apresentam-se de seguida.

A área envidraçada (X7) está expressa em percentagem, sendo esta relativa à área do chão.

As áreas da parede (X3), do telhado (X4) e da superfície interna (X2) estão expressas em metros quadrados.

A altura (X5) está expressa em metros.

Relativamente à compacticidade, fica claro pela análise da correlação, na Secção 4.1 que esta é inversamente proporcional à área da superfície, o que significa que valores mais elevados correspondem a edifícios menos compactos, contrariamente ao que seria de esperar. A compacticidade é habitualmente definida pelo rácio entre a superfície total envolvente e o volume, sendo exprimida em m^{-1} [25].

Relativamente à orientação (X6) os autores não dão uma indicação clara sobre os valores, indicando apenas que cada valor (que pode ser 2, 3, 4 ou 5) será uma orientação segundo cada um dos quatro pontos cardeais principais (Norte, Sul, Este ou Oeste). A indicação de a que orientação corresponde cada valor não é especificada pelos autores.

No que toca às saídas o *Ecotect* tipicamente utiliza o *kWh*.

3.4 Pré processamento de dados

O algoritmo implementado recorre à distância euclidiana para realizar as comparações entre os vários vetores de entrada, ou para comparar vetores com centróides. Isto coloca alguns problemas se forem usados dados sem qualquer transformação, dado que algumas variáveis originais não são numéricas. De facto, a distribuição da área envidraçada (X8) está representada numericamente nos

dados, no entanto cada algarismo corresponde a uma de seis situações distintas conforme descrito na secção anterior, ou seja, este tipo de dado é qualitativo.

De forma a ultrapassar este problema as variáveis X7 e X8 foram recodificadas em quatro novas variáveis. Estas quatro variáveis representam a área envidraçada em metros quadrados segundo cada um dos quatro principais pontos cardeais.

A variável X7 representa a percentagem de área envidraçada relativamente à área do chão. Esta área não é incluída nos dados mas foi possível ser calculada.

Assumiu-se que a área do chão seria numericamente igual à área do telhado (X4). Para tal ser verdade teria que se verificar que a soma da área do telhado (X4) com a área do chão com a área das paredes (X3) daria um valor igual à área total da superfície interior (X2). Realizou-se o cálculo para todos os casos e constatou-se isso mesmo:

$$X2 - X3 - (2 * X4) = 0$$

Retirando à superfície total interna do edifício (X2) a área das paredes e duas vezes a área do chão (X4) obtivemos zero em todos os casos.

Usando as descrições relativas a cada situação de X8, como explicado na Secção 3.4, podemos multiplicar a percentagem de área envidraçada (percentagem esta que está incluída na descrição de X8) pela área total envidraçada (X7) pela área do telhado (X4) e obter as áreas envidraçadas segundo cada um dos quatro principais pontos cardeais:
Assumindo o cenário 0, em que X8 = 0:

- Área a Sul = Área Norte = Área Este = Área Oeste = 0

Assumindo o cenário 1, em que X8 = 1:

- Área a Sul = Área Norte = Área Este = Área Oeste = 25% * X7 * X4

Assumindo o cenário 2, em que X8 = 2:

- Área a Norte = 55% * X7 * X4
- Área a Sul = Área Este = Área Oeste = 15% * X7 * X4

Assumindo o cenário 3, em que X8 = 3:

- Área a Este = 55% * X7 * X4
- Área a Sul = Área Norte = Área Oeste = 15% * X7 * X4

Assumindo o cenário 4, em que X8 = 4:

- Área a Sul = 55% * X7 * X4
- Área a Norte = Área Este = Área Oeste = 15% * X7 * X4

Assumindo o cenário 5, em que X8 = 5:

- Área a Oeste = 55% * X7 * X4
- Área a Sul = Área Este = Área Norte = 15% * X7 * X4

Relativamente à variável Orientação (X6) não temos informação nenhuma acerca do seu significado. Como tal, optou-se por recodificar esta variável em 4 novas que indicam em binário os quatro possíveis valores que X6 pode tomar.

Após escolhido o método de recodificação obtemos as novas 13 variáveis com as respetivas designações e unidades:

- **X1** – Compacticidade [m^{-1}];
- **X2** – Área da superfície [m^2];
- **X3** – Área das paredes [m^2];
- **X4** – Área do telhado [m^2];
- **X5** – Altura [m];
- **X6** – Orientação 1;
- **X7** – Orientação 2;
- **X8** – Orientação 3;
- **X9** – Orientação 4;
- **X10** – Área Envidraçada a Sul [m^2];
- **X11** – Área Envidraçada a Norte [m^2];
- **X12** – Área Envidraçada a Este [m^2];
- **X13** – Área Envidraçada a Oeste [m^2];
- **Y1** – Consumo energético para aquecimento [kWh];
- **Y2** – Consumo energético para arrefecimento [kWh];

3.5 Algoritmo desenvolvido

Nesta secção serão descritos os objetivos e funcionamento do algoritmo de *clustering*. Este algoritmo foi implementado com base em sistemas imunológicos e pretende agrupar os vários casos presentes nos dados com propriedades semelhantes em *clusters* comuns, e no fim apresentar também um detetor (protótipo) que caracterize cada um desses *clusters*.

De seguida encontra-se uma descrição geral do funcionamento do algoritmo:

Passo 1: Preparação dos dados

- Lê dados do *excel* já recodificados;
- Normaliza os dados;
- Escolhe o primeiro detetor;

Passo 2: Ciclo para a formação inicial dos *clusters*

- Calcula as distâncias dos vetores de entrada ao detetor;
- As distâncias dos vetores mais próximos do detetor são casadas com esse detetor;
- Se não houver pelo menos N casamentos, o critério de distância é aumentado e volta ao inicio do ciclo, sem incrementar o *cluster* atual;
- Se houver mais de N casamentos incrementa o contador de *clusters*;
- Escolhe o vetor mais distante do detetor atual para se tornar o novo detetor;
- Enquanto todos os vetores não estiverem casados, este ciclo vai ser executado;

Passo 3: Recalcular as atribuições baseadas nos centróides dos detetores

- Calcula o centróide de cada *cluster*;
- Calcula as distâncias de cada vetor de entrada a todos os centróides;
- Atribui a cada entrada a classificação do *cluster*, baseado na menor distância deste aos centróides;
- Executa o ciclo algumas vezes de modo a garantir que a solução converge para um ótimo;

O fluxograma da Figura 3.2 sintetiza a estrutura do modelo implementado.

Na secção seguinte faz-se uma descrição mais detalhada das opções tomadas na conceção do algoritmo.

O primeiro passo é então normalizar os dados de entrada. Para isto recorreu-se à função do *Matlab*, o *zscore* que normaliza os dados para média zero e variância unitária. O processo de normalização permite que o algoritmo lide com dados que apresentem diferentes tipos de atributos, cumprindo o requisito 2 da Subsecção 2.2.1.

De seguida foi necessário escolher um detetor inicial que servisse de ponto de partida para realizar as comparações. Neste estudo a prioridade é obter *clusters* muito separados, razão pela qual faz sentido começar por um extremo. Escolheu-se então o menor valor de consumo energético para aquecimento (poderia ter sido escolhido também o mais elevado). Este detetor foi escolhido a partir da lista dos dados e apresenta as seguintes características de saída:

$$\text{Y1\_HL} = 6.01\ kWh \text{ e Y2\_CL}=10.94\ kWh$$

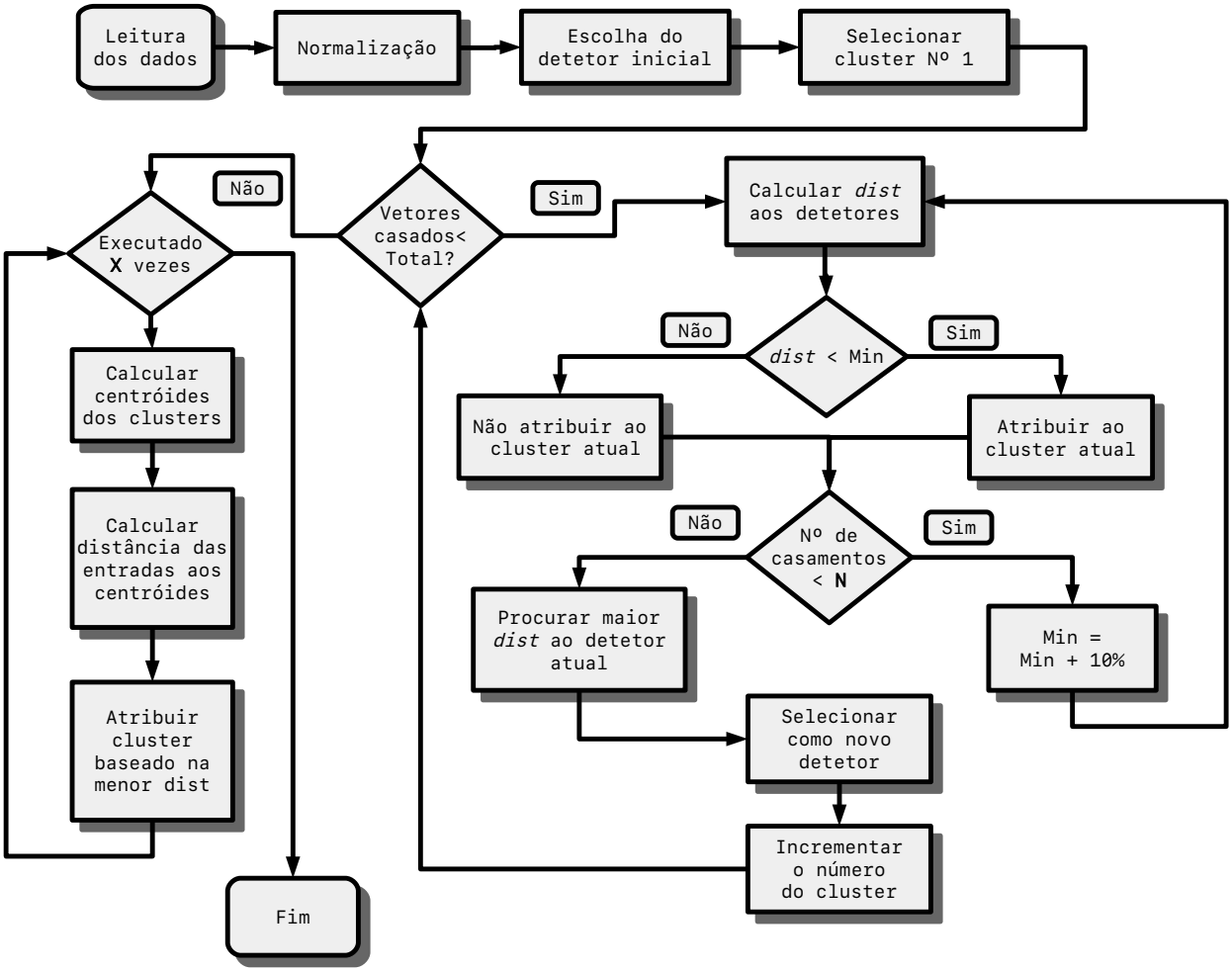


Figura 3.2: Fluxograma do funcionamento do algoritmo

O vetor que corresponde ao detetor escolhido acima é atribuído ao *cluster* #1.

O algoritmo segue então para um ciclo (*while cycle*) que irá ser executado enquanto todos os vetores de entrada não forem casados com um *cluster*. Dentro do ciclo o primeiro passo é calcular as distâncias de todos os vetores ao detetor atual. O algoritmo utiliza a distância euclidiana para comparar as diversas entradas com o detetor atual, conforme a Equação 3.1.

$$D = \sqrt{(X1_{Entrada} - X1_{Detetor})^2 + ... + (X13_{Entrada} - X13_{Detetor})^2} \tag{3.1}$$

De seguida é escolhido um critério de distância em que, dado todo o espaço dos resultados encontrados anteriormente, aceita-se apenas os valores mais próximos. Esta percentagem de aceitação (%A) é parametrizável no algoritmo. Espera-se que um algoritmo de *clustering* seja capaz de discriminar características entre os dados disponíveis e associar apenas os vetores semelhantes. Foi implementado este parâmetro, baseado na taxa de afinidade, que é característica dos algoritmos imunológicos. Para isto a Equação 3.2 foi desenhada para selecionar apenas os valores mais próximos do detetor atual para atribuição.

$$Alcance\,de\,aceitacão = [MAX(D) - MIN(D) * \%A] + MIN(D) \tag{3.2}$$

Visto que o detetor inicialmente é escolhido a partir da lista dos dados, a distância mínima seria sempre zero, pois o detetor seria comparado consigo próprio. Para isto o algoritmo foi preparado para ignorar esta comparação.

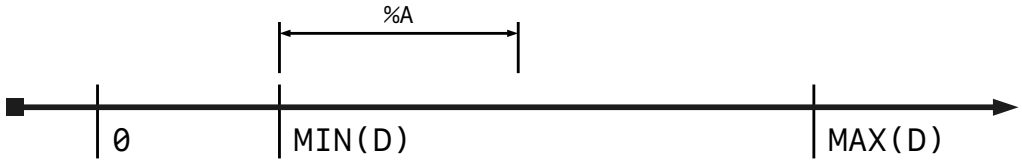


Figura 3.3: Processo de atribuição baseado numa certa percentagem de aceitação

Com isto todos os vetores de entrada são testados, e aqueles que ainda não tiverem um *cluster* atribuído e cumprirem o requisito da distância máxima, são casados com o *cluster* atual.

Como próximo passo o algoritmo verifica se existe um número mínimo de atribuições. Pretende-se com isto limitar o número de *clusters* pois, por uma questão de simplicidade das conclusões, não é desejável obter um elevado número de *clusters* muito especializados com características altamente distintas. Para isto foi implementado uma variável parametrizável que define o número mínimo de atribuições por *cluster*.

Assim se o *cluster* após o ciclo de atribuições apresentar menos de um certo número de vetores atribuídos, o ciclo recomeça com o critério da distância alargado em 5%:

$$Alcance\,de\,aceitacão = [MAX(D) - MIN(D) * (\%A + 0.05)] + MIN(D) \tag{3.3}$$

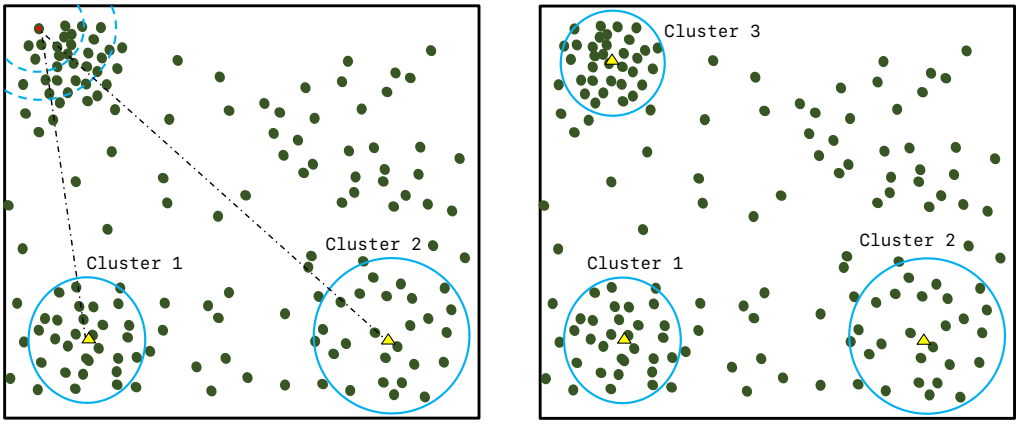
Este alargamento do critério é realizado o número de vezes que for necessário para garantir o mínimo de atribuições mencionado.

Caso no final ocorra uma situação em que restam menos vetores para serem casados do que o número mínimo de atribuições acima mencionado o programa iria entrar num ciclo infinito. Para lidar com esta situação, o algoritmo está preparado para agrupar todos esses vetores num *cluster* final.

Para finalizar este ciclo o algoritmo procura todas as posições não casadas e escolhe a que apresenta a maior distância ao conjunto de detetores atuais para ser o novo detetor. A Figura 3.4 descreve o comportamento da rotina implementada.

No *cluster* final pode existir um número considerável de vetores que representam *outliers*. A definição destes vetores por um detetor não iria levar a uma representação fiel das características das suas atribuições, dado que terão com certeza características muito diferenciadas. Por este motivo, o *cluster* final é eliminado e os seus vetores serão alvo de reatribuições num ciclo posterior.

No final da execução total deste ciclo inicial, teremos um certo número de *clusters* com a totalidade dos vetores de entrada atribuídos, e distribuídos de acordo com as distâncias. Isto dará ao algoritmo um ponto de referência das posições dos vetores.



Os clusters 1 e 2 já estão definidos. Os protótipos de cluster são representado pelo triângulos amarelos. O ponto vermelho é o vetor mais distante, que será a base de definição do Cluster 3. As linhas a tracejado a circundar o ponto vermelho correspondem a duas iterações do alcance de aceitação.

Após satisfação do critério de aceitação, o protótipo de cluster é recalculado (triângulo amarelo), o que resulta na definição do Cluster 3.

Figura 3.4: Descrição do processo de escolha do próximo detetor

Partindo destas informações, foi implementado um novo ciclo que realiza uma série de comparações baseadas novamente na distância, mas desta vez entre os vetores de entrada e os centróides dos *clusters* encontrados. Inicialmente o algoritmo acede à lista das atribuições encontradas anteriormente e calcula o centro do *cluster* fazendo a média para cada variável de todos os vetores atribuídos, conforme o esquema da Figura 3.5.

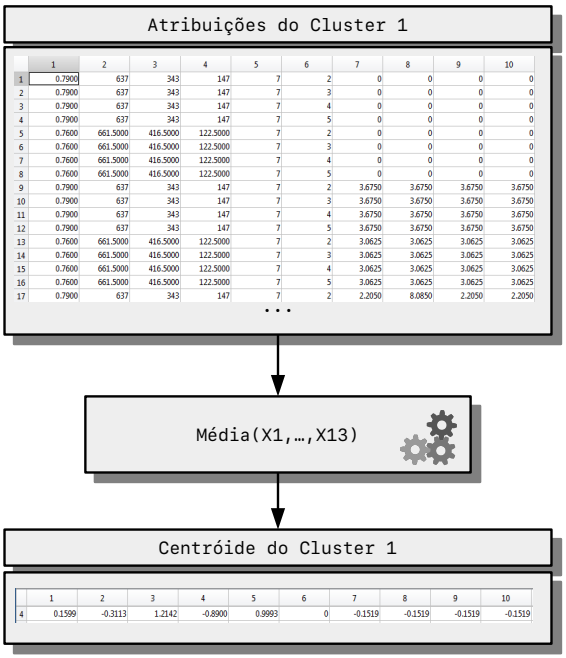


Figura 3.5: Processo cálculo dos centróides baseado nos vetores atribuídos a um *cluster*

Após todos os centróides dos *clusters* serem encontrados, o algoritmo compara todas os vetores de entrada com todos os centróides existentes e realiza novas atribuições baseando-se na menor

distância. Depois de todos os vetores estarem casados, os centróides dos *clusters* serão recalculados. Isto irá ser repetido até os resultados não apresentarem alterações significativas ou até atingir um determinado número de iterações. A cada iteração os detetores vão sendo recalculados e os novos detetores serão os centróides que incluem todos os vetores casados com o *cluster*.

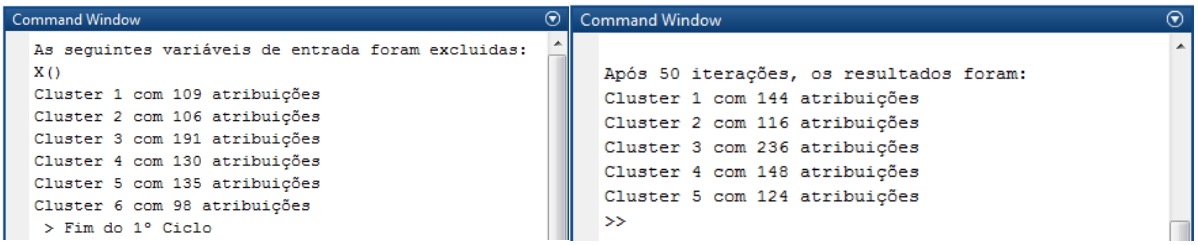
Como o programa lida com os dados normalizados desde o início é necessário, no final deste ciclo, reverter o processo de normalização para obter os verdadeiros valores dos centróides. Recorreu-se à documentação do *Matlab* para compreender o processo de normalização através da função *zscore*. Esta função calcula para uma série de amostras, a média (μ) e o desvio padrão (σ) e utiliza estes valores para calcular o valor normalizado (z):

$$z = (x - \mu)/\sigma \tag{3.4}$$

Resolvendo a equação anterior em ordem ao valor não normalizado temos:

$$x = \sigma * Z + \mu \tag{3.5}$$

A Figura 3.6a apresenta um exemplo de saída de dados do algoritmo através da linha de comandos. Após o primeiro ciclo o número de *clusters* é 7, com as respetivas atribuições. Após o segundo ciclo, algumas entradas vão ser redistribuídas, passando a pertencer a um novo *cluster*, como se pode ver na Figura 3.6b.



(a) Resultados após primeiro ciclo (b) Resultados após ciclo final

Figura 3.6: Exemplo de uma atribuição de classes após o primeiro ciclo, e no final

Para finalizar o programa produz gráficos que ajudam a interpretação dos resultados.
O primeiro conjunto de gráficos apresentam as características dos centróides para cada *cluster*:

- Gráficos de barras com o valor dos centróides, por *cluster*;
- Diagramas de caixa com as variações dos centróides, por *cluster*;

Os gráficos de barras serão em igual quantidade ao número de *clusters* e apresentam os valores normalizados dos centróides para cada variável de entrada (X1 a X13). Os diagramas de caixa serão igualmente na mesma proporção que o número de *clusters* e indicam, dentro de cada *cluster*, como se distribuem os valores para cada variável de entrada. Estes gráficos permitem-nos avaliar se os *clusters* (através dos centróides) apresentam características distintas.

O segundo conjunto de gráficos apresenta a distribuição das entradas agrupadas em cada *cluster* em ordem aos respetivos valores das saídas:

- Gráfico de dispersão que relaciona as entradas e os respetivos agrupamentos por *cluster* com o valor da saída (Y1 - Heating Load);
- Gráfico de dispersão que relaciona as entradas e os respetivos agrupamentos por *cluster* com o valor da saída (Y2 - Cooling Load);
- Diagrama de caixa com as variações das saídas, por *cluster* (Y1 - Heating Load);
- Diagrama de caixa com as variações das saídas, por *cluster* (Y1 - Cooling Load);

O terceiro conjunto de gráficos é composto diagramas de caixa com vista à análise individual de cada variável, onde se incluem os valores para todos os *clusters*, permitindo uma comparação mais direta entre estes.

Os gráficos de dispersão permitem avaliar se a cada *cluster* foram atribuídas entradas com características de saída distintas. No entanto devido ao elevado número de entradas, os gráficos de dispersão tornam-se difíceis de interpretar. Utiliza-se então os diagramas de caixa, que permitem uma análise semelhante mas com a possibilidade de saber como estão distribuídos estatisticamente os valores ao longo do espaço de resultados.

O último gráfico é do tipo gráfico de barras e indica as distâncias de cada *cluster* aos restantes *clusters*. Com este gráfico podemos ter uma ideia do afastamento que os centróides tem uns dos outros.

3.6 Regressão com rede neuronal

De forma a complementar o algoritmo de *clustering* recorreu-se a uma regressão baseada em redes neuronais. O objetivo principal é implementar um processo de estimação das cargas térmicas de edifícios a partir das suas características construtivas. Um segundo objetivo consiste em determinar quais as variáveis de entrada irão ter mais peso nas saídas, e por conseguinte no rendimento térmico do edifício, e além disso ter uma ferramenta capaz de prever com precisão os valores numéricos das saídas.

Para isto recorreu-se à ferramenta *nftool*, do *Matlab*. Esta ferramenta permite treinar rapidamente uma rede neuronal e com isto, desenvolveu-se um algoritmo capaz de encontrar as sensibilidades de cada variável de entrada. A descrição deste algoritmo está na Subsecção 3.6.1.

Nas secções seguintes irá descrever-se a implementação da rede neuronal utilizando o *nftool* e o funcionamento do algoritmo implementado, utilizado para a regressão e cálculo das sensibilidades.

3.6.1 Rede Neuronal - *nftool*

Para criar a rede neuronal foram utilizados os dados de entrada recodificados e normalizados. Estes dados foram introduzidos no *nftool* através da interface gráfica.

Das 768 amostras a ferramenta escolheu aleatoriamente 538 (70%) para o conjunto de treino, 115 (15%) para fazerem parte do conjunto de validação e 115 (15%) para o conjunto de teste.

De seguida após várias experiências foram considerados 10 neurónios para a camada escondida. Finalmente a rede foi treinada com o algoritmo *Levenberg-Marquardt*.

O resultado deste processo é uma função que recebe dados de entrada, que deverão ter o mesmo número de variáveis, e retorna a previsão para a saída. São sempre treinadas duas redes para cada estudo, uma para cada saída (Y1 e Y2).

3.6.2 Algoritmo para o cálculo das sensibilidades

Esta secção pretende descrever o funcionamento do algoritmo implementado pelo autor em *Matlab*, e que tem como objetivo obter as sensibilidades da saída relativamente a cada uma das entradas (X1,...,X13). Estas sensibilidades representam o peso que o algoritmo de treino das redes neuronais atribuiu a estas variáveis na previsão das saídas. A descrição seguinte corresponde ao cálculo da sensibilidade da variável X1. O algoritmo irá proceder de igual modo para as restantes 12 variáveis.

Em primeiro lugar a matriz dos vetores de entrada é clonada. A esta nova matriz, em todas as posições de entrada relativamente à variável X1, o seu valor irá ser incrementado de Delta (Δ). Este valor foi escolhido arbitrariamente e pretende-se que seja um valor reduzido face aos valores típicos das variáveis. Foi escolhido então $\Delta = 0.0001$.

A matriz original terá em diante a designação [X] e a matriz que foi incrementada terá a designação [X'].

O próximo passo foi passar ambas as matrizes como argumento em ambas as funções da rede neuronal (Previsão de Y1 - Heating Load e Y2 - Cooling Load), resultando em quatro matrizes de saída. As matrizes cujos dados de entrada não sofreram o incremento serão designadas [Y1] e [Y2] e as restantes duas matrizes [Y1'] e [Y2'].

Finalmente estamos em condições de calcular as sensibilidades (S) para X1, fazendo:

$$S_{X1\_Y1} = \frac{dY1}{dX1} = \frac{\Delta Y2}{\Delta X1} = \frac{[Y] - [Y']}{[X] - [X']} \tag{3.6}$$

$$S_{X1\_Y2} = \frac{dY2}{dX1} = \frac{\Delta Y2}{\Delta X1} = \frac{[Y] - [Y']}{[X] - [X']} \tag{3.7}$$

O algoritmo vai percorrer cada uma das 768 entradas realizando os cálculos conforme descritos na Equação 3.6 e na Equação 3.7. No final temos uma matriz que contém a sensibilidade de X1 relativamente à saída para cada um dos 768 casos. A média dos valores absolutos de cada um destes casos irá dar um valor que será uma boa aproximação da importância de X1, relativamente às restantes variáveis, no rendimento energético dos edifícios.

Note-se que se, para um determinado estado do sistema, a variável hipotética X_a afeta mais a saída que a variável X_b, então a a derivada $dY1/dX_a$ será mais elevada que $dY1/dXb$. No caso limite de derivada nula, então a saída Y não seria alterada por X_b naquele estado. Aplicando este conceito a todos os estados do sistema (ou seja, a todos os exemplos disponíveis), este algoritmo

dará uma informação sobre a influência média de cada variável na saída. Uma descrição gráfica do algoritmo está disponível na Figura 3.7.

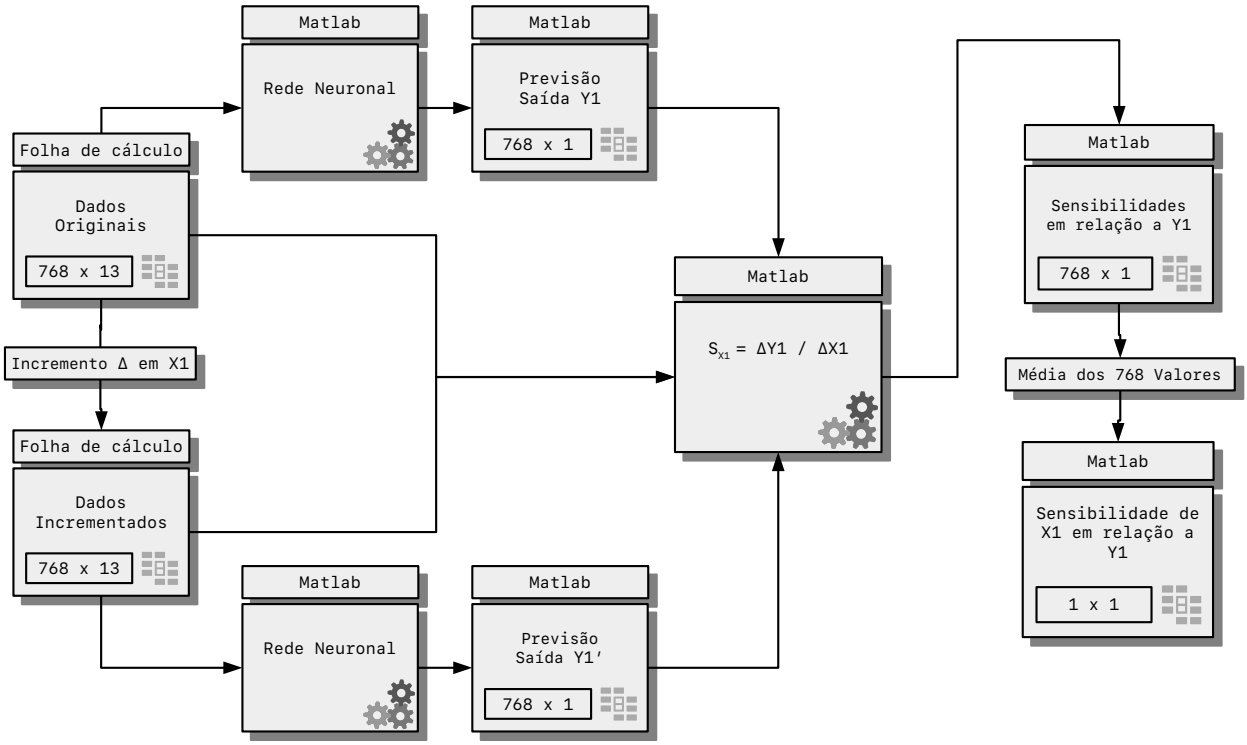


Figura 3.7: Esquema do algoritmo para o cálculo da sensibilidade de X1 em relação a Y1

Capítulo 4

Exposição e análise dos resultados

Neste capítulo são apresentados os resultados decorrentes da implementação dos procedimentos descritos na metodologia bem como uma análise dos resultados obtidos.

Antes de aplicar os algoritmos, foi feita uma análise da correlação entre as variáveis de entrada. Esta análise permite encontrar, caso existam, grupos de variáveis cuja informação seja redundante, o que permite simplificar a análise por eliminação de dados supérfluos.

Em relação ao algoritmo imunológico implementado, irão ser apresentados os resultados detalhados, incluindo as características de cada *cluster* (classe). Os *clusters* serão comparados em termos de características de entrada dos vetores casados e dos centróides. No final serão classificados de acordo com as características que os distinguem. O tempo de execução do programa e o *hardware* utilizados serão também apresentados, para se perceber quão eficiente é a implementação real.

Em relação às redes neuronais irão ser apresentados vários índices de performance que irão caracterizar a qualidade da estimação. Irá ser feita uma análise gráfica aos erros individuais e a identificação de possíveis *outliers*.

Nos testes de análise de sensibilidades, serão considerados quer o conjunto inicial de variáveis quer os subconjuntos resultantes da eliminação de variáveis correlacionadas. Em seguida, serão comparados os resultados obtidos nas diferentes situações.

No final utilizando todo o conhecimento adquirido irá ser apresentada uma árvore de decisão que, a partir do valor das entradas, deverá ser capaz de propor um meio de classificação (identificação do *cluster*) estruturado com base nas características de entrada.

4.1 Correlação

O cálculo do coeficiente de correlação entre as variáveis foi realizada manualmente no *software LibreOffice Calc*. Foram analisadas as correlações entre todas as variáveis, não só entre as recodificadas (X1 a X13) mas também entre as originais (X1 a X8) e incluindo as saídas. Os resultados mais notáveis apresentam-se sintetizados nos seguintes pontos:

- A variável original X6 (Orientação) e as variáveis que resultaram da sua recodificação (X6 a X9) não apresentam nenhuma correlação com as restantes variáveis de entrada.
- As quatro variáveis correspondentes às áreas envidraçadas (X10 a X13) estão correlacionadas positivamente com a Área do telhado (X4) por 26% e por 25% entre si.
- A Compacticidade (X1) apresenta uma correlação de -87% com a Área do telhado (X4), de +83% com a Altura (X5) e de -99% com a Área total da superfície interior (X2).
- A altura (X5) apresenta uma correlação de +83% com a Compacticidade (X1), de -86% com a Área da superfície (X2) e de -97% com a Área do telhado (X4).
- Em relação às saídas, ambas apresentam uma correlação de -86% com a Área do telhado (X4) e de cerca de 89% com a Altura (X5).

	X2_SA	X3_WA	X4_RA	X5_H	X6_O	X7_SGA	X8_NGA	X9_EGA	X10_WGA	Y1_HL	Y2_CL
X1_C	-99.2	-20	-87	83	0	-23	-23	-23	-23	62.2	63.4
X2_SA		20	88	-86	0	23	23	23	23	-65.8	-67.3
X3_WA			-29	28	0	-8	-8	-8	-8	45.6	42.7
X4_RA				-97.3	0	26	26	26	26	-86.2	-86.3
X5_H					0	-25	-25	-25	-25	88.9	89.6
X6_O						0	0	0	0	0.0	0.1
X7_SGA							24	24	24	-7.6	-10.7
X8_NGA								24	24	-7.5	-10.7
X9_EGA									24	-8.6	-12.4
X10_WGA										-8.8	-12.2
Y1_HL											97.6

Figura 4.1: Valores percentuais da correlação entre variáveis

Em relação às variáveis Orientação, a inexistência de correlação indica que estas são linearmente independentes, relativamente às restantes variáveis de entrada. De facto esta variável contrasta com as restantes variáveis como a Compacticidade, a Área do telhado e a Altura pois estas últimas representam ultimamente características geométricas do edifício. Tendo isto em conta é natural que estas características apresentem alguma correlação entre si.

No que toca à forte correlação negativa entre a Área do telhado e a Altura esta pode ser compreendida se tivermos em conta a natureza artificial dos dados gerados. De facto todos os edifícios apresentam o mesmo volume e, assim sendo, com o aumento da Altura, a Área do telhado tem necessariamente que diminuir.

Por outro lado a forte correlação negativa entre a Compacticidade e a Área da superfície é explicada atendendo ao facto de que, neste caso de estudo, estas duas variáveis representam o mesmo. Para o mesmo volume, um edifício com maior área interna será menos compacto do que

um edifício com menor área interna. O facto da correlação ser negativa é explicada pelas unidades em que X1 aparece (m^{-1}), tal como descrito na Secção 3.3.

Para finalizar esta análise refira-se que a correlação relativamente elevada das saídas com as duas variáveis de entrada (X4 e X5) poderá ser um indicador do peso que estas terão no que toca à eficiência energética do edifício.

4.2 Algoritmo Imunológico

4.2.1 Parametrização

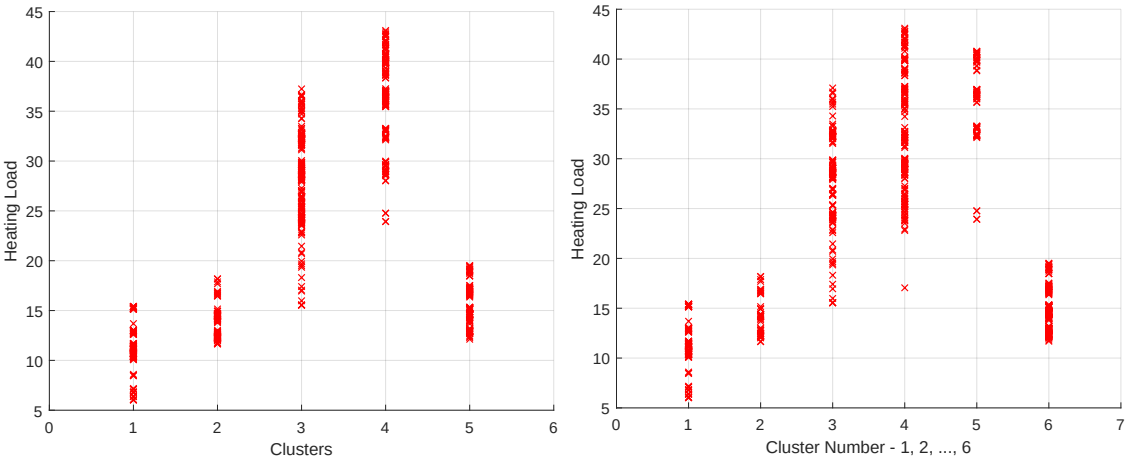
Durante o processo de conceptualização do algoritmo foi considerada necessária a existência de alguns parâmetros, para controlo da evolução do processo, de modo a otimizar o resultado final e também para testar diferentes hipóteses de obtenção de classes. Uma das grandes vantagens dos AIS é a sua flexibilidade. De facto, nestes algoritmos é possível considerar diferentes medidas de afinidade (ou de distâncias entre vetores), diferentes alternativas de geração de novos detetores, diferentes critérios de paragem, entre outros.

Por um lado o algoritmo deve ser capaz de discriminar entre os dados disponíveis e realizar atribuições entre vetores similares. Para isto foram realizados vários testes com diferentes valores da percentagem de aceitação (%A) e considerou-se que um valor de 35%, aliado a um incremento de 5%, seria aceitável para manter as características dos *clusters* distintas entre si. Baseado no funcionamento do algoritmo descrito na metodologia, cada vez que um novo detetor é encontrado, de todas as distâncias calculadas a este, apenas os menores valores (35%) serão casados com o *cluster*.

Por outro lado, e por questões práticas, não é bom ter um número elevado de *clustering* com características relativamente próximas (como descrito no requisito 6 da Subsecção 2.2.1). Para isto foi implementado um número de atribuições mínimas por *cluster*. Após vários testes considerou-se que 100 atribuições seria um valor de partida razoável, podendo ser aumentado caso se pretenda um menor número de *clusters* ou diminuído caso se pretenda aumentar, o que conduziu a a 5 *clusters* no final. Assim, se a cada iteração o *cluster* atual apresentar menos de 100 vetores casados, a percentagem de aceitação é incrementada em 5% e as atribuições realizadas novamente.

Em relação ao segundo ciclo, este inclui várias iterações de modo a permitir que os resultados tendam para um valor ótimo. Os testes iniciais mostraram que um baixo número de iterações neste ciclo poderia conduzir a resultados algo diferentes. No entanto após vários testes, concluiu-se que cerca de 30 execuções do ciclo são suficientes para estabilizar os resultados. Foi então implementado um valor conservador de 50 execuções.

Como é possível verificar na Figura 4.2 com um aumento nas restrições das variáveis parametrizáveis (redução da percentagem de aceitação e do número mínimo de atribuições) chegou-se a resultados semelhantes. De facto os *clusters* 1, 2 e 5 na Figura 4.2a parecem não sofrer alterações, sendo representados na Figura 4.2b pelos *clusters* 1, 2 e 6 respetivamente.



(a) Atribuições com os parâmetros referidos

(b) Atribuições com um aumento nas restrições

Figura 4.2: Comparação das atribuições após restrição dos parâmetros do algoritmo

Relativamente aos *clusters* 3 e 4 na Figura 4.2a sofrem uma pequena redistribuição dando origem aos *clusters* 3, 4 e 5 mas sem uma melhoria significativa na precisão.

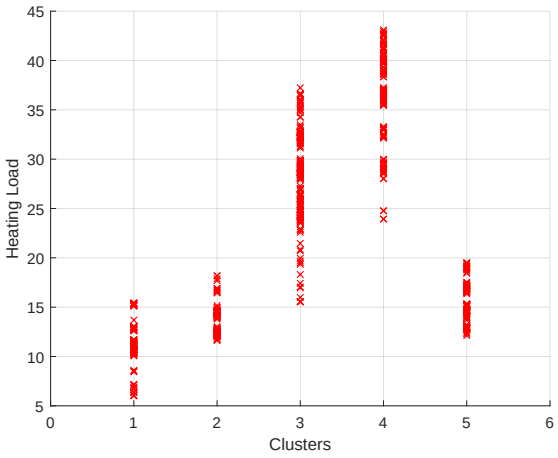
4.2.2 Recodificação

De forma a justificar a necessidade das recodificações descritas na metodologia foram executadas as seguintes variações do algoritmo: Primeiro sem recodificações, com recodificação apenas das variáveis X7 e X8 (Áreas envidraçadas), com recodificação apenas da orientação e finalmente com ambas as recodificações.

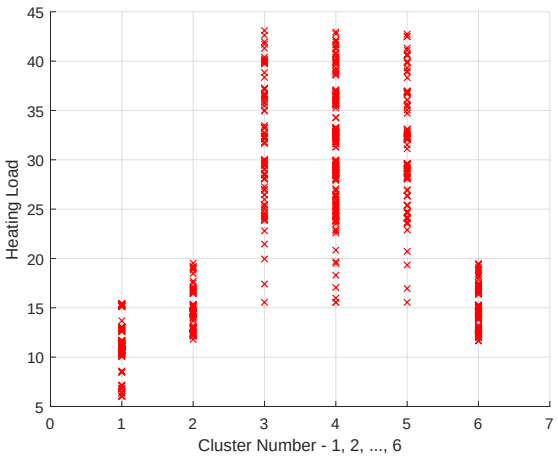
Em primeiro lugar tornou-se bastante óbvio que recodificar estas variáveis produzia sempre melhores resultados, devido certamente ao facto de que muita informação se encontra escondida por trás dos valores da Distribuição da área envidraçada (X8 nos dados originais) sob a forma de descrições.

De seguida foi averiguado se a recodificação da Orientação (X6 nos dados originais) introduzia melhorias na performance do algoritmo.

Pela análise da Figura 4.3 fica claro que o algoritmo perdeu a capacidade de diferenciar entre o *cluster* 3 e 4 (Figura 4.3a), no que toca à saída da Carga de aquecimento. Com a introdução da recodificação da orientação, estes dois grupos foram distribuídos por três novos grupos (Figura 4.3b) sem aparente diferenciação entre eles. Em ambos os casos a parametrização do algoritmo foi a mesma, tal como descito na secção anterior.



(a) Atribuições com recodificação apenas das áreas envidraçadas



(b) Atribuições com ambas as recodificações

Figura 4.3: Comparação das atribuições com diferentes recodificações

Com esta análise ficou claro que a recodificação da orientação não trouxe melhorias ao algoritmo e, como tal, de agora em diante as variáveis utilizadas serão conforme o descrito na tabela seguinte:

X1 - Compacticidade [%] **X2** - Área da superfície interna [m^2]
X3 - Área das paredes [m^2] **X4** - Área do telhado [m^2]
X5 - Altura [m] **X6** - Orientação
X7 - Área envidraçada a Sul [m^2] **X8** - Área envidraçada a Norte [m^2]
X9 - Área envidraçada a Este [m^2] **X10** - Área envidraçada a Oeste [m^2]

4.2.3 Reatribuições

Após a execução do algoritmo, obtivemos do primeiro ciclo seis *clusters* com diferentes atribuições, e que sofreram algumas alterações após a remoção do último *cluster* e subsequente passagem pelo segundo ciclo. Estes resultados das atribuições apresentam-se de seguida:

Primeiro Ciclo:

Cluster 1 - 109 atribuições **Cluster 2** - 106 atribuições
Cluster 3 - 191 atribuições **Cluster 4** - 130 atribuições
Cluster 5 - 135 atribuições **Cluster 6** - 97 atribuições

Segundo Ciclo:

Cluster 1 - 144 atribuições **Cluster 2** - 116 atribuições
Cluster 3 - 236 atribuições **Cluster 4** - 148 atribuições
Cluster 5 - 124 atribuições

Após análise das atribuições, concluímos que 341 vetores mudaram de *cluster* após o segundo ciclo. Descontando os elementos do *cluster* 6 que forçosamente foram redistribuídos, chegamos a um valor efetivo de 244 vetores que mudaram a sua atribuição, como resultado do agrupamento baseado em centróides.

A título de exemplo consideremos os quatro primeiros edifícios, representados pelos dados da Figura 4.4.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	Y1	Y2	Ciclo 1	Ciclo 2
1	0.98	515	294	110	7	2	0	0	0	0	15.6	21.3	3	3
2	0.98	515	294	110	7	3	0	0	0	0	15.6	21.3	3	3
3	0.98	515	294	110	7	4	0	0	0	0	15.6	21.3	3	3
4	0.98	515	294	110	7	5	0	0	0	0	15.6	21.3	6	3

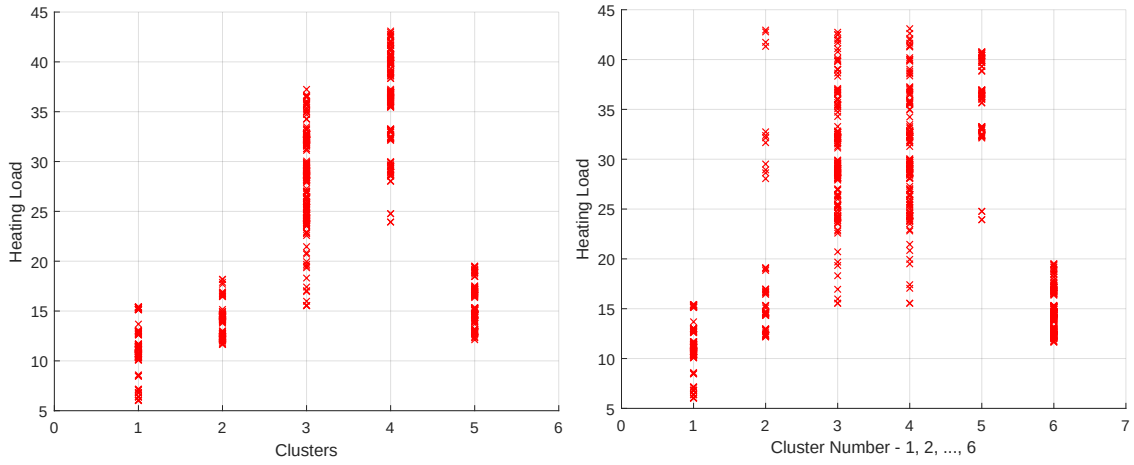
Figura 4.4: Exemplo de quatro vetores e respetivas atribuições por ciclo

Como se pode verificar estes dados diferem apenas na Orientação (X6), no entanto após o primeiro ciclo os primeiros três vetores foram atribuídos ao *cluster* 3 e o quarto vetor ao *cluster* 6. Após a execução do segundo ciclo, os primeiros 3 vetores mantiveram-se no mesmo cluster e o quarto passou para o 3. Estando confirmada a semelhança entre estes quatro vetores, prova-se a necessidade da implementação deste segundo ciclo. Com isto cumpre-se o requisito 3 da Subsecção 2.2.1.

4.2.4 Remoção de variáveis redundantes

Semelhante como ocorre com as redes neuronais , teorizou-se inicialmente que a remoção de variáveis de entrada, que pouca informação nova introduziam, pudesse levar a que o algoritmo produzisse melhores resultados.

Analisando os gráficos de dispersão temos os seguintes resultados:

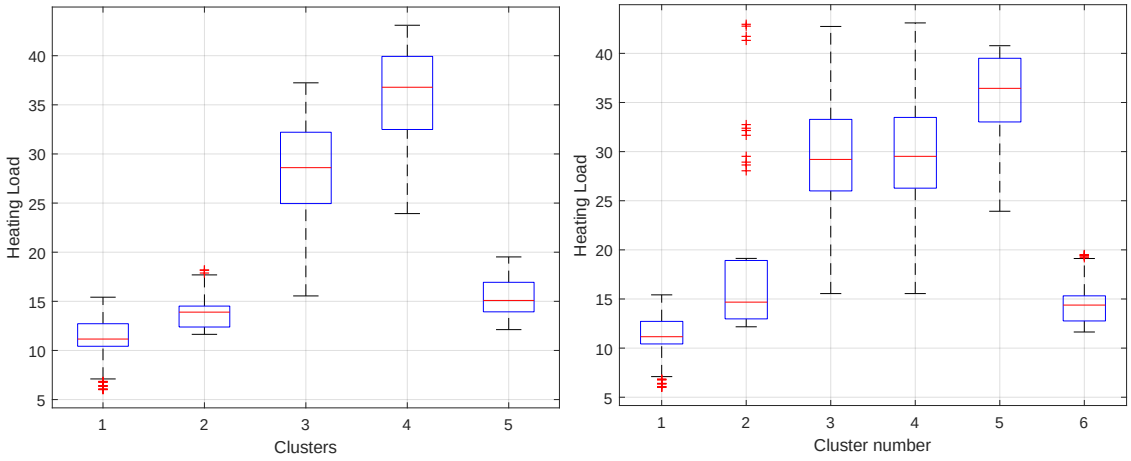


(a) Sem remoção de variáveis

(b) Após a remoção de X2 e X5

Figura 4.5: Comparação das dispersões das atribuições após a remoção de variáveis correlacionadas

Baseados nos resultados da análise das correlações o algoritmo foi executado inicialmente sem remoção de variáveis e de seguida com a remoção da Área da superfície (X2) e da Altura (X5). Tendo X2 uma correlação de 99% com X1 e tendo X5 uma correlação de 97% com X4 seria de esperar que não existissem grandes alterações nos resultados fruto destas remoções. No entanto verifica-se que após a remoção, a dispersão das atribuições face às saídas é intensificada, significando que cada *cluster* se relaciona cada vez menos com a saída Y1. A análise dos diagramas de caixa, relativos à mesma informação que os gráficos acima, revela a mesma conclusão com a presença de muito mais *outliers* dentro dos *clusters*.



(a) Sem remoção de variáveis

(b) Após remoção de X2 e X5

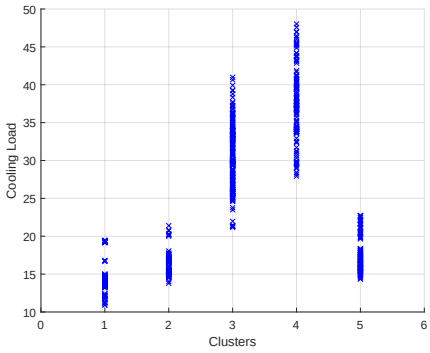
Figura 4.6: Comparação dos diagramas de caixa das atribuições antes e após a remoção de variáveis correlacionadas

No que toca à saída Y2 (Cooling load), deparamo-nos com um panorama muito semelhante. As atribuições pioram significativamente com a remoção destas variáveis como se pode observar na Figura 4.7.

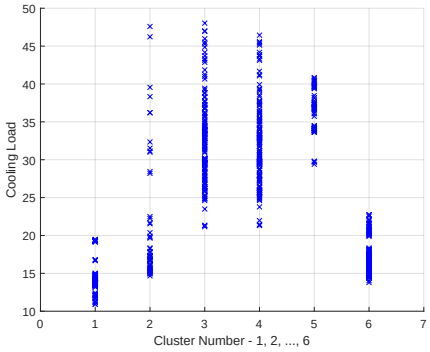
Por oposição a uma rede neuronal, o algoritmo implementado é baseado em comparações de distâncias. Apesar de as variáveis removidas estarem fortemente correlacionadas, esta não é completa o que significa que o algoritmo pode utilizar esses pequenos acréscimos de informação que cada uma introduz para diferenciar vetores de entrada semelhantes.

Foram realizadas também outras variações do algoritmo para validar estes resultados:

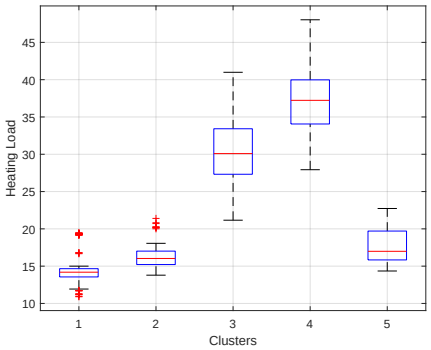
- Remoção de apenas X2;
- Remoção de apenas X1;
- Remoção de apenas X5;
- Remoção de apenas X4;
- Remoção de X2 e X4;
- Remoção de X1 e X5;



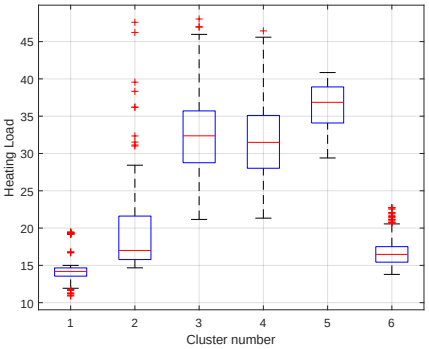
(a) Atribuições sem remoção de variáveis



(b) Atribuições após a remoção de X2 e X5



(c) Atribuições sem remoção de variáveis



(d) Atribuições após a remoção de X2 e X5

Figura 4.7: Comparação das atribuições antes e após a remoção de variáveis correlacionadas

- Remoção de X1 e X4.

Após a análise de todas estas variações de implementação verificou-se sempre a tendência a piorar os resultados em relação à situação original. Então concluiu-se inequivocamente que o algoritmo produz melhores resultados sem qualquer remoção de variáveis de entrada.

4.2.5 Escolha do primeiro detetor

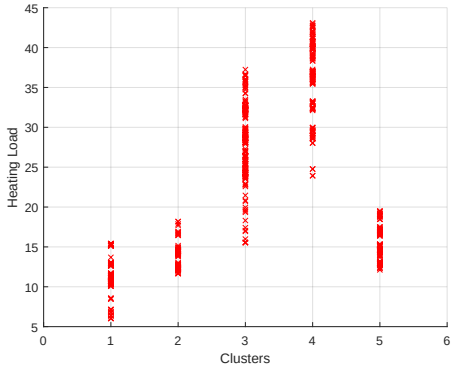
De forma a produzir um algoritmo robusto capaz de lidar com dados independentemente da ordem em que estes aparecem foram testados diversos parâmetros de escolha do detetor inicial que dá inicio ao processo de atribuições. Como caso base foi escolhido o vetor na posição 27, correspondente à carga térmica de aquecimento mais baixa. O resultado deste caso é apresentado na Figura 4.8a. Foi realizado um teste escolhendo também o vetor como a menor carga de arrefecimento como de detetor inicial e os resultados das atribuições são exatamente os mesmos que no caso anterior.

Para além destes dois casos foi implementado um parâmetro de escolha do detetor inicial de modo aleatório, a cada execução, que após algumas testes levou aos resultados apresentados na Figura 4.8b, Figura 4.8c e na Figura 4.8d.

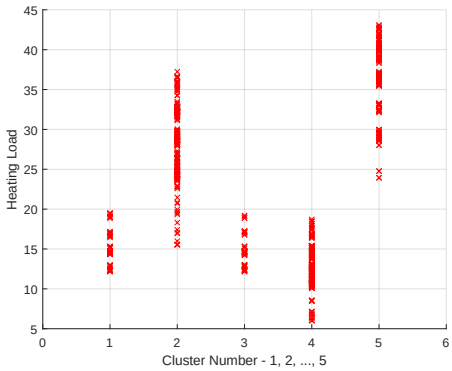
Os resultados das atribuições não são exatamente os mesmos, no entanto alguns *clusters* repetem-se com alguma semelhança entre os vários casos, apesar de aparecem por uma ordem

diferente. A título de exemplo temos o cluster 3 da Figura 4.8a que se apresenta com atribuições equivalentes no cluster 2 da Figura 4.8b, no cluster 5 da Figura 4.8c e novamente no cluster 2 da Figura 4.8d.

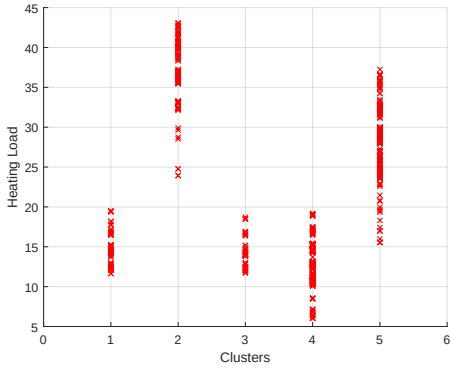
Por outro lado o *cluster* 4 da Figura 4.8c apresenta semelhanças com o *cluster* 4 da Figura 4.8d, mas com mais nenhum *cluster* das restantes figuras.



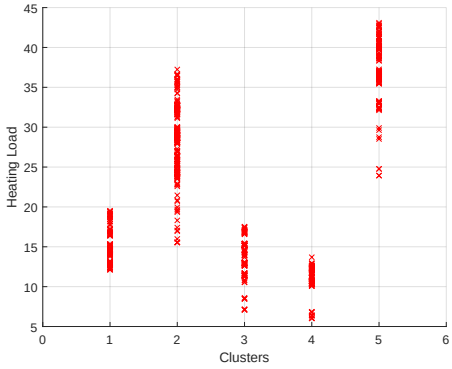
(a) Atribuições utilizando o detetor inicial do caso base



(b) Atribuições utilizando um detetor aleatório



(c) Atribuições utilizando um detetor aleatório

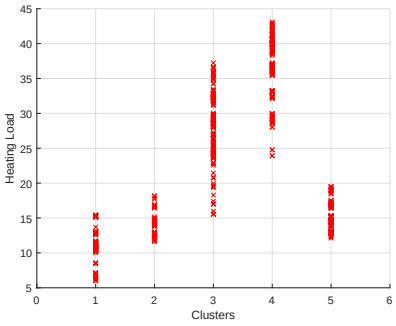


(d) Atribuições utilizando um detetor aleatório

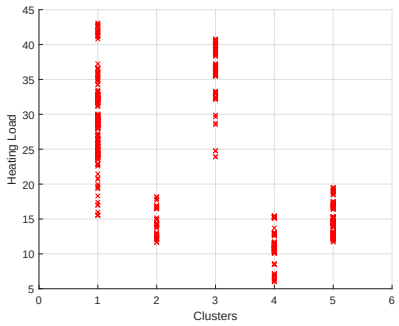
Figura 4.8: Comparação de vários *clusters* gerados com diferentes condições iniciais

O requisito quatro da Subsecção 2.2.1 indica que o algoritmo deve ser capaz de produzir *clusters* com características semelhantes, independentemente da ordem a que são gerados. No entanto este requisito não é universal, sendo que podem existir conjuntos de dados que não apresentem grupos facilmente separáveis. No caso em estudo devido à origem artificial dos dados, estamos perante tal caso, com um número considerável de casos gerados a partir de casos anteriores com pequenas variações em poucos parâmetros. Ainda que os *clusters* não apareçam exatamente iguais considera-se uma boa aproximação que pouca interferência vai causar nos protótipos dos *clusters* tal como se pode ver na Figura 4.9.

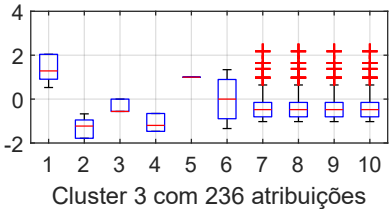
De modo a facilitar as comparações, ao longo do presente trabalho optou-se por definir o detetor inicial tal como descrito no caso base com a menor carga de aquecimento.



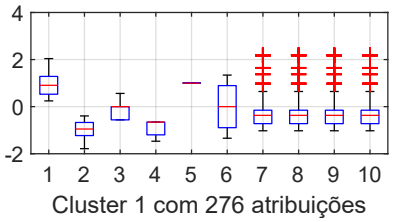
(a) Atribuições utilizando o detetor inicial do caso base



(b) Atribuições utilizando um detetor aleatório



(c) Protótipo do *cluster* 3 relativo ao caso base



(d) Protótipo do cluster 1 relativo à figura (b)

Figura 4.9: Comparação dos protótipos de dois *clusters* gerados com diferentes condições iniciais

4.2.6 Distâncias entre centróides

De forma a garantir que os centróides (protótipo de cada classe) encontrados apresentam características diferentes, foi implementado no algoritmo uma rotina para análise das distâncias entre estes, sob a forma de um gráfico de barras. Como se verifica pela Figura 4.10, todos os agrupamentos se distribuem pelo espaço de resultados relativamente bem, sendo que os dois *clusters* mais próximos serão o 3 e o 4. Isto vai de acordo com a dificuldade que se verifica no algoritmo em distinguir estes dois *clusters*, como se pode verificar nos gráficos de dispersão (por ex. na Figura 4.5).

Apesar disto considera-se que o objetivo da formação de grupos diferenciados foi atingido.

4.2.7 Dispersão dos resultados

Para se ter uma ideia do quão efetiva realmente foi a separação em *clusters* é necessário fazer a comparação entre as atribuições e a saída para ambos os casos. Assim o algoritmo a cada execução produz gráficos de dispersão e diagramas de caixa contendo esta informação. Relativamente à saída Y1 (Heating Load) temos a distribuição representada na Figura 4.11.

O *cluster* 3 é o mais abrangente ocupando uma parte significativa do espaço de resultados da saída. Relativamente ao *cluster* 4 tipicamente representa vetores com as saídas mais elevadas. Para os *clusters* 1, 2 e 5 temos uma correspondência com as saídas menos elevadas.

Analisando o diagrama de caixas na Figura 4.12 podemos ver que apesar de haver sobreposições relativamente à saída as medianas são distintas para todos os agrupamentos. Assim podemos concluir que edifícios que sejam agrupados no *cluster* 4 em média representam edifícios com a

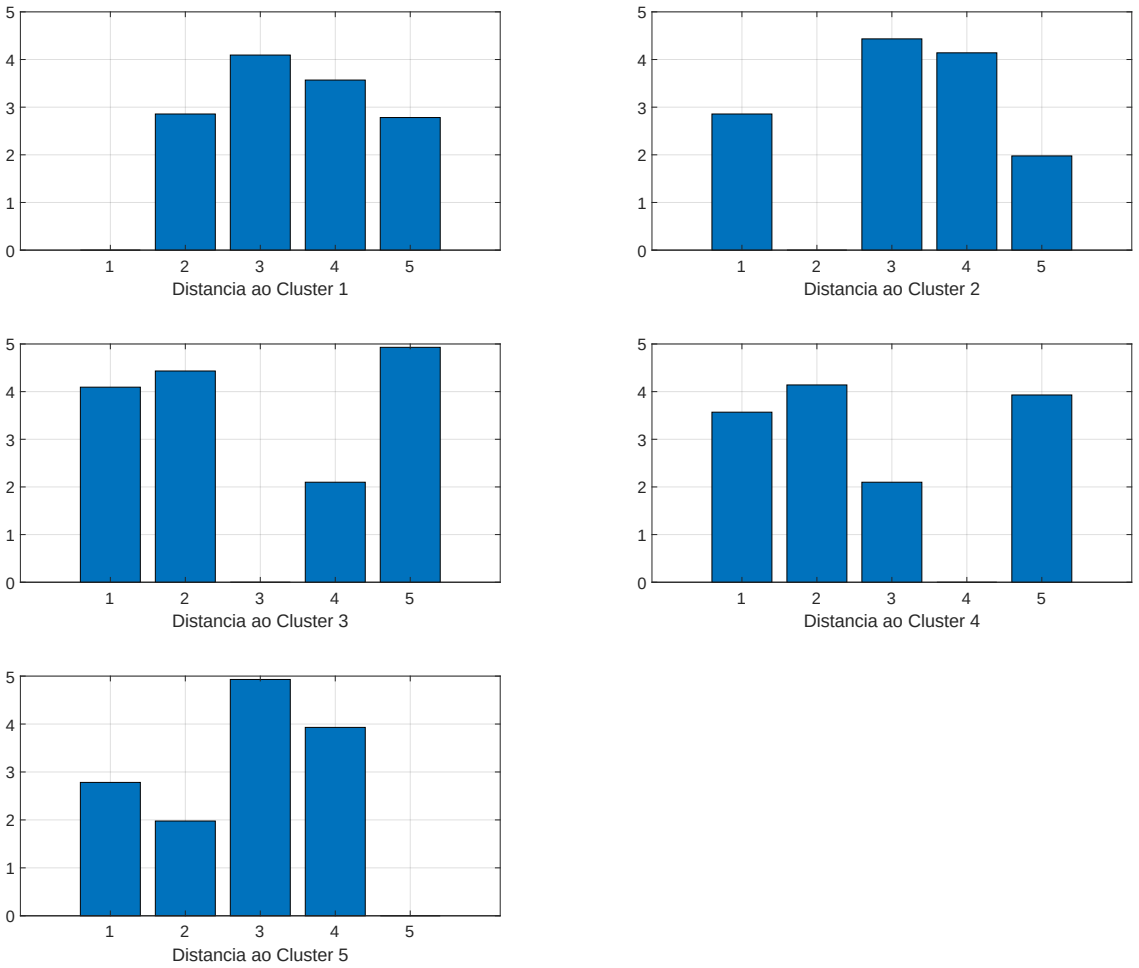


Figura 4.10: Distância de cada *cluster* aos restantes *clusters*

menor eficiência energética. Partindo deste grupo, por ordem decrescente de eficiência, temos o *cluster* 3, o *cluster* 5, o *cluster* 2 e finalmente o *cluster* 1 correspondendo aos edifícios mais eficientes.

Em relação às caixas delimitadas pelo primeiro e terceiro quartil também apresentam pouca sobreposição entre si, com a exceção do 2º e o 5º cluster que apresentam medianas muito próximas.

Partindo agora para uma análise semelhante relativamente à variável Y2 (Cooling Load), notamos a partir da Figura 4.13 que os resultados são muito semelhantes aos anteriores.

Em termos de consumo energético a ordem pelo qual aparecem os *clusters* é a mesma, com o *cluster* 4 a corresponder aos mais elevados e o *cluster* 1 aos mais baixos. No que toca ao diagrama de caixas a separação é também muito semelhante à anterior, desta vez apenas com mais outliers a aparecerem nos *clusters* 1 e 2. Finalmente outra tendência que se verifica é que estes edifícios apresentam consumos energéticos mais elevados no que toca ao arrefecimento.

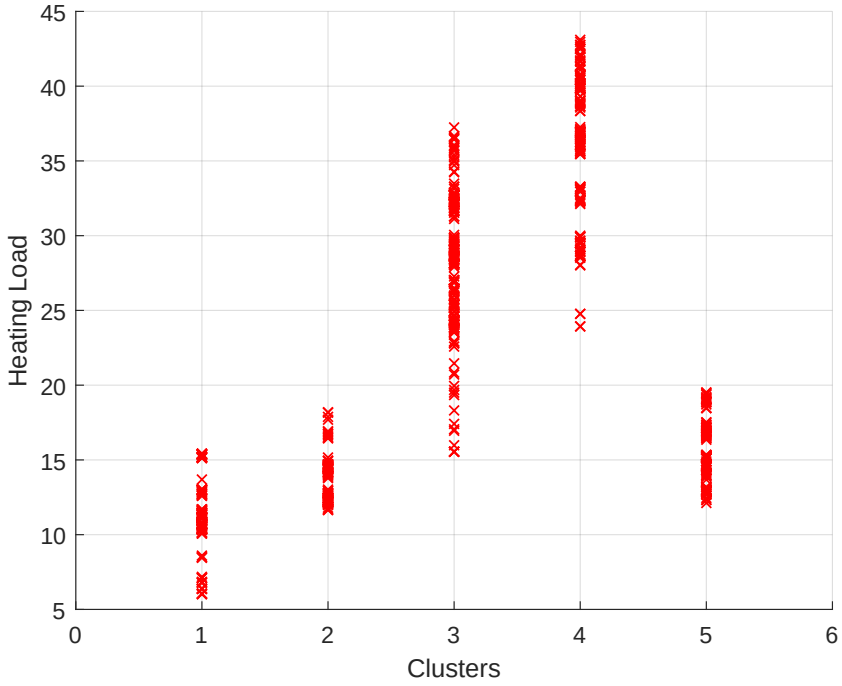


Figura 4.11: Comparação das atribuições com o consumo energético para aquecimento

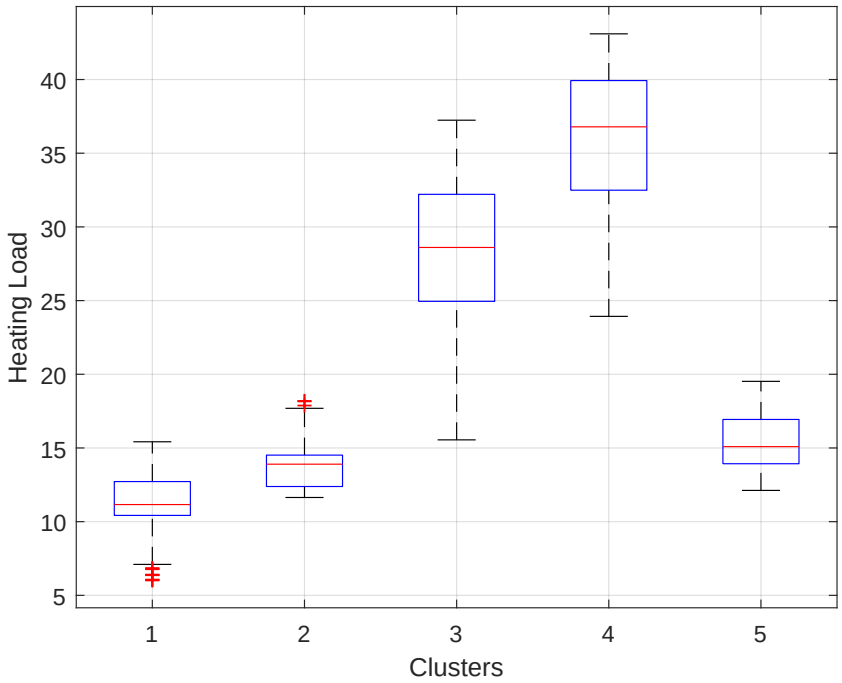
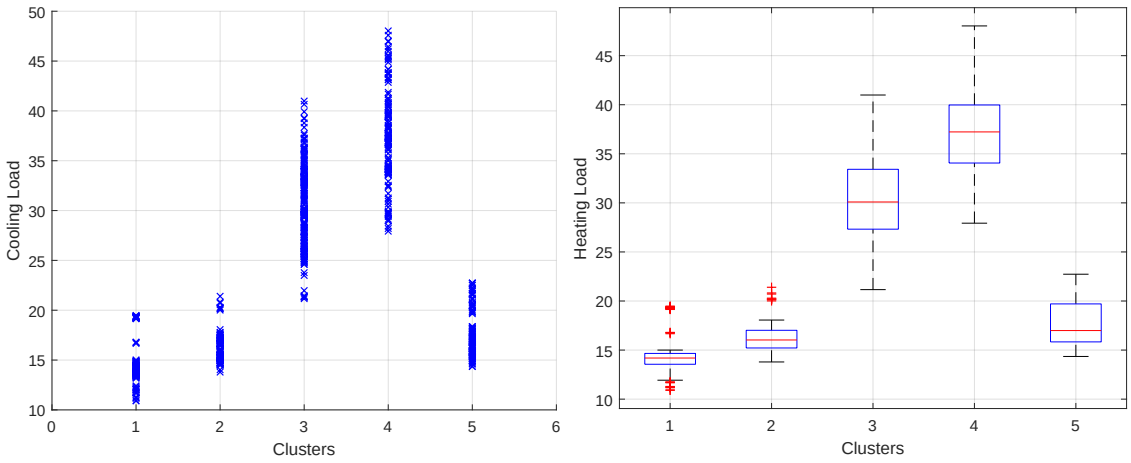


Figura 4.12: Comparação das atribuições com o consumo energético para aquecimento

4.2.8 Análise dos centróides

Nesta secção iremos analisar as características dos centróides apoiados em dois gráficos produzidos pelo algoritmo implementado. Em primeiro lugar temos o gráfico de barras que apresenta para cada centróide, a média das atribuições por variável (X1 a X10). Os valores das distâncias



(a) Gráfico de dispersão (b) Diagrama de caixa

Figura 4.13: Comparação das atribuições com o consumo energético para arrefecimento

no eixo *yy* vem dos valores dos vetores normalizados, pelo que estas distâncias aparecem também normalizadas.

Baseados na Figura 4.14 imediatamente podemos visualizar que a variável X5 (Altura) apresenta valores mais baixos para os *clusters* 1, 2 e 5, e mais elevados para os *clusters* 3 e 4. Olhando novamente para o gráfico de dispersão da Figura 4.11 notamos imediatamente que os *clusters* 3 e 4 correspondem a edifícios que tipicamente apresentam uma eficiência energética mais baixa (maior carga térmica para aquecimento).

Seguindo para a próxima variável, X2 (Área da superfície), nota-se uma tendência semelhante em que os *clusters* 1, 2 e 5 apresentam uma tendência para valores mais elevados contrastando com os valores mais reduzidos dos *clusters* 3 e 4.

No que toca à variável X1 (Compacticidade) a tendência é também relativamente semelhante. O *cluster* 3 apresenta os maiores valores de da variável X1 (ou seja, menor compacticidade) seguido do *cluster* 4 que apresenta valores de compacticidade ligeiramente acima da média. Os restantes *clusters* apresentam todos valores abaixo da média sendo o cluster 5 o que apresenta o valor mais baixo.

Para a variável X4 (Área do telhado) novamente a mesma tendência, com as áreas dos telhados mais elevadas a pertencer aos *clusters* 1, 2 e 5 e as mais baixas aos *clusters* 3 e 4.

Com isto termina a análise das variáveis que permitem uma associação mais direta com as cargas térmicas dos diferentes *clusters*.

Seguindo para a análise das restantes variáveis, para X6 (Orientação), temos que a média para todos os *clusters* é zero. Sabendo que esta variável apresenta quatro valores distintos, podemos concluir que para cada *cluster* foi atribuído igual número de edifícios com cada um dos possíveis valores. Esta conclusão será reforçada na análise dos diagramas de caixa (Figura 4.15).

No que toca às áreas envidraçadas temos que o *cluster* 1 apresenta em média valores mais reduzidos segundo as quatro direções, sendo provavelmente característica destes edifícios apresentar as menores áreas envidraçadas totais. Os *clusters* 3 e 4 parecem apresentar valores semelhantes

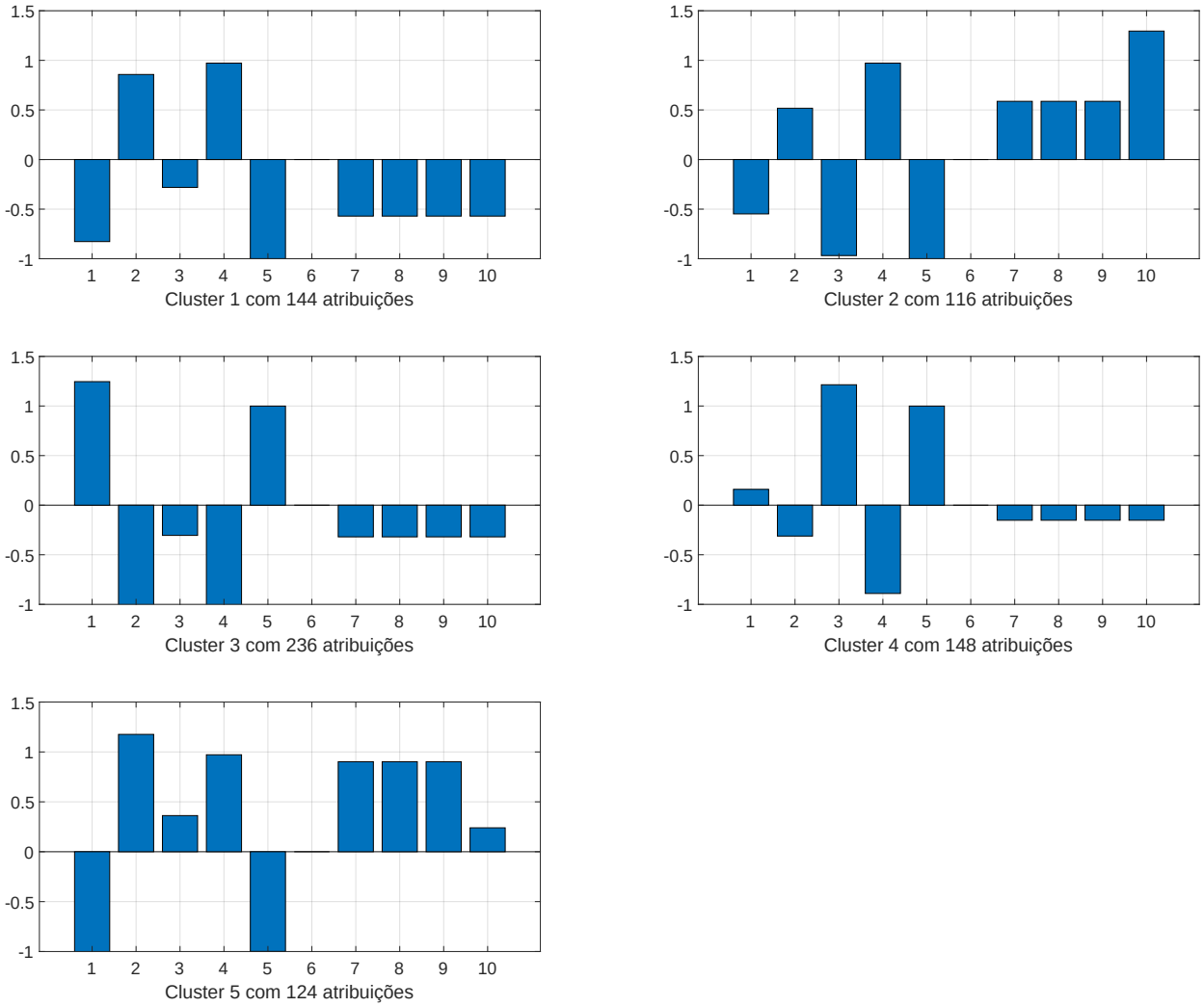


Figura 4.14: Gráfico de barras com as características de cada *cluster*

segundo as quatro direções, enquanto que o *cluster* 2 apresenta valores relativamente elevados para as áreas, sendo a variável área envidraçada a oeste um valor destacado das restantes. Para o *cluster* 5, temos uma situação semelhante em que as áreas são relativamente elevadas e a área envidraçada a oeste se destaca mas desta vez com valores mais reduzidos.

Como a média nem sempre é um indicador perfeito dos dados contidos nos agrupamentos a análise dos centróides foi complementada com um diagrama de caixa conforme a Figura 4.15. Novamente, no eixo *yy* apresentam-se valores normalizados.

No que toca às variáveis X1, X2, X4 e X5 a análise do diagrama leva às mesmas conclusões. Os valores da média (lida no gráfico de barras) e da mediana (linha vermelha no diagrama de caixa) são praticamente coincidentes entre ambos os gráficos, com dispersões relativamente reduzidas e sem *outliers*.

No caso da variável X6 (Orientação) vemos que a mediana do diagrama coincide em todos os casos com o valor da média encontrado no gráfico de barras (zero) e, além disso, através do dia-

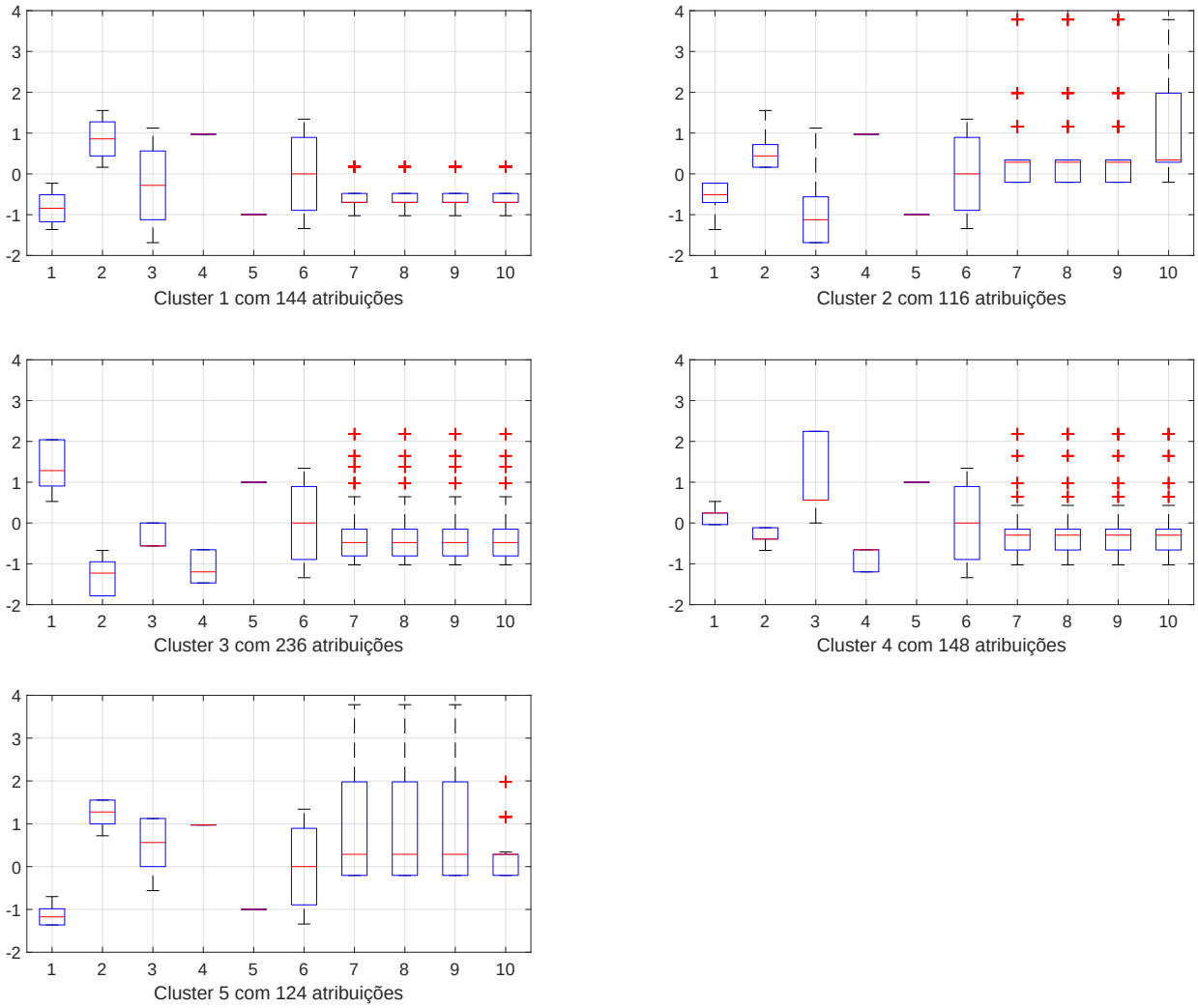


Figura 4.15: Diagrama de caixa com as características de cada *cluster*

grama podemos ver ainda que a dispersão dos valores é a mesma para todos os *clusters*, validando assim a conclusão retirada pela análise do gráfico de barras.

Em relação às áreas envidraçadas as conclusões são semelhantes salvo certas exceções. Para a área envidraçada a oeste (X10), relativamente ao *cluster* 2, temos que a média se destaca bastante do valor da mediana, no entanto mantém-se a conclusão que neste grupo X10 é mais elevado que o normal. Para a mesma variável, no cluster 5, nota-se uma tendência semelhante. Para finalizar temos que no geral praticamente todas estas áreas apresentam uma quantidade considerável de *outliers*.

Com base nas observações desta secção podemos verificar que as variáveis X1 (Compacticidade), X2 (Área da superfície), X4 (Área do telhado) e X5 (Altura) apresentam valores distintos entre os vários clusters *clusters*. Relativamente às restantes variáveis não é possível achar uma relação entre o consumo e valores característicos que estas apresentem. Posto isto pode-se concluir que as quatro variáveis mencionadas poderão ter mais relevância na definição dos *clusters* do que

as restantes.

4.2.9 Análise dos resultados finais

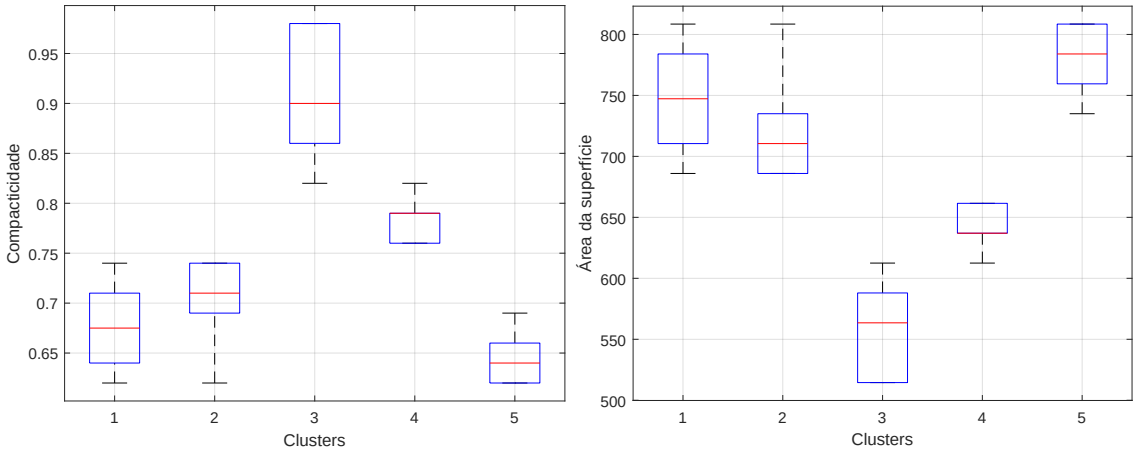
Nesta secção iremos realizar uma análise dos valores não normalizados das atribuições em cada capítulo. Para cada variável de entrada serão calculados o mínimo, o máximo e a média para cada *cluster*, o que nos irá dar uma base para criar a árvore de decisão.

A Tabela 4.1 apresenta os valores mínimos, médios e máximos para as variáveis X1 (Compacticidade) e X2 (Área da superfície interna).

Tabela 4.1: Valores notáveis para X1 e X2, por *cluster*

	X1 - Compacticidade		
	Mínimo	**Média**	**Máximo**
Cluster 1	0.62	0.68	0.74
Cluster 2	0.62	0.71	0.74
Cluster 3	0.82	0.90	0.98
Cluster 4	0.76	0.78	0.82
Cluster 5	0.62	0.65	0.69

	X2 - Área da superfície		
	Mínimo	**Média**	**Máximo**
Cluster 1	686	747	808
Cluster 2	686	717	808
Cluster 3	514	566	612
Cluster 4	612	644	661
Cluster 5	735	775	808



(a) Distribuição da variável X1

(b) Distribuição da variável X2

Figura 4.16: Comparação da distribuição de duas variáveis correlacionadas

Estes valores permitem identificar uma tendência, confirmada pela análise da correlação, que estas duas variáveis praticamente não introduzem informação adicional.

Para implementação da árvore de decisão, basta apenas analisar uma das variáveis. Dividimos então o espaço de resultados de X1 em dois, o que nos vai permitir a implementação da mesma. O primeiro grupo com compacticidade alta que corresponde a valores superiores acima de 0.75 e segundo para valores abaixo. Esta separação vai, mais tarde, permitir implementar a árvore de decisão ao diferenciar os *clusters* 1, 2 e 5 dos *clusters* 3 e 4.

Para a variável X3 (Área das paredes) temos:

Tabela 4.2: Valores notáveis para X3, por *cluster*

X3 - Área das paredes	**Mínimo**	**Média**	**Máximo**
Cluster 1	245	306	367
Cluster 2	245	276	367
Cluster 3	294	305	318
Cluster 4	318	371	416
Cluster 5	294	334	367

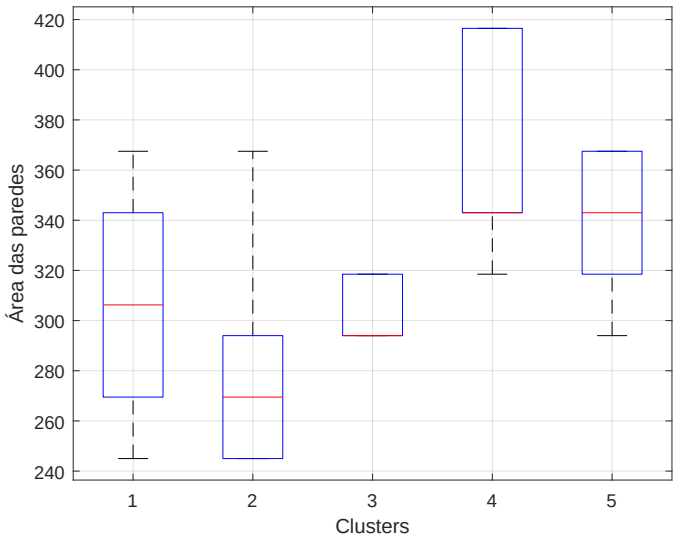


Figura 4.17: Distribuição da variável X3

Pela análise dos valores da tabela e da Figura 4.17 podemos concluir que esta variável pode servir apenas de fator de diferenciação entre o *cluster* 3 e 4, pois entre os restantes existe bastante sobreposição.

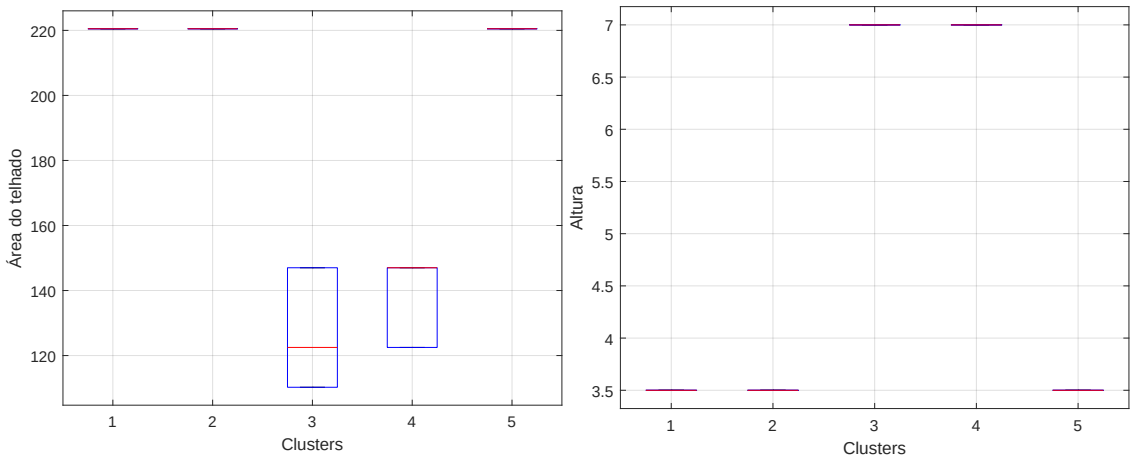
Para a variável X4 (Área do telhado) e X5 (Altura) temos os valores:

Tabela 4.3: Valores notáveis para X4 e X5, por *cluster*

X4 - Área do telhado	**Mínimo**	**Média**	**Máximo**
Cluster 1	220	220	220
Cluster 2	220	220	220
Cluster 3	110	130	147
Cluster 4	110	130	147
Cluster 5	220	220	220

X5 - Altura	**Mínimo**	**Média**	**Máximo**
Cluster 1	3.5	3.5	3.5
Cluster 2	3.5	3.5	3.5
Cluster 3	7	7	7
Cluster 4	7	7	7
Cluster 5	3.5	3.5	3.5

Mais uma vez note-se que para estas variáveis correlacionadas não apresentam informação nova no que toca a diferenciar os *clusters*. Pela análise da Figura 4.18b podemos concluir que a Altura permite separar unicamente os *clusters* 1, 2 e 5 dos *clusters* 3 e 4.



(a) Distribuição da variável X4

(b) Distribuição da variável X5

Figura 4.18: Comparação da distribuição de duas variáveis correlacionadas

Relativamente à Orientação (X6) basta apenas uma análise do diagrama de caixa na Figura 4.21 para perceber que esta variável não irá ajudar no processo de construção da árvore de decisão.

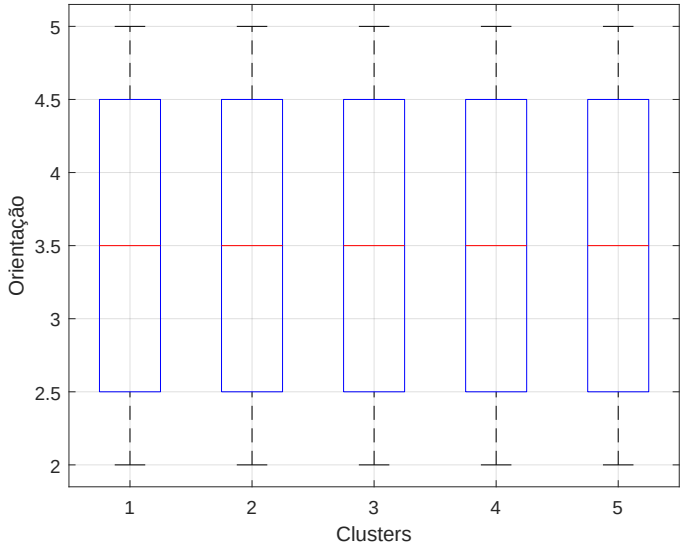


Figura 4.19: Distribuição da variável X6

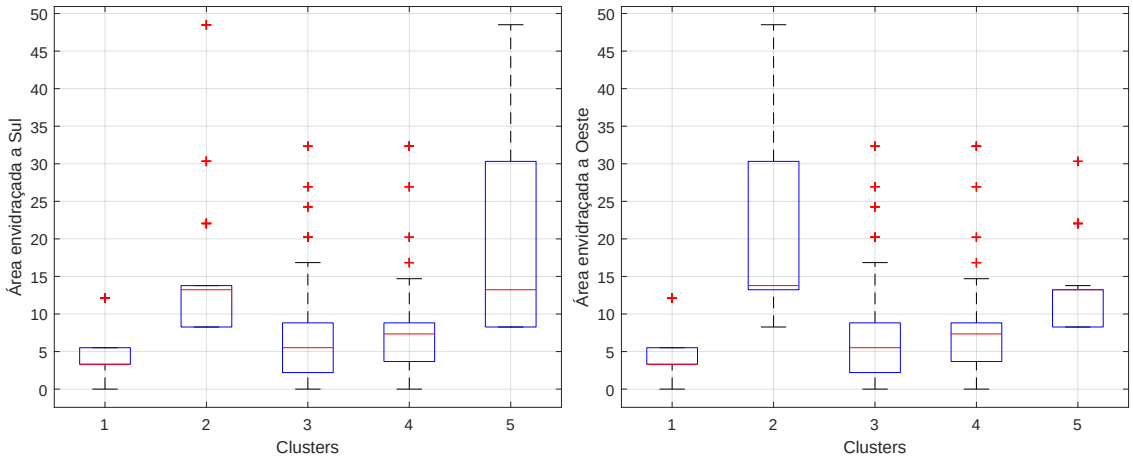
Relativamente às Áreas envidraçadas, uma análise da Figura 4.15 indica-nos que as suas distribuições apresentam praticamente os mesmos valores entre si, exceto para X10 (Área envidraçada a Oeste). Então para X7 (Área envidraçada a sul) e X10 (Área envidraçada a Oeste) temos a distribuição de valores:

Apoiados na análise dos valores das variáveis X7 a X9 e da Figura 4.20 podemos concluir numa primeira fase que estas variáveis permitem possivelmente a distinção entre os *clusters* 2 e 5 do *cluster* 1, se ignorarmos os *outliers* presentes no *cluster* 1. Uma análise mais detalhada do diagrama de caixa revela a presença de 24 pontos (cruz a vermelho) relativo ao *cluster* 1. Este número apesar de reduzido ainda é significativo.

Tabela 4.4: Valores notáveis para X7 e X10, por *cluster*

X7 - Área envidraçada a Sul	**Mínimo**	**Média**	**Máximo**
Cluster 1	0	4	12
Cluster 2	8	16	48
Cluster 3	0	7	32
Cluster 4	0	8	32
Cluster 5	8	19	48

X10 - Área envidraçada a Oeste	**Mínimo**	**Média**	**Máximo**
Cluster 1	0	4	12
Cluster 2	8	23	48
Cluster 3	0	7	32
Cluster 4	0	8	32
Cluster 5	8	12	30



(a) Distribuição da variável X7

(b) Distribuição da variável X10

Figura 4.20: Comparação da distribuição de duas Áreas envidraçadas

Para contornar a existência destes *outliers* foi realizada uma análise extra, com o cálculo da área total envidraçada. Esta área corresponde à soma das áreas envidraçadas segundo as quatro direções e os resultados apresentam-se na Figura 4.21. Com esta análise a separação entre o *cluster* 1 e os *clusters* 2 e 5 não apresenta sobreposições, tendo os *outliers* sido eliminados.

4.2.10 Tempo de execução

Como indicador adicional da performance do algoritmo nesta secção indica-se os tempos de execução bem como o sistema em que foi implementado.

O hardware utilizado foi um Intel® Core™ i5-3320M 2.6 GHz, lançado em 2012. O software foi implementado e executado numa máquina virtual utilizando o *software Oracle VM VirtualBox* a correr o *Windows 7 64 bits* com 4 Gb de memória ram e um *core* alocado. O tempo total de execução do algoritmo foi cerca de 23 segundos. Desse total, os tempos parciais foram:

- Para leitura dos dados de entrada de 2.6 segundos;
- Para recodificação dos dados de 0.1 segundos;
- Para execução dos ciclos de *clustering* de 0.3 segundos;
- Para criação de todos os gráficos de 12 segundos

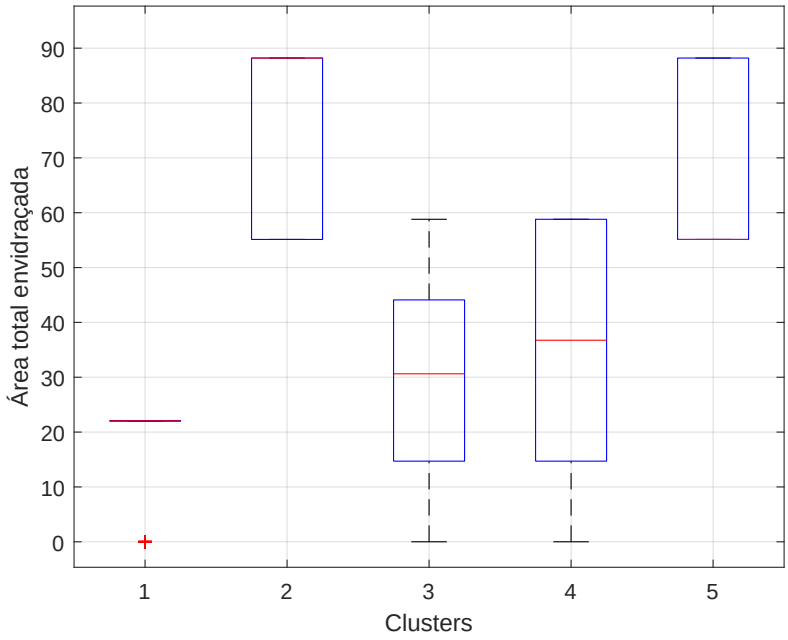


Figura 4.21: Distribuição das Áreas totais envidraçadas

- Para escrita dos resultados para a folha de cálculo de 8 segundos;

O algoritmo trabalhou sobre uma matriz de [768x10] que totaliza 7.680 valores individuais. Todos estes valores foram comparados seis vezes com os detetores encontrados o que totaliza 46.080 comparações durante o primeiro ciclo. Para serem encontrados novos detetores foram feitas comparações entre todos os detetores atuais e todos os elementos de entrada, totalizando 115.200 comparações. Durante o segundo ciclo foram realizadas mais cinquenta comparações entre todos os vetores de entrada e todos os centróides encontrados totalizando 1.920.000 comparações.

Com um número de operações considerável e um tempo de execução baixo, concluímos que o algoritmo foi implementado de maneira eficaz e que pode ser facilmente escalável para abranger grandes quantidades de dados, mantendo um tempo de execução aceitável. Com isto o primeiro requisito para uma boa implementação, como descrito na Subsecção 2.2.1, está garantido.

4.3 Redes Neuronais e sensibilidades

De forma a evitar variáveis que contenham informação igual ou muito semelhante, foram criadas várias redes neuronais com combinações de remoção das variáveis redundantes para cada uma das variáveis de saída. As alternativas experimentadas foram:

- **Rede 1** - Sem remoções;
- **Rede 2** - Remoção de X2 (Área da superfície);
- **Rede 3** - Remoção de X4 (Área do telhado);

- **Rede 4** - Remoção de X4 (Área do telhado) e X2 (Área da superfície)

4.3.1 RN1 - Sem remoções

Em relação à rede neuronal 1, sem remoções, os índices de performance obtidos em relação à saída Y1 (Heating Load), foram:

MSE (Testing) = 0.1877 R (Testing) = 0.9991
MAPE (Global) = 1.472

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X4** (Área do telhado) > **X2** (Área da superfície)

Para a mesma rede neural sem remoções, mas desta vez para Y2 (Cooling Load) temos:

MSE (Testing) = 1.8513 R (Testing) = 0.9896
MAPE (Global) = 15.216

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X4** (Área do telhado) > **X2** (Área da superfície)

Os desvios numéricos dos valores reais em relação aos valores obtidos pela rede neuronal pode ser avaliados na Figura 4.22. Para se compreender a dimensão dos desvios relativamente ao valor das variáveis apresenta-se também a comparação entre os valores reais e os saídos da rede neuronal, para uma seleção de vetores. A seleção escolhida foram os vetores 195 e 220, que aparentam ter desvios mais elevados. Como se pode ver na Figura 4.23 e Figura 4.24 os desvios são pouco significativos relativamente aos valores das variáveis.

4.3.2 RN2 - Remoção de X2

Em relação à rede neuronal 2, com remoção de X2, os índices de performance obtidos em relação à saída Y1 (Heating Load), foram:

MSE (Testing) = 0.1638 R (Testing) = 0.9992
MAPE (Global) = 1.4251

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X4** (Área do telhado) > **X3** (Área das paredes)

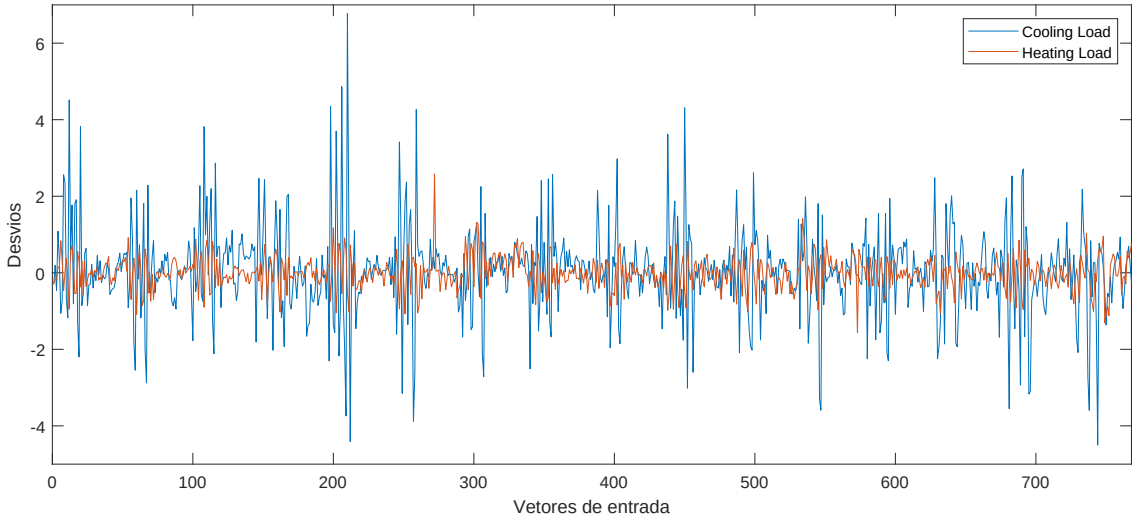


Figura 4.22: RN1 - Desvios de cada entrada relativamente ao dados reais

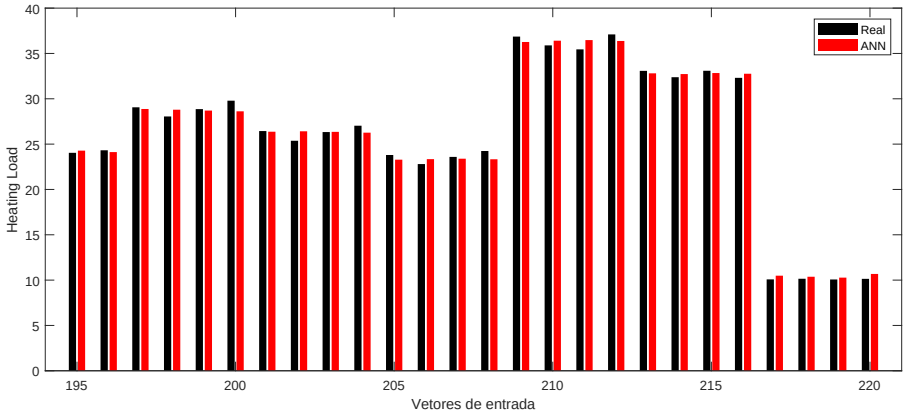


Figura 4.23: RN1 - Comparação de uma seleção de entradas relativamente ao dados reais

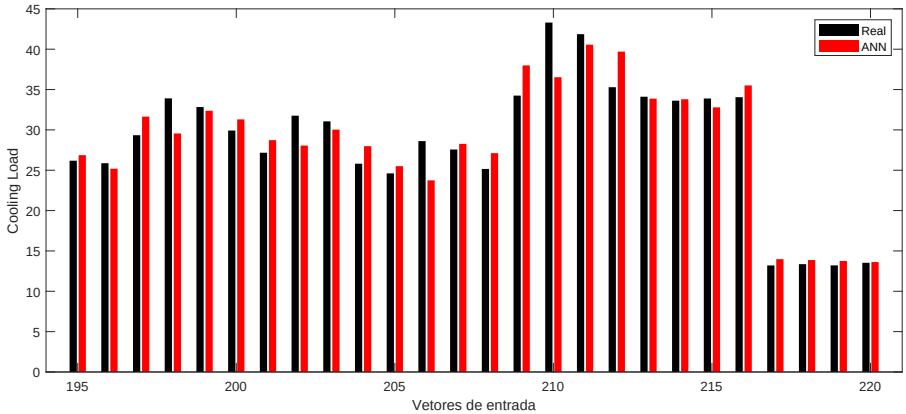


Figura 4.24: RN1 - Comparação de uma seleção de entradas relativamente ao dados reais

Para a mesma rede neural mas desta vez para Y2 (Cooling Load) temos:

MSE (Testing) = 1.246 R (Testing) = 0.9934
MAPE (Global) = 15.617

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X4** (Área do telhado) > **X3** (Área das paredes)

Os desvios dos valores reais em relação aos valores obtidos pela rede neuronal pode ser avaliados na Figura 4.25. Novamente, apresentam-se os gráficos que fazem a comparação dos valores reais com os da rede neuronal, na Figura 4.26 e Figura 4.27, que comprovam que os desvios são pouco significativos. A seleção dos valores foi a mesma, ou seja, os vetores 195 a 220.

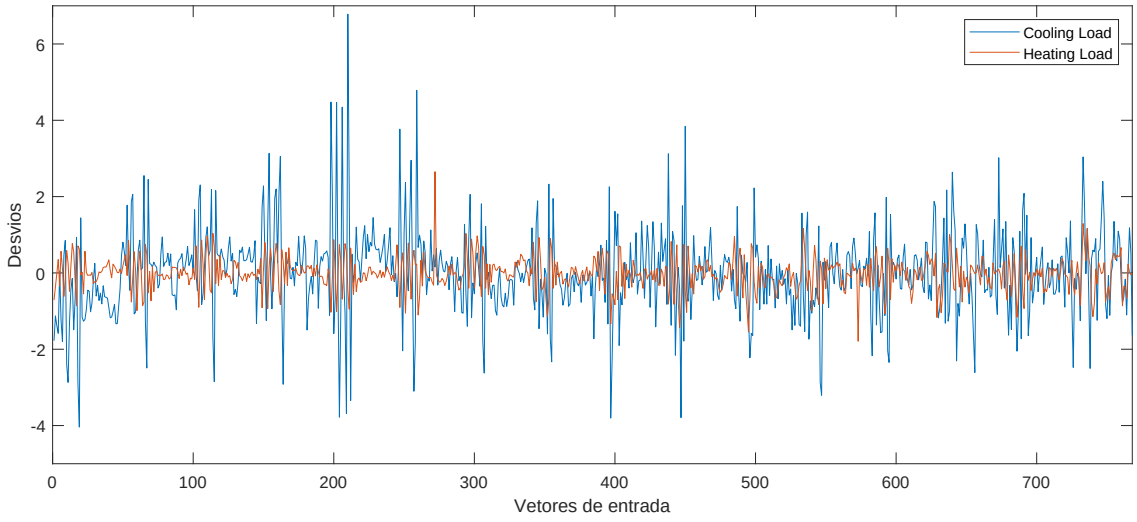


Figura 4.25: RN2 - Desvios de cada entrada relativamente ao dados reais

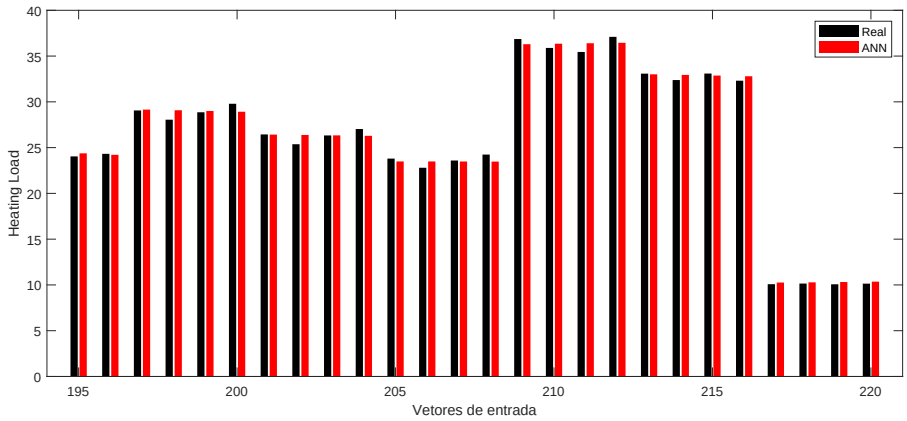


Figura 4.26: RN2 - Desvios de cada entrada relativamente ao dados reais

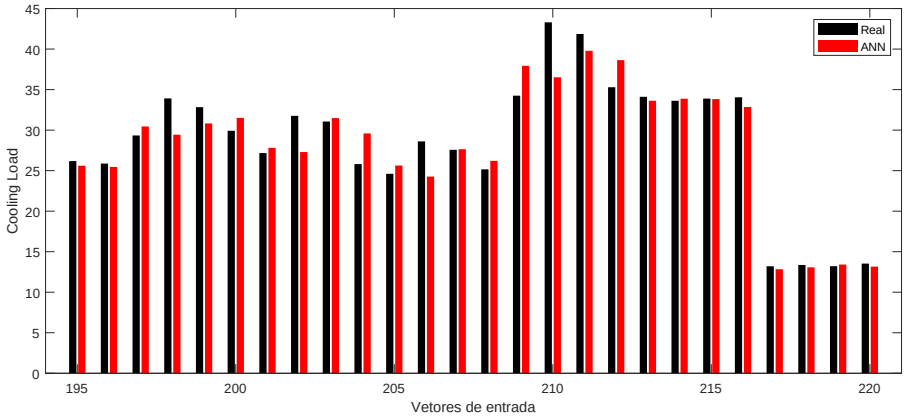


Figura 4.27: RN2 - Desvios de cada entrada relativamente ao dados reais

4.3.3 RN3 - Remoção de X4

Em relação à rede neuronal 2, com remoção de X4 (Área do telhado), os índices de performance obtidos em relação à saída Y1 (Heating Load), foram:

MSE (Testing) = 0.2039 R (Testing) = 0.9989
MAPE (Global) = 1.4631

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X2** (Área da superfície) > **X3** (Área das paredes)

Para a mesma rede neural mas desta vez para Y2 (Cooling Load) temos:

MSE (Testing) = 2.073 R (Testing) = 0.9883
MAPE (Global) = 15.8969

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X6** (Orientação) > **X2** (Área da superfície)

Os desvios dos valores reais em relação aos valores obtidos pela rede neuronal pode ser avaliados na Figura 4.28.

4.3.4 RN4 - Remoção de X2 e X4

Em relação à rede neuronal 2, com remoção de X4 (Área do telhado), os índices de performance obtidos em relação à saída Y1 (Heating Load), foram:

MSE (Testing) = 0.2915 R (Testing) = 0.9985
MAPE (Global) = 1.615

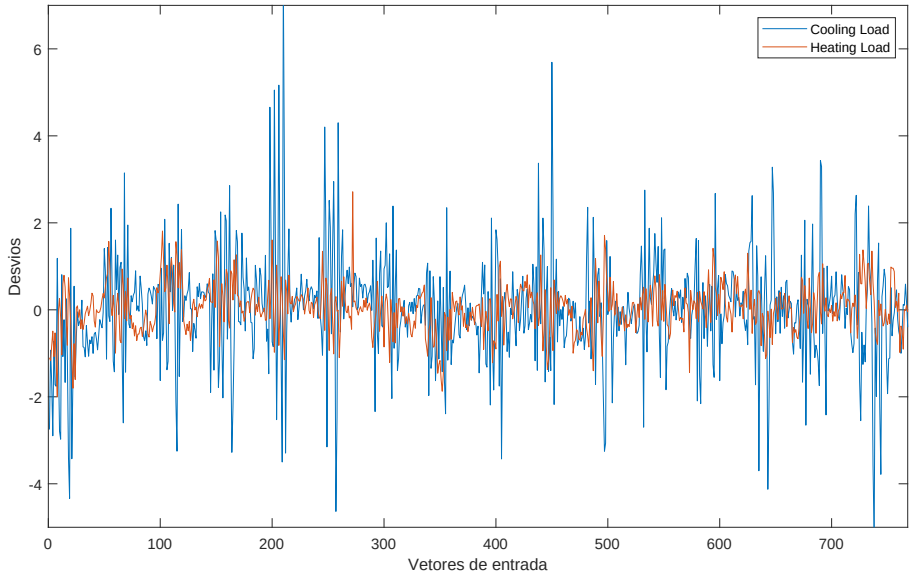


Figura 4.28: RN3 - Desvios de cada entrada relativamente ao dados reais

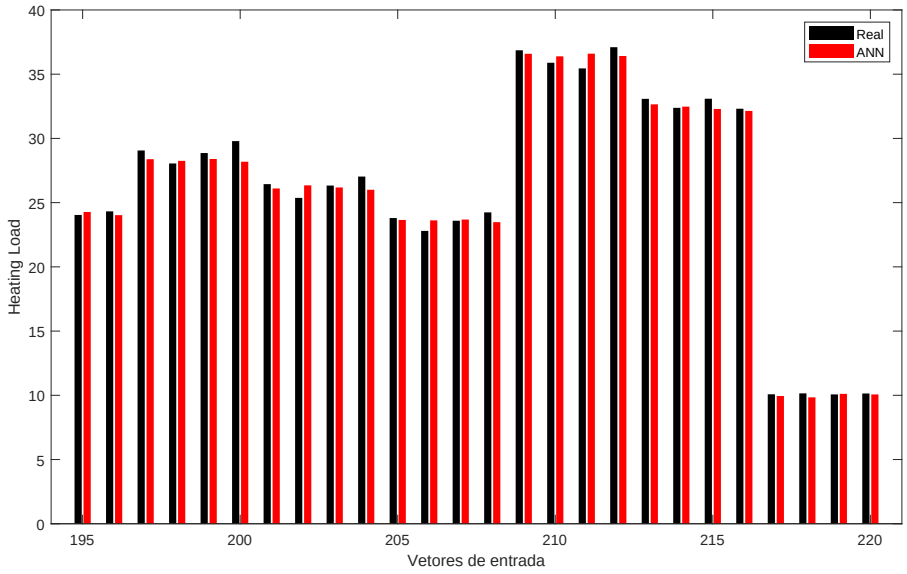


Figura 4.29: RN3 - Desvios de cada entrada relativamente ao dados reais

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X3** (Área das paredes)

Para a mesma rede neural mas desta vez para Y2 (Cooling Load) temos:

MSE (Testing) = 2.080 R (Testing) = 0.9880
MAPE (Global) = 15.6326

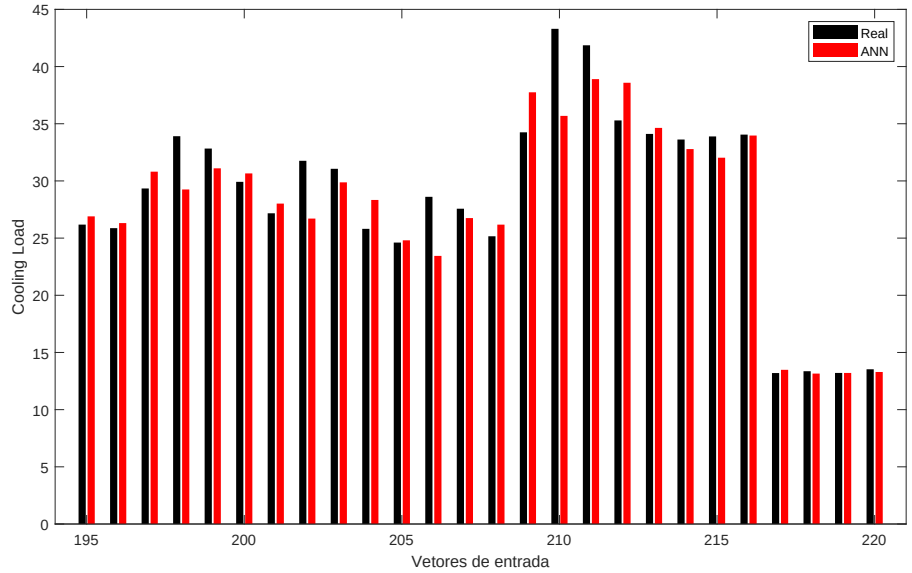


Figura 4.30: RN3 - Desvios de cada entrada relativamente ao dados reais

Para esta rede neuronal, a análise das sensibilidades permitiu concluir que as variáveis mais importantes, ordenadas por ordem decrescente de importância serão:

X1 (Compacticidade) > **X5** (Altura) > **X3** (Área das paredes)

Os desvios dos valores reais em relação aos valores obtidos pela rede neuronal pode ser avaliados na Figura 4.31. A comparação com dos valores da previsão com os reais é feita desta vez entre os vetores 429 a 454, na Figura 4.32 e Figura 4.33.

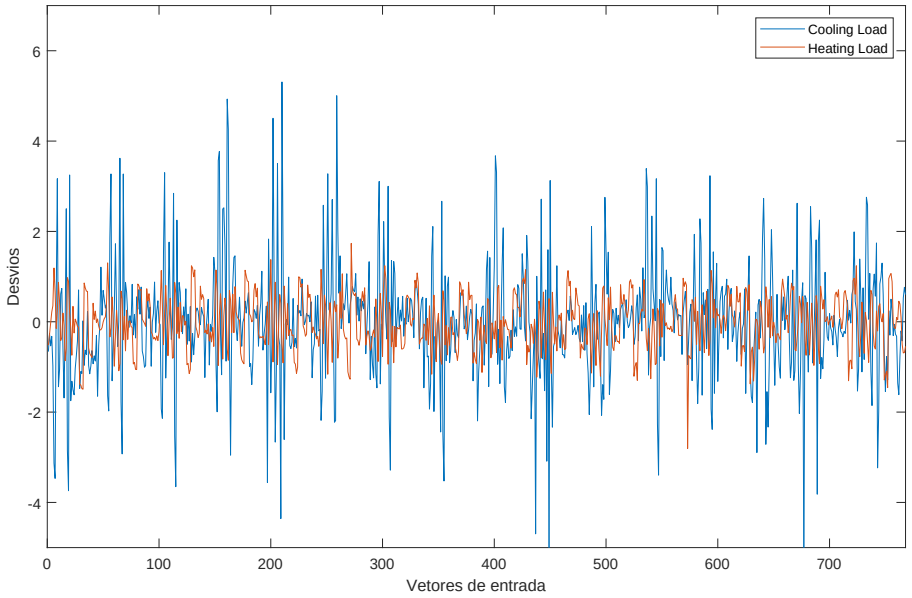


Figura 4.31: RN4 - Desvios de cada entrada relativamente ao dados reais

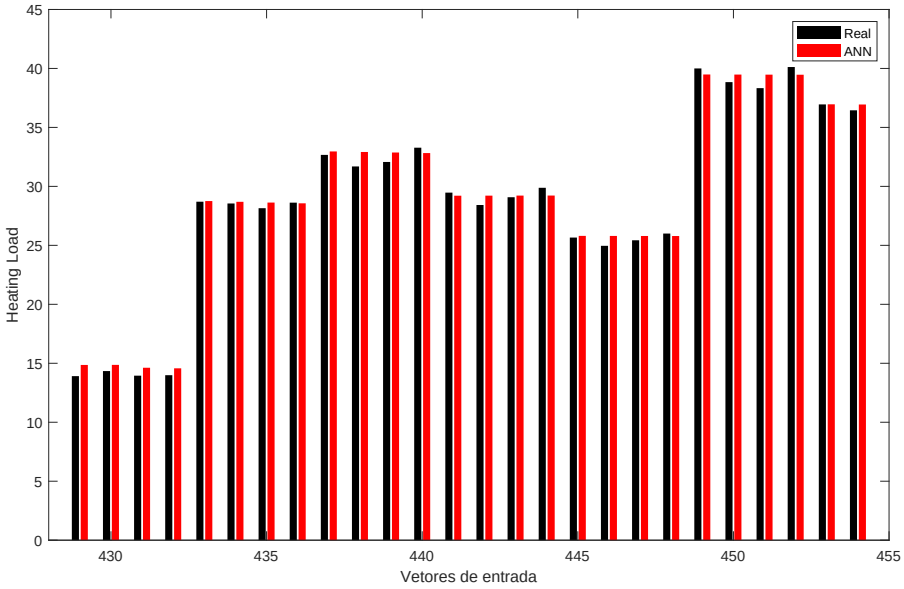


Figura 4.32: RN4 - Desvios de cada entrada relativamente ao dados reais

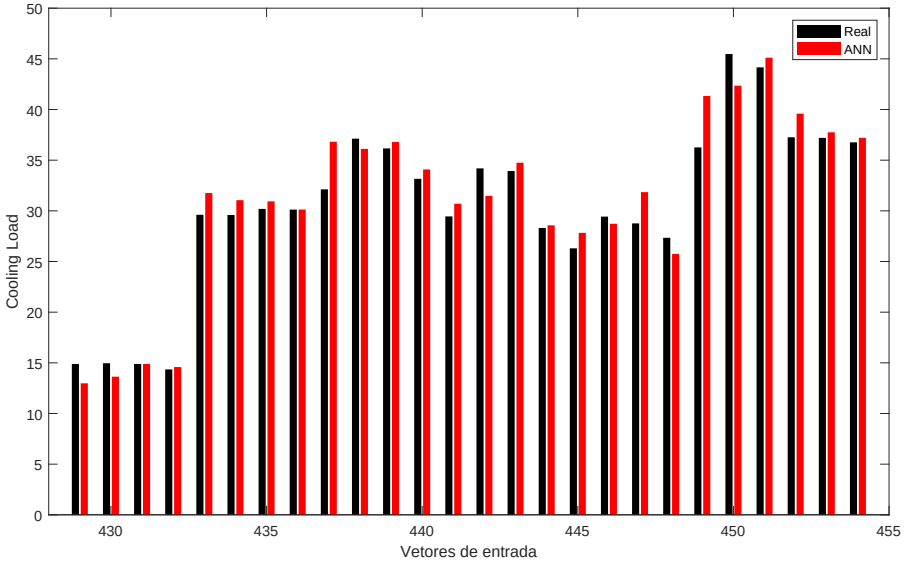


Figura 4.33: RN4 - Desvios de cada entrada relativamente ao dados reais

4.4 Análise dos Resultados

Resumindo os resultados das redes neuronais, temos que as variáveis X1 (Compacticidade), X5 (Altura) e X3 (Área das paredes) aparecem consistentemente como mais importantes para a previsão das saídas. Estes resultados vão de encontro a uma conclusão mais empírica que se retirou dos resultados obtidos da análise dos centróides. Com isto retira-se também uma ordem de importância que nos vai ajudar a implementar uma árvore de decisão.

Relativamente aos resultados do algoritmo temos as seguintes possibilidades de separação de clusters:

- A compacticidade (X1) permite separar os *clusters* 3 e 4 dos *clusters* 1, 2 e 5;
- A Área das paredes (X3) permite separar o *cluster* 3 do 4;
- A Altura (X5) permite separar os *clusters* 3 e 4 dos *clusters* 1, 2 e 5;
- A variável correspondente ás áreas envidraçadas totais permitem separar os *clusters* 2 e 5 dos *clusters* 1, 3 e 4.

Com isto, fica possível a separação de todos os *clusters* exceto o 2 e 5. Uma análise mais aprofundada é necessária para a implementação da árvore de decisão.

4.5 Árvore de decisão

De forma a realizar a separação entre os *clusters* 2 e 5 foi implementado um algoritmo simples no software *RStudio* ® uma árvore de decisão com o objetivo de separar estes *clusters*.

Os resultados obtidos ao longo do trabalho foram condensados na Figura 4.34.

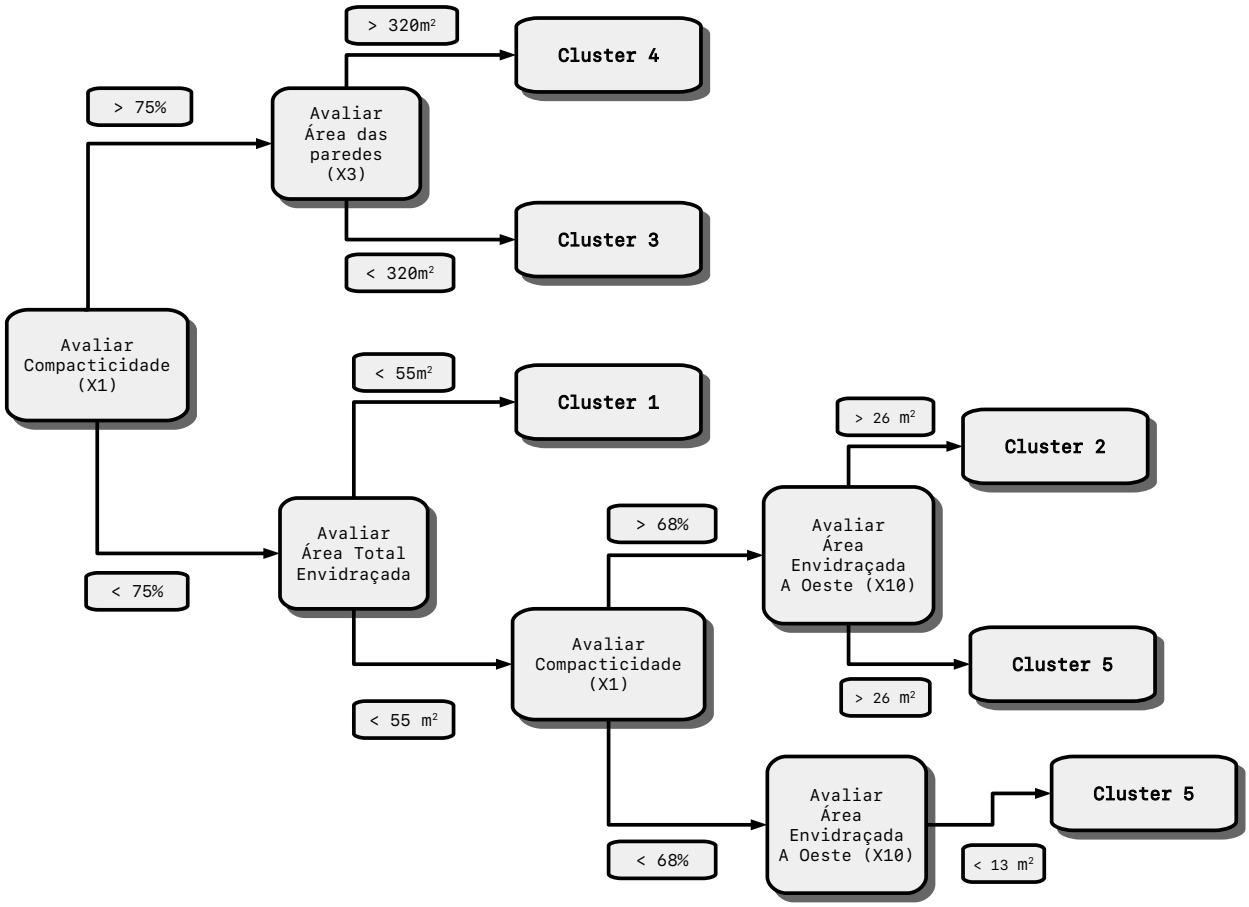


Figura 4.34: Árvore de decisão

Capítulo 5

Conclusões e Trabalho Futuro

Os objetivos estabelecidos pela proposta de trabalho dividem-se em três partes essenciais, com foco na análise de dados relativos a consumos de aquecimento e arrefecimento de edifícios residenciais.

Como primeira meta temos a diferenciação de estados e o agrupamento de edifícios com características similares e, desejavelmente com alguma correspondência em termo de consumos energéticos, apoiado no algoritmo imunológico implementado.

Com base neste agrupamento, o segundo objetivo passa pela caracterização das circunstâncias que condicionam o desempenho energético destes grupos de edifícios.

Finalmente o terceiro objetivo consiste na estimação numérica desse consumo energético.

Nas secções seguintes será apresentada uma análise crítica aos resultados obtidos ao longo de todas estas fases.

5.1 Diferenciação de estados

Tido como principal objetivo do trabalho, a implementação do algoritmo imunológico permitiu com sucesso a diferenciação de estados, caraterizados pelos seus centróides, que apresentaram boa diferenciação entre si, como se pode constatar por exemplo pela análise do gráfico das distâncias entre centróides.

A análise destes grupos permitiu identificar para cada um, características únicas, como evidenciado pela análise dos vários gráficos dos centróides, que permitem facilmente realizar atribuições de novos edifícios baseadas apenas na comparação das características. A cada um dos centróides corresponde uma classificação energética relativamente compacta e partindo deste resultado é possível estimar o grupo energético de um novo edifício em análise.

Apesar de estas classes apresentarem características distintas, estas não permitem contudo a identificação inequívoca do consumo ou classe de consumo energético, apresentando sobreposições notáveis entre si, como se pode constatar nos gráficos que comparam as atribuições por *cluster* com os diferentes consumos.

5.2 Caracterização dos estados

Baseados unicamente na análise dos *clusters* podemos retirar algumas conclusões acerca da influência das várias características estruturais nos consumos energéticos.

Em primeiro lugar, evidenciado pelos resultados das redes neuronais, temos que o consumo energético para arrefecimento é de mais difícil estimação, relativamente ao consumo energético para aquecimento.

No que toca à orientação do edifício vemos que pouco ou nenhuma influência nos consumos irá ter, pelo menos quando comparada com a orientação das áreas envidraçadas.

Ainda relativamente às áreas envidraçadas, ficou evidente ao longo do trabalho, que o levantamento destas informações não deve ser armazenado sob a forma de valores numéricos aleatórios, acompanhados de uma descrição detalhada para cada um. A apresentação dos valores em campos próprios, com a única descrição a ser as unidades em que estes se apresentam, é sem dúvida superior no que toca à obtenção de bons resultados através de cada um dos métodos testados no trabalho.

Relativamente aos valores da compacticidade temos que os valores mais elevados correspondem a edifícios pouco compactos, como se pode constatar pela relação inversa com a área da superfície, e que, para o caso de teste estudado, esta apresenta bastante preponderância na definição dos consumos energéticos.

Em relação à altura, os dados continham apenas dois valores distintos, mas revelou-se suficiente para perceber que esta apresenta também bastante importância no que toca aos consumos energéticos.

No que toca às áreas envidraçadas, individualmente apresentaram alguma dificuldade em introduzir diferenciação nos *clusters*. No entanto se basearmos a análise utilizando as variáveis em conjunto já apresentam uma melhor capacidade de diferenciar entre os vários *clusters*. Recomenda-se então que a análise das áreas envidraçadas não seja dissociada.

A introdução da variável área total envidraçada para além da pequena distinção que introduziu entre alguns *clusters* permitiu também perceber que o *cluster* 1 é algo atípico no que toca a esta variável, apresentando valores muito mais baixos inclusive alguns com ausência de área envidraçada.

Finalmente concluiu-se que das dez variáveis utilizadas, uma apresenta pouca influência nos resultados, duas são produto de uma forte correlação e quatro devem ser analisadas em conjunto.

5.3 Previsão do consumo energético

A estimação de consumos energéticos foi feita com recurso a redes neuronais, tendo-se obtido um desempenho de grande qualidade.

As redes neuronais implementadas permitiram, para além de prever os valores dos consumos, identificar as variáveis com mais peso nestes consumos, resultado que poderá ser utilizado na conceção de novos edifícios eficientes.

5.4 Dificuldades encontradas

As maiores dificuldades foram sobretudo encontradas numa fase inicial e compreendem sobretudo a apresentação e organização dos dados.

Numa primeira fase ficou clara a falta de informação acerca das unidades em que as mais diversas variáveis se apresentam. No documento que acompanha os dados apenas o volume, que é partilhado por todos os edifícios, apresenta unidades (metros cúbicos). Ao longo do documento é também feita uma referência passageira a outras variáveis que ajudaram a construir os modelos como a humidade (percentual), velocidade do ar (metros por segundo), nível de iluminação (lux), alcance do termostato (graus), entre outros. Contudo em relação às unidades relativas aos dados em concreto, o documento não faz qualquer menção.

Para além da questão das unidades temos a questão da falta de indicação do significado da variável da orientação. Mesmo após a conclusão do trabalho não estamos mais próximos de compreender a que se referem os valores.

5.5 Satisfação dos objetivos e trabalho futuro

Apesar das pequenas dificuldades encontradas ao longo do trabalho considera-se que os três objetivos principais foram atingidos e que se no final se termina com duas boas ferramentas de classificação e previsão do consumo energético de edifícios residenciais.

Apesar de se ter concluído com sucesso os objetivos fica sempre alguma coisa extra que podia ter sido realizada. Como indicação para trabalhos futuros ficam as seguintes sugestões:

- Testar o algoritmo com a introdução de dados alguns edifícios reais não simulados e comparar os resultados das atribuições com os *clusters* já existentes;
- Obter outra base de dados, que comporte dados reais, e repetir a metodologia utilizada, incluindo o *clustering* com o algoritmo imunológico partindo do zero no que toca à existência de detetores;
- Realizar a comparação entre os resultados dos dados reais e os atuais;
- Treinar novas redes neuronais com outros dados e analisar novamente as sensibilidades;
- Procurar a utilização de um método de armazenamento alternativo que consuma menos recursos em detrimento das folhas de cálculo. A maior fatia relativa ao tempo de execução do algoritmo pertence à leitura e escrita de dados e este ponto, com a introdução de outras bases de dados, será essencial para manter o tempo de execução aceitável, em caso de grandes quantidades de dados;
- Implementar a árvore de decisão em código semelhante e comparar os resultados com o algoritmo imunológico.

Referências

[1] Diana Ürge Vorsatz, Luisa F. Cabeza, Susana Serrano, Camila Barreneche, e Ksenia Petrichenko. *Heating and cooling energy trends and drivers in buildings*. Renewable and Sustainable Energy Reviews, 2015.

[2] Intergovernmental Panel on Climate Change. Fifth assessment report (ar5), 2014. URL: https://archive.ipcc.ch/pdf/assessment-report/ar5/syr/AR5_SYR_FINAL_SPM.pdf.

[3] Pablo Bermejo, Luis Redondo, Luis de la Ossa, Daniel Rodríguez, M Flores, Carmen Urea, José Gámez, e Jose Puerta. Design and simulation of a thermal comfort adaptive system based on fuzzy logic and on-line learning. *Energy and Buildings*, 49, 2012.

[4] Jin Woo Moon e Jong-Jin Kim. Ann-based thermal control models for residential buildings. *Building and Environment*, 45:1612–1625, 07 2010.

[5] Fernando Parra dos Anjos Lima. Análise de distúrbios de tensão em sistemas de distribuição de energia elétrica baseada em sistemas imunológicos artificiais. Tese de mestrado, Março 2013.

[6] Jagat Kishore Pattanaik, Mousumi Basu, e Deba Prasad Dash. Optimal power flow with facts devices using artificial immune systems. páginas 1–6, 12 2017.

[7] Gwo-Ching Liao. Short-term thermal generation scheduling using improved immune algorithm. *Electric Power Systems Research*, 2006.

[8] Tsong-Liang Huang, Ying-Tung Hsiao, C. H. Chang, e Joe-Air Jiang. Optimal placement of capacitors in distribution systems using an immune multi-objective algorithm. *International Journal of Electrical Power and Energy Systems*, 2008.

[9] Kay Chen Tan, Chi-Keong Goh, Abdullah Al-Mamun, e E Z. Ei. An evolutionary artificial immune system for multi-objective optimization. *European Journal of Operational Research*, 187, 2008.

[10] Farhath Zareen e Robert Karam. Detecting rtl trojans using artificial immune systems and high level behavior classification. 2018.

[11] Khaled Al-Sheshtawi, Hatem Abd elkader, e Nabil Ismail. Artificial immune clonal selection classification algorithms for classifying malware and benign processes using api call sequences. *International Journal of Computer Science and Network Security*, 2010.

[12] A Bagheri, Mostafa Zandieh, Iraj Mahdavi, e Mehdi Yazdani. An artificial immune algorithm for the flexible job-shop scheduling problem. *Future Generation Computer Systems*, 2010.

[13] Fernando Parra dos Anjos Lima, Fábio Roberto Chavarette, Simone Silva Frutuoso de Souza, Adriano dos Santos e Souza, e Mara Lúcia Martins Lopes. *Artificial Immune Systems Applied to the Analysis of Structural Integrity of a Building*. Trans Tech Publications, 2004.

[14] Jiawei Zhu, Fabrice Lauri, A Koukam, Vincent Hilaire, e Marcelo Simoes. Improving thermal comfort in residential buildings using artificial immune system. 2013.

[15] Mehryar Mohria, Afshin Rostamizadeh, e Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press, 2012.

[16] Informazione e Bioingegneria (DEIB) Dipartimento di Elettronica. A tutorial on clustering algorithms. Disponível em http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/.

[17] David Male, Jonathan Brostoff, David Roth, e Ivan Roitt. *Immunology*. Capítulo 6. Saunders, 7th ed. edição, 2012. URL: https://web.archive.org/web/20070216144000/http://pathmicro.med.sc.edu:80/ghaffar/innate.htm.

[18] Jason Brownlee. Clever algorithms: Nature-inspired programming recipes, 2011. URL: http://cleveralgorithms.com/nature-inspired/immune/negative_selection_algorithm.html.

[19] Y. Ishida. *Fully distributed diagnosis by PDP learning algorithm: towards immune network PDP models*. IJCNN International Joint Conference on Neural Networks, 1990.

[20] Leandro De Castro, Fernando José, e Antonio Augusto von Zuben. Artificial immune systems: Part i-basic theory and applications. 2000.

[21] Dipankar Dasgupta. *Artificial Immune Systems and Their Applications*. Springer-Verlag Berlin Heidelberg, 1 edição, 1999.

[22] M Ayara, Jon Timmis, R De Lemos, Leandro De Castro, e R Duncan. Negative selection: How to generate detectors. *Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS)*, 2002.

[23] Guoqiang Zhang, B. Eddy Patuwo, e Michael Y. Hu. *Forecasting with artificial neural networks: The state of the art*, volume 14. Internation Journal of Forecasting, 1998.

[24] Athanasios Tsanas e Angeliki Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. 2012. UCI Machine Learning Repository, disponível em https://archive.ics.uci.edu/ml/datasets/Energy+efficiency.

[25] Steven Vogel. *Life's Devices: The Physical World of Animals and Plants*. Capítulo 3. PUP, 1989.