Previsão de Vendas no Setor do Retalho: Uma Nova Abordagem Baseada em *Clustering*

José Carlos Rodrigues Nogueira

Dissertação de Mestrado

Orientador na FEUP: Prof. Américo de Azevedo
Orientador na Empresa: Prof. Patrícia Ramos

Mestrado Integrado em Engenharia e Gestão Industrial

2020-01-26

Resumo

Os avanços tecnológicos têm provocado melhorias significativas no mundo empresarial. Seja numa vertente operacional ou numa vertente estratégica, as novas tecnologias garantem vantagens competitivas indispensáveis às empresas que procuram dominar o setor em que se inserem. Em particular, no setor do retalho, as inovações tecnológicas tornaram possível o armazenamento de grandes quantidades de dados relativos a vendas e à sua posterior previsão. Quando realizada de uma forma eficiente, esta previsão pode ditar o sucesso de uma empresa.

Numa tentativa de acelerar o processo de previsão de vendas, assegurando a sua eficácia, surgiu este trabalho. Característico pelo recurso a técnicas desenvolvidas pela literatura de uma maneira inovadora, garante previsões robustas e rápidas sob resposta ao problema: criação de uma metodologia baseada em agrupamento de dados capaz de gerar previsões de vendas suportadas por características das séries temporais.

Procedeu-se à análise dos resultados das previsões de uma variedade de métodos diferentes de forma a calcular medidas de erro. Seguidamente foram extraídas as características de cada série temporal e realizado o agrupamento. Este procedimento foi realizado com o intuito de aferir que método de previsão se adequa melhor a cada tipo de série temporal e, desta forma, garantir o cumprimento do objetivo proposto.

Na implementação do código, bem como na avaliação dos resultados, foi usada a ferramenta RStudio que garante um ambiente de desenvolvimento amigável e uma visualização eficiente dos resultados. Resultados estes que revelam um desempenho superior, previsões mais robustas e menor poder computacional do que exigido à aplicação de todos os modelos mais comuns de previsão. Incrementalmente, o algoritmo desenvolvido permite a previsão de vários tipos de séries temporais, dado que transmite a informação que lhe foi previamente fornecida.

Abstract

Technologic advances have been improving entrepreneurship world significatively. On an operational or strategic perspective, new technologies ensure fundamental competitive advantages for those who seek to overpower its own business area. Specifically, in retail business, technologic innovation allowed storing large-scale sales information and its posterior forecast. When efficiently done, these forecasts can ensure company success.

Trying to speed up forecasting process, while ensuring its effectiveness, this work arose. Based on previously developed techniques, used in an innovative manner, solves the problem (Creation of a clustering-based forecasting methodology, able to produce time series feature supported sales forecasts) with fast and robust forecasts.

Different methods' results were analyzed in order to compute error' measures. Then features were extracted from each time series and the clusters were made. This procedure was carried out in order to find which forecasting method is suited to each time series and, thereby, warrant for proposed objective fulfilment.

RStudio was used to code deployment and result assessment, providing efficient result' visualization as well as a friendly deployment environment. The results show superior performance, more accurate results and less computational power required than the deployment of known common forecasting methods. Moreover, the proposed algorithm allows different time series types' forecast since the method suggestions are based on previously conveyed information.

Agradecimentos

Em primeiro lugar quero agradecer aos meus orientadores. A Professora Patrícia Ramos, que me transmitiu uma quantidade exorbitante de conhecimento durante o desenvolvimento deste projeto, a realização desta dissertação não seria possível sem a sua ajuda. E o Professor Américo de Azevedo, que contribuiu para a minha formação enquanto professor e se mostrou sempre disponível para me ajudar, principalmente quando me abriu as portas a esta instituição que contribuiu em tanto para a minha formação. Agradeço imenso a ambos.

Deixo também o meu agradecimento ao INESC TEC. A instituição que me acolheu e facultou todos os meios necessários durante a realização deste projeto.

Quero agradecer aos meus pais por todos os ensinamentos e pelo apoio que me transmitiram. Um obrigado por me terem moldado da melhor forma que conseguiram para enfrentar não só as adversidades que este percurso me trouxe, mas também as que estão por vir.

Por fim quero agradecer aos amigos que me acompanharam ao longo deste percurso. Deixo um especial obrigado a quem, de uma forma presente, ausente, próxima ou distante, me apoiou, me motivou e me aconselhou nos momentos fundamentais da minha vida.

José Carlos Rodrigues Nogueira

"Life can only be understood backwards, but it must be lived forwards."

Søren Kierkegaard

Conteúdo

Acronyms and Symbols

AIC	Akaike Information Criteria
ARIMA	Autoregressive Integrated Moving Average
BIC	Bayesian Information Criteria
ETS	ExponenTial Smoothing
MAE	Mean Absolute Error
MASE	Mean Absolute Scaled Error
RelMAE	Relative Mean Absolute Error
AvgRelMAE	Average Relative Mean Absolute Error
RSS	Residual Sum of Squares
SKU	Stock-Keeping Unit
TBATS	Trigonometric Box-Cox transform, ARMA errors, Trend, and Seasonal components

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

A previsão de vendas é atualmente uma das questões fundamentais para as decisões estratégicas e de planeamento de operações eficazes e eficientes das cadeias de abastecimento. Para empresas de retalho lucrativas, uma previsão exata de vendas é crucial na organização e no planeamento da compra, produção, transporte e mão-de-obra.

Especificamente no setor do retalho, a competitividade do mesmo fomenta a procura de metodologias cada vez mais eficazes e rápidas de previsão. O aumento crescente do consumo e a globalização obrigam a que as empresas se desenvolvam no sentido da eficiência das operações. Para tal, é importante minimizar os custos logísticos enquanto se garante um serviço ao cliente propício à sua retenção.

Os avanços tecnológicos tornaram possível o armazenamento de grandes quantidades de informação, o que tornou o processo de previsão exigente mesmo para as tecnologias mais desenvolvidas. As soluções mais básicas para prever uma quantidade elevada de séries temporais são a escolha de um modelo que se ajuste a todas as séries ou a escolha de um modelo para cada série. Os estudos disponíveis revelam que não há nenhum modelo que apresente um bom desempenho em todo o tipo de séries temporais (Timmermann, 2006). Por sua vez, a escolha do melhor modelo a ajustar a cada série temporal é uma opção cada vez menos válida devido ao tempo e ao poder computacional exigidos para seguir este procedimento e à impossibilidade de usar os critérios de seleção de modelos na comparação entre as classes de métodos de previsão (Hyndman and Koehler, 2006).

Estudos mais recentes acerca do tema incidem sobre várias áreas em crescente desenvolvimento. A literatura, relacionada com o objetivo desta dissertação, revela bastante interesse no desenvolvimento e melhoria de técnicas de agrupamento de dados (Aghabozorgi et al., 2015) para que os mesmos traduzam a sua utilidade, quando aplicados a séries temporais. Na vertente supervisionada, os estudos incidem na avaliação da eficácia dos métodos (Talagala et al., 2018; Montero-Manso et al., 2020).

Relativamente às metodologias supervisionadas referidas anteriormente, as soluções incidem, consideravelmente, no seu uso direcionado para a determinação do melhor método de previsão para aplicar a uma série temporal. Vários estudos na literatura apontam na direção de invalidar

a sua adequabilidade à solução do problema aliado a esta dissertação. Dos vários procedimentos revistos durante a conceptualização da *framework*, encontraram-se dois focos principais. Ou se procura atingir resultados bastante eficazes, exigindo um tempo de processamento elevado. Ou se procura reduzir o tempo de processamento, sacrificando a eficácia das previsões. Ambos se cruzam com dificuldades centradas na densidade de SKUs (*Stock Keeping Units*) ao longo do espaço das características, da grande dimensionalidade dos dados e da presença de uma forte componente associada ao ruído (Wang et al., 2006), o que se traduz em dificuldades na distinção entre séries temporais quando representadas pelas suas características (Vlachos et al., 2003).

Quanto ao agrupamento de dados, os estudos revelam uma aplicabilidade promissora relativamente ao problema em questão (Aghabozorgi et al., 2015). Por um lado permite uma flexibilidade acrescida. A realização dos grupos pode ser realizada com um conjunto de variáveis e a comparação das distâncias de cada série temporal, a cada grupo, pode ser realizada com um subconjunto, desde que pertencente ao conjunto de variáveis inicial. Vantagens associadas a este facto residem na redução do tempo de processamento requerido numa fase de aproximação, aliada a um incremento de informação usada para a realização dos grupos. Após o agrupamento dos grupos, a aplicação de técnicas de visualização permite extrair informações valiosas a uma gestão eficiente (Wijk, van and Selow, van, 1999). Uma análise breve dos resultados permite perceber de que forma é que os SKUs se agrupam entre si e, consequentemente, obter conhecimento prévio do comportamento de um produto, mesmo antes da sua introdução no mercado. Uma outra vantagem associada ao uso de agrupamento de dados em conjunto com métodos de previsão ascende quando uma relação entre característica-modelo possa ser do interesse da empresa.

Desta forma, este trabalho é motivado pela necessidade de desenvolver uma *framework* de previsão de séries temporais que, para além de assegurar resultados satisfatoriamente exatos, garanta um tempo e poder de processamento reduzidos. Mais ainda, que o desempenho da *framework* torne favorável a sua implementação nos processos de gestão do Grupo Jerónimo Martins. Para a sua realização, numa parceria com o INESC TEC, o grupo disponibilizou um conjunto de dados significativo que suporta um estudo e posterior avaliação da *framework* de previsão ao nível do SKU.

1.1 Objetivos

O principal objetivo desta dissertação centra-se no desenvolvimento de uma *framework* de previsão de vendas baseada em agrupamento de dados, aplicável ao setor do retalho.

Como foi referido anteriormente, abordagens supervisionadas apresentam dificuldades na distinção entre séries temporais, enquanto o agrupamento de dados consegue lidar mais facilmente com uma distribuição menos uniforme das características no espaço. Para além disso, a ideia de que o agrupamento das séries temporais consegue traduzir-se em vantagens adicionais relacionadas com a extensão da análise, não só sugere o seu uso, como também representa uma oportunidade de explorar uma abordagem inovadora. A grande questão reside na definição dos fundamentos da sugestão dos métodos, por parte dos grupos.

O agrupamento de dados tem sido usado em variadas áreas para funções distintas, porém, é previsível que o seu uso neste contexto também garanta bons resultados.

A *framework* recorrerá a técnicas de agrupamento de dados e a métodos de previsão de séries temporais para determinar o modelo de previsão a ajustar a cada série. O processo de agrupamento usará os erros obtidos no conjunto de modelos e as características das séries temporais para elaborar grupos caracterizados pelo bom desempenho de um determinado modelo. Prevê-se que o agrupamento dos dados seja um procedimento bastante rápido, porém, da mesma forma que a determinação do melhor modelo para cada série, estará dependente de um processo moroso de ajustamento de um elevado número de modelos a cada série temporal. Toda esta fase prévia é idealizada de uma forma *offline* procurando não atrasar a aplicação da *framework*.

Após a realização dos grupos, já na fase *online*, uma série temporal será introduzida como *input*, calculadas as suas características e as distâncias entre a série temporal e o elemento representativo de cada grupo. Finalmente, o modelo com melhor desempenho no grupo mais próximo é considerado como o melhor modelo a ajustar à nova série temporal.

Toda a implementação será auxiliada por funções e pacotes disponíveis no *software* R.

Este trabalho segue uma abordagem inovadora e bastante complexa, pelo que constitui um enorme desafio. A sua avaliação será realizada de uma forma comparativa. Os resultados obtidos através da *framework* serão comparados com uma aplicação generalizada de métodos na vanguarda do estado da arte, sendo o sucesso caracterizado por um desempenho superior. Cumpre-se o objetivo caso a *framework* desenvolvida resulte em previsões robustas e o mais exatas possível, determinadas num intervalo de tempo que permita a sua implementação nos processos de gestão das lojas Pingo Doce do Grupo Jerónimo Martins.

1.2 Estrutura da Dissertação

A escrita desta dissertação encontra-se dividida em 4 capítulos.

No primeiro capítulo é introduzida a ideia fundamental do trabalho realizado, acompanhada de uma contextualização que serve de auxilio à justificação da escolha da abordagem adotada para desenvolver a *framework*.

No capítulo 2 são apresentados os temas sobre os quais a *framework* foi desenvolvida. São descritos os métodos de previsão de séries temporais e de agrupamento de dados numa perspectiva que facilite o seu entendimento e é fornecida a informação fundamental à compreenção do modo de funcionamento da *framework*.

O capítulo 3, constituído pelo caso de estudo, contém informação relativa à *framework* desenvolvida, ao procedimento experimental para a sua validação e à análise e apresentação de resultados. Foi utilizado o *software* R tanto para o desenvolvimento e implementação da *framework* aos dados disponíveis, como para a avaliação dos resultados.

Finalmente, no capítulo 4, estão presentes as principais conclusões obtidas com a realização deste trabalho e sugestões de trabalho futuro.

Capítulo 2

Estado da Arte

2.1 Séries Temporais

De um modo geral, define-se por série temporal qualquer resposta ou sinal aleatórios que possam ser medidos de uma forma contínua ou sequencial ao longo do tempo (Grossmann and Rinderle-Ma, 2015).

Por sua vez, a análise de uma série temporal permite a extração de conhecimento útil à resolução de problemas nos demasiados sistemas da atualidade.

Um aspeto a considerar quando se pretende realizar este tipo de análises é a profundidade das mesmas. Enquanto uma simples representação gráfica pode facultar explicações de eventos passados, uma analise mais complexa permite retirar ilações sobre o seu comportamento futuro.

Os avanços no estudo deste tema concluíram que cada série temporal pode ser caracterizada pelos seus componentes intrínsecos, ou seja, pode ser vista como um objeto único (Kumar and Nagabhushan, 2006). Inserem-se neste estudo padrões e características, nomeadamente tendência, sazonalidade, ciclo, frequência e autocorrelações.

No caso particular de dados de vendas, e mais ainda no setor do retalho, as séries temporais apresentam muitas vezes fortes componentes sazonais e de tendência, acompanhadas por uma componente irregular característica. Neste setor é ainda notória a predominância de séries temporais sequenciais medidas em intervalos de tempo regulares (Murlidhar et al., 2012).

2.2 Agrupamento de Dados

O agrupamento de dados (*Clustering*) faz parte de um conjunto de metodologias não supervisionadas focadas na extração de conhecimento de um conjunto de dados. O seu princípio de funcionamento baseia-se na aproximação de amostras similares entre si sem que haja conhecimento prévio das características de cada grupo (Rai and Shubha, 2010).

Este tipo de análise pode ser usada individualmente ou sob a forma de base à implementação de algoritmos mais complexos. Estudos na literatura mostram que o uso de agrupamento de dados

como suporte resulta em vantagens significativas, não só ao nível do desempenho como também na possibilidade de implementação (Paparrizos and Gravano, 2015).

Embora exista uma procura extensiva pelo melhor método, é custoso encontrar uma metodologia adequada a todos os casos. A escolha do número de grupos, do método e da medida de similaridade depende sempre de cada problema e dos objetivos da abordagem. Adicionalmente, devido à complexidade característica dos dados no setor do retalho, esta abordagem não supervisionada revela-se bastante robusta (Vlachos et al., 2003).

Com o avanço tecnológico, tornou-se possível o armazenamento de grandes quantidades de séries temporais, o que despoletou um interesse crescente no seu estudo e proporcionou uma alteração dos métodos de agrupamento para atender às necessidades deste novo tipo de dados.

Note-se que os avanços não incidiram apenas nos algoritmos usados para realizar o agrupamento mas também nas abordagens de agrupamento adotadas. Como foi dito anteriormente, uma série temporal pode ser transformada num objeto através da extração de características. É notória a presença de estudos na literatura que abordam este tema que se revela vantajoso na abstração a ruídos e na redução de dimensão (Aghabozorgi et al., 2015).

2.2.1 Medidas de Similaridade

Os algoritmos de agrupamento de dados necessitam de se basear na similaridade entre amostras, não só para escolher que grupos ligar ou que amostras atribuir a cada grupo, mas também para posteriormente averiguar quão bom foi o agrupamento. Há vários métodos para a realização deste cálculo mas todos resultam numa matriz de distâncias entre as amostras.

A escolha do método a usar é um passo crucial na implementação de cada algoritmo e a matriz de similaridades tem influência tanto na forma como na constituição de cada grupo. Dois dos métodos mais eficazes no agrupamento de séries temporais são a distância euclidiana e o *Dynamic Time Warping*.

Distância euclidiana: A distância euclidiana é uma das medidas de distância mais conhecidas e uma escolha comum no agrupamento de séries temporais:

$$d_{euc}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2.1}$$

onde x e y representam duas amostras distintas.

Problemas aliados ao seu uso aparecem quando as séries temporais não estão alinhadas no tempo ou quando existe uma forte presença de ruído. Cálculos prévios ajudam a mitigar ambos os problemas, permitindo uma aplicação da distancia euclidiana sobre os seus resultados (Lkhagva et al., 2006).

***Dynamic Time Warping*:** O *Dynamic Time Warping* oferece uma abordagem diferente ao cálculo da distância entre duas séries temporais. A sua aplicação permite um mapeamento não linear que resulta num cálculo baseado num alinhamento perfeito entre as séries (Salvador and Chan, 2004).

Sejam X e Y duas séries temporais, de tamanho $|X|$ e $|Y|$,

$$X = x_1, x_2, ..., x_{|X|} \tag{2.2}$$

$$Y = y_1, y_2, ..., y_{|Y|}. \tag{2.3}$$

Constrói-se um caminho W

$$W = w_1, w_2, ..., w_k, \quad max(|X|, |Y|) \leq K \leq |X| + |Y| \tag{2.4}$$

onde K é o tamanho do caminho e o elemento k é:

$$w_k = (i, j) \tag{2.5}$$

sendo i o índice da série X e j o índice da série Y. O caminho deve começar no início de cada série temporal $w_1 = (1,1)$ e acabar no fim de de ambas as séries temporais $w_k = (|X|, |Y|)$ de forma a garantir que todos os índices das séries são usados no caminho. Note-se que:

$$w_k = (i, j), \ w_{k+1} = (i', j'), \quad i \leq i' \leq i+1, \quad j \leq j' \leq j+1. \tag{2.6}$$

O caminho ótimo é aquele que minimiza

$$Dist(W) = \sum_{k=1}^{k=K} Dist(w_{k,i}, w_{k,j}) \tag{2.7}$$

onde $Dist(W)$ é a distância do caminho, normalmente a distância euclidiana, e $Dist(w_{k,i}, w_{k,j})$ é a distância entre dois pontos i e j no elemento k do caminho.

2.2.2 *K-means*

Quando se pretende dividir o conjunto de dados em k partições, o método *K-means* é a opção mais simples e mais usada (Jain, 2010). Vários estudos relativos ao tema mostram que para além da rapidez, é notória uma eficácia aliada à aplicação do método (Dai et al., 2015).

Num processo iterativo, este algoritmo atribui cada amostra a um grupo com base na sua similaridade ao respetivo centroide. As características do centroide, calculadas a cada iteração, podem ser definidas como a média das características dos elementos pertencentes a cada grupo.

A ideia fundamental do *k-means* consiste na definição dos grupos de uma forma que minimize a variação intra-grupo. Há vários avanços na literatura relativos ao tema (Jain, 2010). O algoritmo padrão (MacQueen, 1967) define a variação total intra-grupo como a soma do quadrado das distâncias euclidianas entre as amostras e o centroide correspondente:

$$W(C_k) = \sum_{x_i \in C_k} (x_i - \mu_k)^2 \tag{2.8}$$

onde x_i é uma amostra pertencente ao grupo C_k e μ_k é a média das amostras atribuídas ao grupo C_k.

Cada amostra é atribuída ao respetivo grupo de forma a minimizar o soma dos quadrados das distâncias referidas anteriormente. A variação total intra-grupo define-se por

$$VarTot = \sum_{k=1}^{k} \sum_{x_i \in C_k} (x_i - \mu_k)^2 \tag{2.9}$$

e indica quão compacto é o agrupamento.

Embora bastante eficiente e rápido, o *K-means* apresenta desvantagens nomeadamente na necessidade de definir o número de grupos k de uma forma prévia, na sensibilidade a *outliers* e na influência que a ordem dos dados pode causar nos resultados.

Determinação do Número Ótimo de Grupos

Para além dos dados a agrupar, o *K-means* tem o número de grupos como dado de entrada, como foi referido anteriormente. A solução final e a sua qualidade dependem da escolha realizada neste passo que pode ser levada a cabo de uma forma automática ou baseada na experiência do analista. Com os avanços relativos ao tema, foram desenvolvidas metodologias para facilitar a determinação de k de uma forma fundamentada em cálculos estatísticos. Os métodos mais populares são o método *Elbow*, o método *Silhouette* (Kaufman and Rousseeuw, 1990) e o método *Gap statistic* (Tibshirani et al., 2001).

Método *Elbow*: Dado que a ideia por detrás do *Clustering* é agrupar os dados de forma a minimizar a variação total intra-grupo, uma abordagem possível é: realizar o agrupamento para $k = 1, 2, ..., n$, calcular a variação total intra-grupo para cada valor de k, elaborar um gráfico com o valor da variação em função de k e, finalmente, escolher o valor de k com base na localização do "cotovelo". O "cotovelo" (*elbow*) diz respeito ao ponto em que o incremento de k não revela uma diminuição acentuada no valor da variação total intra-grupo e é um indicador do número apropriado de grupos a usar.

A escolha de k por este método é subjetiva, o que representa uma grande desvantagem relativamente aos métodos seguintes.

Método *Average Silhouette*: Este abordagem avalia a qualidade do agrupamento, ou seja, determina quão adequada foi a atribuição das amostras ao grupo correspondente. O número ótimo de grupos é dado pelo valor de k que maximiza o valor da *Average Silhouette*. Desta forma, quanto maior for o seu valor, melhor é o agrupamento.

O procedimento é similar ao método referido anteriormente. Consiste em: determinar os grupos para diferentes valores de k, calcular a *average silhouette* das amostras para cada agrupamento, elaborar o gráfico da *average silhouette* em função do valor de k e finalmente escolher o valor de k que corresponde ao valor máximo.

Este método baseia-se no cálculo da "largura" dos grupos, estimando assim a distância média entre os mesmos. O seu valor representa a proximidade entre um ponto pertencente a um grupo e os pontos dos grupos vizinhos.

Para cada amostra i, a *silhouette width* s_i é calculada da forma seguinte (Kassambara, 2017):

1. Para cada observação i, calcular a dissimilaridade média, a_i entre i e todos os pontos do grupo a que i pertence;

2. Para todos os grupos C, aos quais i não pertence, calcular a dissimilaridade $d(i,C)$ entre i e todas as observações de C. O menor valor de $d(i,C)$ é definido como $b_i = \min_C d(i,C)$. O valor de b_i pode ser visto como a dissimilaridade ente i e o seu grupo vizinho, ou seja, o grupo mais próximo ao qual i não pertence.

3. Finalmente, $S_i = (b_i - a_i)/\max(a_i, b_i)$.

Método *Gap statistic*: O método *Gap statistic* funciona como uma comparação entre o conjunto de dados a agrupar e um conjunto de dados de referência, criado de forma a que a distribuição associada não é propícia a agrupamento, ou seja, não tem grupos inerentes.

O conjunto de dados de referência é gerado usando simulações de Monte Carlo do processo de amostragem, ou seja, para cada variável x_i pertencente ao conjunto de dados a agrupar, são determinados o seu máximo $\max(x_i)$ e o seu mínimo $\max(x_j)$ e são gerados n pontos distribuídos uniformemente ao longo do intervalo.

Para ambos os conjuntos de dados, calcula-se a variação total intra-grupo para diferentes valor de k:

$$Gap_n(k) = E_n^* \log(W_k) - \log(W_k) \tag{2.10}$$

onde $E_n^* \log(W_k)$ é definida através de *bootstrapping* (B), gerando B cópias do conjunto de dados de referência e calculando a média, $\log(W_k^*)$. A *Gap statistic* mede o desvio entre o valor de W_k observado e o valor esperado, representado pela hipótese nula. Desta forma, k será o valor que maximiza $Gap_n(k)$, o que indica que a estrutura do agrupamento está distante de uma distribuição uniforme das amostras no espaço do agrupamento.

O cálculo segue os seguintes passos:

1. Agrupar os dados para diferentes valores de k e determinar o w_k correspondente;

2. Gerar B conjuntos de dados de referência e agrupar os mesmos variando o valor de k. Calcular o valor da *Gap statistic* apresentada na equação 2.10.

3. Sendo $\bar{w} = (1/B)\sum_b \log(W_{kb}^*)$, calcular o desvio padrão $sd(k) = \sqrt{(1/b)\sum_b \log(W_{kb}^* - \bar{w})^2}$ e definir $s_k = sd_k \times \sqrt{1+1/B}$.

4. Escolher o número de grupos como sendo o menor k tal que: $Gap(k) \geq Gap(k+1) - s_{k+1}$.

Algoritmo

Para proceder à implementação do algoritmo *k-means*, geralmente é necessário algum pré-processamento de forma a organizar o conjunto de dados e tornar possível o agrupamento. Desta forma, as linhas devem conter as observações e as colunas devem dizer respeito às variáveis. Não devem existir valores não definidos e os dados devem estar padronizados de forma a permitir uma comparação justa entre variáveis.

O algoritmo em discussão inclui os seguintes passos:

1. Indicar o número de grupos, k;
2. Escolher k amostras do conjunto de dados como sendo os centroides iniciais;
3. Atribuir cada amostra ao centroide mais próximo com base nas dissimilaridades entre as amostras e os centroides;
4. Para cada grupo k, atualizar o centroide do grupo calculando a média de todas as amostras pertencentes ao grupo.
5. De uma forma iterativa, repetindo os passos 3 e 4 até que os centroides do grupo parem de alterar, minimizar $VarTot$.

2.2.3 *K-medoids*

O *K-medoids* é uma boa alternativa ao método referido anteriormente. Neste caso não temos centroides mas sim medoides. Os medoides, por sua vez, são os elementos que apresentam uma maior similaridade média com os elementos do grupo em que se inserem. Esta alteração relativamente ao *K-means* altera significativamente a atribuição das amostras aos grupos.

Um dos algoritmos mais comuns associados a esta alternativa é o *PAM - Partitioning Around Medoids*, desenvolvido por Kaufman and Rousseeuw (2008).

Uma vez que a representação do grupo é feita por um elemento integrante do mesmo, o *k-medoids* apresenta vantagens na capacidade de lidar com *outliers*.

Depois da definição de k medoides, que pode ser feita de uma forma automática pelo algoritmo ou de uma forma manual sob a forma de dado de entrada, os grupos são construídos atribuindo cada amostra ao mediode mais próximo. Seguidamente, cada medoide, m, e cada amostra são trocados e a função custo é calculada. Esta corresponde à soma das dissimilaridades entre todas as amostras e o seu medoide mais próximo.

O passo seguinte envolve uma troca entre as amostras e o respetivo medoide sob a tentativa de melhorar o agrupamento. Se a função custo diminuir com a troca, então a amostra torna-se o novo medoide. Este passo repete-se até que a função custo deixe de revelar alterações descendentes. De uma forma geral, o objetivo é encontrar k elementos representativos que minimizem o soma das dissimilaridades entre as observações e o seu elemento representativo mais próximo.

Note-se que k pode ser determinado pelos mesmos métodos referidos no método apresentado anteriormente.

Algoritmo

O algoritmo em discussão inclui os seguintes passos:

1. Indicar o número de grupos, k, ou fornecer diretamente os medoides iniciais;
2. No caso de ter sido fornecido o número de grupos, k, escolher aleatoriamente k amostras como medoides;
3. Calcular a matriz de similaridades, caso esta não tenha sido fornecida;
4. Para cada grupo averiguar se algum dos seus elementos diminui a dissimilaridade, no caso positivo, o elemento que minimizar a dissimilaridade intra-grupo deve ser escolhido como medoide.
5. Se algum medoide alterar voltar ao passo 3, caso contrário terminar o algoritmo.

2.2.4 Agrupamento Hierárquico

Quando as amostras não são totalmente independentes e faz sentido que existam sub-grupos, o agrupamento hierárquico de dados pode ser implementado para obter bons resultados. Este método é bastante usado e apresenta vantagens notórias na visualização dos resultados devido à apresentação dos mesmos sob a forma de "árvore" ou dendograma (Vlachos et al., 2003).

Para além das vantagens centradas na visualização, o agrupamento hierárquico de dados não requer uma escolha prévia do número de grupos. Embora não seja adequado para conjuntos de dados grandes por causa da exigência de poder computacional elevada, esta metodologia contorna uma das maiores dificuldades da área (Murlidhar et al., 2012).

Existem duas formas de agrupar os dados hierarquicamente: de uma forma aglomerativa e de uma forma divisiva. A primeira, também conhecida por *Agglomerative nesting*, começa por considerar todas as amostras como um grupo e, a cada passo, agrupa os grupos mais similares num novo grupo. O procedimento termina quando não há mais grupos para agrupar, ou seja, quando o grupo final é todo o conjunto de dados. Por sua vez, quando os grupos são formados de uma forma divisiva, o procedimento é o oposto do anterior. Também conhecido por *Divisive analysis*, o algoritmo começa por considerar todo o conjunto de dados como um grupo e, a cada passo, o grupo mais dissimilar é separado em dois grupos. O algoritmo termina quando o número de amostras é igual ao número de grupos. Note-se que, de um modo geral, o procedimento divisivo é mais eficaz a encontrar grupos pequenos, enquanto que o agrupamento aglomerativo é mais eficaz a encontrar grupos grandes (Kassambara, 2017).

Para além da dissimilaridade entre amostras, o agrupamento hierárquico necessita de calcular a dissimilaridade entre os grupos. Para tal existem diferentes métodos sendo os mais comuns: o *Maximum or complete linkage clustering*, *Minimum or single linkage clustering*, *Mean or average linkage clustering*, *Centroid linkage clustering* e o *Ward's minimum variance method*.

***Maximum or complete linkage clustering*:** Assume a maior distância entre amostras de grupos diferentes como a distância entre esses mesmos grupos. O seu uso favorece a formação de grupos mais compactos.

***Minimum or single linkage clustering*:** Assume a menor distância entre amostras de grupos diferentes como a distância entre esses mesmos grupos. O seu uso favorece a formação de grupos mais longos.

***Mean or average linkage clustering*:** Assume a média das distâncias entre amostras de grupos diferentes como a distância entre esses mesmos grupos.

***Centroid linkage clustering*:** Representa a distância entre os centroides de dois grupos.

***Ward's minimum variance method*:** Representa a variância total intra-grupo.

Algoritmo

O agrupamento hierárquico inclui os seguintes passos:

1. Calcular a matriz das dissimilaridades entre as amostras;
2. Usar o método de ligação para juntar/separar dois grupos;
3. Determinar onde cortar a árvore em grupos.

2.3 Métodos de Previsão

A previsão de eventos é uma disciplina que se revela importante na otimização de processos de gestão e planeamento nas áreas em que tem sido usada (Talagala et al., 2018). Devido à sua flexibilidade temporal, isto é, a possibilidade de ser realizada uma previsão com um intervalo de tempo a definir, permite que esta disciplina seja adequada a uma variedade enorme de áreas. Em específico no setor do retalho, a previsão de vendas revela um impacto significativo na redução de custos e na manutenção da relação com o cliente, indicando um intervalo de valores previstos de vendas que permita à gestão tomar decisões que suportem ambos.

Certos eventos podem ser previstos mais facilmente do que outros. De uma forma geral, a predictibilidade de um evento depende muito do entendimento que o analista tem dos fatores que influenciam esse mesmo evento, da quantidade de informação disponível e da influência que as previsões revelam no comportamento do evento (Hyndman and Athanasopoulos, 2014). No setor do retalho, nomeadamente, as promoções, o clima e até a altura do mês têm um impacto significativo nas vendas. Uma boa previsão deve ter em consideração fatores como os indicados e carece de uma análise crítica dos resultados por parte do analista. Não só as previsões podem resultar em

valores que não fazem sentido, como também decisões baseadas nessas mesmas previsões podem influenciar o comportamento das vendas.

Num ambiente tão volátil como é o setor do retalho, os analistas devem assegurar que mais que um modelo explicativo, o seu trabalho resulta num modelo robusto o suficiente para prever as mudanças do ambiente. Para tal, e mais uma vez limitados pela quantidade e qualidade dos dados disponíveis, é imperativo testar uma vasta quantidade de opções, uma vez que nenhum modelo revela sempre os melhores resultados como Talagala et al. (2018) conseguiu comprovar.

Tem sido desenvolvido algum trabalho relacionado com este tema e importa referir os desenvolvimentos de Bates and Granger (1969), Clemen (1989) e Timmermann (2006) incidentes na combinação de previsões que, através de comparação de resultados, foram capazes de mostrar que uma previsão combinada resulta em melhores resultados de previsão. Uma grande dificuldade assenta na escolha de pesos a atribuir às previsões de cada modelo, dificuldade esta que Montero-Manso et al. (2020) tentaram mitigar através de uma abordagem mais complexa com recurso a aprendizagem de máquina. Apesar dos bons resultados atingidos por desenvolvimentos mais complexos existentes na literatura, este tema ainda carece de uma metodologia rápida o suficiente para resultar em previsões satisfatórias em tempo útil (Talagala et al., 2018).

Note-se que de agora em diante, $y_{t|T}$ refere-se à variável aleatória y_t em função das T observações conhecidas. De uma forma análoga, $\hat{y}_{T+h|T}$ diz respeito à previsão de y_{T+h} considerando as observações $y_1, y_2, y_3, ..., y_T$, onde h é o número de períodos ou passos à frente que se pretende prever.

2.3.1 Métodos Básicos de Previsão

A previsão de séries temporais é um tema amplamente abordado na literatura (De Gooijer and Hyndman, 2005). Com o avanço da investigação relativa ao tema foram aparecendo métodos novos e mais complexos, no entanto nenhum garante uma eficácia geral (Talagala et al., 2018).

Na sequência da necessidade de comparação de métodos recorrendo a uma base de comparação, são habitualmente usados métodos de previsão mais simples que, embora pouco sofisticados, têm surpreendido na medida em que revelam uma capacidade de previsão elevada.

Método Naïve

No método naïve, todas as previsões futuras tomam o valor da última observação, ou seja:

$$\hat{y}_{T+h|T} = y_T. \tag{2.11}$$

Quando as séries temporais seguem um passeio aleatório, as previsões resultantes deste método apresentam geralmente bons resultados.

Método Naïve sazonal

O método abordado anteriormente não apresenta bons resultados quando aplicado a séries temporais sazonais. Por outro lado, quando se aplica o método Naïve sazonal a este tipo de séries temporais os resultados podem ser satisfatórios. As previsões deste método são obtidas da forma seguinte:

$$\hat{y}_{T+h|T} = y_{T+h-m(k+1)}. \tag{2.12}$$

Como se pode aferir desta equação, onde m representa o período de sazonalidade e k a parte inteira de $(h-1)/m$, cada previsão é igual ao último valor observado do respetivo período homólogo.

2.3.2 Modelos de Espaço de Estados

O alisamento exponencial, usado para prever o comportamento futuro de séries temporais, nasceu por volta de 1950 (Billah et al., 2006) e tem vindo a motivar alguns dos métodos mais usados até aos dias de hoje. Os seus métodos produzem previsões fiáveis de uma forma rápida e são aplicáveis a um vasto conjunto de séries temporais univariadas.

Estes métodos evoluíram no sentido de reconhecer a tendência e a sazonalidade das séries temporais e na forma como estes componentes são incluídos no método de alisamento, tornando assim possível a sua aplicação a séries temporais com várias características distintas.

Como o nome sugere, as previsões geradas por estes métodos são médias ponderadas das observações de períodos anteriores cujos pesos decrescem exponencialmente com antiguidade das observações.

A representação da forma de componente destes métodos contém uma equação de previsão e uma equação de alisamento para cada um dos componentes considerados: o nível, a tendência e a sazonalidade. As possibilidades são:

- Tendência $= \{N(None), A(Additive), A_d(Additive\ damped)\}$;
- Sazonalidade $= \{N(None), A(Additive), M(Multiplicative)\}$.

Cada método é especificado usando um par de letras, (T,S) sendo que T diz respeito à tendência e S à sazonalidade. Considerando todos as combinações possíveis de tendência e sazonalidade, é possível construir nove métodos de alisamento exponencial.

Por exemplo, na forma de componente o método aditivo de Holt-Winters, (A,A) é

$$\hat{y}_{t+h|t} = l_t + hb_t + s_{t+h-m(k+1)} \tag{2.13}$$

$$l_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(l_{t-1} + b_{t-1}) \tag{2.14}$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1-\beta^*)b_{t-1} \tag{2.15}$$

$$s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m} \tag{2.16}$$

$$0 \leq \alpha \leq 1, \quad 0 \leq \beta^* \leq 1, \quad 0 \leq \gamma \leq 1$$

onde y_t representa a série temporal para $t = 1,2,...,n$ e $\hat{y}_{t+h}$ representa a previsão de y_{t+h} com base nas observações até ao instante n. Note-se que l_t representa o nível da série temporal no instante t, b_t representa a tendência no instante t, s_t representa a componente sazonal no instante t, m representa o período de sazonalidade e k é a parte inteira de $(h-1)/m$. De forma a permitir que as equações de alisamento possam ser interpretadas como médias ponderadas, os parâmetros de alisamento α, β^* e γ estão limitados pela restrição indicada.

Os valores ajustados são calculados através da definição de $h = 1$ para $t = 1,2,...,n$. Previsões h passos à frente podem ser calculadas para $h = 1,2,...$ usando os últimos valores de tendência e sazonalidade estimados.

De forma a tornar possível a geração de intervalos de confiança e usar um critério de seleção de modelos, foi desenvolvida por Hyndman et al. (2008) uma *framework* estatística onde um modelo inovativo de espaço de estados pode ser determinado para cada método de alisamento exponencial. Cada modelo compreende uma equação da observação, que descreve os dados, e uma ou mais equações de estado, que descrevem a variação, ao longo do tempo, das componentes não observadas: o nível, a tendência e a sazonalidade. Para cada método de alisamento exponencial, são considerados dois modelos de espaço de estados: um com erros aditivos e um com erros multiplicativos, resultando num total de 18 modelos. De forma a permitir a distinção entre os modelos com erro aditivo e os modelos com erro multiplicativo, é adicionada uma letra referente ao Erro. Assim, (E,T,S) identifica o tipo de erro, tendência e sazonalidade. Na forma geral, o modelo de espaço de estados é dado por:

$$y_t = \omega(x_{t-1}) + r(x_{t-1})\varepsilon_t \tag{2.17}$$

$$x_t = f(x_{t-1}) + g(x_{t-1}), \varepsilon_t \tag{2.18}$$

onde y_t indica a observação no instante t, x_t é o vetor de estados, $\{\varepsilon_t\}$ é um processo de ruído branco com variância σ^2, $\omega(.)$ é a função de observação, $r(.)$ é a função relativa ao erro, $f(.)$ é a função de estado e $g(.)$ é a função de persistência.

A equação da observação indica a relação entre as observações e os estados não observados, por sua vez, a equação de estado mostra a evolução do estado ao longo do tempo.

Por exemplo, o modelo $ETS(A,A,A)$ é definida da forma (Hyndman and Athanasopoulos, 2014)

$$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t \tag{2.19}$$

$$l_t = l_{t-1} + b_{t-1} + \alpha\varepsilon_t \tag{2.20}$$

$$b_t = b_{t-1} + \beta\varepsilon_t \tag{2.21}$$

$$s_t = s_{t-m} + \gamma\varepsilon_t \tag{2.22}$$

e o modelo $ETS(M,A,A)$ é definido da forma (Hyndman and Athanasopoulos, 2014)

$$y_t = (l_{t-1} + b_{t-1} + s_{t-m})(1+\varepsilon_t) \tag{2.23}$$

$$l_t = l_{t-1} + b_{t-1} + \alpha(l_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t \tag{2.24}$$

$$b_t = b_{t-1} + \beta(l_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t \tag{2.25}$$

$$s_t = s_{t-m} + \gamma(l_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t. \tag{2.26}$$

Estimação de Modelos de Espaço de Estados

As estimativas de máxima verosimilhança dos parâmetros e estados iniciais dos modelos de espaço de estados podem ser obtidas maximizando a sua verosimilhança. A função de densidade de probabilidade para $y = (y_1, ..., y_n)'$ é dada por (Hyndman et al., 2008):

$$p(y|\theta, x_0, \sigma^2) = \prod_{t=1}^{n} p(y_t|x_{t-1}) = \prod_{t=1}^{n} p(\varepsilon_t)/|r(x_{t-1})|, \tag{2.27}$$

onde θ é o vetor dos parâmetros, x_0 é o vetor dos estados iniciais e σ^2 é a variância do ruído branco.

Considerando que $\{\varepsilon_t\}$ segue uma distribuição Gaussiana,

$$\mathscr{L}(\theta, x_0, \sigma^2|y) = (2\pi\sigma^2)^{-n/2} \left| \prod_{t=1}^{n} r(x_{t-1}) \right|^{-1} \exp\left(-\frac{1}{2} \sum_{t=1}^{n} \varepsilon_t^2/\sigma^2 \right), \tag{2.28}$$

por conseguinte, o seu logaritmo é:

$$\log\mathscr{L} = -\frac{n}{2}\log(2\pi\sigma^2) - \sum_{t=1}^{n} \log|r(x_{t-1})| - \frac{1}{2}\sum_{t=1}^{n} \varepsilon_t^2/\sigma^2. \tag{2.29}$$

A estimativa de σ^2 pela máxima verosimilhança pode ser obtida derivando parcialmente a equação 2.29 em ordem a σ^2 e igualando a mesma a 0:

$$\sigma^2 = n^{-1} \sum_{t=1}^{n} \varepsilon_t^2. \tag{2.30}$$

Usando esta estimativa, pode-se eliminar σ^2 da equação 2.28, resultando em:

$$\mathscr{L}(\theta, x_0|y) = (2\pi e \sigma^2)^{-n/2} \left| \prod_{t=1}^{n} r(x_{t-1}) \right|^{-1}. \tag{2.31}$$

Desta forma,

$$-2\log \mathscr{L}(\theta, x_0|y) = c_n + n\log\left(\sum_{t=1}^{n} \varepsilon_t^2\right) + 2\log|r(x_{t-1})|, \tag{2.32}$$

onde $c_n = n\log(2\pi e) - n\log(n)$. Assim as estimativas de máxima verosimilhança para os parâmetros θ e os estados iniciais x_0 podem ser obtidos minimizando a equação:

$$\mathscr{L}^*(\theta, x_0) = n\log\left(\sum_{t=1}^{n} \varepsilon_t^2\right) + 2\log|r(x_{t-1})|. \tag{2.33}$$

Os modelos de espaço de estados podem assim ser calculados usando as relações:

$$\varepsilon_t = [y_t - \omega(x_{t-1})]/r(x_{t-1}) \tag{2.34}$$

$$x_t = f(x_{t-1}) + g(x_{t-1})\varepsilon_t. \tag{2.35}$$

Critério de Informação para Seleção de Modelos

Embora normalmente não seja possível, devido à quantidade diminuta de erros de previsão disponíveis, as medidas de erro podem ser usadas para a seleção de modelos, desde que calculadas num conjunto de dados de teste. Para contornar este facto, é possível usar um critério de informação baseado na verosimilhança, $\mathscr{L}^*(\theta, x_0)$, que inclua um termo de regularização para evitar *overfitting*. O *Akaike Information Criteria* para modelos de espaço de estados é definido da forma seguinte (Hyndman and Athanasopoulos, 2014):

$$AIC = -2\log \mathscr{L}(\theta, x_0|y) + 2k, \tag{2.36}$$

onde $\mathscr{L}^*(\theta, x_0)$ é a verosimilhança e k é o número de parâmetros e estados iniciais do modelo estimado. O *Akaike Information Criteria* é baseado no *Kullback-Liebler (K-L) discrimination information*, também conhecido por entropia negativa, que é dado por:

$$I(f, g) = \int f(x) \log\left(\frac{f(x)}{g(x|\theta)}\right) dx, \tag{2.37}$$

mede a informação perdida quando o modelo g é usado para estimar o modelo real f. *Kullback-Liebler* descobriu que é possível estimar a probabilidade do critério de informação de *K-L* através da correção da maximização do logaritmo da verosimilhança para o enviesamento. Assim, escolhe-se o modelo que, de entre os candidatos, apresenta menor valor de *AIC*. O critério de informação Bayesiano é definido por (Schwarz, 1978):

$$BIC = AIC + k[log(n) - 2]. \tag{2.38}$$

O *BIC* está em consistência com a ordem, porém, não é tão eficiente assimptoticamente como o *AIC*. O *AIC* corrigido de enviesamento para pequenas amostras, representado por AIC_c, é dado por (Hyndman et al., 2008):

$$AIC_c = AIC + \frac{k(k+1)}{n-k-1}. \tag{2.39}$$

Desta forma, *AIC*, *BIC* e AIC_c favorecem a escolha de modelos apropriados.

2.3.3 Modelos ARIMA

Os modelos ARIMA (*Autoregressive Integrated Moving Average*) permitem a representação de vários tipos de séries temporais estocásticas sazonais ou não sazonais. A sua aplicação compreende processos puramente auto-regressivos (AR), puramente médias móveis (MA) e processos mistos (AR) e (MA), desde que aplicados a dados estacionários (Oliveira and Ramos, 2019).

Embora muitas séries temporais não sejam estacionárias, é possível transforma-las através de diferenciação. É habitual realizar diferenciação regular, sazonal ou ambas. Frequentemente, os modelos ARIMA são aceites como uma das classes de modelos mais versáteis para a previsão de séries temporais (Oliveira and Ramos, 2019).

A forma do modelo ARIMA sazonal multiplicativo, $ARIMA(p,d,q) \times (P,D,Q)_m$ é dada por (Wilson, 2016):

$$\phi_p(B)\Phi_p(b^m)(1-B)^d(1-B^m)^D y_t = c + \theta(B)\Theta_Q(B^m)\varepsilon_t, \tag{2.40}$$

onde

$$\phi_p(B) = 1 - \phi_1 B - ... - \phi_p B^p,$$

$$\theta_q(B) = 1 + \theta_1 B + ... + \theta_q B^q,$$

$$\Phi_P(B^m) = 1 - \Phi_1 B^m - ... - \Phi_P B^{Pm},$$

$$\Theta_Q(B^m) = 1 + \Theta_1 B^m + ... + \Theta_Q B^{Qm},$$

sendo m o período da sazonalidade, D o grau de diferenciação sazonal, d o grau de diferenciação regular, B o operador de atraso, $\phi_p(B)$ e $\theta_q(B)$ os polinomiais autorregressivos regulares de ordem p e q respetivamente, $\Phi_P(B)$ e $\Theta_Q(B)$ os polinomiais autorregressivos sazonais de ordem P e Q respetivamente e $c = \mu(1-\phi_1-...-\phi_p)(1-\Phi_1-...-\Phi_P)$, onde μ é a média de $(1-B)^d(1-B^m)^D y_t$ e ε_t é um processo de ruído branco Gaussiano de média 0 e variância σ^2.

De forma a assegurar causalidade e invertibilidade, as raízes dos polinómios $\phi_p(B)$, $\theta_q(B)$, $\Phi_P(B^m)$ e $\Theta_Q(B^m)$ devem estar fora do círculo unitário. Determinar os valores de p,q,P,Q,d e D é uma das principais tarefas associadas ao uso de modelos ARIMA. Para tal, o procedimento a seguir deve abordar a representação visual da série temporal, identificação de *outliers* e escolha de uma forma adequada de transformação para estabilização de variância, se necessário. Para o efeito, pode ser aplicada a transformação de *Box-Cox* definida da forma (Box and Cox, 1964):

$$y_t^{'} = \begin{cases} \ln(y_t), & \lambda = 0 \\ (y_t^{\lambda}-1)/\lambda, & \lambda \neq 0^{'} \end{cases} \tag{2.41}$$

onde o parâmetro λ é um número real, habitualmente entre -1 e 2.

Outro passo importante é escolher os graus de diferenciação adequados, D e d. Para tal, podem ser calculados o *ACF* (*AutoCorrelation Function*) e o *PACF* (*Partial AutoCorrelation Function*). De forma de alternativa, podem ser aplicados testes de raizes unitárias. O teste *Canova–Hansen* (Canova and Hansen, 1995) pode ser usado para escolher D; depois de selecionado D, d pode ser escolhido aplicando o teste *KPSS (Kwiatkowski, Phillips, Schmidt & Shin)* (Kwiatkowski et al., 1992). Finalmente o *ACF* e o *PACF* da amostra são cruzados com padrões teóricos dos modelos conhecidos para determinar p,q,P e Q.

Critério de Informação para Seleção de Modelos

Da mesma forma que os modelos de espaço de estados, os valores de p,q,P e Q podem ser determinados através de um critério de informação, tal como *Akaike Information Criteria* (Hyndman and Athanasopoulos, 2014):

$$AIC = -2\log\mathscr{L}(\theta,\sigma^2|y) + 2(p+q+P+Q+k+1), \tag{2.42}$$

onde $k = 1$ se $c \neq 0$ e 0 caso contrário, k é o número de parâmetros estimados e $\log\mathscr{L}(\theta,\sigma^2|y)$ é o logaritmo da verosimilhança ajustado aos dados transformados e diferenciados, dado por (Box and Jenkins, 1994):

$$\log\mathscr{L}(\theta,\sigma^2|y) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \sum_{t=1}^{n}\frac{\varepsilon_t^2}{2\sigma^2} \ , \tag{2.43}$$

onde θ é o vetor dos parâmetros do modelo e σ^2 é a variância de um processo de ruído branco.

É importante referir que o *AIC* é definido considerando os mesmos princípios da máxima verosimilhança e entropia negativa referidos na secção 2.3.2.

O AIC_c é dado por:

$$AIC_c = AIC + \frac{2(p+q+P+Q+k+1)(p+q+P+Q+k+2)}{n-p-q-P-Q-k-2}. \tag{2.44}$$

O Critério de informação Bayesiano é dado por:

$$BIC = AIC + [\log(n) - 2](p+q+P+Q+k+1). \tag{2.45}$$

A minimização de qualquer uma das opções, *AIC*, AIC_c ou *BIC*, favorece a escolha de um modelo *ARIMA* apropriado.

2.3.4 Modelo TBATS

O modelo TBATS (*Trigonometric Box-Cox transform, ARMA errors, Trend, and Seasonal components*) (Livera et al., 2011) garante flexibilidade e capacidade para lidar com séries não lineares. Uma particularidade reside nas suas componentes sazonais baseadas em séries de Fourier:

$$y_t^{(\omega)} = \begin{cases} \frac{y_t^{(\omega)}-1}{\omega}, & \omega \neq 0 \\ \log y_t, & \omega = 0 \end{cases} \tag{2.46}$$

$$y_t^{(\omega)} = l_{t-1} + \phi b_{t-1} + \sum_{j=1}^{k_i} s_t^{(i)} + d_t \tag{2.47}$$

$$l_t = l_{t-1} + \phi b_{t-1} + \alpha d_t \tag{2.48}$$

$$b_t = (1-\phi)b + \phi b_{t-1} + \beta d_t \tag{2.49}$$

$$s_t^{(i)} = s_{t-m_i}^{(i)} + \gamma_i d_t \tag{2.50}$$

$$d_t = \sum_{i=1}^{p} \varphi_i d_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t, \tag{2.51}$$

$$s_t^{(i)} = \sum_{j=1}^{k_j} s_{j,t}^{(i)} \tag{2.52}$$

$$s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \tag{2.53}$$

$$s_{j,t}^{*(i)} = -s_{j,t-1} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t \tag{2.54}$$

onde

- $y_t^{(\omega)}$ representa a série y_t transformada por Box-Cox com parâmetro ω;
- $m_1, m_2, \ldots, m_T$ correspondem aos vários períodos de sazonalidade;

- l_t corresponde ao nível no instante t;
- b corresponde à tendência num horizonte temporal longo;
- b_t corresponde à tendência num horizonte temporal curto;
- $s_t^{(i)}$ representa a componente sazonal de ordem i no instante t;
- d_t representa o processo ARMA(p,q);
- ε_t representa ruído branco;
- α, β e γ_i para $i = 1,2,...,T$ correspondem aos parâmetros de alisamento;
- $s_{j,t}^{(i)}$ representa o nível estocástico da componente sazonal de ordem i;
- $s_{j,t}^{*(i)}$ representa o nível estocástico de crescimento da componente sazonal i e descreve as oscilações de sazonalidade ao longo do tempo;
- k_j corresponde ao número de harmónicos para a componente sazonal i;

atribuindo-se valores pares a m_i quando $k_i = m_i/2$ e ímpares quando $k_i = (m_i - 1)/2$ para abordar o problema.

2.3.5 Lasso

O Lasso (*Least Absolute Shrinkage and Selection Operator*) revela-se bastante útil quando aplicado na previsão de séries temporais com um elevado número de regressores e um número limitado de observações. Sendo possível controlar o número de regressores a incluir no modelo através do parâmetro λ, as vantagens da aplicação deste método tendem a diminuir quando o número de regressores diminui relativamente ao tamanho do conjunto de dados (Chan-Lau, 2017). As suas previsões tendem a apresentar um erro baixo, revelar fácil interpretação e captar as correlações locais das séries temporais (Li and Chen, 2014).

Considerando $\beta = (\beta_0, \beta_1, \beta_2, ..., \beta_p)^T$ e $X = (X_1, ...X_p)^T$, o *Lasso* assume o modelo de regressão usual dado por:

$$Y = \beta_0 + X_1\beta_1 + ... + X_p\beta_p + \varepsilon \tag{2.55}$$

onde os erros seguem uma distribuição normal e são independentes.

O procedimento de ajuste dos mínimos quadrados estima $\beta_0, \beta_1, \beta_2, ..., \beta_p$ determinando os valores que minimizam

$$RSS = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{i,j} \right)^2. \tag{2.56}$$

O Lasso funciona de uma forma análoga. Os seus coeficientes minimizam

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{i,j} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| = RSS + \lambda \sum_{j=1}^{p} |\beta_j| \tag{2.57}$$

onde $\lambda \geq 0$ é um parâmetro de regularização. Da mesma forma forma que os mínimos quadrados, o *LASSO* procura um ajuste aos dados tal que 2.57 seja mínimo. Porém, o segundo termo $\lambda|\beta_j|$ funciona como uma penalização e torna-se pequeno quando $\beta_1, \beta_2, ..., \beta_p$ são próximos de 0. Sob a forma de consequência, as estimativas de β_j tomam o valor 0.

O parâmetro de regularização λ é usado para controlar o impacto relativo destes dois termos presentes na estimativa dos coeficientes. Quando $\lambda = 0$ a penalização não surte efeito e a regressão resulta nas estimativas do método dos mínimos quadrados. Por outro lado, um crescimento de λ implica um crescimento no impacto do termo relativo à penalização, podendo resultar numa não inclusão de certas variáveis no modelo. Consequentemente, cada valor de λ resulta num conjunto diferente de estimativas, tornando a escolha do valor de λ fundamental.

A escolha do parâmetro λ, embora de extrema importância, pode ser realizada recorrendo a um processo bastante simples de validação cruzada. De uma forma geral, são escolhidos valores para λ, para cada um dos valores é ajustado um modelo e calculada uma medida de erro e finalmente é escolhido o valor de λ para o qual o erro é mínimo. Após realizado este procedimento, o modelo é reajustado usando todas as observações disponíveis.

2.3.6 Regressão de Componentes Principais

A Regressão de Componentes Principais (*Principal Components Regression - PCR*) faz parte de um conjunto de abordagens focadas na redução de dimensão. Este tipo de abordagens pode ser vantajoso quando aplicado de uma forma cuidada.

Considerando que $Z_1, Z_2, ..., Z_M$ representam $M < p$ combinações lineares das variáveis explicativas iniciais, vem (James et al., 2014):

$$z_m = \sum_{j=1}^{p} \phi_{mj} X_j \tag{2.58}$$

para as constantes $\phi_{1m}, \phi_{2m}, ..., \phi_{pm}$, $m = 1, 2, .., M$. Pode-se assim ajustar o modelo de regressão linear dado por:

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{i,} + \varepsilon_i, \quad i = 1, 2, ..., n, \tag{2.59}$$

usando o método dos mínimos quadrados.

Todos os métodos de redução de dimensão são constituídos por dois passos. Num passo inicial, os previsores $Z_1, Z_2, ..., Z_M$ são obtidos. Seguidamente o modelo, constituído pelos M previsores, é ajustado. Uma das opções para a realização da escolha dos previsores $Z_1, Z_2, ..., Z_M$ é a Regressão de Componentes Principais.

A aplicação da *PCR* envolve a determinação dos previsores e, numa fase posterior, o ajustamento de uma regressão linear usando o método dos mínimos quadrados. Este método segue o princípio de que um número reduzido de componentes é suficiente para explicar a variabilidade

dos dados, bem como a sua relação com a variável dependente. Vantagens associadas à *PCR* aparecem à medida que os dados seguem o princípio referido anteriormente, podendo a sua aplicação ajudar a evitar *overfitting*.

Importa referir que a *PCR* não faz parte do conjunto de abordagens focadas na seleção de regressores. Uma vez que cada M componentes principais usados na regressão são uma combinação linear das p variáveis explicativas originais, trata-se de uma abordagem focada na redução de dimensão.

Um cuidado a ter na implementação desta metodologia é a padronização das variáveis (Hastie et al., 2001). Na sua ausência, as variáveis com maior variância tendem a ter mais influência nos componentes principais resultantes.

2.3.7 Erros de Previsão

Devido a uma disponibilidade considerável de modelos de previsão, é sempre possível adequar um modelo de previsão a uma série temporal e gerar previsões do seu comportamento futuro. Porém, nem sempre é possível obter boas representações da realidade. Devido aos avanços da literatura em relação a este tema bem como à complexidade de alguns modelos disponíveis, é possível adequar um modelo aos dados de uma forma exaustiva incorrendo em *overfitting*, ou seja, aproximar o modelo em demasia na afinação dos parâmetros do mesmo podendo levar assim a que tenha um mau comportamento na realização das previsões futuras.

Conjuntos de treino e de teste

Daí nasce a necessidade de avaliar qual o melhor modelo e a sua exatidão, procedimento que é realizado com recurso à análise do erro. Para tal, uma prática possível é dividir o conjunto de dados em período de treino (usado para estimar os parâmetros do método de previsão) e período de teste (usado para avaliar a exatidão das previsões). Esta divisão permite avaliar o desempenho do modelo com base em previsões genuínas, eliminando a possibilidade de considerar adequado um modelo demasiado aproximado aos dados e que posteriormente poderia vir a realizar más previsões. Como os dados de teste não são usados para estimar o modelo, são uma boa indicação do desempenho futuro do mesmo.

É prática comum dividir o conjunto de dados de forma a que o tamanho do conjunto de dados de teste seja aproximadamente 20% do tamanho da amostra disponível. Porém, é aconselhável que o tamanho do conjunto de dados de teste não seja superior ao horizonte temporal máximo que se pretende prever (Hyndman and Athanasopoulos, 2014).

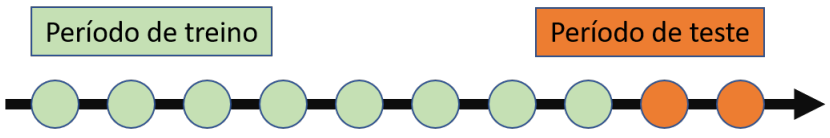

Figura 2.1: Representação da divisão do conjunto de dados em período de treino e teste.

Definição do Erro

Define-se erro de previsão como a diferença entre o valor real e a sua previsão gerada com base num método ou modelo. Os erros podem ser calculados com base em previsões a um passo ou então em previsões a $h > 1$ passos:

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T} \tag{2.60}$$

O cálculo dos erros de previsão serve como base à determinação da exatidão do modelo de várias formas. As medidas de exatidão permitem distinguir um modelo adequado ao problema de um modelo não adequado e permitem fazer a escolha do mesmo recorrendo à sua comparação direta. Problemas que abordam dados de diferentes dimensões carecem de uma escolha mais cuidada da medida de exatidão escolhida devido à existência de medidas que dependem da escala.

Medidas de Erro Dependentes de Escala: As medidas que dependem da escala são medidas bastante úteis quando usadas para calcular o desempenho de modelos de previsão aplicados ao mesmo conjunto de dados, porém a sua utilidade é praticamente nula quando usadas na determinação da performance geral de um modelo aplicado a um conjunto de dados de diferentes dimensões. Como exemplo destas medidas temos:

Erro Médio (*ME*)

$$ME = mean(e_t) = \frac{1}{m}\sum_{t=1}^{m} e_t \tag{2.61}$$

Erro Quadrático Médio (*MSE*)

$$MSE = mean(e_t^2) = \frac{1}{m}\sum_{t=1}^{m} (e_t^2) \tag{2.62}$$

Raiz Quadrada do Erro Quadrático Médio (*RMSE*)

$$RMSE = \sqrt{MSE} = \sqrt{mean(e_t^2)} = \sqrt{\frac{1}{m}\sum_{t=1}^{m} (e_t^2)} \tag{2.63}$$

Erro Absoluto Médio (*MAE*)

$$MAE = mean(|e_t|) = \frac{1}{m}\sum_{t=1}^{m} |e_t| \tag{2.64}$$

Estes erros apresentam vantagens centradas na simplicidade de cálculo e compreensão, porém, enquanto cada um deles apresenta desvantagens em tópicos distintos, todos eles apresentam desvantagens na comparação entre modelos de previsão aplicados a conjuntos de dados de diferentes dimensões, tornando impossível fazer uma análise a vários níveis para mitigar as desvantagens dessa forma.

Medidas de Erro Independentes de Escala Decorrente da necessidade de comparação do desempenho de modelos diferentes aplicados a conjuntos de dados de diferente dimensão, as medidas baseadas em erros relativos recorrem a um termo de comparação comum, normalmente o método naïve, para mitigar as diferenças de escala e assim, tornar a comparação independente das diferenças de dimensão dos conjuntos de dados. Desta forma, considerando $r_t = e_t/e_t^b$ para $t = 1,2,3,...,m$, define-se como:

Erro Absoluto Relativo Médio (*MRAE*)

$$MRAE = mean(|r_t|) = \frac{1}{m}\sum_{t=1}^{m}|r_t| \tag{2.65}$$

Note-se que neste caso, o cálculo de r_t carece da divisão pelo erro da previsão de referência diferente de 0 e obriga a que $e_t \neq e_t^b$. Para além disso, um dos grandes problemas das medidas baseadas em erros relativos aparece quando e_t^b se revela relativamente pequeno, aumentando o valor de r_t. Para contornar este facto, podem ser usadas medidas baseadas em medidas relativas. Revelando vantagens na sua interpretação, estas medidas apresentam desvantagens devido à necessidade de aplicação de vários modelos à mesma série temporal para permitir a sua determinação, podendo ser um problema quando o poder computacional ou o tempo são limitados.

Erro Absoluto Médio Relativo (*RelMAE*)

$$RelMAE = \frac{MAE}{MAE^b}, \tag{2.66}$$

$MAE = mean(|e_t|) = \frac{1}{m}\sum_{t=1}^{m}|e_t|, \quad MAE^b = mean(|e_t^b|) = \frac{1}{m}\sum_{t=1}^{m}|e_t^b|.$

Raiz Quadrada do Erro Quadrático Médio Relativo (*RelRMSE*)

$$RelRMSE = \frac{RMSE}{RMSE^b} \tag{2.67}$$

$RMSE = \sqrt{mean(e_t^2)} = \sqrt{\frac{1}{m}\sum_{t=1}^{m}(e_t^2)}, \quad RMSE^b = \sqrt{mean(e_t^2)^b} = \sqrt{\frac{1}{m}\sum_{t=1}^{m}(e_t^2)^b}.$

Média Geométrica do Erro Absoluto Médio Relativo (*AvgRelMAE*)

$$AvgRelMAE = gmean(\frac{MAE_i}{MAE_i^b}) = \sqrt[N]{\prod_{i=1}^{N}\frac{MAE_i}{MAE_i^b}} = exp\left[\frac{1}{N}\sum_{i=1}^{N}log(\frac{MAE_i}{MAE_i^b}).\right] \tag{2.68}$$

É importante ter em consideração que o valor do MAE do método de referência, seja este o método naïve ou não, deve ser calculado apenas com o conjunto de dados de treino de forma a obter uma medida estável da escala dos dados.

Erro Escalado Absoluto Médio (*MASE*) Por vezes, contornar as diferenças de escala do conjunto de dados não é suficiente. Os erros relativos têm uma distribuição probabilística de média indefinida e variância infinita enquanto que as medidas relativas podem ser determinadas apenas quando existem várias previsões para a série temporal. Para contornar este problema, Hyndman and Koehler (2006) criaram uma alternativa que consiste em escalar o erro com base no erro médio absoluto intrínseco proveniente da previsão usando o método naïve. Desta forma, para uma série temporal não sazonal, o erro é definido como:

$$q_t = \frac{e_t}{\frac{1}{n-1}\sum_{i=2}^{n}|Y_i - Y_{i-1}|} \quad para \quad t = 1,2,3,...,m. \tag{2.69}$$

Já para séries temporais sazonais, considerando s como o número de períodos por época, vem:

$$q_t = \frac{e_t}{\frac{1}{n-s}\sum_{i=s+1}^{n}|Y_i - Y_{i-s}|} \quad para \quad t = 1,2,3,...,m \tag{2.70}$$

Assim, o erro escalado absoluto médio pode ser calculado da seguinte forma:

$$MASE = mean(q_t) = \frac{1}{m} = \sum_{t=1}^{m}|q_t| \tag{2.71}$$

Revelando-se apenas inadequado quando o conjunto de dados apresenta sempre o mesmo valor, o MASE é defendido pela literatura como sendo mais robusto e abrangente de entre as opções. A sua utilidade revela-se na capacidade de conseguir lidar, ao mesmo tempo, com os problemas acima referidos para cada método, o MASE prova-se útil para previsões a um passo ou para previsões a mais que um passo, tem um comportamento satisfatório em séries temporais com valores indefinidos ou infinitos, não se deixa afetar pelas diferenças de escala do conjunto de dados e é de fácil interpretação. Quando toma valores superiores a 1, significa que o método de previsão em análise resulta em previsões piores, em média, que as previsões resultantes do método naïve.

Em contrapartida, alguns erros apresentam-se mais adequados que o MASE em alguns exemplos, porém apenas devido à sua simplicidade de cálculo relativamente ao mesmo. Em suma, em situações em que o conjunto de dados apresenta bastantes variações de escala, entre outras características, o MASE é a medida mais indicada para o cálculo da exatidão da previsão. Hyndman and Koehler (2006)

2.4 Características das Séries Temporais

As características das séries temporais não só permitem a distinção entre séries temporais, como também reduzem a influência de *outliers*. Um estudo realizado por Fulcher and Jones (2014) identificou 9000 operações para extrair características de uma série temporal.

Como o que se pretende com o uso das características é um bom agrupamento das séries temporais e um tempo de processamento reduzido, foram escolhidas características que vão de acordo com as exigências. A Tabela A.1 descreve de uma forma breve cada uma das características usadas e os parágrafos seguintes complementam a informação apresentada na mesma.

Características baseadas numa decomposição STL: A força de tendência, a força de sazonalidade, a linearidade, a curvatura, a *spikiness* e o primeiro coeficiente de autocorrelação da série *remainder* são calculadas com base numa decomposição da série temporal. Considerando $y_1, y_2, ..., y_T$ como sendo uma série temporal, num primeiro passo a mesma é transformada por um processo de Box-Cox (Guerrero, 1993) de forma a estabilizar a variância e tornar o efeito sazonal aditivo. A série é decomposta usando STL (Cleveland et al., 1990) para resultar em $y_t^* = T_t + S_t + R_t$, onde T_t representa a tendência, S_t representa a componente sazonal e R_t representa a componente *remainder*. Para séries temporais não sazonais, *Friedman's super smoother* (Friedman, 1984) é usado para decompor $y_t^* = T_t + R_t$ e $S_t = 0$ para qualquer t. A série desprovida de tendência é dada por $y_t^* - T_t = S_t + R_t$ e a série desprovida de sazonalidade é dada por $y_t^* - S_t = T_t + R_t$.

A força de tendência é medida comparando a variância da série desprovida de sazonalidade e da série *remaider* (Smith-Miles, 2008):

$$Trend = \max[0, 1 - var(R_t)/var(T_t + R_t)]. \tag{2.72}$$

A força de sazonalidade é dada por:

$$Seasonality = \max[0, 1 - var(R_t)/var(S_t + R_t)]. \tag{2.73}$$

A linearidade e a curvatura são baseada nos coeficientes de uma regressão quadrática ortogonal dada por:

$$T_t = \beta_0 + \beta_1 \phi_1(t) + \beta_2 \phi_2(t) + \varepsilon_t, \tag{2.74}$$

onde $t = 1, 2, .., T$ e ϕ_1 e ϕ_2 são polinomiais ortogonais de ordem 1 e 2. O valor estimado de β_1 é usado como medida de linearidade enquanto o valor estimado de β_2 é considerado como medida de curvatura. Estas características são usadas por Hyndman et al. (2015). A linearidade e a curvatura dependem da escala da série temporal, então as séries temporais são escaladas de forma a que a média seja 0 e a variância 1 antes do cálculo destas características.

A *spikeness* é útil quando a série temporal é afetada por *outliers* ocasionais. Hyndman et al. (2015) introduziram um índice para medir a *spikeness*, sendo esta a variância das variâncias *leave one out* de r_t. Determina-se o primeiro coeficiente de autocorrelação da série *remainder*.

Estabilidade e granulosidade: A estabilidade e a granulosidade de uma série temporal são calculadas com base numa divisão da série em intervalos não sobrepostos. Para cada janela, a média e a variância são calculadas. A estabilidade diz respeito à variância das médias enquanto a granulosidade diz respeito à variância das variâncias. Estas foram as primeiras características usadas por Hyndman et al. (2015).

Entropia espectral de uma série temporal: A entropia espectral é baseada em teoria da informação e pode ser usada como indicador da possibilidade de previsão de uma série temporal.

Séries de fácil previsão são características de uma entropia espectral reduzida, por outro lado, séries bastante ruidosas tendem a resultar em valores grandes desta característica. Para o seu cálculo é usada a medida introduzida por Goerg (2013). Esta estima a entropia de *Shannon* da densidade espectral normalizada, dada por:

$$H_s(y_t) = -\int_{-\pi}^{\pi} \hat{f}_y(\lambda) \log \hat{f}_y(\lambda) d\lambda, \tag{2.75}$$

onde $\hat{f}_y$ representa a estimativa da densidade espectral introduzida por Nuttall and Carter (1982).

Expoente de Hurst: O expoente de Hurst mede a memória de longo prazo de uma série temporal. É dado por $H = d + 0.5$, onde d é o grau de diferenciação fracional obtido pelo ajuste de um modelo $ARFIMA(0,d,0)$. O seu valor é determinado usando o método da máxima verosimilhança (Haslett and Raftery, 1989). Esta medida foi também usada por Smith-Miles (2008).

Não Linearidade: Para medir o grau de não linearidade de uma série temporal de uma série temporal, foi usado o valor da estatística calculada no teste de rede neuronal para não linearidade de Terasvirta (Teräsvirta et al., 1993). Esta medida foi também usada por Smith-Miles (2008). Note-se que esta característica assume valores elevados quando a série é não linear e valores reduzidos quando a série é linear.

Estimativas de parâmetros de um modelo ETS: O processo $ETS(A,A,N)$ (Hyndman et al., 2008) pode ser definido por:

$$y_t = l_{t-1} + b_{t-1} + \varepsilon_t \tag{2.76}$$

$$l_t = l_{t-1} + b_{t-1} + \alpha\varepsilon_t \tag{2.77}$$

$$b_t = b_{t-1} + \beta\varepsilon_t, \tag{2.78}$$

onde α é o parâmetro de alisamento para o nível e β é o parâmetro de alisamento para a tendência. Ambos os parâmetros são incluídos no conjunto de características calculadas. Representam a variabilidade no nível e na tendência, respetivamente.

Por sua vez, o processo $ETS(A,A,A)$ (Hyndman et al., 2008) pode ser definido por:

$$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t \tag{2.79}$$

$$l_t = l_{t-1} + b_{t-1} + s_{t-m} + \alpha\varepsilon_t \tag{2.80}$$

$$b_t = b_{t-1} + \beta\varepsilon_t, \tag{2.81}$$

$$s_t = s_{t-m} + \gamma\varepsilon_t, \tag{2.82}$$

onde γ é o parâmetro de alisamento para a componente sazonal. O seu valor traduz a variabilidade da componente sazonal de uma série temporal. O paramento γ também foi incluído no conjunto de características calculadas.

Testes de raiz unitária: O teste de Phillips-Perron é baseado na regressão:

$$y_t = c + \alpha y_{t-1} + \varepsilon_t \tag{2.83}$$

Diz respeito ao valor da estatística de teste "Z-alpha" com o parâmetro janela de Bartlett igual ao valor inteiro de $4(T/100)^{0.25}$ (Pfaff, 2008).

O teste KPSS é baseado na regressão:

$$y_t = c + \delta_t + \alpha y_{t-1} + \varepsilon_t \tag{2.84}$$

Diz respeito ao valor da estatística de teste "KPSS" com o parâmetro janela de Bartlett igual ao valor inteiro de $4(T/100)^{0.25}$ (Pfaff, 2008).

Características baseadas em coeficientes de autocorrelação: É calculado o primeiro valor da função de autocorrelação e a soma dos quadrados dos cinco primeiros valores da função de autocorrelação da série original, da série diferenciada, da série diferenciada duas vezes e da série respetiva ao *lag* sazonal. Um modelo linear é ajustado à série temporal e o primeiro valor da função de autocorrelação dos resíduos é calculado. É também calculada a soma dos quadrados dos cinco primeiros valores da função de autocorrelação parcial da série temporal, da série diferenciada e da série diferenciada duas vezes.

Capítulo 3

Caso de Estudo

3.1 Conjunto de Dados

O Grupo Jerónimo Martins procura responder às necessidades de um vasto número de consumidores no que diz respeito a retalho especializado e distribuição alimentar. Contanto com uma história iniciada em 1972, emprega hoje em dia mais de 10800 colaboradores distribuídos por três países: Portugal, Polónia e Colômbia. De entre as áreas de negócio, destaca-se a distribuição alimentar que assegura mais de 95% das vendas consolidadas do grupo, garantindo assim a liderança no setor através da cadeia de lojas Pingo Doce e Recheio.

Os dados sobre os quais este trabalho foi desenvolvido englobam as vendas de 12 lojas de supermercado Pingo Doce durante um período de 173 semanas compreendido entre 3 de janeiro de 2012 e 27 de abril de 2015. Os mesmos foram fornecidos ao INESC TEC no âmbito do um protocolo de colaboração entre as duas empresas. Importa referir que as análises foram realizadas com dados de vendas semanais, uma vez que esta é a periodicidade que mais vantagens oferece ao retalhista.

Foram selecionados para a realização desta dissertação os SKUs que apresentam vendas em todas as semanas. Estes SKUs são habitualmente referidos na literatura com a designação de *fast mooving goods* (Huang et al., 2014; Trapero et al., 2013, 2012, 2014). Acrescente-se que são estes produtos que têm o maior impacte na receita da empresa e que exigem uma gestão mais eficiente da logística da *Supply Chain*.

Uma previsão o mais exata possível deste tipo de SKUs é imperativa. Uma previsão deficiente pode levar a que as lojas não garantam a satisfação dos clientes ou que incorram em custos de armazenamento desnecessários. No limite, relativamente a produtos perecíveis, pode levar a perdas elevadas (Hugos, 2018).

Os produtos nos quais esta dissertação é baseada estão distribuídos pelas seis áreas principais do retalhista: perecíveis especializados, mercearia, perecíveis não especializados, produtos pessoais, detergentes e produtos de limpeza e bebidas.

Para o estudo desenvolvido neste trabalho foram selecionadas as lojas de maior e menor dimensão disponibilizadas, referenciadas a partir de agora como loja 412 (com 988 *fast mooving*

goods) e loja 843 (com 717 *fast mooving goods*), respetivamente.

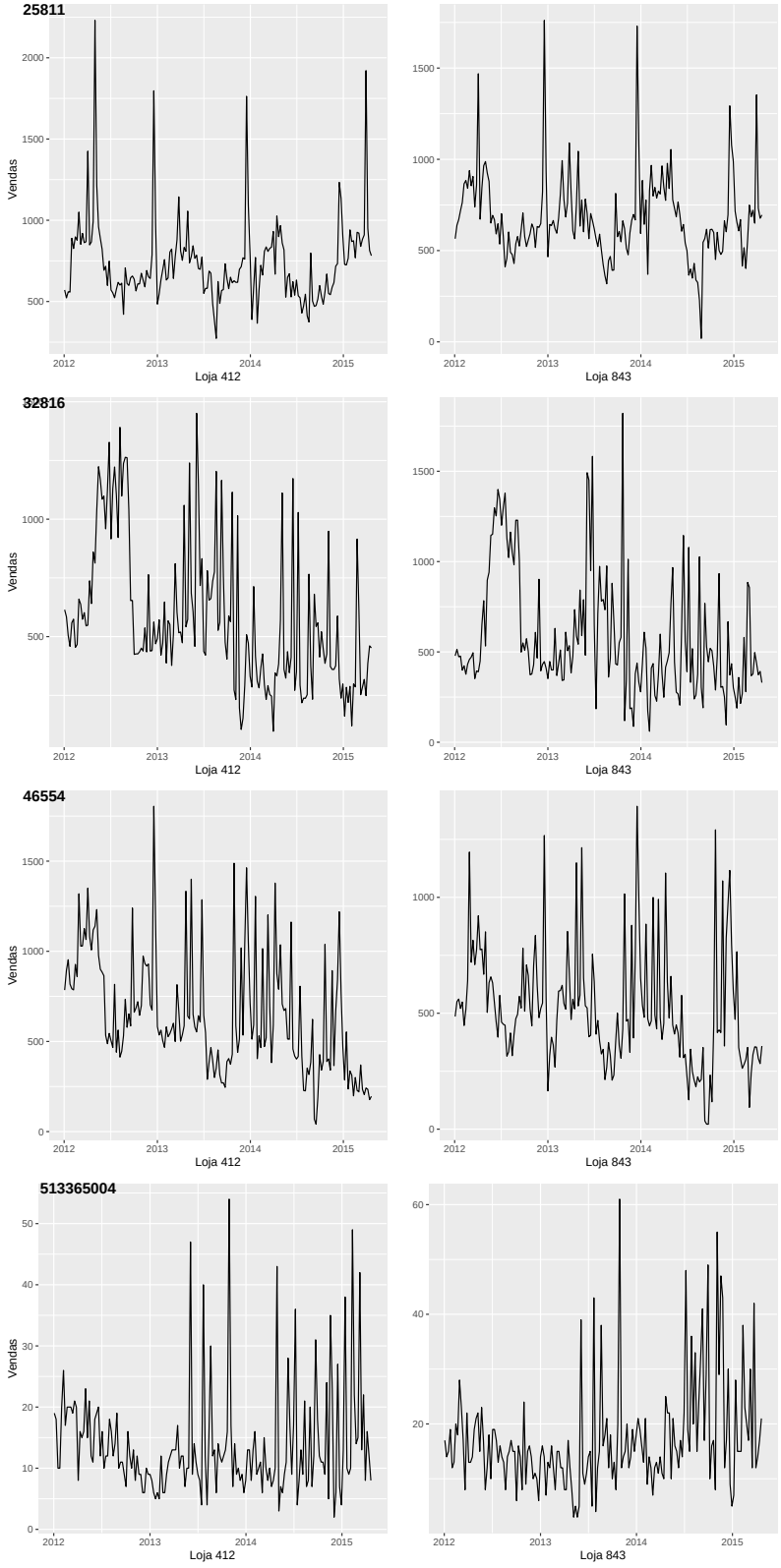

Figura 3.1: Seleção de SKUs das lojas 412 (à esquerda) e 843 (à direita).

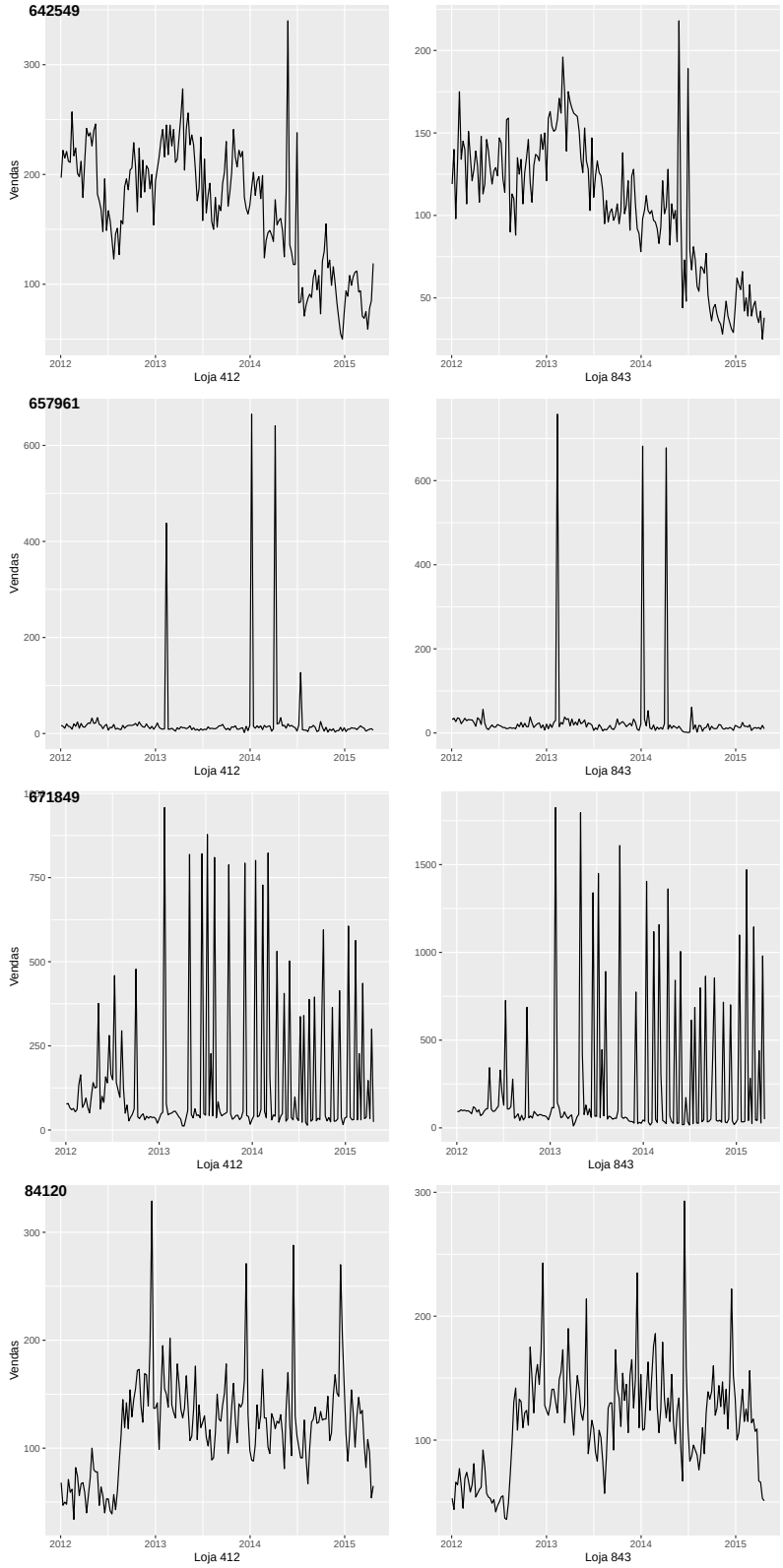

Figura 3.2: Seleção de SKUs das lojas 412 (à esquerda) e 843 (à direita).

Estas duas lojas apresentam 558 SKUs comuns, o que representa 56% dos SKUs da loja 412

(grande) e 78% dos SKUs da loja 843 (pequena). Uma análise comparativa entre estas lojas permitiu concluir que a evolução temporal das vendas semanais de cada SKU, embora numa escala diferente, é em certos casos semelhante. As Figuras 3.1, 3.2 e A.1 apresentam uma seleção de SKUs comuns a ambas as lojas. Como se pode observar, cada SKU revela um comportamento equivalente seguindo padrões idênticos, quer de tendência quer de sazonalidade. Rice (1976) defende que a eficiência dos métodos de previsão varia de acordo com a natureza dos dados. Explorar essa natureza pode ser útil para a seleção do modelo mais apropriado para uma série temporal, Assim, é expectável que o modelo mais adequado para os SKUs comuns a ambas as lojas seja o mesmo.

3.2 Características de Séries Temporais

Hyndman (2001), Lawrence (2001) e Armstrong (2001) argumentam que as características de uma série temporal podem proporcionar informação útil para a determinação dos métodos mais apropriados para previsão. Neste trabalho, em vez de utilizar diretamente as observações das séries temporais, propõe-se caracterizar as mesmas através de um conjunto de características. As características das séries temporais permitem a distinção entre as mesmas, mas ao invés das observações mitigam a influência de *outliers* e ausência de dados (Wang et al., 2006). Define-se por característica qualquer medida que possa ser calculada com base nos valores de uma série temporal (Talagala et al., 2018).

Uma vez que o objetivo é a previsão, deverão ser selecionadas características de séries temporais que possuam poder discriminatório na seleção de um modelo de previsão apropriado. Neste trabalho foram consideradas as características utilizadas por Talagala et al. (2018) e Montero-Manso et al. (2020) visto que o foco é coincidente.

A Tabela A.1 descreve de uma forma sumária cada uma dessas características usadas neste trabalho, num total de 36. Estas características devem captar a estrutura dinâmica das séries temporais, nomeadamente, tendência, sazonalidade, autocorrrelação, não linearidade, heterogeidade, etc. Acrescente-se que estas características são independentes de escala e assimptoticamente independentes do comprimento das séries temporais. Uma vez que a metodologia de previsão proposta objetiva a previsão rápida de séries temporais, baseada nas suas características, só foram escolhidas as que não exigem esforço computacional.

Uma vez calculadas as características de cada uma das séries temporais, pode usar-se um método de redução de dimensão, nomeadamente análise de componentes principais, que as projete num espaço bidimensional de modo a que seja possível a sua visualização. A Figura 3.3 mostra os dois primeiros componentes principais das características das séries temporais de vendas semanais dos SKUs das lojas 412 e 843. As características dos SKUs apresentados nas Figuras 3.1, 3.2 e A.1 encontram-se a vermelho (loja 412) e a azul (loja 843). Como se pode observar na Figura 3.3, as características do mesmo SKU nas duas lojas encontram-se próximas, o que motiva a utilização do método de agrupamento de dados para determinação do modelo de previsão mais adequado.

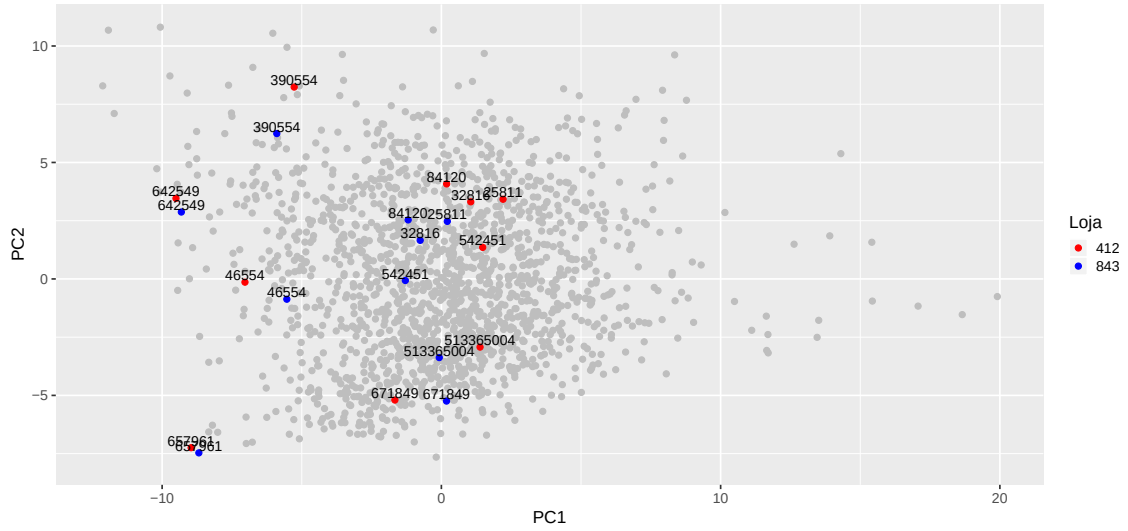

Figura 3.3: Distribuição dos SKUs da loja 412 e da loja 843 no espaço das características.

3.3 *Framework*

O trabalho desenvolvido evoluiu no sentido de criar uma *framework* de previsão rápida de vendas construída com base em modelos estatísticos e métodos de *clustering*. Constituída por duas fases, uma *offline* e uma *online*, da *framework* resulta a seleção do modelo apropriado a aplicar a uma série temporal (ver Figura 3.4).

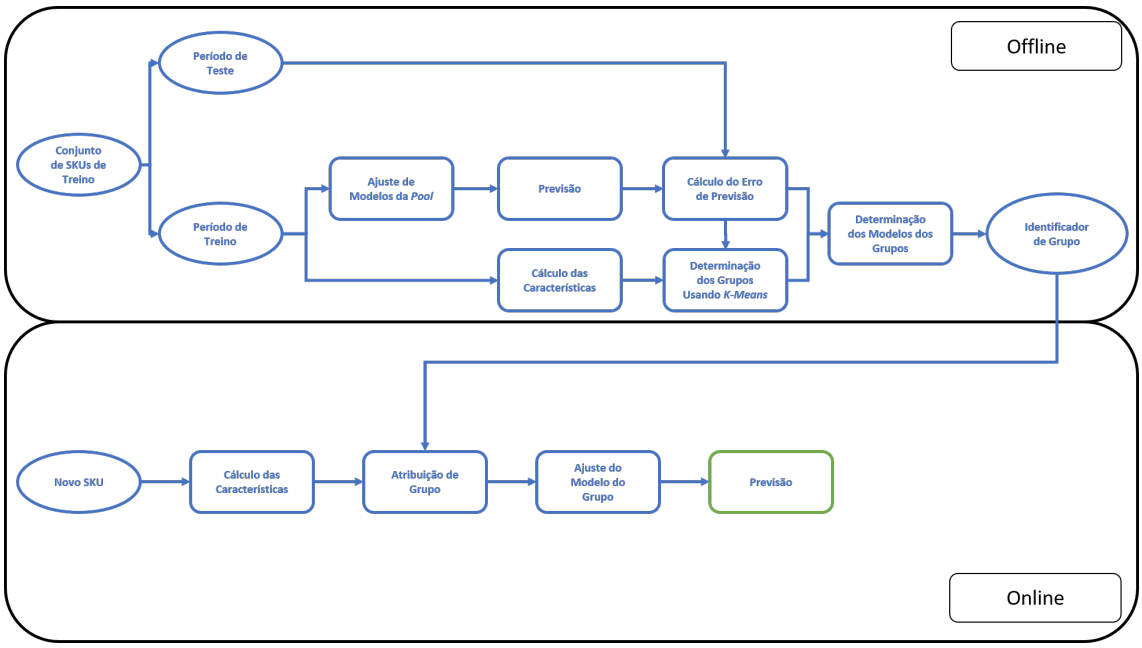

Figura 3.4: Representação gráfica da *framework* proposta: funcionamento *offline* e *online*.

A *framework* agrupa as séries temporais de treino com base nas suas características e nos erros resultantes da aplicação dos modelos de previsão considerados na *pool*. Para tal, o conjunto de dados de treino é dividido em período de treino e período de teste. Após esta divisão, o período de treino é utilizado para: (1) o cálculo do valor das características e para, (2) o ajuste de todos os modelos da *pool*. Os modelos ajustados são posteriormente usados para calcular as previsões para o conjunto de teste. Os erros de previsão resultantes e o valor das características são de seguida utilizados para formar os grupos de séries temporais usando o método *k-means*. Numa fase posterior, é determinado o modelo de previsão associado a cada grupo. Esta determinação é feita com base no desempenho de cada modelo em cada grupo, sendo atribuído ao grupo o modelo com o melhor desempenho, com base num erro de previsão.

Todos os passos referidos anteriormente constituem a fase *offline* da *framework*. Na fase *online*, apenas é necessário o cálculo das características (usando o histórico de observações) e a identificação do centroide mais próximo, obtida usando o Identificador de Grupo treinado na fase *offline*. Depois de calculadas as distâncias entre as características da nova série temporal e as características dos centroides de cada grupo, é atribuído à série o grupo cujo centroide lhe está mais próximo. Identificado o modelo apropriado para a série temporal em causa, este pode ser ajustado usando o histórico de observações e de seguida calculada a previsão.

Toda a metodologia da *framework* pode ser analisada em detalhe no Algoritmo 1.

Atente-se que o algoritmo de agrupamento sugerido funciona de uma forma inovadora. Dado que os erros de previsão resultantes da aplicação de cada modelo favorecem o agrupamento na perspectiva da previsão, estes são incluídos na formação dos grupos. Para tal, é necessária uma avaliação prévia do desempenho dos modelos da *pool*. Por sua vez, este desempenho é também posteriormente utilizado na determinação do modelo do grupo, sendo escolhido o modelo da *pool* que apresenta o menor erro na previsão das séries temporais incluídas no grupo. O uso dos erros de previsão só é possível na fase *offline*, ou seja, na fase *online* são usadas apenas as características das séries temporais.

Pretende-se identificar de uma forma rápida um modelo a aplicar a uma série temporal de modo a obter uma previsão com o menor erro possível. Como referido anteriormente, a escolha do modelo do grupo é feita tendo em conta o desempenho global dos modelos em cada grupo, com base num erro de previsão. Isto significa que a escolha do modelo do grupo não reflete o melhor modelo para cada série temporal desse grupo. Reflete sim o modelo mais apropriado a ajustar a uma nova série cujo comportamento se revele similar ao do grupo de séries temporais ao qual está associada.

A fase *offline* é mais exigente do que a fase *online* em termos computacionais e consumo de tempo, visto que cada um dos modelos da *pool* tem de ser aplicado a cada uma das séries temporais do conjunto de treino. Contudo, uma vez que esta tarefa não é realizada continuamente, o tempo envolvido e custo computacional associado não constituem um impedimento à utilização desta *framework*. Quanto maior for o número de modelos da *pool*, maior será o esforço computacional e o tempo necessário para a realização da fase *offline*. No entanto, os ganhos em termos de exatidão da previsão podem ser significativos, justificando-se uma escolha exaustiva adequada. Como já

referido anteriormente, o cálculo de cada uma das características é extremamente rápido pelo que a fase *online* é rápida em termos de tempo de processamento.

Escolheu-se o método *k-Means* (MacQueen, 1967) para realizar o agrupamento dos dados visto ser o procedimento mais utilizado na literatura, dadas as suas vantagens em termos de eficácia e rapidez de implementação (Aghabozorgi et al., 2015).

Algoritmo 1: *Framework*

Fase *offline*: Treino da *framework*

Dado:

$O = \{x_1, x_2, x_3, ..., x_n\}$: coleção de n séries temporais.

L : *Pool* de m modelos (ARIMA, ES, TBATS, LASSO, ...).

F : Função de cálculo das características.

Saída:

Framework de atribuição de modelos.

Dividir o conjunto de séries temporais de treino em período de treino e período de teste.
Para $i = 1$ até n:
    Usar F para calcular as características usando o período de treino de x_i.
    Para $j = 1$ até m:
        Ajustar o modelo l_j ao período de treino da série temporal x_i.
        Realizar as previsões para o período de teste usando o modelo ajustado.
        Calcular o erro de previsão associado a x_i.
Determinar o número de grupos, k usando as características e os erros de previsão com base em *k-means*.
Realizar o agrupamento em k grupos usando *k-means*.
Determinar o modelo de cada grupo com base nos erros de previsão.

Fase *online*: Previsão

Dado:

x: Nova série temporal.

Saída:

Modelo apropriado a ajustar à série temporal e previsão.

Calcular as características de x.
Calcular a matriz de distâncias entre as características de x e os centroides dos k grupos.
Atribuir x a um dos grupos com base na distância.
Ajustar o modelo do grupo a x.
Realizar a previsão.

Num processo iterativo, este método atribui n amostras a k grupos, previamente definidos, de forma a maximizar a semelhança entre amostras do mesmo grupo, enquanto tenta minimizar as

similaridades entre amostras de grupos diferentes.

Como já referido na Secção 2.2, há várias opções disponíveis para a determinação do número ótimo de grupos, sendo as mais comuns o método *Elbow*, o método *Silhouette* (Rousseeuw, 1987), o método *Gap statistic* (Tibshirani et al., 2001) e o *Dunn index* (Dunn, 1974). Neste trabalho foram utilizados os métodos *Silhouette* e *Gap statistic* para a escolha do número de grupos.

Quer o *k-Means* quer os métodos de determinação do número de grupos carecem do cálculo prévio da similaridade entre as características das séries temporais. Para este cálculo a *framework* utiliza a distância euclidiana (Lkhagva et al., 2006), que constitui uma referência na literatura quando as características das séries são utilizadas (Aghabozorgi et al., 2015).

Um dos aspetos fundamentais de ambas as fases relaciona-se com a distância entre as características das séries temporais. Assumindo que características próximas correspondem a séries temporais que se comportam de uma forma semelhante, é importante que as características das séries temporais usadas na fase *offline* mapeiem o espaço de características das séries temporais que se pretende prever na fase *online*.

3.4 Procedimento Experimental

Para testar a capacidade da *framework* na identificação de métodos apropriados para previsão, foram considerados quatro cenários experimentais que se encontram descritos na Tabela 3.1. No cenário 1 foram utilizados os SKUs da loja 412 (num total de 988), dos quais 788 foram usados para treino e 200 para teste. Da mesma forma, no cenário 2 foram utilizados os SKUs da loja 843 (num total de 717), dos quais 572 foram usados para treino e 145 para teste. No cenário 3 foram utilizados os SKUs da loja 412 para treino (num total de 988) e os SKUs da loja 843 para teste (num total de 717). De igual modo, no cenário 4 foram utilizados os SKUs da loja 843 para treino (num total de 717) e os SKUs da loja 412 para teste (num total de 988). A seleção dos SKUs de treino e teste em cada cenário foi feita de forma aleatória.

Tabela 3.1: Cenários usados no procedimento experimental para validação da *framework*.

	Cenário 1		Cenário 2		Cenário 3		Cenário 4	
	Loja	SKUs	Loja	SKUs	Loja	SKUs	Loja	SKUs
SKUs de treino	412	788	843	572	412	988	843	717
SKUs de teste	412	200	843	145	843	717	412	988

A Figura 3.5 mostra os dois primeiros componentes principais das características das séries temporais de vendas semanais dos SKUs de treino (a cor laranja) e dos SKUs de teste (a cor verde), para cada um dos cenários. Em todos os cenários observa-se que a distribuição das características dos SKUs de treino e teste estende-se por todo o espaço indicando a existência de diversidade de comportamento nas séries temporais incluídas em cada cenário.

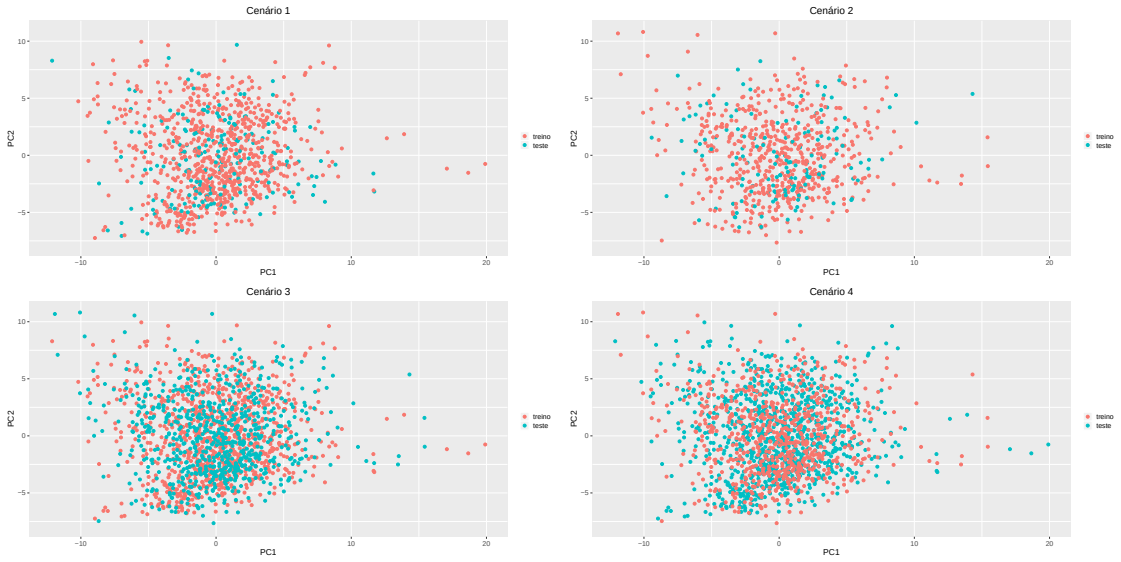

Figura 3.5: Distribuição dos SKUs de treino e teste no espaço das características, nos diferentes cenários.

Para cada cenário foi realizada a fase *offline* usando o conjunto de SKUs de treino. Cada uma das séries de vendas deste conjunto foi dividida em período de treino e período de teste. O período de treino compreendeu as primeiras 160 semanas e o período de teste as últimas 13 semanas.

De seguida, o período de treino de cada série foi utilizado para: (1) o cálculo do valor das respectivas características (num total de 36), e para, (2) o ajuste de todos os modelos da *pool* (num total de 17). Devido à diferença de ordem de grandeza entre as características das séries de vendas semanais dos SKUs das duas lojas, como se pode observar nas Tabelas A.2 e A.3, todas as características dos SKUs de treino foram padronizadas para média 0 e variância 1. Cada um dos modelos ajustado foi posteriormente utilizado para calcular as 13 previsões para o período de teste. Para cada uma das séries, usando as 13 observações do período de teste e as 13 previsões de cada um dos modelos, foram calculados o RelMAE e o MASE para cada um destes. Ambos os erros foram escalados usando o método Naïve sazonal com período de sazonalidade de 52 semanas.

O valor do RelMAE/MASE de cada modelo (num total de 17) e o valor das características (num total de 36) para cada uma das séries foram de seguida utilizados para formar os grupos de SKUs usando o método *k-means*. Como já referido, o método *k-means* exige a especificação prévia do número de grupos. Assim, para cada cenário, foram considerados entre 1 e 10 grupos possíveis e o número ótimo de grupos foi determinado utilizando os métodos *Silhouette* e *Gap statistic*. Na Figura 3.6 mostram-se os resultados desse estudo, baseado no RelMAE. Resultados semelhantes foram obtidos para o MASE. De acordo com o método *Silhouette* devem ser especificados 3 grupos em cada cenário. De acordo com o método *Gap statistic* deve ser especificado 1 grupo nos cenários 2 e 4, 9 grupos no cenário 1, e 8 grupos no cenário 4.

Na Figura 3.7 mostram-se os grupos de SKUs para os cenários 1, 2, 3, e 4, com o número ótimo de grupos determinado utilizando os métodos *Silhouette* e *Gap statistic*, com base no RelMAE. Resultados semelhantes foram obtidos para o MASE.

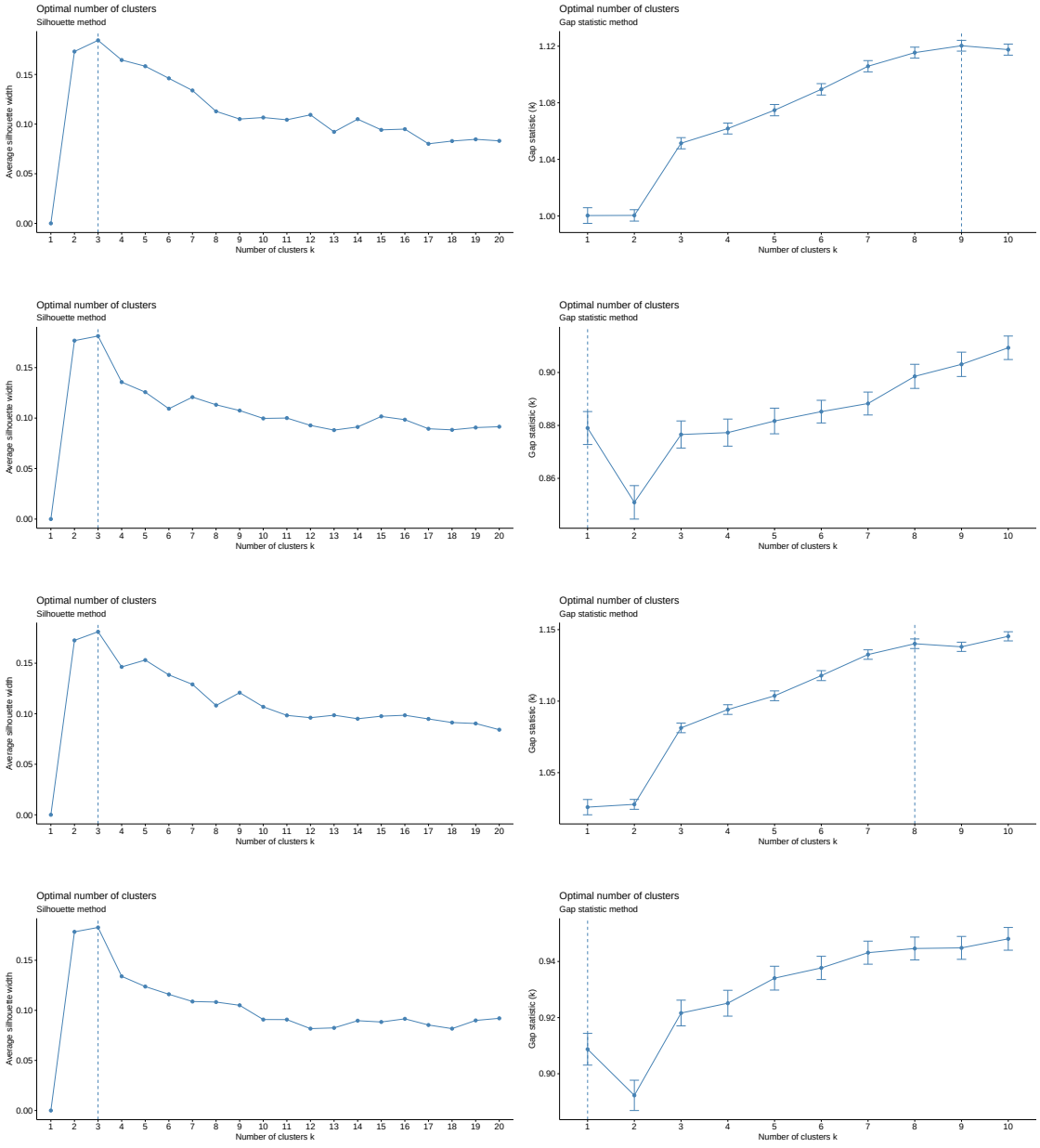

Figura 3.6: Número ótimo de grupos determinado utilizando os métodos *Silhouette* (à esquerda) e *Gap statistic* (à direita), para os cenários 1, 2, 3, e 4 (de cima para baixo), baseado no RelMAE.

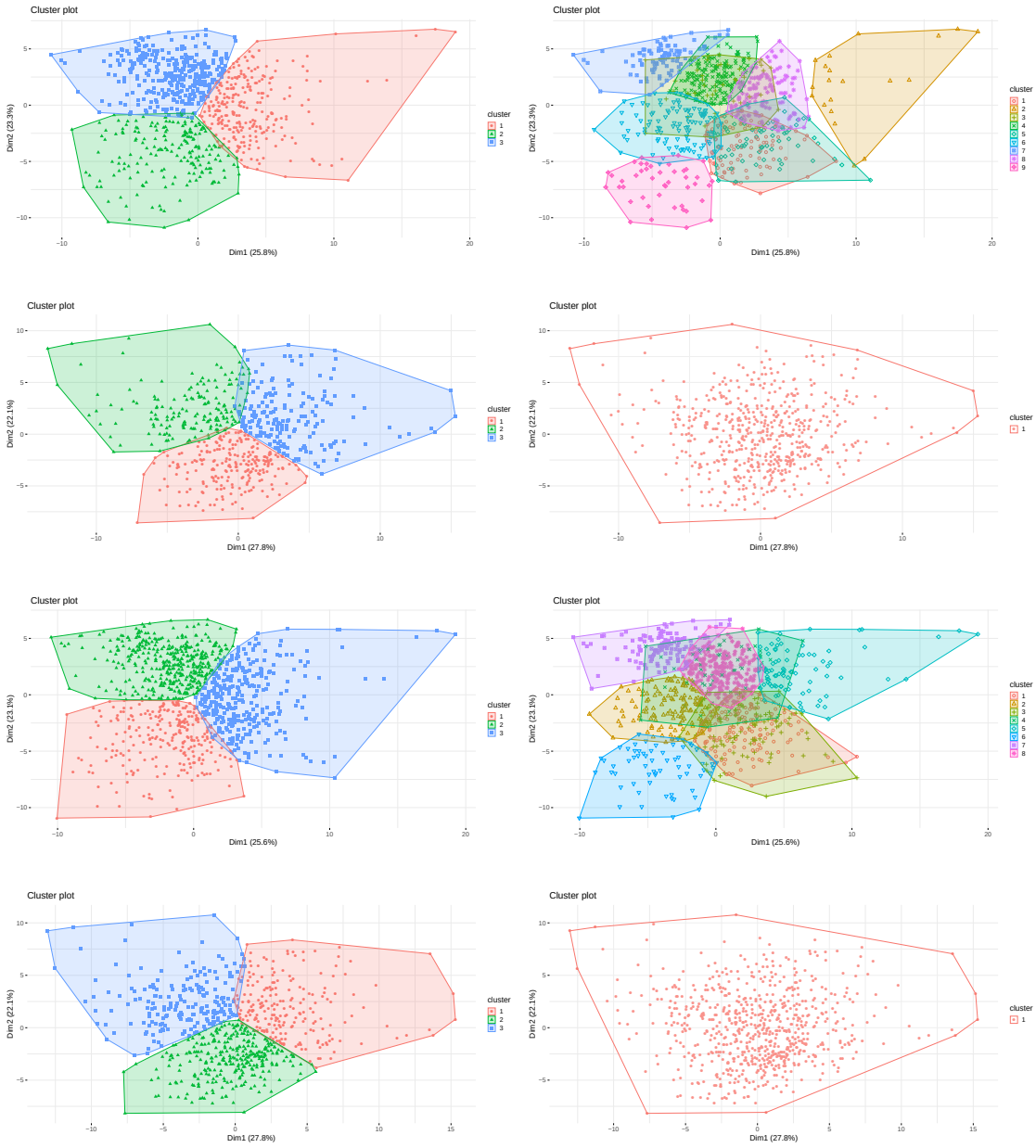

Figura 3.7: Grupos de SKUs para os cenários 1, 2, 3, e 4 (de cima para baixo), com o número ótimo de grupos determinado utilizando os métodos *Silhouette* (à esquerda) e *Gap statistic* (à direita), baseado no RelMAE.

Cada grupo é representado pelo respetivo centroide. Cada característica do centroide é a média dos valores dessa característica para os SKUs incluídos no respetivo grupo.

Especificados os grupos de SKUs, é determinado o modelo de previsão associado a cada grupo. Esta determinação é feita de acordo com o desempenho de cada modelo para o conjunto de séries do grupo, com base no AvgRelMAE/MASE. O AvgRelMAE é obtido fazendo a média geométrica do RelMAE de todas as séries do grupo e o MASE é obtido fazendo a média aritmética do MASE de todas as séries do grupo. Assim, é escolhido para modelo do grupo aquele que apresenta o menor valor do AvgRelMAE/MASE.

A *pool* de modelos candidatos foi constituída de acordo com Oliveira et al. (2020), tendo sido utilizados os respetivos algoritmos implementados no software estatístico *R*, considerando as opções de *default*:

1. ARIMA - função *auto.arima()* do package *forecast*;
2. ARIMA Fourier - função *auto.arima()* do package *forecast*;
3. ES - função *es()* do package *smooth*;
4. TBATS - função *tbats()* do package *forecast*;
5. LASSO-PACF - função *glmnet()* do package *glmnet*;
6. LASSO-PACF Fourier - função *glmnet()* do package *glmnet*;
7. ARIMA + LASSO - funções *auto.arima()* do package *forecast* e *glmnet()* do package *glmnet*;
8. ARIMA Fourier + LASSO - funções *auto.arima()* do package *forecast* e *glmnet()* do package *glmnet*;
9. ES + LASSO - funções *es()* do package *smooth* e *glmnet()* do package *glmnet*;
10. TBATS + LASSO - funções *tbats()* do package *forecast* e *glmnet()* do package *glmnet*;
11. LASSO-PACF + LASSO - função *glmnet()* do package *glmnet*;
12. LASSO-PACF Fourier + LASSO - função *glmnet()* do package *glmnet*;
13. PCR com erros ARIMA - funções *auto.arima()* do package *forecast* e *prcomp()* do package *stats*;
14. PCR Fourier com erros ARIMA - funções *auto.arima()* do package *forecast* e *prcomp()* do package *stats*;
15. PCR com erros ES - funções *es()* do package *smooth* e *prcomp()* do package *stats*;
16. LASSOX-PACF - função *glmnet()* do package *glmnet*;
17. LASSOX-PACF Fourier - função *glmnet()* do package *glmnet*;

Nos modelos em que, para além do histórico de vendas, foram incluídos regressores externos estes integraram informação sobre o preço do SKU, número de dias em promoção na semana, semana de final de mês, feriados e eventos de calendário tais como Natal, Páscoa, Carnaval, entre outros, considerando em alguns casos *leads* e *lags*. Estes modelos constituem o estado da arte no que toca à previsão de vendas nos sectores da distribuição alimentar e retalho especializado. Mais detalhes sobre estes modelos podem ser encontrados em Oliveira et al. (2020).

Como referido, o trabalho foi desenvolvido no software estatístico *R* com recurso à ferramenta *RStudio*. O cálculo do valor das características foi realizado utilizando a função *features()* do package *feasts* e o agrupamento dos dados foi levado a cabo usando a função *kmeans()* do package *stats*. Algum do processamento da fase *offline* foi realizado no serviço GridFEUP disponibilizado pela Faculdade de Engenharia da Universidade do Porto.

Nas Tabelas 3.2 e 3.3 pode observar-se o modelo de previsão atribuído a cada grupo em cada cenário, com base no RelMAE e MASE, com o número de grupos determinado pelo método *Silhouette* e pelo método *Gap statistic*. Observa-se que por vezes os modelos diferem com base em cada um dos erros. Esta situação é bastante frequente, muitas vezes diferentes medidas de erro conduzem a conclusões divergentes sobre qual o melhor método de previsão (Hyndman and Athanasopoulos, 2014). É de salientar a diversidade de modelos existente, o que permite identificar zonas no espaço de características que têm mais "afinidade" com determinados modelos. Também se pode observar que todos os modelos incluem regressores o que revela a importância desta informação para a previsão dos dados em causa.

Tabela 3.2: Modelo de previsão atribuído a cada grupo em cada cenário, com base no RelMAE e MASE. Número de grupos determinado pelo método *Silhouette*.

Cenário 1		
	RelMAE	MASE
G1	PCR com erros ES	PCR Fourier com erros ARIMA
G2	PCR com erros ES	ES + LASSO
G3	ES + LASSO	ES + LASSO
Cenário 2		
	RelMAE	MASE
G1	ARIMA Fourier + LASSO	ARIMA Fourier + LASSO
G2	PCR Fourier com erros ARIMA	ES + LASSO
G3	ES + LASSO	ES + LASSO
Cenário 3		
	RelMAE	MASE
G1	ARIMA Fourier + LASSO	ARIMA Fourier + LASSO
G2	PCR com erros ES	ES + LASSO
G3	LASSOX-PACF	PCR Fourier com erros ARIMA
Cenário 4		
	RelMAE	MASE
G1	LASSOX-PACF	ARIMA Fourier + LASSO
G2	PCR com erros ES	PCR com erros ES
G3	ARIMA Fourier + LASSO	PCR Fourier com erros ARIMA

Tabela 3.3: Modelo de previsão atribuído a cada grupo em cada cenário, com base no RelMAE e MASE. Número de grupos determinado pelo método *Gap statistic*.

	Cenário 1	
	RelMAE	MASE
G1	ES + LASSO	ES + LASSO
G2	ARIMA Fourier + LASSO	PCR Fourier com erros ARIMA
G3	ES + LASSO	LASSOX-PACF
G4	PCR com erros ES	ARIMAX
G5	LASSOX-PACF	ES + LASSO
G6	PCR Fourier com erros ARIMA	PCR com erros ES
G7	ES + LASSO	PCR com erros ES
G8	PCR com erros ES	ARIMA Fourier + LASSO
G9	ES + LASSO	PCR com erros ES
	Cenário 2	
	RelMAE	MASE
G1	ARIMA Fourier + LASSO	ARIMA Fourier + LASSO
	Cenário 3	
	RelMAE	MASE
G1	PCR com erros ES	PCR Fourier com erros ARIMA
G2	ARIMA Fourier + LASSO	PCR Fourier com erros ARIMA
G3	ARIMA Fourier + LASSO	PCR Fourier com erros ARIMA
G4	PCR Fourier com erros ARIMA	PCR com erros ES
G5	LASSOX-PACF	ES + LASSO
G6	PCR com erros ES	PCR com erros ES
G7	PCR com erros ES	ARIMA Fourier + LASSO
G8	PCR com erros ES	ARIMA Fourier + LASSO
	Cenário 4	
	RelMAE	MASE
G1	PCR com erros ES	PCR com erros ES

3.5 Resultados

O desempenho de previsão da *framework* em cada cenário foi avaliado utilizando o respetivo conjunto de SKUs de teste, "imitando" o procedimento da fase *online* da mesma (ver Figura 3.8).

Cada uma das séries de vendas deste conjunto foi dividida em período de treino e período de teste. O período de treino compreendeu as primeiras 160 semanas e o período de teste as últimas 13 semanas. De seguida, o período de treino de cada série foi utilizado para o cálculo do valor das respectivas características (num total de 36), que foram de seguida padronizadas para média 0 e variância 1. Depois de calculadas as distâncias entre as características de cada série temporal e

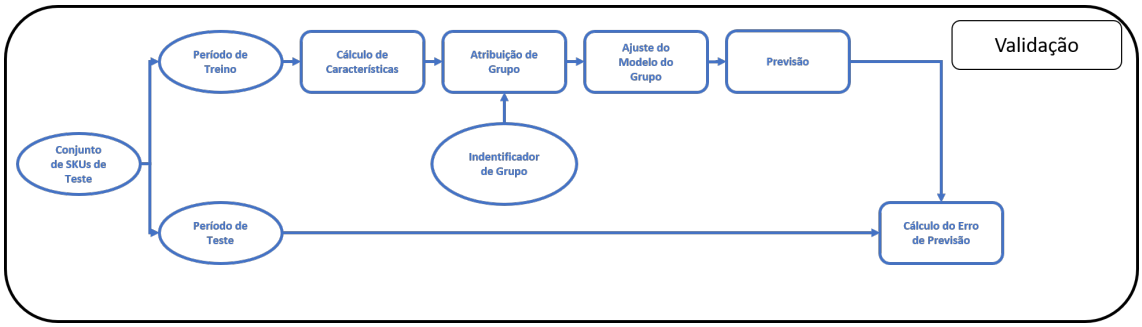

Figura 3.8: Representação gráfica da validação da *framework* proposta.

as características de cada um dos centroides, utilizando o Identificador de Grupo treinado na fase *offline*, foi atribuído a cada série o grupo cujo centroide lhe está mais próximo.

Identificado o grupo de cada série, foi-lhe atribuído o modelo de previsão do respetivo grupo, tendo este sido ajustado, usando naturalmente o período de treino da mesma. Utilizando o modelo de previsão ajustado foram calculadas as 13 previsões para o período de teste. Para cada uma das séries, usando as 13 observações do período de teste e as 13 previsões do respetivo modelo, foram calculados o RelMAE e o MASE. Ambos os erros foram escalados usando o método Naïve sazonal com período de sazonalidade de 52 semanas.

Finalmente, foi obtido o AvgRelMAE fazendo a média geométrica do RelMAE de todas as séries do conjunto de teste. De modo idêntico, obteve-se o MASE fazendo a média aritmética do MASE de todas as séries do conjunto de teste. Este procedimento foi repetido em cada um dos cenários considerando o número ótimo de grupos determinado utilizando os métodos *Silhouette* e *Gap statistic*.

A exatidão de previsão da *framework* desenvolvida foi comparada com a exatidão de previsão de todos os modelos da *pool*. Assim, para cada cenário, usando o conjunto de SKUs de teste respetivo, foi repetido para cada um dos 17 modelos o procedimento de validação levado a cabo para a *framework*.

Assim, cada uma das séries de vendas deste conjunto foi dividida em período de treino e período de teste. O período de treino compreendeu as primeiras 160 semanas e o período de teste as últimas 13 semanas. Cada modelo foi ajustado ao período de treino de cada uma das séries e de seguida foram calculadas as 13 previsões para o período de teste. Para cada série, usando as 13 observações do período de teste e as 13 previsões de cada um dos modelos, foram calculados o RelMAE e o MASE. Ambos os erros foram escalados usando o método Naïve sazonal com período de sazonalidade de 52 semanas. Finalmente, para cada um dos modelos, foi obtido o AvgRelMAE fazendo a média geométrica do RelMAE respetivo relativo a todas as séries do conjunto de teste. O respetivo MASE foi obtido de modo idêntico.

As Tabelas 3.4, 3.5, 3.6 e 3.7 apresentam os resultados obtidos. O *rank* é calculado fazendo a média da ordenação das colunas do AvgRelMAE e do MASE.

Pode dizer-se que de um modo geral a *framework* tem um desempenho razoável nos quatro cenários experimentais. Contudo, é no cenário 3 que se apresenta inequivocamente como o melhor

método de previsão com base nas duas medidas de erro, quer o número ótimo de grupos seja determinado pelo método *Silhouette* ou pelo método *Gap statistic*. Recorde-se que no cenário 3 são utilizados todos os SKUs da loja 412 (num total de 988) para treino da *framework* e os SKUs da loja 843 para teste (num total de 717). Como era espectável, este corresponde ao melhor cenário para treino da *framework* visto que o conjunto de SKUs de treino apresenta a maior diversidade de características possível.

No cenário experimental 4, a *framework* também se apresenta como o melhor método de previsão, à excepção do caso em que o número ótimo de grupos é determinado pelo método *Silhouette*, tendo por base o RelMAE. Este cenário já não é tão favorável, visto que, apesar do conjunto de treino incluir todos os SKUs da loja 843, esta apresenta uma diversidade de características inferior à da loja 412.

Nos cenários experimentais 1 e 2 o desempenho da *framework* é manifestamente pior, como esperado. Nestes dois cenários, parte dos SKUs da loja são utilizados para teste, diminuindo-se assim a diversidade de características do conjunto de treino. Esta situação é mais notória no cenário 2 uma vez que o número de SKUs de treino (572) é inferior ao número de SKUs de treino do cenário 1 (788) (em ambos os casos o conjunto de teste compreende cerca de 20% dos SKUs da loja), o que explica a superioridade do desempenho da *framework* no cenário 1 comparativamente com o cenário 2.

Atente-se que o número ótimo de grupos determinado quer pelo método *Silhouette* quer pelo método *Gap statistic* é independente da medida de erro considerada, o que revela a robustez do desempenho destes métodos na metodologia desenvolvida.

Estes resultados mostram ainda que, num cenário ideal de treino da *framework*, o método *Gap statistic* tem tendência para gerar mais grupos do que o método *Silhouette*. Ou seja, quando a diversidade de características do conjunto de treino é maior o método *Gap statistic* tem tendência para apresentar um maior número de grupos. Esta característica, intrínseca ao procedimento do método, é bastante favorável para o relacionamento das características das séries temporais com o modelo mais apropriado.

Poder-se-ia argumentar que o elevado desempenho ao nível de previsão da *framework* é devido, no caso do número ótimo de grupos ser determinado pelo método *Silhouette*, ao baixo número de grupos existente. Contudo, este argumento é falso, visto que no caso do número ótimo de grupos ser determinado pelo método *Gap statistic*, existe efetivamente um elevado número de grupos.

Seria de esperar que quando a *framework* é treinada com base numa medida de erro e testada com base numa medida de erro diferente, o seu desempenho seria pior do que se fosse treinada e testada com base na mesma medida de erro. Contudo, os resultados obtidos mostram que isso não acontece o que evidencia a sua robustez.

Este trabalho permite concluir que de facto as características de uma série temporal podem proporcionar informação útil para a determinação dos métodos mais apropriados para previsão, corroborando os resultados de Hyndman (2001), Lawrence (2001) e Armstrong (2001).

Tabela 3.4: Resultados de previsão da *framework* e dos modelos da *pool* para o cenário 1.

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
ES + LASSO	0,5524	0,658	1	ES + LASSO	0,5524	0,658	1
Clustering	0,5529	0,6608	2	PCR com erros ES	0,5533	0,661	2
PCR com erros ES	0,5533	0,661	3	PCR Fourier com erros ARIMA	0,5568	0,6636	3
PCR Fourier com erros ARIMA	0,5568	0,6636	4	Clustering	0,558	0,6651	4
ARIMA Fourier + LASSO	0,5613	0,67	5	ARIMA Fourier + LASSO	0,5613	0,67	5
TBATS + LASSO	0,5679	0,6739	6	TBATS + LASSO	0,5679	0,6739	6
LASSOX-PACF	0,5694	0,6813	7	LASSOX-PACF	0,5694	0,6813	7
PCR com erros ARIMA	0,5711	0,6816	8	PCR com erros ARIMA	0,5711	0,6816	8
LASSOX-PACF Fourier	0,577	0,6877	9	LASSOX-PACF Fourier	0,577	0,6877	9
LASSO-PACF Fourier + LASSO	0,5823	0,6924	10	LASSO-PACF Fourier + LASSO	0,5823	0,6924	10
ARIMA + LASSO	0,5826	0,6963	11	ARIMA + LASSO	0,5826	0,6963	11
LASSO-PACF + LASSO	0,5935	0,7057	12	LASSO-PACF + LASSO	0,5935	0,7057	12
ES	0,6452	0,7921	13	ES	0,6452	0,7921	13
ARIMA Fourier	0,6494	0,7947	14	ARIMA Fourier	0,6494	0,7947	14
TBATS	0,6576	0,8	15	TBATS	0,6576	0,8	15
ARIMA	0,6679	0,8127	16,5	ARIMA	0,6679	0,8127	16,5
LASSO-PACF Fourier	0,6766	0,8126	17	LASSO-PACF Fourier	0,6766	0,8126	17
LASSO-PACF	0,6745	0,8143	17,5	LASSO-PACF	0,6745	0,8143	17,5

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
ES + LASSO	0,5524	0,658	1	ES + LASSO	0,5524	0,658	1
PCR com erros ES	0,5533	0,661	2	Clustering	0,5541	0,6608	2,5
Clustering	0,5534	0,6626	3	PCR com erros ES	0,5533	0,661	2,5
PCR Fourier com erros ARIMA	0,5568	0,6636	4	PCR Fourier com erros ARIMA	0,5568	0,6636	4
ARIMA Fourier + LASSO	0,5613	0,67	5	ARIMA Fourier + LASSO	0,5613	0,67	5
TBATS + LASSO	0,5679	0,6739	6	TBATS + LASSO	0,5679	0,6739	6
LASSOX-PACF	0,5694	0,6813	7	LASSOX-PACF	0,5694	0,6813	7
PCR com erros ARIMA	0,5711	0,6816	8	PCR com erros ARIMA	0,5711	0,6816	8
LASSOX-PACF Fourier	0,577	0,6877	9	LASSOX-PACF Fourier	0,577	0,6877	9
LASSO-PACF Fourier + LASSO	0,5823	0,6924	10	LASSO-PACF Fourier + LASSO	0,5823	0,6924	10
ARIMA + LASSO	0,5826	0,6963	11	ARIMA + LASSO	0,5826	0,6963	11
LASSO-PACF + LASSO	0,5935	0,7057	12	LASSO-PACF + LASSO	0,5935	0,7057	12
ES	0,6452	0,7921	13	ES	0,6452	0,7921	13
ARIMA Fourier	0,6494	0,7947	14	ARIMA Fourier	0,6494	0,7947	14
TBATS	0,6576	0,8	15	TBATS	0,6576	0,8	15
ARIMA	0,6679	0,8127	16,5	ARIMA	0,6679	0,8127	16,5
LASSO-PACF Fourier	0,6766	0,8126	17	LASSO-PACF Fourier	0,6766	0,8126	17
LASSO-PACF	0,6745	0,8143	17,5	LASSO-PACF	0,6745	0,8143	17,5

Tabela 3.5: Resultados de previsão da *framework* e dos modelos da *pool* para o cenário 2.

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
ES + LASSO	0,584	0,6795	1	ES + LASSO	0,584	0,6795	1
PCR com erros ES	0,5895	0,6844	2	PCR com erros ES	0,5895	0,6844	2
TBATS + LASSO	0,5904	0,6856	3,5	Clustering	0,5897	0,6874	4
ARIMA Fourier + LASSO	0,5897	0,6874	3,5	TBATS + LASSO	0,5904	0,6856	4
PCR Fourier com erros ARIMA	0,5931	0,6874	5,5	ARIMA Fourier + LASSO	0,5897	0,6874	4
ARIMA + LASSO	0,5908	0,6903	6	PCR Fourier com erros ARIMA	0,5931	0,6874	6,5
LASSOX-PACF	0,5943	0,6875	6,5	ARIMA + LASSO	0,5908	0,6903	7
Clustering	0,6019	0,6958	8	LASSOX-PACF	0,5943	0,6875	7,5
LASSO-PACF + LASSO	0,61	0,7093	9,5	LASSO-PACF + LASSO	0,61	0,7093	9,5
LASSO-PACF Fourier + LASSO	0,6106	0,7121	10,5	LASSO-PACF Fourier + LASSO	0,6106	0,7121	10,5
PCR com erros ARIMA	0,6091	0,7148	10,5	PCR com erros ARIMA	0,6091	0,7148	10,5
LASSOX-PACF Fourier	0,6137	0,7135	11,5	LASSOX-PACF Fourier	0,6137	0,7135	11,5
ARIMA Fourier	0,6481	0,7768	13	ARIMA Fourier	0,6481	0,7768	13
ES	0,6483	0,7791	14	ES	0,6483	0,7791	14
ARIMA	0,649	0,7811	15	ARIMA	0,649	0,7811	15
TBATS	0,6548	0,7815	16	TBATS	0,6548	0,7815	16
LASSO-PACF	0,6648	0,7928	17	LASSO-PACF	0,6648	0,7928	17
LASSO-PACF Fourier	0,6727	0,7978	18	LASSO-PACF Fourier	0,6727	0,7978	18

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
ES + LASSO	0,584	0,6795	1	ES + LASSO	0,584	0,6795	1
PCR com erros ES	0,5895	0,6844	2	PCR com erros ES	0,5895	0,6844	2
TBATS + LASSO	0,5904	0,6856	3,5	Clustering	0,5897	0,6874	4
ARIMA Fourier + LASSO	0,5897	0,6874	3,5	TBATS + LASSO	0,5904	0,6856	4
ARIMA + LASSO	0,5908	0,6903	6	ARIMA Fourier + LASSO	0,5897	0,6874	4
PCR Fourier com erros ARIMA	0,5931	0,6874	6	PCR Fourier com erros ARIMA	0,5931	0,6874	6,5
Clustering	0,5919	0,6905	7	ARIMA + LASSO	0,5908	0,6903	7
LASSOX-PACF	0,5943	0,6875	7	LASSOX-PACF	0,5943	0,6875	7,5
LASSO-PACF + LASSO	0,61	0,7093	9,5	LASSO-PACF + LASSO	0,61	0,7093	9,5
LASSO-PACF Fourier + LASSO	0,6106	0,7121	10,5	LASSO-PACF Fourier + LASSO	0,6106	0,7121	10,5
PCR com erros ARIMA	0,6091	0,7148	10,5	PCR com erros ARIMA	0,6091	0,7148	10,5
LASSOX-PACF Fourier	0,6137	0,7135	11,5	LASSOX-PACF Fourier	0,6137	0,7135	11,5
ARIMA Fourier	0,6481	0,7768	13	ARIMA Fourier	0,6481	0,7768	13
ES	0,6483	0,7791	14	ES	0,6483	0,7791	14
ARIMA	0,649	0,7811	15	ARIMA	0,649	0,7811	15
TBATS	0,6548	0,7815	16	TBATS	0,6548	0,7815	16
LASSO-PACF	0,6648	0,7928	17	LASSO-PACF	0,6648	0,7928	17
LASSO-PACF Fourier	0,6727	0,7978	18	LASSO-PACF Fourier	0,6727	0,7978	18

Tabela 3.6: Resultados de previsão da *framework* e dos modelos da *pool* para o cenário 3.

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
Clustering	0,5662	0,6533	1	Clustering	0,5656	0,654	1,5
PCR com erros ES	0,5667	0,6535	2	PCR com erros ES	0,5667	0,6535	1,5
PCR Fourier com erros ARIMA	0,5692	0,6541	3	PCR Fourier com erros ARIMA	0,5692	0,6541	3
ES + LASSO	0,5696	0,659	4,5	ES + LASSO	0,5696	0,659	4,5
ARIMA Fourier + LASSO	0,5705	0,6577	4,5	ARIMA Fourier + LASSO	0,5705	0,6577	4,5
TBATS + LASSO	0,5744	0,6607	6	TBATS + LASSO	0,5744	0,6607	6
LASSOX-PACF	0,5804	0,6683	7	LASSOX-PACF	0,5804	0,6683	7
PCR com erros ARIMA	0,5811	0,6716	8	PCR com erros ARIMA	0,5811	0,6716	8
ARIMA + LASSO	0,5811	0,6731	9	ARIMA + LASSO	0,5811	0,6731	9
LASSOX-PACF Fourier	0,5977	0,6897	10	LASSOX-PACF Fourier	0,5977	0,6897	10
LASSO-PACF + LASSO	0,601	0,6933	11	LASSO-PACF + LASSO	0,601	0,6933	11
LASSO-PACF Fourier + LASSO	0,6042	0,6958	12	LASSO-PACF Fourier + LASSO	0,6042	0,6958	12
ARIMA Fourier	0,6493	0,7724	13	ARIMA Fourier	0,6493	0,7724	13
ES	0,6499	0,7774	14	ES	0,6499	0,7774	14
ARIMA	0,6585	0,7852	15,5	ARIMA	0,6585	0,7852	15,5
TBATS	0,6598	0,7827	15,5	TBATS	0,6598	0,7827	15,5
LASSO-PACF	0,6777	0,7997	17	LASSO-PACF	0,6777	0,7997	17
LASSO-PACF Fourier	0,6867	0,8077	18	LASSO-PACF Fourier	0,6867	0,8077	18

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
Clustering	0,5654	0,6512	1	Clustering	0,5617	0,6482	1
PCR com erros ES	0,5667	0,6535	2	PCR com erros ES	0,5667	0,6535	2
PCR Fourier com erros ARIMA	0,5692	0,6541	3	PCR Fourier com erros ARIMA	0,5692	0,6541	3
ES + LASSO	0,5696	0,659	4,5	ES + LASSO	0,5696	0,659	4,5
ARIMA Fourier + LASSO	0,5705	0,6577	4,5	ARIMA Fourier + LASSO	0,5705	0,6577	4,5
TBATS + LASSO	0,5744	0,6607	6	TBATS + LASSO	0,5744	0,6607	6
LASSOX-PACF	0,5804	0,6683	7	LASSOX-PACF	0,5804	0,6683	7
PCR com erros ARIMA	0,5811	0,6716	8	PCR com erros ARIMA	0,5811	0,6716	8
ARIMA + LASSO	0,5811	0,6731	9	ARIMA + LASSO	0,5811	0,6731	9
LASSOX-PACF Fourier	0,5977	0,6897	10	LASSOX-PACF Fourier	0,5977	0,6897	10
LASSO-PACF + LASSO	0,601	0,6933	11	LASSO-PACF + LASSO	0,601	0,6933	11
LASSO-PACF Fourier + LASSO	0,6042	0,6958	12	LASSO-PACF Fourier + LASSO	0,6042	0,6958	12
ARIMA Fourier	0,6493	0,7724	13	ARIMA Fourier	0,6493	0,7724	13
ES	0,6499	0,7774	14	ES	0,6499	0,7774	14
ARIMA	0,6585	0,7852	15,5	ARIMA	0,6585	0,7852	15,5
TBATS	0,6598	0,7827	15,5	TBATS	0,6598	0,7827	15,5
LASSO-PACF	0,6777	0,7997	17	LASSO-PACF	0,6777	0,7997	17
LASSO-PACF Fourier	0,6867	0,8077	18	LASSO-PACF Fourier	0,6867	0,8077	18

Tabela 3.7: Resultados de previsão da *framework* e dos modelos da *pool* para o cenário 4.

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
PCR com erros ES	0,5664	0,651	1	Clustering	0,5664	0,651	1,5
PCR Fourier com erros ARIMA	0,5675	0,6516	2	PCR com erros ES	0,5664	0,651	1,5
ES + LASSO	0,5679	0,6546	3	PCR Fourier com erros ARIMA	0,5675	0,6516	3
Clustering	0,5683	0,6547	4	ES + LASSO	0,5679	0,6546	4
ARIMA Fourier + LASSO	0,5742	0,6613	5	ARIMA Fourier + LASSO	0,5742	0,6613	5
TBATS + LASSO	0,5778	0,6649	6	TBATS + LASSO	0,5778	0,6649	6
LASSOX-PACF	0,5812	0,6696	7,5	LASSOX-PACF	0,5812	0,6696	7,5
PCR com erros ARIMA	0,5803	0,6771	8	PCR com erros ARIMA	0,5803	0,6771	8
ARIMA + LASSO	0,5829	0,6729	8,5	ARIMA + LASSO	0,5829	0,6729	8,5
LASSOX-PACF Fourier	0,5911	0,6804	10	LASSOX-PACF Fourier	0,5911	0,6804	10
LASSO-PACF Fourier + LASSO	0,5937	0,6824	11	LASSO-PACF Fourier + LASSO	0,5937	0,6824	11
LASSO-PACF + LASSO	0,6008	0,691	12	LASSO-PACF + LASSO	0,6008	0,691	12
ARIMA Fourier	0,6613	0,787	13,5	ARIMA Fourier	0,6613	0,787	13,5
ES	0,6592	0,788	13,5	ES	0,6592	0,788	13,5
ARIMA	0,6646	0,7918	15	ARIMA	0,6646	0,7918	15
TBATS	0,6666	0,7924	16	TBATS	0,6666	0,7924	16
LASSO-PACF	0,6827	0,8045	17,5	LASSO-PACF	0,6827	0,8045	17,5
LASSO-PACF Fourier	0,686	0,8043	17,5	LASSO-PACF Fourier	0,686	0,8043	17,5

Silhouette	AvgRelMAE	MASE	Rank	**GAP**	AvgRelMAE	MASE	Rank
Clustering	0,5643	0,6507	1	Clustering	0,5664	0,651	1,5
PCR com erros ES	0,5664	0,651	2	PCR com erros ES	0,5664	0,651	1,5
PCR Fourier com erros ARIMA	0,5675	0,6516	3	PCR Fourier com erros ARIMA	0,5675	0,6516	3
ES + LASSO	0,5679	0,6546	4	ES + LASSO	0,5679	0,6546	4
ARIMA Fourier + LASSO	0,5742	0,6613	5	ARIMA Fourier + LASSO	0,5742	0,6613	5
TBATS + LASSO	0,5778	0,6649	6	TBATS + LASSO	0,5778	0,6649	6
LASSOX-PACF	0,5812	0,6696	7,5	LASSOX-PACF	0,5812	0,6696	7,5
PCR com erros ARIMA	0,5803	0,6771	8	PCR com erros ARIMA	0,5803	0,6771	8
ARIMA + LASSO	0,5829	0,6729	8,5	ARIMA + LASSO	0,5829	0,6729	8,5
LASSOX-PACF Fourier	0,5911	0,6804	10	LASSOX-PACF Fourier	0,5911	0,6804	10
LASSO-PACF Fourier + LASSO	0,5937	0,6824	11	LASSO-PACF Fourier + LASSO	0,5937	0,6824	11
LASSO-PACF + LASSO	0,6008	0,691	12	LASSO-PACF + LASSO	0,6008	0,691	12
ARIMA Fourier	0,6613	0,787	13,5	ARIMA Fourier	0,6613	0,787	13,5
ES	0,6592	0,788	13,5	ES	0,6592	0,788	13,5
ARIMA	0,6646	0,7918	15	ARIMA	0,6646	0,7918	15
TBATS	0,6666	0,7924	16	TBATS	0,6666	0,7924	16
LASSO-PACF	0,6827	0,8045	17,5	LASSO-PACF	0,6827	0,8045	17,5
LASSO-PACF Fourier	0,686	0,8043	17,5	LASSO-PACF Fourier	0,686	0,8043	17,5

Capítulo 4

Conclusões e Trabalho Futuro

4.1 Conclusões

Uma previsão de vendas eficaz garante vantagens competitivas e confere facilidades na gestão logística das empresas. Em particular no setor do retalho, caracterizado por elevados custos relativos a transporte e armazenamento de *stock*, é fundamental assegurar um controlo rígido de forma a evitar incorrer em custos desnecessários ou *stockouts*. O estudo de metodologias de agrupamentos de dados e métodos de previsão foi fundamental para a realização desta dissertação e sustentou o desenvolvimento da *framework* sugerida.

Numa fase prévia, o estudo dos dados disponibilizados pelo Grupo Jerónimo Martins conferiu conhecimento essencial ao desenvolvimento. Foi possível comprovar uma possibilidade de generalização de comportamento de SKUs entre lojas, princípio em que se baseia a *framework*. Também é notória a semelhança de comportamento entre SKUs diferentes na mesma loja e em lojas diferentes. Este facto garante robustez à *framework* e estende a sua utilidade.

A *framework* proposta gera previsões de séries temporais através de métodos de previsão desenvolvidos pela literatura. A inovação conferida reside na escolha do método a usar, escolha esta realizada com base no desempenho que cada método revela quando aplicado aos SKUs do grupo em que cada série temporal a prever se insere. A avaliação da *framework* foi realizada sob a forma de comparação entre as previsões resultantes da sua aplicação e o desempenho resultante da aplicação dos métodos de previsão disponíveis na literatura. A comparação foi realizada com base em medidas de erro plausíveis de comparações entre séries temporais de escalas diferentes.

Todos os estudos e previsões foram realizados sob dados de periodicidade semanal, não só para estar em conformidade com as necessidades do setor, como também com as necessidades do Grupo. Devido à possibilidade de generalização confirmada nos estudos prévios e referida anteriormente, recorreu-se a dados relativos a duas lojas para a implementação da metodologia, a loja 412 e a loja 843.

A implementação foi realizada tendo em consideração quatro cenários, dois incidiram em conjunto de treino e teste pertencentes a lojas diferentes e dois incidiram em conjuntos de treino

e teste pertencentes à mesma loja. A implementação revelou o contributo de uma fase de treino caracterizada por um conjunto de dados bem distribuído ao longo do espaço de características.

A *framework* aliada à realização desta dissertação revela melhorias relativas aos métodos de previsão disponíveis na literatura, resultando em previsões mais exatas no conjunto de dados em que foi aplicada. Conclui-se então que a *framework* proposta confere vantagens significativas na previsão de séries temporais não intermitentes, características do setor do retalho, atingindo assim o objetivo desta dissertação.

4.2 Trabalho Futuro

A *framework* proposta apresenta ainda algumas possibilidades de melhoria. Embora tenha resultado numa *framework* de previsão de vendas eficiente, caracterizada por resultados bastante satisfatórios, revela possibilidades de melhoria relacionadas com os cenários em que apresenta piores resultados.

Uma melhoria na capacidade de generalização da *framework* apresenta vantagens na sugestão do método de previsão indicado. Alterações à fase de treino para assegurar que as séries temporais se distribuem de uma forma mais uniforme no espaço de características são propícias a conferir melhorias significativas na aplicação a conjuntos de dados menos completos.

Também uma sugestão combinada de métodos de previsão poderia trazer vantagens aliadas ao desempenho da *framework*, embora sob a possibilidade de exigir mais tempo de processamento e, consequentemente, invalidar a sua utilidade.

Finalmente, uma comparação entre esta abordagem e uma abordagem supervisionada traria grande utilidade. Por um lado, incluir os erros na definição dos grupos, característica que confere flexibilidade e bons resultados à *framework*, é uma abordagem realizada mais facilmente com recurso a *clustering*, porém, seria interessante atacar o problema de outro ângulo para comparar os resultados. Seria ainda vantajosa a inclusão dos erros da *pool* de modelos no treino do modelo supervisionado, problema que carece de bastante tempo para que seja orquestrada uma solução.

Todas as sugestões carecem de implementação e estudos centrados na avaliação de desempenho, no entanto, são consideradas vantajosas, não só por oferecerem um termo de comparação, como também por se considerar que são plausíveis de gerar bons resultados.

Bibliografia

Aghabozorgi, S., Shirkhorshidi, A. S., and Wah, T. (2015). Time-series clustering - a decade review. *Information Systems*, 53.

Armstrong, J. S. (2001). Should we redesign forecasting competitions? *International Journal of Forecasting*, 17:542–543.

Bates, J. M. and Granger, C. W. J. (1969). The combination of forecasts. *OR*, 20(4):451–468.

Billah, B., King, M., Snyder, R., and Koehler, A. (2006). Exponential smoothing model selection for forecasting. *International Journal of Forecasting*, 22:239–247.

Box, G. E. P. and Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological*, pages 211–252.

Box, G. E. P. and Jenkins, G. M. (1994). *Time Series Analysis: Forecasting and Control*. Prentice Hall PTR, USA, 3rd edition.

Canova, F. and Hansen, B. E. (1995). Are seasonal patterns constant over time? a test for seasonal stability. *Journal of Business & Economic Statistics*, 13(3):237–252.

Chan-Lau, J. (2017). Lasso regressions and forecasting models in applied stress testing. *IMF Working Papers*, 17:1.

Clemen, R. T. (1989). Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583.

Cleveland, R., Cleveland, W. S., McRae, J. E., and Terpenning, I. J. (1990). Stl: A seasonal-trend decomposition procedure based on loess.

Dai, W., Chuang, Y.-Y., and Lu, C.-J. (2015). A clustering-based sales forecasting scheme using support vector regression for computer server. *Procedia Manufacturing*, 2:82 – 86. 2nd International Materials, Industrial, and Manufacturing Engineering Conference, MIMEC2015, 4-6 February 2015, Bali, Indonesia.

De Gooijer, J. G. and Hyndman, R. (2005). 25 years of iif time series forecasting: A selective review. *SSRN Electronic Journal*.

Dunn, J. C. (1974). Well-separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*, 4(1):95–104.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica*, 50(4):987–1007.

Friedman, J. H. (1984). A variable span smoother.

Fulcher, B. and Jones, N. (2014). Highly comparative feature-based time-series classification. *IEEE Transactions on Knowledge and Data Engineering*, 26.

Goerg, G. M. (2013). Forecastable component analysis. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page II–64–II–72. JMLR.org.

Grossmann, W. and Rinderle-Ma, S. (2015). *Data Mining for Temporal Data*, pages 207–244. Springer Berlin Heidelberg, Berlin, Heidelberg.

Guerrero, V. M. (1993). Time-series analysis supported by power transformations. *Journal of Forecasting*, 12(1):37–48.

Haslett, J. and Raftery, A. E. (1989). Space-time modelling with long-memory dependence: Assessing ireland's wind power resource. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 38(1):1–50.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.

Huang, T., Fildes, R., and Soopramanien, D. (2014). The value of competitive information in forecasting fmcg retail product sales and the variable selection problem. *European Journal of Operational Research*, 237(2):738 – 748.

Hugos, M. H. (2018). *Essentials of Supply Chain Management, 4th Edition*. Wiley.

Hyndman, R. (2001). It's time to move from 'what' to 'why'. *International Journal of Forecasting*, 17:567–570.

Hyndman, R. and Athanasopoulos, G. (2014). *Forecasting: principles and practice*. OTexts.

Hyndman, R., Koehler, A., Ord, K., and Snyder, R. (2008). *Forecasting with exponential smoothing. The state space approach*.

Hyndman, R. J. and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679 – 688.

Hyndman, R. J., Wang, E., and Laptev, N. (2015). Large-scale unusual time series detection. In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1616–1619.

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651 – 666. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR).

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2014). *An Introduction to Statistical Learning: With Applications in R*. Springer Publishing Company, Incorporated.

Kassambara, A. (2017). *Practical Guide to Cluster Analysis in R: Unsupervised Machine Learning*, volume 1. STHDA.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley.

Kaufman, L. and Rousseeuw, P. J. (2008). *Partitioning Around Medoids (Program PAM)*, pages 68–125. John Wiley Sons, Inc.

Kumar, R. P. and Nagabhushan, P. (2006). Time series as a point - a novel approach for time series cluster visualization. In *DMIN*.

Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., and Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root : How sure are we that economic time series have a unit root? *Journal of Econometrics*, 54(1-3):159–178.

Lawrence, M. (2001). Commentaries on the m3-competition. why another study? *International Journal of Forecasting*, 17:574–575.

Li, J. and Chen, W. (2014). Forecasting macroeconomic time series: Lasso-based approaches and their forecast combinations with dynamic factor models. *International Journal of Forecasting*, 30(4):996 – 1015.

Livera, A. M. D., Hyndman, R. J., and Snyder, R. D. (2011). Forecasting time series with complex seasonal patterns using exponential smoothing. *Journal of the American Statistical Association*, 106(496):1513–1527.

Lkhagva, B., Suzuki, Y., and Kawagoe, K. (2006). New time series data representation esax for financial applications. pages x115 – x115.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, pages 281–297, Berkeley, Calif. University of California Press.

Montero-Manso, P., Athanasopoulos, G., Hyndman, R. J., and Talagala, T. S. (2020). Fforma: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1):86 – 92. M4 Competition.

Murlidhar, V., Menezes, B., Sathe, M., and Murlidhar, G. (2012). A clustering based forecast engine for retail sales. *Journal of Digital Information Management*, 10.

Nuttall, A. H. and Carter, G. C. (1982). Spectral estimation using combined time and lag weighting. *Proceedings of the IEEE*, 70(9):1115–1125.

Oliveira, J. and Ramos, P. (2019). Assessing the performance of hierarchical forecasting methods on the retail sector. *Entropy*, 21:436.

Oliveira, J., Ramos, P., Kourentzes, N., and Fildes, R. (2020). On the use of penalized dynamic regression and time series methods for modeling and foreasting retail product sales. *Production and Operations Management*.

Paparrizos, J. and Gravano, L. (2015). k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, SIGMOD '15, pages 1855–1870, New York, NY, USA. ACM.

Pfaff, B. (2008). *Analysis of Integrated and Cointegrated Time Series with R*. Springer Publishing Company, Incorporated, 2nd edition.

Rai, P. and Shubha, S. (2010). A survey of clustering techniques. *International Journal of Computer Applications*, 7.

Rice, J. R. (1976). The algorithm selection problem**this work was partially supported by the national science foundation through grant gp-32940x. this chapter was presented as the george e. forsythe memorial lecture at the computer science conference, february 19, 1975, washington, d. c. volume 15 of *Advances in Computers*, pages 65 – 118. Elsevier.

Rousseeuw, P. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65.

Salvador, S. and Chan, P. (2004). Fastdtw: Toward accurate dynamic time warping in linear time and space.

Schwarz, G. (1978). Estimating the dimension of a model. *Ann. Statist.*, 6(2):461–464.

Smith-Miles, K. (2008). Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Comput. Surv.*, 41.

Talagala, T. S., Hyndman, R. J., and Athanasopoulos, G. (2018). Meta-learning how to forecast time series. Monash Econometrics and Business Statistics Working Papers 6/18, Monash University, Department of Econometrics and Business Statistics.

Teräsvirta, T., Lin, C.-F., and Granger, C. W. J. (1993). Power of the neural network linearity test. *Journal of Time Series Analysis*, 14(2):209–220.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society Series B*, 63:411–423.

Timmermann, A. (2006). Forecast combinations. volume 1, chapter 04, pages 135–196. Elsevier, 1 edition.

Trapero, J., Kourentzes, N., and Fildes, R. (2012). Impact of information exchange on supplier forecasting performance. *OMEGA the International Journal of Management Science*, 40:738–747.

Trapero, J., Kourentzes, N., and Fildes, R. (2014). On the identification of sales forecasting models in the presence of promotions. *Journal of the Operational Research Society*, 66:299–307.

Trapero, J., Pedregal, D., Fildes, R., and Kourentzes, N. (2013). Analysis of judgmental adjustments in the presence of promotions. *International Journal of Forecasting*, 29:234–243.

Vlachos, M., Lin, J., Keogh, E., and Gunopulos, D. (2003). A wavelet-based anytime algorithm for k-means clustering of time series. *Proc. Workshop on Clustering High Dimensionality Data and its Applications*.

Wang, X., Smith-Miles, K., and Hyndman, R. (2006). Characteristic-based clustering for time series data. *Data Min. Knowl. Discov.*, 13:335–364.

Wijk, van, J. and Selow, van, E. (1999). Cluster and calendar based visualization of time series data. In *Proceedings IEEE Symposium on Information Visualization (InfoVis'99, San Francisco CA, USA, October 24-29, 1999)*, pages 4–9, United States. IEEE Computer Society.

Wilson, G. T. (2016). Time series analysis: Forecasting and control, 5th edition, by george e. p. box, gwilym m. jenkins, gregory c. reinsel and greta m. ljung, 2015. published by john wiley and sons inc., hoboken, new jersey, pp. 712. isbn: 978-1-118-67502-1. *Journal of Time Series Analysis*, 37(5):709–711.

Anexo A

Anexo em volume separado

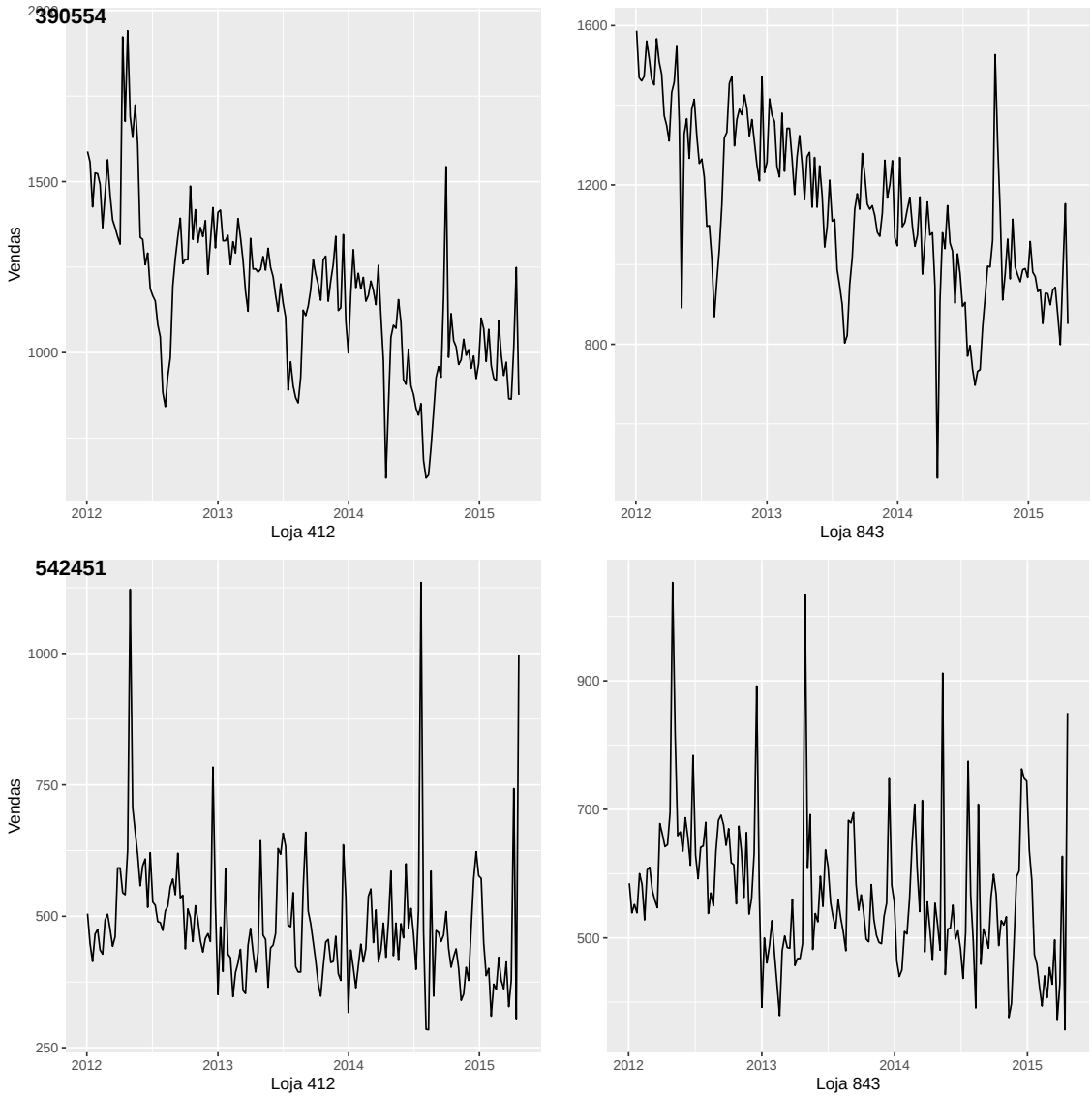

Figura A.1: Seleção de SKUs das lojas 412 (à esquerda) e 843 (à direita).

Tabela A.1: Características de séries temporais usadas na *framework*.

Característica		Descrição
1	trend	Numa decomposição STL da série temporal com r_t como série *remainder* e z_t como a série desprovida de sazonalidade: $max[0, 1 - Var(r_t)/Var(z_t)]$.
2	seasonality	Numa decomposição STL da série temporal com r_t como série *remainder* e x_t como a série desprovida de tendência: $max[0, 1 - Var(r_t)/Var(x_t)]$.
3	linearity	Numa decomposição STL da série temporal sendo T_t a componente tendência, ajustar o modelo quadrático: $T_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \varepsilon_t$. A linearidade é o coeficiente β_1.
4	curvature	Numa decomposição STL da série temporal sendo T_t a componente tendência, ajustar o modelo quadrático: $T_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \varepsilon_t$. A curvatura é a coeficiente β_2.
5	spikiness	Numa decomposição STL da série temporal com r_t como série *remainder*, a variância das variâncias *leave one out* de r_t.
6	e_acf1	O primeiro valor da função de autocorrelação (ACF) da série *remainder* de uma decomposição STL da série temporal.
7	e_acf10	A soma dos quadrados dos 10 primeiros valores da função de autocorrelação (ACF) da série *remainder* de uma decomposição STL da série temporal.
8	stability	A variância das médias baseada numa divisão da série em intervalos não sobrepostos. O tamanho dos intervalos é igual à frequência da série ou 10 no caso da frequência ser 1.
9	lumpiness	A variância das variâncias baseada numa divisão da série em intervalos não sobrepostos. O tamanho dos intervalos é igual à frequência da série ou 10 no caso da frequência ser 1.
10	entropy	A entropia espetral da série temporal. $H_s(x_t) = -\int_{-\pi}^{\pi} f_x(\lambda) \log f_x(\lambda) d\lambda$ onde a densidade é normalizada logo $\int_{-\pi}^{\pi} f_x(\lambda) d\lambda = 1$.
11	hurst	O coeficiente de hurst, que indica o nível de diferenciação fracional de uma série temporal.
12	nonlinearity	Uma estatística de não linearidade baseada no teste Tearsvirta de não linearidade de uma série temporal.
13	alpha	O parâmetro de alisamento do nível num modelo ETS(A,A,N) ajustado à série temporal, α.
14	beta	O parâmetro de alisamento da tendência num modelo ETS(A,A,N) ajustado à série temporal, β.
15	ur_pp	O valor da estatística da versão "Z-alpha"do teste de raízes unitárias de Phillips & Perron com tendência constante e *lag* 1.
16	ur_kpss	O valor da estatística do teste de raízes unitárias de Kwiatkowski et al. com tendência linear e *lag* 1.
17	y_acf1	O primeiro valor da função de autocorrelação (ACF) da série temporal.
18	diff1y_acf1	O primeiro valor da função de autocorrelação (ACF) da série diferenciada.
19	diff2y_acf1	O primeiro valor da função de autocorrelação (ACF) da série diferenciada duas vezes.

Tabela A.1: Características de séries temporais usadas na *framework*.

Característica		Descrição
20	y_acf10	A soma dos quadrados dos 10 primeiros valores da função de autocorrelação (ACF) da série temporal.
21	diff1y_acf10	A soma dos quadrados dos 10 primeiros valores da função de autocorrelação (ACF) da série diferenciada.
22	diff2y_acf10	A soma dos quadrados dos 10 primeiros valores da função de autocorrelação (ACF) da série diferenciada duas vezes.
23	seas_acf1	O coeficiente de autocorrelação do primeiro *lag* sazonal. Se a série é não sazonal então esta característica é 0.
24	y_pacf5	A soma dos quadrados dos 5 primeiros valores da função de autocorrelação parcial (PACF) da série temporal.
25	diff1y_pacf5	A soma dos quadrados dos 5 primeiros valores da função de autocorrelação parcial (PACF) da série diferenciada.
26	diff2y_pacf5	A soma dos quadrados dos 5 primeiros valores da função de autocorrelação parcial (PACF) da série diferenciada duas vezes.
27	seas_pacf	O coeficiente de autocorrelação parcial do primeiro *lag* sazonal. Se a série é não sazonal então esta característica é 0.
28	crossing_point	O número de vezes que a série temporal cruza a mediana.
29	flat_spots	O número de *flat spots* da série temporal, calculado discretizando a série em 10 intervalos de igual comprimento e contando o maior *run lenght* em todos os intervalos.
30	peak	A localização do pico (valor máximo) da componente sazonal de uma decomposição STL da série temporal.
31	trough	A localização do "vale" (valor mínimo) da componente sazonal de uma decomposição STL da série temporal.
32	ARCH.LM	O valor da estatística baseada no teste de Multiplicador de Lagrange de Engle (Engle, 1982) para heterocedasticidade condicional autorregressiva. O valor de R^2 de um modelo autorregressivo com 12 *lags* aplicado ao quadrado da série temporal após a subtração da sua média.
33	arch_acf	Depois da série ser *pre-whitened* usando um modelo AR e elevada ao quadrado. A soma dos quadrados dos 12 primeiros valores da função de autocorrelação (ACF).
34	garch_acf	Depois da série ser *pre-whitened* usando um modelo AR, um modelo GARCH(1,1) é ajustado e os resíduos são calculados. A soma dos quadrados dos 12 primeiros valores da função de autocorrelação (ACF) dos quadrados dos resíduos.
35	arch_r2	Depois da série ser *pre-whitened* usando um modelo AR e elevada ao quadrado, o valor de R^2 de um modelo AR ajustado à mesma.
36	garch_r2	Depois da série ser *pre-whitened* usando um modelo AR, um modelo GARCH(1,1) é ajustado e os resíduos são calculados, o valor de R^2 de um modelo AR ajustado aos mesmos.

Tabela A.2: Estatística descritiva das características das séries de vendas dos SKUs da loja 412.

Característica		Mínimo	Q1	Mediana	Média	Q3	Máximo
1	trend	0.00000	0.07321	0.18836	0.25333	0.38784	0.90287
2	seasonality	0.1993	0.3718	0.4341	0.4600	0.5126	0.9087
3	linearity	-10.380	-4.717	-1.752	-2.026	0.590	9.960
4	curvature	-7.9538	-1.0284	0.3056	0.2505	1.4806	5.4667
5	spikiness	7.984e-07	1.766e-05	4.652e-05	1.189e-04	1.533e-04	1.190e-03
6	e_acf1	-0.41463	-0.02736	0.08588	0.10101	0.21523	0.76637
7	e_acf10	0.003822	0.066616	0.108285	0.160884	0.184196	2.758814
8	stability	0.000188	0.033620	0.108518	0.190888	0.277115	1.069793
9	lumpiness	0.0004943	0.0832366	0.2212171	0.4228534	0.5599705	2.8117457
10	entropy	0.5749	0.8947	0.9508	0.9256	0.9793	0.9996
11	hurst	0.5000	0.6190	0.7557	0.7448	0.8730	0.9978
12	nonlinearity	0.00032	0.13369	0.33184	0.52787	0.74675	3.39546
13	alpha	0.00010	0.03783	0.17188	0.18767	0.29322	0.98413
14	beta	0.000100	0.000100	0.000100	0.000747	0.000100	0.107266
15	ur_pp	-214.047	-147.088	-117.811	-111.378	-78.126	-7.268
16	ur_kpss	0.03133	0.22440	0.54159	0.79023	1.21299	2.87453
17	y_acf1	-0.2153	0.1183	0.3174	0.3191	0.5134	0.9126
18	diff1y_acf1	-0.70967	-0.50280	-0.46235	-0.44817	-0.40701	0.01797
19	diff2y_acf1	-0.8296	-0.6690	-0.6427	-0.6363	-0.6078	-0.3317
20	y_acf10	0.002084	0.108123	0.397614	0.835716	1.185060	6.566201
21	diff1y_acf10	0.04775	0.22785	0.26973	0.29038	0.33620	1.31469
22	diff2y_acf10	0.2040	0.4400	0.5013	0.5266	0.5891	1.7919
23	seas_acf1	-0.486607	-0.006632	0.052982	0.077744	0.139407	0.575433
24	y_pacf5	0.0007382	0.0620870	0.1851944	0.2457416	0.3788272	0.9601689
25	diff1y_pacf5	0.03729	0.31998	0.40256	0.40278	0.48403	0.89685
26	diff2y_pacf5	0.3873	0.8603	0.9536	0.9439	1.0398	1.4411
27	seas_pacf	-0.19827	-0.02460	0.01654	0.02956	0.07390	0.50072
28	crossing_point	8.00	41.00	52.00	51.01	61.00	88.00
29	flat_spots	2.00	5.00	7.00	11.81	12.00	108.00
30	peak	1.00	18.00	24.00	27.97	42.00	52.00
31	trough	1.00	25.00	33.00	31.84	45.00	52.00
32	ARCH.LM	0.000239	0.016968	0.067924	0.111271	0.157520	0.796386
33	arch_acf	0.0002016	0.0123364	0.0409885	0.0656795	0.0895755	0.9516111
34	garch_acf	0.0000014	0.0094119	0.0297973	0.0479609	0.0624214	0.4865402
35	arch_r2	0.000239	0.014025	0.046863	0.071847	0.096315	0.838853
36	garch_r2	0.0002863	0.0117704	0.0356617	0.0575378	0.0720339	0.8389969

Tabela A.3: Estatística descritiva das características das séries de vendas dos SKUs da loja 843.

Característica		Mínimo	Q1	Mediana	Média	Q3	Máximo
1	trend	0.002803	0.101309	0.228334	0.298865	0.456274	0.895842
2	seasonality	0.2428	0.3760	0.4394	0.4610	0.5171	0.9104
3	linearity	-11.0126	-5.5998	-2.3915	-2.4757	0.3494	10.1194
4	curvature	-7.7784	-0.9666	0.1683	0.2404	1.4915	6.3831
5	spikiness	6.637e-7	1.415e-5	3.480e-5	8.743e-5	9.077e-5	1.145e-3
6	e_acf1	-0.37399	-0.02840	0.09355	0.10668	0.22218	0.85913
7	e_acf10	0.01009	0.07248	0.11354	0.17178	0.18846	2.81259
8	stability	0.0003953	0.0547865	0.1591239	0.2451535	0.3628168	1.0825242
9	lumpiness	0.0000762	0.061131	0.0140606	0.3377542	0.459097	2.612448
10	entropy	0.5881	0.8748	0.9380	0.9123	0.9727	0.9992
11	hurst	0.5000	0.6558	0.7902	0.7721	0.8932	0.9990
12	nonlinearity	0.000575	0.149523	0.373522	0.573374	0.831974	3.201985
13	alpha	0.00010	0.07809	0.18384	0.20296	0.29705	0.99990
14	beta	0.000100	0.000100	0.000100	0.0001003	0.000100	0.087527
15	ur_pp	-215.621	-143.707	-107.402	-105.250	-70.551	-5.912
16	ur_kpss	0.03972	0.28809	0.75276	0.96018	1.55562	3.00312
17	y_acf1	-0.2590	0.1556	0.3634	0.3599	0.5451	0.9499
18	diff1y_acf1	-0.72551	-0.49856	-0.45397	-0.44302	-0.39849	0.05675
19	diff2y_acf1	-0.8385	-0.6701	-0.6413	-0.6343	-0.6060	-0.3004
20	y_acf10	0.00307	0.18010	0.55364	1.05014	1.49588	6.44286
21	diff1y_acf10	0.03826	0.22639	0.27701	0.28782	0.33397	1.15326
22	diff2y_acf10	0.2105	0.4451	0.5054	0.5265	0.5893	1.5937
23	seas_acf1	-0.337712	-0.005488	0.059005	0.077462	0.136815	0.538635
24	y_pacf5	0.001012	0.096667	0.231308	0.288777	0.440297	1.009330
25	diff1y_pacf5	0.02981	0.31753	0.39680	0.39881	0.48150	0.93801
26	diff2y_pacf5	0.4074	0.8569	0.9523	0.9413	1.0338	1.4282
27	seas_pacf	-0.16673	-0.02227	0.02470	0.03059	0.07272	0.42898
28	crossing_point	7.00	38.00	49.00	48.35	58.00	87.00
29	flat_spots	3.00	5.00	6.00	10.17	10.00	117.00
30	peak	1.00	18.00	27.00	28.83	44.00	52.00
31	trough	1.00	24.00	33.00	31.74	45.00	52.00
32	ARCH.LM	0.0004881	0.0280477	0.0794947	0.1250048	0.1792217	0.7637426
33	arch_acf	0.0003378	0.0198539	0.0521847	0.0733563	0.0962240	0.6714918
34	garch_acf	0.0001082	0.0131410	0.0361233	0.0495753	0.0676114	0.3583273
35	arch_r2	0.0004881	0.0217880	0.0589084	0.0759717	0.1061966	0.4295418
36	garch_r2	0.000497	0.015827	0.042970	0.058745	0.078850	0.951333