UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Câmpus de Sorocaba

PROGRAMA DE PÓS-GRADUAÇÃO em
Ciências Ambientais

YURI VASCONCELOS DE ALMEIDA SÁ

A UTILIZAÇÃO DE META-HEURÍSTICAS EM SÉRIES TEMPORAIS *FUZZY* PARA UTILIZAÇÃO EM SÉRIES COM EXCESSOS DE ZERO

Sorocaba

2023

YURI VASCONCELOS DE ALMEIDA SÁ

A UTILIZAÇÃO DE META-HEURÍSTICAS EM SÉRIES TEMPORAIS *FUZZY* PARA UTILIZAÇÃO EM SÉRIES COM EXCESSOS DE ZERO

Dissertação apresentada como requisito para a obtenção do título de Mestre em Ciências Ambientais da Universidade Estadual Paulista "Júlio de Mesquita Filho" na Área de Concentração Diagnóstico, Tratamento e Recuperação Ambiental

Orientador: Prof. Dr. José Arnaldo Frutuoso Roveda
Coorientador: Prof. Dr. Henrique Ewbank

Sorocaba

2023

S111u

Sá, Yuri Vasconcelos de Almeida

A utilização de meta-heurísticas em séries temporais fuzzy para utilização em séries com excessos de zero / Yuri Vasconcelos de Almeida Sá. -- Sorocaba, 2023

68 p. : il., tabs.

Dissertação (mestrado) - Universidade Estadual Paulista (Unesp), Instituto de Ciência e Tecnologia, Sorocaba

Orientador: José Arnaldo Frutuoso Roveda

Coorientador: Henrique Ewbank de Miranda Vieira

1. Logica Difusa. 2. Análise de séries temporais. 3. Impacto ambiental. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Instituto de Ciência e Tecnologia, Sorocaba. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

UNIVERSIDADE ESTADUAL PAULISTA

Câmpus de Sorocaba

CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: A UTILIZAÇÃO DE METAHEURÍSTICAS EM SÉRIES TEMPORAIS FUZZY PARA UTILIZAÇÃO EM SÉRIES COM EXCESSOS DE ZERO

AUTOR: YURI VASCONCELOS DE ALMEIDA SÁ
ORIENTADOR: JOSE ARNALDO FRUTUOSO ROVEDA
COORIENTADOR: HENRIQUE EWBANK DE MIRANDA VIEIRA

Aprovado como parte das exigências para obtenção do Título de Mestre em Ciências Ambientais, área: Diagnóstico, Tratamento e Recuperação Ambiental pela Comissão Examinadora:

Prof. Dr. JOSE ARNALDO FRUTUOSO ROVEDA (Participaçao Presencial)
Departamento de Engenharia Ambiental / Instituto de Ciencia e Tecnologia UNESP Campus de Sorocaba

Prof. Dr. FÁBIO DE OLIVEIRA NEVES (Participaçao Presencial)
Universidade Federal de Alfenas

Prof. Dr. ALEXANDRE DA SILVA SIMÕES (Participaçao Presencial)
Departamento de Engenharia de Controle e Automação / Instituto de Ciência e Tecnologia - UNESP - Câmpus de Sorocaba

Sorocaba, 10 de março de 2023

RESUMO

O presente estudo investigou séries temporais fuzzy com excessos de zeros, empregando técnicas metaheurísticas, e comparou seus resultados de eficiência com diversos modelos propostos na literatura. Para isso, realizou-se uma revisão abrangente da literatura, com o propósito de identificar o estado da arte na área e modelos que pudessem ser implementados e comparados. Foi avaliada uma série de dados proveniente de uma indústria transformadora de madeira, tendo como objetivo direto aprimorar a eficiência da utilização de matéria-prima (madeira), visando racionalizar o impacto ambiental decorrente do processo de fabricação de paletes.

Palavras-chave: lógica difusa; análise de séries temporais; impacto ambiental.

ABSTRACT

This study examined fuzzy time series with excess zeros using metaheuristic optimization techniques, and compared their efficiency results with various models proposed in the literature. To achieve this, a comprehensive review of the literature was conducted to identify the current state-of-the-art in the field and models that could be implemented and compared. The study entailed an evaluation of a series of data obtained from a wood manufacturing industry, with the direct objective of enhancing the efficiency of raw material (wood) utilization, thereby striving to rationalize the environmental impact caused by the pallet manufacturing process.

Keywords: fuzzy logic; time series analysis; environmental impact.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 Introdução

A Cadeia de Suprimentos (CS)[1] define-se como sendo o fluxo de todas as atividades de uma organização, desde a origem (aquisição da matéria-prima) até o consumo do produto acabado pelo usuário final (PIRES, 2004), envolvendo não somente o fluxo de produtos, mas também de serviços e informações (YANG e WEI, 2013).

Neste contexto, nota-se um esboço da Gestão da Cadeia de Suprimentos (GCS)[2] no processo logístico empresarial entre as décadas de 80 e 90, sendo, então, mais amplamente definida como a gestão da cadeia logística para planejamento e integração dos fluxos de processos (suprimentos e recursos), incluindo o levantamento e controle de informações dos usuários finais e fornecedores originais, a fim de adicionar valor ao produto final (LAMBERT et al., 1998).

Acompanhando a alteração comportamental de consumidores cada vez mais exigentes para aquisição de produtos de empresas que possuam boas práticas ambientais, houve o impulso para a Gestão Sustentável da Cadeia de Suprimentos (GSCS)[3]. Este movimento provocou a implementação de ações socioambientais como geração de valor em seu processo logístico (CARTER, 2002; GOVINDAN et al.; 2016).

Não limitado ao consumidor, os indicadores dos princípios *ESG, Environment Social Governance,* são observados de perto por investidores o que também demonstra uma mudança comportamental em relação às empresas que têm responsabilidade ambiental (LITVINENKO, 2022) no mercado financeiro, alinhando até mesmo os interesses de investidores em uma cadeia de suprimento sustentável e socialmente responsável

Em resumo, a previsão de demanda de qualquer atividade comercial é fundamental para o planejamento operacional e consequente eficiência de todo o processo. Através de uma previsão adequada, é possível preparar e ajustar melhor os recursos, sejam humanos ou materiais. Em contrapartida, ao prever o processo de

[1] Do inglês *Supply Chain (SC*).

[2] Do inglês *Supply Chain Management (SCM*).

[3] Do inglês *Sustainable Supply Chain Management* (*SSCM).*

demanda de forma deficiente, pode-se gerar excesso de utilização de recursos, acúmulo ou falta de estoque, entre outros danos. Esse descontrole no processo acarreta ainda mais danos se o objeto da atividade comercial envolve diretamente o meio ambiente, pois sua ineficiência pode aumentar injustificadamente o impacto ambiental (EWBANK et al. 2020).

A produção de *pallets* observada neste trabalho é um processo de transformação primário com baixo valor agregado, onde é feita a manufatura do produto final a partir do beneficiamento de tábuas de madeira de baixa qualidade, pois os pallets, muitas vezes, são descartados após o primeiro uso. A estocagem do produto final é volumosa e muito dispendiosa, gerando um elevado custo de retenção associado a um baixo valor unitário.

A falta de precisão na previsão da demanda de *pallets* causa um descontrole na aquisição de matéria-prima, madeira bruta, para a transformação em tábuas e, então, manufaturadas em *pallets*. A estocagem de longo prazo da matéria prima ocorre em condições adversas que favorecem a deterioração do material, o que gera um aumento da extração da madeira bruta e a criação de dano ambiental desnecessário.

Como consequência desta necessidade de desenvolvimento de processos, começou-se a utilizar a tecnologia como oportunidade de otimização das cadeias logísticas, agregando valor, e possibilitando o surgimento de novos modelos de negócios (HOFMANN e RÜSCH, 2017). As previsões corretas na cadeia logística são de elevada relevância, pois apoiam as tomadas de decisões em curto, médio e longo prazo. Vale ressaltar que a qualidade e a precisão das previsões de demanda impactam áreas distintas dentro de uma empresa de várias maneiras (EWBANK et al. 2020).

As séries temporais compõem um ramo da ciência de dados que envolve uma dimensão temporal, em ordem sucessiva. Este tipo de conjunto de dados pode conter uma só dimensão, como o próprio resultado (y), ou pode conter mais dimensões (x). Ao primeiro, convenciona-se chamar de univariado, enquanto o último, de multivariado.

A evolução de um processo ambiental é invariavelmente temporal, ainda mais quando este processo é antrópico como por exemplo, a produção de *pallets* de madeira. Para calcular seu impacto, e para tornar os processos produtivos mais

eficientes para a redução deste impacto, é imperativo o uso de modelos temporais para análise e apoio à tomada de decisão.

Dentre os modelos tradicionais, os mais comumente utilizados são variações de modelos Auto Regressivos, dentre eles: *ARIMA (Auto Regressive Integrated Moving Average), SARIMA (Seasonal Autoregressive Integrated Moving Average), ARMA (Autoregressive Moving Average)* (CHATFIELD, 2000).

A demanda e produção de *pallets* de madeira de uma pequena fábrica foram estudadas para este trabalho e a manufatura do produto final é bastante simples. Com pouco beneficiamento desde a madeira bruta até a entrega existe pouca tecnologia, o processo se reduz à transformação de madeira bruta em tábuas, corte e montagem. Este processo permite que a produção e seu planejamento sejam ligados diretamente à demanda (venda) do produto final.

Ao analisar a demanda dos pallets nesta fábrica, tornou-se evidente um padrão peculiar na constituição desta série temporal: ela não apresenta uma tendência clara, e muitas vezes, não existe demanda, portanto, a demanda é zero. Mais interessante é que não ocorre em intervalos regulares, o que gera problemas adicionais aos modelos tradicionais, dependentes de médias móveis. Quando há uma contagem alta de zeros, a média móvel que compõe esses modelos perde a informação que carrega ao longo do tempo, diminuindo dramaticamente a eficácia dos mesmos.

Embora cada modelo tenha características e sensibilidades diferentes, em geral, é possível atingir o seu potencial em séries temporais sem a presença de zeros ou ainda com uma baixa contagem de zeros. Para DOSZYŃ (2017), modelos que dependem de análise direta dos valores da série temporal, como médias, desvio, amplitude e variância, são sensivelmente alterados na presença de zeros, apontando para a necessidade de modelos específicos para situações com excessos ou mera presença de zeros na série temporal.

Existem modelos criados especialmente para calcular e prever séries com alta contagem de zero. O mais difundido deles é o *Zero Inflated Poisson* (*ZIP)* (LAMBERT, 1992), no qual o estimador é baseado em uma distribuição probabilística que permite uma grande contagem de zeros. Variações deste modelo adicionando cadeias de Markov e heterocedasticidade melhoraram os resultados para séries específicas,

porém não há um trabalho buscando a generalização de modelos com alta contagem de zeros.

Os modelos de Séries Temporais *Fuzzy* são indicados quando há alguma forma de imprecisão na série de dados. Isso pode acontecer na maneira como os dados foram registrados, ou na maneira como a série de dados é interpretada.

Os modelos temporais *fuzzy* foram propostos por Song (1993) e desde então tem sido desenvolvida uma variedade de modelos e métodos para aprimorar seu desempenho. Em geral, os modelos funcionam dividindo a amplitude do período estudado em conjuntos *fuzzy*, que são ordenados em séries (cadeias), e, após a sua devida conversão (defuzzificação), um número (predição) é produzido (SONG, 1993).

Esses modelos podem acomodar melhor uma alta contagem de zeros, uma vez que não utilizam necessariamente o número zero para calcular a série, mas sim o conjunto *fuzzy* que contém o zero. Ewbank et al. (2020) propuseram uma abordagem focada em séries temporais *fuzzy* com excessos de zero, aplicando uma inteligência na inicialização de algoritmos *fuzzy* de agrupamentos. O modelo proposto apresentou resultados melhores, quando comparado com demais modelos da literatura.

Mesmo assim, essas técnicas ainda possuem um desempenho limitado por outros fatores, tais como sazonalidade, frequência e formação dos conjuntos. A estes pontos podem ser aplicados processos de meta-heurística: algoritmos de busca que visam generalizar os modelos. Esses processos criam e automatizam partes do modelo, buscando o melhor ajuste e otimização de parâmetros. Um exemplo notório é apresentado por Huarng (2001) que automatiza totalmente o particionamento e número de partições, se adequando à própria série temporal.

Este trabalho tem como objetivo aplicar e avaliar modelos específicos de séries temporais *Fuzzy* (*FTS*), em uma base de dados com poucas amostras e com excessos de zero, para avaliar seu desempenho sob tais condições e, assim, poder oferecer uma comparação entre os modelos, testando suas possibilidades.

O rol de modelos escolhidos foram Song (1993), Cheng (2009) e Sadaei (2014), para que se pudesse criar uma evolução dos modelos *FTS* e o que os qualifica para a resolução de previsão em séries temporais com excesso de zeros.

Primeiramente, pretende-se caracterizar e criar uma breve descrição da base de dados e suas métricas críticas, incluindo a análise de zeros. Em seguida, de posse dos dados carregados, será desenvolvido um sistema de combinação entre os diversos componentes de um modelo *FTS* (Tipo de modelo, particionadores e número de partições). Após a execução de todas as combinações, os respectivos resultados de cada modelo serão confrontados e avaliados segundo suas métricas de eficiência.

Pretende-se demonstrar com esta pesquisa que é possível utilizar *FTS* para predizer com eficiência os próximos passos de uma série temporal com excessos de zeros e uma base de dados com poucas amostras.

Nos próximos capítulos será descrito um histórico das séries *FTS* como um modelo de previsão de séries temporais, citando a evolução desta técnica através de suas principais obras.

Após essa revisão bibliográfica, apresenta-se a fonte de dados, bem como uma breve análise quantitativa das suas propriedades e os desafios encontrados pelo caso específico da demanda de *pallets*.

Em seguida, é estabelecida a metodologia para criação dos modelos e respectivos campos experimentais para, então, determinar como os resultados serão comparados e avaliados.

Para efeitos de comparação, outra técnica para regressão que contém algum grau de aprendizagem, e não somente médias móveis, será aplicada aos bancos de dados utilizados no trabalho. Utilizamos *Support Vector Regression (SVR).*

Uma técnica derivada de *Support Vector Machines (SVM)*, o *SVR* é um modelo preditivo e fornece um resultado contínuo quando aplicado a uma série de dados (DASH, 2021).

Após a execução de todos os modelos e obtenção de resultados, segue-se à discussão dos resultados para cada conjunto otimizado com parâmetros de cada modelo *FTS*.

2 Revisão Bibliográfica

Durante os últimos 30 anos, diversas metodologias e técnicas têm sido incorporadas aos modelos Fuzzy Time Series (FTS), iniciando com Song e Chissom (1993). Atualmente, a precisão e utilização desses modelos têm melhorado significativamente (BOSE, 2019). Geralmente, os sistemas FTS passam pelas fases de: (i) definição do universo de discurso; (ii) particionamento; (iii) fuzzificação; (iv) estabelecimento de relações fuzzy; e (v) defuzzificação (se necessária). Essas fases podem ser divididas em duas outras fases: particionamento e predição (KUMAR, 2021).

Os métodos de particionamento têm evoluído de estáticos e uniformes para dinâmicos, incorporando modelos de outras disciplinas da inteligência artificial (YOLCU, 2018), em que o resultado pode depender de uma otimização profunda antes da construção do modelo. A maior parte do desenvolvimento ocorreu em técnicas de agrupamento (clustering) para o particionamento e otimização de hiperparâmetros (BOSE, 2019). Além do particionamento dinâmico, outras seções do processo, como fixação das regras de transição e fuzzificação, evoluíram para combinar parte de seu funcionamento com outros tipos de algoritmos inteligentes (WANG, 2018).

Observa-se uma evolução significativa nos modelos FTS desde sua criação, incluindo melhorias no processo de previsão, como a atribuição de pesos em uma tendência (CHENG, 2006) ou o estudo de modelos com altas ordens de relacionamentos (SILVA, 2018). Sadaei (2014) adicionou um processo de busca harmônica na geração dos relacionamentos e regras de transição, criando um sistema que encontra o melhor candidato para a tarefa, limitando sua busca na base de regras, e adicionando pesos exponenciais quando uma tendência é detectada.

Alguns estudos têm explorado a combinação de modelos de aprendizagem profunda, como Long Short-Term Memory, LSTM (KOCAK, 2021). Entretanto, esses modelos exigem um volume elevado de amostras para que o modelo aprenda e consiga um bom desempenho dentro dos parâmetros esperados. Além disso, parâmetros que constroem modelos FTS a partir de bases de dados multidimensionais têm sido propostos, convergindo as séries temporais fuzzy para modelos evolutivos e análise estatística (SEVERIANO, 2021).

Esses estudos trouxeram novas direções para a exploração dos sistemas FTS, tornando sua aplicação ainda mais ampla e flexível no cenário de muitas fontes de dados (SILVA, 2018). Como resultado, atualmente, os modelos FTS são amplamente utilizados em previsões de demanda, finanças, medicina, dentre outros setores.

3 Metodologia

3.1 Caracterização da base de dados

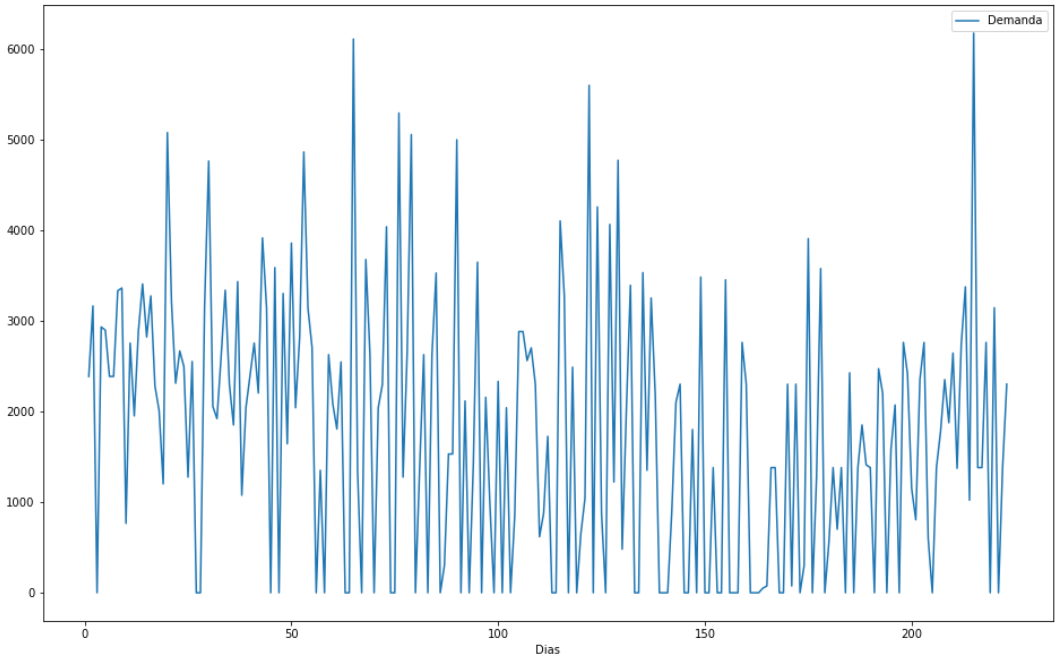
Os dados apresentados neste trabalho foram adquiridos de um fabricante de *pallets* no estado de São Paulo, Brasil.

A série histórica apresentada é única, devido a sua imprevisibilidade com largos saltos na demanda diária, muitas vezes se reduzindo a zero, o que torna a aplicação de métodos tradicionais de predição como médias móveis, exponenciais e até mesmo (S)ARIMA ineficazes, devido aos resultados distorcidos, dada a presença massiva de zeros (EWBANK et al, 2020).

A base de dados da demanda diária de *pallets* da indústria estudada está contida no Anexo - A, a qual todos os dados foram compilados e registrados neste trabalho.

Para uma melhor compreensão da evolução da demanda, e como ela está distribuída ao longo do tempo, pode-se fazer uma análise conforme gráfico demonstrado na Figura 1.

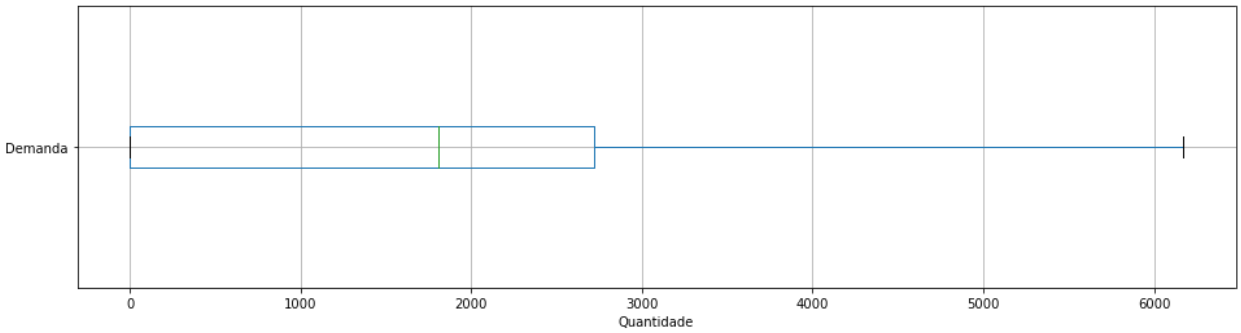
Figura 1 - Gráfico da série completa



Fonte: Elaborado pelo autor (2022).

Através da análise da Figura 1, pode-se observar que a série temporal apresenta uma grande amplitude, oscilando entre zero e 6170 unidades comercializadas em um dia. Além disso, são identificadas tendências de aumento e queda da demanda ao longo do tempo. Essas variações complexas tornam extremamente difícil realizar previsões precisas e úteis para a operação diária sem o suporte de uma metodologia adequada

Figura 2 - *Boxplot* da série temporal



Fonte: Elaborado pelo autor (2022).

Na Figura 2, pode-se observar que existe uma concentração na demanda entre zero e 3000 unidades diárias, com 25% dos pedidos sendo 3000 (aproximadamente) até pouco mais de 6000 unidades, conforme Tabela 1.

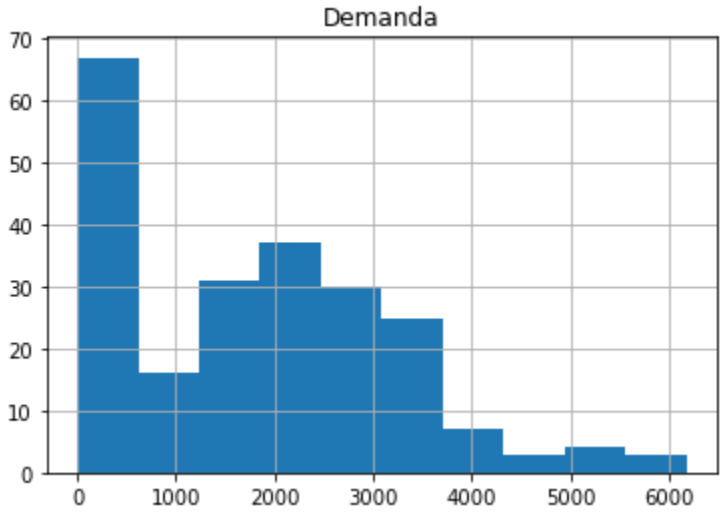
Tabela 1 - Características da série temporal

Dias	Média	Desvio Padrão	Min.	1o Quartil	Mediana	3o Quartil	Máximo
223	1749	1465.4	0	0	1805	2720	6170

Fonte: EWBANK (2020)

Diante de um cenário tão disperso, e considerando a elevada presença de zeros na análise da distribuição das amostras, é possível exemplificar de forma mais notável como os zeros compõem a distribuição de forma quantitativa.

Figura 3 - Histograma da distribuição da demanda



Fonte: EWBANK (2020).

A Figura 3 demonstra que existe uma concentração de pedidos em torno de 2000 unidades, porém a contagem de zeros presentes na distribuição é bastante grande.

3.2 Modelos FTS

Os modelos *FTS* utilizam a lógica difusa proposta por Zadeh (1965), que se diferencia da lógica booleana, ao propor uma categorização não binária, introduzindo o conceito de pertencimento. Sendo assim, os valores podem pertencer a mais de um grupo, em maior ou menor grau.

Este conceito de pertencimento foi utilizado em séries temporais pela primeira vez por Song e Chissom (1993), introduzindo as séries temporais *Fuzzy* ou *Fuzzy Time Series* (FTS), em que é descrito método e algoritmo, utilizando a Lógica *Fuzzy*.

Neste trabalho serão utilizados três modelos para a previsão da série proposta, sendo Song e Chissom (1993) o primeiro modelo a apresentar a técnica.

Cheng (2009) discute e cria evoluções significativas à norma clássica que vinham desde Song e Chissom. Após construir e estabelecer as regras de transição entre os conjuntos *fuzzy,* é aplicado o conceito de tendência (alta, estável ou baixa), e a esta tendência é aplicado um peso baseado na sua recorrência.

E Finalmente Sadaei (2014), que utiliza metaheurísticas para criar um modelo dinâmico para a caracterização das regras de transição entre os conjuntos *fuzzy*.

3.2.1 Song e Chissom (1993)

O modelo proposto por Song e Chissom (1993) é o primeiro modelo de FTS que utiliza a Lógica Fuzzy para prever séries temporais. O modelo é dividido em quatro fases: (i) particionamento do conjunto de amostras (universo de discurso) em classes; (ii) verificação das regras de transição entre essas classes; (iii) previsão do próximo período (step); e (iv) defuzzificação do valor (conversão do pertencimento em um número). Embora esse modelo tenha sido pioneiro, ele não é muito eficiente, mas é considerado um parâmetro para demonstrar a evolução dos modelos desde sua concepção.

Como o modelo foi um estudo pioneiro, os autores não puderam refinar o modelo ou testá-lo em diferentes cenários, devido às dificuldades técnicas da época. Além disso, uma das limitações desse modelo é que ele só computava as regras de transição em pares, na forma Ax -> Ay, sem considerar todas as relações anteriores que poderiam levar a Ay. A base de dados utilizada foi a somatória das inscrições anuais para ingresso na Universidade do Alabama, entre 1972 e 1991, que apresenta uma variação relativamente estável entre 14.000 e 19.000 inscrições no período estudado.

Apesar das limitações, este trabalho é fundamental na evolução da FTS, uma vez que os autores propuseram um novo método para previsão de séries temporais, utilizando a lógica fuzzy. A partir desse trabalho, outros pesquisadores se interessaram pelo assunto e começaram a desenvolver novos modelos, buscando melhorar a eficiência e a precisão das previsões.

3.2.2 Cheng (2009)

O modelo proposto por Cheng (2009) incorpora um balanceamento com a tendência, ou seja, dá ênfase na tendência que é verificada nas regras de transição entre os conjuntos *fuzzy*.

O autor introduz neste modelo o conceito de tendência dentro de sua heurística, considerando que existem 3 tendências possíveis, ascendente (ex.: A1 -> A2); descendente (ex.: A2 -> A1); e sem alteração (ex.: A1 -> A1).

Portanto, logo após estabelecer as regras de transição, Cheng propôs organizá-las em tendências e, após esta definição, designar pesos dinamicamente, através da simples contagem de quantas vezes a mesma se repete ao longo da base de dados. Logo, uma tendência que ocorra mais vezes em sua base de dados terá um peso maior que as outras que não ocorrem tantas vezes.

Após esses pesos calculados e normalizados ao longo da série de dados, são aplicados às regras de transição para adaptar os valores previstos aos pesos.

O trabalho utilizou a base de dados da variação de assinantes de *Digital Subscriber Line (DSL)*, acesso à internet banda larga em Taiwan.

3.2.3 Sadaei (2014)

Neste modelo, o autor (SADAEI, 2014) propõe um modelo que emprega um algoritmo híbrido para a classificação das regras de transição entre os conjuntos *fuzzy*, utilizando o balanceamento exponencial da tendência com uma metaheurística de busca harmônica (GEEM, 2001) para seleção das regras de transição entre as partições, a partir de uma seleção de poucos itens recentes.

O modelo, de fato, cria um algoritmo de busca interno, ao qual o autor utiliza o *Mean Absolute Percentage Error (MAPE)*[4] para encontrar qual a melhor combinação de pesos das tendências como Cheng (2009) em um processo que denomina de busca harmônica. Neste processo, quando uma combinação de pesos satisfatória é

[4] Tradução nossa: Porcentagem da Média do Erro Absoluto.

encontrada, o modelo julga ter atingido a harmonia entre os pesos das regras de transição.

O MAPE neste caso é utilizado como uma métrica de busca do algoritmo, um parâmetro da meta-heurística.

O MAPE é definido pela fórmula 1:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{verdadeiro(i) - previsto\ (i)}{verdadeiro(i)}\right| \times 100$$

(1)

Onde *i* é o valor da iteração, ou passo da base de dados, e *n* é a contagem de passos da série temporal.

Portanto, após calcular as regras de transição, o algoritmo é inicializado com valores aleatórios para os pesos das regras de transição e passa a ser realizada uma busca através de iteradores, nos quais novos pesos são testados. Executa-se o modelo em toda a série novamente e verifica-se o erro através do MAPE. Se essa métrica diminui em relação às execuções anteriores, a matriz de pesos associadas a esta execução é armazenada como sendo a melhor; e o teste é feito novamente até que um critério de parada seja atingido. Esse critério de parada não é definido em sua pesquisa e, para este presente trabalho, será utilizado somente 1 (um) como critério de parada, ou seja, se a métrica MAPE aumentar em uma só execução, a busca é encerrada e a melhor matriz de pesos eleita será a anterior ao aumento do MAPE.

Após os pesos serem encontrados pelo algoritmo de busca harmônica e normalizados ao longo da série de dados, eles são aplicados exponencialmente às regras de transição para adaptar os valores previstos aos pesos. Este estudo utilizou a carga de energia elétrica na França e Inglaterra ao longo do ano de 2005 e fez inferências entre os dias da semana. O modelo proposto demonstrou uma melhoria significativa em relação aos modelos tradicionais de FTS, mostrando a eficácia do processo de busca harmônica para encontrar a melhor combinação de pesos.

3.3 Métodos de Particionamento

Os métodos de particionamento constituem a divisão dos valores em intervalos para que as amostras possam ser caracterizadas como pertencentes ou não a este intervalo (também chamado de conjunto).

Essas partições, principalmente em funções de pertencimentos triangulares, podem ser diretamente associadas a métodos de agrupamento (*clustering*), nas quais o centro de cada *cluster* é o centro do triângulo da função de pertencimento de cada agrupamento *fuzzy*. Os limites de cada triângulo, início e fim, podem ser definidos como as fronteiras deste *cluster* ,ou, então, como o centro dos agrupamentos vizinhos. Essa segunda técnica foi utilizada neste trabalho, garantindo a continuidade do pertencimento ao longo do intervalo da série de dados.

Os métodos selecionados para este trabalho são: particionador em grade (fixos), *C-Means* (agrupamento difuso), *Fuzzy C-Means (FCM)* - Agrupamento difuso, porém ajustado através de metaheurística - e entropia (um modelo baseado em entropia para geração dos intervalos para classificação das amostras).

Embora existam vários métodos de particionamento (função de pertencimento) para criação dos conjuntos (Trapezoidal, Triangular, Senoidal, entre outros), neste trabalho será utilizada somente a forma triangular, uma vez que o número elevado de partições e o grande volume de regras de transição entre os conjuntos não sofrem grande alteração com diferentes formas dos conjuntos.

3.3.1 Particionamento em intervalos regulares (grade)

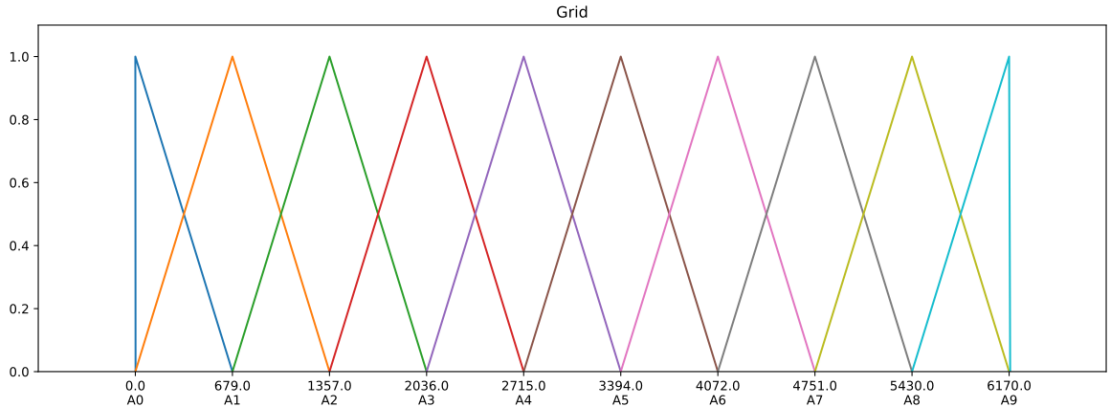
A técnica de particionamento mais intuitiva para a classificação de amostras é aquela que divide o universo de discurso em um número pré-estabelecido de conjuntos regulares, utilizando o critério da formação triangular. Este método tem sido amplamente estudado na literatura científica, tendo sido empregado por pesquisadores renomados como Song e Chissom (1993), bem como em outros trabalhos posteriores.

O princípio subjacente a esta abordagem consiste em avaliar a extensão do universo do discurso e dividir essa extensão pelo número de partições desejado,

gerando assim os conjuntos uniformes. Dessa forma, a técnica de particionamento triangular é capaz de classificar amostras de forma eficiente e precisa, sendo uma ferramenta importante para diversos campos da ciência e da engenharia.

Um exemplo concreto da aplicação dessa técnica pode ser observado na Figura 4, na qual é possível visualizar de forma clara como os conjuntos são formados quando são utilizadas 10 partições para classificar o banco de dados de demanda diária de pallets. Essa análise demonstra a eficácia do método de particionamento triangular na classificação de amostras em diversos contextos práticos.

Figura 4 - Gráfico do particionamento em Grade



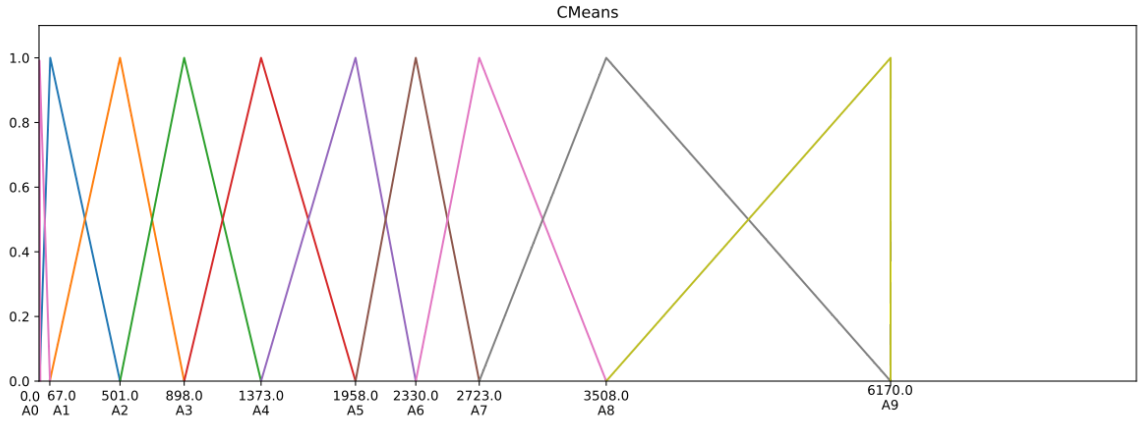
Fonte: Elaborado pelo Autor (2022).

3.3.2 *C-Means*

C-Means é um método de agrupamento (*Clustering*) difuso que implementa um certo grau de lógica *Fuzzy* em sua incepção, ainda que de forma rígida (*hard,* com limites bem definidos), sendo uma modificação do método clássico de agrupamento K-means e foi proposto por DUNN (1973).

A diferença principal é que cada ponto de dados pode pertencer a mais de um conjunto quantitativamente, não gerando necessariamente uma classificação rígida (*hard*).

Figura 5 - Gráfico do particionamento utilizando *C-Means*



Fonte: Elaborado pelo Autor (2022).

Na figura 5 é possível observar a formação de conjuntos utilizando 10 partições quando aplicado ao banco de dados da demanda diária de *pallets*. É possível analisar em detalhe que o conjunto com zero, A0, é muito curto, variando de 0 a 67 em um universo de 6170. A partir disso é possível dizer que criam-se regras de transição que serão bastante específicas quando a previsão apontar para este conjunto.

Ao longo do universo de discurso, ao passo que a demanda aumenta, os conjuntos vão ficando cada vez mais esparsos, com os centros dos conjuntos distantes uns dos outros, o que tende a diminuir sua especificidade.

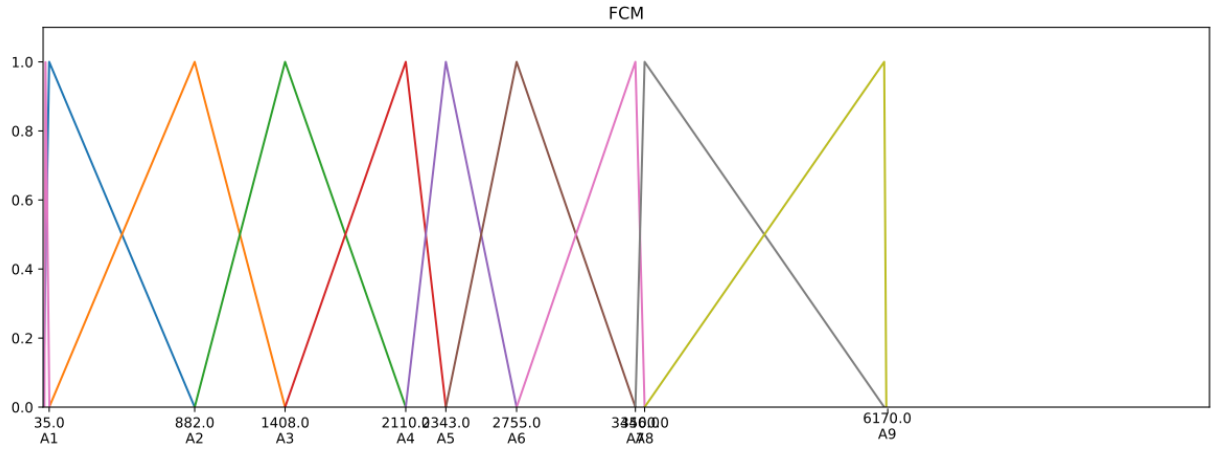
3.3.3 FCM

Embora compartilhe boa parte da lógica e até do nome com o método anterior, o *FCM* também faz o pertencimento difuso como o *C-Means*, porém também aplica isso na classificação do cluster, gerando um conjunto difuso.

A principal diferença é a aplicação do centróide para a geração do pertencimento e sua *defuzificação,* enquanto o método anterior, descrito somente como *C-Means,*

gerava um pertencimento em mais de um grupo. Esta técnica foi proposta por BEZDEK (1984).

Figura 6 - Gráfico do particionamento utilizando FCM



Fonte: Elaborado pelo Autor (2022).

Na figura 6 é possível verificar que o *FCM* é muito semelhante ao *C-Means* com 10 partições aplicado ao banco de dados de *pallets*, uma vez que o *FCM* é baseado em *C-Means*, porém há uma diminuição da amplitude do conjunto A0, aproximando ainda mais o conjunto A0 a zero.

3.3.4 Entropia

A entropia é uma medida de incerteza ou desordem em um sistema, frequentemente utilizada em áreas como física, química e teoria da informação. Na área de inteligência computacional, a entropia tem sido aplicada em diversas técnicas, incluindo as séries temporais difusas.

O método de entropia para séries temporais difusas consiste em utilizar a entropia para particionar a série em intervalos difusos de maneira não arbitrária. Esse

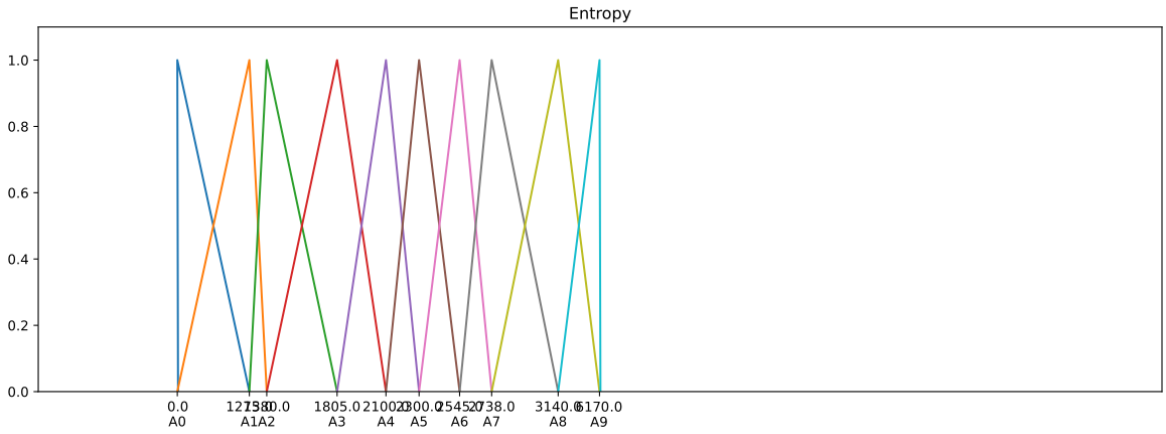
método foi proposto por Cheng (2006) e tem como objetivo criar uma heurística de geração de intervalos difusos mais robusta e precisa.

A ideia principal por trás do uso da entropia é mensurar o grau de incerteza ou aleatoriedade dentro de cada intervalo difuso. Quando a entropia atinge seu valor máximo, isso indica que não há informação suficiente para se determinar um padrão, o que sugere a necessidade de uma divisão em um novo intervalo difuso. Por outro lado, quando a entropia é mínima, significa que existe informação suficiente para determinar um padrão, o que indica que o intervalo difuso atual é adequado. Essa abordagem é altamente eficaz em estudos experimentais e apresenta resultados promissores na previsão de séries temporais difusas.

O método de entropia tem se mostrado eficaz em estudos experimentais, apresentando resultados promissores na previsão de séries temporais difusas. Além disso, ele oferece uma abordagem mais sistemática e objetiva na definição dos intervalos difusos, evitando a subjetividade inerente a outros métodos.

Na figura 7 verifica-se o particionamento de entropia utilizando 10 partições aplicadas no banco de dados da demanda diária de pallets. Aqui, diferentemente dos outros métodos, o conjunto A0 é longo, de zero a 1215, o que pode indicar uma menor especificidade quando as regras de transição apontarem para zero.

Figura 7 - Gráfico do particionamento utilizando Entropia



Fonte: Elaborado pelo Autor (2022).

3.3.5 Comparação entre os métodos de particionamento

Os métodos são muitos e boa parte do desenvolvimento dos sistemas *FTS* se concentra no modo como é realizado o particionamento para atingir resultados mais precisos a um custo computacional menor.

A este trabalho não cabe exatamente uma comparação de eficiência entre tais métodos de particionamento em si, mas, sim, uma análise de como esses métodos se diferenciam entre si dentro de uma base de dados com características únicas e como o número de partições a ser utilizado influencia no resultado.

3.4 Fuzzificação e Regras de Transição

Tradicionalmente, os conjuntos temporais *fuzzy* são nomeados por Ax, onde X representa o numeral associado à ordem crescente a partir da origem dos dados. Portanto, para 10 partições, os conjuntos serão nomeados de A0 a A9, conforme apresentado nas Figuras 4 a 7.

O processo de fuzzificação pode ser compreendido como a associação de um valor numérico a um destes conjuntos.

A partir deste ponto, os modelos *FTS* se diferenciam da metodologia *fuzzy* clássica, pois aqui cada modelo tem uma heurística diferente para as regras de transição. Entretanto, geralmente, essas regras de transição são obtidas a partir da base de dados de treino e armazenadas para prever os próximos valores.

Uma regra de transição pode ser descrita como um caminho e sua consequência, como no exemplo a seguir.

A3, A4 -> A1

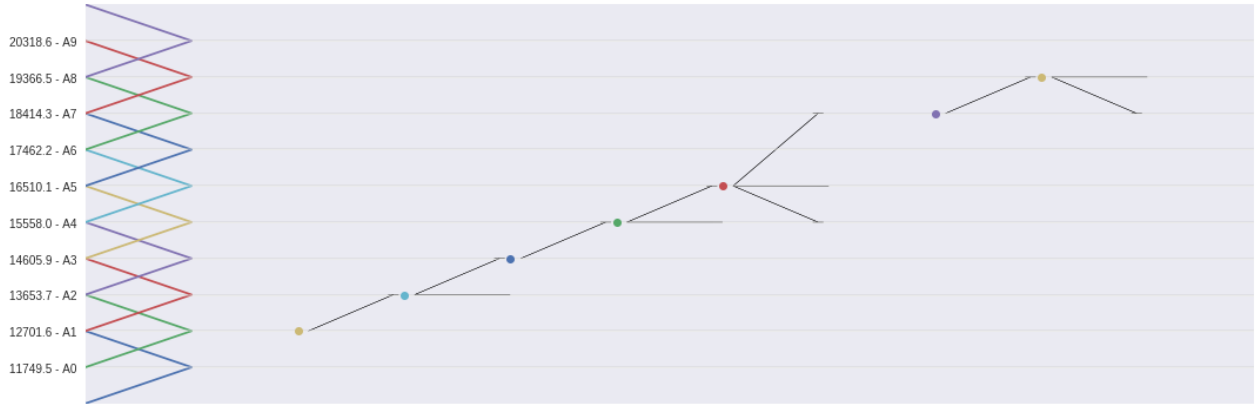
Esta notação representa uma regra de transição da qual uma série temporal parte de um número presente no conjunto A3 e tem seu próximo passo no conjunto A4. Logo, o próximo passo será em A1.

Essas regras são detectadas e tratadas de forma diferente para cada modelo, podendo envolver pesos para uma frequência maior (SADAEI, 2014), ou, então, classificando em ordem de importância conforme um critério de tendência (CHENG, 2009).

A nomenclatura associada a tais regras de transição descreve sua ordem. A contagem de termos precedentes (no exemplo A3 e A4) determina sua ordem. Portanto, nosso exemplo é de segunda ordem.

A escala da base de dados de treinamento e a complexidade do modelo utilizado podem afetar negativamente a eficácia e o desempenho do modelo. Quando a base de dados é muito grande, e o modelo é muito complexo, podem surgir milhares ou até mesmo milhões de regras de transição armazenadas. Essa sobrecarga de informações pode limitar a eficiência do modelo em identificar as relações significativas nos dados, comprometendo sua capacidade de alcançar resultados precisos e ótimos. É importante encontrar um equilíbrio entre a complexidade do modelo e a escala da base de dados para obter um modelo eficaz e com bom desempenho.

Figura 8 - Exemplo gráfico de regras de transição.



Fonte: SILVA (2018)

Na figura 8 é possível verificar os conjuntos *fuzzy* particionados à nossa esquerda e as regras de transição plotadas no gráfico. Várias regras de transição podem ser extraídas deste gráfico. São elas:

A1, A2 -> A2

A1, A2, A3, A4 -> A4

A1, A2, A3, A4, A5 -> A4

A1, A2, A3, A4, A5 -> A5

A1, A2, A3, A4, A5 -> A7

A7, A8 -> A7

A7, A8 -> A8

3.5 Número de partições

O número de partições nada mais é do que a própria divisão da base de dados em grupos, o tamanho de cada divisão varia conforme o método de particionamento.

Neste trabalho serão testados dois hiperparâmetros, método de particionamento e número de partições, para execução de cada um dos três modelos propostos.

Existem métodos que já podem empregar uma busca de parâmetros para definir o número de partições através de metaheurísticas (HUARNG, 2001). Porém, não seria possível comparar com outros modelos que utilizam o número de partições discricionário. No modelo de Huarng de 2001, o número de partições é um dos parâmetros a serem incluídos durante a busca, e não seria possível uma comparação clara com os outros métodos selecionados, nos quais o número de partições é um hiperparâmetro fixo ao longo de toda a execução do modelo.

Além de exigir treinamento, aplicação e verificação da eficiência, tornando um modelo "*post-mortem*", que tem sua aplicação e relevância muito evidentes, mas não para este trabalho que possui uma base de dados curta e com características muito únicas.

O número de partições pode influenciar na eficiência de um modelo, uma vez que verifica as regras de transição entre a mudança de estados. Não existem números de partição ótimos ou uma receita para cada modelo, ou impedimentos para a utilização de qualquer número, porém alguns cuidados devem ser observados na escolha. Por exemplo, na base de dados existe um intervalo total de 6171 valores possíveis (0 à 6170), caso algo muito próximo de 0 seja selecionado, não será possível produzir regras de transição *fuzzy* que reflitam a mudança de estado da série temporal

e, ao contrário, se for utilizado algo muito próximo de 6170, o pertencimento dos intervalos será muito baixo por conjunto, nestas condições os modelos de *FTS* param de executar sua função e começam simplesmente a funcionar como um intermediário dos relacionamentos quantitativos, não criando relacionamentos difusos.

Para a otimização do número de partições, será utilizado um intervalo de 10 a 500 partições para uma avaliação ampla das possibilidades e capacidades de cada modelo.

É possível também avaliar se o um número alto de partições converge para que a predição aconteça entre as associações dos conjuntos *fuzzy* ou através do modelo propriamente dito.

3.6 Métricas e avaliação dos resultados

Para que tenha uma melhor visão da comparação entre os modelos, serão utilizadas métricas comuns para avaliação de séries temporais e uma avaliação global dos resultados obtidos em cada modelo.

O *MSE* (M*ean Square Error* ou Média quadrática do erro) é uma medida de erro amplamente utilizada na análise de séries temporais e modelos preditivos e é calculado segundo a fórmula 2.

$$MSE = \frac{\sum_{i=1}^{n} [previsão(i) - verdadeiro\ (i)]^2}{n} \quad (2)$$

Onde *i* é o valor da iteração, ou passo da base de dados e *n* é a contagem de passos da série temporal.

Existe ainda uma derivação do *MSE,* que nada mais é que a raiz quadrada da própria métrica, à esta variante se dá o nome de *RMSE*, *Root Mean Square error*, definido pela fórmula 3.

$$RMSE = \sqrt{MSE} \quad (3)$$

A avaliação da eficiência de cada resultado obtido neste trabalho será feita através de *RMSE (Root Mean Squared Error*), pois seu resultado é inequívoco e comparável entre os diferentes modelos de previsão (BOSE, 2019). O *RMSE* é utilizado como função de avaliação para pesquisa de hiperparâmetros, pois ele está na mesma escala das observações (ZHANG, 2020) (HYNDMAN, 2006).

3.7 Experimento e obtenção de resultados

Para que aconteça a execução de todos os modelos com todos os parâmetros descritos na metodologia, é preciso criar uma combinação de hiperparâmetros.

Tabela 2 - Descrição das combinações de parâmetros possíveis

Modelo	**Particionadores**	**Número de Partições**
Song e Chissom	Grid, C-Means, FCM, Entropy	10 à 500 em passos de 5
Cheng	Grid, C-Means, FCM, Entropy	10 à 500 em passos de 5
Sadaei	Grid, C-Means, FCM, Entropy	10 à 500 em passos de 5

Fonte: Elaborado pelo autor (2022).

Ao fatorar todas essas combinações, é possível listar um total de 1185 execuções de modelos e avaliações nos modelos propostos.

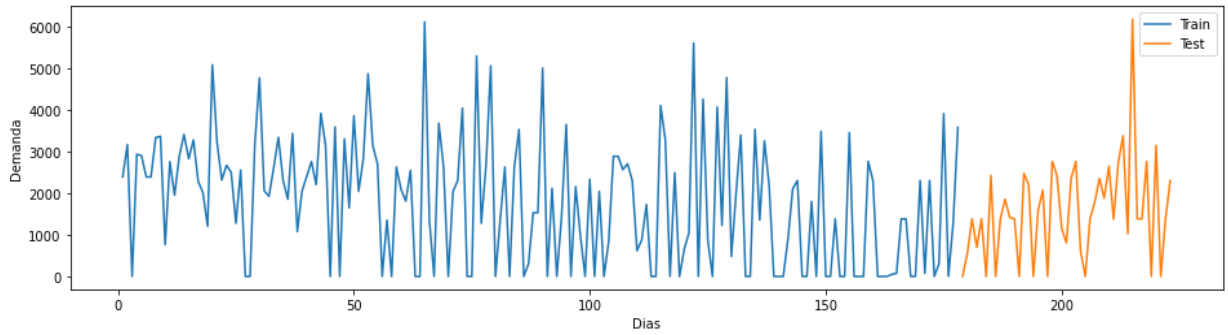
3.8 Preparação da base de dados

Os modelos de *FTS* são modelos de aprendizagem de máquina, nos quais o sistema infere a continuidade da série temporal através de experiências passadas. Para tal, é necessário que haja uma base histórica de dados, daí o nome "aprendizado de máquina". A este histórico, costumeiramente se dá o nome de base de dados de treinamento.

Para que se possa então testar o modelo em dados que não façam parte do aprendizado (o que poderia alterar os resultados de cada modelo), separa-se uma parte (usualmente menor) para a aplicação e posterior aquisição de métricas.

A este trabalho define-se então uma proporção de 80% (178 amostras) para treinamento e 20% (45 amostras) para teste.

Figura 9 - Representação gráfica da divisão da base de dados entre treino e teste.



Fonte: Elaborado pelo autor (2022).

A Figura 9 indica que a primeira porção, representada em azul, é a porção da base de dados do modelo treinará e obterá as regras de transição *fuzzy* e nela os grupos e partições serão criados. A porção amarela representa a base de dados de teste, nesta porção os modelos serão aplicados e as métricas extraídas.

3.9 Ambiente e Execução

Para executar os modelos foram escritos códigos na linguagem *Python* versão 3.10, utilizando-se da biblioteca *PyFTS* (SILVA, 2018). Esta é uma biblioteca que concentra a maior parte dos modelos *FTS* e é possível manipular todos os pontos do processo, da concepção do modelo até a defuzificação do resultado. Esta biblioteca permite tanto a manipulação para confecção e estudo dos modelos quanto a produção em ambiente de execução dedicado.

Além disso, a manipulação dos dados foi feita utilizando a biblioteca pandas (MCKINNEY, 2019), que nos permite trabalhar com uma série de dados em formato de

dataframe, nas quais é possível trazer mais flexibilidade e produtividade acrescentando a funcionalidade de linhas e colunas às estruturas nativas da linguagem (em geral listas e dicionários).

Ao iniciar a execução de cada modelo, os parâmetros modelo *FTS*, Número de partições, método de particionamento, base de dados de treino e de teste são fornecidas à uma função que: (i) executa o particionador; (ii) itera por toda a base de dados de treino coletando todas as informações necessárias para execução (relacionamentos entre os conjuntos) e as armazenam para a execução na a base de dados de teste; (iii) executa a previsão na a base de dados de teste, iterando registro a registro e armazenando para comparação posterior.

3.9.1 Execução dos modelos

Para executar todos os modelos, foi necessário criar uma lista combinatória de todas as 1185 possibilidades, o que exigiu uma análise cuidadosa para garantir que todas as combinações fossem incluídas. Em seguida, uma função foi desenvolvida para executar cada modelo, a fim de obter uma previsão e as métricas de erro (MSE e RMSE). Essas informações foram armazenadas em um *dataframe*, permitindo a posterior análise e comparação dos resultados.

Para garantir a precisão e consistência dos resultados, foram realizadas várias execuções para cada modelo e as métricas foram calculadas com base em várias janelas de tempo. Além disso, foram realizadas análises exploratórias para avaliar a estabilidade do modelo e identificar quais parâmetros tiveram maior impacto na previsão.

Uma vez que todas as execuções foram concluídas e os dados foram coletados, scripts de apoio foram desenvolvidos para a plotagem de gráficos e tabelas suplementares. Essas ferramentas permitiram uma análise mais detalhada dos resultados e uma melhor compreensão das tendências e padrões observados.

3.9.2 Aplicação e comparação do modelo FTS em bancos de dados similares

Após a execução de todos os modelos FTS citados acima, vamos aplicar estes modelos em bancos de dados de incêndios por mês agrupados por estados no Brasil (BRASIL, 2021).

Os dados contidos nesta base foram compilados dos anos de 1998 a 2018 e caracterizados como a soma da contagem de focos de calor por mês, indicando incêndios florestais, separados por estados da federação.

O estado do Acre neste banco de dados contém dados similares ao banco de dados da demanda de *pallets*, ao menos em sua contagem de zeros. O que o torna ideal para uma validação positiva da generalização dos modelos *FTS*, demonstrando sua capacidade de funcionamento em outros bancos de dados diferentes do que ele foi criado para calcular e prever os próximos valores.

3.9.3 Comparação com outro modelo preditivo - *SVR*

A título de comparação e qualificação dos modelos *FTS*, vamos executar outra análise utilizando o modelo *SVR* conforme definido por GIOLA (2021) e DASH (2021).

O trabalho de GIOLA (2021), desenvolve uma metodologia para previsão do desempenho do modelo aplicado a um banco de dados da carga de uma rede de geração e distribuição de energia elétrica, tal como o trabalho de SADAEI (2014).

De posse deste modelo vamos executar a análise utilizando SVR no banco de dados da demanda de produção de *Pallets* e também para o banco de dados de incêndios por estados da federação (BRASIL, 2021), uma vez que eles contêm uma porcentagem parecida de zeros.

4 Discussão e resultados

4.1 Resultados da aplicação dos modelos FTS no banco de dados da demanda de pallets

Para a avaliação e discussão dos resultados, somente as combinações com os menores índices de erro, segundo o *RMSE* foram selecionados para estudo e estão representados na Tabela 3

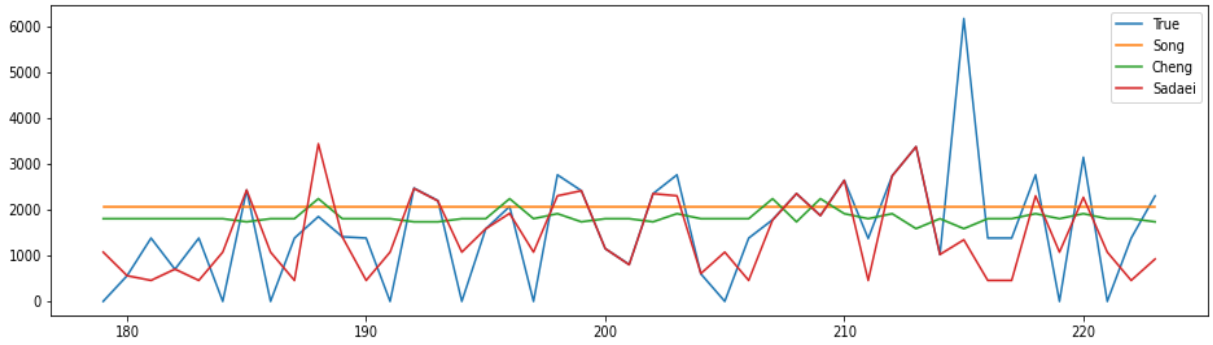
Tabela 3 - Resultados da execução dos parâmetros e modelos propostos.

Modelo	**Particionador**	**N. Partições**	**RMSE**
Sadaei(2014)	Grade	420	1023.79
Cheng(2009)	Entropia	10	1244.66
Song(1993)	FCM	15	1303.40

Fonte: Elaborado pelo Autor (2022).

A Figura 10 mostra graficamente como os modelos foram executados ao longo da base de dados de teste (45 amostras).

Figura 10 - Previsões de cada modelo ao longo da base de dados de teste



Fonte: Elaborado pelo Autor (2022).

Através deste gráfico pode-se avaliar o desempenho de cada modelo ponto a ponto, e algumas avaliações se tornam bastante críticas. A linha azul representa o que de fato aconteceu, sendo a própria base de dados real.

É possível observar que o primeiro modelo desenvolvido, Song (1993) (linha Amarela) não foi capaz de prever um resultado, apresentando uma linha horizontal (como se fosse uma média). Ao analisar seu desempenho com mais detalhes, pode-se verificar que houve sucintas alterações ao longo da base de dados, porém muito aquém de ser um modelo *FTS* viável.

O modelo de Cheng (2009), destacado na linha verde, obteve um desempenho um pouco mais satisfatório e chegou a acompanhar algumas alterações mais notáveis, como entre a amostra 190 e 200, em outros momentos; porém não conseguiu acompanhar as variações exageradas da base de dados, e também seguir a presença dos zeros.

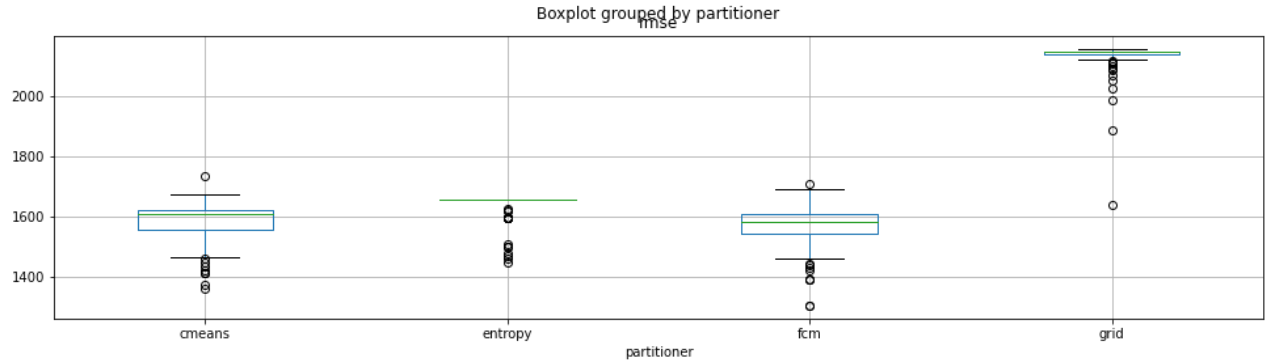
Porém o modelo de Sadaei (2014) (linha vermelha) obteve um desempenho bem superior aos outros modelos previamente descritos. Este modelo de fato acompanha e prediz certos movimentos com precisão (como na amostra 185), chegando a sobrepor a base de dados real por vários períodos (como observado ao redor da amostra 200). Também é importante ressaltar que mesmo durante os movimentos da base de dados a zero, o modelo persegue o real, ainda que não toque o zero.

4.1.1 Resultados do modelo de Song e Chissom (1993)

Como observado anteriormente, o modelo de FTS inicial não apresentou um desempenho satisfatório na base de dados em questão, e todos os resultados de pesquisa de hiperparâmetros apoiaram essa conclusão. Com isso, a pesquisa se concentrou na identificação das causas desse desempenho ruim e no aprimoramento do modelo. O primeiro passo nesse processo foi analisar o particionador utilizado, que é uma etapa crucial na construção de modelos FTS. Na Figura 11, podemos observar a distribuição do RMSE em relação ao erro produzido pelo particionador. Esse tipo de análise permite identificar quais partições contribuem mais para o erro do modelo e, a

partir disso, ajustar o particionador de forma mais precisa. Além disso, outras técnicas de pré-processamento dos dados foram aplicadas para melhorar a qualidade das partições e, consequentemente, o desempenho do modelo.

Figura 11 - *Boxplots* do *RMSE* para cada particionador



Fonte: Elaborado pelo autor (2022).

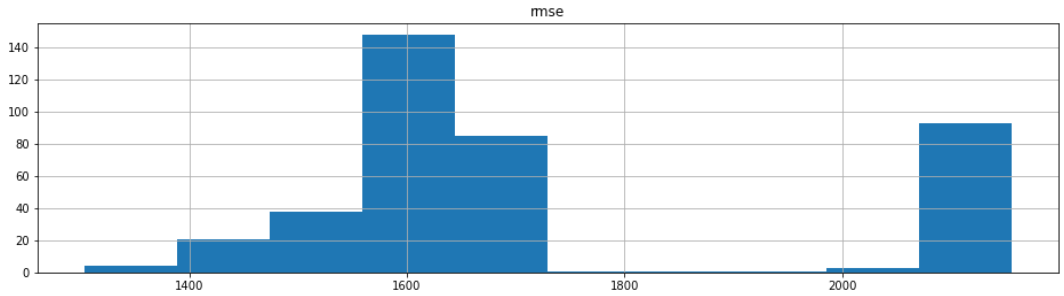
Na Figura 11, pode-se observar que o método de particionamento em grade produziu os piores resultados, e nem mesmo os possíveis outliers inferiores se aproximaram do desempenho dos métodos de particionamento FCM ou C-Means. O método FCM apresentou o melhor resultado, entretanto seu desempenho foi comparável ao do C-Means.

Para uma análise mais detalhada, a distribuição do RMSE foi avaliada no histograma da Figura 12. A análise revela que a maior parte das 396 combinações de hiperparâmetros testados com o modelo de Song ficou em torno de 1600 de RMSE, apresentando poucos resultados até a faixa de 2000 e um número razoável de combinações acima de 2000.

A redução de resultados entre 1700 e 2000 pode ser justificada devido a métodos de particionamento cujo desempenho permita um RMSE abaixo de 2000, como pode ser verificado no caso do método em grade na Figura 11.

Entre os métodos de particionamento elencados (C-Means, Entropia, FCM e Grade), o único método em que não há nenhum dispositivo para melhorar a eficiência do particionamento segundo a distribuição do conjunto é o método em grade.

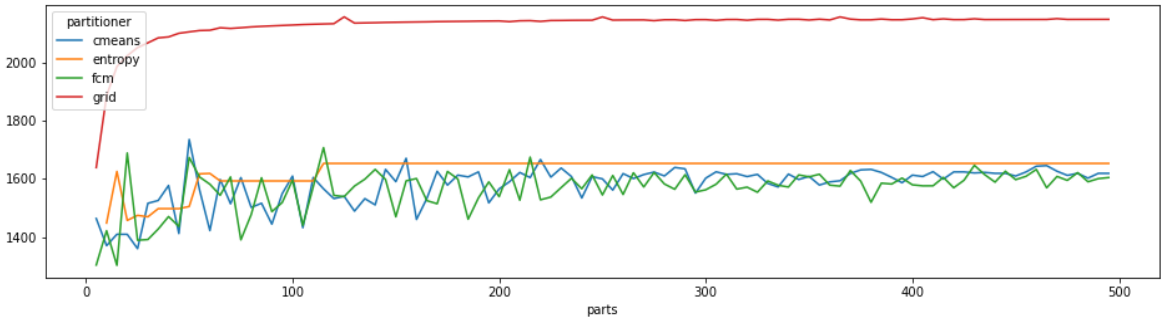
Figura 12 - Histograma da distribuição de *RMSE* no modelo de Song



Fonte: Elaborado pelo autor (2022).

Na figura 13 é possível avaliar o desempenho de cada método de particionamento *versus* o número de partições (que variam de 10 a 500). É possível analisar os quatro métodos de particionamento utilizados no trabalho (*C-Means*, Entropia, *FCM*, *Grid*) e avaliar o *RMSE* de cada um ao longo do intervalo de número de partições variando de 10 a 500.

Figura 13 - Desempenho de cada método de particionamento pelo número de partições



Fonte: Elaborado pelo autor (2022).

O método de Grid (intervalos fixos) apresenta um comportamento inadequado para uma boa eficiência no particionamento de dados. A medida que o número de partições aumenta, o erro cresce exponencialmente, o que justifica a baixa quantidade de amostras com RMSE entre 1600 e 2000, como observado na distribuição da Figura

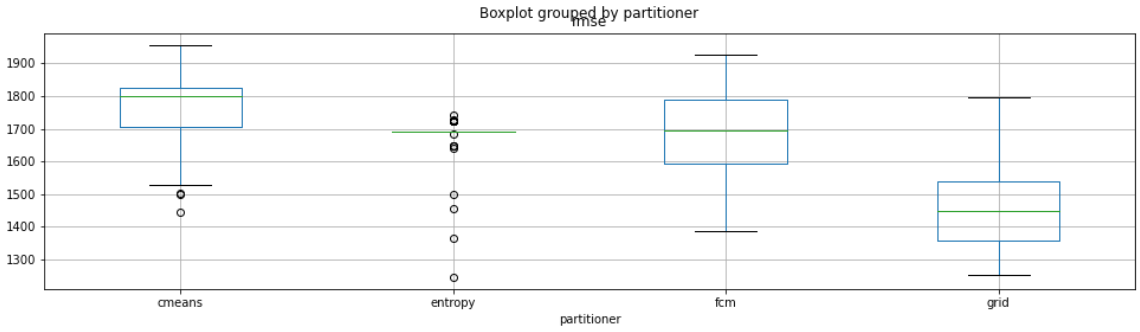
10. Já o método de entropia, possui pouca variação na eficiência após 100 partições, uma vez que sua natureza é reduzir o número de partições (CHENG, 2006). O método de particionamento C-Means apresenta uma ligeira melhora no desempenho com um número reduzido de partições, estabilizando e equiparando-se ao método FCM conforme o número de partições aumenta. O método FCM, por sua vez, demonstrou uma eficiência superior no modelo de Song, porém sua eficiência diminui com o aumento do número de partições.

4.1.2 Resultados do modelo de Cheng (2009)

Diferentemente do modelo de Song (1993), o modelo de Cheng (2009) atingiu melhores resultados utilizando o método particionador por entropia.

Ao iniciar a pesquisa da eficácia do modelo pelo particionador pode-se observar, Figura 14, a distribuição do *RMSE* pelo erro. Observa-se que o método de entropia produziu os melhores resultados, pois apresentou menor *RMSE*.

Figura 14 - *Boxplots* do *RMSE* para cada particionador



Fonte: Elaborado pelo autor (2022).

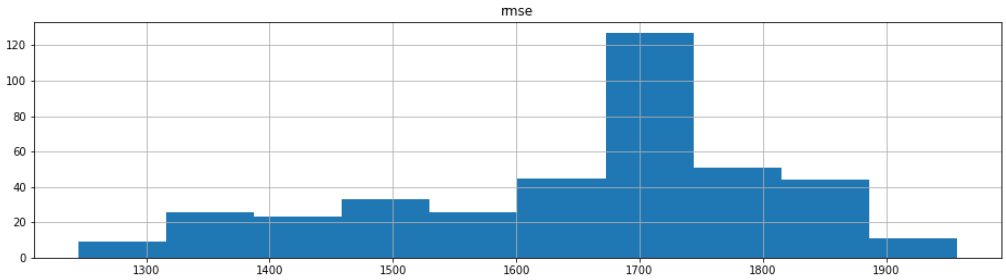
.

Sendo que o melhor resultado obtido por particionador foi o *Entropia,* e o segundo foi o método de grid.

A distribuição do *RMSE* também foi avaliada no histograma da Figura 15, na qual pode-se ver que a maior parte das 396 combinações de hiperparâmetros testados

com o modelo de Cheng, se aproximou de 1700 de *RMSE* e está distribuído de forma razoavelmente normal ao longo de todas as faixas de hiperparâmetros.

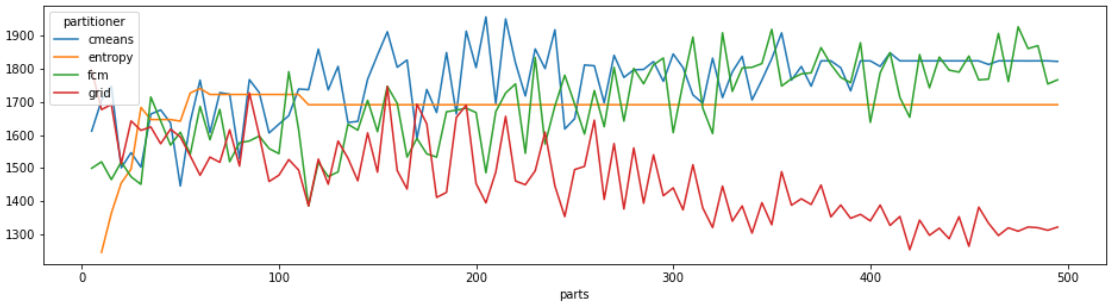
Figura 15 - Histograma da distribuição de *RMSE* no modelo de Cheng



Fonte: Elaborado pelo autor (2022).

Na Figura 16 avalia-se o desempenho de cada método de particionamento *versus* o número de partições (que variam de 10 a 500). É possível analisar os quatro métodos de particionamento utilizados no trabalho (*C-Means*, Entropia, *FCM*, *Grid*) e avaliar o *RMSE* de cada um ao longo do intervalo de número de partições variando de 10 a 500.

Figura 16 - Desempenho de cada método de particionamento pelo número de partições



Fonte: Elaborado pelo autor (2022).

O método de *Grid* (intervalos fixos) aumenta sua eficiência conforme o número de partições aumenta.

O método de entropia apresenta uma estabilidade principalmente após 100 partições, pois é da natureza do método particionador reduzir o número de partições (CHENG, 2006). E sua eficiência é tão grande que somente com 10 partições foi o mais eficiente.

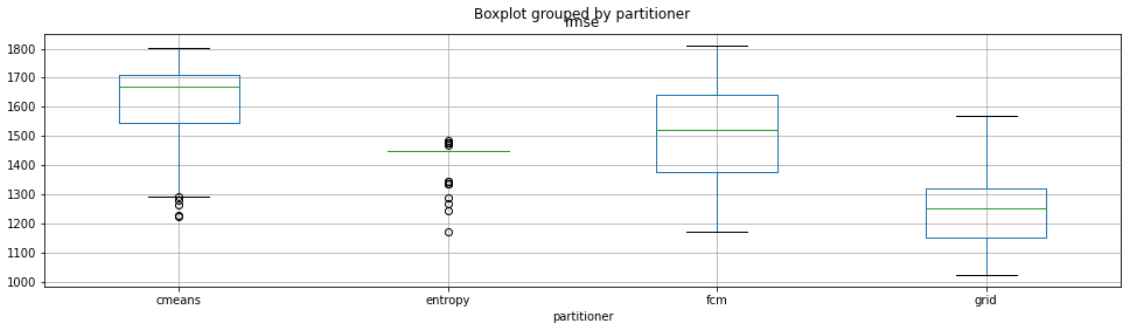
O método de particionamento *C-Means* apresenta uma leve melhora no desempenho com um baixo número de partições, estabilizando e equiparando-se ao método de FCM conforme o número de partições aumenta.

4.1.3 Resultados do modelo de Sadaei (2014)

Diferentemente do modelo de Song (1993), o modelo de Sadaei (2014) atingiu melhores resultados utilizando o método particionador de *Grid*, embora o método *FCM* tenha um RMSE mínimo próximo, os outros quartis estão todos acima dos encontrados no método de grade.

Ao iniciar a pesquisa da eficácia do modelo pelo particionador, na Figura 14 pode-se observar a distribuição do *RMSE* pelo erro.

Figura 14 - *Boxplots* do *RMSE* para cada particionador



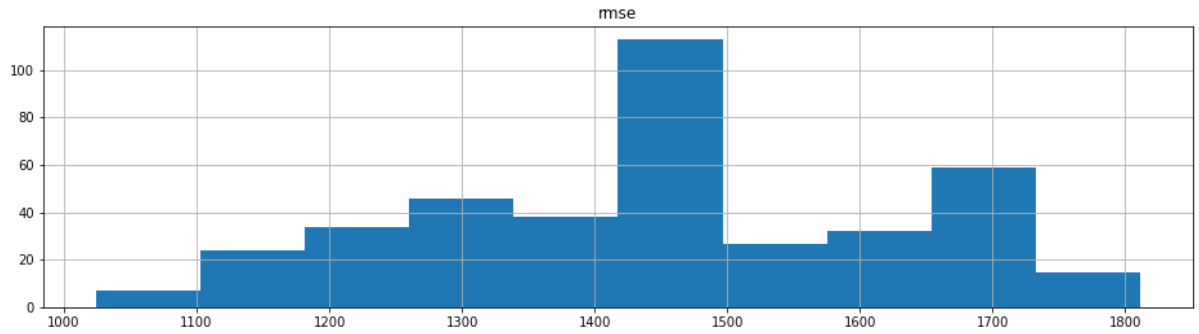
Fonte: Elaborado pelo autor (2022).

Na figura 14, pode-se observar que o método de grade produziu os melhores resultados (em oposição ao modelo de Song).

Sendo que o melhor resultado por particionador foi o *Grid* (seguido pelo método de entropia).

A distribuição do *RMSE* também foi avaliada no histograma da Figura 15.

Figura 15 - Histograma da distribuição de *RMSE* no modelo de Cheng

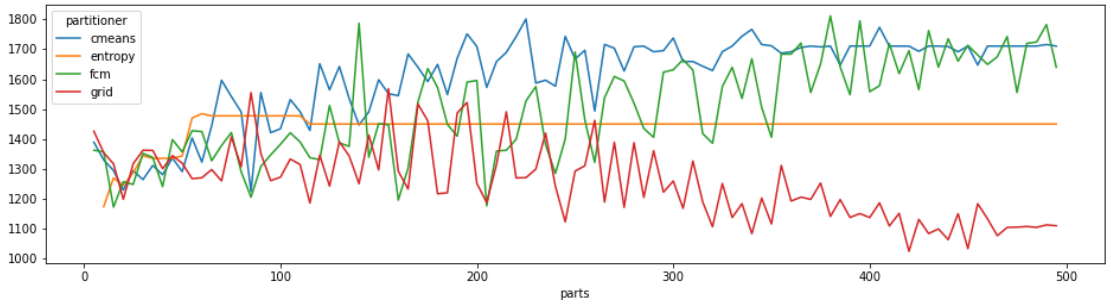


Fonte: Elaborado pelo autor (2022).

Na figura 15 pode-se observar que a maior parte das 396 combinações de hiperparâmetros testados com o modelo de Sedaei girou em torno de 1400 de *RMSE* (já superior aos outros dois) e está distribuído de forma razoavelmente normal ao longo de todas as faixas de hiperparâmetros (com uma pequena elevação na faixa de 1700).

Na figura 16 avalia-se o desempenho de cada método de particionamento *versus* o número de partições (que variam de 10 a 500).

Figura 16 - Desempenho de cada método de particionamento pelo número de partições



Fonte: Elaborado pelo autor (2022).

Na figura 16 acima é possível analisar os quatro métodos de particionamento utilizados no trabalho (*C-Means*, Entropia, *FCM*, *Grid*) e avaliar o *RMSE* de cada um ao longo do intervalo de número de partições variando de 10 a 500.

O método de Grid, que utiliza intervalos fixos, apresenta uma tendência a melhorar seu desempenho à medida que o número de partições é aumentado, um comportamento semelhante ao observado no modelo de Cheng. Por outro lado, o método de entropia, embora tenha uma tendência natural de reduzir o número de partições, demonstrou estabilidade após cerca de 100 partições, o que é uma característica importante a ser considerada.

Vale destacar que, mesmo com apenas 10 partições, o método de entropia foi extremamente eficaz, obtendo resultados comparáveis aos do melhor método avaliado, que foi o Grid com 420 partições.

Quanto ao método de particionamento C-Means, verificou-se uma leve melhoria em seu desempenho com um baixo número de partições, estabilizando e se equiparando ao método de FCM à medida que o número de partições aumenta.

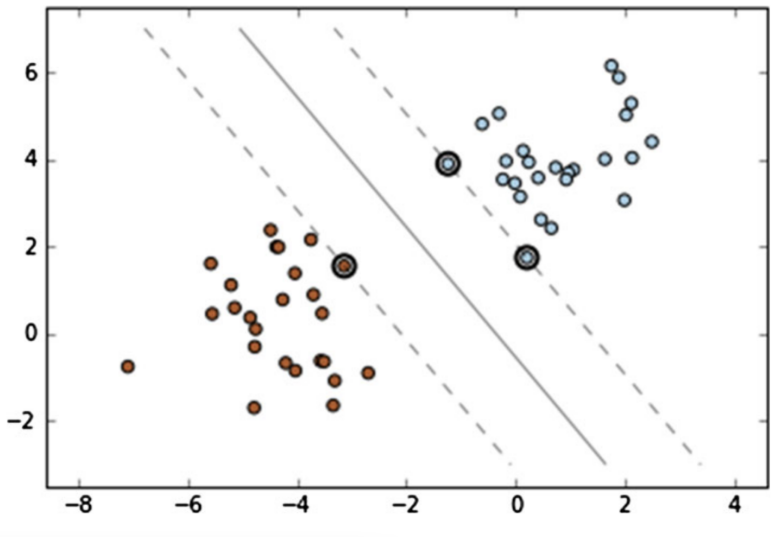
4.2 Resultados do modelo SVR aplicado no banco de dados da demanda diária de pallets

4.2.1 O modelo SVR

O método de *SVR - Support Vector Regression* é um método derivado de *SVM - Support Vector Machines*, que em sua natureza é um método de classificação binária (CHAUHAN, 2019). Estes modelos separam classes com a maior distância entre um agrupamento (*cluster*) de amostras e o limite de borda (também chamado de vetor de suporte - *Support Vector*), na Figura 17 podemos observar os vetores de suporte pontilhados e as amostras separadas pelos mesmos. Os vetores são traçados em torno de uma regressão comumente chamada de *kernel*, que pode ser qualquer função matemática, até mesmo uma regressão linear, em geral o SVR utiliza três kernels diferentes, Regressão, *RBF (Radial Basis Function)* e Polinomial (DASH, 2021). O *SVM*

pode ser estendido a infinitas dimensões, com diferentes abordagens ao seu modelo de treinamento e previsão.

Figura 17 - Separação de grupos através de Vetores de Suporte



Fonte: CHAUHAN (2019).

4.2.2 Relação dos modelos SVR e FTS

Embora o *SVM,* origem do *SVR* seja um modelo de classificação binária, existem métodos muito simples para prever um intervalo contínuo ou semicontínuo, isso é atingido através do aumento do número de categorias de saída e o que o modelo prevê. Muito similar com o conceito de partições dos modelos FTS, introduzido por Song e Chissom em 1993. No entanto os modelos FTS ainda executam a defuzificação e resultam em um valor contido dentro da partição, o que não ocorre nos modelos *SVR*.

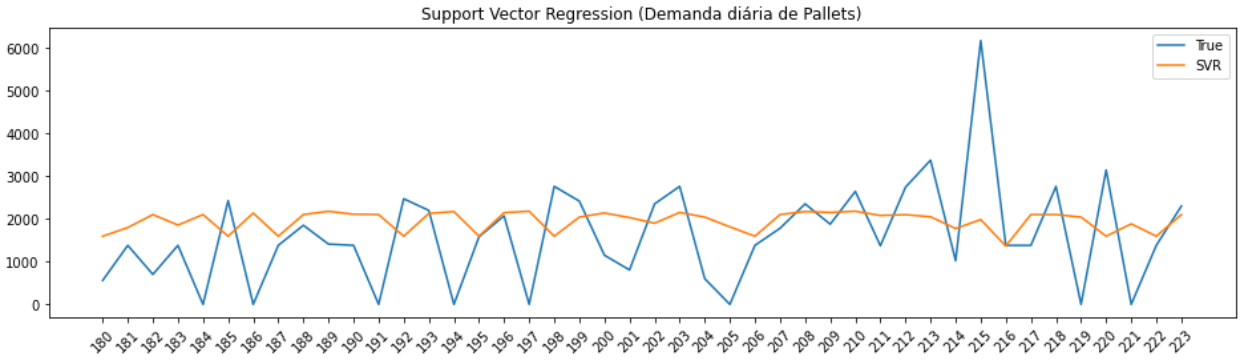
Os modelos *FTS* aqui apresentados criam uma relação de precedência entre os conjuntos *Fuzzy* e a partir das regras de transição executam a previsão do próximo passo, no caso dos modelos *SVR* ele não utiliza uma regra de sucessão de valores, mas sim vetores de classificação que predizem os próximos valores.

4.2.3 A aplicação do modelo SVR no dataset da demanda de pallets

Para a descoberta dos hiperparâmetros ótimos uma busca de grade foi feita em torno dos três hiperparâmetros que os *kernels* utilizam, C - parâmetro de regularização, variando de 0.001 a 10000 em passos de 10, Epsilon - a margem de tolerância antes de definir erros à previsões, variando de 0.001 a 10000 em passos de 10 e o Grau - específico para o *kernel* polinomial, variando de 1 à 5.

Todas essas combinações foram executadas e o melhor resultado obtido aplicando os modelos *SVR* ao banco de dados da demanda de pallets é um *RMSE* de 1619738.24, um erro muito superior ao obtido pelo método *FTS* de Sadaei. Este resultado foi encontrado com os parâmetros C (regularização) = 100, Epsilon = 0.1 , utilizando a função (*kernel*) *RBF* como método para definição dos vetores de suporte.

Figura 18 - Previsões de cada modelo ao longo das amostras de teste do banco de dados de demanda diária de *pallets*



Fonte: Elaborado pelo Autor (2022).

Na figura 18 podemos observar que o modelo SVR obteve um desempenho baixo, com todas as previsões entre 1362 e 2180 *pallets*, e mesmo quando acompanhava os movimentos, era com um atraso (*lag*).

4.2.4 Aplicação dos modelos de previsão no banco de dados de incêndios florestais

Para avaliar a capacidade de generalização dos modelos preditivos FTS e SVR em séries temporais com excessos de zero, é necessário aplicá-los em um novo conjunto de dados. Para este propósito, o banco de dados de incêndios florestais do Brasil, disponível no portal de acesso a dados do governo (BRASIL, 2021), foi escolhido. Este banco de dados contém informações dos focos de incêndios florestais em todos os estados da federação, agregados por mês e separados por estado, abrangendo o período de 1998 a 2018.

A fim de realizar uma comparação mais próxima do banco de dados inicial (demanda de pallets), o estado do Acre foi escolhido para análise devido ao alto volume de zeros presentes em sua composição.

Serão utilizados os modelos com melhor desempenho encontrado no banco de dados da demanda de pallets para avaliar sua capacidade de generalização em um cenário diferente e desafiador, com grande presença de zeros.

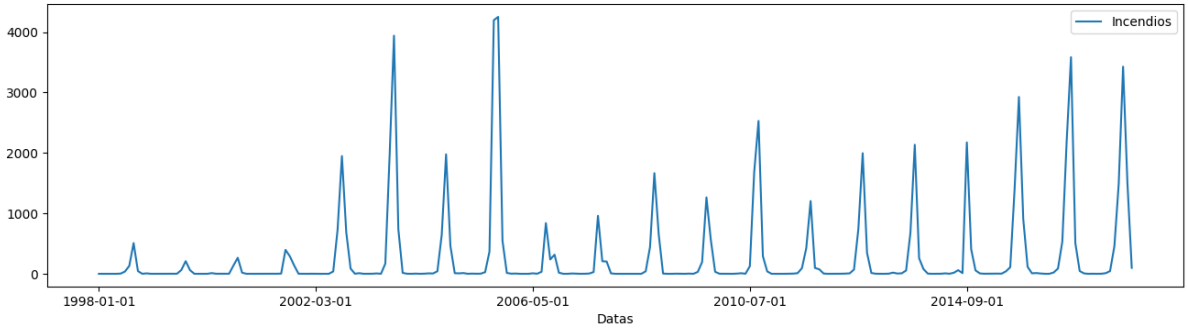
A partir desta análise, será possível verificar se os modelos apresentam resultados satisfatórios mesmo em um contexto de alta heterogeneidade e dificuldade de previsão.

4.2.4.1 Caracterização do banco de dados de incêndios florestais

O banco de dados foi obtido através do Sistema Nacional de Informações Florestais e está disponível no portal de acesso à dados do Brasil (BRASIL, 2021).

Os mesmos procedimentos de preparação adotados no primeiro banco de dados (demanda de pallets) foram aplicados neste banco de dados, na Figura 19 podemos verificar um gráfico dos dados.

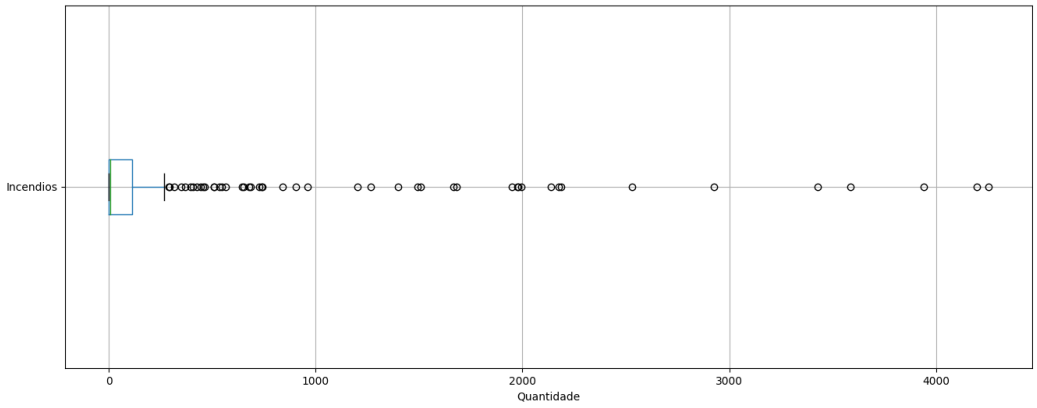
Figura 19 - Dados de incêndios florestais



Fonte: BRASIL (2021).

Podemos observar que o banco de dados tem uma amplitude muito elevada, porém uma diferença observada em relação ao banco de dados da demanda diária de pallets é uma recorrência facilmente detectada, embora sempre retorne ao zero, os incêndios ocorrem mais em determinados meses do ano, como ilustrado pela Figura 20.

Figura 20 - Boxplot do banco de dados de incêndios florestais



Fonte: BRASIL (2021).

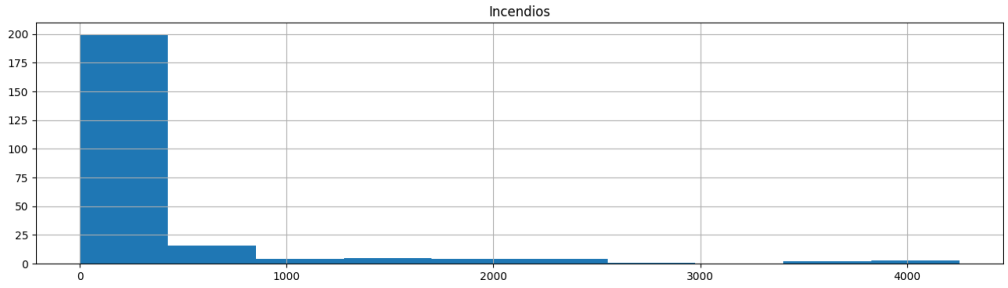
Tabela 4 - Características da série temporal, incêndios florestais

Meses	Média	Desvio Padrão	Min.	1o Quartil	Mediana	3o Quartil	Máximo
239	2126	724.9	0	0	5	112	4253

Fonte: BRASIL (2021).

Embora o cenário seja semelhante ao banco de dados da demanda diária de pallets, os pontos de máxima são muito isolados e mantêm a média e a concentração próximas de zero. A Tabela 4 demonstra a concentração de dados próxima a zero, com o 3o quartil contendo apenas 112 amostras.

Figura 21 - Histograma da distribuição de incêndios



Fonte: BRASIL (2021).

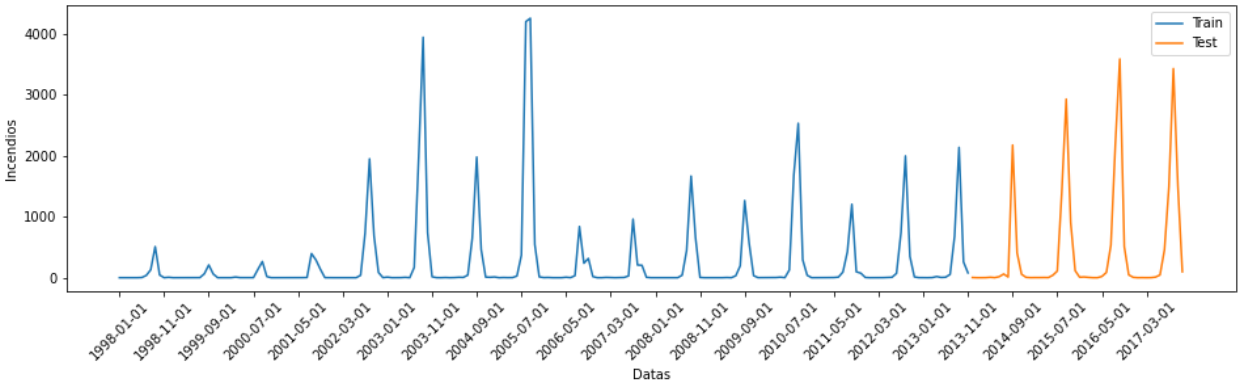
No histograma da Figura 21 podemos observar a distribuição dos eventos e como eles se concentram próximos ao zero, os dados estão muito esparsos da faixa de 1000 ao máximo de 4253 eventos em um mês.

Essa concentração é inerente ao objeto que se refere este banco de dados, em estações do ano onde existe uma frequência maior de chuvas (que podem se extender por vários meses), não acontecem incêndios, limitando a contagem à zero.

Apesar destas diferenças fundamentais entre os bancos de dados estudados, vamos usar este banco de dados para comparação, pois séries temporais com excessos de zero são escassas na literatura e de acesso público.

O banco de dados será dividido na proporção de 80% para o treino e 20% para os testes, da mesma forma como a série de demanda diária de *pallets* foi dividida. Como podemos observar na figura 22.

Figura 22 - Representação gráfica da divisão da base de dados entre treino e teste do banco de dados de incêndios florestais



Fonte: Elaborado pelo Autor (2022).

4.2.4.2 Aplicação dos modelos FTS no banco de dados de incêndios florestais

Para demonstrar a aplicabilidade dos modelos *FTS* apresentados neste trabalho a mesma busca de parâmetros estabelecida no item 3.7 acima foi executada neste banco de dados, após a execução das 1185 execuções obtemos os resultados observados na tabela 5:

Tabela 5 - Resultados da execução dos parâmetros e modelos propostos aplicados ao banco de dados de incêndios florestais

Modelo	**Particionador**	**N. Partições**	**RMSE**
Sadaei(2014)	Grade	485	105.63
Cheng(2009)	Grade	485	111.7
Song(1993)	Grade	10	379

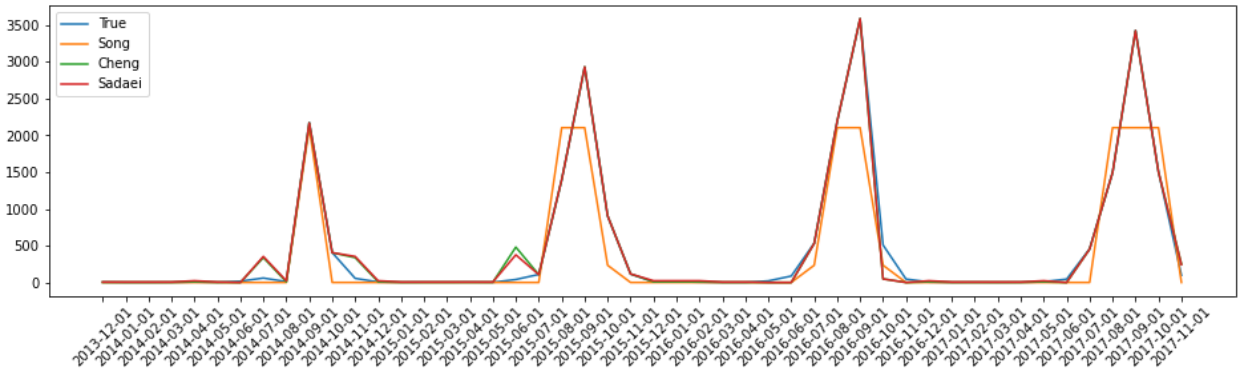
Fonte: Elaborado pelo Autor (2022).

Novamente, o modelo de Sadaei obteve o menor erro, como foi observado no banco de dados da demanda diária de pallets. No entanto, há uma diferença fundamental entre a execução dos modelos nos dois bancos de dados: a presença de

somente um particionador entre os melhores resultados. O particionador de grade obteve os melhores resultados nos três modelos, o que é notável, já que é o único particionador que não executa nenhum tipo de meta-heurística. Esse resultado sugere que a estratégia de particionamento de grade é bastante eficiente para previsões precisas em séries temporais com excessos de zeros, mesmo sem a utilização de meta-heurísticas.

Para avaliar as previsões dos modelos selecionados como melhores, foram realizados testes no banco de dados de demanda diária de pallets. Na figura 23, é possível observar as previsões ao longo das amostras de teste dos três modelos FTS selecionados como melhores. A partir da figura, é possível notar que o modelo de Sadaei tem a menor variação e o menor erro de previsão em comparação com os outros modelos selecionados. Além disso, as previsões de todos os modelos parecem ser bastante precisas e seguir a tendência geral dos dados observados.

Figura 23 - Previsões de cada modelo ao longo das amostras de teste do banco de dados de incêndios florestais .



Fonte: Elaborado pelo Autor (2022).

Neste banco de dados, é observada uma melhora no desempenho de todos os modelos, mesmo o modelo de Song obteve uma melhora significativa, porém devido sua característica estática ele não prevê um número maior de eventos do que a média das amostras de teste, causando o platô na região de 2126 eventos observados.

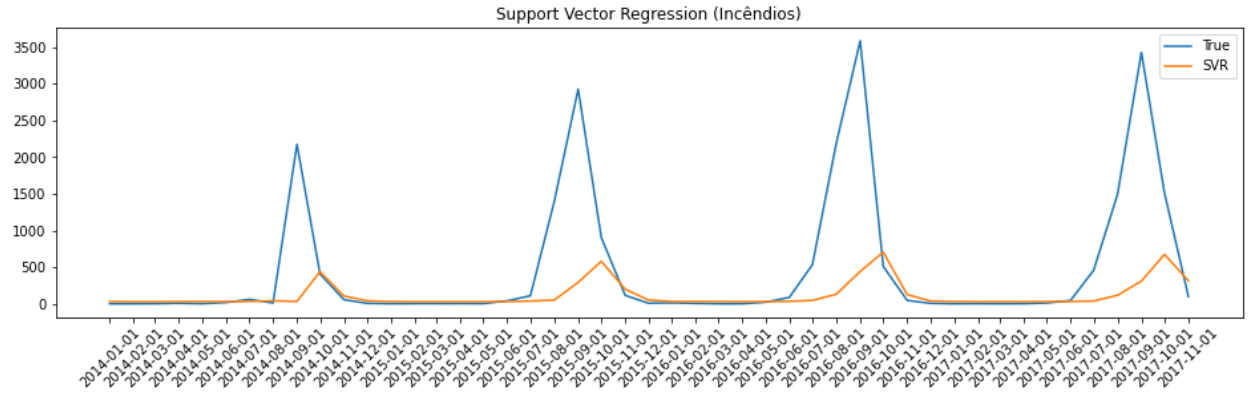
Os modelos de Cheng e Sadaei apresentaram uma capacidade muito elevada de prever os eventos a ponto de confundir as suas previsões com a linha de observações.

4.2.4.3 Aplicação do modelo SVR no banco de dados de incêndios florestais

O modelo SVR foi também aplicado no banco de dados de incêndios florestais utilizando a metodologia descrita no item 4.2.3 acima.

Para aplicação no banco de dados de incêndios florestais os melhores parâmetros encontrados após a busca foram C (regularização) = 100, Epsilon = 30 , utilizando a função (*kernel*) linear como método para definição dos vetores de suporte. O resultado obtido foi um *RMSE* de 860781.6, ordens de magnitude superior aos métodos FTS, na figura 24.

Figura 24 - Previsões de cada modelo ao longo das amostras de teste do banco de dados de incêndios florestais .



Fonte: Elaborado pelo Autor (2022).

5 Comentários finais

Os três modelos investigados abarcam mais de duas décadas de evolução do *FTS*, desde sua incepção em 1993 por Song e Chissom até o modelos de Sadaei em 2014, com resultados que mostram a evolução da técnica com um a base de dados limitada (poucas amostras) e com excessos de zero. Estas duas características são bastante limitantes em modelos preditivos de séries temporais, porém devido às características de *soft-computing* dos modelos de *FTS* é possível lidar com este tipo de situação.

O excesso de zeros impõe ainda mais desafios aos modelos tradicionais (ARIMA, SARIMA, etc) e também aos modelos *FTS*, porém estes conseguem adequar estes problemas de melhor forma.

Para aplicação de modelos de Machine Learning modernos como LSTM, RNN entre outros, o volume de dados disponíveis para treino é demasiadamente baixo.

Durante a análise e execução, fica claro que o modelo de Sadaei possui desempenho satisfatório em um ambiente de produção no qual pode ser utilizado para previsões do dia-a-dia, orientando decisões na produção de *pallets* de madeira. E, naturalmente, caso o modelo seja inserido em um ambiente de produção computacional, haverá mais material de treino, tendendo a aumentar o desempenho do modelo.

Toda a análise deste trabalho foi feita objetivando a generalização dos modelos, ou seja, nenhum tipo de transformação ou ação específica para o caso da demanda diária de *pallets*, gerando uma análise que poderia ser aplicada em qualquer tipo de série temporal, sendo executadas somente a aplicação e otimização (busca por melhores hiperparâmetros) dos modelos. Para demonstrar esta capacidade, foi executada a análise utilizando os modelos FTS em um banco de dados de incêndios florestais, dados oficiais disponibilizados pelo Governo Brasileiro através de seu portal de dados. Este banco de dados contém informações de 20 anos de incêndios florestais, Janeiro de 1998 à Novembro de 2018, acontecidos em unidades da federação agrupados por mês. O estado do Acre contém uma contagem de zeros similar ao banco de dados de demanda diária de *pallets*, por isso foi selecionado para a comparação, um resumo dos resultados pode ser observado na Tabela 6, e podemos

concluir que o desempenho é satisfatório também em outros bancos de dados, indicando uma forte generalização dos modelos *FTS*.

A título de validação, outro modelo foi aplicado nos dois bancos de dados e seus resultados também podem ser observados na Tabela 6. O *SVR* não desempenhou bem a previsão, em nenhum dos dois cenários. O *SVR* é um método puramente algébrico, que estabelece limites para que a previsão seja feita, o fato das séries temporais serem univariadas, contam somente com o valor anterior para que próximas previsões, pode limitar a eficácia de muitos modelos que contam com características similares. A natureza caótica do banco de dados de demanda de pallets também cria um problema de previsão, pois nenhuma outra variável pode ser usada para tentar trazer mais informações para a previsão.

Tabela 6 - *RMSE* dos melhores resultados da execução de todos os modelos estudados nos dois bancos de dados

Modelo	**Incendios (BRASIL, 2021)**	**Pallets (EWBANK, 2020)**
FTS Sadaei(2014)	105,63	1023
FTS - Cheng(2009)	111,7	1244
FTS -Song(1993)	379	1303
SVR	860781	1619738

Fonte: Elaborado pelo Autor (2022).

Para avançar nos estudos na área, há diversas ações que podem ser adotadas, como a utilização de modelos híbridos que combinem técnicas de inteligência artificial e aprendizado de máquina, tais como redes neurais e árvores de decisão. Ademais, uma forma de aumentar a eficiência dos modelos é através do uso de técnicas de pré-processamento de dados, como análises e algoritmos de interpolação e manipulação dos dados, que podem tornar o conjunto de dados mais volumoso e criar uma massa de treinamento mais rica em informações. Essas modificações têm o potencial de otimizar a precisão do sistema de previsão e melhorar a performance do

modelo em relação ao conjunto de dados utilizado. Adicionalmente, outros pontos podem ser estudados para a aplicação desses modelos em produção, tais como a eficiência computacional, a eficiência pelo tamanho do corpo amostral, a resiliência e sensibilidade dos dados, dentre outros.

Essas soluções têm o potencial de reduzir o impacto ambiental, minimizando as perdas e a alocação de recursos naturais, como a madeira e a matéria-prima dos pallets, não apenas na empresa citada, mas em toda a indústria que utiliza esses recursos em sua linha de produção. Com essas técnicas, é possível otimizar a produção, torná-la mais eficiente e reduzir o impacto ambiental ao mesmo tempo.

REFERÊNCIAS

BEZDEK, James C.; EHRLICH, Robert; FULL, William. FCM: The fuzzy c-means clustering algorithm. **Computers & geosciences**, v. 10, n. 2-3, p. 191-203, 1984.

BOSE, Mahua; MALI, Kalyani. Designing fuzzy time series forecasting models: A survey. **International Journal of Approximate Reasoning**, v. 111, p. 78-99, 2019.

BRASIL. SISTEMA NACIONAL DE INFORMAÇÕES FLORESTAIS - SNIF. .**Incêndios Florestais - Focos de Calor**: número de focos de calor no brasil, por mês e por estado.. Número de focos de calor no Brasil, por mês e por estado.. 2021. Disponível em: https://dados.agricultura.gov.br/dataset/snif/resource/f192c841-b95f-4e6e-9a32-70db6838e6f1?view_id=8401fe05-5c45-4fb4-ba36-e9c16ce287b1#embed-8401fe05-5c45-4fb4-ba36-e9c16ce287b1. Acesso em: 25 ago. 2022.

CARTER, Craig R.; JENNINGS, Marianne M. Social responsibility and supply chain relationships. **Transportation Research Part E: Logistics and Transportation Review**, v. 38, n. 1, p. 37-52, 2002.

CHATFIELD, Chris. **Time-series forecasting**. Chapman and Hall/CRC, 2000.

CHAUHAN, Vinod Kumar; DAHIYA, Kalpana; SHARMA, Anuj. Problem formulations and solvers in linear SVM: a review. **Artificial Intelligence Review**, v. 52, n. 2, p. 803-855, 2019.

CHENG, Ching-Hsue; CHANG, Jing-Rong; YEH, Che-An. Entropy-based and trapezoid fuzzification-based fuzzy time series approaches for forecasting IT project cost. **Technological Forecasting and Social Change**, v. 73, n. 5, p. 524-542, 2006.

CHENG, Ching-Hsue; CHEN, You-Shyang; WU, Ya-Ling. Forecasting innovation diffusion of products using trend-weighted fuzzy time-series model. **Expert Systems with Applications**, v. 36, n. 2, p. 1826-1832, 2009.

DASH, Ranjan Kumar et al. Fine-tuned support vector regression model for stock predictions. **Neural Computing and Applications**, p. 1-15, 2021.

DOSZYŃ, Mariusz. Statistical determination of impact of property attributes for weak measurement scales. **Real Estate Management and Valuation**, v. 25, n. 4, p. 75-84, 2017.

DUNN, Joseph C. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. 1973.

EWBANK, Henrique et al. Sustainable resource management in a supply chain: a methodological proposal combining zero-inflated fuzzy time series and clustering techniques. **Journal of Enterprise Information Management**, v. 33, n. 5, p. 1059-1076, 2020.

GEEM, Zong Woo; KIM, Joong Hoon; LOGANATHAN, Gobichettipalayam Vasudevan. A new heuristic optimization algorithm: harmony search. **simulation**, v. 76, n. 2, p. 60-68, 2001.

GIOLA, Chiara; DANTI, Piero; MAGNANI, Sandro. Learning curves: A novel approach for robustness improvement of load forecasting. **Engineering Proceedings**, v. 5, n. 1, p. 38, 2021.

GOVINDAN, Kannan et al. Effect of product recovery and sustainability enhancing indicators on the location selection of manufacturing facility. **Ecological indicators**, v. 67, p. 517-532, 2016.

HOFMANN, Erik; RÜSCH, Marco. Industry 4.0 and the current status as well as future prospects on logistics. **Computers in industry**, v. 89, p. 23-34, 2017.

HUARNG, Kunhuang. Effective lengths of intervals to improve forecasting in fuzzy time series. **Fuzzy sets and systems**, v. 123, n. 3, p. 387-394, 2001.

HYNDMAN, Rob J.; KOEHLER, Anne B. Another look at measures of forecast accuracy. **International journal of forecasting**, v. 22, n. 4, p. 679-688, 2006.

KOCAK, Cem; EGRIOGLU, Erol; BAS, Eren. A new deep intuitionistic fuzzy time series forecasting method based on long short-term memory. **The Journal of Supercomputing**, v. 77, n. 6, p. 6178-6196, 2021.

KUMAR, Naresh; SUSAN, Seba. Particle swarm optimization of partitions and fuzzy order for fuzzy time series forecasting of COVID-19. **Applied Soft Computing**, v. 110, p. 107611, 2021.

LAMBERT, Diane. Zero-inflated Poisson regression, with an application to defects in manufacturing. **Technometrics**, v. 34, n. 1, p. 1-14, 1992.

LAMBERT, Douglas M.; COOPER, Martha C.; PAGH, Janus D. Supply chain management: implementation issues and research opportunities. **The international journal of logistics management**, v. 9, n. 2, p. 1-20, 1998.

LITVINENKO, Vladimir et al. Global guidelines and requirements for professional competencies of natural resource extraction engineers: Implications for ESG principles and sustainable development goals. **Journal of Cleaner Production**, p. 130530, 2022.

MCKINNEY, Wes. **Python para análise de dados: Tratamento de dados com Pandas, NumPy e IPython**. Novatec Editora, 2019.

PIRES, Sílvio RI. Gestão da Cadeia de Suprimentos (Supply Chain Management). Conceitos. **Estratégias, Práticas e Casos. S. Paulo: Atlas**, 2004.

SADAEI, Hossein Javedani et al. Short-term load forecasting using a hybrid model with a refined exponentially weighted fuzzy time series and an improved harmony search. **International Journal of Electrical Power & Energy Systems**, v. 62, p. 118-129, 2014.

SEVERIANO, Carlos A. et al. Evolving fuzzy time series for spatio-temporal forecasting in renewable energy systems. **Renewable Energy**, v. 171, p. 764-783, 2021.

SILVA, PCdL et al. pyfts: Fuzzy time series for python. **Belo Horizonte**, 2018.

SONG, Qiang; CHISSOM, Brad S. Forecasting enrollments with fuzzy time series—Part I. **Fuzzy sets and systems**, v. 54, n. 1, p. 1-9, 1993.

YANG, Ching Chiao; WEI, Hsiao Hsuan. The effect of supply chain security management on security performance in container shipping operations. **Supply Chain Management: An International Journal**, 2013.

YOLCU, Ozge Cagcag; ALPASLAN, Faruk. Prediction of TAIEX based on hybrid fuzzy time series model with single optimization process. **Applied Soft Computing**, v. 66, p. 18-33, 2018.

ZADEH, Lotfi A. Fuzzy sets. **Information and control**, v. 8, n. 3, p. 338-353, 1965.

ZHANG, Junfei; LI, Dong; WANG, Yuhang. Toward intelligent construction: Prediction of mechanical properties of manufactured-sand concrete using tree-based models. **Journal of Cleaner Production**, v. 258, p. 120665, 2020.

WANG, Weina. A big data framework for stock price forecasting using fuzzy time series. **Multimedia Tools and Applications**, v. 77, n. 8, p. 10123-10134, 2018.

ANEXO A - Base de dados da demanda de *pallets* diária

Dia	Demanda	Dia	Demanda	Dia	Demanda
1	2385	26	2550	51	2040
2	3162.5	27	0	52	2820
3	0	28	0	53	4860
4	2930	29	3150	54	3140
5	2895	30	4760	55	2701.5
6	2385	31	2055	56	0
7	2385	32	1920	57	1350
8	3330	33	2601	58	0
9	3360	34	3336	59	2625
10	765	35	2300	60	2085
11	2752.5	36	1850	61	1805
12	1950	37	3431	62	2545
13	2885	38	1075	63	0
14	3405	39	2040	64	0
15	2820	40	2392.5	65	6105
16	3272.5	41	2752.5	66	1350
17	2280	42	2202.5	67	0
18	2010	43	3912.5	68	3675
19	1200	44	3142.5	69	2625
20	5074	45	0	70	0
21	3210	46	3585	71	2040
22	2310	47	0	72	2295
23	2665.5	48	3300	73	4035
24	2494	49	1642.5	74	0
25	1275	50	3855	75	0

Dia	Demanda	Dia	Demanda	Dia	Demanda
76	5289	96	0	121	1042
77	1275	97	2155	122	5595
78	2655	98	980	123	0
79	5052	99	0	124	4254
80	0	100	2330	125	875
76	5289	101	0	126	0
77	1275	102	2040	127	4061.5
78	2655	103	0	128	1221
79	5052	104	840	129	4769
80	0	105	2880	130	480
81	1350	106	2880	131	2002
82	2625	107	2560	132	3390
83	0	108	2700	133	0
84	2625	109	2298	134	0
85	3525	110	617.5	135	3530
86	0	111	875	136	1350
87	300	112	1725	137	3250
88	1530	113	0	138	2180
89	1530	114	0	139	0
90	4995	115	4100	140	0
91	0	116	3275	141	0
92	2115	117	0	142	900
93	0	118	2487.5	143	2100
94	1416.5	119	0	144	2300
95	3645	120	650	145	0

Dia	Demanda	Dia	Demanda	Dia	Demanda
146	0	171	75	196	2070
147	1800	172	2300	197	0
148	0	173	0	198	2760
149	3480	174	300	199	2415
150	0	175	3905	200	1150
151	0	176	0	201	805
152	1380	177	1275	202	2350
153	0	178	3575	203	2760
154	0	179	0	204	600
155	3450	180	560	205	0
156	0	181	1380	206	1380
157	0	182	700	207	1780
158	0	183	1380	208	2350
159	2760	184	0	209	1875
160	2300	185	2425	210	2642.5
161	0	186	0	211	1372
162	0	187	1380	212	2738.5
163	0	188	1850	213	3372.5
164	50	189	1410	214	1020.5
165	75	190	1380	215	6170
166	1380	191	0	216	1380
167	1380	192	2470	217	1380
168	0	193	2197.5	218	2760
169	0	194	0	219	0
170	2300	195	1580	220	3143

Dia	Demanda
221	0
222	1380
223	2300

Fonte: EWBANK (2020)

ANEXO B - Base de dados de incêndios florestais de 1998 à 2018 no estado do Acre

Mes	Incendios	Mes	Incendios	Mes	Incendios
janeiro-98	0	janeiro-00	0	janeiro-02	0
fevereiro-98	0	fevereiro-00	0	fevereiro-02	1
março-98	0	março-00	11	março-02	0
abril-98	0	abril-00	1	abril-02	0
maio-98	0	maio-00	1	maio-02	0
junho-98	3	junho-00	1	junho-02	0
julho-98	37	julho-00	1	julho-02	39
agosto-98	130	agosto-00	136	agosto-02	728
setembro-98	509	setembro-00	265	setembro-02	1949
outubro-98	44	outubro-00	18	outubro-02	687
novembro-98	0	novembro-00	0	novembro-02	86
dezembro-98	7	dezembro-00	0	dezembro-02	1
janeiro-99	0	janeiro-01	0	janeiro-03	10
fevereiro-99	0	fevereiro-01	0	fevereiro-03	0
março-99	0	março-01	0	março-03	0
abril-99	0	abril-01	0	abril-03	1
maio-99	0	maio-01	0	maio-03	6
junho-99	0	junho-01	1	junho-03	0
julho-99	1	julho-01	3	julho-03	168
agosto-99	63	agosto-01	396	agosto-03	1976
setembro-99	209	setembro-01	290	setembro-03	3942
outubro-99	60	outubro-01	137	outubro-03	740
novembro-99	0	novembro-01	1	novembro-03	15
dezembro-99	0	dezembro-01	0	dezembro-03	1

Mes	Incendios	Mes	Incendios	Mes	Incendios
janeiro-04	0	janeiro-06	4	janeiro-08	0
fevereiro-04	3	fevereiro-06	0	fevereiro-08	0
março-04	0	março-06	0	março-08	0
abril-04	2	abril-06	0	abril-08	0
maio-04	7	maio-06	8	maio-08	0
junho-04	5	junho-06	1	junho-08	0
julho-04	42	julho-06	33	julho-08	41
agosto-04	645	agosto-06	839	agosto-08	445
setembro-04	1978	setembro-06	237	setembro-08	1666
outubro-04	461	outubro-06	316	outubro-08	652
novembro-04	10	novembro-06	18	novembro-08	4
dezembro-04	7	dezembro-06	0	dezembro-08	0
janeiro-05	12	janeiro-07	0	janeiro-09	0
fevereiro-05	0	fevereiro-07	5	fevereiro-09	2
março-05	3	março-07	2	março-09	1
abril-05	1	abril-07	0	abril-09	0
maio-05	2	maio-07	1	maio-09	3
junho-05	27	junho-07	4	junho-09	0
julho-05	368	julho-07	29	julho-09	31
agosto-05	4198	agosto-07	960	agosto-09	194
setembro-05	4253	setembro-07	208	setembro-09	1265
outubro-05	547	outubro-07	203	outubro-09	565
novembro-05	14	novembro-07	7	novembro-09	33
dezembro-05	2	dezembro-07	0	dezembro-09	1

Mes	Incendios	Mes	Incendios	Mes	Incendios
janeiro-10	1	janeiro-12	0	janeiro-14	0
fevereiro-10	0	fevereiro-12	0	fevereiro-14	0
março-10	0	março-12	1	março-14	1
abril-10	3	abril-12	1	abril-14	7
maio-10	9	maio-12	3	maio-14	1
junho-10	1	junho-12	7	junho-14	17
julho-10	126	julho-12	71	julho-14	60
agosto-10	1682	agosto-12	739	agosto-14	11
setembro-10	2531	setembro-12	1996	setembro-14	2175
outubro-10	292	outubro-12	348	outubro-14	406
novembro-10	39	novembro-12	13	novembro-14	56
dezembro-10	0	dezembro-12	1	dezembro-14	6
janeiro-11	0	janeiro-13	0	janeiro-15	1
fevereiro-11	0	fevereiro-13	0	fevereiro-15	2
março-11	0	março-13	2	março-15	2
abril-11	2	abril-13	19	abril-15	3
maio-11	3	maio-13	4	maio-15	2
junho-11	10	junho-13	8	junho-15	40
julho-11	93	julho-13	54	julho-15	109
agosto-11	425	agosto-13	679	agosto-15	1397
setembro-11	1204	setembro-13	2136	setembro-15	2928
outubro-11	97	outubro-13	258	outubro-15	905
novembro-11	74	novembro-13	79	novembro-15	115
dezembro-11	4	dezembro-13	3	dezembro-15	8

Mes	Incendios
janeiro-16	12
fevereiro-16	5
março-16	0
abril-16	0
maio-16	21
junho-16	87
julho-16	533
agosto-16	2188
setembro-16	3586
outubro-16	509
novembro-16	46
dezembro-16	6
janeiro-17	0
fevereiro-17	1
março-17	0
abril-17	1
maio-17	10
junho-17	45
julho-17	457
agosto-17	1493
setembro-17	3429
outubro-17	1508
novembro-17	98

Fonte: BRASIL (2021)