UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Previsão de séries temporais econômicas usando redes neurais caóticas

Victor Henrique Gonçalves

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Victor Henrique Gonçalves

Previsão de séries temporais econômicas usando redes neurais caóticas

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. João Luís Garcia Rosa

USP – São Carlos
Janeiro de 2018

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

G635p Gonçalves, Victor Henrique
 Previsão de séries temporais econômicas usando
redes neurais caóticas / Victor Henrique Gonçalves;
orientador João Luís Garcia Rosa. -- São Carlos,
2018.
 79 p.

 Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2018.

 1. Redes Neurais. 2. Conjuntos K. 3. Previsão de
Séries Temporais. 4. Aprendizado de Máquina. 5.
Mineração de Dados. I. Rosa, João Luís Garcia,
orient. II. Título.

Victor Henrique Gonçalves

Forecasting economic time series using chaotic neural networks

Master thesis submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP, in partial fulfillment of the requirements for the degree of the Master Program in Computer Science and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. João Luís Garcia Rosa

USP – São Carlos
January 2018

Este trabalho é dedicado ao Prof. Dr. Claudio Kirner, que ainda no início da minha adolescência, me incentivou a seguir a carreira de Ciência da Computação, e à minha mãe e à minha irmã, que estiveram ao meu lado nos momentos mais difíceis.

AGRADECIMENTOS

Ao Prof. Dr. João Luís Garcia Rosa, por ter aceitado me orientar durante todo o desenvolvimento desta pesquisa.

Ao Prof. Dr. Ricardo José Gabrielli Barreto Campello, por ter cedido o computador usado no desenvolvimento deste projeto.

Ao Prof. Dr. André Carlos Ponce de Leon Ferreira de Carvalho e ao Prof. Dr. Rodrigo Fernandes de Mello, por terem cedido os monitores usados nesta pesquisa.

Ao Prof. Dr. Aquiles Elie Guimarães Kalatzis, pela contribuição com conceitos de economia usados nesta dissertação.

Ao Ms. Antonio Rafael Sabino Parmezan, por sua contribuição na área de previsão de séries temporais, cujos conceitos e experimentos foram aproveitados nesta dissertação.

Ao Instituto Brasileiro de Geografia e Estatística e à Sociedade Ibegeana de Assistência e Seguridade, pelo apoio financeiro.

À minha psiquiatra, Dra. Maria Luiza Gomes de Oliveira e ao meu psicólogo Ms. Oliver Zancul Prado, por terem feito o tratamento de saúde que permitiu a conclusão desta dissertação.

“A alegria está na luta, na tentativa, no sofrimento envolvido e não na vitória propriamente dita.”
(Mahatma Gandhi)

RESUMO

GONÇALVES, V. H. **Previsão de séries temporais econômicas usando redes neurais caóticas**. 2018. 79 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

Esta dissertação descreve a aplicação do KIII, um modelo de rede neural biologicamente mais plausível, para a previsão de séries temporais econômicas. Os conjuntos K são modelos conexionistas baseados em populações de neurônios e foram usados em muitas aplicações de aprendizado de máquina, incluindo previsões de séries temporais. Nesta dissertação, este método foi aplicado ao IPCA, um índice de preços ao consumidor brasileiro pesquisado pelo IBGE em 13 regiões metropolitanas. Os valores abrangem o período de agosto de 1994 a junho de 2017. Os experimentos foram realizados utilizando quatro modelos não-paramétricos (KIII, kNN contínuo, RNAs clássicas e SVM) e seis métodos paramétricos: ARIMA, SARIMA, Médias Móveis, SES, Holt, Holt-Winters Aditivo e Holt-Winters Multiplicativo. A métrica estatística RMSE foi utilizada para comparar o desempenho dos métodos. Os conjuntos KIII de Freeman funcionaram bem como um filtro, melhorando o desempenho do método, mas não foram um bom método de previsão, sendo superado, na maior parte dos experimentos, por outros métodos de previsão de séries temporais. Esta dissertação contribui com o uso de modelos não paramétricos para prever a inflação em um país em desenvolvimento.

Palavras-chave: Séries Temporais, Conjuntos K, Previsão, Métodos Não Paramétricos, Métodos Paramétricos.

ABSTRACT

GONÇALVES, V. H. **Forecasting economic time series using chaotic neural networks**. 2018. 79 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

This thesis describes the application of KIII, a biologically more plausible neural network model, for forecasting economic time series. K-sets are connectionist models based on neural populations and have been used in many machine learning applications, including time series prediction. In this thesis, this method was applied to IPCA, a Brazilian consumer price index surveyed by IBGE in 13 metropolitan areas. The values ranged from August 1994 to June 2017. Experiments were performed using four non-parametric models (KIII, continuous kNN, classical ANN, and SVM) and four parametric methods: ARIMA, SARIMA, Moving Average, SES, Holt, Additive Holt–Winters, and Multiplicative Holt–Winters. The statistical metric RMSE was used to compare methods performance. Freeman’s KIII sets worked well as filter, improving method performance, but it was not a good prediction method, and was overcome in most experiments by other time series prediction methods. This thesis contributes with the use of non-parametrics models for forecasting inflation in a developing country.

Keywords: Time Series, K-Sets, Forecasting, Non-Parametric Methods, Parametric Methods.

LISTA DE ILUSTRAÇÕES

Figura 1 – Hierarquia dos Conjuntos K do Conjunto KO ao Conjunto KIV.	27
Figura 2 – Topologia do Conjunto KI com Conexões Inibitórias.	29
Figura 3 – Estrutura de um Conjunto KII com Dois Conjuntos KO Excitatórios e Dois Conjuntos KO Inibitórios.	29
Figura 4 – Representação do Conjunto KIII Baseada no Sistema Olfatório	31
Figura 5 – Arquitetura do Perceptron	45
Figura 6 – Arquitetura de uma MLP com Duas Camadas Ocultas	47
Figura 7 – Hiperplano de Separação Ótima e Seus Hiperplanos de Suporte	48
Figura 8 – Índice Nacional de Preços ao Consumidor Amplo	50
Figura 9 – Métodos Vencedores	68
Figura 10 – Métodos Não Paramétricos Vencedores	71
Figura 11 – Métodos Paramétricos Vencedores	71

LISTA DE TABELAS

Tabela 1 – Pesos Utilizados nos Conjuntos KII de Cada Camada do Conjunto KIII . . .	35
Tabela 2 – Configuração Padrão Utilizada Como Valor Inicial Para Pesos Laterais das Três Camadas do Conjunto KIII	35
Tabela 3 – Configuração das Conexões Entre Camadas do Conjunto KIII	36
Tabela 4 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Uma Etapa à Frente	69
Tabela 5 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Uma Etapa à Frente	69
Tabela 6 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Duas Etapas à Frente	69
Tabela 7 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Duas Etapas à Frente	69
Tabela 8 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Três Etapas à Frente	69
Tabela 9 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Três Etapas à Frente	69
Tabela 10 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Quatro Etapas à Frente	70
Tabela 11 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Quatro Etapas à Frente	70
Tabela 12 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Cinco Etapas à Frente	70
Tabela 13 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Cinco Etapas à Frente	70
Tabela 14 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Seis Etapas à Frente	70
Tabela 15 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Seis Etapas à Frente	70

LISTA DE ABREVIATURAS E SIGLAS

AEP	<i>Averaged Evoked Potential</i>
AHW	<i>Additive Holt–Winters</i>
AM	<i>Amplitude Modulation</i>
ANN	<i>Artificial Neural Networks</i>
AON	<i>Anterior Olfactory Nuclei</i>
AR	<i>AutoRegressive</i>
ARIMA	<i>AutoRegressive Integrated Moving Average</i>
CSV	<i>Comma Separated Values</i>
EDO	Equações Diferenciais Ordinárias
EEG	Eletroencefalograma
Holt	Método Linear de Holt
IBGE	Instituto Brasileiro de Geografia e Estatística
IPCA	Índice de Preços ao Consumidor Amplo
KIII	Conjunto KIII com algoritmo kNN
kNN	<i>k-Nearest Neighbors</i>
kNN-TSP	<i>kNN - Time Series Prediction</i>
kNN-TSPI	<i>kNN - Time Series Prediction with Invariances</i>
MA	<i>Moving Average</i>
MHW	<i>Multiplicative Holt–Winters</i>
MSE	<i>Mean Squared Error</i>
OB	<i>Olfactory Bulb</i>
PC	<i>Prepiriform Cortex</i>
PDF	<i>Portable Document Format</i>
PEPS	Pesquisa de Especificação de Produtos e Serviços
PLC	Pesquisa de Locais de Compra
POF	Pesquisa de Orçamentos Familiares
RMSE	<i>Root Mean Square Error</i>
SARIMA	<i>Seasonal AutoRegressive Integrated Moving Average</i>
SES	<i>Simple Exponential Smoothing</i>
SGS	Sistema Gerenciador de Séries Temporais do Banco Central – v2.1
SNIPC	Sistema Nacional de Índices de Preços ao Consumidor
SVM	<i>Support Vector Machines</i>

SUMÁRIO

1	INTRODUÇÃO	23
1.1	Contexto	23
1.2	Motivação	24
1.3	Hipótese	24
1.4	Organização do Trabalho	24
2	CONCEITOS	27
2.1	Conjuntos K	27
2.1.1	Conjuntos KO - Pontos Atratores Nulos	28
2.1.2	Conjuntos KI - Pontos Atratores Não Nulos	29
2.1.3	Conjuntos KII - Atratores Ciclo Limite	29
2.1.4	Conjuntos KIII - Atratores Caóticos	30
2.1.4.1	Aprendizagem no Conjunto KIII	32
2.1.4.2	Classificação com o Conjunto KIII	33
2.1.5	Simulador de Conjuntos K	34
2.2	Outros Métodos Usados Para a Previsão de Séries Temporais	36
2.2.1	Métodos Paramétricos	36
2.2.1.1	Médias Móveis	36
2.2.1.2	Suavização Exponencial Simples	37
2.2.1.3	Suavização Exponencial de Holt	38
2.2.1.4	Suavização Exponencial Sazonal de Holt–Winters	38
2.2.1.5	Modelos ARIMA e SARIMA	40
2.2.2	Métodos Não Paramétricos	44
2.2.2.1	Redes Neurais Artificiais Clássicas	45
2.2.2.2	Máquinas de Vetores de Suporte	46
2.2.2.3	k-Vizinhos Mais Próximos	48
2.3	IPCA – Índice de Preços ao Consumidor Amplo	49
3	TRABALHOS RELACIONADOS	53
3.1	Trabalhos Usando Conjuntos K	53
3.2	Trabalhos Sobre Previsões de Séries Temporais	56
4	MATERIAIS E MÉTODOS	61
4.1	Considerações Iniciais	61

4.2	Execução de Métodos Não Paramétricos	61
4.3	Execução de Métodos Paramétricos	64
4.4	Resultados	66
5	CONCLUSÃO	73
5.1	Conclusões	73
5.2	Trabalhos Futuros	73
REFERÊNCIAS		75
GLOSSÁRIO		79

INTRODUÇÃO

1.1 Contexto

Esta dissertação trata da previsão de séries temporais econômicas utilizando uma abordagem conexcionista, os conjuntos K de Freeman. Séries temporais econômicas servem para dois propósitos. O primeiro é fazer previsões futuras, que é o foco desta dissertação. O segundo é prover um meio de apresentar fatos associados com o movimento de séries temporais. Ambos os papéis não são completamente independentes, podendo um modelo que representa as características principais da série também ser um bom modelo de previsão ([HARVEY, 1985](#)). As séries temporais econômicas também servem para o Banco Central planejar a política econômica. O Banco Central do Brasil utiliza vários indicadores econômicos e financeiros para decidir a política econômica, entre eles o nível de ociosidade dos fatores de produção, índice de utilização da capacidade da indústria, taxa de desemprego, medidas de inflação, como o Índice de Preços ao Consumidor Amplo (IPCA) e componentes como preços administrados e alimentação no domicílio, taxa de juros Selic e taxa de câmbio R\$/US\$ ([BANCO CENTRAL DO BRASIL, 2017a](#)). Com as séries temporais econômicas, pode-se também detectar choques econômicos, como os efeitos da política monetária nos indicadores econômicos ([STOCK; WATSON, 2005](#)).

Foi utilizada uma série temporal da atividade econômica, um índice de preços ao consumidor, o IPCA, coletado e divulgado mensalmente pelo Instituto Brasileiro de Geografia e Estatística (IBGE), o instituto de pesquisas estatísticas oficial do governo brasileiro. Os valores da série temporal foram extraídos do Sistema Gerenciador de Séries Temporais do Banco Central – v2.1 (SGS) ([BANCO CENTRAL DO BRASIL, 2017b](#)).

1.2 Motivação

A previsão do índice de preços ao consumidor é uma tarefa bastante complexa porque os índices de inflação brasileiros são altos e voláteis. Antes do Plano Real, eles eram extremamente altos e muito mais voláteis, o que impossibilitava a previsão desses índices. Logo após o Plano Real, a inflação diminuiu e sua volatilidade também, mas ambas ainda permaneceram altas, o que é um cenário ideal para que redes neurais, especialmente os conjuntos K de Freeman, prevessem valores mais realistas que os demais métodos. Somente nos anos mais recentes a volatilidade diminuiu ainda mais, permitindo que métodos de suavização exponencial tivessem um desempenho melhor na previsão dos índices de preços. Mas, ainda sim, essa volatilidade é alta o suficiente para que os conjuntos KIII de Freeman possam ter um bom desempenho nessa tarefa.

1.3 Hipótese

O objetivo desta dissertação foi desenvolver um método capaz de prever séries temporais econômicas mensais da atividade econômica, mais especificamente de um índice de preços ao consumidor, usando os conjuntos K de Freeman e compará-lo com outros métodos de previsão por similaridade, métodos não paramétricos e paramétricos.

A hipótese formulada nesta dissertação foi, portanto, que o uso de conjuntos K de Freeman contribui para previsões mais realistas de séries temporais econômicas em relação a modelos já existentes na literatura porque esses modelos trabalham bem com problemas onde há séries temporais instáveis e difíceis de se preverem, que é o caso da previsão de séries temporais econômicas brasileiras. A aplicação desta técnica no problema desta dissertação pode trazer contribuições para o uso da rede neural conjuntos K em tarefas de predição de séries temporais em geral.

Nesta dissertação, conseguimos comprovar parcialmente a hipótese inicial, uma vez que o conjunto KIII de Freeman funcionou bem como filtro, melhorando o desempenho do método *k-Nearest Neighbors* (kNN) contínuo, porém não foi um bom método de previsão, sendo superado, na maior parte dos experimentos, por outros métodos de previsão de séries temporais.

1.4 Organização do Trabalho

O restante desta dissertação é organizado como se segue.

No capítulo 2, serão apresentados conceitos da literatura necessários ao entendimento desta dissertação, como os conjuntos K de Freeman e o índice de inflação IPCA. No capítulo 3, serão apresentados trabalhos da literatura relacionados à esta dissertação e que contribuíram para seu desenvolvimento, juntamente com uma descrição de sua contribuição. No capítulo 4, serão

apresentados todos os passos para a execução dos experimentos necessários para a comprovação ou refutação da hipótese desta dissertação, bem como seus resultados. No capítulo 5, serão apresentadas as conclusões e considerações finais, bem como ideias para trabalhos futuros.

CONCEITOS

2.1 Conjuntos K

Figura 1 – Hierarquia dos Conjuntos K do Conjunto KO ao Conjunto KIV.

Fonte: Adaptada de [Kozma et al. \(2007\)](#).

Os conjuntos K de Freeman formam uma hierarquia aninhada de modelos da dinâmica de populações de $\sim 10^4$ neurônios com $\sim 10^8$ sinapses. Os conjuntos K de Freeman são modelos mesoscópicos e representam um nível intermediário entre os neurônios microscópicos e a estrutura macroscópica do cérebro. A topologia das conexões é modelada por redes de populações de neurônios excitatórias e inibitórias. A dinâmica é aproximada por partes não lineares de Equações Diferenciais Ordinárias (EDO) ([FREEMAN; ERWIN, 2008](#)), com parâmetros distribuídos para descrever dinâmicas de populações neurais começando de microcolunas a hemisférios ([KOZMA; FREEMAN, 2001](#)).

Nesta dissertação, foi utilizada uma abordagem hierárquica para a neurodinâmica espaço-tempo, baseada em conjuntos K de Freeman. Conjuntos de baixo nível K foram introduzidos por [Freeman \(1975\)](#), que descreveu a neurodinâmica do sistema olfatório dos mamíferos, e receberam esse nome em homenagem a Aharon Katzir-Katchalsky, um dos pioneiros da dinâmica neural,

que estava interessado em aplicar os princípios da termodinâmica em seres vivos, particularmente na relação entre grupos de células (FREEMAN, 1975) (FREEMAN, 1992) (FREEMAN, 2000). Os conjuntos K são modelos multiescala, descrevendo o aumento da complexidade de estruturas dinâmicas e também são especificações topológicas da hierarquia da conectividade em populações de neurônios. A hierarquia dos modelos K começa a ser construída a partir de uma unidade básica (KO). Redes de maior complexidade são construídas a partir dessa unidade básica.

Quando introduzidos, os conjuntos K foram modelados usando um sistema de equações diferenciais ordinárias não lineares (FREEMAN, 1975). A dinâmica dos conjuntos K prediz as ondas oscilatórias que são geradas por populações de neurônios. Conjuntos K descrevem os padrões espaciais da fase e da amplitude das oscilações. Eles modelam campos observáveis da atividade neural compreendendo eletroencefalogramas, potenciais de campo local e magnetoencefalogramas.

2.1.1 Conjuntos KO - Pontos Atratores Nulos

Conjuntos KO representam massas não interativas de neurônios formando microcolunas corticais, a estrutura de colunas dos neurônios do cérebro. Eles são a menor massa de neurônios não interagentes e são os blocos de construção mais básicos para os conjuntos K. Um conjunto KO modela uma população de neurônios de ≈ 10.000 neurônios. O KO modela a integração dendrítica em neurônios médios e uma não linearidade estática sigmoide assimétrica durante a transmissão dos sinais neurais feitas nos axônios. O KO é governado por um ponto atrator com saída nula e permanece em equilíbrio exceto quando perturbado.

Nos conjuntos K originais, KO são descritos por uma equação diferencial ordinária linear dependente de estados de 2ª ordem (Equação 2.1) (FREEMAN, 1975).

$$(a * b) \frac{d^2 P(t)}{dt^2} + (a + b) \frac{dP(t)}{dt} + P(t) = F(t) \quad (2.1)$$

Aqui a e b são constantes de tempo determinadas experimentalmente. $P(t)$ denota a ativação do nó como uma função do tempo. $F(t)$ inclui uma função sigmoide assimétrica $Q(x)$, (veja Equação 2.2), atuando na soma ponderada de ativação dos nós vizinhos e qualquer entrada externa (BELIAEV; KOZMA, 2007).

$$Q(X) = q * \left(1 - \exp \left(\frac{-1}{q(e^x - 1)} \right) \right) \quad (2.2)$$

onde q é um parâmetro que especifica a inclinação e a assíntota máxima da curva.

Figura 2 – Topologia do Conjunto KI com Conexões Inibitórias.

Fonte: Adaptada de [Freeman \(1975\)](#).

2.1.2 Conjuntos KI - Pontos Atratores Não Nulos

Conjuntos KI são o próximo nível na construção da hierarquia dos modelos de populações neurais. Conjuntos KI são a forma mais simples de interação entre populações de neurônios e são feitos pela interação de conjuntos KO, ou excitatórios, ou inibitórios, com retorno positivo. Tipicamente o KI é construído com duas unidades KO, mas um conjunto de KIs do mesmo tipo conectados somente por conexões laterais ainda é uma rede KI. A dinâmica do KI é descrita como a convergência para um ponto fixo não nulo. Se o KI tem uma densidade de conexão suficientemente funcional, ele é capaz de manter um estado não nulo de atividade de fundo por excitação ou inibição mútua. KI opera tipicamente longe do equilíbrio termodinâmico. Sua contribuição crítica é a saída de nível excitatória sustentada. Interações neurais por excitação ou inibição mutualmente estáveis são fundamentais para entender a dinâmica do cérebro.

2.1.3 Conjuntos KII - Atratores Ciclo Limite

Figura 3 – Estrutura de um Conjunto KII com Dois Conjuntos KO Excitatórios e Dois Conjuntos KO Inibitórios.

Fonte: Adaptada de [Li e Kozma \(2003\)](#).

Conjuntos KII consistem de interações excitatória e inibitória de conjuntos KI com

retorno negativo. Eles são formados por um conjunto KI excitatório e um KI inibitório interagindo entre si. O KII é construído a partir de pelo menos dois KIs ou quatro KOs densamente conectados. Exemplos incluem o bulbo olfatório, regiões do hipocampo e o córtex prepiriforme. KII tem quatro tipos de interações: excitatória excitatória, inibitória inibitória, excitatória inibitória e inibitória excitatória. Conjuntos KII são responsáveis pelo surgimento de oscilações ciclo limite devido ao retorno negativo entre populações de neurônios. Transições de pontos atratores para atratores ciclo limite podem ser atingidas através de um nível apropriado de ganho de retorno ou por estímulos de entrada. Análise sistemática tem identificado regiões de estabilidade e de transições entre dinâmicas ciclo limite e ponto fixo. Os conjuntos KI que formam o KII, quando ativos, oscilam em uma frequência característica em resposta a um estímulo de entrada. Esta frequência é vista no Eletroencefalograma (EEG) e no Potencial Evocado Médio (*Averaged Evoked Potential* (AEP)) como componentes oscilatórios e emergem como efeito de uma entrada periódica (como a respiração) ou através da interação com outros conjuntos, com os quais o KII está conectado em níveis maiores (KIII). O comportamento mais interessante do KII surge quando ele é construído por KIs ou KOs com polaridade inversa. As conexões do KII têm pesos fixos e os KII formam osciladores quando essas conexões são configuradas.

2.1.4 Conjuntos KIII - Atratores Caóticos

Enquanto os conjuntos KO, KI e KII estão relacionados a níveis da análise eletrofisiológica, o KIII é o menor nível da análise comportamental e o primeiro nível com aplicações práticas das redes neurais dos conjuntos K (ROSA; PIAZENTIN, 2016). Nesse nível, os padrões de atividade espaço-tempo podem ser chamados sinais neurais e são correlacionados a estímulos externos. Os aromas, por exemplo, são representados por padrões de atividade espaço-tempo em cada conjunto. A hipótese formulada por Freeman (1975) é que esses sinais neurais existem devido à interação da população de neurônios, formando macroestados em conjuntos neurais em níveis mais baixos. Conjuntos KIII são construídos de três camadas de conjuntos KII interagindo entre si. A primeira camada corresponde ao Bulbo Olfatório (*Olfactory Bulb* (OB)) e tem o tamanho da entrada da rede. O OB então conecta-se aos Núcleos Olfativos Anteriores (*Anterior Olfactory Nuclei* (AON)) e o Córtex Prepiriforme (*Prepiriform Cortex* (PC)), a segunda e a terceira camada respectivamente. As camadas AON e PC têm conexões entre si e de volta para OB. Como a oscilação é dispersada na camada PC do KIII, um segundo algoritmo é tipicamente usado para traduzir ou classificar esta saída. Os algoritmos mais comumente usados para esta tarefa são os k-vizinhos mais próximos e análise discriminante linear. A equação para o KIII é:

$$\frac{1}{ab} \left[\frac{d^2 v_i(t)}{dt^2} + (a+b) \frac{dv_i(t)}{dt} + abv_i(t) \right] = \sum_{j \neq i}^N [W_{ij} \times Q(v_j(t), q)] + I_i(t) \quad (2.3)$$

onde v_i é a ativação da i -ésima população de neurônios, N é o número de populações no conjunto, I é o estímulo de entrada externo, W é o vetor de pesos representando a força de conexão entre as populações i e j , t é o tempo e $Q(\bullet)$ é a função de conversão onda para pulso.

Figura 4 – Representação do Conjunto KIII Baseada no Sistema Olfatório

Fonte: Adaptada de [Principe et al. \(2001\)](#).

Exemplos de conjuntos KIII incluem o sistema olfatório e outros córtices sensoriais e o hipocampo com as seções CA1, CA2 e CA3. O KIII já foi usado com sucesso na detecção de faces ([ZHANG et al., 2005](#))([ZHANG et al., 2006b](#))([ZHANG et al., 2006a](#))([CARLOS; ROSA, 2014](#)), navegação de robôs ([HARTER, 2005](#)), reconhecimento de dígitos falados ([LI; ZHANG; FREEMAN, 2007](#))([ZHANG; LI; FREEMAN, 2006a](#)), classificação de estruturas em tecidos ([XIAOMIN et al., 2010](#)), chás ([YANG et al., 2006](#)) e texto ([ZHANG; LI; FREEMAN, 2006b](#)), previsão de séries temporais ([KOZMA; BELIAEV, 2004](#)), imagética motora ([PIAZENTIN; ROSA, 2014](#)), agrupamento ([KOZMA; ROSA; PIAZENTIN, 2013](#))([ROSA; PIAZENTIN, 2016](#)) e na classificação do EEG ([KOZMA; FREEMAN, 2001](#))([KOZMA; FREEMAN, 2002](#))([HU et al., 2006b](#))([HU et al., 2006a](#))([ZHANG et al., 2008](#)). Conjuntos KIII geram oscilações caóticas, aperiódicas e de banda larga como atividade de fundo através de retornos positivos e negativos

entre vários conjuntos KII com frequências incomensuráveis. O aumento no ganho de retorno não linear que é impulsionado pela entrada resulta na desestabilização da atividade de fundo e leva ao surgimento de um padrão de Modulação de Amplitude (*Amplitude Modulation* (AM)) espacial em KIII.

O KIII pode servir como uma memória associativa, codificando dados de entrada em padrões AM de espaço-tempo (KOZMA; FREEMAN, 2001). Memórias caóticas KIII têm várias vantagens quando comparadas a redes recorrentes convergentes: 1. elas produzem memória robusta baseada em relativamente poucos exemplos de aprendizagem mesmo em ambientes ruidosos; 2. a capacidade de codificação de uma rede com um dado número de nós é exponencialmente maior que suas contrapartes convergentes; 3. elas podem recuperar os dados armazenados muito rapidamente, assim como humanos e animais podem reconhecer um padrão aprendido em uma fração de segundo.

2.1.4.1 Aprendizagem no Conjunto KIII

O treinamento no conjunto KIII é não supervisionado e feito por dois processos: a regra do aprendizado Hebbiano, que é usada para ajustar os pesos das conexões excitatórias entre os conjuntos KIII, e um processo de habituação, onde os pesos de conexão são reduzidos gradualmente para introduzir um processo de esquecimento na rede, tornando-a mais resistente aos ruídos nos dados. Este processo ocorre nas conexões laterais excitatórias entre os conjuntos KII da terceira camada PC. Pode-se incluir também um terceiro processo de aprendizagem, que consiste da normalização da atividade dos nós para manter a estabilidade, mas normalmente este processo ocorre somente na parametrização inicial do modelo e não está incluído no treinamento.

Habituação é um processo automático que filtra estímulos irrelevantes confusos, ambíguos e indesejados, reduzindo o impacto do ruído de fundo. O processo de habituação funciona através de um decaimento de pesos lento e constante. A taxa de decaimento do processo de habituação tem que ser menor que a taxa que a regra de Hebb usa para atualizar seus pesos, de modo que somente as conexões não reforçadas pela taxa da regra de Hebb decaiam.

O processo de habituação no conjunto KIII é dado por

$$w = pw \quad (2.4)$$

onde p é a taxa de decaimento e w é uma conexão lateral excitatória entre dois KOs. Em sua forma mais simples, a hipótese de aprendizado Hebbiano é descrita por:

$$\Delta w_{kj}(n) = \eta y_k(n) x_j(n) \quad (2.5)$$

onde Δw_{kj} é o ajuste aplicado ao peso sináptico do neurônio k com sinal x_j pré-sináptico e sinal y_k pós-sináptico e η é uma constante positiva que determina a taxa de aprendizado. Desta forma, a regra de Hebb aumenta exponencialmente os pesos sinápticos, o que leva à saturação da conexão e à perda da capacidade de armazenamento de informação da sinapse. A hipótese de

covariância supera esta limitação, substituindo os sinais pré e pós-sinápticos pelas diferenças em relação às suas respectivas médias $\bar{x}$ e $\bar{y}$ sobre um intervalo de tempo específico:

$$\Delta w_{kj}(n) = \eta(x_j - \bar{x})(y_k - \bar{y}) \quad (2.6)$$

Nos conjuntos K, a regra de Hebb é usada para modificar as conexões entre populações de neurônios mesoscópicas ao invés de neurônio individuais.

Uma vez que modelos K oscilam constantemente, a regra de Hebb tem que ser adaptada para ser usada com eles. Uma das modificações empregadas consiste em calcular a ativação média durante um período de tempo T e então calcular o desvio padrão das oscilações sobre T , σ . Considerando $\bar{\sigma}$ a amplitude de oscilação média do vetor de ativação $\vec{\sigma}$ e w_{pq} o peso da conexão bilateral entre o p-ésimo e o q-ésimo KO, a conexão é reforçada:

$$w_{pq} = w_{pq} + \Delta w_{pq} \quad (2.7)$$

onde

$$\Delta w_{pq} = \begin{cases} \eta(\sigma_p - \bar{\sigma})(\sigma_q - \bar{\sigma}) & \text{se } \sigma_p > \bar{\sigma}, \sigma_q > \bar{\sigma} \\ 0, & \text{caso contrário.} \end{cases} \quad (2.8)$$

Uma vez que o aprendizado Hebbiano ocorre localmente no espaço de estados, ele frequentemente reforça a estabilidade dos padrões aprendidos, reforçando memórias em particular.

2.1.4.2 Classificação com o Conjunto KIII

A classificação com o KIII tem quatro fases. Primeiro, o modelo KIII é parametrizado, inicializando os pesos sinápticos da rede com valores obtidos experimentalmente e atribuindo pequenos valores próximo de zero para as ativações de cada conjunto KO pertencente à rede.

Na fase de treinamento, os padrões são apresentados um por um durante o período ativo, que normalmente dura de 50 a 200ms. Durante este período, as séries temporais de ativação, que são os padrões de oscilação, de cada unidade da camada PC são gravadas e o desvio padrão é calculado. A saída então é um vetor de desvio padrão da unidade KII excitatória mais alta da camada PC. O sistema, logo em seguida, entra em um período de relaxamento que dura 150-300ms. Durante este período, o aprendizado Hebbiano e o processo de habituação acontecem.

A próxima fase é a rotulação. Nesta fase, os padrões usados durante o treinamento são novamente apresentados ao KIII. O vetor saída é então lido e armazenado em uma tabela referência junto com o rótulo da classe, com a finalidade de registrar a associação entre os padrões de oscilação e as classes do problema.

Durante a fase de testes, cada padrão é então apresentado ao sistema KIII. O vetor saída é então lido e comparado com os vetores armazenados na tabela de referência. O algoritmo mais

comumente usado para esta comparação é o *k*-Vizinhos Mais Próximos (*k*-Nearest Neighbors–kNN), apesar de outros algoritmos, como a análise discriminante linear de Fisher, também serem usados.

2.1.5 Simulador de Conjuntos K

Um simulador para os conjuntos de níveis KO até KIII foi construído (PIAZENTIN; ROSA, 2015) para remediar algumas das deficiências encontradas no simulador disponível em MATLAB, como defeitos que causam instabilidades na simulação, a falta de flexibilidade na parametrização do conjunto KIII e o alto tempo computacional necessário para usar o simulador, que ocorrem devido a decisões de projeto e linguagem (PIAZENTIN; ROSA, 2015). O Java foi escolhido porque é uma linguagem popular, multiplataforma, orientada a objetos e eficiente computacionalmente. O software desenvolvido em Java também permite fácil integração com MATLAB, SciLab e outros ambientes (PIAZENTIN; ROSA, 2015).

Na hierarquia dos conjuntos K, os conjuntos KI e KII são feitos de unidades básicas KO e conexões entre elas. Cada conjunto KO armazena um registro de todas as conexões que ele recebe, com seus respectivos pesos e atrasos. Conjuntos KII, por sua vez, são usados para formar um nível intermediário que corresponde a uma camada de conjuntos KII e os conjuntos KIII foram construídos com várias camadas conectadas de um modo específico. O conjunto KIII foi construído de forma modular, permitindo a flexibilidade de usar camadas de tamanho arbitrário, o que não era possível anteriormente, e também a captura da saída da rede em qualquer camada. Esse simulador também permite que o sistema seja parametrizado. É importante notar que o excesso de treinamento pode desestabilizar o KIII devido à superexcitação do sistema, trazendo comportamentos e resultados indesejados.

As conexões entre as três camadas KII que formam o KIII têm, além dos pesos, um retardo de tempo. Assim, a saída em uma conexão destino no tempo t com retardo de tempo sempre retorna o valor de ativação do nó de origem no tempo $t - z$. As conexões também são definidas em relação à conexão entre nós excitatórios ou inibitórios nas camadas KII. A conexão pode portanto ser originada de nós excitatórios ou inibitórios de uma camada e recebidos por nós excitatórios ou inibitórios da camada destino.

Foram mantidos os parâmetros padrão do simulador que, segundo (PIAZENTIN; ROSA, 2015), foram obtidos a partir de buscas globais feitas em (CHANG; FREEMAN, 1996), (CHANG; FREEMAN; BURKE, 1998) e (BELIAEV; ILIN; KOZMA, 2005), feitas de acordo com os critérios definidos em (FREEMAN *et al.*, 1997) e (KOZMA; FREEMAN, 2001).

De acordo com (CHANG; FREEMAN; BURKE, 1998), os pesos não podem ser medidos diretamente dos experimentos. Por isso, os valores deles foram estimados simulando um modelo biológico da atividade medida (ondas EEG endógenas) e potenciais evocados médios (respostas de impulsos AEP evocadas por estímulos elétricos). Devido ao espaço de parâmetros ter muitas

dimensões e, conseqüentemente, a complexidade dos controladores dos parâmetros para o sistema todo ser alta, é praticamente impossível estimar todos os parâmetros ao mesmo tempo. Por isso, [Chang, Freeman e Burke \(1998\)](#), baseados em ([FREEMAN, 1975](#)), ([FREEMAN; SHIMOIDE, 1994](#)) e ([SHIMOIDE; FREEMAN, 1995](#)), fizeram uma abordagem em três passos para estimarem esses parâmetros.

1. Fixe os valores dos parâmetros que podem ser medidos experimentalmente, por exemplo, os valores das constantes da taxa da membrana passiva e os atrasos de condução no laço aberto, e o estado não interativo sob anestesia profunda ([FREEMAN, 1975](#)).
2. Particione o sistema olfatório e otimize os valores dos parâmetros internos de cada parte com relação aos AEPs nos intervalos quase lineares delas e nas frequências características delas ([FREEMAN; SHIMOIDE, 1994](#)).
3. Fixe os valores dos parâmetros que já foram determinados e otimize os valores dos parâmetros da conexão sináptica entre as partes com referência aos EEGs e à atividade unitária múltipla das quatro partes ([CHANG; FREEMAN, 1996](#)).

Os parâmetros estão detalhados nas Tabelas 1, 2 e 3.

Pesos dos Conjuntos KII				
Camada	w_{ee}	w_{ei}	w_{ie}	w_{ii}
1	1,8	1,0	-2,0	-0,8
2	1,6	1,6	-1,5	-2,0
3	1,6	1,9	-0,2	-1,0

Tabela 1 – Pesos Utilizados nos Conjuntos KII de Cada Camada do Conjunto KIII

Pesos Laterais		
Camada	Excitatório	Inibitório
1	0,15	-0,10
2	0,20	-0,20
3	0,15	-0,10

Tabela 2 – Configuração Padrão Utilizada Como Valor Inicial Para Pesos Laterais das Três Camadas do Conjunto KIII

Cabe ressaltar que os valores da Tabela 2 são utilizados apenas para camadas de tamanho $n = 1$. Para camadas com tamanho $n > 1$, o valor utilizado é $\frac{w}{n-1}$, onde w é o respectivo peso excitatório ou inibitório.

O simulador de [Rosa e Piazzentin \(2016\)](#) necessita de um programa em Java para ser usado, e a entrada e a saída de dados e informações é feita em código. Para tornar o uso do simulador mais fácil e mais amigável para o usuário, [Piazzentin e Ferreira \(2017\)](#) criaram uma interface de usuário para que seja possível lidar mais facilmente com o simulador. A interface

Conexões Entre Camadas				
Camada de Origem	Camada de Destino	Tipo de Conexão	Peso	Atraso
1	2	excitatória para excitatória	0,15	1
1	3	excitatória para excitatória	0,60	1
2	1	excitatória para excitatória	0,05	17
2	1	excitatória para inibitória	0,25	25
3	1	inibitória para inibitória	-0,05	25
3	2	excitatória para inibitória	0,20	25

Tabela 3 – Configuração das Conexões Entre Camadas do Conjunto KIII

de usuário para os conjuntos K está disponível tanto no modo gráfico quanto no modo texto. A nova interface pode ser usada de forma manual ou integrada com outros códigos. A entrada e a saída é feita no formato *Comma Separated Values* (CSV). [Piazzentin e Ferreira \(2017\)](#) também disponibilizaram um manual do usuário em *Portable Document Format* (PDF). A interface visa a expansão futura do uso dos conjuntos K nos ambientes de pesquisa.

2.2 Outros Métodos Usados Para a Previsão de Séries Temporais

2.2.1 Métodos Paramétricos

2.2.1.1 Médias Móveis

O modelo de médias móveis de ordem r , $MA(r)$, é uma técnica de predição que usa a média aritmética das r observações anteriores das séries temporais para prever o valor seguinte. Durante todas as execuções do modelo, r permanece constante, de modo a explorar a estrutura de autocorrelação dos resíduos de predição do período atual com aqueles ocorridos em períodos anteriores. A Equação 2.9 define o modelo médias móveis.

$$z_{t+1} = \frac{z_t + z_{t-1} + z_{t-2} + \dots + z_{t-r+1}}{r} \quad (2.9)$$

Aqui r é o número de observações da média z_{t+1} .

Quanto maior for o valor de r , mais suavizados ficam os dados preditos. Desse modo, r deve ser grande para séries com pequenas distorções em seus padrões ou flutuações aleatórias, evitando o efeito de mudanças na distribuição e ruídos. Por outro lado, se a série tiver pouca ou nenhuma aleatoriedade ou apresentar alterações significativas em pontos de inflexão de curvas, é melhor utilizar um r menor para que a série temporal atualize mais rápido as mudanças ocorridas na distribuição.

Entre as desvantagens do modelo de médias móveis, está o fato dele não lidar bem com séries temporais com componentes tendência e sazonalidade. Isso ocorre porque, nesse

método, a previsão de valores sempre considera a adição de novos dados e desconsidera os dados anteriores. Outro problema é que esse modelo atribui a todos os r o mesmo peso, mesmo para as observações mais antigas. Uma alternativa para contornar esse problema seria atribuir pesos às r observações, tornando o método uma média ponderada. Porém, esses pesos precisam ser atribuídos manualmente pelo especialista, com base em um conhecimento profundo sobre a série temporal.

2.2.1.2 Suavização Exponencial Simples

O método de Suavização Exponencial Simples (SES) é equivalente ao método Médias Móveis considerando todos os valores da série temporal, com a diferença de que existe um peso diferente para cada um desses valores. Esses pesos crescem exponencialmente com o tempo, de modo que as observações mais recentes têm maior importância no cálculo do valor a ser previsto. O modelo SES pode ser previsto pela Equação 2.10.

$$L_t = \alpha z_t + \alpha(1 - \alpha)z_{t-1} + \alpha(1 - \alpha)^2 z_{t-2} + \cdots + \alpha(1 - \alpha)^{m-1} z_1 \quad (2.10)$$

Na Equação 2.10, L_t é o nível no instante t , $\alpha(0 < \alpha < 1)$ é o peso ou constante de suavização e z_t é o último valor observado.

Para evitar a utilização da série temporal inteira a cada nova estimativa de L , foi criada uma forma recursiva e mais simples da Equação 2.10, que utiliza o valor atual da série e o nível calculado anteriormente, resultando na Equação 2.11.

$$L_t = \alpha z_t + \alpha(1 - \alpha)L_{t-1} \quad (2.11)$$

No início do procedimento de previsão da série temporal, o primeiro valor ajustado é igual ao primeiro valor da série e o procedimento de ajuste só começa a partir da segunda observação. A predição do instante $m + 1$ é dada pelo nível L_m . Este método suporta somente previsões para horizontes uma etapa a frente. Quando ele é requerido para horizontes maiores, todas as previsões são dadas pelo valor L_m .

O valor de α pode ser escolhido de forma objetiva, através de uma técnica de estimação de parâmetros, ou de forma subjetiva, através de uma regra de escolha. Quando o interesse é obter o componente tendência, é interessante que α seja próximo de zero, de modo a amortizar o efeito da sazonalidade e do resíduo. Valores pequenos de α aumentam a importância dos valores antigos da série. Já quando o objetivo é a previsão da série temporal, α deve ser próximo de 1, para que as observações mais recentes tenham maior influência na previsão do valor futuro.

A popularidade da SES se deve à flexibilidade, simplicidade matemática e razoável precisão. Para fazer uma nova previsão, o método precisa apenas do último valor da série, o

último valor predito e o valor de α . A principal desvantagem é a dificuldade de encontrar o valor adequado de α .

2.2.1.3 Suavização Exponencial de Holt

O modelo SES não funciona bem quando a série apresenta uma tendência linear crescente ou decrescente. Nesse caso, as previsões subestimam ou superestimam os dados reais. Para evitar esse problema, podemos usar outros métodos, como a Suavização Exponencial de Holt (HES). A estrutura desse modelo é parecida com a do modelo SES. A diferença está no acréscimo de uma segunda constante de suavização, chamada de β , para modelar a tendência da série temporal. Veja nas Equações 2.12, 2.13 e 2.14 a estrutura deste modelo.

$$L_t = \alpha z_t + (1 - \alpha)(L_{t-1} + T_{t-1}) \quad (2.12)$$

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta)T_{t-1} \quad (2.13)$$

$$z_{t+h} = L_t + hT_t \quad (2.14)$$

As constantes de suavização α e β variam no intervalo $[0, 1]$, o componente nível é dado pela Equação 2.12 e o componente tendência é dado pela Equação 2.13. Essas equações, por estarem em um método de suavização exponencial, modificam uma estimativa anterior cada vez que uma nova estimativa é calculada. A Equação 2.14 é o valor previsto para o instante $t + h$, onde h é o horizonte de previsão.

Geralmente, quando o método é implementado, assume-se como valores iniciais $L_1 = z_1$ e $T_1 = z_2 - z_1$. Como o método se baseia na auto-aprendizagem, isto é, aprende com seus erros, os valores iniciais não influem nas previsões. Porém, o mesmo não vale para as constantes de suavização, que são difíceis de serem escolhidas, e uma escolha ruim pode prejudicar o desempenho da previsão.

2.2.1.4 Suavização Exponencial Sazonal de Holt–Winters

Os métodos de Holt–Winters são recomendados quando a série apresenta tanto o componente tendência quanto a sazonalidade. Séries com essas características possuem padrões cíclicos que se repetem em períodos constantes de tempo. A estrutura do modelo é definida por três equações com constantes de suavização distintas. Os modelos podem ser aditivos ou multiplicativos, dependendo do padrão sazonal da série investigada.

O modelo Holt–Winters multiplicativo (MHW) é usado para séries temporais com tendência e sazonalidade multiplicativa, ou seja, quando a amplitude da variação sazonal aumenta

com o crescimento do nível médio da série. As equações que definem a estrutura do modelo MHW são as seguintes:

$$L_t = \alpha \frac{z_t}{S_{t-s}} + (1 - \alpha)(L_{t-1} + Tt - 1) \quad (2.15)$$

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta)T_{t-1} \quad (2.16)$$

$$S_t = \gamma \frac{z_t}{L_t} + (1 - \gamma)S_{t-s} \quad (2.17)$$

$$z_{t+h} = (L_t + hT_t)S_{t-s+h} \quad (2.18)$$

Nas equações acima, α , β e γ são constantes de suavização que estão no intervalo $[0, 1]$, s é a duração do período sazonal e z_{t+h} é o valor previsto para o período $t + h$. A escolha dos valores das constantes de suavização deve seguir um critério estabelecido previamente.

Da mesma forma que a SES e a HES, o algoritmo do MHW funciona pelo uso recursivo das três equações acima sobre a série temporal. O modelo deve ser inicializado estimando previamente os valores de L , T e S no período inicial. Uma maneira simples de fazer essa inicialização é estimar o nível e a tendência no mesmo período s . Desse modo, o nível pode ser determinado a partir da média da primeira estação.

$$L_s = \frac{1}{s}(z_1 + z_2 + \dots + z_s) \quad (2.19)$$

A tendência pode ser inicializada usando duas estações completas, conforme a seguinte equação:

$$T_s = \frac{1}{s} \left(\frac{z_{s+1} - z_1}{s} + \frac{z_{s+2} - z_2}{s} + \dots + \frac{z_{s+s} - z_s}{s} \right) \quad (2.20)$$

Os índices sazonais podem ser inicializados pela razão entre as primeiras observações e a média do primeiro período, conforme a seguinte equação:

$$S_1 = \frac{z_1}{L_s}, S_2 = \frac{z_2}{L_s}, \dots, S_s = \frac{z_s}{L_s} \quad (2.21)$$

O método de Holt–Winters aditivo funciona melhor quando a tendência e a sazonalidade são aditivos, ou seja, a amplitude da sazonalidade é sempre a mesma. O algoritmo de previsão desse modelo pode ser descrito pelas seguintes equações:

$$L_t = \alpha(z_t - S_{t-s}) + (1 - \alpha)(L_{t-1} + Tt - 1) \quad (2.22)$$

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta)T_{t-1} \quad (2.23)$$

$$S_t = \gamma(z_t - L_t) + (1 - \gamma)S_{t-s} \quad (2.24)$$

$$z_{t+h} = L_t + hT_t + S_{t-s+h} \quad (2.25)$$

Como pode-se observar, as equações que calculam a tendência (2.16 e 2.23) são idênticas nos dois modelos. A diferença está nas outras equações, onde no modelo multiplicativo a sazonalidade é multiplicada e dividida, e no aditivo ela é somada e subtraída. As inicializações de L e T também são feitas da mesma forma no modelo aditivo e no multiplicativo. Já os valores iniciais dos índices sazonais são calculados pela seguinte equação:

$$S_1 = z_1 - L_s, S_2 = z_2 - L_s, \dots, S_s = z_s - L_s \quad (2.26)$$

A aplicação correta dos métodos de Holt–Winters está associada com a forma das sazonalidades da série temporal, independente da existência de tendência ou não. Nesses modelos, quando $\gamma = 0$, não significa que não há sazonalidade, mas sim que os valores iniciais já estão corretos e não precisam ser corrigidos ao longo da execução do método.

2.2.1.5 Modelos ARIMA e SARIMA

Os modelos ARIMA(p,d,q) são a combinação de três métodos estatísticos: Autorregressão (AR(p)), Integração e Médias Móveis (MA(q)). O uso simultâneo desses três componentes não impede a modelagem de séries temporais com padrões sazonais, uma vez que eles podem ser usados de forma que um complemente o outro. Com essa perspectiva e considerando que a integração pode ser feita durante o pré-processamento, ARIMA também pode referir-se às seguintes estruturas:

- ARIMA(p,0,0) = AR(p);
- ARIMA(0,0,q) = MA(q);
- ARIMA(p,0,q) = ARMA(p,q).

No modelo Autorregressivo AR(p), o valor atual da série é previsto como um agregado linear dos fatores históricos adicionado de um ruído branco, que é um valor de natureza

desconhecida que não pode ser estimado ou previsto com base nos valores anteriores da série temporal.

$$z_t = \delta + \sum_{i=1}^p \phi_i z_{t-i} + e_t \quad (2.27)$$

Na Equação 2.27, z_t é o valor da série no momento t , p é o número de observações, ϕ_i é o i -ésimo coeficiente auto-regressivo, δ é o nível inicial do modelo e e_t representa o ruído branco em uma média zero e variância σ_e^2 . Assume-se também que e_t é independente dos valores passados da série temporal. O valor de δ é calculado pela seguinte equação:

$$\delta = \mu(1 - \phi_1 - \phi_2 - \dots - \phi_p) \quad (2.28)$$

onde μ é a média do processo estacionário.

O modelo de Médias Móveis MA(q) calcula o valor atual da série temporal através de uma combinação linear dos fatores de inovação do período atual com aqueles dos períodos anteriores. O método pode ser descrito pela seguinte equação:

$$z_t = \mu + \sum_{i=1}^q \theta_i e_{t-i} + e_t \quad (2.29)$$

onde θ_i é o i -ésimo valor de ponderação usado e e_t é o ruído branco no instante t . Os pesos θ_i podem assumir valores positivos ou negativos. Para prever o valor z_{t+1} , basta mover os pesos utilizados.

Combinando os modelos AR(p) e MA(q), temos o modelo ARMA(p, q), que é o modelo Autorregressivo de Médias Móveis com ordem (p, q). Na prática, o modelo ARMA estima um valor considerando que a série pode ser prevista pela combinação de um processo autorregressivo com um processo de Médias Móveis. A equação desses modelos é a seguinte:

$$z_t = \delta + \sum_{i=1}^p \phi_i z_{t-i} + \sum_{i=1}^q \theta_i e_{t-i} + e_t \quad (2.30)$$

A grande vantagem dos modelos ARMA é a maior facilidade de ajustar a sua estrutura às séries temporais estacionárias complexas, frequentemente necessitando de uma menor quantidade de termos que a exigida pelos modelos puramente AR ou puramente MA.

Os modelos AR, MA e ARMA são apropriados quando a série é estacionária, isto é, suas propriedades estatísticas básicas (média, variância, covariância) não mudam ao longo do tempo. Porém, quando a série é não estacionária, pode-se transformá-la em estacionária utilizando o processo de diferenciação dos dados. O acréscimo desse processo de integração no método

ARMA resulta no método Autorregressivo Integrado de Médias Móveis com ordem (p,d,q) ARIMA (p,d,q) , conforme a equação abaixo:

$$I'_t = \delta + \sum_{i=1}^p \phi_i I'_t - i + \sum_{i=1}^q \theta_i e_{t-i} + e_t \quad (2.31)$$

Na Equação 2.31, $I'_t = \Delta^d z_t = \Delta(\Delta^{d-1} z_t)$ e d é o grau do operador de diferença; ϕ_p é o parâmetro do procedimento autorregressivo com comprimento de defasagem p , θ_q é o parâmetro do procedimento Médias Móveis, com comprimento de defasagem q ; e e_t é o ruído branco que não pode ser previsto pelo modelo. Em resumo, o modelo ARIMA é a transformação de uma série não estacionária em uma série estacionária pela d -ésima diferença e a aplicação do ARMA para estimá-la. Desse modo, qualquer série que não tenha tendência explosiva, tendo portanto não-estacionariedade homogênea ou sendo estacionária, pode ser modelada pelo ARIMA.

A constante δ pode ser omitida quando a série é não estacionária, uma vez que esse tipo de série é diferenciada para obter uma série estacionária. Quando a série já é estacionária na forma original, porém a média é diferente de zero ou o desvio padrão é diferente de um, essa constante é necessária. Quando o modelo não tem a parte autorregressiva (AR(p)), a constante recebe como valor a média dos valores da série temporal.

O ARIMA funciona usando a autocorrelação entre observações sucessivas. Porém, quando os dados são observados em períodos cíclicos de até um ano, a série temporal também pode ter autocorrelação em um período s . Diante disso, surge o modelo SARIMA (Autorregressivo Integrado de Médias Móveis Sazonal), que é uma generalização do ARIMA, acrescentando uma parte sazonal (P,D,Q).

$$I''_t = \delta + \sum_{i=1}^P \Phi_{is} I''_t - is + \sum_{i=1}^Q \Theta_{is} e_{t-is} + e_t \quad (2.32)$$

Na Equação 2.32, $I''_t = \Delta^D z_t = \Delta(\Delta^{D-1} z_t)$ e D é o grau do operador de diferença sazonal. A constante δ é calculada na Equação 2.33 e funciona da mesma maneira que no ARIMA. Φ_P é o parâmetro do procedimento autorregressivo sazonal de comprimento P e Θ é o parâmetro do procedimento Médias Móveis de comprimento Q . e_t é o ruído branco, que não pode ser previsto pelo modelo.

$$\delta = \mu(1 - \Phi_1 - \Phi_2 - \dots - \Phi_P) \quad (2.33)$$

O SARIMA é definido pela Equação 2.34, que contém as partes sazonal e não sazonal.

$$I_t = \delta + \sum_{i=1}^p \phi_i I'_t - i + \sum_{i=1}^q \theta_i e_{t-i} + \sum_{i=1}^P \Phi_{is} I''_t - is + \sum_{i=1}^Q \Theta_{is} e_{t-is} + e_t \quad (2.34)$$

O SARIMA é mais adequado em aplicações em séries temporais com sazonalidade que não podem ser tratadas pela primeira diferença.

Os modelos ARIMA e SARIMA são concebidos com base no ciclo iterativo de Box–Jenkins. As quatro etapas do método serão detalhadas a seguir:

1. **Seleção da Estrutura do Modelo:** A estrutura é escolhida com base nas características da série temporal. Se a série tiver apenas o componente tendência, usa-se o ARIMA. Porém, se além do componente tendência a série possuir o componente sazonalidade, usa-se o SARIMA.
2. **Identificação das Ordens do Modelo:** Os valores p , d e q , do $ARIMA(p,d,q)$, ou os valores p , d , q , P , D , Q do $SARIMA(p,d,q) \times (P,D,Q)_s$, podem ser estimados por um dos dois procedimentos a seguir. No procedimento de correlograma, o número de integrações (d ou D) é calculado através de uma iteração de integrações, onde a sequência é diferenciada até que sua variância seja menor que a da série original. Então, observam-se as funções de autocorrelação e autocorrelação parcial amostrais da série temporal nas defasagens $1, 2, 3, 4, \dots$ para obter o valor de p e q e nas defasagens $s, s \times 2, s \times 3, s \times 4, \dots$ para obter o valor de P e Q . Outro procedimento possível seria a utilização de critérios de informação. Um desses critérios de informação é o Critério de Informação de Akaike (AIC), descrita pela Equação 2.35

$$AIC = -2 \times LL + (\log(n) + 1) \times NP \quad (2.35)$$

Na Equação 2.35, LL é o logaritmo da função de verossimilhança, n é o número de observações e NP é a quantidade de parâmetros necessária. O objetivo é selecionar o AIC com o menor tamanho. Isso beneficia modelos com menos parâmetros.

3. **Estimação dos Coeficientes do Modelo:** As primeiras estimativas dos coeficientes ϕ_p e Φ_P , da parte autorregressiva, e dos coeficientes θ_q e Θ_Q das Médias Móveis, podem ser obtidas pelas autocorrelações da série temporal integrada durante a fase identificação das ordens do modelo. Depois da inicialização desses valores, usa-se a maximização de uma função de verossimilhança. Cabe ressaltar que a função de verossimilhança é maximizada usando-se os mínimos quadrados não lineares combinados com o algoritmo Levenberg–Marquardt.
4. **Diagnóstico do Modelo Ajustado:** O modelo escolhido pelos passos anteriores é então testado para verificar-se se ele está adequado para a série temporal. Para isso, as estimativas de erro (resíduos) são calculadas usando autocorrelação residual. Se esses resíduos tiverem comportamento de ruído branco, o modelo pode ser usado para prever horizontes futuros. Caso contrário, será necessário escolher outro modelo e repetir as etapas anteriores.

2.2.2 Métodos Não Paramétricos

Os métodos não paramétricos, também chamados métodos por Aprendizado de Máquina, buscam descrever as propriedades dos dados sem antes conhecer a distribuição deles. Como não dependem de parâmetros para modelar os fenômenos, são mais simples de serem ajustados e têm bom desempenho mesmo em séries complexas e altamente não lineares. De acordo com o modo como as observações das séries temporais são usadas no modelo de previsão, o método não paramétrico pode ter dois tipos de abordagem: global e local.

Na abordagem global, os métodos de Aprendizado de Máquina constroem modelos usando no treinamento como entrada todas as observações da série. Normalmente, quando se usa essa abordagem, é necessário transpor a sequência de dados para uma tabela atributo-valor, de modo que ela possa ser usada como entrada para os algoritmos de Aprendizado de Máquina de regressão. Definido o horizonte de previsão, os dados da tabela atributo valor são particionados em conjunto de treinamento e conjunto de teste. O conjunto de treinamento é usado para a confecção do modelo e o conjunto de teste é usado para a avaliação de desempenho do modelo de previsão. A acurácia de predição é calculada comparando os valores previstos com os valores reais observados.

Existem duas estratégias de previsão multi-etapa à frente, passo aproximado e passo atualizado. A previsão com passo aproximado enfrenta dificuldades como acúmulo de erro, desempenho reduzido e aumento da incerteza. Esses problemas aumentam com o horizonte de previsão, uma vez que os valores previstos acabam não sendo substituídos pelos valores reais. Como vantagem, essa estratégia permite o estudo dos algoritmos quanto à propagação do erro de previsão.

A aproximação global possui algumas limitações. Uma delas é que os pares atributo-valor são considerados independentes e identicamente distribuídos pelos algoritmos de Aprendizado de Máquina. Essa suposição tem como consequência a perda da informação temporal, e na degradação no desempenho de previsão do modelo de regressão. Entre os métodos dessa abordagem, podemos citar as funções polinomiais e racionais, as Redes Neurais Artificiais, incluindo os Conjuntos K de Freeman, e as Máquinas de Vetores de Suporte.

A abordagem local inclui algoritmos de Aprendizado de Máquina que foram adaptados para incluir informação temporal dos dados no processo de previsão. Esses métodos particionam a série temporal em subsequências cujos valores mais importantes ou mais próximos aos valores atuais são combinados para prever o valor futuro. Essas combinações são feitas por funções de aproximação, como média local simples ou ponderada. Entre os métodos dessa abordagem encontram-se as diferentes variações do kNN (*k-Nearest Neighbors* ou *k Vizinhos Mais Próximos*).

2.2.2.1 Redes Neurais Artificiais Clássicas

As redes neurais artificiais são modelos computacionais que buscam simular o processamento feito pelo cérebro humano. Elas são compostas por neurônios que se unem por conexões chamadas sinapses. Cada conexão é altamente especializada e envia sinais de um neurônio para o outro. Essas redes podem ser implementadas usando-se componentes eletrônicos ou programas de computador.

O Perceptron é a forma mais simples de Rede Neural Artificial. A arquitetura dele é mostrada na Figura 5. Ele pode ser usado para classificar padrões linearmente separáveis. Um conjunto é linearmente separável quando ele pode ser separado em dois conjuntos disjuntos usando apenas um hiperplano.

Figura 5 – Arquitetura do Perceptron

Fonte: Adaptada de Haykin *et al.* (2009).

No Perceptron, há um único neurônio que abrange todas as entradas. A cada entrada, é associado um peso sináptico que reflete a importância da entrada para a rede. Esse peso pode ser negativo (sinapse inibitória) ou positivo (sinapse excitatória). A combinação linear das entradas com os pesos, mais um limiar (bias), produz um valor *net*. Esse valor *net* será enviado para uma função de ativação. A função de ativação define a saída do neurônio (valor *y*).

O bias tem como finalidade corrigir, aumentando ou diminuindo, o valor de *net*. Essa correção serve para que a saída do neurônio fique a mais próxima possível do esperado. A função de ativação pode ser a função degrau (neurônio de McCulloch e Pitts), função linear, função rampa e função sigmoideal. O domínio de *y* pode ser binário ($y \in \{0, 1\}$ ou $y \in \{-1, 1\}$) ou contínuo ($y \in \mathbb{R}$).

Os pesos sinápticos do Perceptron são adaptados por um processo de aprendizado que emprega um número finito de iterações. A aprendizagem é conduzida pelo algoritmo de

convergência do Perceptron, que é uma regra de correção de erros. O algoritmo visa encontrar pesos de modo que as condições da função de ativação estejam satisfeitas.

O estudo do Perceptron causou grande impacto nas pesquisas de Inteligência Artificial. Porém, ele foi subitamente interrompido quando foi demonstrado que ele não resolve problemas não linearmente separáveis, como o ou exclusivo. Esse problema poderia ser solucionado acrescentando camadas intermediárias a sua estrutura. Porém, não se sabia se era possível construir um algoritmo de convergência para modelos com múltiplas camadas com baixo custo computacional.

O desenvolvimento de redes neurais de duas camadas só foi retomado com o surgimento do algoritmo *Backpropagation*. Nesse algoritmo, a rede neural é treinada em dois passos. No primeiro passo, um padrão é apresentado à camada de entrada da rede. O processamento propaga-se camada por camada até chegar na camada de saída.

Na camada de saída, é produzida uma resposta e o erro é calculado pela diferença entre a saída obtida e a saída esperada. O erro é retropropagado da camada de saída para a camada de entrada. Durante esse processo, os pesos das sinapses das camadas intermediárias são ajustados. Esse processo é repetido até a convergência da rede.

Um tipo de rede neural com várias camadas, treinada pelo *Backpropagation*, é o *Multi-layer Perceptron* (MLP). A arquitetura de uma MLP é mostrada na Figura 6. No MLP, podem existir uma ou mais camadas de neurônios entre a camada de entrada e a camada de saída. Essas camadas intermediárias não interagem diretamente com o ambiente e são extratoras de características. Se existirem conexões apropriadas entre as unidades de entrada e um conjunto considerável de unidades intermediárias, sempre será possível encontrar a representação que irá produzir o mapeamento correto entre a entrada de dados e a saída do resultado. A aplicação do MLP possibilita a manipulação eficiente de grandes volumes de dados. A capacidade de generalização é uma das principais características do MLP. Contudo, o MLP resulta em modelos pouco compreensíveis.

A precisão do MLP está associada à três aspectos: determinação do número de camadas ocultas, definição do número de neurônios em cada uma das camadas e especificação dos pesos sinápticos que ligam os neurônios nas diferentes camadas.

São necessárias no máximo duas camadas intermediárias, com um número suficiente de unidades por camada, para se produzir quaisquer mapeamentos. Com apenas uma única camada intermediária é possível aproximar qualquer função contínua.

2.2.2.2 Máquinas de Vetores de Suporte

Uma Máquina de Vetores de Suporte é uma técnica de Aprendizado de Máquina baseada na Teoria do Aprendizado Estatístico. Ela tem a capacidade de resolver diversos problemas, incluindo os de classificação e os de regressão. A estrutura é semelhante à das redes neurais, porém o modo de aprendizado é diferente. As redes neurais tentam minimizar o erro do modelo

Figura 6 – Arquitetura de uma MLP com Duas Camadas Ocultas

Fonte: Adaptada de [Parmezan \(2016\)](#).

criado usando os dados de treinamento. As SVM buscam pelo menor erro de treino minimizando um limite superior para um erro de generalização do modelo.

Para compreender melhor a generalização, usaremos a classificação binária. Nesse caso, temos duas classes e um conjunto de pontos que pertencem a estas. A SVM vai determinar um hiperplano para separá-las. Esse hiperplano coloca a maior quantidade possível de pontos da mesma classe do mesmo lado. Ao mesmo tempo, a distância de cada classe para o hiperplano tem que ser maximizada. Isso pode ser melhor visualizado na Figura 7.

A técnica anteriormente descrita funciona somente quando o problema é linearmente separável. Em outras situações, é necessário mapear os dados para um espaço de dimensão maior usando uma função kernel. Os kernels mais utilizados são os polinomiais, os gaussianos, as Funções de Base Radial (RBF) e os sigmoidais. Cada um deles tem parâmetros que precisam ser configurados pelo usuário.

Por exemplo, o modelo SVM com kernel RBF exige dois parâmetros: $\mathbb{C}$ e σ . A constante $\mathbb{C}$ é um termo de regularização que impõe um peso à minimização dos erros no conjunto de treinamento em relação à minimização da complexidade do modelo. O valor de σ é a largura da gaussiana da função kernel. O número de funções radiais e seus respectivos centros são determinados pelos vetores de suporte encontrados.

A construção de uma SVM implica na resolução de um problema quadrático, com restrições lineares. Esse problema depende do conjunto de dados de entrada, de alguns parâmetros e da margem de separação. Durante o treinamento desse modelo, são obtidos os multiplicadores de Lagrange que caracterizam os vetores de suporte. Estes, por sua vez, definem as margens do hiperplano de separação ótima.

Figura 7 – Hiperplano de Separação Ótima e Seus Hiperplanos de Suporte

Fonte: Adaptada de [Parmezan \(2016\)](#).

Assim como as redes neurais, as SVM podem resolver problemas de alta complexidade. Porém, elas não permitem interpretar adequadamente os resultados ou entender a evolução do processo adaptativo que levou às respostas. Essa carência de legibilidade da solução está atrelada à sua própria estrutura, que reside em uma combinação linear de funções kernel.

Os modelos SVM podem ser estendidos para resolver tarefas de regressão e, consequentemente, de previsão de séries temporais. Contudo, para viabilizar o seu treinamento, é necessário reformular o problema de otimização para lidar com as particularidades e propósitos dessa tarefa. O *Sequential Minimal Optimization* é o algoritmo de codificação SVM que dispõe dos recursos suficientemente necessários para a projeção de dados temporais. Ele foi desenvolvido para decompor o problema de minimização quadrático em vários subproblemas de programação quadrática.

2.2.2.3 *k*-Vizinhos Mais Próximos

O algoritmo kNN é um algoritmo de aprendizado de máquina supervisionado do paradigma baseado em exemplos. Ele visa encontrar, segundo uma medida de similaridade, os k exemplos mais próximos de um exemplo ainda não rotulado e, baseado nos rótulos desses k exemplos mais próximos, rotular o novo exemplo. Assim, se $k = 1$, o novo exemplo será classificado como pertencente à mesma classe do único exemplo mais próximo de acordo com a

medida de similaridade. Por outro lado, se $k > 1$, a classe predominante entre os k vizinhos mais próximos será atribuída ao novo exemplo.

A quantidade de vizinhos próximos considerada para a classificação pode influenciar diretamente no resultado da aplicação do algoritmo. O valor do parâmetro k é particular a cada problema e estimá-lo incide em uma tarefa delicada que merece atenção especial. Outro fator que deve ser levado em consideração é a definição da medida de similaridade, que deve ser escolhida mediante às características dos exemplos. O kNN demanda baixo custo computacional durante a fase de treinamento, porém o custo de classificação pode ser alto. Isso ocorre porque no pior caso o algoritmo utilizará todos os exemplos do conjunto de treinamento para a comparação.

2.3 IPCA – Índice de Preços ao Consumidor Amplo

Nesta dissertação, foi utilizada uma série temporal da atividade econômica, mais especificamente um índice de preços ao consumidor, o Índice de Preços ao Consumidor Amplo (IPCA), coletado e divulgado pelo Instituto Brasileiro de Geografia e Estatística (IBGE). Os dados foram extraídos do Sistema Gerenciador de Séries Temporais do Banco Central – v2.1 (SGS) ([BANCO CENTRAL DO BRASIL, 2017b](#)). Desta série, foram considerados somente os dados referentes ao período de agosto de 1994 até junho de 2017, para evitar a contaminação dos dados com a volatilidade do período pré Plano Real ([FRANCO et al., 1994](#)). Os valores dos índices estão na Figura 8.

O IPCA faz parte do Sistema Nacional de Índices de Preços ao Consumidor (SNIPC). Esse sistema tem como objetivo produzir índices de preços ao consumidor. Isso é feito acompanhando os preços das cestas de produtos e serviços das famílias.

O sistema abrange as regiões metropolitanas do Rio de Janeiro, Porto Alegre, Belo Horizonte, Recife, São Paulo, Belém, Fortaleza, Salvador, Curitiba e Vitória, além do Distrito Federal e dos municípios de Goiânia e Campo Grande. O índice nacional é obtido a partir da união dos índices regionais calculados para uma mesma faixa de renda.

Os índices mensais são obtidos comparando-se os preços dos 30 dias daquele mês com os 30 dias do mês anterior. Os preços são coletados em um período de 30 dias que é segmentado em 4 subperíodos de cerca de 7 dias cada um.

Os índices podem ser obtidos para famílias de diferentes faixas de rendimento mensal, desde que as respectivas estruturas de ponderação estejam disponíveis.

Além dos índices mensais, podem ser obtidos índices para 2 meses ou mais a partir das séries históricas.

O SNIPC permite calcular diversos índices de preços ao consumidor, entre eles o IPCA. O IPCA abrange as famílias das regiões urbanas do SNIPC com rendimentos de 1 a 40 salários mínimos.

Figura 8 – Índice Nacional de Preços ao Consumidor Amplo

Fonte: Elaborada pelo autor.

O SNIPC usa informações de três pesquisas básicas: Pesquisa de Orçamentos Familiares (POF), Pesquisa de Locais de Compra (PLC) e Pesquisa de Especificação de Produtos e Serviços (PEPS).

Os preços coletados são os preços pagos à vista pelo consumidor. A pesquisa é realizada em estabelecimentos comerciais, prestadores de serviços, concessionárias de serviços públicos e domicílios.

O índice é calculado para cada região. Uma vez coletados os preços, são estimados os movimentos desses preços para cada produto pesquisado.

Essas estimativas são obtidas calculando-se a média aritmética simples dos preços do produto nos locais onde ele é pesquisado. Comparando-se as estimativas em dois meses

consecutivos, temos os relativos dos produtos.

A variação de preços de cada subitem é obtida agregando-se os relativos dos produtos que o compõem através da média geométrica. O subitem é a menor agregação do índice.

A partir daí, basta aplicar a fórmula Laspeyres para obter os demais níveis: item, subgrupo, grupo e o índice geral da região.

A fórmula Laspeyres é definida pela Equação 2.36.

$$L(P_1, P_0) = \sum P_1 X_0 / \sum P_0 X_0 = \sum P_1 X_0 / Y_0 \quad (2.36)$$

onde P_1 é o preço de comparação e P_0 é o preço de referência, X_0 é o item pesquisado e Y_0 é o custo inicial do produto pesquisado (BRAITHWAIT, 1980).

O índice nacional é calculado pela média aritmética ponderada dos índices regionais.

A coleta é feita entre os dias 1 e 30 de cada mês e a divulgação é feita em aproximadamente 8 dias úteis após a coleta.

A pesquisa iniciou-se em janeiro de 1979 para Rio de Janeiro; junho de 1979 para Porto Alegre, Belo Horizonte e Recife; janeiro de 1980 para São Paulo, Brasília e Belém; outubro de 1980 para Fortaleza, Salvador e Curitiba; janeiro de 1991 para Goiânia; e janeiro de 2014 para Vitória e Campo Grande. O índice nacional começou a ser divulgado em setembro de 1981.

TRABALHOS RELACIONADOS

3.1 Trabalhos Usando Conjuntos K

Um trabalho que se aproxima muito da proposta desta dissertação é o trabalho (LI; KOZMA, 2003). Esse trabalho começa descrevendo os conjuntos K. Em seguida, é apresentada uma visão geral da previsão de séries temporais. Existem as séries temporais periódicas, que podem ser previstas por combinações de senos e cossenos modulados em amplitude, frequência e fase. Contudo, séries temporais mais complicadas (dinâmicas), são altamente não lineares, e precisam ser previstas por outros métodos. O método mais básico para se prever essas séries é a regressão linear. O segundo método é a reconstrução do espaço de estados com atraso temporal. O terceiro método utiliza redes neurais. As redes neurais podem aprender padrões dos dados, generalizar e prever valores desconhecidos a partir dos valores históricos. Entre as redes neurais, estão os conjuntos K. Um nível dos conjuntos K é o KIII. O conjunto KIII é uma simulação do córtex sensorial em um modelo computacional dinâmico. Ele é feito de três camadas conectadas entre si. O trabalho utilizou a classificação de padrões em vez de um valor real. Foi utilizada como classe a direção da variação da taxa de câmbio. Foram utilizadas três séries temporais. Os dados foram normalizados pela diferença de primeira ordem e para o intervalo $[-1 \ 1]$. Cada janela tem 40 dados de séries temporais consecutivos. Inicialmente, os dados foram discretizados em três classes: alta, baixa e de lado. A última classe é muito pequena e foi excluída para simplificar. O desempenho do algoritmo foi em torno de 55% de classificações corretas. As principais diferenças entre o trabalho (LI; KOZMA, 2003) e esta dissertação: o primeiro usa séries temporais diárias, com sazonalidade semanal, mais voláteis e com padrões discretos, enquanto esta dissertação usa séries temporais mensais, com sazonalidade anual, menos voláteis e com valores reais contínuos. Do trabalho (LI; KOZMA, 2003), embora o KIII usado seja uma implementação muito simples, pode-se tirar a inspiração para o pré-processamento de séries temporais e o tratamento de séries temporais financeiras, que são altamente não lineares, não estacionárias e ruidosas.

O trabalho que mais se aproxima da proposta desta dissertação, entre todos os trabalhos relacionados aqui, é o que descreve a ferramenta de conjuntos K para MATLAB e sua aplicação em séries temporais financeiras (BELIAEV; ILIN; KOZMA, 2005). Existiam até então várias implementações computacionais para os conjuntos K, e esse trabalho pretendia unificar todas essas implementações computacionais em uma única implementação para MATLAB. O artigo começa descrevendo brevemente os conjuntos K. Depois ele apresenta uma descrição do aprendizado Hebbiano implementado computacionalmente. Então, são descritos detalhes do projeto, do modelo de seleção de parâmetros e da análise de complexidade computacional. A aplicação do modelo é uma previsão de taxa de câmbio dólar/euro e usa uma série temporal com 100 cotações diárias sendo 20 para teste. No artigo, não é citada a fonte das taxas de câmbio dólar/euro usadas no trabalho relacionado. Foram utilizados como classificadores o KIII combinado com o kNN. Como medida de erro, foi usado o *Mean Squared Error* (MSE) (Erro Quadrático Médio). O MSE do KIII foi $0,9419 * 10^{-5}$ e o do caminho aleatório $6,4968 * 10^{-4}$. A ferramenta usada no trabalho citado tem as mesmas bases da ferramenta utilizada nesta dissertação (desenvolvida por Piazzentin e Rosa (2015)) e a previsão de séries temporais feita foi bastante semelhante, tendo parte do algoritmo sido usada para a construção do algoritmo usado nesta dissertação. As diferenças foram a utilização de uma série diária ao invés de mensal e da medida MSE ao invés de RMSE. A medida RMSE, por ser a raiz quadrada da MSE, tende a dar valores mais suaves, tornando-a mais fácil de ser compreendida por leitores humanos, sem perder a capacidade de comparação de desempenho.

Dois trabalhos relacionados a este, muito próximos um do outro, foram os trabalhos (KOZMA; BELIAEV, 2004) e (BELIAEV; KOZMA, 2007). Ambos os trabalhos abordaram a previsão de séries temporais usando o conjunto KIII. O trabalho (KOZMA; BELIAEV, 2004) foi submetido em 2004 como parte da competição IJCNN 2004 CATS, e o conjunto de dados, que tem natureza desconhecida, recebeu o nome de IJCNN 2004 CATS *Benchmark Test*. No artigo, o desempenho do KIII superou o desempenho do caminho aleatório, o que sugere uma natureza não estocástica do conjunto de dados. Segundo Kozma e Beliaev (2004), esse desempenho superior pode ser devido ao fato do modelo KIII funcionar bem em ambientes de previsão difíceis com propriedades dinâmicas não lineares. A previsão foi feita para trechos conhecidos, com a finalidade de comparar o desempenho do método com a realidade, e para trechos desconhecidos. A descrição feita no trabalho aqui relacionado foi bastante semelhante à descrição feita no trabalho (LI; KOZMA, 2003). Após o processamento com o KIII, os dados foram classificados com um kNN. Os dados foram normalizados pela diferença de primeira ordem e colocados no intervalo $[-1 \ 1]$. Aparentemente, a série tem um comportamento não gaussiano, de tipo $1/f$ e vários picos oscilatórios em frequências mais altas, indicando uma dinâmica não linear complexa, uma área na qual os conjuntos K funcionam bem. Kozma e Beliaev (2004) também calcularam os melhores valores de k para cada lacuna e horizonte de previsão nos dados desconhecidos. As lacunas são os segmentos de tempo na série temporal original que precedem e seguem as partes desconhecidas. O trabalho (BELIAEV; KOZMA, 2007) é uma extensão do trabalho

(KOZMA; BELIAEV, 2004). Nele, as previsões foram refeitas. Como os dados e os resultados da IJCNN 2004 CATS já haviam sido divulgados, foi possível comparar os valores previstos com a série verdadeira. Foi possível ajustar um conjunto maior de parâmetros do KIII que não foram ajustados no artigo (KOZMA; BELIAEV, 2004) devido à falta de tempo. O KIII teve um desempenho superior ao caminho aleatório e ao kNN sozinho, porém somente com horizonte de previsão de até dez passos à frente. Isso provavelmente ocorreu devido à dificuldade de criar e manter um atrator caótico em horizontes muito longos.

Uma importante contribuição desses dois trabalhos foi no pré-processamento dos dados. Embora a diferença de primeira ordem não foi usada, por razões de tempo e de necessidade de verificar o desempenho em dados brutos, usou-se a normalização no intervalo $[-1, 1]$, caso contrário os algoritmos de previsão por similaridade não funcionariam. Aqui foram usadas 3 lacunas, enquanto que nos trabalhos (KOZMA; BELIAEV, 2004) e (BELIAEV; KOZMA, 2007) foram usadas 5 lacunas. Nos trabalhos (KOZMA; BELIAEV, 2004) e (BELIAEV; KOZMA, 2007), as lacunas A-D têm 10 pontos perdidos para frente e 10 para trás, e a lacuna E tem 20 pontos perdidos para frente. Por outro lado, aqui usou-se lacunas A-C com seis pontos perdidos para frente, uma vez que os modelos paramétricos não permitiram que fossem feitas previsões para trás. Foram fixados os conjuntos $A = \{s(t)\}_{t=1}^{92}$, $B = \{s(t)\}_{t=93}^{183}$, $C = \{s(t)\}_{t=184}^{275}$, uma memória M de tamanho 5, um conjunto de treinamento $T = \{(s_i, s_{i+1}, \dots, s_{i+M-1}) : i \in I_T\}$, um conjunto referência $R = \{(s_i, s_{i+1}, \dots, s_{i+M-1}) : i \in I_R\}$ e um conjunto de validação $C = \{(s_i, s_{i+1}, \dots, s_{i+M-1}) : i \in I_C\}$, onde I é um conjunto índice que define unicamente a estrutura de treinamento, $I_T \subset I$, $I_R \subset I$, $I_C \subset I$ e $|I_C| = 5$. O conjunto de validação é composto pelos últimos 5 meses de cada lacuna. Podemos definir $I = \{j : 1 < j < 85\}$ para o conjunto A, uma vez que a lacuna tem tamanho 85. Como novidade desta dissertação, na fixação dos conjuntos I_T e I_R , houve a criação do conjunto pai I_P , que contém os meses anteriores ao conjunto validação, e a divisão dele em I_T com os índices ímpares e I_R com os índices pares.

Um outro trabalho bastante importante é o trabalho (PIAZENTIN; ROSA, 2015). Piazentin e Rosa (2015) desenvolveram um simulador para conjuntos K de Freeman dos níveis KO a KIII na linguagem Java. Esse simulador foi construído para remediar problemas do simulador disponível anteriormente construído em MATLAB (BELIAEV; ILIN; KOZMA, 2005), como defeitos que causam instabilidades, falta de flexibilização na parametrização e alto tempo computacional requerido para o uso do simulador. Piazentin e Rosa (2015) alegam ter escolhido Java por ela ser popular, multiplataforma, orientada a objetos e computacionalmente eficiente. O software desenvolvido em Java também permite fácil integração com MATLAB, Scilab e outros ambientes. O software foi avaliado originalmente em uma aplicação de agrupamento de dados de redes sociais. O trabalho descreve não só detalhes da composição e estrutura do KIII, mas também de como funciona o seu aprendizado e a classificação usando esses modelos. O simulador dos conjuntos K foi apresentado. Esse simulador foi feito de maneira modular, iniciando do KO e formando os demais conjuntos da hierarquia a partir dessa unidade básica, o que o torna bastante flexível na construção de camadas. As configurações de conexões, pesos

e atrasos padrões são apresentados em três tabelas, sendo obtidos empiricamente de trabalhos anteriores. Foi citado também o tempo padrão do período de ativação e repouso. A aplicação foi em agrupamento de dados de redes sociais. Os dados têm origem no *Stanford Large Network Dataset Collection* (LESKOVEC; KREVL, 2014) e em duas importantes redes sociais: Facebook e Google+. Esses dados possuem como atributos características dos perfis das pessoas. Foram utilizados diversos algoritmos de agrupamentos e diversas medidas de validação de grupos. O método usado superou os métodos existentes na literatura ao detectar comunidades (ROSA; PIAZENTIN, 2016). O trabalho (PIAZENTIN; ROSA, 2015) gerou o simulador em Java que foi incorporado ao código Scilab usado por esta dissertação para a predição de séries temporais por similaridade. As configurações foram mantidas em seus valores padrão devido à falta de tempo para testar outras configurações. Foi usado o kNN contínuo como regressor.

3.2 Trabalhos Sobre Previsões de Séries Temporais

A avaliação do desempenho do KIII foi feita usando vários métodos de previsão estado da arte e medidas de desempenho. Esses métodos e medidas foram extraídos do artigo (GOOIJER; HYNDMAN, 2006). Esse artigo é comemorativo dos 25 anos do *Journal of Forecasting* e do *International Journal of Forecasting*. Ele é basicamente um resumo dos principais artigos publicados no periódico de 1981 a 2006 e de alguns outros artigos e livros relevantes sobre o assunto de previsão de séries temporais. Apesar do artigo ser antigo (publicado 12 anos antes da entrega da versão revisada desta dissertação), é ainda bastante atual, uma vez que os métodos estado da arte pouco mudaram nos últimos anos. Os artigos citados foram organizados por cada modelo.

Temos, a princípio, a descrição dos métodos de suavização exponencial, juntamente com a sua evolução ao longo desses 25 anos. Um modelo de suavização exponencial que merece destaque é a *Simple Exponential Smoothing* (SES) (Suavização Exponencial Simples). Ela é o mais simples dos métodos de suavização exponencial. Os principais métodos, e que inclusive foram usados no trabalho que originou esta dissertação, são a SES (sem tendência e sem sazonalidade), método linear de Holt (tendência aditiva e sem sazonalidade), método aditivo de Holt–Winters (tendência aditiva e sazonalidade aditiva) e método multiplicativo de Holt–Winters (tendência aditiva e sazonalidade multiplicativa). Todos esses métodos foram usados no trabalho que originou esta dissertação.

Depois, o artigo descreve os métodos ARIMA, a começar do trabalho (YULE, 1927), que criou o conceito de série estocástica e os modelos Autorregressivos (*AutoRegressive* (AR)) e Média Móvel (*Moving Average* (MA)). Também cita o livro *Time Series Analysis: Forecasting and Control* (BOX; JENKINS; REINSEL, 1970), uma obra importantíssima para a previsão de séries temporais e que criou a abordagem Box–Jenkins. Cita ainda que, com o surgimento dos computadores, foi possível a popularização da técnica *AutoRegressive Integrated Moving Average*

(ARIMA) (Autorregressivo Integrado de Médias Móveis). Foram usados no trabalho desta dissertação dois tipos de ARIMA: o ARIMA puro e o SARIMA (ARIMA com sazonalidade).

O artigo também citou várias abordagens para se lidar com a sazonalidade em séries temporais, entre elas o método de extração de sazonalidade X-11 e suas variantes, inclusive o X-12-ARIMA, que era a versão mais recente até então. Outras abordagens e métodos também foram citados. O artigo também fala sobre os modelos de espaço de estados. O principal deles é o filtro de Kalman. Outras classes de modelos de espaço de estados são os modelos lineares dinâmicos (DLM), os modelos estruturais não gaussianos, modelos de espaço de estados equilibrados e os modelos de fonte única de erro.

O artigo também fala sobre os modelos não lineares, começando a partir do trabalho (VOLTERRA; MATHEMATICS, 1930), que introduziu o uso do primeiro tipo de modelo não linear. O artigo cita, como métodos não lineares, os modelos de mudança de regime, como CTAR, SETAR e STAR, e o modelo de coeficiente funcional (FCAR), porém não serão detalhados nesta dissertação porque não foram usados no trabalho que a originou. Também são citados muitos outros modelos não lineares, porém eles não menos conhecidos e frequentemente pouco usados.

Um tipo de modelo não linear descrito no artigo e fortemente usado nesta dissertação são as redes neurais artificiais. Uma área de aplicação importante das redes neurais é a previsão, principalmente a previsão de dados financeiros. É provável que as redes neurais funcionem melhor para dados financeiros de alta frequência e que necessitem de uma longa série temporal para garantir ótimos resultados do treinamento da rede neural. As redes neurais são mais propensas a superar outros métodos quando os dados de entrada são mantidos o mais atuais possíveis, usando uma modelagem recursiva. Um problema geral com modelos não-lineares é a “maldição da complexidade do modelo e da parametrização do modelo”. O artigo cita o risco de super-parametrização e superposição, e complementa apresentando que saber se uma rede neural está ou não parametrizada, ainda está sob investigação.

O artigo também descreve duas classes de métodos bastante usados, os modelos de memória longa, entre eles o FARMA e o ARFIMA e os modelos ARCH e GARCH. O artigo também descreve trabalhos sobre previsões em dados de contagem. O artigo citou uma série desconcertante de medidas de precisão que têm sido utilizadas para avaliar o desempenho dos métodos de previsão. As medidas mais comuns são: MSE, RMSE, MAE, MdAE, MAPE, MdAPE, sMAPE, sMdAPE, MRAE, MdRAE, GMRAE, RelMAE, RelRMSE, LMR, PB, PB(MAE) e PB(MSE). Foi usada no trabalho que originou esta dissertação a medida RMSE. O artigo também fez referência sobre métodos de combinações de previsão feitas por métodos diferentes para produzir resultados melhores. O artigo também descreve o uso dos intervalos de predição. Cabe ressaltar a diferença entre intervalo de predição e intervalo de confiança. Um intervalo de confiança é para um parâmetro de modelo, enquanto um intervalo de predição é para uma variável aleatória. O artigo citou em suas conclusões lacunas da área de previsão de séries temporais levantadas durante sua pesquisa bibliográfica e espera que elas sejam resolvidas no futuro.

[Parmezan \(2016\)](#) trabalhou na criação de um novo algoritmo de previsão de séries temporais, uma nova extensão do algoritmo de Aprendizado de Máquina *k-Nearest Neighbors* (kNN), que chamou de *kNN - Time Series Prediction with Invariances* (kNN-TSPI). O novo algoritmo difere da versão convencional pela incorporação de três técnicas para a obtenção de invariância à amplitude e deslocamento, invariância à complexidade e tratamento de casamentos triviais. [Parmezan \(2016\)](#) demonstrou ao longo de sua dissertação que o uso simultâneo dessas técnicas proporciona ao kNN-TSPI uma melhor correspondência entre as subsequências de dados e a consulta de referência. Os resultados de uma das avaliações empíricas mais extensas, imparciais e abrangentes já conduzidas no tema de predição de séries temporais evidenciaram, a partir do confronto de dez métodos de projeção, que o algoritmo kNN-TSPI, além de ser conveniente para a predição automática de dados a curto prazo, é competitivo com os métodos estado da arte ARIMA e SARIMA. Embora o modelo SARIMA atingiu uma precisão relativamente superior à do método baseado em similaridade, o kNN-TSPI é consideravelmente mais simples de ser ajustado. Os 95 conjuntos de dados empregados nos experimentos juntamente com todas as projeções analisadas em termos de Erro Quadrático Médio, coeficiente U de Theil e taxa de acerto *Prediction Of Change In Direction* encontram-se disponíveis no portal Web ICMC-USP *Time Series Prediction Repository*.

O autor, [Parmezan \(2016\)](#), para iniciar o seu trabalho, fez uma revisão sistemática da literatura, buscando identificar lacunas a serem preenchidas. Depois, apresentou conceitos de Séries Temporais, Análise de Séries Temporais e Mineração de Dados Temporais. Em seguida, houve um capítulo sobre Previsão de Séries Temporais, onde foram dados conceitos teóricos sobre Previsão de Séries Temporais de uma maneira geral, e foi projetada avaliação empírica de maneira teórica, com os métodos para a construção dos modelos de previsão, paramétricos e não paramétricos, as técnicas para a estimação de parâmetros e as medidas de avaliação da qualidade de predição. No capítulo seguinte, são descritos conceitos de Aprendizado de Máquina, o algoritmo kNN-TSP e kNN-TSPI. O próximo capítulo é sobre similaridade em séries temporais. Esse capítulo fala sobre distorções em séries temporais e técnicas empregadas para obter invariâncias a esses eventos indesejados. Ele apresenta a invariância à amplitude e ao deslocamento, à escala local, à escala uniforme, à fase, e à complexidade. Depois são apresentadas diversas medidas de distância, incluindo medidas invariantes à complexidade. Os dois capítulos seguintes são sobre a avaliação experimental na prática. No primeiro capítulo da avaliação experimental, foram usados 55 conjuntos de dados reais e foi feita uma sequência de experimentos de previsão de séries temporais por similaridade. O algoritmo proposto (kNN-TSPI) foi comparado com outros dois métodos de previsão por similaridade e o desempenho preditivo também foi comparado. Foram também inspecionadas as invariâncias às distorções, medidas de distância, medidas de complexidade e funções de predição. O segundo capítulo da avaliação experimental faz a comparação do kNN-TSPI com nove métodos tradicionais da literatura. Foram utilizadas 55 séries temporais reais e 40 séries temporais sintéticas. Essa avaliação empírica teve como objetivo: 1. viabilizar a consolidação da eficiência e da efetividade de cada algoritmo

apurado; 2. caracterizar, em termos de vantagens e desvantagens de uso, os distintos métodos existentes para a construção de modelos de previsão. O último capítulo é a conclusão, onde o autor coloca as considerações finais, as limitações e as possibilidades de trabalhos futuros. A avaliação empírica do trabalho (PARMEZAN, 2016) serviu de modelo para a avaliação empírica desta dissertação, que usou os mesmos dez métodos de projeção, e algumas de suas técnicas para a estimação de parâmetros. Os parâmetros usados foram os mesmos e variaram da mesma maneira. A diferença é que não foram usadas algumas técnicas adicionais, como validação holdout, validação cruzada e método Box-Jenkins, devido a limitações de tempo.

MATERIAIS E MÉTODOS

4.1 Considerações Iniciais

Foram utilizados, para a avaliação do modelo KIII na previsão de séries temporais econômicas, as bases de dados do IPCA e scripts escritos para serem executados nos ambientes Scilab e R. No Scilab, foram feitos experimentos com os métodos não paramétricos. No R, foram feitos os experimentos com os métodos paramétricos.

4.2 Execução de Métodos Não Paramétricos

Para executar os testes não paramétricos, primeiro um arquivo CSV em português brasileiro foi extraído do SGS ([BANCO CENTRAL DO BRASIL, 2017b](#)). Como era de se esperar em arquivos desse formato neste idioma, as duas colunas são separadas por ponto e vírgula e os números têm seus decimais separados por vírgula. Esse arquivo tem um cabeçalho com os nomes das duas colunas e um rodapé citando a fonte. As colunas recebem o nome de “Data”, coluna esta que contém os meses, e a coluna “433 - Índice nacional de preços ao consumidor-amplio (IPCA) - Var. % mensal” que descreve o índice, com o código, nome e tipo de variação, e os valores dessa coluna são a variação do índice naquele mês.

No começo da execução, o arquivo foi lido e convertido em *double* pelo Scilab. O cabeçalho e o rodapé foram retirados, uma vez que não puderam ser convertidos para *double* e assumiram valores NaN, além de serem meramente descritivos e irrelevantes para os experimentos que foram realizados em seguida. Também foi retirada a coluna “Data”, uma vez que não foi possível convertê-la para *double* e, portanto, usá-la no experimento. Uma vez isolada a coluna com os índices mensais, passou-se para a etapa de tornar todos os índices positivos. Essa etapa foi necessária porque alguns métodos não funcionam com inflação negativa. O procedimento foi bastante simples: somou-se a toda a série o valor absoluto do menor valor da série, para que

esse valor se igualasse a zero, e depois acrescentou-se a essa mesma série um valor positivo extremamente pequeno (nesta dissertação, foi usado o valor 10^{-40}), para que ele se tornasse levemente positivo.

Uma vez processada a base, ela foi dividida em três subconjuntos: o conjunto A, que contém os meses de agosto de 1994 a março de 2002 e tamanho de 92 meses, o conjunto B, com os meses de abril de 2002 a outubro de 2009 e tamanho de 91 meses, e o conjunto C, com os índices dos meses de novembro de 2009 a junho de 2017 e tamanho de 92 meses. Ao final de cada conjunto, foi retirada uma lacuna, que foi usada como região desconhecida para testar a precisão do método de previsão. A lacuna A abrange os meses de setembro de 2001 a fevereiro de 2002, tendo o tamanho de 6 meses, a lacuna B vai de abril de 2009 a outubro de 2009, com duração de 6 meses, e a lacuna C, com 7 meses, contendo os índices dos meses de dezembro de 2016 a junho de 2017.

Após a remoção das lacunas, foram criados os conjuntos índices. Inicialmente, foi configurado um tamanho de memória M . Nesta dissertação, o tamanho foi 5. Então, foi criado um vetor índice que vai de 1 até o tamanho do conjunto sem a lacuna e sem a memória M . Foi também criado um vetor com os índices dos últimos 5 meses do conjunto para ser usado na validação, chamado vetor de índices de validação. Depois, foi criado um vetor de índices que foi denominado vetor de índices pai, que contém os 30 índices anteriores ao índices do vetor de validação. Para gerar o vetor de treinamento, foram utilizados os índices pares do vetor pai, e para o vetor de referência, os índices ímpares.

Uma vez criados os vetores índices, passou-se para a etapa da criação das matrizes índices. As matrizes índices foram criadas percorrendo cada coluna do respectivo vetor índice e aplicando a memória M a cada coluna. Então, foram finalmente criados os conjuntos validação, treinamento, referência e teste. O conjunto de validação foi gerado aplicando a matriz de índices de validação a cada conjunto de dados pré-processado, e o mesmo foi feito para os conjuntos teste e referência, usando seus respectivos índices. O conjunto de testes é o conjunto de validação transposto. Por fim, o valor real do horizonte que se deseja prever foi guardado em uma variável para comparação posterior ao longo dos experimentos.

No momento em que todos os conjuntos de dados já estavam criados, começou-se a execução do KIII. Ele está implementado no paradigma orientado a objetos. Então, no início da execução, foi criado um objeto da classe KIII, tendo como parâmetro o tamanho da camada de conjuntos KII. A camada tem tamanho 5, que é o tamanho da entrada. Depois foram feitas outras configurações e iniciada a rede.

A próxima etapa foi a configuração dos contadores a serem usados no laço iterativo. Dentro do laço iterativo, os parâmetros variaram dentro de uma faixa pré-definida, baseada no trabalho de [Parmezan \(2016\)](#). Inicialmente, as configurações foram $n = 0$ e $k = 0$, onde n é o número de passos consecutivos em que o experimento foi executado sem que o erro tenha diminuído e k é o número de vizinhos mais próximos a ser considerado. O k variou de 1 a 10 e o

n tem tamanho máximo de 10. O valor n era zerado em toda a execução onde havia diminuição do erro. Também foram iniciados os valores da medida *Root Mean Square Error* (RMSE) (Raiz do Erro Quadrático Médio) de cada modelo e cada conjunto. Como o objetivo era diminuir os valores dessa medida, eles foram inicializados com o valor 10^{40} . Esse valor foi usado por ser um valor muito grande, e esperava-se que todos os valores RMSE que surgissem ao longo da dissertação fossem menores do que ele. Caso surgissem valores maiores que esta medida inicial, esse valor seria aumentado de modo que ele fosse sempre um valor máximo.

Dentro do laço, os objetos KIII foram inicialmente treinados com os conjuntos de treinamento e os resultados foram obtidos filtrando-se os conjuntos de referência através dos respectivos conjuntos KIII, gerando a matriz *kresult*. Para comparar a previsão, foram selecionados os últimos 15 índices pares dos três conjuntos iniciais, A, B e C, formando a matriz *kprediction*. Cada coluna do conjunto de teste foi percorrida pelo respectivo objeto KIII e os resultados concatenados em uma matriz *kexp*. As três matrizes, mais o número de vizinhos mais próximos, foram passados ao kNN, que retornou um resultado de previsão. O valor foi comparado com o RMSE atual. Se fosse menor, o valor obtido passava a ser o novo RMSE, o contador n era zerado e o processo era repetido. Se fosse maior, o RMSE permanecia o mesmo, n era incrementado em mais 1 e o processo era repetido até n atingir o valor 10.

O processo para o kNN bruto foi mais simples. O valor k foi variado de 0 a 10. Como parâmetros ao kNN, foram passados o conjunto de referência, o conjunto de treinamento, a matriz *kprediction* e o número de vizinhos mais próximos (k). O RMSE foi calculado. O valor calculado foi comparado com o valor atual. Se o valor obtido fosse menor que o valor comparado, o valor obtido passava a ser o novo valor.

Outro método não paramétrico utilizado para a comparação foi as redes neurais artificiais clássicas. Para preparar os dados para a previsão usando esse modelo, inicialmente os conjuntos de dados brutos, de treinamento e de teste foram normalizados para o intervalo [0 1]. Uma variável correspondente ao valor que se quer comparar foi separada para descobrir-se a precisão da previsão. Todos os conjuntos de dados gerados desse modo foram redimensionados para terem o mesmo tamanho. Os tamanhos das camadas de entrada e saída seguiram os tamanhos dos conjuntos de dados que entravam e saíam e, portanto, variou-se apenas o tamanho da camada intermediária. Esse tamanho variou de 3 até 11, que é o número ímpar imediatamente inferior a 12, tamanho do período sazonal. Uma vez definido o tamanho da rede, foram inicializados e ajustados os pesos, executados os testes para definir a acurácia da rede, calculado o resultado da rede e minimizado o RMSE usando os mesmos procedimentos dos métodos anteriores.

O último modelo não paramétrico utilizado foi a SVM. O tipo de SVM usada foi a nu-SVR com núcleo função de base radial (CHANG; LIN, 2016). Na utilização desse modelo foram considerados dois parâmetros: o parâmetro de regularização, que variou de um número positivo muito pequeno, próximo de zero, até 1, com um passo de 0,25, e a largura da gaussiana da função base radial, que variou de 0,005 a 0,25, em um passo de 0,05. Para fazer as previsões,

primeiro os modelos foram treinados usando um vetor com elementos aleatórios e distribuição normal, os dados de treinamento normalizados já utilizados na previsão com redes neurais clássicas, o tipo de SVM, o tipo de função de núcleo, o parâmetro de regularização e a largura da gaussiana da função base radial. Em seguida, as previsões foram calculadas, passando-se para a função do Scilab correspondente o vetor com os elementos aleatórios, o vetor com os dados e o objeto com os modelos treinados. Então, o RMSE para aquele passo foi calculado e comparado com o RMSE que estava armazenado naquele momento. Se o novo RMSE fosse menor, o novo RMSE substitua o antigo. Caso contrário, o RMSE continuava como estava.

Ao final da execução do programa, todos os RMSE foram exibidos para que pudessem ser copiados para uma planilha Libre Office e posteriormente comparados entre si e com RMSE de métodos paramétricos.

Os procedimentos descritos até aqui foram repetidos para cada horizonte d meses à frente da série com a lacuna retirada, com d variando de 1 até o tamanho da lacuna.

4.3 Execução de Métodos Paramétricos

Para executar os testes paramétricos, um arquivo CSV no idioma inglês foi extraído do SGS ([BANCO CENTRAL DO BRASIL, 2017b](#)). Como era de se esperar em arquivos desse formato nesse idioma, as duas colunas são separadas por ponto e vírgula e os números têm seus decimais separados por ponto. Esse arquivo tem um cabeçalho com os nomes das duas colunas e um rodapé citando a fonte. As colunas recebem o nome de “Date”, coluna esta que contém os meses, e a coluna “433 - Broad National Consumer Price Index (IPCA) - Monthly % var.”, que descreve o índice, com o código, nome em inglês e tipo de variação, também em inglês, e os valores dessa coluna são a variação do índice naquele mês.

Inicialmente, o arquivo CSV foi lido e convertido em um *data frame*. O cabeçalho do arquivo se transformou no nome dos campos do *data frame*. Em seguida, o rodapé foi removido do *data frame*. A coluna contendo os valores dos índices foi copiada para um vetor, que foi convertido para o tipo *double*. Após a conversão, adicionou-se o valor absoluto do menor elemento do vetor e um número positivo extremamente pequeno (neste caso, 10^{-40}) para que todos os valores fossem positivos. Por fim, os dados foram encapsulados em uma variável do tipo série temporal com início em agosto de 1994 e período sazonal de 12 meses.

Após esse processamento inicial, a série inicial foi dividida em três séries para poder-se comparar as previsões. A série 1 vai de agosto de 1994 até março de 2002. A série 2 abrange os meses de abril de 2002 a outubro de 2009. Por fim, temos a série 3, com dados de novembro de 2009 a junho de 2017. Depois, lacunas ao final de cada série foram retiradas, para que fossem usadas para comparar as previsões com valores reais. A série 1 sem a lacuna vai de agosto de 1994 a agosto de 2001, a série 2 com a lacuna retirada vai de abril de 2002 a abril de 2009 e, por fim, a série 3 sem lacuna vai de novembro de 2009 a novembro de 2016.

Por fim, foram geradas as séries de teste, que nada mais são que os dados imediatamente posteriores às séries sem lacunas, até o horizonte máximo de previsão usado nesta dissertação, que foi 6. A série de teste 1 vai de setembro de 2001 até fevereiro de 2002, a série de teste 2 vai de maio a outubro de 2009 e a série de teste 3 vai de dezembro de 2016 a maio de 2017.

Uma vez criada a base, iniciou-se a execução do programa. Como o objetivo desta dissertação foi minimizar o RMSE, inicialmente os RMSE da Média Móvel, da SES, do método linear de Holt, do método aditivo de Holt–Winter e do método multiplicativo de Holt–Winter foram configurados para um valor máximo muito grande, neste caso 10^{40} .

Com os RMSE configurados, iniciaram-se as previsões usando as médias móveis. Esse cálculo foi feito variando o parâmetro tamanho do período de sazonalidade de 3 até 12, em passos de 2. No início da previsão, foram executados métodos de médias móveis com as séries sem as lacunas e o tamanho do período de sazonalidade. Em seguida, os valores NA gerados no início e no fim de cada média móvel foram retirados. Então, foram geradas séries de teste específicas para as médias móveis. A previsão foi feita e os pontos previstos foram extraídos para uma variável. Neste momento, calculou-se o erro e o RMSE. Se o RMSE fosse menor que o valor armazenado na memória, ele substituiria o RMSE antigo. Caso contrário, o RMSE permaneceria o mesmo.

Na sequência, foram feitas as previsões usando o método SES. Nesse método, o parâmetro α variou de 10^{-4} (mínimo permitido pelo R) até 1, em passos de 0,05. A previsão foi feita passando para o método SES as séries temporais sem lacunas, o horizonte de previsão e o parâmetro α . As previsões feitas foram extraídas e o erro foi calculado comparando-as com as séries de teste. Foi calculado então o RMSE para cada α . Se o RMSE calculado fosse menor que todos os RMSE anteriores, ele substituiria o RMSE armazenado na memória. Caso contrário, o RMSE permaneceria sem modificações na memória.

Na previsão de séries temporais usando o método linear de Holt, existem dois parâmetros, α e β . Ambos variaram de 10^{-4} a 1, em passos de 0,05. Para o método funcionar, β tem que ser menor ou igual a α . A previsão foi feita, para cada combinação de α e β , passando-se para o método as séries temporais sem a lacuna, o horizonte de previsão, e os parâmetros α e β . Os valores previstos foram calculados, extraídos e posteriormente comparados com a série temporal de teste. O erro foi calculado e, a partir dele, o RMSE. Sempre que um RMSE calculado era menor que o RMSE armazenado, ele substituiu o RMSE armazenado.

O modo de previsão feito pelos métodos aditivo e multiplicativo de Holt–Winter é bastante parecido. Ambos os modelos têm 3 parâmetros: α , β e γ . O que difere um modelo do outro é o valor da variável *seasonal*. No método aditivo, esse valor é configurado para *additive* e no multiplicativo, para *multiplicative*. Além desses parâmetros, os modelos recebem também as séries temporais com a lacuna retirada e o horizonte de previsão. Os parâmetros α , β e γ variaram de 10^{-4} a 1, em passos de 0,05. Existem muitas combinações de α , β e γ que impedem a previsão. As combinações mais triviais foram evitadas mediante estruturas de

decisão *if* aninhadas. Porém, combinações mais complexas são muito difíceis de serem evitadas e precisaram ser tratadas mediante o comando *try*. Quando a previsão dava erro, o modelo de previsão retornava um objeto da classe *try-error*. Quando a classe era outra, a previsão era calculada, extraída e comparada com as séries de teste. O erro e o RMSE foram calculados. O menor RMSE de todas as iterações foi usado ao final.

4.4 Resultados

As Tabelas 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14 e 15 mostram os resultados da execução dos métodos de previsão de séries temporais desta dissertação sobre os conjuntos A, B e C, já descritos anteriormente na Página 62. Nas tabelas 4, 6, 8, 10, 12 e 14, estão os resultados obtidos na execução dos métodos não paramétricos, e nas tabelas, 5, 7, 9, 11, 13 e 15, os resultados dos métodos paramétricos. A Figura 9 mostra o desempenho geral dos métodos usados na comparação. Já a Figura 10 mostra o desempenho dos métodos nos experimentos feitos utilizando somente os métodos não paramétricos e a Figura 11 mostra o desempenho considerando somente os métodos paramétricos. Os métodos não paramétricos usados foram o Conjunto KIII com algoritmo kNN (KIII), kNN, *Artificial Neural Networks* (ANN) (Redes Neurais Artificiais) e *Support Vector Machines* (SVM) (Máquinas de Vetores de Suporte). Já os métodos paramétricos usados foram *AutoRegressive Integrated Moving Average* (ARIMA) (Modelo Autorregressivo Integrado de Médias Móveis), *Seasonal AutoRegressive Integrated Moving Average* (SARIMA) (Modelo Autorregressivo Integrado de Médias Móveis Sazonal), *Moving Average* (MA) (Médias Móveis), *Simple Exponential Smoothing* (SES) (Suavização Exponencial Simples), Método Linear de Holt (Holt), *Additive Holt–Winters* (AHW) (Método Aditivo de Holt–Winters) e *Multiplicative Holt–Winters* (MHW) (Método Multiplicativo de Holt–Winters).

Os resultados destacados em *itálico* indicam que aquele resultado foi obtido pelo método de melhor desempenho naquele conjunto (A, B ou C), naquele tipo de método (paramétrico ou não paramétrico) e naquele horizonte. Os resultados destacados em **negrito** indicam que aquele resultado foi obtido pelo método de melhor desempenho naquele conjunto e naquele horizonte, considerando tanto os métodos paramétricos quanto os métodos não paramétricos. A medida de comparação usada foi a *Root Mean Squared Error* (RMSE) (Raiz Quadrada do Erro Quadrático Médio). Ela foi escolhida porque a raiz quadrada suaviza os valores da medida MSE, tornando-a mais fácil de ser compreendida por leitores humanos, sem perder a capacidade de comparação de desempenho. A equação da medida pode ser vista logo abaixo (Equação 4.1).

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (\hat{t}_t - y_t)^2}{n}} \quad (4.1)$$

onde $\hat{y}_t$ é o valor previsto para o tempo t , y_t é o valor real do tempo t e n é o número de previsões.

Nosso objetivo e hipótese iniciais foram que a aplicação de um filtro KIII melhoraria o desempenho na previsão das séries temporais econômicas. Pelos resultados aqui relatados, essa

hipótese foi parcialmente confirmada, uma vez que o desempenho do KIII com o kNN superou o desempenho das previsões feitas somente com o kNN, exceto no conjunto A, horizonte 4 etapas à frente e no conjunto C, horizonte 6 etapas à frente. Porém, o KIII superou os demais métodos somente quando aplicado ao conjunto A, horizonte 3 etapas à frente, conjunto B, horizonte de previsão 5 e 6 etapas à frente; e ao conjunto C, horizonte de previsão três etapas à frente. Ele também superou os demais métodos não paramétricos no conjunto A, horizonte de previsão duas e cinco etapas à frente; e no conjunto C, horizontes um, dois e quatro etapas à frente. O kNN contínuo bruto, na maioria dos casos, também não teve um bom desempenho. Foi o vencedor apenas no conjunto A, horizonte 4 etapas à frente e no conjunto C, horizonte 6 etapas à frente, foi o segundo melhor apenas no conjunto A, horizontes 2 e 3 etapas à frente e no conjunto C, horizonte 3 etapas à frente; terceiro colocado em 11 casos e perdendo até mesmo para as SVM nos conjuntos B e C, horizonte 4 etapas à frente.

No conjunto A, entre os métodos não paramétricos, o filtro KIII teve melhor desempenho em três dos horizontes de previsão, as redes neurais clássicas em dois e o kNN em um. Tanto as redes neurais clássicas quanto o filtro KIII são redes neurais e são mais robustos a ruídos e à volatilidade. Como esse conjunto de dados contém o período com os primeiros meses imediatamente após o Plano Real, a volatilidade do IPCA ainda era muito grande naquela época, apresentando picos de 2,81% em novembro de 1994 e -0,51% em agosto de 1998, além de outras grandes variações. Entre os métodos paramétricos, o método aditivo de Holt–Winters ganhou em 2 horizontes de previsão, enquanto os métodos ARIMA, SARIMA, linear de Holt e multiplicativo de Holt–Winters ganharam em um horizonte cada. Como o período do conjunto era bastante instável, os métodos paramétricos não obtiveram um bom sucesso, não havendo um vencedor entre eles, e o melhor desempenho entre um e outro foi aparentemente aleatório.

No conjunto B, entre os métodos não paramétricos, as redes neurais clássicas tiveram melhor desempenho em quatro horizontes de previsão e o KIII em dois horizontes. Isso indica uma grande instabilidade, porém menor que a existente no período de referência do conjunto A. Ainda nesse conjunto, com relação aos métodos paramétricos, os métodos multiplicativo de Holt–Winters, aditivo de Holt–Winters e SARIMA ganharam em dois horizontes cada. Isso deixa bem claro que a série possui sazonalidade. Porém, não está claro se a sazonalidade é aditiva ou multiplicativa.

No conjunto C, entre os métodos não paramétricos, o maior vencedor foi o KIII, que venceu todos os primeiros quatro horizontes de previsão. As redes neurais clássicas venceram o horizonte 5, mais difícil. O kNN venceu o horizonte 6, muito difícil e de previsão aleatória, de modo que o melhor desempenho do kNN pode ser mera coincidência. Novamente, temos a indicação de que os índices de inflação de países em desenvolvimento são bastante instáveis e voláteis, de modo que as redes neurais tiveram um melhor desempenho mesmo neste conjunto menos instável e menos volátil, que contém os índices de inflação mais recentes. Já nos métodos paramétricos, ainda considerando o conjunto C, houve a vitória do método de Holt–Winters,

sendo 2 horizontes para o aditivo e um para o multiplicativo. O SARIMA ganhou em dois horizontes e a SES no horizonte seis. Novamente, percebe-se uma série com tendência linear e forte sazonalidade. Em alguns horizontes predominou a sazonalidade aditiva, e outros a multiplicativa. Nos últimos horizontes, mais difíceis de serem previstos, houve um melhor desempenho de métodos mais simples. O método SES, que é uma suavização exponencial sem tendência e sem sazonalidade, venceu no horizonte 5. Já o método SARIMA venceu no horizonte 6, o mais difícil de ser previsto entre todos os horizontes desta pesquisa.

O KIII foi melhor do que o kNN em quase todos os conjuntos de dados e horizontes de previsão, demonstrando que ele é um bom filtro. Nos métodos não paramétricos, as RNAs e o KIII tiveram o melhor desempenho nos índices mais antigos, evidenciando uma inflação era mais instável nesse período. Nos índices mais recentes, outros métodos tiveram uma maior chance, devido à menor instabilidade da série, porém as redes neurais continuaram dominando. Nos métodos paramétricos, o método de Holt–Winters foi o vencedor, porque a série temporal tem tendência linear e sazonalidade. Não está claro se esta sazonalidade é aditiva ou multiplicativa, uma vez que, em alguns casos, o Holt–Winters aditivo venceu, e em outros, foi o Holt–Winters multiplicativo que venceu. Os métodos (S)ARIMA também tiveram um bom desempenho, principalmente a versão sazonal, fato que ocorreu devido à presença de sazonalidade e à presença de autocorrelação entre os elementos da série.

Figura 9 – Métodos Vencedores

Fonte: Elaborada pelo autor.

	KIII	KNN	ANN	SVM
A	0,016	0,09700000	0,0021658	0,5978732
B	0,030	0,08900000	0,0004215	0,4812814
C	0,040	0,34005714	0,1180266	0,6484109

Tabela 4 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Uma Etapa à Frente

	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
A	0,122047	0,003843	0,129837	0,355504	0,026181	0,000347	0,005131
B	0,000457	0,000006	0,070000	0,003110	0,002152	0,000438	0,000252
C	0,000659	0,007384	0,040749	0,008638	0,001102	0,000015	0,001919

Tabela 5 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Uma Etapa à Frente

	KIII	KNN	ANN	SVM
A	0,0013333	0,0020000	0,0044409	0,7635358
B	0,0024000	0,0093333	0,0001022	0,4472257
C	0,0160000	0,2100000	0,2048598	0,7000238

Tabela 6 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Duas Etapas à Frente

	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
A	0,048369	0,050470	0,569990	0,050457	0,000619	0,189207	0,001674
B	0,084522	0,112670	0,060000	0,072609	0,022415	0,009235	0,005674
C	0,000484	0,000053	0,004939	0,018293	0,011606	0,009100	0,000513

Tabela 7 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Duas Etapas à Frente

	KIII	KNN	ANN	SVM
A	2,22044604925031E-16	0,0053333	0,0111459	0,8770559
B	0,008	0,0240000	0,0016697	0,6868843
C	2,22044604925031E-16	0,1000000	0,2096667	0,8790697

Tabela 8 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Três Etapas à Frente

	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
A	0,003147	0,035196	0,013553	0,010550	0,000714	0,000264	0,001253
B	0,204773	0,215681	0,050000	0,192609	0,055265	0,000467	0,005920
C	0,103295	0,003918	0,016758	0,017823	0,007110	0,000687	0,073486

Tabela 9 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Três Etapas à Frente

	KIII	KNN	ANN	SVM
A	0,0012	0,0008	0,0033901	0,2039829
B	0,1060	0,2050	0,0023299	0,0053654
C	0,0140	0,2210	0,1882608	0,0699648

Tabela 10 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Quatro Etapas à Frente

	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
A	0,000506	0,036886	0,139990	0,014495	0,044809	0,003751	0,314678
B	0,293304	0,130000	0,036643	0,282609	0,026677	0,001008	0,003839
C	0,218234	0,001395	0,005881	0,058638	0,044178	0,113015	0,000015

Tabela 11 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Quatro Etapas à Frente

	KIII	KNN	ANN	SVM
A	0,00056667	0,030	0,0035916	0,7653237
B	0,00000000	0,024	0,0001754	0,5723659
C	0,11600000	0,321	0,0922634	0,5970874

Tabela 12 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Cinco Etapas à Frente

	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
A	0,1010652	0,0000000	0,1183606	0,1155048	0,0011275	0,0948402	0,0462471
B	0,1995173	0,0004853	0,1564554	0,1926099	0,0713806	0,1400928	0,0900865
C	0,3513492	0,0026639	0,0258530	0,1686385	0,1518710	0,1145268	0,0197270

Tabela 13 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Cinco Etapas à Frente

	KIII	KNN	ANN	SVM
A	0,0943333	0,2080000	0,0069655	0,7123337
B	0,0000000	0,0385000	0,0004977	0,6675241
C	0,0065000	0,0031429	0,1781777	0,6924037

Tabela 14 – Resultado das Previsões com os Métodos Não Paramétricos Para o Horizonte Seis Etapas à Frente

	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
A	0,2571433	0,0374895	0,0111151	0,2755048	0,0163146	0,1691680	0,0023422
B	0,1563772	0,1180324	0,1700000	0,1526099	0,1194382	0,0016472	0,0001998
C	0,1966414	0,0374863	0,0030527	0,0013614	0,0204368	0,3316610	0,2071911

Tabela 15 – Resultado das Previsões com os Métodos Paramétricos Para o Horizonte Seis Etapas à Frente

Figura 10 – Métodos Não Paramétricos Vencedores

Fonte: Elaborada pelo autor.

Figura 11 – Métodos Paramétricos Vencedores

Fonte: Elaborada pelo autor.

CONCLUSÃO

5.1 Conclusões

O principal objetivo desta dissertação foi demonstrar que uma abordagem conexcionista pouco conhecida, os conjuntos K de Freeman, mais especificamente o nível KIII, fornece previsões mais realistas para as séries temporais econômicas, mais especificamente uma série temporal da atividade econômica, um índice de preços ao consumidor, o Índice de Preços ao Consumidor Amplo (IPCA), coletado e divulgado mensalmente pelo Instituto Brasileiro de Geografia e Estatística (IBGE). Ao longo do desenvolvimento desta dissertação, foram feitas previsões usando o KIII, que foram comparadas com outras técnicas estado da arte, baseando-se na dissertação ([PARMEZAN, 2016](#)) e no artigo ([GOOIJER; HYNDMAN, 2006](#)). Ao final da pesquisa, a hipótese foi parcialmente verificada, porque os conjuntos KIII funcionaram bem como um filtro para melhorar o desempenho do kNN contínuo. Porém, eles superaram o desempenho de outras técnicas testadas em apenas alguns poucos resultados, indicando que não se trata de um bom modelo de regressão. Esse problema pode ter ocorrido devido à técnica de regressão utilizada, porém uma limitação do trabalho foi não confirmar ou refutar essa hipótese, uma vez que não foram feitas as avaliações empíricas para tal. A principal contribuição do trabalho foi a utilização de métodos não paramétricos, especialmente redes neurais clássicas e caóticas, para a previsão de um índice de inflação de um país em desenvolvimento.

5.2 Trabalhos Futuros

Como um trabalho futuro, propõe-se a utilização dos conjuntos KIII como um filtro não somente para o k-NN, mas também para outras técnicas, a fim de confirmar se o baixo desempenho do KIII foi devido ao algoritmo de regressão kNN contínuo ou ao filtro KIII em si. Outra possibilidade de continuação do trabalho seria a utilização de outros métodos de previsão por similaridade, como o kNN-TSPI ([PARMEZAN; BATISTA, 2015](#)) e o *kNN - Time Series*

Prediction (kNN-TSP) (FERRERO, 2009), para a previsão de séries temporais. Pretende-se refazer os experimentos, considerando métodos semi-paramétricos e não paramétricos mais avançados, que são usados na economia para a previsão de séries temporais, e compará-los com a previsão feita usando o KIII. Também pretende-se utilizar técnicas para estimativa de parâmetros, como a validação *holdout*, a validação cruzada e o método Box–Jenkins, para obter parâmetros mais otimizados e um desempenho melhor. Um trabalho futuro interessante seria uma avaliação empírica mais abrangente, com base nos conjuntos de dados disponíveis no ICMC-USP *Time Series Prediction Repository* (PARMEZAN; BATISTA, 2014). Uma questão de pesquisa interessante seria o estudo aprofundado dos conjuntos K, especialmente o nível KIII, com o objetivo de criar uma nova variante com melhor desempenho baseada nos conjuntos K de Freeman. Por fim, pretende-se publicar um artigo relacionado à pesquisa que originou esta dissertação em uma conferência ou periódico relevante para a área.

REFERÊNCIAS

BANCO CENTRAL DO BRASIL. **Notas da 209ª Reunião do Comitê de Política Monetária (Copom) do Banco Central do Brasil**. 2017. Disponível em: <<http://www.bcb.gov.br/?COPOM209>>. Citado na página 23.

_____. **SGS - Sistema Gerenciador de Séries Temporais - v2.1**. 2017. Disponível em: <<https://www3.bcb.gov.br/sgspub/localizarseries/localizarSeries.do?method=prepararTelaLocalizarSeries>>. Citado nas páginas 23, 49, 61 e 64.

BELIAEV, I.; ILIN, R.; KOZMA, R. Nonlinear neurodynamics tool for system analysis and application for time series prediction. In: **2005 IEEE International Conference on Systems, Man and Cybernetics**. [S.l.: s.n.], 2005. Citado nas páginas 34, 54 e 55.

BELIAEV, I.; KOZMA, R. Time series prediction using chaotic neural networks on the CATS benchmark. **Neurocomputing**, v. 70, n. 13–15, p. 2426–2439, aug 2007. Citado nas páginas 28, 54 e 55.

BOX, G. E. P.; JENKINS, G. M.; REINSEL, G. C. **Time Series Analysis: Forecasting and Control**. [S.l.]: BLACKWELL PUBL, 1970. ISBN 1118675029. Citado na página 56.

BRAITHWAIT, S. D. The substitution bias of the laspeyres price index: An analysis using estimated cost-of-living indexes. **The American Economic Review**, JSTOR, p. 64–77, 1980. Citado na página 51.

CARLOS, L. F. M.; ROSA, J. L. G. Face recognition through a chaotic neural network model. In: IEEE. **2014 International Joint Conference on Neural Networks (IJCNN)**. [S.l.], 2014. p. 859–863. Citado na página 31.

CHANG, C.-C.; LIN, C.-J. **LIBSVM – A Library for Support Vector Machines**. 2016. Disponível em: <<https://www.csie.ntu.edu.tw/~cjlin/libsvm/>>. Citado na página 63.

CHANG, H.-J.; FREEMAN, W. J. Parameter optimization in models of the olfactory neural system. **Neural Networks**, Elsevier, v. 9, n. 1, p. 1–14, 1996. Citado nas páginas 34 e 35.

CHANG, H.-J.; FREEMAN, W. J.; BURKE, B. C. Optimization of olfactory model in software to give 1/f power spectra reveals numerical instabilities in solutions governed by aperiodic (chaotic) attractors. **Neural Networks**, Elsevier, v. 11, n. 3, p. 449–466, 1998. Citado nas páginas 34 e 35.

FERRERO, C. A. **Algoritmo kNN para previsão de dados temporais: funções de previsão e critérios de seleção de vizinhos próximos aplicados a variáveis ambientais em limnologia**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2009. Citado na página 74.

FRANCO, I.; CORRÊA, M.; CARDOSO, F. H.; BARELLI, W.; SANTOS, S. C. dos; STEPANENKO, A.; PEREIRA, A. L.; CANHIM, R. **Plano Real**. 1994. MEDIDA PROVISÓRIA No 434, DE 27 DE FEVEREIRO DE 1994. Disponível em: <http://www.planalto.gov.br/ccivil_03/MPV/1990-1995/434.htm>. Citado na página 49.

FREEMAN, W.; ERWIN, H. Freeman k-set. **Scholarpedia**, v. 3, n. 2, p. 3238, 2008. Citado na página 27.

FREEMAN, W. J. **Mass action in the nervous system: Examination of the neurophysiological basis of adaptive behavior through the EEG**. [S.l.]: Walter J Freeman, 1975. Citado nas páginas 27, 28, 29, 30 e 35.

_____. Tutorial on neurobiology: from single neurons to brain chaos. **International Journal of Bifurcation and Chaos**, World Scientific, v. 2, n. 03, p. 451–482, 1992. Citado na página 28.

_____. **How brains make up their minds**. [S.l.]: Columbia University Press, 2000. Citado na página 28.

FREEMAN, W. J.; CHANG, H.-J.; BURKE, B.; ROSE, P.; BADLER, J. Taming chaos: stabilization of aperiodic attractors by noise [olfactory system model]. **IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications**, IEEE, v. 44, n. 10, p. 989–996, 1997. Citado na página 34.

FREEMAN, W. J.; SHIMOIDE, K. New approaches to nonlinear concepts in neural information processing: Parameter optimization in a largescale, biologically plausible corticle network. In: _____. **An introduction to neural and electronic networks**. [S.l.: s.n.], 1994. p. 119–137. Citado na página 35.

GOOIJER, J. G. D.; HYNDMAN, R. J. 25 years of time series forecasting. **International Journal of Forecasting**, v. 22, n. 3, p. 443–473, jan 2006. Citado nas páginas 56 e 73.

HARTER, D. Evolving neurodynamic controllers for autonomous robots. In: IEEE. **2005 IEEE International Joint Conference on Neural Networks, 2005. IJCNN'05. Proceedings**. [S.l.], 2005. v. 1, p. 137–142. Citado na página 31.

HARVEY, A. C. Trends and cycles in macroeconomic time series. **Journal of Business & Economic Statistics**, Taylor & Francis, v. 3, n. 3, p. 216–227, 1985. Citado na página 23.

HAYKIN, S. S.; HAYKIN, S. S.; HAYKIN, S. S.; HAYKIN, S. S. **Neural networks and learning machines**. [S.l.]: Pearson Upper Saddle River, NJ, USA:, 2009. v. 3. Citado na página 45.

HU, M.; LI, J.; LI, G.; FREEMAN, W. J. Analysis of early hypoxia eeg based on a novel chaotic neural network. In: SPRINGER. **International Conference on Neural Information Processing**. [S.l.], 2006. p. 11–18. Citado na página 31.

HU, M.; LI, J.; LI, G.; TANG, X.; FREEMAN, W. J. Normal and hypoxia eeg recognition based on a chaotic olfactory model. In: SPRINGER. **International Symposium on Neural Networks**. [S.l.], 2006. p. 554–559. Citado na página 31.

KOZMA, R.; AGHAZARIAN, H.; HUNTSBERGER, T.; TUNSTEL, E.; FREEMAN, W. J. Computational aspects of cognition and consciousness in intelligent devices. **IEEE Computational Intelligence Magazine**, IEEE, v. 2, n. 3, p. 53–64, 2007. Citado na página 27.

KOZMA, R.; BELIAEV, I. Time series prediction using chaotic neural networks: case study of IJCNN CATS benchmark test. In: **2004 IEEE International Joint Conference on Neural Networks**. [S.l.]: IEEE, 2004. Citado nas páginas 31, 54 e 55.

KOZMA, R.; FREEMAN, W. J. Chaotic resonance – methods and applications for robust classification of noisy and variable patterns. **International Journal of Bifurcation and Chaos**, World Scientific, v. 11, n. 06, p. 1607–1629, 2001. Citado nas páginas 27, 31, 32 e 34.

_____. Classification of eeg patterns using nonlinear dynamics and identifying chaotic phase transitions. **Neurocomputing**, Elsevier, v. 44, p. 1107–1112, 2002. Citado na página 31.

KOZMA, R.; ROSA, J. L. G.; PIAZENTIN, D. R. Cognitive clustering algorithm for efficient cybersecurity applications. In: IEEE. **The 2013 International Joint Conference on Neural Networks (IJCNN)**. [S.l.], 2013. p. 1–8. Citado na página 31.

LESKOVEC, J.; KREVL, A. **SNAP Datasets: Stanford Large Network Dataset Collection**. 2014. Disponível em: <<http://snap.stanford.edu/data>>. Citado na página 56.

LI, G.; ZHANG, J.; FREEMAN, W. J. Mandarin digital speech recognition based on a chaotic neural network and fuzzy c-means clustering. In: IEEE. **IEEE International Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007**. [S.l.], 2007. p. 1–5. Citado na página 31.

LI, H.; KOZMA, R. A dynamic neural network method for time series prediction using the KIII model. In: IEEE. **2003 IEEE International Joint Conference on Neural Networks**. [S.l.], 2003. v. 1, p. 347–352. Citado nas páginas 29, 53 e 54.

PARMEZAN, A. R. S. **Predição de séries temporais por similaridade**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, 2016. Citado nas páginas 47, 48, 58, 59, 62 e 73.

PARMEZAN, A. R. S.; BATISTA, G. E. **ICMC-USP time series prediction repository**. 2014. Disponível em: <http://sites.labc.icmc.usp.br/icmc_tspr>. Citado na página 74.

_____. A study of the use of complexity measures in the similarity search process adopted by knn algorithm for time series prediction. In: IEEE. **2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)**. [S.l.], 2015. p. 45–51. Citado na página 73.

PIAZENTIN, D. R. M.; FERREIRA, W. Q. **Freeman's K III Reference guide**. [S.l.], 2017. Disponível em: <<https://github.com/wqferr/ksets>>. Citado nas páginas 35 e 36.

PIAZENTIN, D. R. M.; ROSA, J. L. G. Motor imagery classification for brain-computer interfaces through a chaotic neural network. In: IEEE. **2014 International Joint Conference on Neural Networks (IJCNN)**. [S.l.], 2014. p. 4103–4108. Citado na página 31.

_____. A simulator for freeman k-sets in java. In: **2015 International Joint Conference on Neural Networks (IJCNN)**. [S.l.]: IEEE, 2015. Citado nas páginas 34, 54, 55 e 56.

PRINCIPE, J. C.; TAVARES, V. G.; HARRIS, J. G.; FREEMAN, W. J. Design and implementation of a biologically realistic olfactory cortex in analog vlsi. **Proceedings of the IEEE**, IEEE, v. 89, n. 7, p. 1030–1051, 2001. Citado na página 31.

ROSA, J. L. G.; PIAZENTIN, D. R. M. A new cognitive filtering approach based on freeman k3 neural networks. **Applied Intelligence**, Springer Nature, v. 45, n. 2, p. 363–382, mar 2016. Citado nas páginas 30, 31, 35 e 56.

SHIMOIDE, K.; FREEMAN, W. J. Dynamic neural network derived from the olfactory system with examples of applications. **IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences**, The Institute of Electronics, Information and Communication Engineers, v. 78, n. 7, p. 869–884, 1995. Citado na página 35.

STOCK, J. H.; WATSON, M. W. **Implications of Dynamic Factor Models for VAR Analysis**. [S.l.], 2005. Citado na página 23.

VOLTERRA, V.; MATHEMATICS. **Theory of Functionals and of Integral and Integro-Differential Equations (Dover Books on Mathematics)**. [S.l.]: Dover Publications, 1930. Citado na página 57.

XIAOMIN, B.; XIAOQING, N.; YAMING, W.; YANJIANG, Z. Recognition of fabric structures based on improved olfactory neural network. In: IEEE. **2010 International Conference on Artificial Intelligence and Computational Intelligence (AICI)**. [S.l.], 2010. v. 1, p. 325–328. Citado na página 31.

YANG, X.; FU, J.; LOU, Z.; WANG, L.; LI, G.; FREEMAN, W. J. Tea classification based on artificial olfaction using bionic olfactory neural network. In: SPRINGER. **International Symposium on Neural Networks**. [S.l.], 2006. p. 343–348. Citado na página 31.

YULE, G. U. On a method of investigating periodicities in disturbed series, with special reference to wolfer's sunspot numbers. **Philosophical Transactions of the Royal Society of London**, Series A, Containing Papers of a Mathematical or Physical Character, n. 226, p. 267–298, 1927. Citado na página 56.

ZHANG, J.; LI, G.; FREEMAN, W. J. Application of novel chaotic neural networks to mandarin digital speech recognition. In: IEEE. **International Joint Conference on Neural Networks, 2006. IJCNN'06**. [S.l.], 2006. p. 653–658. Citado na página 31.

_____. Application of novel chaotic neural networks to text classification based on pca. In: SPRINGER. **Pacific-Rim Symposium on Image and Video Technology**. [S.l.], 2006. p. 1041–1048. Citado na página 31.

ZHANG, J.; LI, G.; WANG, L.; FREEMAN, W. Application of chaotic neural network on face recognition. In: IEEE. **International Conference on Neural Networks and Brain, 2005. ICNN&B'05**. [S.l.], 2005. v. 3, p. 1367–1372. Citado na página 31.

ZHANG, J.; LIU, X.; LI, R.; MA, Z.; GUO. Improved kiii model and its application. In: IEEE. **7th World Congress on Intelligent Control and Automation, 2008. WCICA 2008**. [S.l.], 2008. p. 2218–2220. Citado na página 31.

ZHANG, J.; LOU, Z.; LI, G.; FREEMAN, W. J. Application of a novel neural network to face recognition based on dwt. In: IEEE. **The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006**. [S.l.], 2006. p. 1042–1046. Citado na página 31.

ZHANG, J.; WANG, Y.; LI, G.; FREEMAN, W. J. Application of bionic neural network on face recognition based on svd and dct. In: IEEE. **The Sixth World Congress on Intelligent Control and Automation, 2006. WCICA 2006**. [S.l.], 2006. v. 1, p. 2733–2736. Citado na página 31.

GLOSSÁRIO

Interação Excitatória: A interação excitatória é aquela cujo o peso é um número positivo, e representa o aumento no potencial de ação do neurônio biológico.

Interação Inibitória: A interação inibitória é aquela cujo o peso é um número negativo, e representa a diminuição no potencial de ação do neurônio biológico.

Peso Excitatório: Peso de uma sinapse cujo valor é maior que zero.

Peso Inibitório: Peso de uma sinapse cujo valor é menor que zero.

População de Neurônios Excitatória: Uma população de neurônios é excitatória quando todas as sinapses que partem dela são excitatórias.

População de Neurônios Inibitória: Uma população de neurônios é inibitória quando todas as sinapses que partem dela são inibitórias.

Sinapse Excitatória: O mesmo que interação excitatória.

Sinapse Inibitória: O mesmo que interação inibitória.