UNIVERSIDADE DE SÃO PAULO

FFCLRP - DEPARTAMENTO DE COMPUTAÇÃO E MATEMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

**Classificação e previsão de séries temporais através de
redes complexas**

Leandro Anghinoni

Dissertação apresentada à Faculdade de Filosofia,
Ciências e Letras de Ribeirão Preto da USP, como
parte das exigências para a obtenção do título de
Mestre em Ciências, Área: Computação Aplicada.

Ribeirão Preto - SP
2018

Classificação e previsão de séries temporais através de redes complexas

Leandro Anghinoni

ORIENTADOR: PROF. DR. ZHAO LIANG

CO-ORIENTADOR: PROF. DR. THIAGO CHRISTIANO SILVA

Versão Revisada

Ribeirão Preto - SP
2018

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Anghinoni, Leandro

Classificação e previsão de séries temporais através de redes complexas. Ribeirão Preto, 2018.

89 p. : il. ; 30 cm

Dissertação de Mestrado, apresentada à Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto/USP. Área de concentração: Computação Aplicada.

Orientador: Liang, Zhao.

1. Séries temporais. 2. Redes complexas. 3. Detecção de comunidades. 4. Classificação de tendência. 5. Previsão de tendência. 6. Aprendizado de máquina

**Time series trend classification and forecasting using
complex network analysis**

Leandro Anghinoni

ADVISOR: PROF. DR. ZHAO LIANG

CO-ADVISOR: DR. THIAGO CHRISTIANO SILVA

Revised Version

Ribeirão Preto - SP
2018

Leandro Anghinoni

Classificação e previsão de séries temporais através de redes complexas

Dissertação apresentada à Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto da USP, como parte das exigências para a obtenção do título de Mestre em Ciências, Área: Computação Aplicada.

Aprovado em: 06 de novembro de 2018

Banca Examinadora

Prof. Dr. Zhao Liang
Orientador

Prof. Dr. Renato Tinós

Prof. Dr. Tabajara Pimenta Junior

Prof. Dr. Elbert Einstein Nehrer Macau

Ribeirão Preto - SP
2018

À minha esposa Tais.

AGRADECIMENTOS

Agradeço à minha família pelo apoio incondicional a qualquer iniciativa que busque novos conhecimentos e ao Prof. Dr. Zhao Liang, meu orientador, pela oportunidade, incentivo e confiança durante o desenvolvimento do trabalho.

“Those who can imagine anything, can create the impossible.”

Alan Turing

RESUMO

O estudo de séries temporais para a geração de conhecimento é uma área que vem crescendo em importância e complexidade ao longo da última década, à medida que a quantidade de dados armazenados cresce exponencialmente. Considerando este cenário, novas técnicas de mineração de dados têm sido constantemente desenvolvidas para lidar com esta situação. Neste trabalho é proposto o estudo de séries temporais baseado em suas características topológicas, observadas em uma rede complexa gerada com os dados da série temporal. Especificamente, o objetivo do modelo proposto é criar um algoritmo de detecção de tendências para séries temporais estocásticas baseado em detecção de comunidades e caminhadas nesta mesma rede. O modelo proposto apresenta algumas vantagens em relação à métodos tradicionais, como o número adaptativo de classes, com força mensurável, e uma melhor absorção de ruídos. Resultados experimentais em bases artificiais e reais mostram que o método proposto é capaz de classificar as séries temporais em padrões locais e globais, melhorando a previsibilidade das séries ao se utilizar métodos de aprendizado de máquina para a previsão das classes.

Palavras-chave: séries temporais, redes complexas, detecção de comunidades, classificação de tendência, previsão de tendência, aprendizado de máquina.

ABSTRACT

Extracting knowledge from time series analysis has been growing in importance and complexity over the last decade as the amount of stored data has increased exponentially. Considering this scenario, new data mining techniques have continuously developed to deal with such a situation. In this work, we propose to study time series based on its topological characteristics, observed on a complex network generated from the time series data. Specifically, the aim of the proposed model is to create a trend detection algorithm for stochastic time series based on community detection and network metrics. The proposed model presents some advantages over traditional time series analysis, such as adaptive number of classes with measurable strength and better noise absorption. Experimental results on artificial and real datasets shows that the proposed method is able to classify the time series into local and global patterns, improving the predictability of the series when using machine-learning methods.

Keywords: time series, complex networks, community detection, trend classification, trend forecasting, machine learning.

LISTA DE FIGURAS

Figura 1. Exemplo de rede neural (MLP) feed-forward. Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).	35
Figura 2. Separação de duas classes utilizando o algoritmo SVM. Note que o hiperplano ótimo maximiza a distância de separação entre as duas classes e utiliza os pontos sobre as bordas como pontos de suporte. Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).	37
Figura 3. Exemplo de árvore decisão gerada a partir da base de treinamento apresentada na Tabela 1. Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).	38
Figura 4. Exemplo de redes com comunidades destacadas. Na rede à esquerda nota-se a presença de três comunidades, definidas por áreas de maior densidade de conexão. Na rede à direita observam-se duas comunidades. Fonte: (BARABÁSI, 2016).	41
Figura 5. Série temporal exemplo e sua rede de transição de estados discretos	43
Figura 6. Conjunto de séries temporais e sua rede de transição derivada	43
Figura 7. Fluxograma das etapas do modelo proposto de classificação da série temporal	48
Figura 8. Fluxograma das etapas do modelo proposto de previsão de tendência da série temporal	49
Figura 9. Série temporal composta do preço de fechamento do índice Bovespa de 02/01/1996 a 08/05/2018.	50
Figura 10. Série temporal artificial utilizada no trabalho. A figura ilustrada aqui não possui ruído.....	51
Figura 11. (a) Série temporal original correspondente a 50 dias do preço de fechamento do índice Bovespa; (b) Características de curto prazo normalizadas (ruído em vermelho, gradiente em azul e posição relativa máximo-mínimo em verde; (c) Características de curto prazo normalizadas e discretizadas (mesmo padrão de cores).....	54
Figura 12. Representação ilustrativa da rede clusterizada em duas classes. Fonte: (BARABÁSI, 2016)	56
Figura 13. Exemplo de classificação de comunidade identificada no processo de detecção de comunidades. Neste caso a comunidade 2741 está destacada e algumas ocorrências podem ser vistas no período plotado. A comunidade 2741 se trata de uma comunidade de baixa.	59
Figura 14. Exemplo de dados inseridos nos classificadores utilizados nos experimentos.	60

Figura 15. Série temporal interpolada em intervalos simétricos. Os dados da série são classificados de acordo com o ponto inicial e final de cada segmento de reta. Nesta figura as tendências de alta estão marcadas em azul e as tendências de baixa em vermelho.	62
Figura 16. Desvio padrão da curva real - índice Bovespa	66
Figura 17. Histograma do grau (k) dos vértices da rede gerada pela curva artificial	67
Figura 18. Rede gerada a partir da curva artificial	67
Figura 19. Histograma do grau (k) dos vértices da rede gerada pela curva artificial	69
Figura 20. Rede gerada a partir da curva real	69
Figura 21. Curva sinusoidal classificada pelo modelo proposto em diferentes níveis de ruído. Em (a) $\sigma^2 = 0,00$, em (b) $\sigma^2 = 0,01$, em (c) $\sigma^2 = 0,02$ e em (d) $\sigma^2 = 0,03$	70
Figura 22. (a) Acurácia de classificação em função do ruído presente na curva sinusoidal. (b) Modularidade da rede com base no ruído presente na curva sinusoidal.	71
Figura 23. Performance de classificação do modelo proposto e do método de particionamento da base e interpolação dos intervalos.	72
Figura 24. Últimos 1000 pontos da série temporal real classificados de acordo com as comunidades identificadas na rede gerada.	73
Figura 25. Últimos 1000 pontos da série temporal real classificados de acordo com a tendência de alta ('UP') ou baixa ('DW').	76
Figura 26. Rede gerada a partir da curva real e classificada nas tendências de alta (azul) e baixa (vermelho).	76
Figura 27. Acurácia de previsão para a curva artificial utilizando como regra de classificação a interpolação dos dados com particionamento da base em períodos de 25 dias..	78
Figura 28. Acurácia de previsão para a curva artificial utilizando como regra de classificação a interpolação dos dados com particionamento da base em períodos de 125 dias.	78
Figura 29. Acurácia de previsão para a curva artificial utilizando como regra de classificação o modelo proposto baseado na topologia da rede.	79
Figura 30. Acurácia de previsão combinada para a curva artificial.	79
Figura 31. Acurácia de classificação para o método de referência utilizando um tamanho de partição variável.	80
Figura 32. Acurácia de classificação para o modelo proposto utilizando um período de curto prazo variável e uma relação longo prazo/curto prazo fixa.	81
Figura 33. Acurácia de classificação combinada.	81

LISTA DE TABELAS

Tabela 1. Exemplo de dados de treinamento para a construção da árvore de decisão. Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).	38
Tabela 2. Dados da rede criada a partir da curva artificial.....	66
Tabela 3. Dados da rede criada a partir da curva real.....	68
Tabela 4. Dados da rede gerada com detalhes sobre as comunidades	74

LISTA DE SÍMBOLOS

Notação	Descrição
srt	Tamanho, em dias, do período de curto prazo
lng	Tamanho, em dias, do período de longo prazo
N_b	Tamanho do <i>bin</i> utilizado
t	Período de tempo da observação do processo estudado
T	Espaço de tempo total do processo estudado
G	Grafo
n	Vértice
l	Aresta
k	Grau do vértice
k_{out}	Grau do vértice considerando apenas arestas que partem do vértice
M	Matriz de adjacência
Q	Modularidade da rede
S	Série característica
v	Vetor característico
cp	Série temporal do preço de fechamento do índice Bovespa

SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO.....	23
1.1 Séries Temporais	23
1.2 O Índice Bovespa e os Mercados Eficientes.....	25
1.3 Motivação.....	26
1.4 Definição do Problema e Objetivos	27
1.5 Organização do Trabalho.....	29
CAPÍTULO 2 - FUNDAMENTAÇÃO TEÓRICA.....	31
2.1 Séries Temporais e o Modelo ARIMA	31
2.2 Algoritmos de Aprendizado de Máquina ou Classificadores	34
2.2.1 <i>Naive Bayes</i> (NB)	34
2.2.2 <i>Multilayer Perceptron</i> (MLP).....	35
2.2.3 <i>Support Vector Machine</i> (SVM)	36
2.2.4 <i>Decision Tree</i> (DT)	37
2.2.5 <i>k-Nearest Neighbors</i> (k-NN)	39
2.3 Redes Complexas	39
2.3.1 Redes Reais e suas Propriedades.....	40
2.3.2 Redes Artificiais e Modelos de Formação.....	41
2.3.3 Estrutura da Rede – Comunidades	44
2.4 Considerações Gerais	45
CAPÍTULO 3 - CLASSIFICAÇÃO E PREVISÃO DE SÉRIES TEMPORAIS ATRAVÉS DE REDES COMPLEXAS	47
3.1 Considerações Iniciais.....	47
3.2 Obtenção e Tratamento dos Dados	49
3.3 Extração das Características da Série Original	51
3.4 Conversão da Série Temporal em Rede.....	55
3.5 Detecção de Comunidades	55
3.6 Classificação da Tendência da Série Temporal	58
3.7 Previsão de Tendência de uma Nova Observação	60
3.8 Análise dos Resultados.....	61

3.9 Comparação com Métodos Existentes	61
3.10 Considerações Finais	63
CAPÍTULO 4 - RESULTADOS EXPERIMENTAIS	65
4.1 Métricas da Rede Gerada	65
4.1.1 Métricas da Rede Gerada – Curva Artificial	65
4.1.2 Métricas da Rede Gerada – Curva Real (Índice Bovespa)	68
4.2 Classificação da Tendência com o Modelo Proposto	69
4.2.1 Classificação da Tendência com o Modelo Proposto – Curva Artificial	70
4.2.2 Classificação da Tendência com o Modelo Proposto – Curva Real	71
4.3 Previsão da Tendência com o Modelo Proposto	77
4.3.1 Previsão da Tendência com o Modelo Proposto – Curva Artificial	77
4.3.2 Previsão da Tendência com o Modelo Proposto – Curva Real	80
CAPÍTULO 5 - CONCLUSÕES	83
5.1 Publicações	84
5.2 Trabalhos Futuros	85
REFERÊNCIAS	87

Capítulo 1

INTRODUÇÃO

O presente capítulo apresentará uma breve introdução sobre séries temporais e o índice Bovespa, cobrindo a importância do estudo deste tipo de série temporal. Em seguida será apresentada a motivação deste trabalho e a definição do problema e objetivos.

1.1 Séries Temporais

A análise de séries temporais faz parte do dia a dia da maioria das áreas de estudos e dos negócios atuais (BAHETI; TOSHNIWAL, 2014). Sua aplicação abrange os mais variados campos de pesquisa, como por exemplo:

- Economia: evolução de dados como PIB, juros, taxa de cambio e cotações de ativos;
- Industria: consumo energético, produtividade e taxa de qualidade;
- Meteorologia: evolução de variáveis climáticas, como temperatura e humidade;
- Medicina: evolução de contaminação de uma doença, batimentos cardíacos e temperatura corporal.

Uma série temporal pode ser definida como uma sequência de medidas coletadas ao longo do tempo, normalmente em intervalos regulares (RANI *et. al.*, 2014). Na série temporal a ordem dos dados é importante para que a análise tenha significado. O estudo de dados históricos tem a seguinte relevância:

- Descrição: as séries temporais descrevem o comportamento de uma variável ao longo do tempo;

- Entendimento: através desta descrição é possível estudar características como sazonalidade, ciclos, tendência, previsibilidade, correlação entre séries, volatilidade, dados coletados errados, entre outras;
- Previsão: através do entendimento da série pode-se propor um modelo de previsão do fenômeno observado.

Estima-se que mais dados foram gerados nos últimos dois anos do que em toda a história anterior da humanidade e que, apesar disso, apenas 0,5% de todos os dados é analisado de alguma forma. O trabalho de *data-mining* ou mineração de dados pode ser definido como o processo de extração de modelos e padrões de grandes bases de dados. Este trabalho em séries temporais pode ser abordado de diversas formas e o desenvolvimento de novas técnicas se torna cada vez mais importante neste contexto. (RANI *et. al.*, 2014) descreve várias técnicas para o tratamento de séries temporais compostas de um grande número de dados. O modelo geral para a análise de séries temporais pode ser dividido em quatro passos: definição do problema, pré-processamento dos dados, criação do modelo e, por fim, análise e predição do modelo.

Ainda segundo (RANI *et. al.*, 2014), a análise da tendência da série temporal (direção na qual os dados evoluem) possui alguns componentes que podem ajudar na análise e no estudo das séries temporais:

- Movimentos de longo prazo: mostram a direção (ou tendência) em que os dados estão indo. Normalmente é obtida pela média móvel dos dados ou pela soma dos mínimos quadrados (regressão polinomial);
- Movimentos sazonais: incluem padrões que estão relacionados ao calendário anual, ou seja, padrões que se repetem em determinados dias, semanas ou meses. Um exemplo disto seria um aumento de vendas em dezembro, em decorrência do Natal;
- Movimentos aleatórios: são comportamentos da curva não previsíveis, como enchentes e greves, ou seja, eventos que tem um impacto na série, mas que não possuem uma previsibilidade histórica;
- Movimentos cíclicos: são padrões que se repetem em tempos em tempos, mesmo que escalas diferentes e não relacionadas ao calendário.

Desta forma, a série temporal poderia ser simplificada como a soma ou produto de cada componente, representado na Equação (1):

$$Y = T + S + R + C \text{ ou } Y = T * S * R * C \quad (1)$$

Onde Y é a função que representa a série temporal, T é componente do movimento de longo prazo (*trend*), S é a componente dos movimentos sazonais (*seasonal*), R é a componente aleatória (*random*) e C é a componente cíclica (*cyclic*).

O objetivo final de um sistema de *data-mining* é encontrar padrões contidos na série original ou transformada. O processo de mineração dos dados envolve diversas atividades, como o agrupamento de dados, a classificação, a mineração por regras, a sumarização, entre outras. Este trabalho fará o uso de várias dessas técnicas para a formulação do modelo proposto.

1.2 O Índice Bovespa e os Mercados Eficientes

A principal base de dados de estudo deste trabalho será o preço de fechamento do índice Bovespa ao longo dos últimos 22 anos. O índice Bovespa tem sido o indicador mais importante para medir a performance das empresas de capital aberto brasileiras ao longo dos últimos 50 anos. Sua metodologia foi criada em 1962 e, após passar por ajustes em 1966 e 1967, o índice foi adotado pela bolsa de valores de São Paulo em 1968. Desde então, o índice passou por momentos de estabilidade, mas também por fortes altas e depressões (LEITE; SANVINCENTE, 1995). Do ponto de vista de um investidor ou de um analista de mercado, entender e prever esses momentos de alta e baixa do índice são de grande valor.

No entanto, a hipótese dos mercados eficientes afirma que os mercados são eficientes em relação à informação, ou seja, um agente não pode ter um retorno maior que o mercado no longo prazo utilizando apenas a informação pública disponível (MALKIEL; FAMA, 1970). Do ponto de vista da análise de séries temporais, isso significa que não seria possível um modelo gerar previsões futuras apenas com dados passados, mesmo sendo um passado de curtíssimo prazo.

Existem três versões da hipótese dos mercados eficientes:

- Hipótese fraca: considera que o preço reflete toda a informação histórica disponível sobre o ativo;
- Hipótese semiforte: considera que o preço reflete toda a informação histórica disponível sobre o ativo e que qualquer nova informação pública é refletida instantaneamente nos preços do ativo;

- Hipótese forte: considera que parte do mercado (como investidores institucionais) possui acesso à informação privilegiada, que interfere na formação do preço do ativo.

Desde que a hipótese foi formulada e, especialmente, depois da depressão das bolsas mundiais de 2008, muitos argumentos têm sido levantados em relação à esta teoria, principalmente em relação à quantidade de informação privilegiada existente no mercado e o impacto disto no conceito de retorno esperado ou *fair game*. O conceito de retorno esperado ou *fair game* considera que, em um mercado em equilíbrio (com todas as informações refletidas nos preços), não é possível que um modelo que considere apenas dados passados tenha um retorno maior do que o do próprio ativo no longo prazo (MALKIEL; FAMA, 1970). Desta forma, este trabalho pode ser visto também como um teste empírico desta hipótese.

Neste trabalho o índice Bovespa será tratado como uma série temporal estocástica, sem uma componente determinística conhecida. Esta abordagem traz alguns desafios para a criação de um modelo de classificação e previsão, especialmente em relação ao tratamento do ruído presente nas tendências e ao conceito de curto, médio e longo prazo.

1.3 Motivação

Compreender o mercado financeiro não é uma tarefa simples. Muitos modelos tradicionais, propostos nas últimas décadas, consideram o mercado como racional, no sentido que o valor dos ativos é definido por um modelo lógico (MALKIEL; FAMA, 1970). Na prática, flutuações e ruídos nestes tipos de séries temporais mostram que o valor atribuído à um ativo é bastante volátil e que o comportamento, ou até mesmo as emoções, dos agentes envolvidos geram uma grande imprevisibilidade quando analisamos os dados.

Considerando essa premissa, o estudo da série temporal através de redes complexas pode gerar um melhor entendimento destes estados ao longo do tempo, permitindo uma classificação mais assertiva do momento em que o mercado se encontra, já que as redes podem absorver características recorrentes de um processo de uma forma mais objetiva, como será descrito mais adiante. Desde o início da análise técnica dos mercados financeiros, é comum a classificação do mercado em três estados: (i) alta, quando a curva dos dados possui um gradiente positivo; (ii) baixa, quando a curva dos dados possui um gradiente negativo ou (iii) consolidação, quando a curva dos dados possui um gradiente próximo de zero, ou seja, os

valores variam em torno de uma constante. No entanto, através do estudo da série pela topologia da rede construída a partir dela, espera-se que seja possível classificar a tendência de cada ponto da curva em um número maior de classes, de acordo com o número de comunidades presentes na rede. Além disso, as tendências baseadas em comunidades poderão ser avaliadas em relação à sua intensidade e probabilidade de reversão, já que cada ponto da série temporal será representado por um vértice e diversas medidas da rede poderão ser avaliadas.

Espera-se, por exemplo, que a intensidade de conexão dos vértices gere informação sobre como os ciclos de alta e baixa se repetem e como eles iniciam e terminam. Com isto, além da classificação histórica dos dados, um modelo de previsão será proposto com base em modelos de aprendizado de máquina.

1.4 Definição do Problema e Objetivos

A classificação e previsão de séries temporais podem ser divididas de duas formas: (i) classificação e previsão de valores ou (ii) classificação e previsão de classes ou tendências (LI; LIAO, 2017).

A previsão de valores usualmente trata de uma tarefa de regressão, onde o seu modelo mais conhecido para o estudo de séries estocásticas é o modelo ARIMA (detalhado no Capítulo 2). Os modelos de regressão normalmente se ajustam aos dados históricos e fazem a previsão em forma de um valor e uma margem de erro. Diversos trabalhos foram publicados utilizando o modelo ARIMA, ou alguma variante dele, para o estudo de séries estocásticas, como em (TSENG *et. al.*, 2001), (CHEN *et al.*, 2010) e (HILLMER; TIAO, 1982). No entanto, o modelo ARIMA não captura padrões não lineares facilmente e, por isto, alguns trabalhos mais recentes propõe o uso do ARIMA combinado com técnicas como SVM (PAI; LIN, 2005) e redes neurais (ZHANG, 2003).

Já a previsão de tendências (ou classes) é uma tarefa de classificação e, portanto, algoritmos de classificação como o *Naive Bayes*, o *Multilayer Perceptron*, as Árvore de Decisão e assim por diante podem ser selecionados para resolver o problema. No entanto, para que esses classificadores tenham uma boa performance é necessário que a base esteja classificada corretamente. Isto por si só é um grande desafio quando se trata de uma série estocástica, já que o horizonte de tempo de cada classe e o ruído inerente de séries estocásticas

tornam o processo de classificação da base algo bastante subjetivo, o que impacta diretamente nos resultados de previsão.

Em (LI; LIAO, 2017), por exemplo, o autor utiliza seis classificadores (Árvore de Decisão, *Naive Bayes*, *Support Vector Machine*, *Multilayer Perceptron*, Redes Neurais Recorrentes e *Long Short-Term Memory*) para prever a tendência de um ativo da bolsa de valores na China. Os resultados são apresentados e, apesar dos classificadores terem performances diferentes (com uma melhor performance de acurácia para os modelos de aprendizagem profunda MLP, RNN e LSTM), os melhores resultados ainda ficam abaixo de 50% para a acurácia.

No exemplo citado acima, a base é rotulada utilizando uma regra pré-definida e é passado para os classificadores uma base com diversos indicadores diários e o rótulo. Os indicadores são gerados com base nos dados históricos (são utilizados indicadores de análise técnica de mercado financeiro). O rótulo (que representa a tendência do dia) é gerado com base na variação diária do ativo, ou seja, se o preço de fechamento do dia é maior que o dia anterior o dia é classificado como “alta”, se for menor como “baixa” e se for igual como “consolidação”. Desta forma, a classificação da série não absorve nenhum ruído, já que considera todas as variações diárias e torna o trabalho de previsão muito difícil já que é preciso que a previsão da classe seja precisa para o movimento diário.

A utilização de janelas maiores, apesar de resolver o problema de absorção de ruídos, não resolve o problema de forma completa por dois motivos: (i) o tamanho da janela fica a critério do usuário e (ii) os dados contidos em cada janela não possuem correlação com os dados dos indicadores históricos.

Diante dos problemas expostos, o objetivo deste trabalho é desenvolver um modelo de conversão de séries temporais em redes complexas que capture características subjacentes dos dados e classifique os mesmos através da detecção de comunidades. A análise da série temporal através de suas características topológicas gera um melhor entendimento do comportamento do mercado, uma vez que cada vértice concentra conhecimento de um mesmo estado ao longo do tempo. Esta nova proposta buscará resolver algumas das dificuldades dos métodos tradicionais, que em sua maioria, têm dificuldade em reagir rápido à uma inversão de tendência ou desconsiderar ruídos que não alteram a tendência futura. Mais especificamente, serão buscadas as seguintes propriedades no modelo proposto:

- Número de classes (ou tendências): o modelo não deve se restringir a três classes (alta, baixa e consolidação) e sim ter um número de classes correspondente ao número de comunidades presente na rede que representa a série temporal;

- Classificação da série temporal: a classificação não deve ser feita com base em um horizonte de tempo definido, mas sim com base nas comunidades presentes na rede. Assim, as tendências serão atribuídas após a formação das comunidades e as tendências poderão ter duração e intensidade variáveis e mensuráveis;
- Previsão da série temporal: as classes deverão ser agrupadas de acordo com padrões históricos da série estudada, com isso as características diárias dos dados terão relação com a classe atribuída ao dia. Desta forma espera-se que a previsão através do uso de algoritmos de aprendizado de máquina tenha um resultado superior ao da previsão de classes atribuídas manualmente (o processo de classificação manual será explicado na Seção 3.9).

1.5 Organização do Trabalho

O Capítulo 2, intitulado “Fundamentação Teórica”, apresenta uma revisão sobre os conceitos de séries temporais e o modelo ARIMA, um modelo de regressão muito utilizado. Em seguida é feita uma revisão sobre alguns classificadores que serão utilizados neste trabalho e por fim são apresentados os conceitos de redes complexas e sua estrutura.

No Capítulo 3, intitulado “Classificação e Previsão de Séries Temporais através de Redes Complexas”, é apresentado o modelo proposto. Este capítulo é dividido em cada etapa do modelo proposto, desde a fase de tratamento dos dados, até a metodologia de análise dos resultados obtidos.

Já no Capítulo 4, intitulado “Resultados”, serão apresentados os resultados experimentais obtidos com o modelo proposto. Como será explicado ao longo do trabalho, o modelo proposto será comparado com um método de referência.

Por fim, o Capítulo 5, intitulado “Conclusões”, apresenta as conclusões com base nos resultados obtidos e contribuições do trabalho. Além disso, é feito um pequeno resumo sobre possíveis trabalhos futuros.

Capítulo 2

FUNDAMENTAÇÃO TEÓRICA

O presente capítulo apresentará a fundamentação teórica do trabalho desenvolvido. Neste capítulo são apresentados os principais conceitos utilizados na formulação do modelo proposto para abordar o problema apresentado no Capítulo 1.

2.1 Séries Temporais e o Modelo ARIMA

Além da forma conceitual apresentada na equação (1), uma série temporal pode ser generalizada pela Equação (2), onde $y(t)$ é o valor do processo observado ao longo do tempo, $\alpha x(t)$ é a componente determinística do processo, representado por um polinômio de ordem n e $\varphi(t)$ é o ruído do processo, ou componente estocástica (EHLERS, 2009). O processo $y(t)$ varia de acordo com o tempo t e dentro do espaço de tempo total observado T .

$$y(t) = \alpha x(t) + \varphi(t), \quad t \in T \quad (2)$$

Um processo pode ser regido por uma função totalmente determinística ($\varphi(t) = 0$, $\forall t \in T$) e neste caso a análise e previsão do processo se dá com base nesta função (neste caso o método dos mínimos quadrados encontrará um polinômio com erro próximo de zero e que se ajustará a observações futuras com o mesmo nível de erro). No entanto, os processos reais costumam se apresentar na forma mista, ou seja, possuem uma função que rege o processo, com algum tipo de ruído aleatório. Em alguns casos mais particulares, o processo possui apenas a componente estocástica. Este último caso pode ser visto de duas formas: (i) o processo realmente é aleatório ou (ii) o processo possui uma função determinística desconhecida ou muito complexa para ser captada pelos métodos atuais.

Os processos estocásticos podem ser divididos em estacionários ou divergentes. Nos processos estocásticos estacionários, de sentido amplo, o processo possui uma média constante e uma distribuição característica, como por exemplo a distribuição normal. No exemplo apresentado na Equação (3) a função possui um ruído com média constante μ e distribuição normal σ^2 em torno da média.

$$\varphi \sim N(\mu, \sigma^2) \quad (3)$$

Uma série temporal de preços de uma ação pode ser vista como uma série estocástica não estacionária, sem componente determinística. A princípio, a melhor estimativa de um valor futuro seria o valor atual, ou seja, uma abordagem ingênua (*naive*).

No entanto, diversos métodos foram criados para tentar tornar este tipo de série estacionária, permitindo algum tipo de previsão dos dados. Um método muito popular utilizado para a previsão de valores deste tipo de série é o modelo auto regressivo integrado de médias móveis (*Autoregressive Integrated Moving Average* ou ARIMA). Diversos trabalhos utilizam este modelo ou alguma variante dele, como em (TSENG *et al.*, 2001), (CHEN *et al.*, 2010) e (HILLMER; TIAO, 1982). O modelo ARIMA é uma evolução dos modelos AR e MA.

Em um modelo auto regressivo (AR) a função $y(t)$ é aproximada por valores passados da série, ou seja, o valor atual é uma consequência das observações passadas do processo ponderadas por coeficientes β_t e um ruído residual φ . O número de termos passados utilizados na função é chamado de ordem do processo (p). Assim, o processo AR é denotado como AR(p) (EHLERS, 2009), conforme mostrado na Equação (4).

$$y(t) = \beta_0 + \beta_1 y(t-1) + \beta_2 y(t-2) + \dots + \beta_p y(t-p) + \varphi \quad (4)$$

Em um modelo de médias móveis (MA) a função $y(t)$ é aproximada pela soma dos ruídos passados do processo ponderados por coeficientes θ_t . Novamente, o número de termos passados utilizados é a ordem do processo, neste caso (q), e o processo é denotado como MA(q) (EHLERS, 2009), conforme mostrado na Equação (5).

$$y(t) = \theta_0 + \theta_1 \varphi(t-1) + \theta_2 \varphi(t-2) + \dots + \theta_q \varphi(t-q) \quad (5)$$

A combinação dos processos $AR(p)$ e $MA(q)$ gera um processo $ARMA(p, q)$ definido pela soma das funções (EHLERS, 2009), mostrado na Equação (6).

$$y(t) = \beta_0 + \beta_1 y(t-1) + \dots + \beta_p y(t-p) + \varphi + \theta_0 + \theta_1 \varphi(t-1) + \dots + \theta_q \varphi(t-q) \quad (6)$$

Um modelo $ARMA(p, q)$, no entanto, é adequado apenas para o estudo de séries temporais estocásticas estacionárias. Uma série divergente (como o caso dos preços de uma ação) pode ser transformada em uma série estacionária através da diferenciação dos dados. Deve-se notar que nem toda série pode ser transformada em uma série estacionária. Para alguns casos a série não se tornar estacionária, independentemente do número de diferenciações. A diferenciação consiste em considerar a diferença entre uma observação e outra (Equação (7)).

$$y(t) = \frac{dy}{dx} \text{ ou } y_t = y(t) - y(t-1) \quad (7)$$

Uma série que já se encontra na forma estacionária é da ordem $d = 0$, uma série que se torna estacionária após uma diferenciação é da ordem $d = 1$ e assim por diante. A série se torna estacionária quando os dados assumem uma distribuição característica, como a distribuição normal (3). Portanto, o processo $ARIMA$ é denotado como $ARIMA(p, d, q)$, onde d é a ordem de diferenciação do processo estudado (EHLERS, 2009). No caso do índice Bovespa, por exemplo, a série dos preços das ações é divergente, porém com $d = 1$ ela se torna estacionária. Diferenciar uma vez esta série corresponde a considerar a variação diária dos preços e não o preço do índice.

Para a previsão de dados futuro utilizando o modelo $ARIMA$, o método mais conhecido é o *Box-Jenkins* que consiste em avaliar todos os parâmetros do processo $ARIMA$ para o processo estudado e com base na função aproximada estimar os valores futuros. A previsão neste formato possui uma faixa de erro e um intervalo de confiança e o dado previsto é o valor da função.

No entanto, o modelo $ARIMA$ não captura padrões não lineares facilmente e, em virtude disto, alguns trabalhos mais recentes propuseram a combinação do processo $ARIMA$ com técnicas modernas de aprendizado de máquina, como o *Support Vector Machine* (PAI; LIN, 2005) e redes neurais (ZHANG, 2003).

Além disto, a previsão de valores com uma margem de erro, apesar de ser de muito valor para muitos tipos de processo, no caso do índice Bovespa acaba sendo menos relevante do que a previsão da classe de tendência, já que esta não assume uma margem de erro.

2.2 Algoritmos de Aprendizado de Máquina ou Classificadores

A classificação de tendências pode ser abordada como uma classificação discreta de um conjunto de dados em n classes. Os classificadores, diferentemente dos processos de regressão, exigem que a base esteja previamente agrupada em classes. A partir de características e a classe de cada observação é possível criar um modelo de previsão.

Os algoritmos de classificação supervisionados também são conhecidos como algoritmos de indução, pois a previsão de um novo dado é induzida com base na classificação dos dados conhecidos. Cada instância, neste caso, é representada pelo mesmo conjunto de características, sejam elas contínuas, categóricas ou binárias (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

Neste trabalho cinco modelos serão utilizados na fase de previsão das classes do modelo proposto. Estes cinco modelos são descritos a seguir. Dado que o foco deste trabalho é o estudo de séries temporais através de redes complexas, diversos aspectos dos algoritmos de aprendizado de máquina não serão cobertos em detalhes, como seleção da base, *overfitting*, seleção dos algoritmos, *ensembles*, etc. Os cinco algoritmos foram selecionados por serem conceitualmente diferentes entre si e por serem muito utilizados em trabalhos semelhantes na fase de avaliação da capacidade de previsão de uma base de dados rotulada. A revisão dos cinco algoritmos escolhidos foi feita com base em (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

2.2.1 Naive Bayes (NB)

Redes Bayesianas ingênuas são modelos simples de redes Bayesianas, que são compostas por grafos acíclicos, direcionados, com um nó pai (representando o nó não observado) e diversos nós filhos (nós observados), onde se assume a independência entre os nós filhos. Desta forma, a classe do nó não observado é dada pela Equação (8), onde i e j são, respectivamente, as classes do problema:

$$R = \frac{P(i|X)}{P(j|X)} = \frac{P(i)P(X|i)}{P(j)P(X|j)} \quad (8)$$

Se $R > 1$, a classe é prevista como i , caso contrário a classe é prevista como j .

A maior vantagem deste método é o tempo computacional baixo para treinamento com grandes bases de dados. Por outro lado, a premissa de não dependência entre os atributos impõe uma grande restrição para diversos problemas.

2.2.2 Multilayer Perceptron (MLP)

O *Multilayer Perceptron*, um tipo de rede neural artificial, foi criado para tentar resolver problemas que não são separáveis linearmente no espaço. Uma rede neural multicamadas consiste em um grande número de unidades (neurônios) conectados em um certo padrão. As unidades são geralmente separadas em três tipos: unidades de entradas, que recebem a informação a ser processada; unidades de saída, que mostram o resultado computado; e as camadas intermediárias, também conhecidas como camadas ocultas. A Figura 1 mostra um exemplo simples de uma rede neural *feed-forward*, onde o sinal é propagado em apenas uma direção, da unidade de entrada para a unidade de saída.

Figura 1. Exemplo de rede neural (MLP) *feed-forward*. Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

Este tipo de rede é treinada utilizando um conjunto de dados para determinar o mapeamento entre a entrada e a saída. Os pesos das arestas entre os neurônios são fixados e a rede é usada para determinar a classificação de um novo conjunto de dados.

Durante a classificação o sinal é propagado das unidades de entrada até as unidades de saída para determinar os valores de ativação das unidades de saída. Cada unidade de entrada tem um valor de ativação que representa uma característica externa da rede (ou uma característica do problema estudado). Cada unidade de entrada envia o seu valor de ativação para a unidade oculta à qual está conectada e cada unidade oculta calcula seu valor de ativação e o envia à unidade de saída (ou à unidade oculta à que estiver conectada). O valor de ativação é calculado usando uma função de ativação que soma a contribuição de todas as unidades que enviam algum sinal (a contribuição e o valor de ativação da unidade multiplicado pelo peso da aresta por qual o sinal passa). Esta soma é normalmente modificada para ficar entre 0 e 1 ou para ser 0, a não ser que um certo limite seja atingido.

As redes neurais dependem de três aspectos fundamentais: a entrada e a função de ativação da unidade, a arquitetura da rede e o peso de cada conexão. Como os dois primeiros parâmetros são fixos, o comportamento da rede será determinado pelo ajuste dos pesos das conexões. Desta forma a rede normalmente é iniciada com valores aleatórios de pesos, que são ajustados através de sucessivas iterações e medidas de erro resultante.

Para ajustar os pesos (e assim reduzir o erro) o método mais conhecido é o *back-propagation*, que retorna o erro final para a rede e ajusta os pesos até que o erro final seja otimizado conforme o objetivo do usuário.

2.2.3 Support Vector Machine (SVM)

Os SVMs se baseiam na noção da maximização de uma margem a partir de um hiperplano que separa linearmente duas classes. Maximizar a distância entre o hiperplano e os dados de cada classe tem se provado uma boa estratégia para diminuir o erro durante o processo de generalização do modelo.

No caso de as classes estudadas serem linearmente separáveis, uma vez que o hiperplano ótimo é encontrado, os pontos que se encontram sobre as margens são conhecidos como pontos de suporte (*support vector points*) e a solução é apresentada como uma combinação somente destes pontos. Isso faz com que a complexidade dos SVMs não seja impactada pela quantidade de classes da base de treinamento, já que a quantidade de vetores de suporte encontrados normalmente é baixa.

O conceito simplificado (duas classes separáveis linearmente) pode ser visto na Figura 2 e, por se tratar de um modelo matematicamente extenso, sugere-se a leitura de

(KOTSIANTIS; ZHARAKIS; PINTELAS, 2007) para detalhes de implementação e formulação matemática.

Figura 2. Separação de duas classes utilizando o algoritmo SVM. Note que o hiperplano ótimo maximiza a distância de separação entre as duas classes e utiliza os pontos sobre as bordas como pontos de suporte.
Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

2.2.4 Decision Tree (DT)

As árvores de decisão são algoritmos que, assim como todos os demais, classificam as instâncias separando-as pelos valores de suas características. No entanto, as árvores de decisão fazem isto de forma explícita, produzindo como resultado final uma classificação possível de ser entendida passo a passo. Cada nó da árvore representa uma divisão em uma das características da instância a ser classificada e cada ramificação representa qual valor a instância pode assumir. Cada nó gera, portanto, um hiperplano no espaço, que separa as instâncias em dois grupos a cada decisão (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

Construir uma árvore binária de decisão ótima é um problema NP-completo e, portanto, os métodos desenvolvidos se utilizam da heurística para desenvolver árvores quase-ótimas. Uma das técnicas é escolher como raiz da árvore o nó que melhor divide a base de treinamento

e fazer isso sucessivamente nos nós subsequentes, até que toda a base tenha sido classificada. Alguns indicadores foram desenvolvidos para medir a qualidade da divisão, como o ganho de informação e o índice Gini. O uso desses indicadores permite também a poda da árvore, ou seja, uma generalização do modelo sem que a capacidade de classificação seja perdida. A Tabela 1 representa um exemplo simples de uma base de treinamento e Figura 3 a árvore de decisão gerada a partir desta base de treinamento.

Tabela 1. Exemplo de dados de treinamento para a construção da árvore de decisão.

Atrib. 1	Atrib. 2	Atrib. 3	Atrib. 4	Classe
A1	A2	A3	A4	Sim
A1	A2	A3	B4	Sim
A1	B2	A3	A4	Sim
A1	B2	B3	B4	Não
A1	C2	A3	A4	Não
A1	C2	A3	B4	Não
B1	B2	B3	B4	Não
C1	B2	B3	B4	Não

Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

Os algoritmos de árvore de decisão mais conhecidos da literatura são o ID3, o C4.5, o C5.0 e o CART. Este trabalho usará uma implementação do CART na fase de testes, contida na biblioteca *Sci-Kit Learn* do Python.

Figura 3. Exemplo de árvore decisão gerada a partir da base de treinamento apresentada na Tabela 1.

Fonte: Adaptado de (KOTSIANTIS; ZHARAKIS; PINTELAS, 2007).

2.2.5 *k*-Nearest Neighbors (k-NN)

O algoritmo *k*-Nearest Neighbor é baseado no princípio que instâncias com propriedades similares estarão localizadas próximas no espaço n dimensional. Se uma instância não está classificada, mas possui instâncias classificadas nas proximidades, então a classe da instância não classificada pode ser induzida pela classe dos k vizinhos mais próximos.

Este algoritmo é de fácil implementação e requer pouco tempo computacional para a fase de treinamento, porém mais tempo para a fase de classificação. O nível de generalização do algoritmo pode ser ajustado através do valor de k (número de vizinhos) e, de acordo com o problema estudado, diversas medidas de distâncias podem ser testadas, como a Manhattan, Euclideana, Camberra, Minkowsky, entre outras.

2.3 Redes Complexas

Segundo (BARABÁSI, 2016) as redes se encontram no coração dos sistemas complexos por sua natureza interdisciplinar, quantitativa, matemática e computacional. Desta forma, as redes complexas têm um papel importante no mundo atual, pois permitem o estudo de sistemas onde a relação entre os componentes era anteriormente desconhecida ou difícil de captar. Diferentemente de sistemas determinísticos, regidos por funções onde as correlações podem ser obtidas por métodos de análise linear, os sistemas complexos são representados através de redes onde as relações se dão através de vértices e arestas.

Uma rede ou grafo G possui três parâmetros básicos, dos quais derivam a maioria das métricas que caracterizam a rede (BARABÁSI, 2016):

- Número de vértices, denotado por n . Esses vértices são normalmente rotulados e determinam o tamanho da rede.
- Número de arestas, denotado por l . As arestas representam as interações entre os vértices. Elas raramente são rotuladas, mas podem conter direção e peso.
- Grau de um vértice, denotado por k . O grau mede a quantidade de arestas que chegam ou partem de um determinado vértice.

A descrição de uma rede pode ser feita de algumas formas, como uma representação visual ou uma lista de vértices e arestas. No entanto, a forma mais usual é através de uma matriz

de adjacência M . Uma matriz com n vértices possui n linhas e n colunas, sendo que para o caso de uma rede direcionada e sem peso:

$$M_{ij} = 1 \text{ se existe uma aresta apontando de } i \text{ para } j \quad (9)$$

$$M_{ij} = 0 \text{ se os vértices } i \text{ e } j \text{ não estão conectados} \quad (10)$$

Como base nos três parâmetros citados e na matriz de adjacência da rede, diversas medidas de uma rede aleatória podem ser inferidas, como o grau médio da rede, a probabilidade de uma rede com n vértices ter exatamente l arestas e o caminho médio entre todos os pares de vértices. (BARABÁSI, 2016) detalha todas estas medidas de uma rede aleatória. As redes reais, no entanto, possuem algumas particularidades.

2.3.1 Redes Reais e suas Propriedades

As redes reais, conforme descreve (BARABÁSI, 2016), geralmente possuem uma distribuição de arestas que não segue uma distribuição binomial pois em redes reais existe uma preferência de conexão para novas arestas, mesmo que ela seja desconhecida a priori. Se considerarmos o caso clássico de uma rede social, vemos que regiões mais densas da rede tendem a se tornar cada vez mais densas, ou seja, as novas arestas possuem uma probabilidade maior de ocorrer em regiões específicas.

Essa característica leva ao surgimento de *hubs* (vértices com grau $k \gg 1$), que por sua vez faz com que as redes reais apresentem a característica de livre de escala, ou seja, a distribuição de grau dos vértices da rede segue uma distribuição da lei de potência. Neste tipo de distribuição muitos vértices apresentarão poucas arestas e alguns vértices apresentarão um grande número de arestas (os *hubs*).

Outra característica importante das redes reais que este trabalho irá explorar é o fato de que muitas redes reais, diferentemente de redes aleatórias, possuem comunidades estruturadas e essas comunidades tendem a conter elementos com alguma semelhança entre si (Figura 4). Uma comunidade pode ser definida como uma região densa da rede, onde o número de conexões entre os vértices da mesma comunidade é maior que o número de conexões com vértices de comunidades diferentes. Com base nessa premissa, alguns métodos são capazes de detectar comunidades em redes com grande complexidade, como será detalhado mais adiante.

Figura 4. Exemplo de redes com comunidades destacadas. Na rede à esquerda nota-se a presença de três comunidades, definidas por áreas de maior densidade de conexão. Na rede à direita observam-se duas comunidades. Fonte: (BARABÁSI, 2016).

Tendo isto em vista, a análise de séries temporais através de redes complexas pode ser uma nova alternativa, abrindo um novo horizonte no entendimento de comportamentos que se repetem ao longo de décadas, já que as redes complexas estão no centro do estudo de problemas que envolvem grande quantidade de dados e interconexões (BARABÁSI, 2016). As métricas próprias do estudo de redes não só podem facilitar o entendimento destes tipos de problema como podem revelar características apenas notáveis a partir da análise da topologia dos dados.

2.3.2 Redes Artificiais e Modelos de Formação

Alguns problemas se apresentam já na forma de redes, como por exemplo, a análise de redes sociais, problemas de caminhos logísticos, entre outros. No entanto uma nova aplicação para o estudo das redes é a conversão de problemas lineares multidimensionais em redes complexas e o subsequente estudo sob esta ótica.

Na última década diversos métodos para conversão de séries temporais em redes foram propostos. (DONNER *et al.*, 2011) analisa alguns desses métodos e os diferencia em três diferentes classes:

- Redes de proximidade: leva em consideração a proximidade mútua entre diferentes segmentos de uma mesma série temporal;
- Gráficos de visibilidade: considera a convexidade de observações consecutivas;

- Redes de transição: considera a probabilidade de transição entre estados discretos do processo.

Estas estruturas são normalmente representadas por matrizes binárias que podem ser utilizadas para o estudo das características topológicas. Este trabalho se baseia no modelo de redes de transição, assumindo que os estados podem ser representados por dias similares da série temporal e que as probabilidades de transição podem ser capturadas pelo estudo das comunidades da rede.

Ainda segundo (DONNER *et al.*, 2011), em uma rede de transição os dados são compartimentados em conjunto de classes $\{S_1, \dots, S_k\}$, o que permite a análise da probabilidade de transição entre estes estados a partir de uma rede direcionada e com peso. Esta abordagem é equivalente a fazer uma discretização dos dados, o que acaba os agrupando de forma estática. Este método faz uso explícito da ordem temporal das observações, o que faz com que as conexões da rede representem a casualidade das relações do sistema estudado. Este método também limita o número de vértices da rede, ao limitar o número de estados possíveis durante a discretização dos dados e criação dos estados. Isto implica na presença de estados recorrentes na rede.

Este tipo de rede é especialmente útil em sistemas onde existe uma relação causal entre a sequência de dados pois o processo pode ser estudado com base em medidas como a centralidade dos vértices e medidas relacionadas. No entanto, esta rede tende a perder informação durante o processo de discretização, já que dados similares são agrupados em uma mesma classe. Por outro lado, em sistemas onde o ruído atrapalha o entendimento da informação principal ela pode ser de muita utilidade, já que a informação relevante se mantém no agrupamento dos dados. Por fim, outro ponto a se destacar é que os parâmetros escolhidos durante o processo de discretização interferem diretamente na topologia da rede resultante, ou seja, o número de classes definidos neste processo tornam a rede mais ou menos detalhadas em relação à informação contida na série original.

Para exemplificar o conceito das redes de transição, considere uma série temporal artificial univariada $y(t) = [y(0), y(1), y(2), \dots, y(T)]$, $T \in \mathbb{N} \mid T \leq 40$ (Figura 5.a). Dado um intervalo de discretização de $y(t)$ de 0,2, é possível classificar os dados em cinco classes diferentes, isto é, $S = \{0.2, 0.4, 0.6, 0.8, 1.0\}$. Cada classe está marcada em uma cor na figura e a rede construída a partir da série pode ser visualizada na Figura 5.b. A direção da aresta é sinalizada na rede por uma linha mais espessa.

Figura 5. Série temporal exemplo e sua rede de transição de estados discretos

Note que o conceito exposto acima pode ser expandido para um conjunto de séries, sendo necessário apenas um ajuste no processo de discretização dos dados. Considere agora duas séries $y(t) = [y_1, y_2, y_3, \dots, y_T], y_t \in [0,1]$ e $T = 40$ e $z(t) = [z_1, z_2, z_3, \dots, z_T], z_t \in [0,1]$ e $T = 40$. Neste caso, cada classe pode ser considerada como o conjunto das combinações possíveis entre as séries analisadas, ou seja, $S = \{(0.2, 0.4), (0.4, 0.6), (0.6, 0.8), (0.8, 0.6), (1.0, 0.8), (0.4, 0.2), (0.8, 1.0)\}$. Cada classe é representada por um ponto bidimensional ao absorver informações de duas séries que acontecem no mesmo intervalo de tempo T . A Figura 6 representa o conceito exposto. Como será visto mais adiante, o modelo proposto utiliza estados de seis dimensões.

Figura 6. Conjunto de séries temporais e sua rede de transição derivada

2.3.3 Estrutura da Rede – Comunidades

Assim como os métodos de conversão dos dados para redes complexas, diversas propostas para a detecção de comunidades surgiram nos últimos anos. Estes métodos podem ser divididos entre aglomerativos (*bottom-up*) ou divisivos (*top-down*). O conceito fundamental por trás de todos os métodos é de que o número de ligações entre os vértices participantes de uma comunidade deve ser maior do que o número de ligações entre as comunidades, ou seja, pode-se dizer que os vértices pertencentes à mesma comunidade estão mais densamente relacionados do que vértices pertencentes a comunidades diferentes.

Nos métodos divisivos, o processo inicia considerando que toda a rede forma apenas uma comunidade. São então calculados os caminhos mínimos entre todos os vértices e feitos cortes em ligações específicas, normalmente os caminhos mais usados (NEWMAN; GIRVAN, 2004)

Nos métodos aglomerativos, cada vértice da rede é uma comunidade no início do processo e eles são agrupados entre si de maneira que os vértices possuam mais conexões com os vértices da comunidade que pertence do que com outras comunidades. Os métodos tradicionais, como a maximização da modularidade (CLAUSET; NEWMAN; MOORE, 2004), atribuem uma comunidade a cada vértice, enquanto métodos mais recentes, como o de competição de partículas, possuem uma classificação *fuzzy* das comunidades, o que permite que cada vértice tenha um grau de comprometimento específico com cada comunidade. Podemos citar, por exemplo, o método baseado em cliques (PALLA *et al.*, 2005) e a competição de partículas (QUILES *et al.*, 2008).

Este projeto irá focar no método da maximização da modularidade Q proposto em (CLAUSET; NEWMAN; MOORE, 2004).

Em (NEWMAN; GIRVAN, 2004) os autores introduzem o índice de modularidade Q , que mede a qualidade de divisão de uma rede. Este índice tem como valor máximo 1, no caso de todos os vértices da rede fazerem parte de uma mesma comunidade. Utilizar este índice em um método aglomerativo é inviável, já que o melhor valor se dará após todos os vértices serem aglomerados em uma única comunidade.

Com isso, é proposto em (CLAUSET; NEWMAN; MOORE, 2004) um método de aglomeração hierárquica dos vértices baseado no índice Q da rede estudada, mas que o compara com o valor esperado em uma rede aleatória com o mesmo tamanho da rede estudada. A rede aleatória não possui comunidades estruturadas (BARABÁSI, 2016) e, portanto, a diferença da modularidade ΔQ entre a rede estudada e a rede aleatória teórica atinge um valor máximo antes

que todos os vértices sejam aglomerados. Tipicamente, valores acima de 0,3 mostram que a rede estudada possui uma estrutura significativa de comunidades. O método proposto utiliza o algoritmo detalhado por (FERREIRA; PINTO; LIANG, 2012) e obtido em (CLAUSET; NEWMAN; MOORE, 2004). No entanto algumas alterações foram feitas para a consideração de uma rede direcionada e com peso. Estas alterações estão explicitadas no modelo proposto.

Já o método de competição de partículas, proposto em (QUILES *et al.*, 2008), considera que várias partículas caminham na rede e competem pela dominância de cada vértice. A medida que caminham, as partículas marcam seus territórios e rejeitam partículas invasoras. Esta abordagem possui uma heurística interessante, pois simula diversos processos naturais, como a competição por recursos por animais ou ganhos de território em campanhas eleitorais. Uma característica desse processo é que é necessário que se defina o número de comunidades esperada a priori, já que o processo atinge equilíbrio quando cada comunidade tem a dominância de uma partícula. Outra característica do modelo é que as partículas caminham de forma aleatória, porém é proposto uma forma de controlar esta aleatoriedade, definindo um nível exploratório e de defesa de território durante o processo de caminhada.

2.4 Considerações Gerais

Em função das características de cada método, este trabalho irá utilizar apenas o método da maximização da modularidade proposto em (CLAUSET; NEWMAN; MOORE, 2004). Espera-se que este método mostre se a rede gerada na etapa de conversão da série temporal para rede complexa possui uma estrutura de comunidades útil para a classificação que se pretende, já que este método faz uma comparação direta com uma rede aleatória. Após aplicar este método já se terá uma ideia se a rede apresenta uma modularidade aceitável e o número de comunidades existentes. O método proposto em (QUILES *et al.*, 2008) foi testado na base de dados utilizada, porém não foi utilizado nas avaliações experimentais por apresentar um tempo de convergência maior que o método de (CLAUSET; NEWMAN; MOORE, 2004) nas fases iniciais de teste. Possivelmente isso ocorreu pois foram utilizadas apenas duas comunidades para o modelo de (QUILES *et al.*, 2008), uma comunidade de alta e uma de baixa, e neste cenário existem regiões com alta sobreposição de comunidades na rede, onde a competição por dominância é muito alta. Futuramente pode-se utilizar este método com o número de

comunidades gerado pelo método de (CLAUSET; NEWMAN; MOORE, 2004) para analisar a sobreposição das comunidades.

Capítulo 3

CLASSIFICAÇÃO E PREVISÃO DE SÉRIES TEMPORAIS ATRAVÉS DE REDES COMPLEXAS

Neste capítulo será proposto um modelo de classificação e previsão de séries temporais baseado nos conceitos descritos no Capítulo 2.

3.1 Considerações Iniciais

Dado o problema apresentado, este trabalho irá buscar classificar os dados da série linear univariada do índice Bovespa em relação à sua tendência com base na rede derivada desta série. A rede será construída com base em um conjunto de séries características extraídas da série original. Conforme detalhado na fundamentação teórica, uma rede pode ser construída a partir de um conjunto de séries temporais se cada vértice desta rede for representado por um vetor multidimensional. Neste caso, cada elemento do vetor representa uma característica da série original no momento t .

A escolha de quatro das seis características foi feita com base nos conceitos do modelo ARIMA, enquanto duas das características foram escolhidos de forma heurística. As séries contendo os ruídos de curto e longo prazo fazem referência ao processo $\text{ARMA}(p, q)$ descrito, onde a função do processo neste caso é aproximada pela média móvel de curto e longo prazo (ou seja, a função depende dos valores históricos do processo) e o ruído é medido em relação à esta curva. Os gradientes de curto e longo prazo fazem referência ao processo de diferenciação dos dados, já que as médias móveis são diferenciadas para tornar o processo estacionário. As

últimas duas características, as posições relativas ao máximo e mínimo histórico de curto e longo prazo, foram escolhidas pois os máximos e mínimos do processo estudado tendem a ser barreiras para o índice Bovespa no mundo real.

Apesar deste trabalho ter escolhido seis características da curva univariada original, outras poderiam ser utilizadas, como por exemplo a média móvel do desvio padrão dos dados, o que poderia criar regiões na rede de alta e baixa volatilidade dos mercados. Além disto, séries de outros processos poderiam ser adicionadas, como por exemplo índices que têm clara correlação com o processo estudado, como a cotação de moedas e índice de juros. O vetor, que neste caso tem seis elementos, pode conter um número maior de elementos, porém vale ressaltar que uma nova característica pode melhorar o processo de classificação da tendência, assim como piorar, caso a característica reduza a modularidade da rede, isto é, misture mais os vértices.

O processo de classificação proposto deverá se dividir nas etapas descritas no fluxograma abaixo (Figura 7) e descritas em detalhes nos tópicos seguintes.

Figura 7. Fluxograma das etapas do modelo proposto de classificação da série temporal

Posteriormente será proposto um modelo para a previsão da tendência e os erros de classificação e previsão do método serão avaliados. Como mencionado na motivação deste trabalho, espera-se que a classificação proposta rotule os dados de uma forma que torne a previsão por algoritmos de aprendizado de máquina mais precisa. Desta forma, os dados classificados pelo modelo proposto serão inseridos nos cinco classificadores escolhidos e a acurácia de previsão será comparada com métodos de classificação manuais (descritos no Capítulo 3.9). O fluxograma abaixo ilustra o processo de previsão proposto (Figura 8).

Figura 9. Série temporal composta do preço de fechamento do índice Bovespa de 02/01/1996 a 08/05/2018.

Além da base de dados real, será utilizada nos testes experimentais uma base de dados artificial, que será gerada a partir de uma função sinusoidal no formato da Equação (13). Esta base será utilizada para simular a base de dados real em um ambiente onde a componente determinística do processo é conhecida, possibilitando a medida de acurácia e de absorção de ruído do modelo proposto. A série artificial foi criada de forma que tivesse alguma semelhança com a série real, desta forma, os ciclos de alta são compostos de três segmentos de alta, com suas respectivas correções e os ciclos de baixa de três segmentos de baixa e suas respectivas correções (Figura 10). Este comportamento simula a teoria proposta por Charles Dow (criador do índice americano Dow Jones) de que o mercado passa por três fases, tanto nos movimentos de alta como nos movimentos de baixa. Essas três fases representam a fase acumulativa, onde a tendência começa, a fase da participação pública, onde a maior parte dos investidores começa a visualizar a tendência e a fase do excesso, onde ativo já está precificado além do que deveria e a tendência está em seu fim.

$$y(t) = \beta(t) \pm \alpha(t), \quad t \in T \quad (13)$$

$$\text{Onde, } \beta(t) = 400 + 30 * \sin\left(\frac{t}{300}\right) + 10 * \sin\left(\frac{t}{40}\right) \quad (14)$$

$$\text{e } \alpha(t) = \Phi_{\beta(t), \sigma^2}, \quad \sigma^2 \in [0, 0.05] \quad (15)$$

Figura 10. Série temporal artificial utilizada no trabalho. A figura ilustrada aqui não possui ruído.

3.3 Extração das Características da Série Original

O modelo assume três parâmetros que precisam ser definidos para a extração das características da série original:

- Um período de curto prazo ($srt \in \mathbb{N}$). Neste trabalho foi utilizado como período de curto prazo o valor de 25 dias (aproximadamente um mês útil);
- Um período de longo prazo ($lng \in \mathbb{N}$). Neste trabalho foi utilizado como período de longo prazo o valor de 125 dias (aproximadamente seis meses úteis);
- Um tamanho de bin para a tornar os dados discretos ($N_b \in \mathbb{N}$). Neste trabalho o tamanho de bin utilizado foi de 0,25, ou seja, após a normalização dos dados eles serão aproximados para o múltiplo de 0,25 mais próximo do valor obtido.

Deve-se destacar que estes parâmetros são utilizados para criar as séries características do processo e não determinam a duração das tendências. Estes parâmetros foram escolhidos através de testes com a base de dados real devido ao bom resultado apresentado nas fases finais de acurácia de classificação (apesar de não serem ajustes ótimos).

Uma vez definidos estes três parâmetros, são extraídas da série estudada seis séries características $S(t)$, conforme descritas abaixo nas Equações (16) a (21). Estas séries são

derivadas da série estudada (real ou artificial) e irão compor os estados dos dias observados mais adiante.

- Ruído de curto prazo ($S1(t)$): para cada ponto de $cp(t)$ o ruído de curto prazo é calculado como o desvio percentual do preço no dia t em relação à média móvel de curto prazo da série $cp(t)$. Um ruído de curto prazo igual a zero significa que o valor de $cp(t)$ é igual à média dos últimos 25 valores da série.

$$S1(t) = \frac{cp(t) - \frac{\sum_{n=t-srt}^{n=t} cp(n)}{srt}}{\frac{\sum_{n=t-srt}^{n=t} cp(n)}{srt}}, \quad t \in [srt, T] \quad (16)$$

- Ruído de longo prazo ($S2(t)$): para cada ponto de $cp(t)$ o ruído de longo prazo é calculado como o desvio percentual do preço no dia t em relação à média móvel de longo prazo da série $cp(t)$. Um ruído de longo prazo igual a zero significa que o valor de $cp(t)$ é igual à média dos últimos 125 valores da série.

$$S2(t) = \frac{cp(t) - \frac{\sum_{n=t-lng}^{n=t} cp(n)}{lng}}{\frac{\sum_{n=t-lng}^{n=t} cp(n)}{lng}}, \quad t \in [lng, T] \quad (17)$$

- Gradiente de curto prazo ($S3(t)$): para cada ponto de $cp(t)$ o gradiente de curto prazo é calculado como a variação percentual da média móvel de curto prazo.

$$S3(t) = \frac{\frac{\sum_{n=t-srt}^{n=t} cp(n)}{srt} - \frac{\sum_{n=t-1-srt}^{n=t-1} cp(n)}{srt}}{\frac{\sum_{n=t-srt}^{n=t} cp(n)}{srt}}, \quad t \in [srt, T] \quad (18)$$

- Gradiente de longo prazo ($S4(t)$): para cada ponto de $cp(t)$ o gradiente de longo prazo é calculado como a variação percentual da média móvel de longo prazo.

$$S4(t) = \frac{\frac{\sum_{n=t-lng}^{n=t} cp(n)}{lng} - \frac{\sum_{n=t-1-lng}^{n=t-1} cp(n)}{lng}}{\frac{\sum_{n=t-lng}^{n=t} cp(n)}{lng}}, \quad t \in [lng, T] \quad (19)$$

- Posição relativa máximo-mínimo de curto prazo ($S5(t)$): para cada ponto de $cp(t)$ a posição relativa de curto prazo é calculada como a posição do valor de $cp(t)$ em relação ao máximo e mínimo histórico de curto prazo.

$$S5(t) = \frac{cp(t) - \min(cp(t-srt):cp(t))}{\max(cp(t-srt):cp(t)) - \min(cp(t-srt):cp(t))}, \quad t \in [srt, T] \quad (20)$$

- Posição relativa máximo-mínimo de longo prazo ($S6(t)$): para cada ponto de $cp(t)$ a posição relativa de longo prazo é calculada como a posição do valor de $cp(t)$ em relação ao máximo e mínimo histórico de longo prazo.

$$S6(t) = \frac{cp(t) - \min(cp(t-lng):cp(t))}{\max(cp(t-lng):cp(t)) - \min(cp(t-lng):cp(t))}, \quad t \in [lng, T] \quad (21)$$

As séries (16) a (21) são então normalizadas e discretizadas de forma que cada valor assuma o valor entre -1 e 1 mais próximo do bin definido. A discretização dos dados é feita para reduzir a quantidade de valores possíveis em cada posição do vetor característico e assim agrupar os dias que possuem características semelhantes. Na etapa de normalização dos dados foi utilizada a medida de z -score (Equação (22)). Desta forma períodos com diferentes níveis de ruído ao longo do tempo são tratados da mesma forma.

$$Z = \frac{x - \mu}{\sigma} \quad (22)$$

Como o tamanho do bin definido foi de 0,25, os valores das séries características assumem valores contidos em $[0,00, 0,25, 0,75, 1,00]$. Cada ponto da série original é então convertido em um vetor com seis elementos, onde cada elemento é o valor de uma das séries características no tempo t . Ao fazer isso, a série temporal original é convertida em uma sequência de vetores, onde cada vetor representa um estado do processo, já que ele contém as seis características no tempo t . Desta forma, $cp(t)$ é convertida $v(t)$, onde $v_t = [S1_t, S2_t, S3_t, S4_t, S5_t, S6_t]$, conforme Equação (23) e (24).

$$v(t) = [v_1, v_2, v_3, \dots, v_t], \quad t \in T \quad (23)$$

$$v_t = [S1_t, S2_t, S3_t, S4_t, S5_t, S6_t], \quad t \in T \quad (24)$$

A série $v(t)$, diferentemente de $cp(t)$, possui uma grande quantidade de dados iguais, ou seja, dias t onde o vetor v_t se repete. Isto ocorre devido ao processo de discretização dos dados, onde dias com características similares são agrupados no mesmo bin. Por exemplo, se $v_{45} = v_{543} = v_{1289} = v_{4321} = [0.0, 0.4, 0.4, 0.2, 0.8, 0.6]$, esses quatro dias (representados por vetores) possuem valores iguais de ruído de curto e longo prazo, gradiente de curto e longo prazo e posição relativa máximo-mínimo de curto e longo prazo. Assim este processo agrupa os dias em estados similares do processo. A Figura 11 mostra um segmento de 50 observações da série original (a) e suas respectivas características de curto prazo normalizadas (b) e, por fim, discretizadas (c).

Figura 11. (a) Série temporal original correspondente a 50 dias do preço de fechamento do índice Bovespa; (b) Características de curto prazo normalizadas (ruído em vermelho, gradiente em azul e posição relativa máximo-mínimo em verde); (c) Características de curto prazo normalizadas e discretizadas (mesmo padrão de cores).

Note que dois objetivos importantes são atingidos com este pré-processamento dos dados: (i) toda a base de dados original, que é uma série divergente, passa a ser representada por séries características que oscilam entre -1 e 1 e (ii) dias similares são pré-agrupados nesta fase. Observando a Figura 11.c pode-se observar que os dias 5019, 5020 e 5021 são similares, ao menos em termos das características de curto prazo.

3.4 Conversão da Série Temporal em Rede

Para a conversão da série $v(t)$ em uma rede, cada ponto v_t é considerado como um vértice da rede. A fim de facilitar o uso dos dados, os vértices similares são rotulados de acordo com ordem em que aparecem na série $v(t)$, ou seja, se $v_5 = v_{70} = v_{213} = v_{1040}$, então todos estes vértices são rotulados como $node_5$.

Por fim os vértices são conectados, também respeitando a ordem em que aparecem na série $v(t)$, ou seja, se $node_i$ precede $node_j$ na série $v(t)$, então uma aresta direcionada é criada de $node_i$ para $node_j$. A criação da rede $G(n, l)$, através de sua representação em uma matriz de adjacência M é exibida no Algoritmo 1.

Algoritmo 1 – Geração da rede

```

1: procedimento Geração da rede
2:   for  $i \leftarrow lng$  to  $t$  do:
3:      $node_i \leftarrow [S1_i, S2_i, S3, S4_i, S5_i, S6_i]$ 
4:   end for
5:    $new\_node \leftarrow 1$ 
6:   for  $i \leftarrow lng$  to  $t$  do:
7:     if  $label(node_i) = \emptyset$  then:
8:        $label(node_i) \leftarrow new\_node$ 
9:       for  $j \leftarrow i + 1$  to  $t$  do:
10:        if  $node_j = node_i$  then:
11:           $label(node_j) \leftarrow label(node_i)$ 
12:        end if
13:      end for
14:    end if
15:     $new\_node \leftarrow new\_node + 1$ 
16:  end for
17:   $M_{xy} \leftarrow \emptyset$ 
18:  for  $i \leftarrow lng + 1$  to  $t$  do:
19:     $x = label(node_{i-1}), y = label(node_i)$ 
20:     $M_{xy} \leftarrow M_{xy} + 1$ 
21:  end for
22:  return  $M$ 
23: fim do procedimento

```

3.5 Detecção de Comunidades

Nesta etapa é aplicado o método de (CLAUSET; NEWMAN; MOORE, 2004) mencionado na fundamentação teórica. Este método divide a rede criada em comunidades com base na modularidade máxima. Como as arestas foram criadas com base na ordem temporal dos

vértices em relação à série $v(t)$, espera-se que as tendências sejam representadas por diferentes comunidades da rede e que a estrutura final seja parecida com a representação ilustrativa da Figura 12, onde uma das classes é representada por vértices em verde e a outra classe é representada pelos vértices em vermelho. Os vértices de transição são balanceados em relação às conexões entre as duas classes e são representados no detalhe da Figura 12. Essa estrutura é esperada pois cada vértice da rede contém as seis características de cada dia da série temporal e as características escolhidas tendem a apresentar valores próximos e recorrência dentro de uma mesma tendência. Com isso espera-se que as diferentes tendências se apresentem na forma de regiões mais conectadas da rede.

Figura 12. Representação ilustrativa da rede clusterizada em duas classes. Fonte: (BARABÁSI, 2016)

Conforme mencionado anteriormente o método de detecção descrito em (FERREIRA; PINTO; LIANG, 2012) foi modificado para considerar uma rede direcionada e com peso. Desta forma o algoritmo é apresentado abaixo com as alterações propostas. No método utilizado, as estruturas ΔQ_{ij} e a_i apresentam a probabilidade da presença de uma aresta em uma determinada região da rede. Para o caso de uma rede direcionada e com peso, estas probabilidades foram calculadas conforme apresentadas abaixo. O algoritmo mantém três estruturas de dados:

- Uma matriz contendo ΔQ_{ij} para cada par i e j de comunidades que contenham pelo menos uma aresta entre elas;
- Uma pilha H de dados contendo os elementos máximos de cada linha da matriz ΔQ_{ij} ;
- Uma lista com os elementos a_i contendo a fração de arestas que terminam em vértices da comunidade i .

As estruturas ΔQ_{ij} e a_i são inicializadas de acordo com as Equações (25) e (26), onde k é o grau de saída (*out*) do vértice n e l é o número total de arestas da rede.

$$\Delta Q_{ij} = \begin{cases} \frac{k_{out}}{l} - \frac{k_i k_j}{l^2} & \text{se } i \text{ e } j \text{ estão conectados} \\ 0 & \text{caso contrário} \end{cases} \quad (25)$$

$$a_i = \frac{k_i}{l} \quad (26)$$

Depois de inicializar estas estruturas de dados o Algoritmo 2 de detecção de comunidades é executado de acordo a descrição abaixo (esta parte não difere da referência citada).

Algoritmo 2 – Detecção de comunidade

```

1: procedimento Detecção de comunidade
2:   Calcular os valores iniciais de  $\Delta Q_{ij}$  e  $a_i$  de acordo com 25 e 26 respectivamente;
3:   Colocar o maior elemento de cada linha da matriz  $\Delta Q_{ij}$  na pilha  $H$ ;
4:    $Q \leftarrow 0$ 
5:   repetir
6:     Selecionar o maior  $\Delta Q_{ij}$  da pilha  $H$ ;
7:     Juntar as comunidades  $i$  e  $j$ ;
8:     Atualizar a matriz  $\Delta Q_{ij}$ , a pilha  $H$  e a lista  $a_i$ ;
9:      $Q \leftarrow Q + \Delta Q_{ij}$ ;
10:  até que reste apenas uma comunidade
11: fim do procedimento

```

Quando as comunidades i e j são aglomeradas, nomeia-se a nova comunidade como j , atualiza-se todo k elemento da linha j e coluna j e remove-se a linha i e a coluna i . Na atualização de ΔQ_{ij} são considerados três situações:

- Se a comunidade k está conectada com i e j :

$$\Delta Q'_{jk} = \Delta Q_{ik} + \Delta Q_{jk} \quad (27)$$

- Se a comunidade k está conectada à i mas não à j :

$$\Delta Q'_{jk} = \Delta Q_{ik} - a_j a_k \quad (28)$$

- Se a comunidade k está conectada à j mas não à i :

$$\Delta Q'_{jk} = \Delta Q_{jk} - a_i a_k \quad (29)$$

Por fim atualiza-se a lista $a_j' = a_j + a_i$ e $a_i = 0$. Para cada iteração do algoritmo, Q é incrementado com o maior valor de ΔQ_{ij} , ou seja, $Q = Q + \Delta Q_{ij}$. A melhor divisão da rede em comunidades ocorre quando Q para de aumentar.

No entanto, o algoritmo de detecção de comunidade se trata de um processo de agrupamento não supervisionado e que, portanto, não dá nenhum significado aos dados agrupados. Em vista disto, será proposto um algoritmo para a classificação das comunidades na etapa a seguir.

3.6 Classificação da Tendência da Série Temporal

Para classificar as comunidades em relação à tendência dos dados da série original $cp(t)$, é feita uma caminhada na rede $G(n, l)$ com uma partícula, de forma que os vértices sejam percorridos na ordem em que aparecem em $v(t)$ (note-se que cada vértice corresponde a um vetor v_t como detalhado na etapa 2). A caminhada inicia, portanto, no vértice correspondente a v_1 e termina em v_T , onde T é o último período da série. Toda vez que a partícula atravessa uma comunidade, ou seja, entra e sai da comunidade, a variação percentual do valor de $cp(t)$ é atribuído à comunidade. Ao fim da caminhada, cada comunidade terá uma média de ganho ou perda percentual em relação à série original $cp(t)$. O Algoritmo 3 descreve este processo.

Algoritmo 3 – Classificação da comunidade

```

1: procedimento Classificação da comunidade
2:   for community in G:
3:      $pct\_change(community) \leftarrow \emptyset$ 
4:      $trend(community) \leftarrow 0$ 
5:     for  $i \leftarrow lng$  to  $t$  do:
```

```

5:     if community(nodei) = community do:
7:         pct_change(community) ← change to CPi
8:     end if
9: end for
10: trend(community) ← mean(pct_change(community))
11: if trend(community) > 0 then:
12:     trend(community) ← label 'UP'
13: else if trend(community) < 0 then:
14:     trend(community) ← label 'DW'
15: end if
16: end for
17: fim do procedimento

```

Para ilustrar este processo, a Figura 13 mostra os últimos 2000 dados da série original com uma comunidade destacada em vermelho (comunidade rotulada como 2741). Isto é possível pois cada valor da série $cp(t)$ possui um vértice correspondente na rede e é possível fazer o caminho inverso do processo de conversão da série em rede, ou seja, atribuir uma comunidade a cada ponto da série $cp(t)$.

Figura 13. Exemplo de classificação de comunidade identificada no processo de detecção de comunidades. Neste caso a comunidade 2741 está destacada e algumas ocorrências podem ser vistas no período plotado. A comunidade 2741 se trata de uma comunidade de baixa.

A partícula passa diversas vezes por essa comunidade ao longo das mais de 5000 observações que compõem a série $cp(t)$. Se considerarmos o período observado na Figura 13, vemos nove trechos, cada um com uma duração diferente e um Δcp distinto, mas em geral, as passagens desta comunidade registram perda de valor em relação à $cp(t)$. Como veremos nos resultados, a comunidade 2741 é uma comunidade rotulada como baixa (“DW”). Este processo

detalhado para a comunidade 2741 é feito para todas as comunidades (como detalhado no Algoritmo 3) e assim todas as comunidades recebem um rótulo “UP” ou “DW”.

3.7 Previsão de Tendência de uma Nova Observação

O último passo do método proposto utiliza os cinco métodos de aprendizado de máquina descritos da revisão bibliográfica para prever a tendência de um novo ponto da série de dados estudada.

Para isso cada modelo recebe como base de treinamento os dados da rede, ou seja, o vetor com as seis características e o rótulo de cada vértice da rede (Figura 14). Os pesos das arestas são passados para os classificadores através da repetição dos vértices na base de dados.

Figura 14. Exemplo de dados inseridos nos classificadores utilizados nos experimentos.

Todos os classificadores são treinados utilizando a estratégia *n-fold*, no caso, são utilizadas 10 partições, sendo 9 para treinamento e 1 para teste. A acurácia de cada classificador em relação à previsão de um novo dado é feita com base na média das 10 partições. Com isso cada classificador é ajustado e sua acurácia de previsão é conhecida. Cada classificador possui suas particularidades em relação ao ajuste, como o *kernel* utilizado (caso do SVM), a função de distância utilizada (caso do *k-NN*), o ponto de poda (caso da árvore de decisão), entre diversos outros ajustes possíveis. Este trabalho utilizou os parâmetros padrões da biblioteca *Sci-Kit Learn* do Python, já que todas tiveram uma boa performance e o objetivo do trabalho nesta etapa era apenas mostrar que a classificação do modelo proposto é mais previsível que a classificação de referência proposta.

Para a previsão de um novo ponto, ou seja, para se prever a tendência futura, assume-se que a tendência do atual será mantida e que esta deverá ser prevista. Desta forma, a cada novo ponto observado calcula-se o vetor característico e o rótulo é previsto pelo classificador ajustado conforme descrito.

3.8 Análise dos Resultados

Os resultados serão apresentados em três partes. Primeiros serão apresentadas as métricas das redes geradas para as bases de dados utilizadas (série artificial e série real). Depois serão apresentados os resultados relativos à classificação feita pelo modelo. Para a curva artificial, o algoritmo será testado na curva sinusoidal descrita, onde a função determinística é conhecida, de forma que o erro de classificação poderá ser avaliado. Para tornar o teste mais próximo de uma série estocástica, será adicionado ruído à esta curva e o erro, bem como a modularidade da rede, serão avaliados à medida que o ruído aumenta. Isto permitirá medir a performance do algoritmo em uma curva semelhante à curva real, mesmo não sendo gerada por um processo estocástico.

Para os testes de classificação utilizando a curva real, o modelo proposto será comparado com o modelo de classificação descrito na Seção 3.9. Além disso serão apresentados os detalhes das classes obtidas através da topologia da rede e sua representação gráfica nas séries temporais e na rede.

Posteriormente, será avaliada a capacidade de previsão do algoritmo. Assim como na classificação, será utilizada uma curva sinusoidal como apoio para a avaliação de performance e o algoritmo será também aplicado na curva $cp(t)$. Todos os resultados serão detalhados no Capítulo 4.

3.9 Comparação com Métodos Existentes

A fim de comparar a performance de classificação da curva real $cp(t)$, diversos modelos de classificação poderiam ser utilizados como base ou, até mesmo, a classificação da base de dados por um especialista. Por se tratar de um processo estocástico, consideramos neste trabalho

que o método que mais aproxima a classificação feita por um especialista seria a interpolação linear dos dados (RANI *et. al.*, 2014).

Na interpolação linear a série temporal é dividida em segmentos de tamanho pré-determinados ou de acordo com padrões identificados manualmente. Neste trabalho usaremos partições de tamanho pré-determinados. Para cada partição de tamanho Δx (representada na Figura 15 pela linha tracejada) a interpolação será feita de forma a criar um segmento de reta entre os pontos de cruzamento da série e da partição.

Figura 15. Série temporal interpolada em intervalos simétricos. Os dados da série são classificados de acordo com o ponto inicial e final de cada segmento de reta. Nesta figura as tendências de alta estão marcadas em azul e as tendências de baixa em vermelho.

Suponha portanto que a partição inicie em x_i e termine em x_j . A interpolação linear criará um segmento de reta ligando os pontos (x_i, y_i) e (x_j, y_j) . A tendência da partição é definida pelo gradiente da reta formada e todos os pontos da partição recebem a mesma classificação. Na Figura 15 isto foi ilustrado marcando as tendências de alta em azul e as tendências de baixa em vermelho.

Na fase de testes o tamanho da partição será testado para diversos valores, já que quanto menor o tamanho da partição, mais precisa será a classificação, porém menos ruído ela absorve, o que deve ter um impacto na performance de previsão deste tipo de classificação.

3.10 Considerações Finais

Ao final, será feita a conclusão deste trabalho baseada nos resultados obtidos. A conclusão deverá compreender aspectos como a contribuição do trabalho desenvolvido para o meio acadêmico e os conceitos envolvidos, a avaliação da forma que foi executada a metodologia proposta e comentários sobre a implementação dos métodos, bem como detalhamento dos resultados obtidos em relação ao objetivo inicial proposto. Por fim será feita uma análise qualitativa do cenário em que este trabalho se insere e como ele pode ser aproveitado para a melhoria da assertividade de previsões financeiras e também como ele pode ser aplicado em outras áreas.

Capítulo 4

RESULTADOS EXPERIMENTAIS

Nesta parte do trabalho serão apresentados alguns resultados obtidos através do modelo proposto. Primeiro serão apresentadas métricas da rede criada a partir da série temporal. Em seguida serão apresentados os resultados da classificação da série temporal através do modelo e por último os algoritmos de classificação descritos serão utilizados para a previsão da série. Todos os resultados serão apresentados para a curva artificial proposta e para a curva real, composta do preço de fechamento do índice Bovespa.

4.1 Métricas da Rede Gerada

A fim de caracterizar a rede gerada a partir da série temporal, algumas medidas foram extraídas. Serão apresentados para cada base de dados (curva artificial e curva real) o número de vértices da rede, o número de arestas, o grau médio dos vértices e a distância máxima prevista (ou teórica) entre dois vértices da rede. Como veremos a seguir, cada rede apresentará características particulares, já que cada curva tem um comportamento diferente.

4.1.1 Métricas da Rede Gerada – Curva Artificial

A curva artificial possui um ruído controlável e, desta forma, x redes podem ser geradas, uma para cada nível de ruído. Para exemplificar, utilizaremos neste experimento um desvio padrão de 0,02 ($\sigma^2 = 0,02$), que corresponde ao valor do desvio padrão das variações diárias da curva real, ou seja, 95% das variações diárias da curva real estão dentro dos limites de mais ou menos dois desvios padrão (0,04), conforme Figura 16.

Figura 16. Desvio padrão da curva real - índice Bovespa

Considerando, portanto, $\sigma^2 = 0,02$ a rede gerada a partir da curva artificial gerou os indicadores apresentados na Tabela 2. Pode-se ver a grande capacidade de redução de informação ao se transformar a série temporal em uma rede (6000 pontos observados são convertidos em 1559 vértices). O número de arestas é igual ao número de observações menos um, já que o modelo de formação da rede prevê que toda observação gera uma aresta na rede.

O grau médio, por sua vez, mostra que, em média, um estado da série temporal ocorre aproximadamente 7 vezes no período estudado, mostrando a recorrência dos estados ao longo do tempo. Já a distância máxima prevista entre dois vértices (ou dois estados do processo) é menor do que 4, mostrando que existe um caminho possível entre qualquer estado do processo em menos de 4 dias. No entanto, a probabilidade de se atingir cada vértice não é a mesma, já que os pesos das arestas não são levados em conta neste indicador.

Tabela 2. Dados da rede criada a partir da curva artificial

	DADOS DA REDE
	Curva artificial (6000 pontos de dado)
Vértices (n)	1559
Arestas (l)	5999
Grau médio ($\langle k \rangle$)	7,07
Distância máxima prevista ($\frac{\ln N}{\ln \langle k \rangle}$)	3,76

Como o grau médio não explicita a estrutura da rede, o grau dos vértices foi então plotado em um histograma (Figura 17) e a rede foi plotada na Figura 18. Mais da metade dos vértices possuem grau 2, ou seja, são estados que ocorrem apenas uma vez no período analisado (uma aresta de entrada e uma aresta de saída). Esses vértices são pouco relevantes no entendimento do processo, já que podem ser considerados estados pouco comuns do processo. A rede apresenta, no entanto, diversos vértices muito conectados, que geram hubs e regiões mais densas (Figura 18). Essas características serão detalhadas nos experimentos de classificação das séries, onde as comunidades serão detectadas pelo modelo proposto.

Figura 17. Histograma do grau (k) dos vértices da rede gerada pela curva artificial

Figura 18. Rede gerada a partir da curva artificial

4.1.2 Métricas da Rede Gerada – Curva Real (Índice Bovespa)

A rede gerada a partir dos dados reais gerou os indicadores apresentados na Tabela 3. Como podemos ver, as propriedades mencionadas para a curva artificial se mantêm na curva real, mas com algumas diferenças.

Podemos ver que a redução na quantidade de dados é ainda maior, transformando 5530 pontos observados em uma rede com apenas 991 vértices. Outro ponto importante é que o grau médio é ainda menor, indicando que a quantidade de vértices com grau baixo deve ser ainda maior que na curva artificial.

Tabela 3. Dados da rede criada a partir da curva real

	DADOS DA REDE
	Curva real (5530 pontos de dado)
Vértices (n)	991
Arestas (l)	5529
Grau médio ($\langle k \rangle$)	5,20
Distância máxima prevista ($\frac{\ln N}{\ln \langle k \rangle}$)	4,18

Com isto, foi plotado o histograma de grau dos vértices da curva real (Figura 19). Pode-se observar que a quantidade de vértices com grau 2 continua sendo bastante alta e, proporcionalmente ao total de vértices, possui uma participação muito maior na rede, o que explica o grau médio mais baixo.

Do ponto de vista prático, a rede real possivelmente pode ser estudada com um número ainda menor que os 991 vértices gerados, já que os vértices mais relevantes (que possuem $k > 2$) são menos que 300 vértices. Os vértices com $k \gg 2$, por sua vez, são os estados do processo que foram testados por muitas vezes ao longo dos 22 anos da base de dados e que devem gerar informações relevantes sobre as probabilidades históricas.

Por fim, a rede gerada a partir da série temporal do índice Bovespa foi plotada (Figura 20) e podemos observar visualmente os vértices com grau 2 (a maioria plotado nas bordas da rede) e as regiões mais densas, incluindo algumas comunidades. Na seção seguinte as redes serão mostradas com suas comunidades identificadas.

Figura 19. Histograma do grau (k) dos vértices da rede gerada pela curva artificial

Figura 20. Rede gerada a partir da curva real

4.2 Classificação da Tendência com o Modelo Proposto

Após a geração das redes para a curva artificial e para a curva real, o algoritmo proposto foi aplicado para a classificação da tendência em ambos os cenários. Novamente os resultados serão apresentados separadamente.

4.2.1 Classificação da Tendência com o Modelo Proposto – Curva Artificial

O algoritmo foi aplicado primeiramente na curva sinusoidal, para avaliação do erro de classificação e comportamento da modularidade em função do ruído presente na curva. A curva utilizada possui uma componente determinística, representada pela função sinusoidal (13) e uma componente aleatória ($\varphi(t)$) que varia conforme uma distribuição Gaussiana com σ^2 variando entre 0,00 e 0,05. A Figura 21 mostra a curva classificada em diferentes níveis de ruído (respectivamente, a) $\sigma^2 = 0,00$, b) $\sigma^2 = 0,01$, c) $\sigma^2 = 0,02$ e d) $\sigma^2 = 0,03$). Como era esperado o erro de classificação aumenta com o aumento do ruído. A medida do erro será apresentada no próximo experimento.

Figura 21. Curva sinusoidal classificada pelo modelo proposto em diferentes níveis de ruído. Em (a) $\sigma^2 = 0,00$, em (b) $\sigma^2 = 0,01$, em (c) $\sigma^2 = 0,02$ e em (d) $\sigma^2 = 0,03$.

O próximo passo foi conduzir um teste para avaliar o comportamento da acurácia de classificação e da modularidade em função do ruído presente na curva. Os valores foram medidos variando o desvio padrão de 0,00 a 0,05. A medida que o ruído aumenta a acurácia de classificação cai linearmente até atingir 0,5 (Figura 22.a). A modularidade permaneceu acima de 0,3 durante todo o teste, mostrando que a rede apresenta comunidade estruturadas durante toda a faixa de ruído utilizada (Figura 22.b). Como os rótulos de classificação são apenas dois (“UP” ou “DW”), um classificador com erro de 50% não possui nenhuma assertividade. No

entanto, o resultado mostra uma boa tolerância do algoritmo em faixas de desvio padrão de até 0,02 onde o algoritmo tem uma assertividade próxima de 80% na curva artificial.

Deve-se observar que, conforme mostrado na Figura 16, o processo real estudado, composto da série temporal do índice Bovespa, possui um desvio padrão (se considerarmos as variações diárias de preço) abaixo de 0,02. Assim é de se esperar que a acurácia medida nesta etapa se repita para a curva real. Esta hipótese, no entanto, não poderá ser confirmada na prática, como veremos nos testes com a curva real.

Figura 22. (a) Acurácia de classificação em função do ruído presente na curva sinusoidal. (b) Modularidade da rede com base no ruído presente na curva sinusoidal.

4.2.2 Classificação da Tendência com o Modelo Proposto – Curva Real

Ao contrário da série artificial proposta, a série real, composta pelo preço de fechamento do índice Bovespa, não possui uma componente determinística (ao menos conhecida). Desta forma o processo pode ser classificado como estocástico e não é possível calcular a acurácia da classificação, já que não existe uma função de comparação.

No entanto, é possível medir a performance de classificação comparando a classificação feita pelo modelo com a classificação proposta na Seção 3.9.

A forma utilizada neste trabalho como medida de performance será o cálculo do número de pontos acumulados, ou seja, Δcp durante o período analisado. Este cálculo é feito dividindo a série temporal em n partições e classificando todos os pontos dentro da partição como ‘UP’ ou DW’ de acordo com o valor inicial e final da partição (conforme descrito Capítulo 3.9). Este método assume, portanto, que não existe erro de classificação, apenas uma variação no número de pontos acumulados em função do número de partições. Quanto menor o tamanho de cada partição (maior o número de partições) mais precisa é a classificação. A divisão da série em janelas diárias, por exemplo, implicaria no máximo de pontos possíveis, ou seja, seria uma classificação que acerta todos os movimentos diários.

Como podemos ver na Figura 23, o modelo proposto possui uma performance inferior de classificação em relação ao método proposto no Capítulo 3.9, especialmente quando se utiliza janelas mais curtas para as partições. Isto era esperado, já que a interpolação acerta a tendências de todos os segmentos. No entanto, como veremos na fase de previsão, esta forma de classificação é pouco previsível. Como o método não identifica nenhum padrão ou relação com dados históricos, sua capacidade de generalização para novos dados é baixa.

Figura 23. Performance de classificação do modelo proposto e do método de particionamento da base e interpolação dos intervalos.

O modelo proposto, por outro lado, consegue manter um número acumulado de pontos constante ao longo de qualquer horizonte de tempo. Isso mostra que o modelo é pouco sensível aos parâmetros de ajuste (srt e lng) na etapa de classificação.

Para verificar visualmente o resultado da classificação da série original através da rede gerada, os últimos 1000 pontos da série foram plotados na Figura 24 e cada ponto foi colorido de acordo a comunidade do vértice relacionado ao ponto da série.

Figura 24. Últimos 1000 pontos da série temporal real classificados de acordo com as comunidades identificadas na rede gerada.

Este trecho contém todas as 38 comunidades identificadas na rede. É possível ver neste gráfico a capacidade de cada comunidade em capturar um padrão diferente da série temporal. Pode-se ver, por exemplo, os segmentos verde-claros no fim de tendências de alta e os segmentos azuis turquesa no fim das correções de alta. Outra propriedade interessante, que pode ser vista neste trecho de 1000 observações, é a capacidade das comunidades em diferenciar as correções, dentro de uma tendência de longo prazo, de uma inversão de tendência.

Todos os dados utilizados para gerar a Figura 24 foram gerados a partir de dados da rede e suas comunidades. A Tabela 4 mostra estes dados em detalhes e cada indicador é descrito abaixo:

- Comunidade: mostra as comunidades presentes na rede estudada. A base de dados utilizada, que continha 5530 observações, foi reduzida para 38 comunidades. Cada comunidade está rotulada de acordo com o Algoritmo 3;
- Vértices: mostra a quantidade de vértices agrupados na comunidade;
- Dias: mostra o número de observações da série original que fazem parte da comunidade;

- Part. %: mostra a participação do número de dias da série que fazem parte da comunidade. Este indicador mostra a relevância da comunidade em termos de dias representados por ela;
- Ocorrências: mostra o número de vezes que a partícula passou pela comunidade no processo de classificação da comunidade descrita no algoritmo 3. Quanto mais passagens, mais confiável é a tendência atribuída;
- Δcp : mostra o valor calculado para o Δcp pelo Algoritmo 3;
- Tendência: mostra o rótulo da tendência de acordo com o Δcp ;
- Probabilidade de mudança: mostra a probabilidade de mudança de tendência com base nas ligações da comunidade com as demais comunidades da rede. Se a comunidade tem esse indicador igual a zero, por exemplo, quer dizer que todas as arestas partindo da comunidade são ligadas a comunidades com a mesma tendência.

Tabela 4. Dados da rede gerada com detalhes sobre as comunidades

Total de observações (comprimento da série): 5530 Modularidade da rede (Q): 0.572 Quantidade de vértices da rede: 991 Quantidade de comunidades da rede: 38							
Comunidade	Vértices	Dias	Part %	Ocorrências	% Δcp	Tendência	%Com_Change
3182	30	322	6	12	11,98	UP	33,3
3139	30	55	1	2	11,11	UP	0
2741	20	627	11,6	24	6,7	UP	37,5
808	28	67	1,2	3	6,15	UP	33,3
1148	20	48	0,9	3	5,22	UP	33,3
1675	13	26	0,5	2	4,54	UP	0
3126	31	63	1,2	2	3,96	UP	0
1984	42	336	6,2	23	3,84	UP	39,1
2891	20	358	6,6	25	3,41	UP	24
3983	52	203	3,8	14	2,94	UP	14,3
2033	30	192	3,6	14	2,35	UP	21,4
1355	32	93	1,7	5	1,44	UP	0
5375	11	289	5,3	34	1,26	UP	38,2
4959	32	168	3,1	26	1,24	UP	34,6
3830	23	51	0,9	6	0,55	UP	16,7
1586	29	78	1,4	2	0,38	UP	50

4186	34	86	1,6	12	0,24	UP	50
4715	22	47	0,9	4	0,11	UP	75
99	18	62	1,1	11	0,09	UP	36,4
4074	8	11	0,2	1	0	DW	100
5205	7	10	0,2	1	0	DW	100
396	7	10	0,2	1	0	DW	100
2548	33	132	2,4	14	-0,14	DW	64,3
81	13	55	1	8	-0,16	DW	37,5
332	25	49	0,9	4	-0,18	DW	50
5126	26	113	2,1	24	-0,43	DW	45,8
2530	19	40	0,7	6	-0,9	DW	16,7
4554	20	112	2,1	14	-1,09	DW	0
1045	45	133	2,5	13	-1,1	DW	30,8
1446	19	44	0,8	7	-1,27	DW	14,3
4929	33	301	5,6	27	-1,4	DW	37
1963	24	94	1,7	9	-1,7	DW	22,2
4530	34	261	4,8	29	-2,26	DW	10,3
1477	35	317	5,9	21	-3,02	DW	42,9
359	8	24	0,4	2	-3,24	DW	0
938	44	109	2	6	-4,48	DW	16,7
4855	40	321	5,9	16	-5,28	DW	62,5
584	34	98	1,8	3	-11,93	DW	66,7

Por fim, o mesmo trecho de 1000 observações da série original foi plotado, porém as comunidades foram identificadas apenas como comunidades de alta ('UP') ou baixa ('DW'). Visualmente pode-se observar que os dados possuem uma boa classificação, especialmente se considerarmos que o processo não foi supervisionado até esta fase. A rede classificada nas tendências de alta e baixa pode ser vista na Figura 26. Esta figura mostra a rede completa e classificada apenas pela tendência binária (não foi plotada cada comunidade, a fim de facilitar a visualização do resultado). Pode-se ver nesta rede a relação entre os vértices de alta e baixa e as zonas de maior densidade que geram as comunidades.

Figura 25. Últimos 1000 pontos da série temporal real classificados de acordo com a tendência de alta ('UP') ou baixa ('DW').

Figura 26. Rede gerada a partir da curva real e classificada nas tendências de alta (azul) e baixa (vermelho).

4.3 Previsão da Tendência com o Modelo Proposto

Os testes de previsão de tendência seguiram a mesma estrutura dos testes de classificação. Primeiro os testes foram realizados na curva artificial e os resultados analisados e posteriormente o mesmo foi feito para a curva real.

4.3.1 Previsão da Tendência com o Modelo Proposto – Curva Artificial

Para testar a acurácia de previsão na curva artificial, três cenários foram considerados. No primeiro a curva artificial foi classificada utilizando o método descrito no Capítulo 3.9 com um tamanho de partição de 25 dias e no segundo foi utilizado um tamanho de partição de 125 dias. Esses valores foram escolhidos por serem os mesmos períodos de curto e longo prazo utilizados como parâmetros do modelo proposto. Por último a curva foi classificada utilizando o modelo proposto.

Os três cenários foram testados conforme descritos no Capítulo 3.7, ou seja, utilizando cinco algoritmos de classificação diferentes (NB, k-NN, MLP, SVM e DT). A acurácia de previsão foi avaliada utilizando diferentes níveis de ruído (ou desvio padrão) para a curva artificial. Novamente, o ruído $\varphi(t)$ foi testado com σ^2 variando entre 0,00 e 0,05. As Figuras 27, 28 e 29 mostram, respectivamente, a acurácia de classificação para o cenário com a classificação pela partição de 25 dias, pela partição de 125 dias e pelo modelo proposto.

Por fim, os três gráficos foram consolidados na Figura 30, que traz a acurácia composta dos cinco modelos de previsão (média simples) em função do ruído. Desta forma é possível comparar os três cenários conjuntamente e na mesma escala. Como podemos observar, o modelo proposto possui uma acurácia maior ao longo de toda a faixa de ruído testada, sendo melhor em 18 dos 20 pontos testados e melhor de forma geral se considerarmos as curvas suavizadas. Este gráfico mostra também que o modelo de classificação proposto torna a base dados mais previsível, principalmente na presença de ruído. Como podemos ver, para a classificação feita por interpolação linear, a acurácia cai rapidamente para 50% após um pequeno acréscimo de ruído, tornando a previsão pouco útil para um problema de duas classes. Ainda assim, a performance do modelo na curva artificial não se destaca em relação ao método de referência.

Figura 27. Acurácia de previsão para a curva artificial utilizando como regra de classificação a interpolação dos dados com particionamento da base em períodos de 25 dias.

Figura 28. Acurácia de previsão para a curva artificial utilizando como regra de classificação a interpolação dos dados com particionamento da base em períodos de 125 dias.

Figura 29. Acurácia de previsão para a curva artificial utilizando como regra de classificação o modelo proposto baseado na topologia da rede.

Figura 30. Acurácia de previsão combinada para a curva artificial.

4.3.2 Previsão da Tendência com o Modelo Proposto – Curva Real

Os mesmos testes aplicados para a curva artificial foram aplicados para a curva real $cp(t)$, conforme descritos no Capítulo 3.7. No entanto, como na curva real o ruído não é controlável e sim uma característica do processo (e varia ao longo do tempo), o que foi testado neste caso foi o comportamento da acurácia de previsão em relação ao tamanho da partição utilizada no método descrito no Capítulo 3.9. Além de variar a partição do método de referência, os parâmetros do modelo (srt e lng) também foram variados para refletir os cenários com horizontes de previsão diferentes.

As Figuras 31 e 32 mostram, respectivamente, a acurácia de classificação para a base de dados real classificada de acordo com o método de referência (Capítulo 3.9) e de acordo com o modelo proposto, utilizando diferentes tamanhos de partição e parâmetros de curto e longo prazo do modelo (srt e lng). O parâmetro de curto prazo do modelo foi ajustado para o tamanho de janela do modelo de referência e a relação lng / srt foi mantida em 5, como na configuração inicial do modelo.

Por fim, a acurácia composta do método de referência e do modelo proposto foram plotadas na Figura 33 para uma avaliação na mesma escala.

Figura 31. Acurácia de classificação para o método de referência utilizando um tamanho de partição variável.

Pode-se concluir através da Figura 33 que o modelo tem uma acurácia de previsão maior do que o método de referência, especialmente para janelas de tempo menores que 150 dias. Isto significa que, apesar do modelo não classificar a tendência de todos os pontos da série temporal corretamente (o que era esperado, já que a absorção de ruído presente no modelo faz com que isso ocorra), ela gera padrões de classificação que são mais previsíveis por modelos de aprendizado de máquina.

Figura 32. Acurácia de classificação para o modelo proposto utilizando um período de curto prazo variável e uma relação longo prazo/curto prazo fixa.

Figura 33. Acurácia de classificação combinada.

Conforme descrito no Capítulo 3.3, o modelo proposto foi ajustado com os parâmetros srt e lng respectivamente iguais a 25 e 125 dias. Se olharmos os resultados para o período de 25 dias na Figura 33, vemos que o modelo possui uma acurácia de previsão de mais de 90%, enquanto o modelo de referência não passa de 70%.

Capítulo 5

CONCLUSÕES

Neste capítulo, são descritas as conclusões do presente trabalho, tanto em relação ao método proposto e suas aplicações, quanto aos resultados obtidos. Além disto, é feita uma breve anotação sobre possíveis trabalhos futuros.

O trabalho apresentado nesta dissertação de mestrado propôs uma nova abordagem para a classificação e previsão de tendência em séries temporais estocásticas. Apesar da metodologia proposta ter sido concebida para séries financeiras (mais especificamente o índice Bovespa), muitas outras aplicações podem ser imaginadas utilizando a mesma estrutura, com eventuais ajustes nos parâmetros.

Neste trabalho foram utilizadas seis séries de características derivadas da série original para construir uma rede composta de vértices multidimensionais. Seria possível, por exemplo, utilizar séries correlacionadas ao processo ou outras características do problema estudado. No caso do índice Bovespa, poder-se-ia coletar os dados históricos da taxa de juros, taxa de câmbio, dados sobre o produto interno bruto, dados sobre a inflação, entre muitos outros dados e utilizar essas informações como dados dos vértices multidimensionais da rede. Posteriormente o algoritmo proposto poderia ser usado para encontrar relações entre os dados de forma conjunta e não par a par, como em uma correlação simples.

Um ponto importante do modelo proposto é que, apesar de não manter histórico ou utilizar os valores da série temporal na rede, a classificação discreta em tendências ainda é de extrema utilidade para diversas aplicações. No caso específico do índice Bovespa, é possível utilizar a classe gerada para manter ou alterar a posição de um investimento no índice, atuando como um gatilho de compra ou venda. Esta informação, em conjunto com estratégias de

gerenciamento de risco e gestão de portfólio podem gerar estratégias bem-sucedidas de investimento.

Além disto, o método pode ser aplicado em outras áreas, com pequenas adaptações, como as características a serem extraídas e valores dos parâmetros utilizados. Pode-se por exemplo utilizar este método para o estudo de produtividade de cultivos, utilizando dados históricos climáticos, como temperatura, precipitação e humidade.

Considerando os resultados apresentados, pode-se dizer que o modelo proposto foi capaz de agrupar diferentes padrões das séries estudadas em diferentes comunidades da rede. Isso, de forma geral, resolveu diversos problemas apresentados como motivação deste trabalho. Além disto, ao agrupar os diferentes padrões em diferentes comunidades, o método proposto tornou a base de dados, que a princípio é gerada por um processo estocástico, em uma base mais previsível se comparada com o método de classificação de referência proposto.

Os resultados obtidos foram melhores na base real (índice Bovespa) do que na base artificial. Uma hipótese possível para este fenômeno é o fato de que a maioria dos períodos da série real possuem baixo nível de ruído se comparado ao valor médio utilizado na curva artificial. Outro fator importante é que o modelo proposto foi modelado para a curva real e, com isso, as características utilizadas podem ser mais relevantes para esta curva. Os máximos e mínimos históricos, por exemplo, podem atuar como resistências da curva real, mas não terem impacto na curva artificial.

Em relação aos cinco diferentes classificadores utilizados na etapa de previsão, eles apresentaram comportamento semelhante dentro de cada teste, mas houve uma variação de qual algoritmo foi melhor para cada um dos testes. Isto pode ter ocorrido pelo fato deste trabalho utilizar os parâmetros padrões da biblioteca *Sci-Kit Learn* do Python. Como não houve otimização dos classificadores não se pode afirmar que um deles foi melhor que o outro, mas como todos apresentaram uma curva semelhante, pode-se afirmar que a série classificada pelo modelo proposto é mais previsível por qualquer um dos cinco classificadores.

5.1 Publicações

O trabalho aqui apresentado gerou um artigo, intitulado “*Time Series Trend Detection and Forecasting Using Complex Network Topology Analysis*” (ANGHINONI, L. et al., 2018), apresentado na conferência *International Joint Conference on Neural Networks (IJCNN 2018)*

do congresso *IEEE World Congress on Computational Intelligence* (IEEE WCCI 2018). O artigo foi elaborado em colaboração com os coautores ZHAO, LIANG; ZHENG, Q.; ZHANG, J.

5.2 Trabalhos Futuros

Podemos concluir que o método proposto mostrou uma nova forma entender séries temporais estocásticas, não como uma sequência de observações aleatórias, mas como uma sequência de estados probabilísticos. Isto gera uma base para mais estudos neste sentido, principalmente utilizando a topologia da rede e outros estudos mais aprofundados da dinâmica deste processo, como o estudo da relação hierárquica entre as comunidades, a sobreposição entre as comunidades e muitos outros estudos que podem melhorar ainda mais o entendimento de séries estocásticas.

Neste trabalho a classificação e previsão da tendência foi testada de forma binária nos resultados, ou seja, a tendência foi classificada como alta ou baixa. Ainda assim, o modelo gerou informações mais detalhadas em relação a cada comunidade que faz parte do agrupamento de alta e do agrupamento de baixa. Estas informações podem ser utilizadas em futuros trabalhos para previsões com níveis de confiança por comunidade e não por tendência, por exemplo.

REFERÊNCIAS

ANGHINONI, L. et al. Time Series Trend Detection and Forecasting Using Complex Network Topology Analysis. In **Proceedings of IEEE World Congress on Computational Intelligence**, v. 1, pp. 5154 - 5160, 2018.

BAHETI, A.; TOSHNIWAL, D. Trend Analysis of Time Series Data Using Data Mining Techniques. In **Big Data (Big Data Congress), IEEE International Congress on**, v. 1, pp. 430 - 437, 2014.

BARABÁSI, A.-L. **Network Science**. Cambridge University Press, 2016.

CHEN, P. et al. ARIMA-Based Time Series Model of Stochastic Wind Power Generation. **IEEE Transactions on Power Systems**, v. 25, n. 2, pp. 667 - 676, 2010.

CHUMACHENKO, O. I. Forecasting the demand for UAV using different neural networks topology. In **Actual Problems of Unmanned Air Vehicles Developments (APUAVD), IEEE 2nd International Conference**, v. 1, pp. 62 - 64, 2013.

CLAUSET, A.; NEWMAN, M. E.; MOORE, C. Finding community structure in very large networks. **Physical Review E**, v. 70, n. 6, pp. 066111, 2004.

COGOLLO, M. R.; VELÁSQUEZ, J. D. Methodological Advances in Artificial Neural Networks for Time Series Forecasting. **IEEE Latin America Transactions**, v. 12, n. 4, pp. 764 - 771, 2014.

CHLÜTER, T.; CONRAD, S. About the Analysis of Time Series with Temporal Association Rule Mining. In **Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium on**, v. 1, pp. 325 - 332, 2011.

DONNER, R. V. et al. Recurrence-based time series analysis by means of complex network methods. **International Journal of Bifurcation and Chaos**, v. 21, n. 4, pp. 1019 - 1046, 2011.

EHLERS, R. S. **Análise de Séries Temporais**, 2009. Disponível em: <<http://conteudo.icmc.usp.br/pessoas/ehlers/stemp/>>. Acesso em: Setembro de 2018.

FERREIRA, L. N.; LIANG, Z. Detecting Time Series Periodicity Using Complex Networks. In **Brazilian Conference on Intelligent Systems (BRACIS)**, v. 1, pp. 402 - 407, 2014.

FERREIRA, L. N.; LIANG, Z. Time series clustering via community detection in networks. **Information Sciences**, v. 326, pp. 227 - 242, 2016.

FERREIRA, L. N.; PINTO, A. R.; LIANG, Z. QK-Means: A clustering technique based on community detection and k-means for deployment of cluster head nodes. In **Neural Networks (IJCNN), The 2012 International Joint Conference on**, v. 1, pp. 1 - 7, 2012.

- HILLMER, S. C.; TIAO, G. C. An ARIMA-Model-Based Approach to Seasonal Adjustment. **Journal of the American Statistical Association**, v. 77, n. 377, pp. 63 - 70, 1982.
- KOTSIANTIS, S. B.; ZHARAKIS, I.; PINTELAS, P. Supervised machine learning: A review of classification techniques. **Emerging artificial intelligence applications in computer engineering**, v. 160, pp. 3-24, 2007.
- LACASA, L.; NICOSIA, V.; LATORA, V. Network structure of multivariate time series. **Scientific Reports**, v. 5, pp. 15508, 2015.
- LEITE, H. D. P.; SANVINCENTE, A. Z. **Índice Bovespa: Um padrão para os investimentos brasileiros**. Atlas, 1995.
- LI, W.; LIAO, J. A comparative study on trend forecasting approach for stock price time series. In **2017 11th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID)**. v. 1, pp. 74 - 78, 2017.
- LI, X.; LIU, X.; CHI, K. T. Recent Advances in Bridging Time Series and Complex Networks. In **Circuits and Systems (ISCAS), 2013 International Symposium on**, v. 1, pp. 2505 - 2508, 2013.
- MAHALAKSHMI, G.; SRIDEVI, S.; RAJARAM, S. A Survey on Forecasting of Time Series Data. In **2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16)**, v. 1, pp. 1 - 8, 2016.
- MALKIEL, B.; FAMA, E. F. Efficient Capital Markets: A Review of Theory and Empirical Work. **The Journal of Finance**, v. 25, n. 2, pp. 383 - 417, 1970.
- NEWMAN, M. E.; GIRVAN, M. Finding and evaluating community structure in networks. **Physical review E**, v. 69, n. 2, pp. 026113, 2004.
- PAI, P. F.; LIN, C. S. A hybrid ARIMA and support vector machines model in stock price forecasting. **Omega**, v. 33, n. 6, pp. 497 - 505, 2005.
- PALLA, G. et al. Uncovering the overlapping community structure of complex networks in nature and society. **Nature**, v. 435, n. 7043, pp. 814 - 818, 2005.
- QUILES, M. G. et al. Particle competition for complex network community detection. **Chaos: An Interdisciplinary Journal of Nonlinear Science**, v. 18, n. 3, pp. 033107, 2008.
- RANI, S. et al. Review on time series databases and recent research trend in Time Series Mining. In **Confluence The Next Generation Information Technology Summit (Confluence), 5th International Conference**, v. 1, pp. 109 - 115, 2014.
- SMALL, M. Complex networks from time series: capturing dynamics. **Circuits and Systems (ISCAS), 2013 International Symposium on**, v. 1, pp. 2509 - 2512, 2013.
- TAIEB, S. B.; ATIYA, A. F. A Bias and Variance Analysis for Multistep-Ahead Time Series Forecasting. **IEEE Transactions on neural networks and learning systems**, v. 27, n. 1, pp. 62 - 76, 2016.

TSENG, F.-M. et al. Fuzzy ARIMA model for forecasting the foreign exchange market. **Fuzzy Sets and Systems**, v. 118, n. 1, pp. 9 - 19, 2001.

ZENG, M. et al. Community Structure Detection in Complex Networks for Characterizing Atmospheric Boundary-Layer Wind Speed Time Series. **In Intelligent Control and Automation (WCICA), 2016 12th World Congress on**, v. 1, pp. 2660 - 2665, 2016.

ZHANG, G. P. Time series forecasting using a hybrid ARIMA and neural network model. **Neurocomputing**, v. 50, pp. 159 - 175, 2003.